MICHIGAN STATE UNIVERSITY LIBRARIES
3 1293 00913 9431

This is to certify that the

dissertation entitled

HYDRAULIC ANALYSIS OF SURFACE IRRIGATION SYSTEMS USING THE FINITE ELEMENT METHOD

presented by

Walid Hani Shayya

has been accepted towards fulfillment
of the requirements for

Ph.D. degree in Agricultural Engineering

Major professor

Date 9/17/91

MSU is an Affirmative Action/Equal Opportunity Institution

O-12771

LIBRARY
Michigan State
University

PLACE IN RETURN BOX to remove this checkout from your record.
TO AVOID FINES return on or before date due.

DATE DUE	DATE DUE	DATE DUE
NOV 16 1997		

MSU is An Affirmative Action/Equal Opportunity Institution
c:\circ\datedue.pm3-p.1

HYDRAULIC ANALYSIS OF SURFACE IRRIGATION SYSTEMS USING THE FINITE ELEMENT METHOD

By

Walid Hani Shayya

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in

Agricultural Engineering

Department of Agricultural Engineering

1991

ABSTRACT

HYDRAULIC ANALYSIS OF SURFACE IRRIGATION SYSTEMS USING THE FINITE ELEMENT METHOD

By

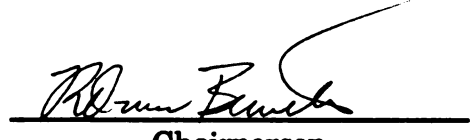
Walid Hani Shayya

A mathematical formulation for the hydraulic analysis of flow conditions in furrow and border irrigation systems is presented in this research study. The methodology is based on applying the one-dimensional Galerkin formulation of the finite element method to the numerical solution of the hydrodynamic or the so-called Saint-Venant equations. Numerical developments of the complete and simplified forms of the hydrodynamic equations were prepared using both linear and quadratic one-dimensional finite element forms of these equations. The studied models include the hydrodynamic, zero-inertia, and kinematic wave models. A general one-dimensional surface irrigation computer model (FE-SURFDSGN) was developed based on this formulation. This computer model simulates the various phases of flow in border and furrow irrigation systems using the hydrodynamic, zero-inertia, and kinematic wave models. Currently, only the kinematic wave finite element analysis is fully operational for a complete irrigation cycle in the present version of the computer model. The

Kostiakov-Lewis equation was used as the infiltration function in this development even though both the mathematical development and the developed finite element model allow for utilization of any other infiltration function. Actual field measurements were utilized to validate FE-SURFDSGN. These data were taken from previous studies that were conducted in Colorado and Idaho. Although the computer model is still in the developmental stage, its application to the simulation of the various phases of flow in surface irrigation systems is very reasonable as demonstrated through the various runs that were conducted. The results of this research work indicate that the finite element method provides accurate simulation of the flow conditions in both border and furrow irrigation systems. These results also suggest that the method developed through this research can be used as an effective tool for the hydraulic analysis of flow conditions in surface irrigation systems.

Approved :

Major Professor



Chairperson

Copyright by

WALID HANI SHAYYA

1991

DEDICATION

I would like to dedicate this research work to my parents, Hani and Salwa, for their love, devotion, tremendous support, and belief in me. My appreciation to them is beyond words. They always encouraged me to pursue my dreams and never failed to extend their help. Even though they had no clue about what I was truly working on, without their encouragement and support this research work would have not been realized.

I would also like to dedicate this work to my best friend and closest companion Cathy Cullen. Her loving support, caring, patience, understanding, and friendship were an important source of strength. Her continued encouragement, assistance, and personal support made the completion of this study a reality. I deeply appreciate all that she has done with and for me.

ACKNOWLEDGMENTS

I would like first to take this opportunity to thank my major professor, Dr. Vincent Bralts, for his constant encouragement, continued moral and financial support, genuine help, and excellent guidance. His efforts in making this learning experience worthwhile are deeply appreciated and his sincere belief in me kept me going.

Special thanks are reserved to my guidance committee members, Dr. Donald Edwards, Dr. Raymond Kunze, Dr. Larry Segerlind, and Dr. David Wiggert. The valuable suggestions and advice extended by both Dr. Edwards and Dr. Raymond Kunze brought this study to a successful completion. A particular debt of gratitude is owed to Dr. Larry Segerlind for warmly sharing his own experience as a teacher, researcher, and learner and for providing an opportunity to put complex numerical theory into practice. The constructive criticism and valuable suggestions and assistance of Dr. David Wiggert are very much appreciated.

My appreciation is also extended to Dr. Charles Cress who originally served on my committee. I would also like to express my sincere gratitude to Dr. Robert von Bernuth, the chairman of our department, who took the time to read and correct my dissertation. His helpful suggestions and advice were critical in making this study successful.

Special thanks are extended to the faculty, staff, and graduate students of the Department of Agricultural Engineering at Michigan State University, a very special group that I have had the privilege of being one of its members.

Finally, I would like to express my deepest appreciation to the Hariri Foundation and particularly to its founder Mr. Rafiq Hariri. Their financial support gave me the opportunity to come to this country and further my education. It is very difficult to put into words my genuine appreciation to the Foundation. However, I would like to try and express my deep gratitude and admiration to them, a feeling that is shared by thousands of other scholars that were privileged by receiving the help and support of the Hariri Foundation. Without their support, none of this would have been possible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

$[A_G]$ = $([C_G]+\theta\Delta t[K_G])$

α = space weighting coefficient of the nonstandard Galerkin formulation

A = cross-sectional area of flow

a = empirical parameter in the Kostiakov-Lewis infiltration equation

A = soil parameter in the Philip's infiltration equation

β_1 = furrow hydraulic section parameter

β_2 = furrow hydraulic section exponent

$[C_G]$ = global capacitance matrix

c = celerity wave

C = Chezy coefficient

$[C]$ = element capacitance matrix

δt or Δt = time step

dV_s = change in the stored volume within the element during time dt

δx = length of the cell

D = drag coefficient (AS_f)

e = empirical parameter in the U.S. Soil Conservation Service infiltration equation

f_0 = empirical parameter in the Kostiakov-Lewis infiltration equation

F_f = force due to friction in a fluid element

$\{F_G^*\}$ = $\Delta t((1-\theta)\{F_G\}_a + \theta\{F_G\}_b)$

$\{F_G\}$ = global force vector

$\{F_G\}_a$ = global force vector at time a

$\{F_G\}_b$ = global force vector at time b

F_p = pressure force

F_{wx} = force due to the weight of a fluid element in the direction of flow

f = Darcy-Weisbach's roughness coefficient

f = empirical parameter in the U.S. Soil Conservation Service infiltration equation

$\{F\}$ = element force vector

γ_1 = furrow flow geometry empirical fitting constant

γ_2 = furrow flow geometry empirical fitting exponent

γ = specific weight of water

g = acceleration due to gravity

g = empirical parameter in the U.S. Soil Conservation Service infiltration equation

h' = suction at the wetting front in the soil in infiltration

i_f = final infiltration rate

I = infiltration rate per unit length

k_1 = coefficient in the general uniform flow equation

k_2 = coefficient in the general uniform flow equation

$[K_G]$ = global stiffness matrix

$[K]$ = element stiffness matrix

k = empirical parameter in the Kostiakov-Lewis infiltration equation

K = hydraulic conductivity in wetted zone

L = length of the element

m = mass of the fluid element

N_i = shape function at node i

n = Manning's roughness coefficient

$[P_G]$ = $([C_G]-(1-\theta)\Delta t[K_G])$

$\{\dot{\Phi}_G\}$ = global vector of the derivatives of the unknowns, $\dot{A}$ and $\dot{Q}$, with respect to time

$\{\Phi_G\}$ = global vector of unknowns, A and Q

$\{\dot{\Phi}\}$ = element vector of the derivatives of unknowns, $\dot{A}$ and $\dot{Q}$, with respect to time

$\{\Phi\}$ = element vector of unknowns

$\{\Phi\}_a$ = element vector of unknowns at time a

$\{\Phi\}_b$ = element vector of unknowns at time b

ϕ = space-averaging coefficient

P = wetted perimeter

Q_0 = inlet flow rate into a furrow or inlet flow rate per unit width into a border

Q = flow rate

$\{R^{(e)}\}$ = element residual vector

$\{R_G\}$ = global residual vector

ρ_1 = furrow hydraulic section parameter

ρ_2 = furrow hydraulic section exponent

ρ = density of water

R = hydraulic radius which represents the ratio of the cross-sectional area of flow, A, to the wetted perimeter, P

S_0 = slope of the channel bed

S_f = slope of energy line or friction slope

σ_1 = furrow flow geometry empirical fitting constant

σ_2 = furrow flow geometry empirical fitting exponent

s = integrand of the direction of flow

S = sorptivity which is a soil parameter in the Philip's equation

t_f = time at which the final infiltration rate is reached

θ_s = saturated soil moisture content

θ_i = initial soil moisture content

θ = time weighting coefficient

t_s = time at which the advancing front reaches distance s

τ = infiltration opportunity time

t = time

T = top width of flow cross section

V_{in} = total inflow during a time step, dt

V_I = total infiltration during a time step, dt

V_{out} = total outflow during a time step, dt

$V_y(t)$ = surface volume of water

$V_z(t)$ = infiltrated volume of water

v = flow velocity

$W(x)$ = weighting function

ω = Escoffier stage variable

x = distance along the direction of flow

y = flow depth

Z = infiltration volume per unit length

I. INTRODUCTION

Surface irrigation is the most important method of irrigation in the world. Kay (1986) reported that surface irrigation still covers more than 95% of all irrigated land worldwide in spite of the popularity of sprinkler and trickle irrigation systems. In the United States, around 62% of irrigated land is currently under surface irrigation (Bajwa et al., 1987). These figures reflect the importance of this type of irrigation practice and the need for additional improvements in both the design and operation of surface irrigation systems. As Bassett et al. (1980) described it, this process may be accomplished through the "skillful combination of experience and thorough understanding of the processes involved". Recent theoretical developments in the area of surface irrigation system design should serve as an excellent means for better understanding the physical processes involved in the hydraulic design and field evaluation of surface irrigation systems. These developments involve the application of numerical analyses to the modeling of surface irrigation systems which will in turn increase the accuracy of the design and improve the performance of the system with the least incurred cost.

The earliest models of surface irrigation systems dealt only with water advance down the border or furrow. Among the many early works that were based on this approach are the developments by Lewis and Milne (1938), Hall (1956), Philip and Farrell (1964), and Hart et al. (1968). These approaches were based on the principle of conservation of mass together with assumptions regarding average depth of surface flow. These have resulted in assumptions that water at the upper end of the border is at normal depth and that both

surface and subsurface water profiles are of a monomial power law of a fixed or assumed degree. The application of these assumptions produced acceptable results at times and fundamental errors at other times (Strelkoff and Katopodes, 1977a).

An alternative approach is to numerically solve the two partial differential equations that govern the unsteady water flow conditions in open channels. These partial differential equations, or the so-called Saint-Venant equations, consist of an equation of continuity and an equation of motion. The latter equations can be developed based on applying the principles of conservation of mass and momentum or energy to a controlled volume of flow in a channel. The hydrodynamic equations, which have been studied since the turn of century, describe the unsteady spatially varied flow of water across the soil surface. With the recent rapid advancement in numerical techniques and the computing potentials of computers in general and microcomputers in particular, the possibilities and alternatives of solving surface irrigation problems based on the numerical approximations of the hydrodynamic equations look increasingly promising. In recent years, the hydrodynamic equations were applied to the analysis of overland flow in watershed hydrology, open channel flow, and surface irrigation. These equations were utilized either in complete form or after implementing some simplifying assumptions which would result in the zero-inertia and the kinematic wave models.

A. Scope and Objectives

The general scope of this research work was to develop finite element Galerkin formulations of the complete and simplified forms of the hydrodynamic equations (equations that were developed based on the

application of the conservation of mass and momentum principles) as applied to the simulation of flow conditions in surface irrigation systems. The physical flow conditions in border and furrow irrigation systems were represented by a mathematical model that could incorporate the aforementioned complete and simplified forms of the hydrodynamic equations with a finite element numerical solution procedure, initial and boundary conditions, and other necessary physical parameters. The specific objectives of this research were

1. To develop a finite element solution procedure of the Saint-Venant equations for the hydraulic analysis of surface irrigation systems.
2. To create a general solution approach that will accommodate the available mathematical models of the Saint-Venant equations in the analysis of surface irrigation systems.
3. To develop an approach to easily incorporate the varying boundary conditions of the advance, ponding, depletion, and recession phases of surface irrigation into the solution process with minimal arbitrary or experimental parameters.
4. To develop a computer model that will utilize the above mathematical concepts for the hydraulic analysis of flow conditions in border and furrow irrigation systems.
5. To numerically evaluate and compare the results of the finite element model to actual field measurements from existing surface irrigation systems.

II. REVIEW OF THEORY AND LITERATURE

The analysis of surface irrigation systems is a process that involves many parameters. One approach for the design is to establish analytical relationships among the various factors affecting the flow conditions in surface irrigation problems. These factors include length of the field, inflow time, inflow rate, surface runoff, deep percolation, application depth, soil intake characteristics, and land slope. This approach has been utilized in the design of surface irrigation systems for many years. However, it is incapable of defining or estimating the spatially distributed flow conditions in surface irrigation systems.

The process of overland flow across a soil surface is both spatially varied and unsteady. The principles of continuity of mass and continuity of momentum or energy can be applied to describe overland flow conditions in irrigation systems. Applying these concepts will result in the so-called hydrodynamic equations that are commonly known as the Saint-Venant Equations. These equations, which have been studied since the turn of century, describe the unsteady spatially varied flow of water across soil surface. Originally, graphical solutions were utilized for the solution of the above equations. However, the application of these equations was limited to restricted, simplified cases (Strelkoff, 1970). With the advent of high speed digital computers, numerical approximations of the Saint-Venant equations became feasible.

This review expounds the basic background of surface irrigation as related to surface flow description and infiltration. Also, it elucidates the

theory in the literature on the numerical solution procedures of the Saint-Venant equations, particularly those utilizing the finite difference and finite element methods as well the method of characteristics to describe the spatially distributed flow conditions in surface irrigation systems.

A. Basic Background of Surface Irrigation

The basic modeling of surface irrigation systems involves many factors that are generally common to the available types of systems. These types include borders, furrows, and basins (Walker and Skogerboe, 1986). The hydraulic flow characteristics of basins is a special case of the border flow (Bassett et al., 1980). For this reason, only the hydraulics of borders and furrows will be discussed in this section. The geometry of flow in both furrow and border irrigation systems are generally similar. The major difference arises from the width of the channel which is narrow for furrows and usually wide for borders. The width of the border strip is generally wide enough to ignore the contribution of channel walls to both flow retardate and infiltration, an assumption which is not applicable to irrigation furrows (Bassett et al., 1980).

The different types of surface irrigation systems involve numerous physical characteristics that may be defined by common terminology. This terminology, together with the basic concepts of surface irrigation systems, is well documented in the literature. This section will review the basic concepts of surface irrigation systems without much detail. The reader is referred to Walker and Skogerboe (1986), Bassett et al. (1980), and Kay (1988) for more in-depth description of the different types of surface irrigation systems.

1. Flow Description

The flow of free water in surface irrigation systems is gradually varied and unsteady. The infiltration of water into the soil seems to dominate this hydraulic characteristic of flow. Bassett et al. (1980) reported that the state of water flow in surface irrigation systems is mostly turbulent or transitional, a flow regime that is characterized by a Reynold's number around or above 1000 even though numbers well below 1000 are frequently encountered. The Reynold's number is a dimensionless ratio of the inertial to viscous forces (Binder, 1973).

The flow regime of water in surface irrigation systems is usually sub-critical. This is characterized by Froude numbers well below unity. The Froude number is a dimensionless ratio of the inertial to the gravitational forces. Bassett et al. (1980) reported that critical and super-critical flow regimes might occur just behind the wetting front in the advance phase of both sloping borders and furrows, and just ahead of the drying front during the recession phase.

2. Flow Phases

There are four phases of flow in a typical surface irrigation cycle. These include the advance, ponding, depletion, and recession phases. The advance phase represents the first portion of irrigation time during which water advances down the furrow or border. The ponding phase starts at the end of the advance phase when the advancing front reaches the end of the furrow or border. This phase extends till water is shut-off at the inlet boundary. The duration of this phase is zero if the inlet water is turned off before water reaches the end of the furrow or border. The next phase of irrigation represents the portion of the total irrigation time between inlet flow shut-off

and the beginning of water recession at the inlet boundary. The final phase of flow is the recession phase which represents the portion of the total irrigation time between the beginning of water recession at the inlet boundary and the complete disappearance of water from the furrow or border.

B. Surface Irrigation Infiltration

The simulation of flow in surface irrigation systems relies on the knowledge of the hydraulic characteristics and infiltration (Walker and Humpherys, 1983, and Strelkoff and Souza, 1984). However, the most critical step in studying the hydraulics and distribution of water in surface irrigation systems is to establish a reasonable estimate of the infiltration function. Clemmens (1981) described the determination of the infiltration characteristics of the soil as "the biggest stumbling block in accurately describing or predicting the irrigation process". Infiltration has always been referred to as the impeding factor to improving both the design and performance of surface irrigation systems (Elliott et al., 1982a, and Elliott et al., 1983b).

1. Infiltration Equations

There are many infiltration equations that have been developed throughout the years. Walker and Skogerboe (1987) classified these equations into three general categories. These include theoretically, physically, and empirically based equations. A good example of an infiltration equation that was developed based on the single-phase flow solution of the one-dimensional Darcy equation is the Richard's equation (Walker and Skogerboe, 1987). This equation has the form

$$\frac{\partial \theta}{\partial t} = \frac{\partial \left[K(\theta) \left(\frac{\partial h}{\partial z} - 1 \right) \right]}{\partial z} \qquad [2.1]$$

where θ is the soil moisture content on volume basis, h is the total pressure head, K is the hydraulic conductivity, z is the vertical distance downward from the soil surface, and t is the infiltration opportunity time.

The Green and Ampt equation is a good example of the physically based equations. It was developed based on the assumption that soil could be modeled as a bundle of capillary tubes (Hillel, 1980). The Green and Ampt equation has the form (Walker and Skogerboe, 1987)

$$I = K \left[1 + \frac{(\theta_s - \theta_i) h'}{Z} \right] \qquad [2.2]$$

where I is the infiltration rate, θ_s is the saturated soil moisture content, θ_i is the initial soil moisture content, h' is the suction at the wetting front in the soil, K is the hydraulic conductivity in the wetted zone, and Z is the cumulative infiltration depth at any specific point in the space dimension.

Infiltration is a complex physical process that is very difficult to characterize in irrigated fields where anistropic and heterogeneous conditions usually prevail. This leads to the conclusion that an empirical approach to assess this process is more practical than a purely theoretical approach (Blair and Smerdon, 1987).

There are many empirical infiltration equations that were developed in the literature. These empirical infiltration equations have always been expressed in either exponential or power form. However, the infiltration equations in power forms have been widely adopted to estimate infiltration in surface irrigation problems because of the simplicity and practicality of these equations (Fok, 1967).

Empirical infiltration equations are the result of fitting observed infiltration data to explicit time-dependent equations. Many infiltration equations fall under this category. A good example of these equations are the Kostiakov and the Kostiakov-Lewis equations. The Kostiakov equation is one of the earliest equations of infiltration (Hillel, 1980). This equation has the form

$$I = akt^{(a-1)} \qquad [2.3]$$

where t is the infiltration opportunity time and a and k are two empirical parameters obtained from infiltration tests in the field. The major draw back of this equation is that the infiltration rate approaches zero at long times. Hence, it is more applicable to horizontal rather than vertical infiltration. The Kostiakov-Lewis equation has the same form as the Kostiakov equation with an added term to correct the latter problem of the Kostiakov equation (James, 1988). This results in

$$I = akt^{(a-1)} + f_0 \qquad [2.4]$$

where f_0 is an empirical parameter that represents the infiltration rate as the infiltration opportunity time, t, becomes considerably large. Elliott and Walker (1980 and 1982) used the Kostiakov-Lewis function which has the additional term for the asymptotic long-time infiltration. Their results suggested that the Kostiakov-Lewis infiltration equation is highly effective in predicting infiltration if a steady state infiltration rate can be assessed.

Clemmens (1981) suggested using a two-branch function of the Kostiakov equation where the cumulative infiltration depth, Z, is represented by two equations. The first equation applies before the steady state infiltration is reached while the second applies after. Clemmens' proposed equations for the cumulative infiltration have the form

$$Z = kt^a \qquad \text{for} \quad t \le t_f$$

$$= kt_f^a + i_f(t - t_f) \qquad \text{for} \quad t > t_f \qquad [2.5]$$

where i_f is the final infiltration rate and t_f is the time at which the final infiltration rate is reached.

Philip (1957a and 1957b) developed an equation similar to the Kostiakov-Lewis equation but with more physical significance. His equation has the form

$$Z = St^{0.5} + At \qquad [2.6]$$

where S is the sorptivity. The parameter A was defined by Philip as

$$A = K_n + \int_x = K_n + \left(\frac{2}{3}K_0 - K_n\right) \qquad [2.7]$$

where K_0 and K_n are soil parameters. Kunze and Kar-Kuri (1983) and Kunze and Shayya (1990) defined the parameter A as the hydraulic conductivity of the soil and suggested the adjustment of the sorptivity term by a factor to implement this denotation. Fangmeier and Ramsey (1978) made a comparison between the Philip equation and the Kostiakov-Lewis equation. They concluded that the Philip equation provided better estimates of infiltration compared to the Kostiakov-Lewis equation. However, the coefficients in the Philip equation are more difficult to obtain.

The U.S. Soil Conservation Service (1974 and 1984) developed an equation that relates the cumulative infiltration and the opportunity time as

$$Z = et^f + g \qquad [2.8]$$

where e, f, and g are empirical parameters.

2. Parameter Estimation of Infiltration Equations

Determining infiltration parameters in the various empirical infiltration equations previously discussed is a very critical step in the design and operation of surface irrigation systems. One approach for the estimation of these parameters can be accomplished through the utilization of direct field measurements of infiltration using ring infiltrometers. However, such measurements do not reflect the actual hydraulic characteristics at the time of irrigation (Bouwer, 1957, and Blair and Trout, 1989). They can also be very time consuming and costly. Many scientists attempted to assess the variability of infiltration parameters and infiltration along furrows using infiltrometers and moisture measurements (Bautista and Wallender, 1985; Izadi and Wallender, 1985; and Bali and Wallender 1986). Brakensiek et al. (1979) discussed the application of a new infiltrometer system and its utilization in the estimation of infiltration parameters in the Green and Ampt function. Blair and Trout (1989) presented a field guide for the construction and operation of a recirculating infiltrometer, a device that can be used in the measurement of infiltration and the estimation of infiltration parameters in various infiltration functions.

Another approach for the parameter estimation of infiltration equations is to implement a numerical solution procedure of surface flow equations in conjunction with advance, storage, and inflow-outflow field measurements. Christiansen et al. (1966) and Fangmeier and Ramsey (1978) implemented the volume balance method in the estimation of infiltration parameters in the Kostiakov equation. Elliott et al. (1983a) simulated advance trajectories of water in furrows using assumed infiltration functions and the zero-inertia model. They then estimated the actual field infiltration using actual field advance trajectories and simulated advance trajectories by the zero-inertia

model. Katopodes et al. (1991) presented a procedure for estimating both infiltration parameters and soil roughness in surface irrigation systems using a linearized zero-inertia model.

In 1987, Kaytal et al. developed an infiltration equation that accounts for the two-dimensional infiltration flow conditions in furrows. Their approach was an attempt to develop infiltration functions that are applicable throughout the growing season instead of one irrigation only. The approach that they followed included the numerical solution of the Richard's infiltration equation in furrows using the finite element method. The Richard's equation was utilized in conjunction with basic soil data and furrow shape parameters in order to estimate the parameters of a power infiltration function. The latter function had two independent variables which include the top width of flow as well as the cultivation depth.

More recently, Walker and Busman (1990) discussed an approach to use a kinematic wave simulation model in conjunction with the simplex method to determine the infiltration parameters from early stages of furrow advance. Their approach was based on minimizing the differences between predicted and measured advance rates. They concluded that this procedure will provide estimates of the infiltration parameters with sufficient accuracy.

The Kostiakov and Kostiakov-Lewis infiltration equations are the most widely used infiltration equations in surface irrigation problems. The parameter-estimation of these equations was the subject of many investigations. Norum and Gray (1970) presented a method for deriving the values of the parameters in the modified Kostiakov function. Smith (1972) used surface irrigation data and a kinematic wave model to characterize the infiltration parameters in the Kostiakov-Lewis equation. Elliott et al. (1982a and 1983b) presented a method for deriving the values of the parameters in the

modified Kostiakov function using dimensionless advance curves and furrow irrigation field data. Elliott and Walker (1980 and 1982) reported on the use of two-point volume balance methodology to establish the parameters of the Kostiakov-Lewis function based on advance data measurements. Sirjani and Wallender (1989) attempted to approximate the mean and variance of the parameters in the Kostiakov-Lewis infiltration equation using the first order analysis.

One last method for estimating the infiltration parameters in various infiltration functions is to analyze the inflow-outflow measurements. The infiltration is calculated as the difference between measured water inflow and outflow from a border or furrow section. However the major disadvantage of this method is the sensitivity of the infiltration estimates to the accuracy of flow measurements (Trout and Mackey, 1988b). The accuracy of these measurements depend on many factors. These factors are affected by the flow characteristics and the geometry of furrows and borders. Bautista and Wallender (1985) reported that infiltration rate is usually greater in blocked furrows with flowing water compared to stagnant tests.

Strelkoff and Souza (1984) considered six different schemes for incorporating the variable depth effect into the computations of infiltration in furrows. They concluded that the "wetted perimeter based on local depth is the best choice of traverse length to characterize furrow intake in mathematical models of furrow irrigation". Izadi and Wallender (1985) statistically examined furrow hydraulic characteristics in space and time and related these characteristics to infiltration. Trout (1986) conducted a study to measure the effect of both overland flow velocity and furrow hydraulic parameters on infiltration. His conclusion was that furrow infiltration increases by increasing the wetted perimeter of flow. He also reported that flow velocity is inversely related to furrow infiltration rate.

To summarize, there are various methods for obtaining infiltration data of surface irrigation systems. These include the ring infiltrometer, ponding, two-point, blocked furrow, inflow/outflow, and recirculating infiltrometer methods. As outlined in this section, these procedures were utilized directly and indirectly by many scientists in the parameter estimation of the infiltration functions for surface irrigation systems. The reader is referred to the texts by James (1988) and Walker and Skogerboe (1987) for a detailed description of the various infiltration measurement devices and the procedures followed in their implementation in the field.

3. Effect of Infiltration on System Performance

Improving the efficiency of furrow irrigation systems can be attained by accurate assessment of infiltration (Elliott and Walker, 1980, and Elliott and Walker, 1982). On the other hand, accounting for the spatial variability in infiltration is very essential in evaluating the performance of surface irrigation systems (Bautista and Wallender, 1985, and Davis and Fry, 1963). Trout and Mackey (1985 and 1988a) attempted to quantify the effect of both inflow and infiltration variability on the uniformity of water application and runoff in furrows. They concluded that the consequences of inflow and infiltration variability are excessive deep percolation as well as runoff losses in some areas of the field while other areas receive inadequate amounts of water. Sirjani and Wallender (1989) conducted a study to assess the effect of temporal and spatial variability of infiltration on the performance of furrow irrigation systems. Fonteh and Podmore (1989) developed a kinematic wave furrow irrigation model using a physically based infiltration function. Their model could simulate spatially varied infiltration along furrows using geostatistics and their results seemed to be most accurate under fine textured soil conditions.

The spatial and temporal variability in infiltration seems to affect crop yield. Kemper et al. (1982) reported that yield decreases substantially by low uniformity of water distribution throughout the length of the run. They presented several treatments which can increase or decrease infiltration in order to improve uniformity. One approach for reducing infiltration rate involves the practice of surge irrigation. Surge irrigation is defined as "the intermittent application of irrigation water to furrows or borders" (Bishop et al., 1981). Surge flow creates a series of on and off inflow conditions at the inlet (Izuno and Podmore, 1986). The characterization of infiltration under surge irrigation systems was studied by many scientists (Bishop et al., 1981; Izuno et al., 1985; Kemper et al., 1988; and Samani et al., 1985). The reader is referred to these references for more details.

C. Surface Flow Equations

The principles of mass, and momentum (or energy) can be utilized to describe the flow of water over soil surface. These principles result in two first-order, nonlinear partial differential equations. The resultant differential equations approximate the spatially varied and unsteady flow conditions in open channels and surface hydrology. These same equations can also be applied to the hydraulic analysis of water flow conditions in surface irrigation systems.

1. Saint-Venant Equations

The two equations that result from applying the above principles are known as the Saint-Venant equations. The first equation results from

applying the conservation of mass principle to a control volume of flow in surface irrigation systems. It is usually referred to as the continuity equation and has the form

$$\frac{\partial A}{\partial t}+\frac{\partial Q}{\partial x}+I=0 \qquad [2.9]$$

where A is the cross sectional area of flow, Q is the flow rate, I is the infiltration rate per unit length, t is time, and x is the distance along the direction of flow. The second equation is referred to as the momentum equation and results from applying the conservation of momentum principle to the same fluid element. This equation has the form (Strelkoff, 1969)

$$S_0=S_f+\left(\frac{2Q}{A^2g}\right)\frac{\partial Q}{\partial x}+\left(1-\frac{Q^2T}{A^3g}\right)\frac{\partial y}{\partial x}+\left(\frac{1}{Ag}\right)\frac{\partial Q}{\partial t} \qquad [2.10]$$

where y is the flow depth, g is the acceleration due to gravity, T is the top width of flow cross section, S_0 is the slope of the channel bed, and S_f is the friction slope. Equation [2.10] describes unsteady non-uniform flow conditions. The last term in the above equation cancels out if steady non-uniform flow conditions are presumed. Of note, with steady uniform flow assumptions, the last three terms of [2.10] mutually cancel.

Another approach can be followed to develop an equation which can replace [2.10]. This development can be based on the principle of conservation of energy. However, both the approach and the resultant equation will be inherently different from the equation developed based on the momentum approach (Martin and Wiggert, 1975; Yen, 1973; and Strelkoff, 1969).

Brutsaert (1971) verified the Saint-Venant equations experimentally for open channel flow conditions. His approach included the comparison of the results from the numerical solution of these equations to the flow

measurements of a physical model. The solution procedure encompassed the implementation of a finite difference computational algorithm with implicit and explicit differences at interior cells and boundaries, respectively. He concluded that the Saint-Venant equations represented the physical flow system reasonably well. However, these positive results are only attainable under proper mathematical conditions and appropriate description of boundary conditions, hydraulic resistance to flow, and channel parameters. The formulation and verification of the Saint-Venant equations for other related problems were the concern of many scientists. These analyses were reported in many references including Morgali and Linsley (1965), Kruger and Bassett (1965), Wooding (1965b), Brakensiek (1966), Chen and Hansen (1966), Ragan (1966), Liggett and Woolhiser (1967), Strelkoff (1969), Brutsaert (1971), and Katopodes and Schamber (1983). These references concluded basically that the application of the Saint-Venant equations to surface flow problems produced good results when the various assumptions implemented in the development of these equations were not violated. Many of these assumptions were discussed in details in the journal articles by Strelkoff (1969) and Yen (1973).

2. Uniform Flow Equations

There are three popular equations for establishing relationships among flow rate, slope of channel bed, and channel geometry. These equations are essential for defining the friction slope in the momentum equation, [2.10]. Originally, these equations were developed for the analysis of uniform flow conditions in open channels. Thus, these equations are frequently referred to as uniform flow equations. However, these equations may be used to approximate the friction slope for nonuniform, unsteady, turbulent flow conditions at a given instant (Morgali and Linsley, 1965). The first equation is

known as the Chezy equation after it was introduced by a French engineer of that name in 1768 (Henderson, 1966). The development of the Chezy equation was based on the dimensional analysis of the resistance equation with the assumption that flow conditions are uniform. The Chezy equation has the form

$$Q = CAR^{1/2}S_f^{1/2} \qquad [2.11]$$

where C is the Chezy coefficient and R is the hydraulic radius which represents the ratio of the area of flow to the wetted perimeter.

A more practical formula was developed in 1889 by an Irish engineer by the name of Robert Manning (Chow, 1959). The Manning equation is widely used for steady flow conditions of incompressible fluids in prismatic open channels (Streeter and Wylie, 1979). The Manning equation is an empirical equation with the form

$$Q = \frac{A}{n}R^{2/3}S_f^{1/2} \qquad [2.12]$$

where n is Manning's roughness coefficient. The Manning equation is very popular in many Western countries (Henderson, 1966). Besides, it has well documented roughness coefficients that were developed over the years.

The third equation is the Darcy-Weisbach equation that was originally developed for pipe-flow. This equation can be expressed as (Brater and King, 1976)

$$Q = \sqrt{\frac{8g}{f}}AR^{1/2}S_f^{1/2} \qquad [2.13]$$

where f is Darcy-Weisbach's roughness coefficient.

The relationships between C, n, and f can be summarized by

$$C = \frac{1}{n} R^{1/6} \qquad [2.14]$$

and

$$f = \frac{8gn^2}{R^{1/3}} \qquad [2.15]$$

These roughness coefficients change with varying conditions (Walker and Skogerboe, 1987). A complete discussion of these flow equations can be found in Chow (1959), Henderson (1966), Streeter and Wylie (1979), White (1979), and Bassett et al. (1980).

D. Surface Irrigation Models

There are several techniques that have been developed over the years to model surface irrigation processes mathematically. Most of these techniques are based on the application of the principles of conservation of mass and momentum which are referred to as the Saint-Venant equations. The Saint-Venant equations are applied either in complete or simplified forms. This results in four general models, one complete form and three simplified forms. The application of the simplified forms of the Saint-Venant equations to the analysis of surface irrigation systems offer simpler and faster surface irrigation modeling approaches. However, the accuracy of the results is reduced either slightly or appreciably depending on the level of simplifications.

Strelkoff (1970) classified the numerical procedures that may be implemented in the solution of the complete or simplified first-order nonlinear equations of hydrodynamic models into two categories. The first approach is to convert the original system of partial differential equations into an equivalent

system of ordinary differential equations. This transformation is accomplished by changing the independent variables through the introduction of an alternate coordinate system that is inclined at an angle to the original space-time system (Strelkoff, 1970). The resulting ordinary system of differential equations is then solved numerically using methods such as the finite difference method. The resultant equations are algebraic instead of being ordinary differential equations. This approach is referred to as the method of characteristics. The theoretical basis for the method of characteristics is reviewed in details by Streeter and Wylie (1967), Strelkoff (1970), Liggett and Cunge (1975), and Wylie and Streeter (1983). The second approach involves the implementation of one of many available numerical solution schemes which directly replace the first-order nonlinear partial differential equations by difference quotients. This approach results in a system of algebraic equations instead of the original partial differential equations.

1. Hydrodynamic Models

Mathematical models that result from applying the Saint-Venant equations to the analysis of surface flow problems without any simplifications are referred to as hydrodynamic models. These models have high potential to be very accurate in predicting flow conditions in surface flow problems and surface irrigation systems in particular. However, these models depend heavily on the accuracy of the provided input information (Strelkoff and Katopodes, 1977a). Bassett et al. (1980) described the roughness and infiltration parameters as the two main sources of input error which affect the precision of such models. Besides, the complex, delicate nature of the

hydrodynamic models represents their major drawback. This results in lengthy computation times and makes such models expensive to operate when it comes to computer cost (Bassett, 1980, and James, 1988).

Over the years, hydrodynamic models were developed by many scientists based on the method of characteristics and applied to the analysis of flow conditions in open channels and surface irrigation systems. Bassett (1972) developed a model of water advance in border irrigation by solving both the continuity and momentum equations in their complete form for unsteady spatially varied flow conditions using the method of characteristics as described by Streeter and Wylie (1967). In his development, Bassett utilized a fixed rectangular grid in the space-time plane. He used the Kostiakov infiltration function and applied the Chezy equation to the evaluation of friction slope. He concluded that the developed model predicts flow conditions in borders with acceptable accuracy. This conclusion was based on the comparison of model results to actual laboratory and field measurements. Kincaid et al. (1972) and Sakkas and Strelkoff (1974) also applied the method of characteristics to the solution of the complete hydrodynamic equations for border irrigation advance. They reported that their developed models predicted the flow conditions in borders reasonably well. However, their developments covered only the advance phase of irrigation instead of the complete phases of surface irrigation. However, they claimed that their approaches can be extended to the simulation of the recession phase with proper boundary conditions.

Bassett and Fitzsimmons (1976) extended the work of Bassett (1972) by presenting a mathematical model for the analysis of the complete phases of the irrigation process in borders. Their model was based on the same approach, i.e, the application of the equations of continuity and momentum in their complete forms using the method of characteristics. They reported good

results in general. However, they have also reported some problems of instability and the high execution costs which were associated with the required extensive computer time. Katopodes and Strelkoff (1977a) applied also the method of characteristics to the approximation of the hydrodynamic equations in borders. In their study, a comparison of the results from their developed model and actual field measurements were made and the associated accuracy and costs of model execution were assessed. They concluded that the solution results were correct to second order accuracy.

The major disadvantage of the method of characteristics usually lies in the total number of unknowns. By implementing this method, the two nonlinear partial differential equations represented by equations [2.9] and [2.10] are transformed into four ordinary differential equations (Katopodes and Strelkoff, 1977a). These equations are

$$\frac{d(v+\omega)}{dt}=g\left[S_0-S_f+\frac{I(v-c)}{Ag}\right] \qquad [2.16]$$

along

$$\frac{dx}{dt}=v+c \qquad [2.17]$$

and

$$\frac{d(v-\omega)}{dt}=g\left[S_0-S_f+\frac{I(v+c)}{Ag}\right] \qquad [2.18]$$

along

$$\frac{dx}{dt}=v-c \qquad [2.19]$$

where v is the flow velocity, ω is the Escoffier stage variable (Escoffier and Boyd, 1962), and c is the celerity wave. The celerity and Escoffier stage variables are defined as

$$c = \sqrt{\frac{gA}{T}} \qquad [2.20]$$

and

$$\omega = \int_0^y \frac{g}{c} dy \qquad [2.21]$$

where y is the flow depth. The Escoffier stage variable, ω, reduces to $2c$ when the above method is applied to the hydraulic analysis of borders.

The ordinary differential equations ([2.16] to [2.19]) that were the result of applying the method of characteristics to the complete first order, nonlinear hydrodynamic equations were never applied successfully to the simulation of flow conditions in furrow irrigation systems (Walker and Skogerboe, 1987).

The finite difference schemes for the direct solution of the Saint-Venant equations are preferred by many scientists. Using these techniques, the hydrodynamic equations are solved at a finite number of grid points in the space-time plane (Liggett and Cunge, 1975). Two basic types of finite-difference schemes are usually used in the literature (Strelkoff, 1970, and Liggett and Cunge, 1975). The first represents the explicit schemes where the algebraic equations are arranged to be solved for one unknown at a time. These numerical schemes, although simple, are unstable and usually require excessive computation times due to the need for the selection of small time steps (Liggett and Woolhiser, 1967, and Strelkoff, 1970). This prompted many investigators to avoid using these schemes, a trend which is especially true in recent developments. On the other hand, the implicit schemes solve for a

group of unknowns simultaneously rather than one at a time. These numerical schemes permit larger time steps but require the solution of a system of nonlinear simultaneous equations at each time step. They are usually more desirable in the direct solution of the Saint-Venant equations because of their stability and high accuracy. Various analyses of stability and accuracy of both explicit and implicit finite difference schemes, as applied to the direct solution of the complete continuity and momentum equations, can be found in Liggett and Woolhiser (1967), Strelkoff (1970), Price (1974), and Liggett and Cunge (1975).

A number of scientists have reported on the successful application of implicit finite difference numerical schemes to the direct solution of the Saint-Venant equations in open channels. The latest development was reported by Swain and Chin (1990) for modeling unsteady flow in regulated open channels. Their development would also allow for the simulation of hydraulic structures, an option that is not available in many current open channel models.

Walker and Skogerboe (1987) presented an implicit finite difference scheme for the direct solution of the continuity and momentum equations in furrows. Their scheme was based on the Eulerian integration approach, a numerical procedure which approximates the hydrodynamic equations using the concept of multi-cell deforming control volume. Using this approach, the continuity and momentum equations as represented by [2.9] and [2.10], respectively, become

$$[\theta(Q_L - Q_R) + (1-\theta)(Q_J - Q_M)]\delta t - [\phi(A_L + Z_L - A_J - Z_J)]\delta x - [(1-\phi)(A_R + Z_R - A_M - Z_M)]\delta x = 0 \qquad [2.22]$$

and

$$\frac{1}{g}\left[\frac{\phi(Q_L-Q_J)+(1-\phi)(Q_R-Q_M)}{\delta t}\right]+\theta\left[\frac{(P+Q^2/Ag)_R-(P+Q^2/Ag)_L}{\delta x}\right]$$
$$+(1-\theta)\left[\frac{(P+Q^2/Ag)_M-(P+Q^2/Ag)_J}{\delta x}\right]$$
$$-S_0\theta[\phi A_L+(1-\phi)A_R]-S_0(1-\theta)[\phi A_J+(1-\phi)A_M]$$
$$+\theta[\phi D_L+(1-\phi)D_R]+(1-\theta)[\phi D_J+(1-\phi)D_M]=0 \qquad [2.23]$$

where A is the cross sectional area of flow, Q is the flow rate across the respective cell boundaries, Z is the infiltrated volume per unit length, δt is the time step, δx is the length of the cell, θ is the time-averaging coefficient to account for the nonlinear variation in the flow profile over time, ϕ is the space-averaging coefficient to account for the nonlinear variation in the flow profile over the cell length, D is the drag (AS_f), and P is the pressure force. The subscripts J and M in [2.23] represent the left and right cell boundaries, respectively, at time t_{i-1} while the subscripts L and R represent the left and right cell boundaries, respectively, at time t_i. Walker and Skogerboe (1987) presented a detailed description of their mathematical development from its inception to the final form as represented by the nonlinear algebraic equations in [2.22] and [2.23]. They discussed the implementation and applications of this implicit finite difference scheme to furrow and border irrigation systems. Also, they discussed the basic advantages of this procedure over the method of characteristics approach. The primary advantage relates to the number of computational unknowns which is half as much in the former method compared to the latter method.

As to the application of the finite element method to the numerical solution of hydrodynamic equations, Katopodes (1984) developed a dissipative Galerkin scheme for the solution of these equations as applied to open channels with discontinuous flow. All the energy diffusing terms were

neglected in the hydrodynamic equations and his development was restricted to one-dimensional flow in a prismatic channel with a rectangular cross section. Katopodes reported excellent results using the dissipative Galerkin scheme and was very optimistic about the utility of the finite element method in computing surges and shocks in open channels. Akanbi and Katopodes (1988) extended the work of Katopodes (1984) to two-dimensional overland flow problems. Their development was for the solution of flood wave propagation on initially dry land. They were very successful in solving the two dimensional shallow-water equations with a dissipative Galerkin scheme which involved a deforming and moving computational grid.

To conclude this section, it is clear that the one-dimensional gradually varied hydrodynamic equations together with appropriate initial and boundary conditions are capable of high accuracy. However, the numerical solution of either the two partial differential motion equations or the four ordinary differential equations developed from applying the method of characteristics is costly due to excessive requirement of computations which can be time-consuming. This makes the solution of the simplified forms of the Saint-Venant equations more desirable especially when such models produce acceptable results (Miller and Cunge, 1975). The models that are based on the complete hydrodynamic equations are not usually intended for the design or specific applications because of the high operational costs. However, such models can be used as standards of comparison for the simplified models which have the potential to produce good results at minimal cost.

2. Zero-Inertia Models

The zero-inertia model is a simplified form of the hydrodynamic model. The continuity equation ([2.9]) is kept unchanged while the acceleration and

inertial terms in the momentum equation ([2.10]) are ignored based on the assumption that such terms are negligible in most flow conditions of surface irrigation systems (Strelkoff and Katopodes, 1977a). This assumption results in the following simplified momentum equation

$$\frac{\partial y}{\partial x} = S_f - S_0 \qquad [2.24]$$

The zero-inertia model was first proposed by Brakensiek et al. (1966) in the context of flood routing where he described the process of propagation of flood hydrographs through the watershed channel system. Brakensiek (1966) discussed the appropriateness of the assumptions of the zero-inertia model. His results revealed that the zero-inertia model produced excellent results compared to the full hydrodynamic model in the regions of slowly accelerating flow conditions in watershed channel systems. More recently, the accuracy of the zero-inertia model in channel routing was the center of attention of many investigations including those of Ponce et al. (1978), Ponce and Theurer (1982), and Ponce (1987).

Strelkoff (1972) and Katopodes (1974) were the first to apply the zero-inertia model to surface irrigation. Their applications were prepared for borders. The solution technique that they followed was similar to the nonlinear shooting technique which was utilized by Brakensiek et al. (1966). However, the latter solution approach had some convergence problems towards the end of the depletion phase. Strelkoff and Katopodes (1977a) presented a new numerical approach for the solution of the zero-inertia model. The numerical approach they developed precluded the problems that were encountered in their previous work. Their model examined the process of irrigation as a deforming control volume with upper and lower boundaries. This numerical approach included the solution of a system of nonlinear

equations for each time step by first linearizing and solving the system using the double-sweep technique as described by Liggett and Cunge (1975). The selected time step was constant. Their linearization technique produced very realistic results for the prediction of both advance and recession phases in border irrigation systems.

The zero-inertia model developed by Strelkoff and Katopodes (1977a) was the center of attention for many scientists. Several investigations were conducted to refine and verify the model against field measurements. Strelkoff and Katopodes (1977b) discussed establishing appropriate boundary conditions for the zero-inertia model when a free over-fall downstream boundary occurs. Clemmens and Fangmeier (1978) discussed the ways to improve the numerical solution of the model when diked-end conditions occur at the downstream boundary following the completion of the advance phase. Clemmens (1979) verified the zero-inertia model for advance and recession in blocked-end borders with actual field measurements. He concluded that the agreement was generally good. However, he reiterated the notion that successful application of the zero-inertia, or any other mathematical model of surface irrigation, is dependent on the accuracy of infiltration and soil roughness measurements. Elliott et al. (1982b) developed a mathematical model to simulate the advance phase of flow in furrow irrigation based on the zero-inertia assumptions. The approach they followed in their development was similar to that of Strelkoff and Katopodes (1977a) which included the integration of the governing equations over finite cells in the space-time plane. The cross sectional area of flow and wetted perimeter were related to flow depth through the implementation of power curve relationships. Their results revealed that the zero-inertia model simulated the hydraulics of advance phase of furrow irrigation very effectively. Jaynes (1986) presented a numerical procedure for the solution of sloping and level borders based on the

zero-inertia model. His model utilized a finite difference scheme to model both advance and recession phases of flow conditions. The depth gradient term in the simplified momentum equation was expressed explicitly and averaged over the entire border. The author reported that this simplification made the model simpler to program and the required computer code less cumbersome.

The zero-inertia approximation of the full hydrodynamic model was utilized by many investigators for different applications in surface irrigation. Fangmeier and Strelkoff (1979) evaluated the U. S. Soil Conservation design criteria for sloping borders without runoff using a mathematical procedure which was based on the zero-inertia model. They concluded that the design charts of the U. S. Soil Conservation Service (1974) were reasonable for graded borders. However, they suggested that these charts be supplemented by a mathematical model, similar to their developed model. This would provide those designing irrigation systems with specific guidelines on the range of applicability of the charts for various irrigation flow parameters. Rayej and Wallender (1985) developed a nonlinear zero-inertia model for furrow irrigation. Later, they developed a zero-inertia model for surge flow irrigation based on their previous work (Wallender and Rayej, 1985). The non-linearity of the governing equations allowed for simultaneous modeling of wet and dry sections of furrows. Their model provided adequate simulations of the advance and recession phases of flow when compared to three sets of field data that utilized for comparison. Schwankl and Wallender (1988) studied the effect of the spatially-varying infiltration and wetted perimeter on furrow advance and infiltrated water distribution using a similar zero-inertia model developed by Rayej and Wallender (1985). More recently, Schmitz and Seus (1990) presented an analytical solution of the zero-inertia model as applied to the

advance phase in sloping and level borders. Their development was based on the assumption of a "moving momentum representative cross section in the water body".

The zero-inertia model was analyzed by many investigators to study its general response. This was accomplished through the process of non-dimensionalizing the solution to produce families of dimensionless advance curves in borders, level basins, and furrows (Katopodes and Strelkoff, 1977b; Clemmens and Strelkoff, 1981; Strelkoff and Clemmens, 1981; and Elliott et al., 1983a). These curves were restricted solely to the advance phase. Hence, they cannot be utilized to predict the entire irrigation process. However, these evaluations have produced very important design methodologies for level and sloping basins and borders (Clemmens and Strelkoff, 1979, and Strelkoff and Clemmens, 1981).

In short, the zero-inertia model represents a simplified hydrodynamic model that is intermediate in computational approach between the fully hydrodynamic model represented by [2.9] and [2.10] and the kinematic wave model which will be discussed in the following section. The zero-inertia model was utilized in various areas of open channels and surface hydrology as well as flood-routing and surface irrigation. The zero-inertia approximation transforms the system of partial differential equations in [2.9] and [2.10] from hyperbolic to parabolic form. This reduces the computer time requirements for the execution of this group of models. Many studies revealed that the zero-inertia model produces excellent results in the simulation of the hydraulic behavior of basin, border, and furrow irrigation systems as long as the assumptions of the model are not violated. The zero-inertia model appears to be very applicable to the above areas of irrigation since the assumption of neglecting the inertial terms in the momentum equation is realistic given the low values of Froude numbers prevailing under actual field conditions

(Strelkoff and Katopodes, 1977b, and Clemmens, 1978). The computational cost of the zero-inertia model is certainly cheaper than that of the complete hydrodynamic model.

3. Kinematic Wave Models

The Kinematic wave model is the most simplified form of the hydrodynamic model. This model is based on the assumption that the inertial terms in the momentum equation together with the term that describes the pressure variation in the direction of flow are negligible ([2.10]). The continuity equation ([2.9]) is kept unchanged. The simplified momentum equation has the form

$$S_0 = S_f \qquad [2.25]$$

This above equation implies that flow is at normal depth throughout the domain of solution (Bassett et al., 1980). The kinematic wave approximation is only applicable when the slope of the channel bed is steep. Given the relation that is depicted in [2.25], a uniform flow equation such as Chezy ([2.11]), Manning ([2.12]), or Darcy-Weisbach ([2.13]) may be used to relate flow rate (Q) to flow depth (y) or cross-sectional area (A).

The kinematic wave model was named after identifying the fact that the method projects the movement of a kinematic shock wave. Since every kinematic wave model utilizes a uniform equation to establish flow-depth relationship, these models are frequently referred to in the literature as "uniform depth" or "uniform flow" models (Walker and Skogerboe, 1987).

The kinematic wave method as a technique was first proposed by Lighthill and Whitham (1955) for modeling overland flow. Later, it was utilized in the solution of watershed problems and in predicting flood

movements in rivers (Henderson and Wooding, 1964; Wooding, 1965a; and Wooding, 1965b). Woolhiser and Liggett (1967) examined the errors introduced by the application of the kinematic wave model to overland flow problems. Their work showed that the simplified hydrodynamic model based on the kinematic wave assumptions is applicable to these problems within a certain range of input parameters. Following their application, the kinematic wave theory was heavily utilized in many investigations on surface runoff in watershed hydrology (Brakensiek, 1967; Woolhiser, 1969; and Singh, 1975). Singh (1976) conducted a study to assess the discretization error of four different finite difference numerical schemes frequently used in solving the kinematic wave equations. He examined the problems of convergence and stability of these schemes. Since then, many other investigations were conducted in different areas of surface hydrology and open channel flow using kinematic wave models. The most recent is a study by Hromadka and DeVries (1988) where they examined the use of the kinematic wave method in open channel flow routing of runoff hydrographs. Their work concentrated on the significance of the computational errors in the application of numerical procedures of the kinematic wave models. They were also interested in assessing the effect of the various assumptions implemented with the kinematic wave model as opposed to the complete hydrodynamic model.

The utilization of the kinematic wave theory in hydrologic applications was extended to sloping, free draining borders by Chen (1970) and Smith (1972). In his development, Chen based the solution of the kinematic wave model on the method of characteristics with the help of initially prescribed initial and boundary conditions. Chen indicated that the depth or discharge of flow at any time and distance from the inlet can "be determined from the family of characteristic curves in the x, t-plane". In his conclusions, he stated that "the kinematic wave method may only be valid for super-critical flow. For

other than super-critical flow, the more general hydrodynamic approach should be adopted". Smith (1972) discussed two methods for solving the kinematic wave model in flood wave movement and attenuation in dry alluvial channels. The first included the application of the method of characteristics outlined by Chen (1970). This approach reduces the partial differential equation as represented by the continuity equation in [2.9] to two-characteristic ordinary differential equations in the x, t-plane combined with a third equation for shock movement. The second included the solution of the same partial differential equation using finite difference approximation and a rectangular grid. After comparing the results of both numerical approaches to available field data, Smith reported that the kinematic wave assumption was very reasonable for the particular cases of unsteady wave flow addressed. These cases included sloping border irrigation and ephemeral flood routing. He cited both the work of Woolhiser and Liggett (1967) and Tinney and Bassett (1961) as the proper indication for the appropriateness of the kinematic wave assumption under the conditions of his study. Once again, Smith reported that the results of the kinematic wave models were more sensitive to the infiltration function.

The formulation of free boundary problems in surface irrigation using both complete hydrodynamic and kinematic wave models was investigated by Sherman and Singh (1978 and 1982). In their work, Sherman and Singh presented explicit formulations of the free boundary problems in surface irrigation. Their approach was also based on the method of characteristics.

As was the case with other models, the kinematic wave model was first developed for border irrigation, and later applied to furrow irrigation systems. Walker and Humpherys (1983) highlighted three essential modifications in the mathematical procedures utilized in the analysis of border irrigation systems before these analyses are implemented in the simulation of furrow irrigation

systems. These include the description of the geometry of flow cross section, implementation of an infiltration function that accounts for both steady and time dependent infiltration rates, and assessing the effect of the wetted perimeter on the infiltration function. Walker and Humpherys accounted for the first two modifications in their study which included the development of an implicit finite difference scheme to the direct solution of the kinematic wave furrow irrigation model. Their scheme was based on the Eulerian integration approach which represents the numerical approximation of the continuity equation based on the concept of multi-cell deforming control volume. Using this approach the continuity equation became

$$\begin{aligned}&\{[\theta Q_R+(1-\theta)Q_M]-[\theta Q_L+(1-\theta)Q_J]\}\delta t\\&\quad+\{[\phi A_L+(1-\phi)A_R]-[\phi A_J+(1-\phi)A_M]\}\delta x\\&\quad+\{[\phi Z_L+(1-\phi)Z_R]-[\phi Z_J+(1-\phi)Z_M]\}\delta x=0\end{aligned}\qquad[2.26]$$

The subscripts J and M represent the left and right cell boundaries, respectively, at time t_{i-1} while the subscripts L and R represent the left and right cell boundaries, respectively, at time t_i. Walker and Humpherys gave a detailed description of their mathematical development. In their conclusions, they reported that the kinematic wave analysis is "a satisfactory tool to predict water advance, intake, and runoff from sloped furrow irrigated systems". After comparing the characteristic furrow model to the integral model, they reported that "the integral model was superior on the basis of its adaptability to both surged and continuous flows, less sensitivity to the size of the time step, and the numerical stability of the solution". This was supported by the work of Walker and Lee (1981). Izuno and Podmore (1985) developed yet another kinematic wave model for surge and continuous irrigation of furrows. The surge infiltration function utilized in their study was based on the

two-branch function of the Kostiakov equation as suggested by Clemmens (1981). They reported acceptable model predictions of advance under surged and continuous furrow irrigation applications.

The kinematic wave assumption simplifies the analysis of surface irrigation systems immensely. However, this same assumption limits the use of kinematic wave models to the free draining graded borders and furrows with relatively steep slopes (James, 1988). Hence, the model is inapplicable to dead-level fields and diked borders or furrows. The limitations can be attributed to the facts that the normal depth is infinite in the first case while the kinematic wave solution would be only influenced by the upstream boundary conditions (no downstream boundary conditions could be imposed on the flow) in the second case (Bassett et al., 1980).

4. Volume Balance Models

Volume balance models are the most simplified form of the fully dynamic equations. These models neglect the entire momentum equation and implement some approximations to the continuity equation. The continuity equation ([2.9]) is applied to the entire flow profile at once (Bassett et al., 1980). The continuity equation is integrated over space, which represents the length of the advancing stream, to produce (Hart et al., 1968, and Bassett et al., 1980)

$$Q_0 = \frac{dV_y(t)}{dt} + \frac{dV_z(t)}{dt} \qquad [2.27]$$

where Q_0 is the inlet flow rate, $V_y(t)$ is the surface volume of water, and $V_z(t)$ is the infiltrated volume of water. Equation [2.27] can be further integrated over time to yield

$$Q_0 t = V_y(t) + V_z(t) \qquad [2.28]$$

The surface and infiltrated volumes of water can be determined by integrating $V_y(t)$ and $V_z(t)$ over the advance distance and substituting the results in [2.28]. This results in (Walker and Skogerboe, 1987)

$$Q_0 t = \int_0^x A(s,t)ds + \int_0^x Z(s,t)ds \qquad [2.29]$$

where s is the integrand of x and Z is the infiltrated volume per unit length. Since the momentum equation which describes the temporal and spatial variation is completely ignored, the volume balance model is based on the assumption that the average area of flow is a constant, $\overline{A}$. If the infiltration is considered to be a function of intake-opportunity time, [2.29] reduces to

$$Q_0 t = \overline{A}x + \int_0^x Z(t - t_s)ds \qquad [2.30]$$

where $t - t_s$ is the intake-opportunity time and t_s is the time at which the advancing front of the stream reaches distance s.

The various volume balance models presented in the literature can be classified into four different categories. The first category is based on the recursive approach. The work by Hall (1956) and Strelkoff (1977) represent two good examples of this category of volume balance models. While Hall (1956) solved the border advance problem, Strelkoff modeled all phases of irrigation in borders including the depletion and recession phases. The second category of the volume balance models is based on the Kernel function approach. The work by Hart et al. (1968) was based on this approach. Their work was also applied to border irrigation systems using the Kostiakov infiltration equation. The third category of volume balance models utilizes the

Laplace transform approach. In this approach the Laplace transform of [2.30] is established. A good example of such an approach is the research work by Philip and Farrell (1964). The last group of volume balance models is based on the power advance approach. Among the many researchers that followed this approach were Fok and Bishop (1965), Wilke and Smerdon (1965), Chen (1966), and Singh and Chauhan (1972).

Since the volume balance model has the basic approximations of the kinematic wave model together with many other assumptions, it must at least have the same limitations that were highlighted in the previous section. As a matter of fact, the volume balance models have additional inaccuracies since the shape functions are presumed arbitrarily (Bassett et al., 1980). However, the volume balance models have the cheapest execution costs among the various models covered to this point. This makes this approach good for quick, rough initial calculations.

E. Numerical Solution of the Hydrodynamic Equations

The solution of the full hydrodynamic equations or a simplified form of these equations requires the implementation of a numerical solution procedure. The partial differential equations that are represented by the Saint-Venant equations could be transformed to ordinary differential equations after utilizing the method of characteristics. Then, the resultant ordinary differential equations are usually solved numerically. Another alternative implements the transformation of the above partial differential equations to a system of algebraic equations amenable to solution.

There are various numerical techniques for solving differential equations. The finite difference and finite element methods represent the two most widely used procedures for obtaining numerical solutions to both ordinary and partial

differential equations. These numerical procedures are utilized in many problems especially overland flow problems which cover various areas of hydraulics and hydrology including flood routing and open channel flow.

1. Finite Difference

The finite difference method approximates ordinary or partial differential equations with difference equations. Using this numerical method, a continuum is replaced by a series of discrete points between which the differentials are approximated. The finite difference method has been the most widely used procedure for approximating differential equations numerically. It was heavily utilized repeatedly in the solution of shallow-water equations which describe flow conditions in both open channels and overland flow (Liggett and Woolhiser, 1967). Over the last several years, many scientists utilized the finite difference method in the direct or indirect solution of the simplified or complete hydrodynamic equations as applied to surface irrigation problems. The review of such work was briefly outlined in the previous section on surface irrigation models and therefore will not be repeated in this section.

2. Finite Element

The finite element method utilizes an integral formulation to generate a system of algebraic equations after approximating a continuum with a continuous piecewise smooth functions (Segerlind, 1984). This method was initially developed for the analysis of problems in structural mechanics. Later, the finite element method was applied to the numerical solution of various

classes of problems. The application of the finite element method to many fluid mechanics problems serves as a good evidence for the wide spread use of this numerical technique.

The application of the finite element method to overland flow problems was the subject of many investigations. Among the many early publications are the studies by Guymon (1972), Taylor et al. (1974), Judah et al. (1975), Desai (1979), Ross et al. (1977; 1979; and 1980), and Heatwole et al. (1982). All of these researchers have reported some success in utilizing the finite element method for modelling the various physical processes in overland flow problems which are governed by the shallow-water equations. The most recent developments in this area were the studies by Katopodes (1984), Akanbi and Katopodes (1988), Hu et al. (1989), Kaneko (1989), Kashiyama and Kawahara (1989), and Vieux (1989).

On the other hand, the application of the finite element method to irrigation problems has been very limited. Much of the prior work focused on utilizing the finite element formulation in the analysis of both pressure and flow conditions in sprinkler and drip irrigation systems as well as pipe-network analyses. The application of the finite element method in these areas was unique in the sense that the solution processes didn't start from the partial or ordinary differential equations but rather from utilizing the direct stiffness procedure of the finite element method. The reader is referred to the work by Bralts (1981) and Bralts and Segerlind (1985) for a detailed description of this approach.

F. Synopsis

After reviewing the theory and literature that is pertinent to this research work, it was clear that an extensive amount of work has been done in the area

of numerical analysis of surface irrigation systems based on the hydrodynamic equations. However, what seems to be lacking is a general numerical formulation of the complete and simplified forms of the hydrodynamic equations in one model. Even though many comparison were made in the literature among the various forms of the hydrodynamic equations as applied to surface irrigation systems, these comparisons were not done on the same basis. On the other hand, it was observed that the finite element method was not exploited in the area of numerical analysis of surface irrigation systems. The application of the finite element method to many fluid mechanics problems serves as a good evidence for the wide spread use of this numerical technique. This also serves as an indication that the method could successfully be used for the hydraulic analysis of surface irrigation systems.

Based on the above, it was felt that there is a need to develop a mathematical formulation of the Saint-Venant equations for the analysis of surface irrigation systems using the finite element method. The numerous features of this numerical technique makes it attractive to the solution of initial and boundary value problems that can be described by first or second order partial differential equations. The simplicity in handling boundary conditions and the ability of the method to accurately handle complex solution domains are two of the many important features of the finite element method.

III. METHODOLOGY

The flow of water across the soil surface in any surface irrigation systems is spatially varied. Moreover, the hydraulic design and analysis of surface irrigation systems is a time dependent process. The hydraulics of flow in both sloping furrow and border irrigation systems is governed by two first-order partial differential equations. These unsteady flow equations were developed originally by A.J.C. Barre De Saint-Venant in 1871 (Miller and Yevjevich, 1975). The development of the Saint-Venant equations was based on the application of the conservation of mass and momentum principles to the analysis of surface flow conditions in open channels. Since their development, the Saint-Venant equations have been used in many areas of hydrology including the study of river floods and propagation of tides in river channels. These equations were later used in the analysis of flow conditions in surface irrigation systems.

There are four general mathematical schemes that result from applying the Saint-Venant equations to the hydraulic analysis of surface flow problems in general and surface irrigation problems in particular. These mathematical approaches result in one of the following models: hydrodynamic, zero-inertia, kinematic wave, and volume balance models. The primary difference among the above list of models lies in the number of assumptions that are implemented in the Saint-Venant equations.

In order to analyze various surface flow problems in any of the above models, a numerical procedure needs to be implemented. Historically, these numerical procedures have been based on the finite difference method, the

method of characteristics, or a combination of both. Such models were developed over the years by many scientists as was discussed in the previous chapter. More recently, the finite element numerical procedure was implemented in the solution of surface flow problems. However such developments were limited to the areas of surface hydrology and open channels.

The sole purpose of this research study is to develop a methodology for implementing the Galerkin formulation of the finite element numerical procedure to the hydraulic analysis of flow conditions in surface irrigation systems. A general finite element model will be developed to perform the hydraulic analysis of surface irrigation problems using the hydrodynamic, zero-inertia, and kinematic wave models.

A. Research Approach

Five fundamental objectives are presented as the goals of this research. The approaches utilized to achieve these objectives are delineated below.

Objective 1. To develop a finite element solution procedure of the Saint-Venant equations for the hydraulic analysis of surface irrigation systems.

The approach to be followed under Objective 1 will be to apply the Galerkin formulation of the finite element method to the solution of the Saint-Venant equations using both linear and quadratic one-dimensional elements. The Galerkin formulation will be applied to both the continuity and momentum equations with respect to the space coordinate for a fixed instant of time. Each will result in a system of first-order differential equations in the time domain. The resultant two systems of ordinary differential equations will be combined into one general system. Then, a finite difference approximation

in the time domain will be applied to the final general system of equations to generate a system of algebraic equations which will then be solved iteratively over time. The direct stiffness procedure will be utilized in building global systems of equations at various time steps.

Objective 2. To create a general solution approach that will accommodate the available mathematical models of the Saint-Venant equations in the analysis of surface irrigation systems.

The approach to be followed under Objective 2 will be to establish the coefficients that will implement the various assumptions utilized in establishing the hydrodynamic, zero-inertia, and kinematic wave mathematical models from the Saint-Venant equations. By choosing these coefficients, the solution process would be performed based on the selected model. The general development would apply to the selected model and the solution process will never be altered by the choice of the mathematical model.

Objective 3. To develop an approach to easily incorporate the varying boundary conditions of the advance, ponding, depletion, and recession phases of surface irrigation into the solution process with minimal arbitrary or experimental parameters.

The approach to be followed under Objective 3 will include the modification of the final system of equations to incorporate given boundary conditions under varying physical phases of an irrigation cycle. The possibility of implementing this approach should be straight forward since including boundary conditions at a later stage of the solution process is one of the primary features of the finite element method. The dimensions of the total system of equations will remain unchanged at any instance in time.

Objective 4. To develop a computer model that will utilize the above mathematical concepts for the hydraulic analysis of flow conditions in border and furrow irrigation systems.

The approach to be followed under Objective 4 will be to implement the finite element mathematical development of the motion equations in building a computer model that will simulate the advance, ponding, depletion, and recession phases of both furrow and border irrigation systems. The computer model will be developed to run on any IBM-compatible microcomputer with a Random Access Memory (RAM) of 512 Kbytes or more and an MS-DOS version 2.00 or higher.

Objective 5. To numerically evaluate and compare the results of the finite element model to actual field measurements from existing surface irrigation systems.

The approach to be followed under Objective 5 will be to compare the results to be obtained from running the developed finite element computer model to those reported from actual field measurements for existing surface irrigation systems. A graphics routine will be developed to display both simulated and actual data of the various flow phases of irrigation on the same graph. The graphical display will include plots of actual and predicted advance and recession trajectories of flow and will be revealed after the conclusion of any surface irrigation simulation run. This will be utilized to evaluate the utility of the developed finite element model and its ability to accurately simulate the hydraulic conditions of flow in surface irrigation systems.

B. Theoretical Development

There are two basic equations that can be utilized in the hydraulic analysis of flow conditions in open channels and surface irrigation systems. The two equations, or the so-called Saint-Venant equations, were developed based on applying the conservation of mass and momentum principles to flow conditions. This research will utilize these equations as the basis for the hydraulic analysis of various surface irrigation systems. The finite element method will be used then to solve these equations numerically. The resultant numerical model will then be applied to the analysis of hydraulic conditions in furrow and border irrigation systems. This general development may also be applied to the analysis of flow conditions in open channels with or without infiltration. However, the assumption that the slope of channel bed is mild will be utilized in the development, an assumption which is very reasonable in surface irrigation systems but not necessarily true in many open channels.

1. Development of the Saint-Venant Equations

The development of the Saint-Venant equations will be repeated in this section to delineate the important principles of these two equations as related to surface irrigation.

a. Continuity Equation

The volume of water stored in a fluid element (Figure 1) within a spatially varied furrow is represented by the following relationship

$$dV_s = V_{in} - V_{out} - V_I \qquad [3.1]$$

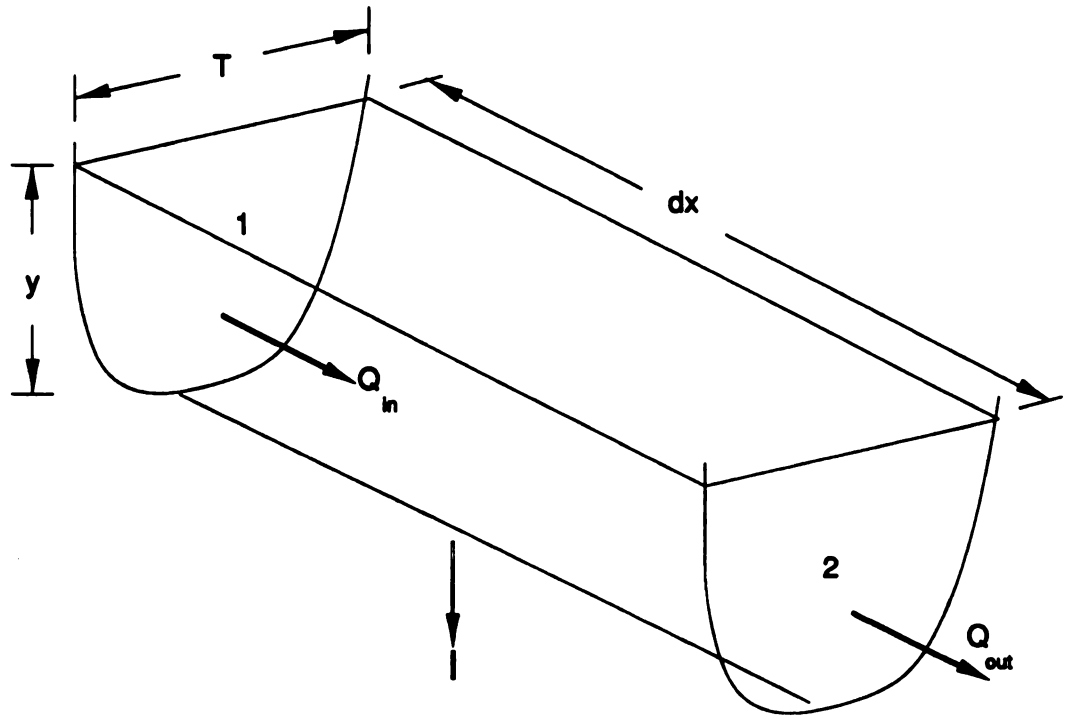


Figure 1. An enlarged fluid element within a furrow with a spatially varied unsteady flow.

where dV_s is the change in the stored volume within the element during time dt, V_{in} is the total inflow during dt, V_{out} is the total outflow during dt, and V_I is the total infiltration during dt. The total inflow during a time step dt is described as

$$\begin{aligned} V_{in} &= \left(\frac{Q_{1(t)} + Q_{1(t+dt)}}{2}\right)dt \\ &= \frac{1}{2}\left[Q + \left(Q + \frac{\partial Q}{\partial t}\right)dt\right]dt \\ &= Q\,dt + \frac{1}{2}\frac{\partial Q}{\partial t}dt^2 \\ &\approx Q\,dt \end{aligned} \qquad [3.2]$$

where $Q_{1(t)}$ and $Q_{1(t+dt)}$ are the inflow rates at times t and $t+dt$, respectively. The total outflow during the time step dt is

$$\begin{aligned} V_{out} &= \left(\frac{Q_{2(t)} + Q_{2(t+dt)}}{2}\right)dt \\ &= \frac{1}{2}\left[\left(Q + \frac{\partial Q}{\partial x}dx\right) + \left(\left(Q + \frac{\partial Q}{\partial x}dx\right) + \frac{\partial\left(Q + \frac{\partial Q}{\partial x}dx\right)}{\partial t}dt\right)\right]dt \\ &= Q\,dt + \frac{\partial Q}{\partial x}dx\,dt + \frac{1}{2}\frac{\partial\left(Q + \frac{\partial Q}{\partial x}dx\right)}{\partial t}dt^2 \\ &\approx Q\,dt + \frac{\partial Q}{\partial x}dx\,dt \end{aligned} \qquad [3.3]$$

where $Q_{2(t)}$ and $Q_{2(t+dt)}$ are the outflow rates at times t and $t+dt$, respectively. The total infiltration volume during the time step dt is

$$V_I = \left(\frac{V_{I(t)} + V_{I(t+dt)}}{2}\right)dt$$

where $V_{I(t)}$ and $V_{I(t+dt)}$ are the average infiltration rates at times t and $t+dt$, respectively. These two terms can be developed separately as

$$V_{I(t)} = \left(\frac{I_{1(t)} + I_{2(t)}}{2}\right) dx$$

$$= \left(I + \frac{\left(I + \frac{\partial I}{\partial x} dx\right)}{2}\right) dx$$

$$= I\,dx + \frac{1}{2}\frac{\partial I}{\partial x} dx^2$$

$$\approx I\,dx$$

and

$$V_{I(t+dt)} = \left(\frac{I_{1(t+dt)} + I_{2(t+dt)}}{2}\right) dx$$

$$= \frac{1}{2}\left[\left(I + \frac{\partial I}{\partial t} dt\right) + \left(\left(I + \frac{\partial I}{\partial x} dx\right) + \frac{\partial\left(I + \frac{\partial I}{\partial x} dx\right)}{\partial t} dt\right)\right] dx$$

$$= I\,dx + \frac{\partial I}{\partial t} dt\,dx + \frac{1}{2}\frac{\partial\left(I + \frac{\partial I}{\partial x} dx\right)}{\partial t} dt\,dx$$

where $I_{1(t)}$ and $I_{1(t+dt)}$ are the left boundary infiltration rates per unit length at times t and $t+dt$, respectively, and $I_{2(t)}$ and $I_{2(t+dt)}$ are the right boundary infiltration rates per unit length at times t and $t+dt$, respectively. After substituting the above two terms, V_I becomes

$$V_I = \frac{1}{2}\left[(I\,dx) + \left(I\,dx + \frac{\partial I}{\partial t} dt\,dx + \frac{1}{2}\frac{\partial\left(I + \frac{\partial I}{\partial x} dx\right)}{\partial t} dt\,dx\right)\right] dt$$

$$= I\,dx\,dt + \frac{1}{2}\frac{\partial I}{\partial t} dt^2 dx + \frac{1}{4}\frac{\partial\left(I + \frac{\partial I}{\partial x} dx\right)}{\partial t} dt^2 dx$$

$$\approx I\,dx\,dt \qquad [3.4]$$

Substituting [3.2], [3.3], and [3.4], in [3.1] results in

$$dV_s = [Q\,dt] - \left[Q\,dt + \frac{\partial Q}{\partial x} dx\,dt \right] - [I\,dx\,dt] \qquad [3.5]$$

or

$$\frac{dV_s}{dt} = -\frac{\partial Q}{\partial x} dx - I\,dx \qquad [3.6]$$

The storages in the element at times t and $t+dt$ can be defined by the following two terms

$$\begin{aligned} V_{s(t)} &= \left[\frac{A_{1(t)} + A_{2(t)}}{2} \right] dx \\ &= \left[\frac{A + \left(A + \frac{\partial A}{\partial x} \right) dx}{2} \right] dx \\ &= A\,dx + \frac{1}{2} \frac{\partial A}{\partial x} dx^2 \\ &\approx A\,dx \end{aligned} \qquad [3.7]$$

and

$$\begin{aligned} V_{s(t+dt)} &= \left[\frac{A_{1(t+dt)} + A_{2(t+dt)}}{2} \right] dx \\ &= \frac{1}{2} \left[\left(A + \frac{\partial A}{\partial t} dt \right) + \left(\left(A + \frac{\partial A}{\partial x} dx \right) + \frac{\partial \left(A + \frac{\partial A}{\partial x} dx \right)}{\partial t} dt \right) \right] dx \\ &= A\,dx + \frac{1}{2} \frac{\partial A}{\partial x} dx^2 + \frac{1}{2} \frac{\partial A}{\partial t} dx\,dt + \frac{\partial \left(A + \frac{\partial A}{\partial x} dx \right)}{2 \partial t} dx\,dt \\ &\approx A\,dx + 2 \left(\frac{1}{2} \frac{\partial A}{\partial t} dx\,dt \right) \\ &= A\,dx + \frac{\partial A}{\partial t} dx\,dt \end{aligned} \qquad [3.8]$$

where $A_{1(t)}$ and $A_{1(t+dt)}$ are the left boundary areas of flow at times t and $t+dt$, respectively; $A_{2(t)}$ and $A_{2(t+dt)}$ are the right boundary areas of flow at times t and $t+dt$, respectively; and $V_{s(t)}$ and $V_{s(t+dt)}$ are the storages at times t and $t+dt$, respectively. Substituting [3.7] and [3.8] in [3.6], the continuity equation would result as follows

$$-\frac{\partial Q}{\partial x}dx - Idx = \frac{V_{s(t+dt)} - V_{s(t)}}{dt}$$

$$= \frac{\left[\left(A dx + \frac{\partial A}{\partial t}dx dt\right) - (A dx)\right]}{dt}$$

$$= \frac{\partial A}{\partial t}dx$$

or

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} + I = 0 \qquad [3.9]$$

b. Forces Acting on the Fluid Element

There are three forces which act upon the fluid element in Figure 2. These include the force due to the weight of the fluid element, the pressure force which represents the resultant of two pressure forces acting on the upstream and downstream boundaries of the fluid element, and the friction force which results from the resistance to water flow due to the viscous force along the wetted perimeter of the element. These forces can be assessed as discussed below.

i. Weight forces : The weight of the fluid element can be expressed as

$$F_w = \left[\frac{F_{w(t)} + F_{w(t+dt)}}{2}\right] \qquad [3.10]$$

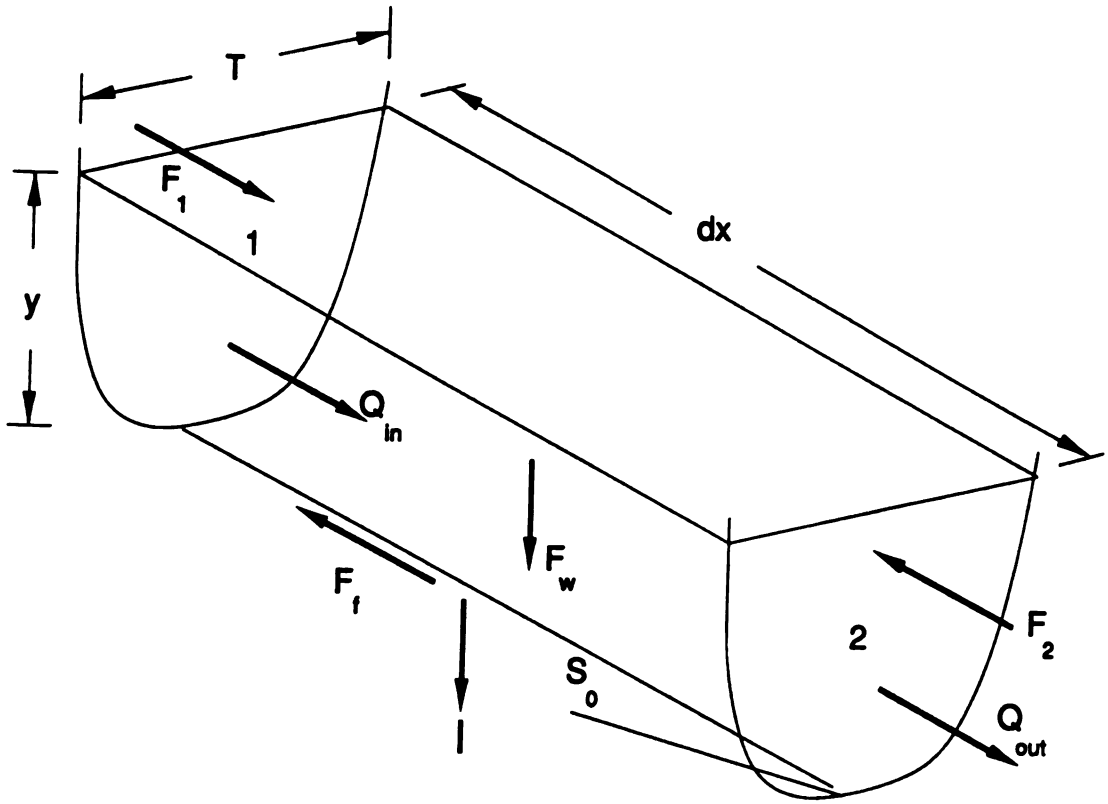


Figure 2. Acting forces on an enlarged fluid element within a furrow with a spatially varied unsteady flow.

where $F_{w(t)}$ and $F_{w(t+dt)}$ are the forces due to the weight of water at times t and $t+dt$, respectively. The first component of [3.10] can be written as

$$\begin{aligned} F_{w(t)} &= \gamma\left[\frac{A_{1(t)}+A_{2(t)}}{2}\right]dx \\ &= \gamma\left[\frac{(A)+\left(A+\frac{\partial A}{\partial x}dx\right)}{2}\right]dx \\ &= \gamma A\,dx+\frac{\gamma}{2}\frac{\partial A}{\partial x}dx^2 \\ &\approx \gamma A\,dx \end{aligned} \qquad [3.11]$$

where γ is the specific weight of water and $A_{1(t)}$ and $A_{2(t)}$ are the cross sectional areas of flow at the upstream and downstream boundaries of the fluid element, respectively, at time t. The second component of [3.10] can be defined as

$$\begin{aligned} F_{w(t+dt)} &= \gamma\left[\frac{A_{1(t+dt)}+A_{2(t+dt)}}{2}\right]dx \\ &= \frac{\gamma}{2}\left[\left(A+\frac{\partial A}{\partial t}dt\right)+\left(\left(A+\frac{\partial A}{\partial x}dx\right)+\frac{\partial\left(A+\frac{\partial A}{\partial x}dx\right)}{\partial t}dt\right)\right]dx \\ &= \gamma\left[A\,dx+\frac{1}{2}\frac{\partial A}{\partial t}dt\,dx+\frac{1}{2}\frac{\partial A}{\partial x}dx^2+\frac{\partial\left(A+\frac{\partial A}{\partial x}dx\right)}{2\partial t}dt\,dx\right] \\ &= \gamma\left[A\,dx+\frac{1}{2}\frac{\partial A}{\partial t}dt\,dx+\frac{1}{2}\frac{\partial A}{\partial t}dt\,dx\right] \\ &= \gamma A\,dx+\gamma\frac{\partial A}{\partial t}dt\,dx \end{aligned} \qquad [3.12]$$

where $A_{1(t+dt)}$ and $A_{2(t+dt)}$ are the cross sectional areas of flow at the upstream and downstream boundaries of the fluid element, respectively, at time $t+dt$. Substituting [3.11] and [3.12] in [3.10] results

$$F_w = \frac{[\gamma A\,dx] + \left[\gamma A\,dx + \gamma \frac{\partial A}{\partial t} dt\,dx\right]}{2}$$

$$= \gamma A\,dx + \frac{\gamma}{2}\frac{\partial A}{\partial t} dt\,dx \qquad [3.13]$$

The component of the weight of the fluid element with the direction of flow, F_{wx}, is obtained from

$$F_{wx} = \sin\alpha \cdot F_w$$

where α is the angle between the lower boundary, or the bottom of the channel, of the fluid element and the horizontal plane. If the slope of the lower boundary (S_0) is assumed to be small, the sine of the angle can be approximated by the tangent. This results in

$$F_{wx} = \mathrm{Tan}\,\alpha \cdot F_w$$

$$= S_0 F_w$$

where S_0 is the slope of the furrow or border. Substituting [3.13] in the above equation results in

$$F_{wx} = \gamma A S_0 dx + \frac{1}{2}\frac{\partial A}{\partial t} dt\,dx S_0 \qquad [3.14]$$

The second term in the equation above is negligible compared to the first term which contains the cross sectional area of flow. Therefore, Equation [3.14] reduces to

$$F_{wx} = \gamma A S_0 dx \qquad [3.15]$$

ii. Pressure forces: The pressure force, F_p, which represents the resultant of the pressure forces acting on the upstream and downstream boundaries of the fluid element can be written as

$$F_p = F_1 - F_2 \qquad [3.16]$$

where F_1 and F_2 are the pressure force acting on left and right boundaries, respectively. The pressure force acting on the left boundary of the fluid element can be determined as follows

$$\begin{aligned} F_1 &= \left[\frac{F_{1(t)} + F_{1(t+dt)}}{2}\right] \\ &= \frac{1}{2}\left[(\gamma hA) + \gamma\left(Ah + \frac{\partial Ah}{\partial t}dt\right)\right] \\ &= \gamma hA + \frac{\gamma}{2}\frac{\partial Ah}{\partial t}dt \end{aligned} \qquad [3.17]$$

where $F_{1(t)}$ and $F_{1(t+dt)}$ are the pressure forces acting on the left boundary at times t and $t+dt$, respectively; γ is the specific weight of water; and h is the distance from the water surface to the centroid of the left and right areas of flow. The pressure force acting on the right side boundary of the fluid element can be written as

$$\begin{aligned} F_2 &= \left[\frac{F_{2(t)} + F_{2(t+dt)}}{2}\right] \\ &= \frac{1}{2}\left[\left(\gamma Ah + \gamma\frac{\partial Ah}{\partial x}dx\right) + \left(\gamma Ah + \gamma\frac{\partial Ah}{\partial x}dx + \gamma\frac{\partial\left(\gamma h + \frac{\partial Ah}{\partial x}dx\right)}{\partial t}dt\right)\right] \\ &= \gamma Ah + \gamma\frac{\partial Ah}{\partial x}dx + \frac{\gamma}{2}\frac{\partial(\gamma h)}{\partial t}dt + \frac{\gamma}{2}\frac{\partial\left(\frac{\partial Ah}{\partial x}dx\right)}{\partial t}dt \\ &\approx \gamma Ah + \gamma\frac{\partial Ah}{\partial x}dx + \frac{\gamma}{2}\frac{\partial(\gamma h)}{\partial t}dt \end{aligned} \qquad [3.18]$$

where $F_{2(t)}$ and $F_{2(t+dt)}$ represent the pressure forces which act on the right boundary of the fluid element at times t and $t+dt$, respectively. Substituting [3.17] and [3.18] in [3.16] produces

$$F_p = \left(\gamma hA + \frac{\gamma}{2}\frac{\partial Ah}{\partial t}dt\right) - \left(\gamma hA + \gamma\frac{\partial Ah}{\partial x}dx + \frac{\gamma}{2}\frac{\partial(Ah)}{\partial t}dt\right)$$

$$= -\gamma\frac{\partial Ah}{\partial x}dx$$

$$= -\gamma\frac{\partial Ah}{\partial h}\frac{\partial h}{\partial x}dx \qquad [3.19]$$

Potter and Wiggert (1991) developed the following relationship using the Leibnitz rule from calculus (refer to page 463 of their text):

$$\frac{\partial(Ah)}{\partial h} = A \qquad [3.20]$$

Substituting [3.20] in [3.19] results in

$$F_p = -\gamma A dx\frac{\partial y}{\partial x} \qquad [3.21]$$

iii. Friction forces: The friction force, F_f, which results from the resistance to water flow can be expressed as

$$F_f = \tau_0 P dx \qquad [3.22]$$

where τ_0 is the shear stress at and P is the wetted parameter. The slope of the friction slope can be defined as (Potter and Wiggert, 1991)

$$S_f = \frac{\tau_0}{\gamma R} \qquad [3.23]$$

Equation [3.22] can be rearranged as

$$F_f = \frac{\tau_0 P A\,\gamma dx}{A\,\gamma}$$

$$= \frac{\tau_0 A\,\gamma dx}{R\,\gamma} \qquad [3.24]$$

where $R = A/P$. Substituting Equation [3.23] in [3.24] results in

$$F_f = \gamma A dx S_f \qquad [3.25]$$

c. Unsteady Momentum

The conservation of momentum in Figure 2 follows Newton's second law of motion (Walker and Skogerboe, 1987). It states that the resultant force which acts on the fluid element in motion is equal to the rate of momentum change within the element and the momentum flux across the element boundaries. This can be represented by the following relationship

$$\Sigma(Forces) = (\rho Q v)_{out} - (\rho Q v)_{in} + \frac{d(mv)}{dt} \qquad [3.26]$$

where m is the mass of the fluid element, Q is the flow rate, v is the flow velocity, ρ is the density of the fluid, $\Sigma(Forces)$ is the summation of forces $(F_{wx} + F_p - F_f)$, $(\rho Q v)_{in}$ is the average momentum flux into the element, $(\rho Q v)_{out}$ is the average momentum flux out of the element, and $(d(mv)/dt)$ is the average momentum change during the time increment, dt. The first two right side terms of Equation [3.26] can be evaluated individually as follows :

$$(\rho Q v)_{in} = \frac{1}{2}\left[(\rho Q v) + \left(\rho Q v + \frac{\partial(\rho Q v)}{\partial t} dt \right) \right]$$

$$= \rho Q v + \frac{\rho dt}{2}\left[v \frac{\partial Q}{\partial t} + Q \frac{\partial v}{\partial t} \right] \qquad [3.27]$$

and

$$(\rho Q v)_{out} = \frac{1}{2}\left[\rho\left(Q+\frac{\partial Q}{\partial x}dx\right)\left(v+\frac{\partial v}{\partial x}dx\right)\right]$$

$$+\frac{1}{2}\left[\rho\left(Q+\frac{\partial Q}{\partial x}dx\right)\left(v+\frac{\partial v}{\partial x}dx\right)+\rho\frac{\partial\left[\left(Q+\frac{\partial Q}{\partial x}dx\right)\left(v+\frac{\partial v}{\partial x}dx\right)\right]}{\partial t}dt\right]$$

$$=\rho Q v+\rho v\frac{\partial Q}{\partial x}dx+\rho Q\frac{\partial v}{\partial x}dx+\rho\frac{\partial Q}{\partial x}\frac{\partial v}{\partial x}dx^2$$

$$+\frac{\rho dt}{2}\left[\left(v+\frac{\partial v}{\partial x}dx\right)\frac{\partial\left(Q+\frac{\partial Q}{\partial x}dx\right)}{\partial t}+\left(Q+\frac{\partial Q}{\partial x}dx\right)\frac{\partial\left(v+\frac{\partial v}{\partial x}dx\right)}{\partial t}\right]$$

$$=\rho Q v+\rho v\frac{\partial Q}{\partial x}dx+\rho Q\frac{\partial v}{\partial x}dx$$

$$+\frac{\rho dt}{2}\left[\left(v+\frac{\partial v}{\partial x}dx\right)\left(\frac{\partial Q}{\partial t}+\frac{\partial^2 Q}{\partial x\partial t}dx\right)+\left(Q+\frac{\partial Q}{\partial x}dx\right)\left(\frac{\partial v}{\partial t}+\frac{\partial^2 v}{\partial x\partial t}dx\right)\right]$$

$$=\rho Q v+\rho v\frac{\partial Q}{\partial x}dx+\rho Q\frac{\partial v}{\partial x}dx+\frac{\rho dt}{2}\left[v\frac{\partial Q}{\partial t}+v\frac{\partial^2 Q}{\partial x\partial t}dx\right]$$

$$+\frac{\rho dt}{2}\left[\frac{\partial v}{\partial x}\frac{\partial Q}{\partial t}dx+\frac{\partial v}{\partial x}\frac{\partial^2 Q}{\partial x\partial t}dx^2+Q\frac{\partial v}{\partial t}+Q\frac{\partial^2 v}{\partial x\partial t}dx\right]$$

$$+\frac{\rho dt}{2}\left[\frac{\partial Q}{\partial x}\frac{\partial v}{\partial t}dx+\frac{\partial Q}{\partial x}\frac{\partial^2 v}{\partial x\partial t}dx^2\right] \qquad [3.28]$$

If the second and third order differential products are assumed negligible, then [3.28] reduces to

$$\rho Q v_{out}=\rho Q v+\rho v\frac{\partial Q}{\partial x}dx+\rho Q\frac{\partial v}{\partial x}dx+\frac{\rho v dt}{2}\frac{\partial Q}{\partial t}+\frac{\rho Q dt}{2}\frac{\partial v}{\partial t} \qquad [3.29]$$

The change of momentum, $d(mv)$, can be determined as

$$d(mv)=(mv)_{t+dt}-(mv)_t \qquad [3.30]$$

where $(mv)_{t+dt}$ and $(mv)_t$ are the average momentums at times t and $t+dt$, respectively. The first term of the right side expression of [3.30] can be expressed as

$$(mv)_t = \rho A_{ave_t} v_{ave_t} dx \qquad [3.31]$$

where A_{ave_t} and v_{ave_t} are the average cross sectional area of flow and average flow velocity, respectively, at time t. These two terms can be evaluated separately as shown below

$$\begin{aligned} A_{ave_t} &= \frac{A_{1(t)} + A_{2(t)}}{2} \\ &= \frac{(A) + \left(A + \frac{\partial A}{\partial x} dx\right)}{2} \\ &= A + \frac{1}{2}\frac{\partial A}{\partial x} dx \end{aligned} \qquad [3.32]$$

and

$$\begin{aligned} v_{ave_t} &= \frac{v_{1(t)} + v_{2(t)}}{2} \\ &= \frac{(v) + \left(v + \frac{\partial v}{\partial x} dx\right)}{2} \\ &= v + \frac{1}{2}\frac{\partial v}{\partial x} dx \end{aligned} \qquad [3.33]$$

Substituting the above two terms in [3.31] results in

$$\begin{aligned} (mv)_t &= \rho\left(A + \frac{1}{2}\frac{\partial A}{\partial x} dx\right)\left(v + \frac{1}{2}\frac{\partial v}{\partial x} dx\right) dx \\ &= \rho\left(Av + \frac{A}{2}\frac{\partial v}{\partial x} dx^2 + \frac{v}{2}\frac{\partial A}{\partial x} dx^2 + \frac{1}{4}\frac{\partial A}{\partial x}\frac{\partial v}{\partial x} dx^3\right) \\ &= \rho A v dx \end{aligned} \qquad [3.34]$$

where both second and third order differential products are assumed negligible. The second term of the right hand side expression of Equation [3.30] can be evaluated as

$$(mv)_{t+dt} = \rho A_{ave_{t+dt}} v_{ave_{t+dt}} dx \qquad [3.35]$$

where $A_{ave_{t+dt}}$ and $v_{ave_{t+dt}}$ are is the average cross sectional area of flow and the average flow velocity, respectively, at time $t+dt$. These two terms can be evaluated individually as

$$\begin{aligned} A_{ave_{t+dt}} &= \frac{A_{1(t+dt)} + A_{2(t+dt)}}{2} \\ &= \frac{1}{2}\left[\left(A + \frac{\partial A}{\partial t}dt\right) + \left(\left(A + \frac{\partial A}{\partial x}dx\right) + \frac{\partial\left(A + \frac{\partial A}{\partial x}dx\right)}{\partial t}dt\right)\right] \\ &= A + \frac{1}{2}\frac{\partial A}{\partial t}dt + \frac{1}{2}\frac{\partial A}{\partial x}dx + \frac{1}{2}\frac{\partial A}{\partial t}dt + \frac{1}{2}\frac{\partial^2 A}{\partial x \partial t}dxdt \\ &= A + \frac{\partial A}{\partial t}dt + \frac{1}{2}\frac{\partial A}{\partial x}dx + \frac{1}{2}\frac{\partial^2 A}{\partial x \partial t}dxdt \end{aligned}$$

and

$$\begin{aligned} v_{ave_{t+dt}} &= \frac{v_{1(t+dt)} + A_{2(t+dt)}}{2} \\ &= \frac{1}{2}\left[\left(v + \frac{\partial v}{\partial t}dt\right) + \left(\left(v + \frac{\partial v}{\partial x}dx\right) + \frac{\partial\left(v + \frac{\partial v}{\partial x}dx\right)}{\partial t}dt\right)\right] \\ &= v + \frac{1}{2}\frac{\partial v}{\partial t}dt + \frac{1}{2}\frac{\partial v}{\partial x}dx + \frac{1}{2}\frac{\partial v}{\partial t}dt + \frac{1}{2}\frac{\partial^2 v}{\partial x \partial t}dxdt \\ &= v + \frac{\partial v}{\partial t}dt + \frac{1}{2}\frac{\partial v}{\partial x}dx + \frac{1}{2}\frac{\partial^2 v}{\partial x \partial t}dxdt \end{aligned}$$

Substituting the above two terms in [3.35] produces

$$(mv)_{t+dt} = \rho\left(A + \frac{\partial A}{\partial t}dt + \frac{1}{2}\frac{\partial A}{\partial x}dx + \frac{1}{2}\frac{\partial^2 A}{\partial x \partial t}dxdt\right)$$

$$\cdot\left(v + \frac{\partial v}{\partial t}dt + \frac{1}{2}\frac{\partial v}{\partial x}dx + \frac{1}{2}\frac{\partial^2 v}{\partial x \partial t}dxdt\right)dx$$

$$= \rho\left(Avdx + A\frac{\partial v}{\partial t}dtdx + \frac{A}{2}\frac{\partial v}{\partial x}dx^2 + \frac{A}{2}\frac{\partial^2 v}{\partial x \partial t}dx^2dt\right)$$

$$+\rho\left(\frac{v\partial A}{\partial t}dtdx + \frac{\partial A}{\partial t}\frac{\partial v}{\partial t}dt^2dx + \frac{1}{2}\frac{\partial A}{\partial t}\frac{\partial v}{\partial x}dx^2dt\right)$$

$$+\rho\left(\frac{1}{2}\frac{\partial A}{\partial t}\frac{\partial^2 v}{\partial x \partial t}dx^2dt^2 + \frac{v}{2}\frac{\partial A}{\partial x}dx^2 + \frac{1}{2}\frac{\partial A}{\partial x}\frac{\partial v}{\partial t}dx^2dt + \frac{1}{4}\frac{\partial A}{\partial x}\frac{\partial v}{\partial x}dx^3\right)$$

$$+\rho\left(\frac{1}{4}\frac{\partial A}{\partial x}\frac{\partial^2 v}{\partial x \partial t}dx^3dt + \frac{v}{2}\frac{\partial^2 A}{\partial x \partial t}dx^2dt + \frac{1}{2}\frac{\partial v}{\partial t}\frac{\partial^2 A}{\partial x \partial t}dx^2dt^2\right)$$

$$+\rho\left(\frac{1}{4}\frac{\partial v}{\partial x}\frac{\partial^2 A}{\partial x \partial t}dx^3dt + \frac{1}{4}\frac{\partial^2 A}{\partial x \partial t}\frac{\partial^2 v}{\partial x \partial t}dx^3dt^2\right) \qquad [3.36]$$

When the second, third, and fourth order differentials are assumed negligible, [3.36] reduces to

$$(mv)_{t+dt} = \rho\left(Avdx + A\frac{\partial v}{\partial t}dtdx + v\frac{\partial A}{\partial t}dtdx\right) \qquad [3.37]$$

Substituting [3.34] and [3.37] in [3.30] results in

$$d(mv) = \left(\rho Avdx + \rho A\frac{\partial v}{\partial t}dtdx + \rho v\frac{\partial A}{\partial t}dtdx\right) - (\rho Avdx)$$

or

$$\frac{d(mv)}{dt} = \rho A\frac{\partial v}{\partial t}dx + \rho v\frac{\partial A}{\partial t}dx \qquad [3.38]$$

The individual relationships that were developed to this point as presented by [3.27], [3.29], and [3.38] are substituted in [3.26] to yield

$$\Sigma(Forces) = \left(\rho Q v + \rho v \frac{\partial Q}{\partial x} dx + \rho Q \frac{\partial v}{\partial x} dx + \frac{\rho v dt}{2} \frac{\partial Q}{\partial t} + \frac{\rho Q dt}{2} \frac{\partial v}{\partial t}\right)$$

$$- \left(\rho Q v + \frac{\rho v dt}{2} \frac{\partial Q}{\partial t} + \frac{\rho Q dt}{2} \frac{\partial v}{\partial t}\right)$$

$$+ \left(\rho A \frac{\partial v}{\partial t} dx + \rho v \frac{\partial A}{\partial t} dx\right)$$

$$= \rho v \frac{\partial Q}{\partial x} dx + \rho Q \frac{\partial v}{\partial x} dx + \rho A \frac{\partial v}{\partial t} dx + \rho v \frac{\partial A}{\partial t} dx$$

$$= \rho \left(v \frac{\partial Q}{\partial x} + Q \frac{\partial v}{\partial x} + A \frac{\partial v}{\partial t} + v \frac{\partial A}{\partial t}\right) dx$$

or

$$F_{wx} + F_p - F_f = \rho \left(v \frac{\partial Q}{\partial x} + Q \frac{\partial v}{\partial x} + A \frac{\partial v}{\partial t} + v \frac{\partial A}{\partial t}\right) dx \qquad [3.39]$$

Substituting the expressions for F_{wx}, F_p, and F_f (i.e., Equations [3.15], [3.21], and [3.25]) in [3.39] results in

$$\gamma A S_0 dx - \gamma A \frac{\partial y}{\partial x} dx - \gamma A S_f dx = \rho \left(v \frac{\partial Q}{\partial x} + Q \frac{\partial v}{\partial x} + A \frac{\partial v}{\partial t} + v \frac{\partial A}{\partial t}\right) dx \qquad [3.40]$$

The specific weight, γ, may then be replaced by ρg where g is the acceleration due to gravity. The next step will be to divide [3.40] by $\rho g A dx$ to produce

$$S_0 - \frac{\partial y}{\partial x} - S_f = \frac{1}{gA} \left(v \frac{\partial Q}{\partial x} + Q \frac{\partial v}{\partial x} + A \frac{\partial v}{\partial t} + v \frac{\partial A}{\partial t}\right) \qquad [3.41]$$

or

$$S_0 - S_f = \frac{\partial y}{\partial x} + \frac{v}{gA} \frac{\partial Q}{\partial x} + \frac{Q}{gA} \frac{\partial v}{\partial x} + \frac{1}{g} \frac{\partial v}{\partial t} + \frac{v}{gA} \frac{\partial A}{\partial t} \qquad [3.42]$$

Since the discharge rate of flow, Q, is a preferable term over the flow velocity, v, Equation [3.42] can be rewritten as a function of Q instead of v based on the expression

$$\frac{\partial Q}{\partial \xi}=\frac{\partial (vA)}{\partial \xi}$$
$$=A\frac{\partial v}{\partial \xi}+v\frac{\partial A}{\partial \xi} \qquad [3.43]$$

where ξ is the independent variable of differentiation which represents either x or t. Equation [3.43] can be rearranged into

$$\frac{\partial v}{\partial \xi}=\left(\frac{\partial Q}{\partial \xi}-v\frac{\partial A}{\partial \xi}\right)\frac{1}{A}$$
$$=\left(\frac{\partial Q}{\partial \xi}-\left(\frac{Q}{A}\right)\frac{\partial A}{\partial \xi}\right)\frac{1}{A}$$
$$=\frac{1}{A^2}\left(A\frac{\partial Q}{\partial \xi}-Q\frac{\partial A}{\partial \xi}\right) \qquad [3.44]$$

Substituting Equation [3.44] in [3.42] results in

$$S_0-S_f=\frac{\partial y}{\partial x}+\frac{(Q/A)}{gA}\frac{\partial Q}{\partial x}+\frac{Q}{gA}\left[\frac{1}{A^2}\left(A\frac{\partial Q}{\partial x}-Q\frac{\partial A}{\partial x}\right)\right]$$
$$+\frac{1}{g}\left[\frac{1}{A^2}\left(A\frac{\partial Q}{\partial t}-Q\frac{\partial A}{\partial t}\right)\right]+\frac{(Q/A)}{gA}\frac{\partial A}{\partial t}$$
$$=\frac{\partial y}{\partial x}+\frac{2Q}{gA^2}\frac{\partial Q}{\partial x}-\frac{Q^2}{gA^3}\frac{\partial A}{\partial x}+\frac{1}{gA}\frac{\partial Q}{\partial t}$$

Since both the cross sectional are of flow, A, and the flow depth, y, are exclusively independent variables, one of these variables can be used instead of both. If the channel is assumed to be prismatic, the term $\partial A/\partial x$ in the above equation can be replaced with $T\partial y/\partial x$. The resultant equation is

$$S_0 - S_f = \left(1 - \frac{Q^2 T}{gA^3}\right)\frac{\partial y}{\partial x} + \frac{2Q}{gA^2}\frac{\partial Q}{\partial x} + \frac{1}{gA}\frac{\partial Q}{\partial t} \qquad [3.45]$$

Equation [3.45] can be rewritten exclusively in terms of A instead of both A and y. This step results in

$$S_0 - S_f = \left(\frac{1}{T} - \frac{Q^2}{gA^3}\right)\frac{\partial A}{\partial x} + \frac{2Q}{gA^2}\frac{\partial Q}{\partial x} + \frac{1}{gA}\frac{\partial Q}{\partial t} \qquad [3.46]$$

d. Steady Momentum

If a steady momentum is assumed, the change in momentum within the fluid element with respect to the time domain is assumed negligible. This reduces Equation [3.26] to

$$\Sigma(Forces) = (\rho Q v)_{out} - (\rho Q v)_{in} \qquad [3.47]$$

Substituting the results that were obtained earlier as represented by Equations [3.27] and [3.29] in Equation [3.47] leads to

$$\begin{aligned}\Sigma(Forces) &= \left(\rho Q v + \rho v \frac{\partial Q}{\partial x}dx + \rho Q \frac{\partial v}{\partial x}dx + \frac{\rho v dt}{2}\frac{\partial Q}{\partial t} + \frac{\rho Q dt}{2}\frac{\partial v}{\partial t}\right) \\ &\quad - \left(\rho Q v + \frac{\rho v dt}{2}\frac{\partial Q}{\partial t} + \frac{\rho Q dt}{2}\frac{\partial v}{\partial t}\right) \\ &= \rho v \frac{\partial Q}{\partial x}dx + \rho Q \frac{\partial v}{\partial x}dx \\ &= \rho\left(v\frac{\partial Q}{\partial x} + Q\frac{\partial v}{\partial x}\right)dx \end{aligned} \qquad [3.48]$$

The expressions that correspond to the various forces that are acting on the fluid element as represented Equations [3.15], [3.21], and [3.25] are then substituted in the equation above to produce

$$\gamma A S_0 dx - \gamma A \frac{\partial y}{\partial x} dx - \gamma A S_f dx = \rho \left[v \frac{\partial Q}{\partial x} + Q \frac{\partial v}{\partial x} \right] dx \qquad [3.49]$$

Following the same step as discussed in the previous section which include the replacement of the specific weight, γ, by ρg and the division of [3.49] by $\rho g A dx$ gives

$$S_0 - \frac{\partial y}{\partial x} - S_f = \frac{1}{gA} \left(v \frac{\partial Q}{\partial x} + Q \frac{\partial v}{\partial x} \right)$$

or

$$S_0 - S_f = \frac{\partial y}{\partial x} + \frac{v}{gA} \frac{\partial Q}{\partial x} + \frac{Q}{gA} \frac{\partial v}{\partial x} \qquad [3.50]$$

The change in flow velocity with respect to x, $\partial v/\partial x$, in the above equation is then replaced by the expression of [3.44] to produce

$$\begin{aligned} S_0 - S_f &= \frac{\partial y}{\partial x} + \frac{(Q/A)}{gA} \frac{\partial Q}{\partial x} + \frac{Q}{gA} \left[\frac{1}{A^2} \left(A \frac{\partial Q}{\partial x} - Q \frac{\partial A}{\partial x} \right) \right] \\ &= \frac{\partial y}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} - \frac{Q^2}{gA^3} \frac{\partial A}{\partial x} \end{aligned} \qquad [3.51]$$

Since the channel is assumed to be prismatic, the term $\partial A/\partial x$ can be exchanged by $T \partial y/\partial x$. Equation [3.51] reduces to

$$S_0 - S_f = \left(1 - \frac{Q^2 T}{gA^3} \right) \frac{\partial y}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} \qquad [3.52]$$

Equation [3.52] can be rewritten in terms of A instead of y. The resultant equation is

$$S_0 - S_f = \left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} \qquad [3.53]$$

2. Finite Element Formulation Using Linear Elements

The numerical solution of the Saint-Venant equations will be presented in this section. This analysis is accomplished using the Galerkin formulation of the finite element method. The space dimension is discretized using linear elements. This development is prepared for the solution of furrow irrigation systems, bearing in mind that the border is a special case of the furrow irrigation problem.

a. Formulation of the Continuity Equation Using Linear Elements

A system of linear equations is generated by evaluating the weighted residual integral

$$R(x)=\int_{x_i}^{x_j} W(x)\left(\frac{\partial A}{\partial t}+\frac{\partial Q}{\partial x}+I\right)dx \qquad [3.54]$$

which is the result of integrating the product of the continuity equation ([3.9]) and a weighting function, $W(x)$, over the length of the element. The Galerkin formulation of [3.54] is based on considering the shape functions N_i and N_j as the weighting functions at nodes i and j, respectively (Segerlind, 1984). Since the element selected in Figure 3 is linear, there will be two linear equations (shape functions) for each element.

The finite element formulation is applied after representing the unknowns by linear approximations of the form

$$\phi^{(e)}=N_i\Phi_i+N_j\Phi_j \qquad [3.55]$$

or

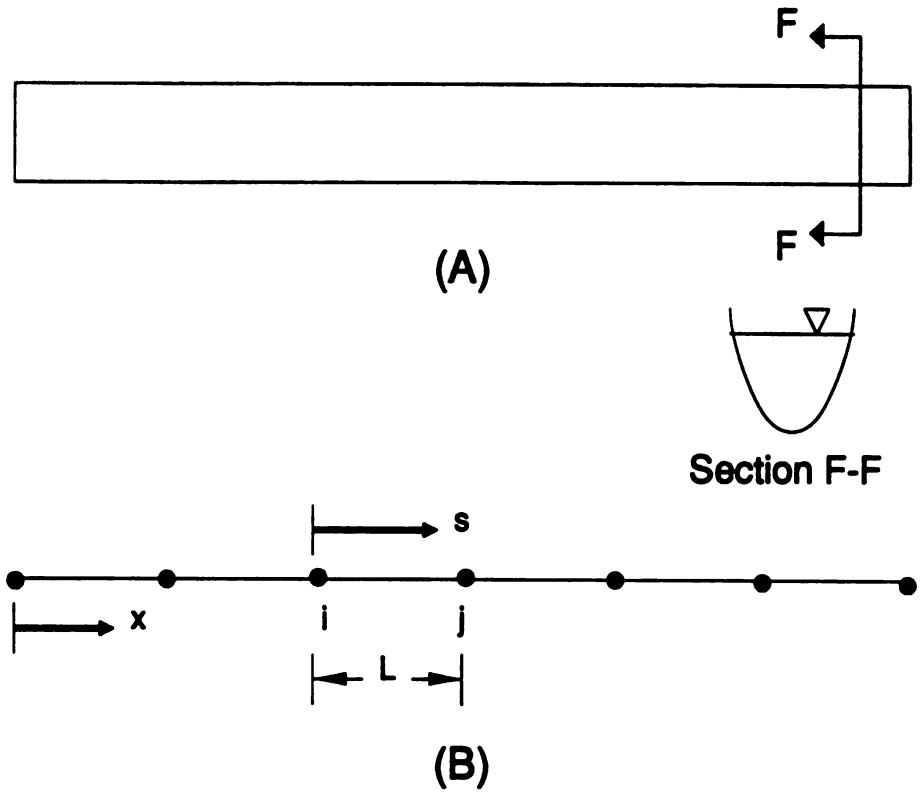


Figure 3. Finite element discretization. (A) Furrow flow. (B) Generic one-dimensional linear element.

$$\phi^{(e)} = [N_i \quad N_j]\begin{Bmatrix}\Phi_i \\ \Phi_j\end{Bmatrix}$$

$$= [N]\{\Phi\} \qquad [3.56]$$

where ϕ is the unknown; N_i and N_j are the shape functions at nodes i and j, respectively; and Φ_i and Φ_j are the values of the unknowns at nodes i and j, respectively. The convention that { } and [ ] represent a vector and matrix quantities, respectively, will be followed throughout this development.

Based on [3.56], the cross sectional area of flow, A, and the flow rate, Q, can be represented by

$$A(x,t) = N_i A_i(t) + N_j A_j(t)$$

$$= [N_i \quad N_j]\begin{Bmatrix}A_i \\ A_j\end{Bmatrix}$$

$$= [N]\{A(t)\} \qquad [3.57]$$

and

$$Q(x,t) = N_i Q_i(t) + N_j Q_j(t)$$

$$= [N_i \quad N_j]\begin{Bmatrix}Q_i \\ Q_j\end{Bmatrix}$$

$$= [N]\{Q(t)\} \qquad [3.58]$$

The shape functions for the one-dimensional linear element are expressed as

$$N_i = \frac{X_j - x}{L} \qquad [3.59]$$

and

$$N_j = \frac{x - X_i}{L} \qquad [3.60]$$

where L is the length of the element.

The shape functions are written in the local system of coordinates which allows for easier integration over the element. The shape functions for a coordinate system located at node i are obtained from [3.59] and [3.60] by replacing x with $X_i + s$ and resulting in

$$N_i = \frac{X_j - (X_i + s)}{L} = \frac{L - s}{L}$$

$$= 1 - \frac{s}{L} \qquad [3.61]$$

and

$$N_j = \frac{(X_i + s) - X_i}{L}$$

$$= \frac{s}{L} \qquad [3.62]$$

Equations [3.61] and [3.62] can be rewritten in the matrix form

$$[N] = [N_i \quad N_j]$$

$$= \left[1 - \frac{s}{L} \quad \frac{s}{L} \right] \qquad [3.63]$$

The partial derivative of [3.58] with respect to x is computed by evaluating the partial derivatives of N_i and N_j since the nodal values $Q_i(t)$ and $Q_j(t)$ are constants with respect to the $x-$ or $s-$space dimension. The partial derivative of [3.58] becomes

$$\frac{\partial Q}{\partial x}=\frac{\partial N_i}{\partial x}Q_i+\frac{\partial N_j}{\partial x}Q_j$$

$$=\begin{bmatrix}\frac{\partial N_i}{\partial x} & \frac{\partial N_j}{\partial x}\end{bmatrix}\begin{Bmatrix}Q_i\\ Q_j\end{Bmatrix}$$

$$=[B]\{Q\} \qquad [3.64]$$

The derivatives of the shape functions with respect to the $x-$ or $s-$space dimension are

$$\frac{\partial N_i}{\partial x}=\frac{\partial N_i}{\partial s}=-\frac{1}{L} \qquad [3.65]$$

and

$$\frac{\partial N_j}{\partial x}=\frac{\partial N_j}{\partial s}=\frac{1}{L} \qquad [3.66]$$

Equations [3.65] and [3.66] then rewritten in the matrix form

$$[B]=\begin{bmatrix}\frac{\partial N_i}{\partial x} & \frac{\partial N_j}{\partial x}\end{bmatrix}$$

$$=\begin{bmatrix}-\frac{1}{L} & \frac{1}{L}\end{bmatrix} \qquad [3.67]$$

Substituting [3.67] in [3.64] produces

$$\frac{\partial Q}{\partial x}=[B]\{Q\}=\begin{bmatrix}-\frac{1}{L} & \frac{1}{L}\end{bmatrix}\begin{Bmatrix}Q_i\\ Q_j\end{Bmatrix} \qquad [3.68]$$

The partial derivative of [3.57] with respect to time is computed by taking the partial derivatives of A_i and A_j with respect to time and multiplying the results by the shape functions. This assumption is true since the latter are considered to be constant with respect to time. The resultant equation is

$$\frac{\partial A}{\partial t} = N_i \frac{\partial A_i}{\partial t} + N_j \frac{\partial A_j}{\partial t}$$

$$= \begin{bmatrix} 1 - \frac{s}{L} & \frac{s}{L} \end{bmatrix} \begin{Bmatrix} \frac{\partial A_i}{\partial t} \\ \frac{\partial A_j}{\partial t} \end{Bmatrix}$$

$$= [N]\{\dot{A}\} \qquad [3.69]$$

Substituting the shape functions in [3.54] results in the system of equations

$$R_i^{(e)} = \int_{x_i}^{x_j} N_i \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} + I \right) dx \qquad [3.70]$$

and

$$R_j^{(e)} = \int_{x_i}^{x_j} N_j \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} + I \right) dx \qquad [3.71]$$

Rewriting Equations [3.70] and [3.71] in matrix form results in

$$\{R_e^{(e)}\} = \begin{Bmatrix} R_i^{(e)} \\ R_j^{(e)} \end{Bmatrix} = \begin{Bmatrix} \int_{x_i}^{x_j} N_i \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} + I \right) dx \\ \int_{x_i}^{x_j} N_j \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} + I \right) dx \end{Bmatrix}$$

$$= \int_{x_i}^{x_j} [N]^T \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} + I \right) dx \qquad [3.72]$$

where $[N]^T$ is the transpose of matrix $[N]$.

Substituting [3.68] and [3.69] in [3.72] yields

$$\{R_e^{(e)}\} = \int_{x_i}^{x_j} [N]^T ([N]\{\dot{A}\} + [B]\{Q\} + I)dx$$

or

$$\{R_e^{(e)}\} = \int_{x_i}^{x_j} [N]^T [N]\{\dot{A}\}dx + \int_{x_i}^{x_j} [N]^T [B]\{Q\}dx + \int_{x_i}^{x_j} [N]^T I\,dx \qquad [3.73]$$

The individual terms of [3.73] are integrated separately over the length of the element in the local system of coordinates to result in

$$\int_{x_i}^{x_j} [N]^T [N]\{\dot{A}\}dx = \int_0^L [N]^T [N]ds\{\dot{A}\}$$

$$= \int_0^L \begin{Bmatrix} 1-\frac{s}{L} \\ \frac{s}{L} \end{Bmatrix} \begin{bmatrix} 1-\frac{s}{L} & \frac{s}{l} \end{bmatrix} ds\{\dot{A}\}$$

$$= \int_0^L \begin{bmatrix} \left(1-\frac{s}{L}\right)^2 & \left(1-\frac{s}{L}\right)\frac{s}{L} \\ \left(1-\frac{s}{L}\right)\frac{s}{L} & \frac{s^2}{L^2} \end{bmatrix} ds\{\dot{A}\}$$

$$= \begin{bmatrix} \frac{L}{3} & \frac{L}{6} \\ \frac{L}{6} & \frac{L}{3} \end{bmatrix} \{\dot{A}\}$$

$$= \frac{L}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \{\dot{A}\}$$

$$= [C_e]\{\dot{A}\} \qquad [3.74]$$

and

$$\int_{x_i}^{x_j}[N]^T[B]\{Q\}dx = \int_0^L [N]^T[B]ds\{Q\}$$

$$= \int_0^L \begin{Bmatrix} 1-\frac{s}{L} \\ \frac{s}{L} \end{Bmatrix} \begin{bmatrix} -\frac{1}{L} & \frac{1}{L} \end{bmatrix} ds\{Q\}$$

$$= \int_0^L \begin{bmatrix} \left(-\frac{1}{L}+\frac{s}{L^2}\right) & \left(\frac{1}{L}-\frac{s}{L^2}\right) \\ -\frac{s}{L^2} & \frac{s}{L^2} \end{bmatrix} ds\{Q\}$$

$$= \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \{Q\}$$

$$= [K_e]\{Q\} \qquad [3.75]$$

The infiltration from the element can be expressed in various forms. These forms result in several equations that were discussed in the previous chapter. Many of these equations are expressed as a function of infiltration opportunity time only. The infiltration term is then constant with respect to the $x-$ or $s-$space dimension and can be moved outside the integral. This results in

$$\int_{x_i}^{x_j}[N]^T I dx = I\int_0^L [N]^T ds = I\int_0^L \begin{Bmatrix} 1-\frac{s}{L} \\ \frac{s}{L} \end{Bmatrix} ds$$

$$= I\begin{Bmatrix} \frac{L}{2} \\ \frac{L}{2} \end{Bmatrix} = -I\begin{Bmatrix} -\frac{L}{2} \\ -\frac{L}{2} \end{Bmatrix}$$

$$= -I\{F_e\} \qquad [3.76]$$

Substituting [3.74], [3.75], and [3.76] in [3.73] results in

$$\{R_e^{(e)}\} = \frac{L}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}\begin{Bmatrix} \dot{A}_i \\ \dot{A}_j \end{Bmatrix} + \begin{bmatrix} -\frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}\begin{Bmatrix} Q_i \\ Q_j \end{Bmatrix} - I\begin{Bmatrix} -\frac{L}{2} \\ -\frac{L}{2} \end{Bmatrix} \qquad [3.77]$$

or

$$\{R_e^{(e)}\} = [C_e]\{\dot{A}\} + [K_e]\{Q\} - I\{F_e\} \qquad [3.78]$$

b. Formulation of the Unsteady Momentum Equation Using Linear Elements

A system of equations is generated by evaluating the following weighted residual integral for the linear element

$$R(x) = \int_{x_i}^{x_j} W(x)\left[\left(\frac{1}{T} - \frac{Q^2}{gA^3}\right)\frac{\partial A}{\partial x} + \frac{2Q}{gA^2}\frac{\partial Q}{\partial x} + \frac{1}{gA}\frac{\partial Q}{\partial t} - (S_0 - S_f)\right]dx \qquad [3.79]$$

which is the result of integrating the product of the unsteady momentum equation, [3.46], and a weighting function, $W(x)$, over the length of the element. Based on the previous discussion, the Galerkin formulation of [3.79] utilizes the shape functions N_i and N_j as the weighting functions at nodes i and j, respectively.

Substituting the shape functions in [3.79] and rewriting the results in a matrix form produces

$$\{R_m^{(e)}\} = \begin{Bmatrix} R_i^{(e)} \\ R_j^{(e)} \end{Bmatrix} = \begin{Bmatrix} \int_{x_i}^{x_j} N_i \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} + \frac{1}{gA} \frac{\partial Q}{\partial t} - (S_0 - S_f) \right] dx \\ \int_{x_i}^{x_j} N_j \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} + \frac{1}{gA} \frac{\partial Q}{\partial t} - (S_0 - S_f) \right] dx \end{Bmatrix}$$

$$= \int_{x_i}^{x_j} [N]^T \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} + \frac{1}{gA} \frac{\partial Q}{\partial t} - (S_0 - S_f) \right] dx \qquad [3.80]$$

or

$$\{R_m^{(e)}\} = \int_{x_i}^{x_j} [N]^T \left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} dx + \int_{x_i}^{x_j} [N]^T \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} dx$$

$$+ \int_{x_i}^{x_j} [N]^T \frac{1}{gA} \frac{\partial Q}{\partial t} dx - \int_{x_i}^{x_j} [N]^T (S_0 - S_f) dx \qquad [3.81]$$

The partial derivative $\partial A/\partial x$ is determined by evaluating the derivative of [3.57] with respect to the space dimension. This is accomplished by evaluating the partial derivatives of N_i and N_j with respect to x since the nodal values $A_i(t)$ and $A_j(t)$ are constant with respect to the $x-$ or $s-$space dimension. This results in

$$\frac{\partial A}{\partial x} = \frac{\partial N_i}{\partial x} A_i + \frac{\partial N_j}{\partial x} A_j$$

$$= \begin{bmatrix} \frac{\partial N_i}{\partial x} & \frac{\partial N_j}{\partial x} \end{bmatrix} \begin{Bmatrix} A_i \\ A_j \end{Bmatrix}$$

$$= [B]\{A\} \qquad [3.82]$$

The partial derivative $\partial Q/\partial x$ was developed earlier as expressed by Equation [3.68].

The partial derivative $\partial Q/\partial t$ is determined by evaluating the derivative of [3.58] with respect to time. This is accomplished by evaluating the partial derivatives of Q_i and Q_j with respect to time and multiplying the results by the shape functions N_i and N_j that are constant with respect to time. This results in

$$\begin{aligned}\frac{\partial Q}{\partial t} &= N_i\frac{\partial Q_i}{\partial t}+N_j\frac{\partial Q_j}{\partial t}\\ &= [N_i \quad N_j]\begin{Bmatrix}\frac{\partial Q_i}{\partial t}\\ \frac{\partial Q_j}{\partial t}\end{Bmatrix}\\ &= [N]\{\dot{Q}\}\end{aligned} \qquad [3.83]$$

Using [3.68], [3.82], and [3.83], the individual terms of Equation [3.81] are integrated individually as follows :

$$\begin{aligned}\int_{x_i}^{x_j}[N]^T\left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)\frac{\partial A}{\partial x}dx &= \int_0^L[N]^T\left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)[B]\{A\}ds\\ &= \left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)\int_0^L\begin{Bmatrix}1-\frac{s}{L}\\ \frac{s}{L}\end{Bmatrix}\begin{bmatrix}-\frac{1}{L} & \frac{1}{L}\end{bmatrix}ds\{A\}\\ &= \left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)\begin{bmatrix}-\frac{1}{2} & \frac{1}{2}\\ -\frac{1}{2} & \frac{1}{2}\end{bmatrix}\{A\}\\ &= \left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)[K_c]\{A\}\end{aligned} \qquad [3.84]$$

$$\int_{x_i}^{x_j}[N]^T\frac{2Q}{gA^2}\frac{\partial Q}{\partial x}dx=\int_0^L[N]^T\frac{2Q}{gA^2}[B]\{Q\}ds$$

$$=\frac{2Q}{gA^2}\int_0^L\begin{Bmatrix}1-\frac{s}{L}\\ \frac{s}{L}\end{Bmatrix}\begin{bmatrix}-\frac{1}{L} & \frac{1}{L}\end{bmatrix}ds\{Q\}$$

$$=\frac{2Q}{gA^2}\begin{bmatrix}-\frac{1}{2} & \frac{1}{2}\\ -\frac{1}{2} & \frac{1}{2}\end{bmatrix}\{Q\}$$

$$=\frac{2Q}{gA^2}[K_e]\{Q\} \qquad [3.85]$$

$$\int_{x_i}^{x_j}[N]^T\frac{1}{gA}\frac{\partial Q}{\partial t}dx=\int_0^L[N]^T\frac{1}{gA}[N]\{\dot{Q}\}ds$$

$$=\frac{1}{gA}\int_0^L\begin{Bmatrix}1-\frac{s}{L}\\ \frac{s}{L}\end{Bmatrix}\begin{bmatrix}1-\frac{s}{L} & \frac{s}{l}\end{bmatrix}ds\{\dot{Q}\}$$

$$=\frac{1}{gA}\begin{bmatrix}\frac{L}{3} & \frac{L}{6}\\ \frac{L}{6} & \frac{L}{3}\end{bmatrix}\{\dot{Q}\}$$

$$=\frac{1}{gA}[C_e]\{\dot{Q}\} \qquad [3.86]$$

$$\int_{x_1}^{x_2}[N]^T(S_0-S_f)dx=(S_0-S_f)\int_0^L[N]^T ds$$

$$=(S_0-S_f)\int_0^L\begin{Bmatrix}1-\frac{s}{L}\\ \frac{s}{L}\end{Bmatrix}ds$$

$$=(S_0-S_f)\begin{Bmatrix}\frac{L}{2}\\ \frac{L}{2}\end{Bmatrix}$$

$$=(S_0-S_f)\{F_c\} \qquad [3.87]$$

Substituting [3.84], [3.85], [3.86], and [3.87] in [3.81] produces

$$\{R_m^{(e)}\}=\left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)[K_c]\{A\}+\left(\frac{2Q}{gA^2}\right)[K_c]\{Q\}$$
$$+\left(\frac{1}{gA}\right)[C_c]\{\dot{Q}\}-(S_0-S_f)\{F_c\}$$
$$=\left(\frac{1}{gA}\right)[C_c]\{\dot{Q}\}+\left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)[K_c]\{A\}$$
$$+\left(\frac{2Q}{gA^2}\right)[K_c]\{Q\}-(S_0-S_f)\{F_c\} \qquad [3.88]$$

or

$$\{R_m^{(e)}\}=[C_m]\{\dot{Q}\}+[K_{mA}]\{A\}+[K_{mQ}]\{Q\}-\{F_m\} \qquad [3.89]$$

c. Formulation of the Steady Momentum Equation Using Linear Elements

The residual vector of the linear element for the steady momentum equation, [3.53], is developed in the similar way as the unsteady momentum equation. The result is

$$\{R_m^{(e)}\} = \begin{Bmatrix} R_i^{(e)} \\ R_j^{(e)} \end{Bmatrix} = \begin{Bmatrix} \int_{x_i}^{x_j} N_i \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} - (S_0 - S_f) \right] dx \\ \int_{x_i}^{x_j} N_j \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} - (S_0 - S_f) \right] dx \end{Bmatrix}$$

$$= \int_{x_i}^{x_j} [N]^T \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} - (S_0 - S_f) \right] dx \qquad [3.90]$$

or

$$\{R_m^{(e)}\} = \int_{x_i}^{x_j} [N]^T \left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} dx + \int_{x_i}^{x_j} [N]^T \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} dx$$

$$- \int_{x_i}^{x_j} [N]^T (S_0 - S_f) dx \qquad [3.91]$$

Substituting [3.84], [3.85], and [3.87] in [3.91] yields

$$\{R_m^{(e)}\} = \left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) [K_c] \{A\} + \left(\frac{2Q}{gA^2} \right) [K_c] \{Q\} - (S_0 - S_f) \{F_c\} \qquad [3.92]$$

or

$$\{R_m^{(e)}\} = [K_{mA}] \{A\} + [K_{mQ}] \{Q\} - \{F_m\} \qquad [3.93]$$

d. Formulation of the Zero-Inertia Equation Using Linear Elements

A special case of the momentum equation is the zero-inertia equation which is based on the assumption that the change in momentum is negligible. This assumption drastically reduces the momentum equation and greatly simplifies the process of numerical analysis. Based on the above assumption, [3.26] reduces to

$$\Sigma(Forces) = 0 \qquad [3.94]$$

Substituting the forces in [3.94] results in

$$F_{wx} + F_p - F_f = 0 \qquad [3.95]$$

Substituting [3.15], [3.21], and [3.25] in [3.95] produces

$$\gamma A S_0 dx - \gamma A \frac{\partial y}{\partial x} dx - \gamma A S_f dx = 0 \qquad [3.96]$$

or

$$S_0 - S_f = \frac{\partial y}{\partial x} \qquad [3.97]$$

or

$$\frac{1}{T}\frac{\partial A}{\partial x} - (S_0 - S_f) = 0 \qquad [3.98]$$

The residual vector of the linear element for the zero-inertia momentum equation, [3.98], can be developed in a similar way as discussed in the previous section. The result is

$$\{R_m^{(e)}\} = \begin{Bmatrix} R_i^{(e)} \\ R_j^{(e)} \end{Bmatrix} = \begin{Bmatrix} \int_{x_i}^{x_j} N_i \left(\frac{1}{T}\frac{\partial A}{\partial x} - (S_0 - S_f) \right) dx \\ \int_{x_i}^{x_j} N_j \left(\frac{1}{T}\frac{\partial A}{\partial x} - (S_0 - S_f) \right) dx \end{Bmatrix}$$

$$= \int_{x_i}^{x_j} [N]^T \left(\frac{1}{T}\frac{\partial A}{\partial x} - (S_0 - S_f) \right) dx$$

or

$$\{R_m^{(e)}\} = \int_{x_i}^{x_j} [N]^T \frac{1}{T}\frac{\partial A}{\partial x} dx - \int_{x_i}^{x_j} [N]^T (S_0 - S_f) dx \qquad [3.99]$$

The first term of [3.99] is developed based on Equation [3.82]. This results in

$$\int_{x_i}^{x_j} [N]^T \frac{1}{T}\frac{\partial A}{\partial x} dx = \int_0^L [N]^T \frac{1}{T}[B]\{A\} ds$$

$$= \frac{1}{T}\int_0^L \begin{Bmatrix} 1 - \frac{s}{L} \\ \frac{s}{L} \end{Bmatrix} \begin{bmatrix} -\frac{1}{L} & \frac{1}{L} \end{bmatrix} ds \{A\}$$

$$= \frac{1}{T}\begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \{A\}$$

$$= \frac{1}{T}[K_c]\{A\}$$

$$= [K_{sA}]\{A\} \qquad [3.100]$$

The second term of [3.99] was developed earlier in [3.87]. However, the friction slope, S_f, can be expressed as a function of flow rate, Q, and cross-sectional area of flow, A, using any of the uniform flow equations that were reviewed in Chapter II. If the Manning equation is selected, Equation [3.87] becomes

$$\int_{x_i}^{x_j}[N]^T(S_0-S_f)dx=(S_0-S_f)\begin{Bmatrix}\frac{L}{2}\\ \frac{L}{2}\end{Bmatrix}$$

$$=\frac{L}{2}\begin{Bmatrix}(S_0-S_f)_i\\ (S_0-S_f)_j\end{Bmatrix}$$

$$=\frac{L}{2}\begin{Bmatrix}S_{0i}-\left(\frac{nQ_i}{A_iR_i^{2/3}}\right)^2\\ S_{0j}-\left(\frac{nQ_j}{A_jR_j^{2/3}}\right)^2\end{Bmatrix}$$

$$=\frac{L}{2}\begin{Bmatrix}S_{0i}\\ S_{0j}\end{Bmatrix}-\frac{L}{2}\begin{Bmatrix}\left(\frac{nQ_i}{A_iR_i^{2/3}}\right)^2\\ \left(\frac{nQ_j}{A_jR_j^{2/3}}\right)^2\end{Bmatrix} \qquad [3.101]$$

Substituting [3.100] and [3.101] in [3.99] yields the system of equations

$$\{R_m^{(e)}\}=\frac{1}{T}\begin{bmatrix}-\frac{1}{2}&\frac{1}{2}\\ -\frac{1}{2}&\frac{1}{2}\end{bmatrix}\begin{Bmatrix}A_i\\ A_j\end{Bmatrix}+\begin{Bmatrix}\frac{L}{2}\frac{n^2Q_i^2}{A_i^2R_i^{4/3}}\\ \frac{L}{2}\frac{n^2Q_j^2}{A_j^2R_j^{4/3}}\end{Bmatrix}-\frac{L}{2}\begin{Bmatrix}S_{0i}\\ S_{0j}\end{Bmatrix} \qquad [3.102]$$

Equation [3.102] is rewritten in a form similar to that of the steady momentum formulation which is expressed by [3.93]. This results in

$$\{R_m^{(e)}\}=\frac{1}{T}\begin{bmatrix}-\frac{1}{2}&\frac{1}{2}\\ -\frac{1}{2}&\frac{1}{2}\end{bmatrix}\begin{Bmatrix}A_i\\ A_j\end{Bmatrix}+\begin{bmatrix}\frac{L}{2}\frac{n^2Q_i}{A_i^2R_i^{4/3}}&0\\ 0&\frac{L}{2}\frac{n^2Q_j}{A_j^2R_j^{4/3}}\end{bmatrix}\begin{Bmatrix}Q_i\\ Q_j\end{Bmatrix}-\frac{L}{2}\begin{Bmatrix}S_{0i}\\ S_{0j}\end{Bmatrix}$$

$$=\frac{1}{T}[K_c]\{A\}+[K_{sQ}]\{Q\}-S_0\{F_c\}$$

or

$$\{R_m^{(e)}\} = [K_{xA}]\{A\} + [K_{xQ}]\{Q\} - \{F_x\} \qquad [3.103]$$

e. Formulation of the Kinematic Wave Equation Using Linear Elements

Surface irrigation analysis can be performed based on both the continuity and momentum equations, or the so-called the Saint-Venant equations ([3.9] and [3.46]). Various assumptions are implemented to simplify the momentum equation ([3.46]) to the so-called kinematic wave approximation. This approximation is based on the assumption that the inertial terms together with the term that describes the pressure variation of flow in the momentum equation are negligible. The simplified momentum equation then has the form

$$S_0 - S_f = 0 \qquad [3.104]$$

while the continuity equation ([3.9]) is kept unchanged. These assumptions imply that surface flow is at normal depth throughout the domain of solution. Based on the above assumptions, Equation [3.104] is utilized in [2.11], [2.12], or [2.13] to calculate the flow rate, Q, at nodes i and j. If the Manning equation ([2.12]) is selected, the following equations result

$$Q_i = \frac{1}{n} A_i R_i^{2/3} S_{0i}^{1/2}$$

and

$$Q_j = \frac{1}{n} A_j R_j^{2/3} S_{0j}^{1/2}$$

or

$$\left(\frac{1}{n} R_i^{2/3} S_{0i}^{1/2}\right) A_i - Q_i = 0 \qquad [3.105]$$

and

$$\left(\frac{1}{n}R_j^{2/3}S_{0j}^{1/2}\right)A_j - Q_j = 0 \qquad [3.106]$$

where R_i and R_j are the hydraulic radii at nodes i and j, respectively. Since we are primarily interested in the contribution of nodes i and j to the element, only half of the flow terms in [3.105] and [3.106] will be utilized as the element contribution to the final system of equations. Equations [3.105] and [3.106] can be combined together in a matrix form to yield

$$\frac{1}{2n}\begin{bmatrix} R_i^{\frac{2}{3}}S_{0i}^{\frac{1}{2}} & 0 \\ 0 & R_j^{\frac{2}{3}}S_{0j}^{\frac{1}{2}} \end{bmatrix}\begin{Bmatrix} A_i \\ A_j \end{Bmatrix} + \begin{bmatrix} -\frac{1}{2} & 0 \\ 0 & -\frac{1}{2} \end{bmatrix}\begin{Bmatrix} Q_i \\ Q_j \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \qquad [3.107]$$

or

$$[K_{hA}]\{A\} + [K_{hQ}]\{Q\} - \{F_k\} = \{0\} \qquad [3.108]$$

where $\{F_k\} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}$.

f. General Finite Element Formulation Using Linear Elements

The linear one-dimensional finite element formulation of the continuity equation, [3.78], can be rewritten as

$$\{R_e^{(e)}\} = \frac{L}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}\begin{Bmatrix} \dot{A}_i \\ \dot{A}_j \end{Bmatrix} + \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix}\begin{Bmatrix} Q_i \\ Q_j \end{Bmatrix} - I\begin{Bmatrix} -\frac{L}{2} \\ -\frac{L}{2} \end{Bmatrix} \qquad [3.109]$$

On the other hand, [3.89], [3.93], [3.103], and [3.108] can be expressed in the general form

$$\{R_m^{(e)}\} = \frac{L}{6}\begin{bmatrix} 2c_1 & c_1 \\ c_1 & 2c_1 \end{bmatrix}\begin{Bmatrix} \dot{Q}_i \\ \dot{Q}_j \end{Bmatrix} + \begin{bmatrix} -\frac{(c_2-c_3)}{2}+\frac{c_i}{2} & \frac{(c_2-c_3)}{2} \\ -\frac{(c_2-c_3)}{2} & \frac{(c_2-c_3)}{2}+\frac{c_j}{2} \end{bmatrix}\begin{Bmatrix} A_i \\ A_j \end{Bmatrix}$$

$$+\begin{bmatrix} \frac{-c_4+c_0}{2} & \frac{c_4}{2} \\ -\frac{c_4}{2} & \frac{c_4+c_0}{2} \end{bmatrix}\begin{Bmatrix} Q_i \\ Q_j \end{Bmatrix} - \begin{Bmatrix} f_i \\ f_j \end{Bmatrix} \qquad [3.110]$$

where c_0, c_1, c_2, c_3, c_4, c_i, c_j, f_i, and f_j are coefficients that vary based on the selected model.

There are four different models that will result based on the previous discussion. These include

i. Hydrodynamic Model I : This model is the result of combining both the continuity equation and the unsteady momentum equation. The various coefficients in [3.110] may be expressed as

$$c_0 = 0, \quad c_1 = \left(\frac{1}{gA}\right), \quad c_2 = \left(\frac{1}{T}\right), \quad c_3 = \left(\frac{Q^2}{gA^3}\right), \quad c_4 = \left(\frac{2Q}{gA^2}\right),$$

$$c_i = 0, \quad c_j = 0, \quad f_i = \frac{L}{2}(S_0 - S_f)_i, \quad \text{and} \quad f_j = \frac{L}{2}(S_0 - S_f)_j .$$

ii. Hydrodynamic Model II : This model is the result of combining both the continuity equation and the steady momentum equation. The various coefficients in [3.110] may be expressed as

$$c_0 = 0, \quad c_1 = 0, \quad c_2 = \left(\frac{1}{T}\right), \quad c_3 = \left(\frac{Q^2}{gA^3}\right), \quad c_4 = \left(\frac{2Q}{gA^2}\right),$$

$$c_i = 0, \quad c_j = 0, \quad f_i = \frac{L}{2}(S_0 - S_f)_i, \quad \text{and} \quad f_j = \frac{L}{2}(S_0 - S_f)_j .$$

iii. Zero-Inertia Model : This model is the result of combining both the continuity equation and the zero-inertia assumptions that result in [3.98]. The various coefficients in [3.110] may be expressed as

$$c_0 = \frac{L}{2}\frac{n^2 Q}{A^2 R^{4/3}}, \qquad c_1 = 0, \qquad c_2 = \left(\frac{1}{T}\right), \qquad c_3 = 0, \qquad c_4 = 0,$$

$$c_i = 0, \qquad c_j = 0, \qquad f_i = \frac{L}{2} S_{0i}, \qquad \text{and} \qquad f_j = \frac{L}{2} S_{0j}.$$

iv. Kinematic Wave Model : This model is the result of combining both the continuity equation and the kinematic wave or the uniform flow assumptions that result in [3.108]. The various coefficients in [3.110] may be expressed as

$$c_0 = -1, \qquad c_1 = 0, \qquad c_2 = 0, \qquad c_3 = 0, \qquad c_4 = 0,$$

$$c_i = \frac{1}{n} R_i^{2/3} S_{0i}^{1/2}, \qquad c_j = \frac{1}{n} R_j^{2/3} S_{0j}^{1/2}, \qquad f_i = 0, \qquad \text{and} \qquad f_j = 0.$$

Equations [3.109] and [3.110] are solved simultaneously for every element to determine both the flow rate, Q, and the cross sectional area of flow, A, at each node. These equations are combined together to produce one system of equations for every element. The resultant system of equations has the general form

$$\{R^{(e)}\}=\begin{Bmatrix}R_{ci}\\R_{mi}\\R_{cj}\\R_{mj}\end{Bmatrix}=\frac{L}{6}\begin{bmatrix}2&0&1&0\\0&2c_1&0&c_1\\1&0&2&0\\0&c_1&0&2c_1\end{bmatrix}\begin{Bmatrix}\dot{A}_i\\\dot{Q}_i\\\dot{A}_j\\\dot{Q}_j\end{Bmatrix}$$

$$+\frac{1}{2}\begin{bmatrix}0&-1&0&1\\-c_2+c_3+c_i&-c_4+c_0&c_2-c_3&c_4\\0&-1&0&1\\-c_2+c_3&-c_4&c_2-c_3+c_j&c_4+c_0\end{bmatrix}\begin{Bmatrix}A_i\\Q_i\\A_j\\Q_j\end{Bmatrix}$$

$$-\begin{Bmatrix}\frac{-I_iL}{2}\\f_i\\\frac{-I_jL}{2}\\f_j\end{Bmatrix} \qquad [3.111]$$

or

$$\{R^{(e)}\}=[C]\{\dot{\Phi}\}+[K]\{\Phi\}-\{F\} \qquad [3.112]$$

3. Finite Element Formulation Using Quadratic Elements

The numerical solution of the Saint-Venant equations is repeated in this section. However, the analysis this time is accomplished based on the c sub Galerkin formulation of the finite element method and using quadratic element.

a. Formulation of the Continuity Equation Using Quadratic Elements

A system of equations is generated by evaluating the weighted residual integral

$$R(x)=\int_{x_i}^{x_k} W(x)\left(\frac{\partial A}{\partial t}+\frac{\partial Q}{\partial x}+I\right)dx \qquad [3.113]$$

which is the result of integrating the product of the continuity equation ([3.9]) and weighting function, $W(x)$, over the length of the quadratic element. The Galerkin formulation of [3.113] is based on considering the shape functions N_i, N_j, and N_k as the weighting functions at nodes i, j, and k, respectively. Since the element in Figure 4 is quadratic, there will be a system of three equations for every element.

The finite element formulation is applied after representing the unknowns by linear approximations of the form

$$\phi^{(e)}=N_i\Phi_i+N_j\Phi_j+N_k\Phi_k \qquad [3.114]$$

or

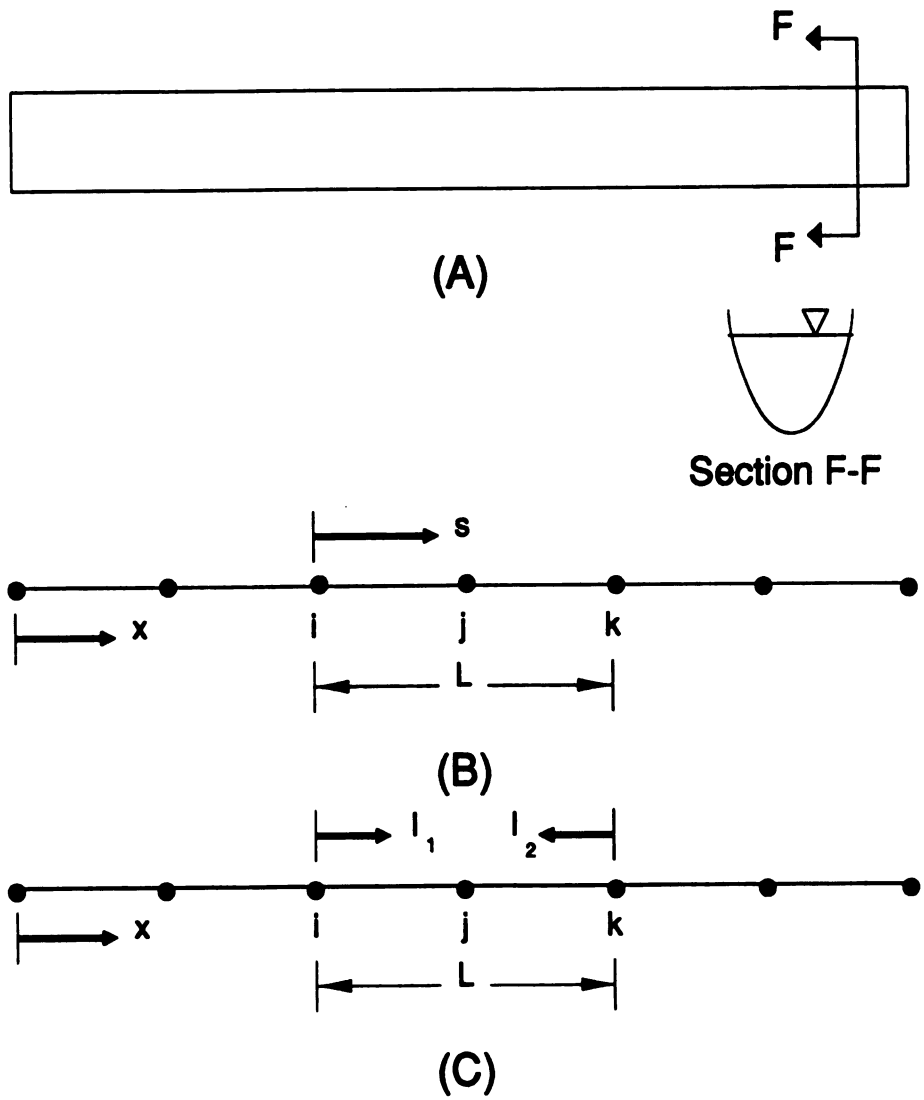


Figure 4. **Finite element discretization. (A) Furrow flow. (B) Generic quadratic element in the system of local coordinates. (C) Generic quadratic element in the system of natural coordinates.**

$$\phi^{(e)} = [N_i \quad N_j \quad N_k]\begin{Bmatrix}\Phi_i\\ \Phi_j\\ \Phi_k\end{Bmatrix}$$

$$= [N]\{\Phi\} \qquad [3.115]$$

where ϕ is the unknown; N_i, N_j, and N_k are the shape functions at nodes i, j, and k, respectively; and Φ_i, Φ_j, and Φ_k are the values of the unknowns at nodes i, j, and k, respectively.

Based on [3.115], the cross sectional area of flow, A, and the flow rate, Q, are expressed as

$$A(x,t) = N_iA_i(t) + N_jA_j(t) + N_k'A_k(t)$$

$$= [N_i \quad N_j \quad N_k]\begin{Bmatrix}A_i(t)\\ A_j(t)\\ A_k(t)\end{Bmatrix}$$

$$= [N]\{A(t)\} \qquad [3.116]$$

and

$$Q(x,t) = N_iQ_i(t) + N_jQ_j(t) + N_kQ_k(t)$$

$$= [N_i \quad N_j \quad N_k]\begin{Bmatrix}Q_i(t)\\ Q_j(t)\\ Q_k(t)\end{Bmatrix}$$

$$= [N]\{Q(t)\} \qquad [3.117]$$

The shape functions for the one-dimensional quadratic element are expressed by the following equations (Segerlind, 1984)

$$N_i = \frac{2}{L^2}(x - X_j)(x - X_k) \qquad [3.118]$$

$$N_j = -\frac{4}{L^2}(x - X_i)(x - X_k) \qquad [3.119]$$

$$N_k = \frac{2}{L^2}(x - X_i)(x - X_j) \qquad [3.120]$$

where L is the length of the element.

The shape functions can be written in the system of local coordinates which allow for easier integration over the element. The shape functions for a system of coordinates located at node i are obtained from [3.118], [3.119], and [3.120] by replacing x with $X_i + s$ (Figure 4a). This results in the following equations:

$$N_i = \frac{2}{L^2}(X_i + s - X_j)(X_i + s - X_k)$$

$$= \frac{2}{L^2}\left(s - \frac{L}{2}\right)(s - L) \qquad [3.121]$$

$$N_j = -\frac{4}{L^2}(X_i + s - X_i)(X_i + s - X_k)$$

$$= -\frac{4}{L^2}(s)(s - L) \qquad [3.122]$$

$$N_k = \frac{2}{L^2}(X_i + s - X_i)(X_i + s - X_j)$$

$$= \frac{2}{L^2}(s)\left(s - \frac{L}{2}\right) \qquad [3.123]$$

The shape functions are then rewritten in the system of natural coordinates which consist of a pair of length ratios as shown in Figure 4b. These ratios are defined as (Segerlind, 1984)

$$l_1 = \left(\frac{L - s}{L}\right) \quad \text{and} \quad l_2 = \left(\frac{s}{L}\right) \qquad [3.124]$$

where s is the distance from node i. The shape functions which are expressed by [3.121], [3.122], and [3.123] can be rewritten in the system of natural coordinates and the results are

$$\begin{aligned} N_i &= \left(\frac{2s-L}{L}\right)\left(\frac{s-L}{L}\right) \\ &= \left(\frac{s}{L}+\frac{s-L}{L}\right)\left(\frac{s-L}{L}\right) \\ &= (l_2-l_1)(-l_1) \\ &= l_1^2-l_1l_2 \end{aligned} \qquad [3.125]$$

$$\begin{aligned} N_j &= 4\left(\frac{s}{L}\right)\left(\frac{L-s}{L}\right) \\ &= 4l_1l_2 \end{aligned} \qquad [3.126]$$

$$\begin{aligned} N_k &= \left(\frac{s}{L}\right)\left(\frac{2s-L}{L}\right) \\ &= \left(\frac{s}{L}\right)\left(\frac{s}{L}+\frac{s-L}{L}\right) \\ &= l_2(l_2-l_1) \\ &= l_2^2-l_1l_2 \end{aligned} \qquad [3.127]$$

The system of natural coordinates is very essential for directly evaluating various integrals that contain the shape functions directly as will be highlighted in the subsequent discussion.

The partial derivative of Equation [3.117] with respect to x is computed by evaluating the partial derivatives of N_i, N_j, and N_k with respect to x since the nodal values $Q_i(t)$, $Q_j(t)$, and $Q_k(t)$ are assumed constant with respect to the x – or s –space dimension. The partial derivative of [3.117] becomes

$$\frac{\partial Q}{\partial x}=\frac{\partial N_i}{\partial x}Q_i+\frac{\partial N_j}{\partial x}Q_j+\frac{\partial N_k}{\partial x}Q_k$$

$$=\begin{bmatrix}\frac{\partial N_i}{\partial x} & \frac{\partial N_j}{\partial x} & \frac{\partial N_k}{\partial x}\end{bmatrix}\begin{Bmatrix}Q_i\\Q_j\\Q_k\end{Bmatrix}$$

$$=[B]\{Q\} \qquad [3.128]$$

The derivatives of the shape functions with respect to the $x-$ or $s-$space dimension are as follows:

$$\frac{\partial N_i}{\partial x}=\frac{\partial N_i}{\partial s}=\frac{2}{L^2}\left((s-L)+\left(s-\frac{L}{2}\right)\right)$$

$$=\frac{4}{L^2}\left(s-\frac{3L}{4}\right) \qquad [3.129]$$

$$\frac{\partial N_j}{\partial x}=\frac{\partial N_j}{\partial s}=-\frac{4}{L^2}((s-L)+s)$$

$$=-\frac{8}{L^2}\left(s-\frac{L}{2}\right) \qquad [3.130]$$

$$\frac{\partial N_k}{\partial x}=\frac{\partial N_k}{\partial s}=\frac{2}{L^2}\left(\left(s-\frac{L}{2}\right)+s\right)$$

$$=\frac{4}{L^2}\left(s-\frac{L}{4}\right) \qquad [3.131]$$

The derivatives of the shape functions which are expressed by [3.129], [3.130], and [3.131] is expressed in the system of natural coordinates as

$$\frac{\partial N_i}{\partial s}=\frac{1}{L}\left(\frac{4s-3L}{L}\right)$$

$$=\frac{1}{L}\left(\frac{3s}{L}+\frac{s-L}{L}-\frac{2L}{L}\right)$$

$$=\frac{1}{L}(3l_2-l_1-2) \qquad [3.132]$$

$$\frac{\partial N_j}{\partial s} = -\frac{4}{L}\left(\frac{2s-L}{L}\right)$$

$$= -\frac{4}{L}\left(\frac{s}{L}+\frac{s-L}{L}\right)$$

$$= -\frac{4}{L}(l_2 - l_1) \qquad [3.133]$$

$$\frac{\partial N_k}{\partial s} = \frac{1}{L}\left(\frac{4s-L}{L}\right)$$

$$= \frac{1}{L}\left(\frac{3s}{L}+\frac{s-L}{L}\right)$$

$$= \frac{1}{L}(3l_2 - l_1) \qquad [3.134]$$

The partial derivative of Equation [3.116] with respect to time is computed by evaluating the partial derivatives of the cross sectional are of flow, A, at nodes i, j, and k. The products of the resulting functions and the respective shape functions which are considered constant with respect to time produce

$$\frac{\partial A}{\partial t} = N_i\frac{\partial A_i}{\partial t} + N_j\frac{\partial A_j}{\partial t} + N_k\frac{\partial A_k}{\partial t}$$

$$= [N_i \quad N_j \quad N_k]\begin{Bmatrix}\frac{\partial A_i}{\partial t}\\ \frac{\partial A_j}{\partial t}\\ \frac{\partial A_k}{\partial t}\end{Bmatrix}$$

$$= [N]\{\dot{A}\} \qquad [3.135]$$

Substituting the shape functions in [3.135] results in the system of equations .

$$R_i^{(e)} = \int_{x_1}^{x_2} N_i \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} + I \right) dx \qquad [3.136]$$

$$R_j^{(e)} = \int_{x_1}^{x_2} N_j \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} + I \right) dx \qquad [3.137]$$

$$R_k^{(e)} = \int_{x_1}^{x_2} N_k \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} + I \right) dx \qquad [3.138]$$

Equations [3.137], [3.137], and [3.138] can be rewritten in the matrix form

$$\{R_e^{(e)}\} = \begin{Bmatrix} R_i^{(e)} \\ R_j^{(e)} \\ R_k^{(e)} \end{Bmatrix} = \begin{Bmatrix} \int_{x_1}^{x_2} N_i \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} + I \right) dx \\ \int_{x_1}^{x_2} N_j \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} + I \right) dx \\ \int_{x_1}^{x_2} N_k \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} + I \right) dx \end{Bmatrix}$$

$$= \int_{x_1}^{x_2} [N]^T \left(\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} + I \right) dx \qquad [3.139]$$

where $[N]^T$ is the transpose of matrix $[N]$.

Substituting [3.128] and [3.135] in [3.139] results in

$$\{R_e^{(e)}\} = \int_{x_1}^{x_2} [N]^T ([N]\{\dot{A}\} + [B]\{Q\} + I) dx$$

or

$$\{R_e^{(e)}\} = \int_{x_1}^{x_2} [N]^T [N] \{\dot{A}\} dx + \int_{x_1}^{x_2} [N]^T [B] \{Q\} dx + \int_{x_1}^{x_2} [N]^T I dx \qquad [3.140]$$

The individual terms of Equation [3.140] are then integrated over the length of the quadratic element in either system of local or natural coordinates. However, the system of natural coordinates is used in the development of this section since such a system simplifies the integration process immensely. The first right side term of Equation [3.140] is evaluated as

$$\int_{x_1}^{x_2} [N]^T [N] \{\dot{A}\} dx = \int_0^L [N]^T [N] ds \{\dot{A}\}$$

$$= \int_0^1 [N]^T [N] L dl_2 \{\dot{A}\}$$

$$= L \int_0^1 \begin{Bmatrix} l_1^2 - l_1 l_2 \\ 4 l_1 l_2 \\ l_2^2 - l_1 l_2 \end{Bmatrix} [l_1^2 - l_1 l_2 \quad 4 l_1 l_2 \quad l_2^2 - l_1 l_2] dl_2 \{\dot{A}\}$$

$$= L \int_0^1 \begin{bmatrix} l_1^4 - 2l_1^3 l_2 + l_1^2 l_2^2 & 4l_1^3 l_2 - 4l_1^2 l_2^2 & l_1^2 l_2^2 - l_1^3 l_2 - l_1 l_2^3 + l_1^2 l_2^2 \\ 4l_1^3 l_2 - 4l_1^2 l_2^2 & 16 l_1^2 l_2^2 & 4l_1 l_2^3 - 4l_1^2 l_2^2 \\ l_1^2 l_2^2 - l_1^3 l_2 - l_1 l_2^3 + l_1^2 l_2^2 & 4l_1 l_2^3 - 4l_1^2 l_2^2 & l_2^4 - 2l_1^3 l_2 + l_1^2 l_2^2 \end{bmatrix} dl_2 \{\dot{A}\} \qquad [3.141]$$

where $s = (L l_2)$, $\frac{ds}{dl_2} = L$, and $l_1 = (1 - l_2)$.

The various integrals in [3.141] are reduced to the form

$$\int_0^1 l_2^{z-1} (1 - l_2)^{(w-1)} dl_2 = \frac{\Gamma(z)\Gamma(w)}{\Gamma(z+w)} \qquad [3.142]$$

where $\Gamma(n+1) = n!$ (Abramowitz and Stegun, 1964). The integrals of the matrix in [3.141] are then evaluated separately as follows:

$$\int_0^1 (l_1^4 - 2l_1^3 l_2 + l_1^2 l_2^2) dl_2 = \int_0^1 l_1^4 l_2^0 dl_2 - 2\int_0^1 l_1^3 l_2^1 dl_2 + \int_0^1 l_1^2 l_2^2 dl_2$$

$$= \frac{4!0!}{(4+0+1)!} - 2\left(\frac{3!1!}{(3+1+1)!}\right) + \frac{2!2!}{(2+2+1)!}$$

$$= \frac{16}{120} \qquad [3.143]$$

$$\int_0^1 (4l_1^3 l_2 - 4l_1^2 l_2^2) dl_2 = 4\left(\frac{3!1!}{(3+1+1)!} - \frac{2!2!}{(2+2+1)!}\right)$$

$$= \frac{8}{120} \qquad [3.144]$$

$$\int_0^1 (l_1^2 l_2^2 - l_1^3 l_2 + l_1 l_2^3 + l_1^2 l_2^2) dl_2 = \frac{2!2!}{(2+2+1)!} - \frac{3!1!}{(3+1+1)!} - \frac{1!3!}{(1+3+1)!}$$

$$+ \frac{2!2!}{(2+2+1)!} = \frac{-4}{120} \qquad [3.145]$$

$$\int_0^1 (16 l_1^2 l_2^2) dl_2 = 16\left(\frac{2!2!}{(2+2+1)!}\right)$$

$$= \frac{64}{120} \qquad [3.146]$$

$$\int_0^1 (4l_1 l_2^3 - 4l_1^2 l_2^2) dl_2 = 4\left(\frac{1!3!}{(1+3+1)!} - \frac{2!2!}{(2+2+1)!}\right)$$

$$= \frac{8}{120} \qquad [3.147]$$

$$\int_0^1 (l_2^4 - 2l_1l_2^3 + l_1^2l_2^2)dl_2 = \frac{0!4!}{(0+4+1)!} - 2\left(\frac{1!3!}{(1+3+1)!}\right) + \frac{2!2!}{(2+2+1)!}$$

$$= \frac{16}{120} \qquad [3.148]$$

Substituting [3.143] through [3.148] in [3.141] results in

$$\int_{x_i}^{x_k} [N]^T [N] \{\dot{A}\} dx = \frac{L}{30} \begin{bmatrix} 4 & 2 & -1 \\ 2 & 16 & 2 \\ -1 & 2 & 4 \end{bmatrix} \begin{Bmatrix} \dot{A}_i \\ \dot{A}_j \\ \dot{A}_k \end{Bmatrix}$$

$$= [C_c]\{\dot{A}\} \qquad [3.149]$$

The second integral of [3.140] is evaluated using the same procedure. This results in

$$\int_{x_i}^{x_k} [N]^T [B] \{Q\} dx = \int_0^L [N]^T [B] ds \{Q\}$$

$$= \int_0^1 [N]^T [B] L dl_2 \{Q\}$$

$$= L \int_0^1 \begin{Bmatrix} l_1^2 - l_1l_2 \\ 4l_1l_2 \\ l_2^2 - l_1l_2 \end{Bmatrix} \begin{bmatrix} \frac{1}{L}(3l_2 - l_1 - 2) & \frac{1}{L}(-4l_2 + 4l_1) & \frac{1}{L}(3l_2 - l_1) \end{bmatrix} dl_2 \{Q\}$$

$$= \int_0^1 \begin{bmatrix} -l_1^3 + 4l_1^2l_2 - 3l_1l_2^2 - 2l_1^2 + 2l_1l_2 & -8l_1^2l_2 + 4l_1l_2^2 + 4l_1^3 & 4l_1^2l_2 - 3l_1l_2^2 - l_1^3 \\ 12l_1l_2^2 - 4l_1^2l_2 - 8l_1l_2 & -16l_1l_2^2 + 16l_1^2l_2 & 12l_1l_2^2 - 4l_1^2l_2 \\ 3l_2^3 - 4l_1l_2^2 + l_1^2l_2 - 2l_2^2 + 2l_1l_2 & -4l_2^3 + 8l_1l_2^2 - 4l_1^2l_2 & 3l_2^3 - 4l_1l_2^2 + l_1^2l_2 \end{bmatrix}$$

$$\cdot dl_2 \{Q\} \qquad [3.150]$$

where $s = (Ll_2)$, $\frac{ds}{dl_2} = L$, **and** $l_1 = (1 - l_2)$.

The integrals of the matrix in [3.150] can be evaluated separately as follows:

$$\int_0^1(-l_1^3+4l_1^2l_2-3l_1l_2^2-2l_1^2+2l_1l_2)dl_2=-\frac{3!0!}{(3+0+1)!}+4\left(\frac{2!1!}{(2+1+1)!}\right)$$

$$-3\left(\frac{1!2!}{(1+2+1)!}\right)-2\left(\frac{2!0!}{(2+0+1)!}\right)+2\left(\frac{1!1!}{(1+1+1)!}\right)$$

$$=\frac{-12}{24} \qquad [3.151]$$

$$\int_0^1(12l_1l_2^2-4l_1^2l_2-8l_1l_2)dl_2=12\left(\frac{1!2!}{(1+2+1)!}\right)-4\left(\frac{2!1!}{(2+1+1)!}\right)$$

$$-8\left(\frac{1!1!}{(1+1+1)!}\right)$$

$$=\frac{-16}{24} \qquad [3.152]$$

$$\int_0^1(-3l_2^3-4l_1l_2^2+l_1^2l_2-2l_2+2l_1l_2)dl_2=3\left(\frac{0!3!}{(0+3+1)!}\right)-4\left(\frac{1!2!}{(1+2+1)!}\right)$$

$$+\frac{2!1!}{(2+1+1)!}-2\left(\frac{0!2!}{(0+2+1)!}\right)+2\left(\frac{1!1!}{(1+1+1)!}\right)$$

$$=\frac{4}{24} \qquad [3.153]$$

$$\int_0^1(-8l_1^2l_2+4l_1l_2^2+4l_1^3)dl_2=-8\left(\frac{2!1!}{(2+1+1)!}\right)+4\left(\frac{1!2!}{(1+2+1)!}\right)$$

$$+4\left(\frac{3!0!}{(3+0+1)!}\right)$$

$$=\frac{16}{24} \qquad [3.154]$$

$$\int_0^1(-16l_1l_2^2+16l_1^2l_2)dl_2=-16\left(\frac{1!2!}{(1+2+1)!}\right)+16\left(\frac{2!1!}{(2+1+1)!}\right)=0 \qquad [3.155]$$

$$\int_0^1 (-4l_2^3 + 8l_1l_2^2 - 4l_1^2l_2)dl_2 = -4\left(\frac{0!3!}{(0+3+1)!}\right) + 8\left(\frac{1!2!}{(1+2+1)!}\right)$$

$$-4\left(\frac{2!1!}{(2+1+1)!}\right)$$

$$= \frac{-16}{24} \qquad [3.156]$$

$$\int_0^1 (4l_1^2l_2 - 3l_1l_2^2 - l_1^3)dl_2 = 4\left(\frac{2!1!}{(2+1+1)!}\right) - 3\left(\frac{1!2!}{(1+2+1)!}\right)$$

$$-\frac{3!0!}{(3+0+1)!}$$

$$= \frac{-4}{24} \qquad [3.157]$$

$$\int_0^1 (12l_1l_2^2 - 4l_1^2l_2)dl_2 = 12\left(\frac{1!2!}{(1+2+1)!}\right) - 4\left(\frac{2!1!}{(2+1+1)!}\right)$$

$$= \frac{16}{24} \qquad [3.158]$$

$$\int_0^1 (3l_2^3 - 4l_1l_2^2 + l_1^2l_2)dl_2 = 3\left(\frac{0!3!}{(0+3+1)!}\right) - 4\left(\frac{1!2!}{(1+2+1)!}\right)$$

$$+\frac{2!1!}{(2+1+1)!}$$

$$= \frac{12}{24} \qquad [3.159]$$

Substituting [3.151] through [3.159] in [3.150] results in

$$\int_{x_i}^{x_k} [N]^T [B] \{Q\} dx = \frac{1}{24} \begin{bmatrix} -12 & 16 & -4 \\ -16 & 0 & 16 \\ 4 & -16 & 12 \end{bmatrix} \begin{Bmatrix} Q_i \\ Q_j \\ Q_k \end{Bmatrix}$$

$$= [K_e]\{Q\} \qquad [3.160]$$

The third integral of [3.140] is evaluated using a similar procedure as shown above. This results in

$$\int_{x_1}^{x_2}[N]^T I\,dx = I\int_0^L [N]^T ds$$

$$= I\int_0^1 [N]^T L\,dl_2$$

$$= IL\int_0^1 \begin{Bmatrix} l_1^2 - l_1 l_2 \\ 4l_1 l_2 \\ l_2^2 - l_1 l_2 \end{Bmatrix} dl_2 \qquad [3.161]$$

where $s = (Ll_2)$, $\frac{ds}{dl_2} = L$, and $l_1 = (1 - l_2)$.

The integrals of the vector in [3.161] are evaluated separately as follows:

$$\int_0^1 (l_1^2 - l_1 l_2)\,dl_2 = \frac{2!0!}{(2+0+1)!} - \frac{1!1!}{(1+1+1)!}$$

$$= \frac{1}{6} \qquad [3.162]$$

$$\int_0^1 (4l_1 l_2)\,dl_2 = 4\left(\frac{1!1!}{(2+0+1)!}\right)$$

$$= \frac{4}{6} \qquad [3.163]$$

$$\int_0^1 (l_2^2 - l_1 l_2)\,dl_2 = \frac{0!2!}{(0+2+1)!} - \frac{1!1!}{(1+1+1)!}$$

$$= \frac{1}{6} \qquad [3.164]$$

Substituting [3.162], [3.163], and [3.164] in [3.161] produces

$$\int_{x_1}^{x_2}[N]^T I\,dx = \frac{IL}{6}\begin{Bmatrix}1\\4\\1\end{Bmatrix}$$

$$= \frac{-IL}{6}\begin{Bmatrix}-1\\-4\\-1\end{Bmatrix}$$

$$= -I\{F_e\} \qquad [3.165]$$

Substituting [3.149], [3.160], and [3.165] in [3.140] results in

$$\{R_e^{(e)}\} = \frac{L}{30}\begin{bmatrix}4 & 2 & -1\\ 2 & 16 & 2\\ -1 & 2 & 4\end{bmatrix}\begin{Bmatrix}\dot{A}_i\\ \dot{A}_j\\ \dot{A}_k\end{Bmatrix}$$

$$+\frac{1}{24}\begin{bmatrix}-12 & 16 & -4\\ -16 & 0 & 16\\ 4 & -16 & 12\end{bmatrix}\begin{Bmatrix}Q_i\\ Q_j\\ Q_k\end{Bmatrix} - \frac{IL}{6}\begin{Bmatrix}-1\\-4\\-1\end{Bmatrix} \qquad [3.166]$$

or

$$\{R_e^{(e)}\} = [C_e]\{\dot{A}\} + [K_e]\{Q\} - I\{F_e\} \qquad [3.167]$$

b. Formulation of the Unsteady Momentum Equation Using Quadratic Elements

A system of equations is generated by evaluating the following weighted residual integral

$$R(x) = \int_{x_1}^{x_2} W(x)\left[\left(\frac{1}{T} - \frac{Q^2}{gA^3}\right)\frac{\partial A}{\partial x} + \frac{2Q}{gA^2}\frac{\partial Q}{\partial x} + \frac{1}{gA}\frac{\partial Q}{\partial t} - (S_0 - S_f)\right]dx \qquad [3.168]$$

which is the result of integrating the product of the unsteady momentum equation, [3.46], and a weighting function, $W(x)$, over the length of the

quadratic element. As discussed in the previous section, the Galerkin formulation of [3.168] is based on considering the shape functions N_i, N_j, and N_k as the weighting functions at nodes i, j, and k, respectively.

Substituting the shape functions in [3.169] and rewriting the results in a matrix form yields the system of equations

$$\{R_m^{(e)}\} = \begin{Bmatrix} R_i^{(e)} \\ R_j^{(e)} \\ R_k^{(e)} \end{Bmatrix} = \begin{Bmatrix} \int_{x_1}^{x_2} N_i \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} + \frac{1}{gA} \frac{\partial Q}{\partial t} - (S_0 - S_f) \right] dx \\ \int_{x_1}^{x_2} N_j \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} + \frac{1}{gA} \frac{\partial Q}{\partial t} - (S_0 - S_f) \right] dx \\ \int_{x_1}^{x_2} N_k \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} + \frac{1}{gA} \frac{\partial Q}{\partial t} - (S_0 - S_f) \right] dx \end{Bmatrix}$$

$$= \int_{x_1}^{x_2} [N]^T \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} + \frac{1}{gA} \frac{\partial Q}{\partial t} - (S_0 - S_f) \right] dx$$

or

$$\{R_m^{(e)}\} = \int_{x_1}^{x_2} [N]^T \left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} dx + \int_{x_1}^{x_2} [N]^T \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} dx$$

$$+ \int_{x_1}^{x_2} [N]^T \frac{1}{gA} \frac{\partial Q}{\partial t} dx - \int_{x_1}^{x_2} [N]^T (S_0 - S_f) dx \qquad [3.169]$$

The partial derivative $\partial A/\partial x$ is determined by evaluating the derivative of [3.116] with respect to the space dimension. This is accomplished through the evaluation of the partial derivatives of the shape functions with respect to x since the nodal values $A_{i(t)}$, $A_{j(t)}$, and $A_{k(t)}$ are assumed constant with respect to the x – or s –space dimension . This results in

$$\frac{\partial A}{\partial x}=\frac{\partial N_i}{\partial x}A_i+\frac{\partial N_j}{\partial x}A_j+\frac{\partial N_k}{\partial x}A_k$$

$$=\begin{bmatrix}\frac{\partial N_i}{\partial x} & \frac{\partial N_j}{\partial x} & \frac{\partial N_k}{\partial x}\end{bmatrix}\begin{Bmatrix}A_i\\A_j\\A_k\end{Bmatrix}$$

$$=[B]\{A\} \qquad [3.170]$$

The partial derivative $\partial Q/\partial x$ was evaluated earlier as expressed by Equation [3.128].

The partial derivative of $\partial Q/\partial t$ is determined by evaluating the derivative of [3.117] with respect to time. This is accomplished through the evaluation of the partial derivative of flow rate, Q, at nodes i, j, and k, respectively, with respect to time and multiplying the results with the respective shape functions which are assumed constant with respect to time. This results in

$$\frac{\partial Q}{\partial t}=N_i\frac{\partial Q_i}{\partial t}+N_j\frac{\partial Q_j}{\partial t}+N_k\frac{\partial Q_k}{\partial t}$$

$$=[N_i \quad N_j \quad N_k]\begin{Bmatrix}\frac{\partial Q_i}{\partial t}\\\frac{\partial Q_j}{\partial t}\\\frac{\partial Q_k}{\partial t}\end{Bmatrix}$$

$$=[N_i \quad N_j \quad N_k]\begin{Bmatrix}\dot{Q}_i\\\dot{Q}_j\\\dot{Q}_k\end{Bmatrix}$$

$$=[N]\{\dot{Q}\} \qquad [3.171]$$

The various terms of Equation [3.169] are integrated individually after utilizing the results of Equations [3.170], [3.128], and [3.171]. The first term is determined as

$$\int_{x_1}^{x_2}[N]^T\left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)\frac{\partial A}{\partial x}dx=\int_0^L[N]^T\left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)[B]ds\{A\}$$

$$=\left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)\int_0^1[N]^T[B]Ldl_2\{A\} \qquad [3.172]$$

The right side integral of [3.172] is evaluated separately as

$$\int_0^1[N]^T[B]Ldl_2=L\int_0^1\begin{Bmatrix}l_1^2-l_1l_2\\4l_1l_2\\l_2^2-l_1l_2\end{Bmatrix}\left[\frac{1}{L}(3l_2-l_1-2)\quad\frac{1}{L}(-4l_2+4l_1)\quad\frac{1}{L}(3l_2-l_1)\right]dl_2$$

$$=\int_0^1\begin{bmatrix}-l_1^3+4l_1^2l_2-3l_1l_2^2-2l_1^2+2l_1l_2 & -8l_1^2l_2+4l_1l_2^2+4l_1^3 & 4l_1^2l_2-3l_1l_2^2-l_1^3\\12l_1l_2^2-4l_1^2l_2-8l_1l_2 & -16l_1l_2^2+16l_1^2l_2 & 12l_1l_2^2-4l_1^2l_2\\3l_2^3-4l_1l_2^2+l_1^2l_2-2l_2^2+2l_1l_2 & -4l_2^3+8l_1l_2^2-4l_1^2l_2 & 3l_2^3-4l_1l_2^2+l_1^2l_2\end{bmatrix}dl_2$$

[3.173]

where $s=(Ll_2)$, $\frac{ds}{dl_2}=L$, and $l_1=(1-l_2)\cdot$

The integrals of the matrix in [3.173] were evaluated previously. Substituting [3.151] through [3.159] in [3.173] yields

$$\int_0^1[N]^T[B]Ldl_2=\frac{1}{24}\begin{bmatrix}-12 & 16 & -4\\-16 & 0 & 16\\4 & -16 & 12\end{bmatrix} \qquad [3.174]$$

Substituting [3.174] in [3.172] yields

$$\int_{x_1}^{x_2}[N]^T\left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)\frac{\partial A}{\partial x}dx=\left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)\left(\frac{1}{24}\right)\begin{bmatrix}-12 & 16 & -4\\-16 & 0 & 16\\4 & -16 & 12\end{bmatrix}\begin{Bmatrix}A_i\\A_j\\A_k\end{Bmatrix}$$

$$=\left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)[K_c]\{A\} \qquad [3.175]$$

The second integral of [3.169] is evaluated using the same procedure. The result is

$$\int_{x_1}^{x_2}[N]^T\left(\frac{2Q}{gA^2}\right)\frac{\partial Q}{\partial x}dx = \int_0^L [N]^T\left(\frac{2Q}{gA^2}\right)[B]ds\{Q\}$$

$$=\left(\frac{2Q}{gA^2}\right)\int_0^1 [N]^T[B]Ldl_2\{Q\} \qquad [3.176]$$

The right side integral of [3.176] was evaluated previously ([3.174]). Substituting [3.174] in [3.176] results in

$$\int_{x_1}^{x_2}[N]^T\left(\frac{2Q}{gA^2}\right)\frac{\partial Q}{\partial x}dx = \left(\frac{2Q}{gA^2}\right)\left(\frac{1}{24}\right)\begin{bmatrix}-12 & 16 & -4\\ -16 & 0 & 16\\ 4 & -16 & 12\end{bmatrix}\begin{Bmatrix}Q_i\\ Q_j\\ Q_k\end{Bmatrix}$$

$$=\left(\frac{2Q}{gA^2}\right)[K_c]\{Q\} \qquad [3.177]$$

The third integral of [3.169] is evaluated using a similar procedure as discussed in the earlier discussion. The result is

$$\int_{x_1}^{x_2}[N]^T\left(\frac{1}{gA}\right)\frac{\partial Q}{\partial t}dx = \int_0^L [N]^T\left(\frac{1}{gA}\right)[N]ds\{\dot{Q}\}$$

$$=\left(\frac{1}{gA}\right)\int_0^1 [N]^T[N]Ldl_2\{\dot{Q}\} \qquad [3.178]$$

The right side integral of [3.178] was evaluated previously ([3.149]). The results are summarized as

$$\int_0^1 [N]^T[N]Ldl_2 = \frac{L}{30}\begin{bmatrix}4 & 2 & -1\\ 2 & 16 & 2\\ -1 & 2 & 4\end{bmatrix} \qquad [3.179]$$

Substituting [3.179] in [3.178] results in

$$\int_{x_i}^{x_k}[N]^T\left(\frac{1}{gA}\right)\frac{\partial Q}{\partial t}dx=\left(\frac{1}{gA}\right)\left(\frac{L}{30}\right)\begin{bmatrix}4&2&-1\\2&16&2\\-1&2&4\end{bmatrix}\begin{Bmatrix}\dot{Q}_i\\\dot{Q}_j\\\dot{Q}_k\end{Bmatrix}$$

$$=\left(\frac{1}{gA}\right)[C_e]\{\dot{Q}\} \qquad [3.180]$$

The last integral of [3.169] is evaluated using the same procedure as discussed above

$$\int_{x_i}^{x_k}[N]^T(S_0-S_f)dx=(S_0-S_f)\int_0^L[N]^T ds$$

$$=(S_0-S_f)\int_0^1[N]^T L dl_2$$

$$=L(S_0-S_f)\int_0^1\begin{Bmatrix}l_1^2-l_1l_2\\4l_1l_2\\l_2^2-l_1l_2\end{Bmatrix}dl_2 \qquad [3.181]$$

The integrals of the vector in [3.181] were evaluated earlier in [3.162] through [3.164]. Substituting these equations in [3.181] produces

$$\int_{x_i}^{x_k}[N]^T(S_0-S_f)dx=(S_0-S_f)\left(\frac{L}{6}\right)\begin{Bmatrix}1\\4\\1\end{Bmatrix}$$

$$=(S_0-S_f)\{F_e\} \qquad [3.182]$$

Substituting [3.175], [3.177], [3.180], and [3.182] in [3.169] produces the system of equations

$$\{R_m^{(e)}\} = \left(\frac{1}{gA}\right)\left(\frac{L}{30}\right)\begin{bmatrix} 4 & 2 & -1 \\ 2 & 16 & 2 \\ -1 & 2 & 4 \end{bmatrix}\begin{Bmatrix} \dot{Q}_i \\ \dot{Q}_j \\ \dot{Q}_k \end{Bmatrix}$$

$$+\left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)\left(\frac{1}{24}\right)\begin{bmatrix} -12 & 16 & -4 \\ -16 & 0 & 16 \\ 4 & -16 & 12 \end{bmatrix}\begin{Bmatrix} A_i \\ A_j \\ A_k \end{Bmatrix}$$

$$+\left(\frac{2Q}{gA^2}\right)\left(\frac{1}{24}\right)\begin{bmatrix} -12 & 16 & -4 \\ -16 & 0 & 16 \\ 4 & -16 & 12 \end{bmatrix}\begin{Bmatrix} Q_i \\ Q_j \\ Q_k \end{Bmatrix}$$

$$-(S_0 - S_f)\left(\frac{L}{6}\right)\begin{Bmatrix} 1 \\ 4 \\ 1 \end{Bmatrix} \qquad [3.183]$$

or

$$\{R_m^{(e)}\} = \left(\frac{1}{gA}\right)[C_c]\{\dot{Q}\} + \left(\frac{1}{T}-\frac{Q^2}{gA^3}\right)[K_c]\{A\}$$

$$+\left(\frac{2Q}{gA^2}\right)[K_c]\{Q\} - (S_0 - S_f)\{F_c\}$$

$$= [C_m]\{\dot{Q}\} + [K_{mA}]\{A\} + [K_{mQ}]\{Q\} - (S_0 - S_f)\{F_m\} \qquad [3.184]$$

c. Formulation of the Steady Momentum Equation Using Quadratic Elements

The residual vector of the three-nodal quadratic element for the steady momentum equation, [3.53], is developed by following the same procedure of the unsteady momentum equation. The result is

$$\{R_m^{(e)}\} = \begin{Bmatrix} R_i^{(e)} \\ R_j^{(e)} \\ R_k^{(e)} \end{Bmatrix} = \begin{Bmatrix} \int_{x_1}^{x_2} N_i \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} - (S_0 - S_f) \right] dx \\ \int_{x_1}^{x_2} N_j \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} - (S_0 - S_f) \right] dx \\ \int_{x_1}^{x_2} N_k \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} - (S_0 - S_f) \right] dx \end{Bmatrix}$$

$$= \int_{x_1}^{x_2} [N]^T \left[\left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} + \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} - (S_0 - S_f) \right] dx \qquad [3.185]$$

or

$$\{R_m^{(e)}\} = \int_{x_1}^{x_2} [N]^T \left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \frac{\partial A}{\partial x} dx + \int_{x_1}^{x_2} [N]^T \frac{2Q}{gA^2} \frac{\partial Q}{\partial x} dx$$

$$- \int_{x_1}^{x_2} [N]^T (S_0 - S_f) dx \qquad [3.186]$$

Substituting [3.175], [3.177], and [3.182] in [3.186] results in

$$\{R_m^{(e)}\} = \left(\frac{1}{T} - \frac{Q^2}{gA^3} \right) \left(\frac{1}{24} \right) \begin{bmatrix} -12 & 16 & -4 \\ -16 & 0 & 16 \\ 4 & -16 & 12 \end{bmatrix} \begin{Bmatrix} A_i \\ A_j \\ A_k \end{Bmatrix}$$

$$+ \left(\frac{2Q}{gA^2} \right) \left(\frac{1}{24} \right) \begin{bmatrix} -12 & 16 & -4 \\ -16 & 0 & 16 \\ 4 & -16 & 12 \end{bmatrix} \begin{Bmatrix} Q_i \\ Q_j \\ Q_k \end{Bmatrix}$$

$$- (S_0 - S_f) \left(\frac{L}{6} \right) \begin{Bmatrix} 1 \\ 4 \\ 1 \end{Bmatrix} \qquad [3.187]$$

or

$$\{R_m^{(e)}\} = \left(\frac{1}{T} - \frac{Q^2}{gA^3}\right)[K_c]\{A\} + \left(\frac{2Q}{gA^2}\right)[K_c]\{Q\} - (S_0 - S_f)\{F_c\}$$

$$= [K_{mA}]\{A\} + [K_{mQ}]\{Q\} - \{F_m\} \qquad [3.188]$$

d. Formulation of the Zero-Inertia Equation Using Quadratic Elements

The residual vector of the quadratic element for the zero-inertia momentum equation, [3.177], is developed based on the same procedure that was followed in the previous section. The result is

$$\{R_m^{(e)}\} = \begin{Bmatrix} R_i^{(e)} \\ R_j^{(e)} \\ R_k^{(e)} \end{Bmatrix} = \begin{Bmatrix} \int_{x_1}^{x_2} N_i \left[\frac{1}{T}\frac{\partial A}{\partial x} - (S_0 - S_f)\right] dx \\ \int_{x_1}^{x_2} N_j \left[\frac{1}{T}\frac{\partial A}{\partial x} - (S_0 - S_f)\right] dx \\ \int_{x_1}^{x_2} N_k \left[\frac{1}{T}\frac{\partial A}{\partial x} - (S_0 - S_f)\right] dx \end{Bmatrix}$$

$$= \int_{x_1}^{x_2} [N]^T \left[\frac{1}{T}\frac{\partial A}{\partial x} - (S_0 - S_f)\right] dx$$

or

$$\{R_m^{(e)}\} = \int_{x_1}^{x_2} [N]^T \left(\frac{1}{T}\right)\frac{\partial A}{\partial x} dx - \int_{x_1}^{x_2} [N]^T (S_0 - S_f) dx \qquad [3.189]$$

Substituting [3.175] and [3.182] in [3.189] results in the system of equations

$$\{R_m^{(e)}\} = \left(\frac{1}{T}\right)\left(\frac{1}{24}\right)\begin{bmatrix} -12 & 16 & -4 \\ -16 & 0 & 16 \\ 4 & -16 & 12 \end{bmatrix}\begin{Bmatrix} A_i \\ A_j \\ A_k \end{Bmatrix} - (S_0 - S_f)\left(\frac{L}{6}\right)\begin{Bmatrix} 1 \\ 4 \\ 1 \end{Bmatrix} \qquad [3.190]$$

where $(1/T)$ is the new coefficient in [3.98] instead of $(1/T - Q^2/gA^3)$. The friction slope, S_f, is expressed as a function of flow rate, Q, and cross sectional area of flow, A, using any of the uniform flow equations that were reviewed in Chapter II. If the Manning equation is selected, [3.190] becomes

$$\{R_m^{(e)}\} = \left(\frac{1}{T}\right)\left(\frac{1}{24}\right)\begin{bmatrix} -12 & 16 & -4 \\ -16 & 0 & 16 \\ 4 & -16 & 12 \end{bmatrix}\begin{Bmatrix} A_i \\ A_j \\ A_k \end{Bmatrix} - \frac{L}{6}\begin{Bmatrix} (S_0 - S_f)_i \\ 4(S_0 - S_f)_j \\ (S_0 - S_f)_k \end{Bmatrix}$$

$$= \left(\frac{1}{T}\right)\left(\frac{1}{24}\right)\begin{bmatrix} -12 & 16 & -4 \\ -16 & 0 & 16 \\ 4 & -16 & 12 \end{bmatrix}\begin{Bmatrix} A_i \\ A_j \\ A_k \end{Bmatrix} + \frac{L}{6}\begin{Bmatrix} S_{fi} \\ 4S_{fj} \\ S_{fk} \end{Bmatrix} - \frac{L}{6}\begin{Bmatrix} S_{0i} \\ 4S_{0j} \\ S_{0k} \end{Bmatrix}$$

$$= \left(\frac{1}{T}\right)\left(\frac{1}{24}\right)\begin{bmatrix} -12 & 16 & -4 \\ -16 & 0 & 16 \\ 4 & -16 & 12 \end{bmatrix}\begin{Bmatrix} A_i \\ A_j \\ A_k \end{Bmatrix}$$

$$+ \frac{L}{6}\begin{Bmatrix} \left(\frac{nQ_i}{A_i R_i^{2/3}}\right)^2 \\ 4\left(\frac{nQ_j}{A_j R_j^{2/3}}\right)^2 \\ \left(\frac{nQ_k}{A_k R_k^{2/3}}\right)^2 \end{Bmatrix} - \frac{L}{6}\begin{Bmatrix} S_{0i} \\ 4S_{0j} \\ S_{0k} \end{Bmatrix}$$

The system of equations above can be rewritten in a similar form as that of the steady momentum formulation ([3.188]). This results in

$$\{R_m^{(e)}\} = \left(\frac{1}{T}\right)\left(\frac{1}{24}\right)\begin{bmatrix} -12 & 16 & -4 \\ -16 & 0 & 16 \\ 4 & -16 & 12 \end{bmatrix}\begin{Bmatrix} A_i \\ A_j \\ A_k \end{Bmatrix}$$

$$+\frac{L}{6}\begin{bmatrix} \frac{n^2 Q_i}{A_i^2 R_i^{4/3}} & 0 & 0 \\ 0 & \frac{n^2 Q_j}{A_j^2 R_j^{4/3}} & 0 \\ 0 & 0 & \frac{n^2 Q_k}{A_k^2 R_k^{4/3}} \end{bmatrix}\begin{Bmatrix} Q_i \\ Q_j \\ Q_k \end{Bmatrix}$$

$$-\frac{L}{6}\begin{Bmatrix} S_{0i} \\ 4S_{0j} \\ S_{0k} \end{Bmatrix} \qquad [3.191]$$

or

$$\{R_m^{(e)}\} = \frac{1}{T}[K_c]\{A\} + [K_{sQ}]\{Q\} - S_0\{F_c\}$$

$$= [K_{sA}]\{A\} + [K_{sQ}]\{Q\} - \{F_s\} \qquad [3.192]$$

e. Formulation of the Kinematic Wave Equation Using Quadratic Elements

As was discussed previously, various assumptions are implemented to simplify [3.46] to the so-called kinematic wave approximation. Under these assumptions, the momentum equation reduces to

$$S_0 - S_f = 0 \qquad [3.193]$$

Equation [3.193] is utilized in the Manning equation ([2.12]) to calculate the flow rate, Q, at individual nodes. This results in

$$Q_i = \frac{1}{n} A_i R_i^{2/3} S_{0i}^{1/2}$$

$$Q_j = \frac{1}{n} A_j R_j^{2/3} S_{0j}^{1/2}$$

and

$$Q_k = \frac{1}{n} A_k R_k^{2/3} S_{0k}^{1/2}$$

where R_i, R_j, and R_k are the hydraulic radii at nodes i, j, and k, respectively. The above equations can be rewritten as

$$\left(\frac{1}{n} R_i^{2/3} S_{0i}^{1/2}\right) A_i - Q_i = 0 \qquad [3.194]$$

$$\left(\frac{1}{n} R_j^{2/3} S_{0j}^{1/2}\right) A_j - Q_j = 0 \qquad [3.195]$$

and

$$\left(\frac{1}{n} R_k^{2/3} S_{0k}^{1/2}\right) A_k - Q_k = 0 \qquad [3.196]$$

Since we are primarily interested in the contribution of nodes i, j, and k to the individual element, only half of the flow terms in Equations [3.194] through [3.196] will be utilized as the element contribution to the final system of equations. Thus, Equations [3.194], [3.195], and [3.196] can be rewritten in the matrix form

$$\frac{1}{2n}\begin{bmatrix} R_i^{2/3} S_{0i}^{1/2} & 0 & 0 \\ 0 & 2R_j^{2/3} S_{0j}^{1/2} & 0 \\ 0 & 0 & R_k^{2/3} S_{0k}^{1/2} \end{bmatrix} \begin{Bmatrix} A_i \\ A_j \\ A_k \end{Bmatrix} + \begin{bmatrix} -\frac{1}{2} & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -\frac{1}{2} \end{bmatrix} \begin{Bmatrix} Q_i \\ Q_j \\ Q_k \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix} \qquad [3.197]$$

or

$$[K_{kA}]\{A\} + [K_{kQ}]\{Q\} - \{F_k\} = \{0\}$$

where $\{F_k\} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}$.

f. General Finite Element Formulation Using Quadratic Elements

The one-dimensional quadratic element formulation of the continuity equation, [3.167], is rewritten as

$$\{R_e^{(e)}\} = \frac{L}{30}\begin{bmatrix} 4 & 2 & -1 \\ 2 & 16 & 2 \\ -1 & 2 & 4 \end{bmatrix}\begin{Bmatrix} \dot{A}_i \\ \dot{A}_j \\ \dot{A}_k \end{Bmatrix} + \begin{bmatrix} -\frac{12}{24} & \frac{16}{24} & -\frac{4}{24} \\ -\frac{16}{24} & 0 & \frac{16}{24} \\ \frac{4}{24} & -\frac{16}{24} & \frac{12}{24} \end{bmatrix}\begin{Bmatrix} Q_i \\ Q_j \\ Q_k \end{Bmatrix} - \begin{Bmatrix} \frac{-I_i L}{6} \\ \frac{-4I_j L}{6} \\ \frac{-I_k L}{6} \end{Bmatrix} \qquad [3.198]$$

On the other hand, Equations [3.183], [3.187], [3.191], and [3.197] are expressed in the general form

$$\{R_m^{(e)}\} = \frac{L}{30}\begin{bmatrix} 4c_1 & 2c_1 & -c_1 \\ 2c_1 & 16c_1 & 2c_1 \\ -c_1 & 2c_1 & 4c_1 \end{bmatrix}\begin{Bmatrix} \dot{Q}_i \\ \dot{Q}_j \\ \dot{Q}_k \end{Bmatrix}$$

$$+\begin{bmatrix} \frac{-(c_2-c_3)+c_i}{2} & \frac{16(c_2-c_3)}{24} & \frac{-4(c_2-c_3)}{24} \\ \frac{-16(c_2-c_3)}{24} & c_j & \frac{16(c_2-c_3)}{24} \\ \frac{4(c_2-c_3)}{24} & \frac{-16(c_2-c_3)}{24} & \frac{(c_2-c_3)+c_k}{2} \end{bmatrix}\begin{Bmatrix} A_i \\ A_j \\ A_k \end{Bmatrix}$$

$$+\begin{bmatrix} \frac{-c_4+c_{0i}}{2} & \frac{16c_4}{24} & \frac{-4c_4}{24} \\ \frac{-16c_4}{24} & c_{0j} & \frac{16c_4}{24} \\ \frac{4c_4}{24} & \frac{-16c_4}{24} & \frac{c_4+c_{0k}}{2} \end{bmatrix}\begin{Bmatrix} Q_i \\ Q_j \\ Q_k \end{Bmatrix} - \begin{Bmatrix} f_i \\ 4f_j \\ f_k \end{Bmatrix} \qquad [3.199]$$

where c_{0i}, c_{0j}, c_{0k}, c_1, c_2, c_3, c_4, c_i, c_j, c_k, f_i, f_j, and f_k are coefficients that vary based on the selected model.

There are four different models that will result based on the previous discussion. These include

i. Hydrodynamic Model I : This model is the result of combining both the continuity equation and the unsteady momentum equation. The various coefficients in [3.199] may be expressed as

$$c_{0i} = 0, \quad c_{0j} = 0, \quad c_{0k} = 0, \quad c_i = 0, \quad c_j = 0, \quad c_k = 0,$$

$$c_1 = \left(\frac{1}{gA}\right), \quad c_2 = \left(\frac{1}{T}\right), \quad c_3 = \left(\frac{Q^2}{gA^3}\right), \quad c_4 = \left(\frac{2Q}{gA^2}\right),$$

$$f_i = \frac{L}{6}(S_0 - S_f)_i, \quad f_j = \frac{L}{6}(S_0 - S_f)_j, \quad \text{and} \quad f_k = \frac{L}{6}(S_0 - S_f)_k.$$

ii. Hydrodynamic Model II : This model is the result of combining both the continuity equation and the steady momentum equation. The various coefficients in [3.199] may be expressed as

$$c_{0i}=0, \quad c_{0j}=0, \quad c_{0k}=0, \quad c_i=0, \quad c_j=0, \quad c_k=0,$$

$$c_1=0, \quad c_2=\left(\frac{1}{T}\right), \quad c_3=\left(\frac{Q^2}{gA^3}\right), \quad c_4=\left(\frac{2Q}{gA^2}\right),$$

$$f_i=\frac{L}{6}(S_0-S_f)_i, \quad f_j=\frac{L}{6}(S_0-S_f)_j, \quad \text{and} \quad f_k=\frac{L}{6}(S_0-S_f)_k.$$

iii. Zero-Inertia Model : This model is the result of combining both the continuity equation and the zero-inertia assumptions that produced [3.98]. The various coefficients in [3.199] may be expressed as

$$c_{0i}=\frac{Ln^2Q_i}{3A_i^2R_i^{4/3}}, \quad c_{0j}=\frac{4Ln^2Q_j}{6A_j^2R_j^{4/3}}, \quad c_{0k}=\frac{Ln^2Q_k}{3A_k^2R_k^{4/3}}, \quad c_i=0,$$

$$c_j=0, \quad c_k=0, \quad c_1=0, \quad c_2=\left(\frac{1}{T}\right), \quad c_3=0, \quad c_4=0,$$

$$f_i=\frac{L}{6}S_{0i}, \quad f_j=\frac{L}{6}S_{0j}, \quad \text{and} \quad f_k=\frac{L}{6}S_{0k}.$$

iv. Kinematic Wave Model : This model is the result of combining both the continuity equation and the kinematic wave or the uniform flow assumptions that produced [3.193]. The various coefficients in [3.199] may be expressed as

$$c_{0i}=-1, \quad c_{0j}=-1, \quad c_{0k}=-1, \quad f_i=0, \quad f_j=0, \quad f_k=0,$$

$$c_1=0, \quad c_2=0, \quad c_3=0, \quad c_4=0,$$

$$c_i=\frac{1}{n}R_i^{2/3}S_{0i}^{1/2}, \quad c_j=\frac{1}{n}R_j^{2/3}S_{0j}^{1/2}, \quad \text{and} \quad c_k=\frac{1}{n}R_k^{2/3}S_{0k}^{1/2}.$$

Equations [3.198] and [3.199] should be solved simultaneously for every element to determine both the flow rate, Q, and the cross sectional area of flow, A, at each node. However, these systems of equations are combined together to produce one system of equations for every element. The resultant system of equations will have the form

$$\{R^{(e)}\}=\begin{Bmatrix} R_{ci}^{(e)} \\ R_{mi}^{(e)} \\ R_{cj}^{(e)} \\ R_{mj}^{(e)} \\ R_{ck}^{(e)} \\ R_{mk}^{(e)} \end{Bmatrix}=\frac{L}{30}\begin{bmatrix} 4 & 0 & 2 & 0 & -1 & 0 \\ 0 & 4c_1 & 0 & 2c_1 & 0 & -c_1 \\ 2 & 0 & 16 & 0 & 2 & 0 \\ 0 & 2c_1 & 0 & 16c_1 & 0 & 2c_1 \\ -1 & 0 & 2 & 0 & 4 & 0 \\ 0 & -c_1 & 0 & 2c_1 & 0 & 4c_1 \end{bmatrix}\begin{Bmatrix} \dot{A}_i \\ \dot{Q}_i \\ \dot{A}_j \\ \dot{Q}_j \\ \dot{A}_k \\ \dot{Q}_k \end{Bmatrix}$$

$$+\begin{bmatrix} 0 & -\frac{12}{24} & 0 & \frac{16}{24} & 0 & -\frac{4}{24} \\ \frac{-c_5+c_i}{2} & \frac{-c_4+c_{0i}}{2} & \frac{16c_5}{24} & \frac{16c_4}{24} & \frac{-4c_5}{24} & \frac{-4c_4}{24} \\ 0 & -\frac{16}{24} & 0 & 0 & 0 & \frac{16}{24} \\ \frac{-16c_5}{24} & \frac{-16c_4}{24} & c_j & c_{0j} & \frac{16c_5}{24} & \frac{16c_4}{24} \\ 0 & \frac{4}{24} & 0 & -\frac{16}{24} & 0 & \frac{12}{24} \\ \frac{4c_5}{24} & \frac{4c_4}{24} & \frac{-16c_5}{24} & \frac{-16c_4}{24} & \frac{c_5+c_k}{2} & \frac{c_4+c_{0k}}{2} \end{bmatrix}\begin{Bmatrix} A_i \\ Q_i \\ A_j \\ Q_j \\ A_k \\ Q_k \end{Bmatrix}-\begin{Bmatrix} \frac{-I_iL}{6} \\ f_i \\ \frac{-4I_jL}{6} \\ 4f_j \\ \frac{-I_kL}{6} \\ f_k \end{Bmatrix} \qquad [3.200]$$

or

$$\{R^{(e)}\}=[C]\{\dot{\Phi}\}+[K]\{\Phi\}-\{F\} \qquad [3.201]$$

where $c_5=c_2-c_3$.

4. Finite Element Development Using Nonstandard Galerkin Formulation

An oscillation free solution can be generated using the nonstandard Galerkin formulation of the linear finite element problem. The formulation utilizes asymmetric weighting functions for approximating the space derivatives. These asymmetric weighting functions have the form (Allen et al., 1988; Lapidus and Pinder, 1982)

$$N_i = \frac{X_j - x}{L} + 3\alpha \frac{(x - X_i)(x - X_j)}{L^2} \qquad [3.202]$$

$$N_j = \frac{x - X_i}{L} - 3\alpha \frac{(x - X_i)(x - X_j)}{L^2} \qquad [3.203]$$

where L is the length of the element (refer to Figure 3). The shape functions are then substituted as the weighting functions in Equations [3.72], [3.80], [3.90], and [3.99] for the continuity, unsteady momentum, steady momumtum, and simplified momentum (with zero-inertia assumption) equations, respectively. Then the entire integration process is repeated. After completing this step, the resultant element stiffness matrix, $[K_e]$, will have the form

$$\int_{x_i}^{x_j} [N]^T \frac{\partial \Phi}{\partial x} dx = \int_0^L [N]^T [B] \{\Phi\} ds = \frac{1}{T} \int_0^L \begin{Bmatrix} N_i \\ N_j \end{Bmatrix} \begin{bmatrix} \frac{\partial N_i}{\partial x} & \frac{\partial N_j}{\partial x} \end{bmatrix} ds \{\Phi\}$$

$$= \begin{bmatrix} -\frac{1}{2} + \frac{\alpha}{2} & \frac{1}{2} - \frac{\alpha}{2} \\ -\frac{1}{2} - \frac{\alpha}{2} & \frac{1}{2} + \frac{\alpha}{2} \end{bmatrix} \{\Phi\} = [K_e] \{\Phi\} \qquad [3.204]$$

where Φ represents either A or Q.

On the other hand, the time derivatives in the above equations may be approximated using yet another nonstandard Galerkin procedure. This procedure involves the lumping of the coefficients of the time derivative. In order to accomplish this step, the variation of $\partial A/\partial t$ and $\partial Q/\partial t$ with respect to x are assumed constant within the midpoints of adjacent linear elements (Segerlind, 1984). This step is discussed in the following section for both linear and quadratic elements.

5. Finite Element Development Using Lumped Formulation

The finite element development of the variation of the $\partial A/\partial t$ and $\partial Q/\partial t$ terms of the Saint-Venant equations with respect to the space dimension was performed in the previous two sections based on the so-called consistent or standard finite element Galerkin formulation. This formulation was utilized in establishing the weighting functions together with approximating the variation of time derivatives of area of flow and flow rate with respect to x for both the continuity, [3.9], and momentum, [3.46], equations. This development was based on the assumption that the variations in the time derivative are linear and quadratic for the linear and quadratic finite elements, respectively (Segerlind, 1984). The linear and quadratic shape functions were then used as a weighting coefficients for the linear and the quadratic element formulations, respectively. The resultant element capacitance matrix for the linear element in both the continuity and momentum equations had the form

$$[C_e]=\frac{L}{6}\begin{bmatrix}2 & 1\\ 1 & 2\end{bmatrix} \qquad [3.205]$$

On the other hand, applying the consistent finite element formulation to the quadratic element resulted in the capacitance matrix

$$[C_e]=\frac{L}{30}\begin{bmatrix} 4 & 2 & -1 \\ 2 & 16 & 2 \\ -1 & 2 & 4 \end{bmatrix} \qquad [3.206]$$

An alternative to the above approach for defining the variation of $\partial A/\partial t$ and $\partial Q/\partial t$ with respect to x is to assume that these are constant within the midpoints of adjacent linear elements (Segerlind, 1984). This concept is referred to as the lumped or nonstandard finite element Galerkin formulation. Under this assumption, the variation of both $\partial A/\partial t$ and $\partial Q/\partial t$ with respect to x are written explicitly using the step function. This concept is applied to both linear and quadratic element formulations.

a. Linear Element

As discussed by Segerlind (1984), the variation of both $\partial A/\partial t$ and $\partial Q/\partial t$ within the element can be written using the step functions

$$\frac{\partial A}{\partial t}(x)=\left[1-h\left(s-\frac{L}{2}\right)\right]\frac{\partial A_i}{\partial t}+\left[h\left(s-\frac{L}{2}\right)\right]\frac{\partial A_j}{\partial t} \qquad [3.207]$$

and

$$\frac{\partial Q}{\partial t}(x)=\left[1-h\left(s-\frac{L}{2}\right)\right]\frac{\partial Q_i}{\partial t}+\left[h\left(s-\frac{L}{2}\right)\right]\frac{\partial Q_j}{\partial t} \qquad [3.208]$$

$$\text{where} \quad h\left(s-\frac{L}{2}\right)=\begin{Bmatrix} 0 & \text{for} & s<\frac{L}{2} \\ 1 & \text{for} & s>\frac{L}{2} \end{Bmatrix} \qquad [3.209]$$

The selection of the step function in [3.209] is based on the assumption that the variations of $\partial A/\partial t$ and $\partial Q/\partial t$ with respect to x are constant within the midpoints of adjacent elements.

The quantities multiplying A_i, A_j, Q_i, and Q_j are the new shape functions, N_i^* and N_j^*, where

$$N_i^* = 1 - h\left(s - \frac{L}{2}\right) \qquad [3.210]$$

$$N_j^* = h\left(s - \frac{L}{2}\right) \qquad [3.211]$$

Substituting [3.210] and [3.211] in [3.207] and [3.208] results in

$$\frac{\partial A}{\partial t}(x) = N_i^* \frac{\partial A_i}{\partial t} + N_j^* \frac{\partial A_j}{\partial t}$$

$$= [N^*]\{\dot{A}\} \qquad [3.212]$$

and

$$\frac{\partial Q}{\partial t}(x) = N_i^* \frac{\partial Q_i}{\partial t} + N_j^* \frac{\partial Q_j}{\partial t}$$

$$= [N^*]\{\dot{Q}\} \qquad [3.213]$$

Substituting the same shape functions (Equations [3.210] and [3.211]) as the weighting functions in the first right side term of [3.73] and the third right side term of [3.81] results in

$$\int_{x_i}^{x_j} [N^*]^T [N^*]\{\dot{A}\}dx = \int_0^L [N^*]^T [N^*]ds\{\dot{A}\}$$

$$= [C_e]\{\dot{A}\} \qquad [3.214]$$

and

$$\int_{x_i}^{x_j} [N^*]^T [N^*]\{\dot{Q}\}dx = \int_0^L [N^*]^T [N^*]ds\{\dot{Q}\}$$

$$= [C_e]\{\dot{Q}\} \qquad [3.215]$$

$$\text{where} \quad [C_e] = \int_0^L [N^*]^T [N^*] ds \qquad [3.216]$$

The element capacitance matrix, $[C_e]$, can be readily evaluated since $N_i^* N_i^* = 1$, $N_i^* N_j^* = 0$, and $N_j^* N_j^* = 1$. These terms are the result of the characteristics of the shape functions. Each shape function has a value of one at its own node and zero at the other nodes (Segerlind, 1984). Substituting these terms in [3.216] results in

$$\begin{aligned}
\int_0^L [N^*]^T [N^*] ds &= \int_0^L \begin{bmatrix} N_i^* N_i^* & N_i^* N_j^* \\ N_i^* N_j^* & N_j^* N_j^* \end{bmatrix} ds \\
&= \int_0^{\frac{L}{2}} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} ds + \int_{\frac{L}{2}}^L \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} ds \\
&= \begin{bmatrix} \frac{L}{2} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \frac{L}{2} \end{bmatrix} \\
&= \frac{L}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}
\end{aligned} \qquad [3.217]$$

The resultant new capacitance matrix in [3.217] together with the new stiffness matrix in [3.204] are then utilized in establishing a general linear element system of equations for the nonstandard finite element Galerkin formulation. This is accomplished by following the same approach that resulted in [3.111] which represents the general linear element system of equations developed based on the consistent finite element formulation. The resultant general system of equations will have the form

$$\{R^{(e)}\}=\begin{Bmatrix}R_{ci}^{(e)}\\R_{mi}^{(e)}\\R_{cj}^{(e)}\\R_{mj}^{(e)}\end{Bmatrix}=\frac{L}{2}\begin{bmatrix}1&0&0&0\\0&c_1&0&0\\0&0&1&0\\0&0&0&c_1\end{bmatrix}\begin{Bmatrix}\dot{A}_i\\\dot{Q}_i\\\dot{A}_j\\\dot{Q}_j\end{Bmatrix}$$

$$+\frac{1}{2}\begin{bmatrix}0&-1+\alpha&0&1-\alpha\\(-1+\alpha)c_5+c_i&(-1+\alpha)c_4+c_0&(1-\alpha)c_5&(1-\alpha)c_4\\0&-1-\alpha&0&1-\alpha\\(-1-\alpha)c_5&(-1-\alpha)c_4&(1+\alpha)c_5+c_j&(1+\alpha)c_4+c_0\end{bmatrix}\begin{Bmatrix}A_i\\Q_i\\A_j\\Q_j\end{Bmatrix}$$

$$-\begin{Bmatrix}\frac{-I_iL}{2}\\f_i\\\frac{-I_jL}{2}\\f_j\end{Bmatrix} \qquad [3.218]$$

or

$$\{R^{(e)}\}=[C]\{\dot{\Phi}\}+[K]\{\Phi\}-\{F\} \qquad [3.219]$$

where c_0, c_1, c_2, c_3, c_4, c_i, c_j, f_i, and f_j are as defined previously.

b. Quadratic Element

In this section, the same approach is followed as shown above. The resulting variations of both $\partial A/\partial t$ and $\partial Q/\partial t$ within a quadratic element are written using the step functions

$$\frac{\partial A}{\partial t}(x)=\left[1-h\left(s-\frac{L}{6}\right)\right]\frac{\partial A_i}{\partial t}+\left[h\left(s-\frac{L}{6}\right)\right]$$

$$\cdot\left[1-h\left(s-\frac{5L}{6}\right)\right]\frac{\partial A_j}{\partial t}+\left[h\left(s-\frac{5L}{6}\right)\right]\frac{\partial A_k}{\partial t} \qquad [3.220]$$

and

$$\frac{\partial Q}{\partial t}(x)=\left[1-h\left(s-\frac{L}{6}\right)\right]\frac{\partial Q_i}{\partial t}+\left[h\left(s-\frac{L}{6}\right)\right]$$
$$\cdot\left[1-h\left(s-\frac{5L}{6}\right)\right]\frac{\partial Q_j}{\partial t}+\left[h\left(s-\frac{5L}{6}\right)\right]\frac{\partial Q_k}{\partial t} \qquad [3.221]$$

where $$h\left(s-\frac{L}{6}\right)=\begin{cases}0 & \text{for } s<\frac{L}{6}\\ 1 & \text{for } s>\frac{L}{6}\end{cases} \qquad [3.222]$$

and $$h\left(s-\frac{5L}{6}\right)=\begin{cases}0 & \text{for } s<\frac{5L}{6}\\ 1 & \text{for } s>\frac{5L}{6}\end{cases} \qquad [3.223]$$

It is important at this point to emphasize the fact that the selection of the step functions in [3.222] and [3.223] is based on the assumption that the variations of $\partial A/\partial t$ and $\partial Q/\partial t$ with respect to x are limited to three constant values within the length of the quadratic element. These values are constant in the intervals 0 to $L/6$, $L/6$ to $5L/6$, and $5L/6$ to L.

The quantities multiplying A_i, A_j, A_k, Q_i, Q_j, and Q_k are the new shape functions (N_i^*, N_j^*, and N_k^*) where

$$N_i^*=1-h\left(s-\frac{L}{6}\right) \qquad [3.224]$$

$$N_j^*=\left[h\left(s-\frac{L}{6}\right)\right]\left[1-h\left(s-\frac{5L}{6}\right)\right] \qquad [3.225]$$

$$N_k^*=h\left(s-\frac{5L}{6}\right) \qquad [3.226]$$

Substituting [3.224], [3.225], and [3.226] in Equations [3.220] and [3.221] results in

$$\frac{\partial A}{\partial t}(x) = N_i^* \frac{\partial A_i}{\partial t} + N_j^* \frac{\partial A_j}{\partial t} + N_k^* \frac{\partial A_k}{\partial t}$$

$$= [N^*]\{\dot{A}\} \qquad [3.227]$$

and

$$\frac{\partial Q}{\partial t}(x) = N_i^* \frac{\partial Q_i}{\partial t} + N_j^* \frac{\partial Q_j}{\partial t} + N_k^* \frac{\partial Q_k}{\partial t}$$

$$= [N^*]\{\dot{Q}\} \qquad [3.228]$$

After implementing the same shape functions (Equations [3.224], [3.225], and [3.226]) as the weighting functions in the first right side term of Equation [3.140] and the third right side term of Equation [3.169], the following equations result

$$\int_{x_1}^{x_2} [N^*]^T [N^*]\{\dot{A}\} dx = \int_0^L [N^*]^T [N^*] ds \{\dot{A}\}$$

$$= [C_c]\{\dot{A}\} \qquad [3.229]$$

and

$$\int_{x_1}^{x_2} [N^*]^T [N^*]\{\dot{Q}\} dx = \int_0^L [N^*]^T [N^*] ds \{\dot{Q}\}$$

$$= [C_c]\{\dot{Q}\} \qquad [3.230]$$

where $$[C_c] = \int_0^L [N^*]^T [N^*] ds \qquad [3.231]$$

The element capacitance matrix, $[C_c]$, can be readily evaluated since $N_i^* N_i^* = 1$, $N_i^* N_j^* = 0$, $N_i^* N_k^* = 0$, $N_j^* N_j^* = 1$, $N_j^* N_k^* = 0$, and $N_k^* N_k^* = 1$. Substituting these terms in [3.231] results in

$$\int_0^L [N^*]^T [N^*] ds = \int_0^L \begin{bmatrix} N_i^* N_i^* & N_i^* N_j^* & N_i^* N_k^* \\ N_j^* N_i^* & N_j^* N_j^* & N_j^* N_k^* \\ N_k^* N_i^* & N_k^* N_j^* & N_k^* N_k^* \end{bmatrix} ds$$

$$= \int_0^{\frac{L}{6}} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} ds + \int_{\frac{L}{6}}^{\frac{5L}{6}} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} ds + \int_{\frac{5L}{6}}^{L} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} ds$$

$$= \begin{bmatrix} \frac{L}{6} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{4L}{6} & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{L}{6} \end{bmatrix}$$

$$= \frac{L}{6} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad [3.232]$$

The resultant capacitance matrix in [3.232] is utilized in establishing a new general system of equations for the quadratic element. This is accomplished by following the same approach that resulted in [3.200] which represents the general system of equations based on the consistent finite element formulation. The resultant system of equations has the form

$$\{R^{(e)}\}=\begin{Bmatrix}R_{ci}^{(e)}\\R_{mi}^{(e)}\\R_{cj}^{(e)}\\R_{mj}^{(e)}\\R_{ck}^{(e)}\\R_{mk}^{(e)}\end{Bmatrix}=\frac{L}{6}\begin{bmatrix}1&0&0&0&0&0\\0&c_1&0&0&0&0\\0&0&4&0&0&0\\0&0&0&4c_1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&c_1\end{bmatrix}\begin{Bmatrix}\dot{A}_i\\\dot{Q}_i\\\dot{A}_j\\\dot{Q}_j\\\dot{A}_k\\\dot{Q}_k\end{Bmatrix}$$

$$+\begin{bmatrix}0&-\frac{12}{24}&0&\frac{16}{24}&0&-\frac{4}{24}\\\frac{-c_5+c_i}{2}&\frac{-c_4+c_{0i}}{2}&\frac{16c_5}{24}&\frac{16c_4}{24}&\frac{-4c_5}{24}&\frac{-4c_4}{24}\\0&-\frac{16}{24}&0&0&0&\frac{16}{24}\\\frac{-16c_5}{24}&\frac{-16c_4}{24}&c_j&c_{0j}&\frac{16c_5}{24}&\frac{16c_4}{24}\\0&\frac{4}{24}&0&-\frac{16}{24}&0&\frac{12}{24}\\\frac{4c_5}{24}&\frac{4c_4}{24}&\frac{-16c_5}{24}&\frac{-16c_4}{24}&\frac{c_5+c_k}{2}&\frac{c_4+c_{0k}}{2}\end{bmatrix}\begin{Bmatrix}A_i\\Q_i\\A_j\\Q_j\\A_k\\Q_k\end{Bmatrix}-\begin{Bmatrix}\frac{-I_iL}{6}\\f_i\\\frac{-4I_jL}{6}\\4f_j\\\frac{-I_kL}{6}\\f_k\end{Bmatrix} \qquad [3.233]$$

or

$$\{R^{(e)}\}=[C]\{\dot{\Phi}\}+[K]\{\Phi\}-\{F\} \qquad [3.234]$$

where c_{0i}, c_{0j}, c_{0k}, c_1, c_2, c_3, c_4, c_i, c_j, c_k, f_i, and f_j are as defined previously and $c_5=c_2-c_3$.

The lumping of the coefficients of the time derivative usually produces smoother numerical solutions compared to the standard finite element Galerkin formulation (Allen et al., 1988). The lumped finite element formulation has fewer constraints on the time step compared to the consistent finite element formulation. The consistent formulation will violate physical reality and produce numerical oscillations if the selected value of the θ

parameter is less than $2/3$ in the two-level time scheme (Segerlind, 1984). The latter scheme will result from implementing the finite difference solution in time as will be demonstrated in the next two sections.

6. Direct Stiffness Method

Segerlind (1984) defined the direct stiffness method as "the procedure for incorporating the element matrices into the final system of equations". Equations [3.219] and [3.233] represent the general finite element system of equations for the linear and quadratic elements, respectively. These element matrices can be incorporated into the final system of equations in a straightforward manner. The element matrices will be assembled into a banded system of equations as discussed in the text by Segerlind (1984). The zero coefficients outside the bandwidth will not to be stored. The bandwidth is defined as one plus the greatest distance between the last non-zero coefficient and the diagonal coefficient in a row (Segerlind, 1984). Since both the linear and quadratic elements of the finite element grid of the surface irrigation problems will be numbered successively from left to right as shown in Figures 3 and 4, the bandwidths of these problems will be four and six, respectively. The reason the bandwidths are of values of twice as much as the classic linear and quadratic one-dimensional grids is due to the number of unknowns per node. As discussed earlier, there are two unknowns per node for the surface irrigation problems which include the area of flow, A, and the flow rate, Q, compared to only one unknown per node for the classic one-dimensional finite element grids.

By applying the direct stiffness procedure, the following global system of equations result

$$\{R_G\} = [C_G]\{\Phi\} + [K_G]\{\Phi\} - \{F_G\} \qquad [3.235]$$

where $\{R_G\}$ is the global residual vector, $\{\Phi\}$ is the global vector of unknowns A and Q, $\{\dot{\Phi}\}$ is the global vector of the unknown derivatives with respect to time, $[C_G]$ is the global capacitance matrix, $[K_G]$ is the global stiffness matrix, and $\{F_G\}$ is the global force vector.

The system of equations in [3.235] represents a system of nonlinear first-order differential equations in the time domain. The boundary conditions should be incorporated into this system before it can be worked out further as will be shown in the next section.

7. Finite Difference Solution in Time

After applying the direct stiffness procedure as discussed in the previous section, the finite element solution of both the continuity and the unsteady momentum equations for both linear and quadratic elements results in a general system of linear-ordinary differential equations in the time domain (Equation [3.235]). This system of equations must be solved numerically to account for the variation in time. Several procedures can be implemented for the numerical solution of [3.235]. The finite difference method is the most commonly used method to approximate $d\{\Phi(t)\}/dt$ and $\Phi(t)$ at successive points in the time domain (Segerlind,,1984).

The mean value theorem for differentiation can be applied to the solution of [3.235] as discussed by Segerlind (1984). Given any function $\phi(t)$ and the interval $[a,b]$ as shown in Figure 5, the mean value theorem states that there is a ξ between a and b such that

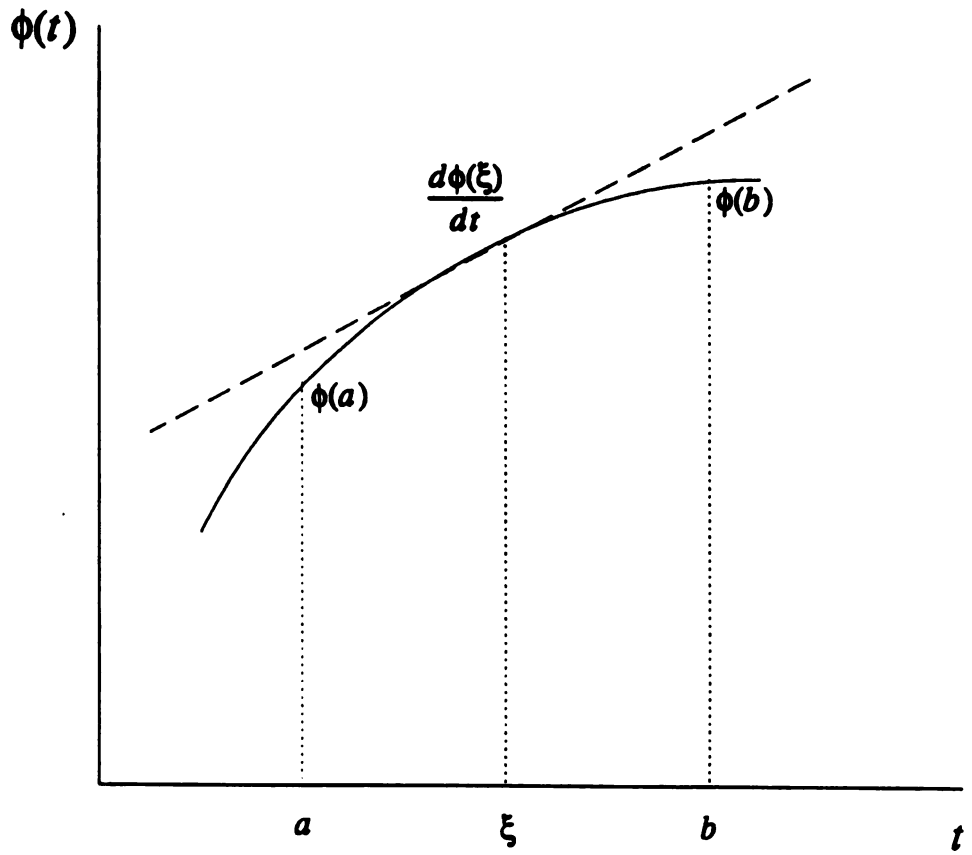


Figure 5. Plot of $\phi(t)$ as a function of time (Segerlind, 1984).

$$\phi(b)-\phi(a)=(b-a)\frac{d\phi}{dt}(\xi) \qquad [3.236]$$

or

$$\frac{d\phi}{dt}(\xi)=\frac{\phi(b)-\phi(a)}{\Delta t} \qquad [3.237]$$

where $\Delta t = a - b$.

The value of $\phi(a)$ can be approximated as

$$\phi(a)=\phi(\xi)-(\xi-a)\frac{d\phi}{dt}(\xi) \qquad [3.238]$$

or

$$\phi(\xi)=\phi(a)+(\xi-a)\frac{d\phi}{dt}(\xi) \qquad [3.239]$$

Substituting [3.237] in [3.239] results in

$$\phi(\xi)=\phi(a)+\frac{\phi(b)-\phi(a)}{\Delta t}(\xi-a) \qquad [3.240]$$

or

$$\begin{aligned}\phi(\xi)&=\phi(a)+[\phi(b)-\phi(a)]\theta\\ &=(1-\theta)\,\phi(a)+\theta\,\phi(b)\end{aligned} \qquad [3.241]$$

where $\theta = (\xi - a)/\Delta t$.

The results of Equations [3.237] and [3.241] can be generalized for column vectors (Segerlind, 1984). The resultant two equations at $t = \xi$ will be

$$\left\{\frac{d\phi(\xi)}{dt}\right\}=\{\Phi\}=\frac{\{\Phi\}_b-\{\Phi\}_a}{\Delta t} \qquad [3.242]$$

$$\{\Phi\} = (1-\theta)\{\Phi\}_a + \theta\{\Phi\}_b \qquad [3.243]$$

where $\{\Phi\}_a$ and $\{\Phi\}_b$ are two column vectors containing the nodal values at times a and b, respectively.

A similar approach can be followed to develop an equation for $\{F\}$ at $t = \xi$. The resultant equation is

$$\{F_G\} = (1-\theta)\{F_G\}_a + \theta\{F_G\}_b \qquad [3.244]$$

where $\{F\}_a$ and $\{F\}_b$ are the force vectors at times a and b, respectively.

Equations [3.242], [3.243], and [3.244] can now be replaced in [3.235]. The result is

$$[C_G]\left(\frac{\{\Phi\}_b - \{\Phi\}_a}{\Delta t}\right) + [K_G]((1-\theta)\{\Phi\}_a + \theta\{\Phi\}_b) - (1-\theta)\{F_G\}_a - \theta\{F_G\}_b = \{0\} \qquad [3.245]$$

or

$$([C_G] + \theta\Delta t[K_G])\{\Phi\}_b = ([C_G] - (1-\theta)\Delta t[K_G])\{\Phi\}_a + \Delta t((1-\theta)\{F_G\}_a + \theta\{F_G\}_b) \qquad [3.246]$$

or

$$[A_G]\{\Phi\}_b = [P_G]\{\Phi\}_a + \{F_G^*\} \qquad [3.247]$$

where

$$[A_G] = ([C_G] + \theta\Delta t[K_G]),$$

$$[P_G] = ([C_G] - (1-\theta)\Delta t[K_G]), \text{ and}$$

$$\{F_G^*\} = \Delta t((1-\theta)\{F_G\}_a + \theta\{F_G\}_b).$$

Equation [3.246] represents a general two-level-time scheme that gives the nodal values, $\{\Phi\}_b$, in terms of a set of known values, $\{\Phi\}_a$, and the force vectors at times a and b. The parameter θ should be specified to obtain a solution to [3.246]. The range of values of θ is in the interval [0, 1]. Selecting θ would in turn determine the location of ξ at which the mean value theorem is applied.

There are four popular methods that result from four choices of θ (Segerlind, 1984). These methods include the following :

i. Forward Difference Method : This method is obtained by specifying $\xi = a$. The resultant θ will be $\theta = 0$. Equation [3.246] reduces to

$$[C_G]\{\Phi\}_b = ([C_G] - \Delta t[K_G])\{\Phi\}_a + \Delta t\{F_G\}_a \qquad [3.248]$$

ii. Central Difference Method : This method is obtained by specifying $\xi = \Delta t/2 + a$. The resultant θ will be $\theta = 0.5$. Equation [3.246] reduces to

$$\left([C_G] + \frac{\Delta t}{2}[K_G]\right)\{\Phi\}_b = \left([C_G] - \frac{\Delta t}{2}[K_G]\right)\{\Phi\}_a + \left(\frac{\Delta t}{2}\{F_G\}_a + \frac{\Delta t}{2}\{F_G\}_b\right) \qquad [3.249]$$

iii. Galerkin Method : This method is obtained by specifying $\xi = 2\Delta t/3 + a$. The resultant θ will be $\theta = 2/3$. Equation [3.246] reduces to

$$\left([C_G] + \frac{2\Delta t}{3}[K_G]\right)\{\Phi\}_b = \left([C_G] - \frac{\Delta t}{3}[K_G]\right)\{\Phi\}_a + \left(\frac{\Delta t}{3}\{F_G\}_a + \frac{\Delta t}{3}\{F_G\}_b\right) \qquad [3.250]$$

iv. Backward Difference Method : This method is obtained by specifying $\xi = b$. The resultant θ will be $\theta = 1$. Equation [3.246] reduces to

$$([C_G] + \Delta t[K_G])\{\Phi\}_b = [C_G]\{\Phi\}_a + \Delta t\{F_G\}_b \qquad [3.251]$$

The global matrices $[C_G]$ and $[K_G]$ and the global vector $\{F_G\}$ which result from applying the direct stiffness procedure should be modified before the

initiation of the solution process. This modification is necessary to include the known nodal boundary conditions without which the problem is undefined. The resultant system of equations is solved then for every time step using a direct approach such as Gaussian elimination.

8. Implementing the Solution Procedure

In order to implement the finite element Galerkin formulation of the motion equations as was described earlier in this section, many relationships should be established to reduce the number of dependent variables in the solution process. These include establishing relationships among flow geometry parameters. Also, such relationships should be implemented in order to reduce the number of dependent parameters in the uniform flow equations which are utilized to establish the friction slope. Moreover, the infiltration function that will be used in the solution process should be selected.

a. Relationships Among Flow Geometry Parameters

In order to apply the finite element Galerkin formulation to the numerical solution of the complete or simplified forms of the hydrodynamic equations in furrows or borders, some mathematical relationships ought to be established among flow geometry parameters. These relationships are to reduce the number of dependent variables in the finite element formulations that were discussed in the Theoretical Development section of this chapter. The number of dependent variables can be reduced by relating the depth of flow, y, and the cross-sectional area of flow, A, in furrows. One approach is to select the following power curve (Elliott et al., 1982; Walker and Skogerboe, 1987):

$$y = \sigma_1 A^{\sigma_2} \qquad [3.252]$$

where σ_1 and σ_2 are empirical fitting constants. These constants reduce to unity in borders given a unit width of flow. The next step is to establish a mathematical relationship between the top width of flow, T, and the cross-sectional area of flow, A. Assuming a parabolic cross sectional area in furrows, the top width can be represented as (Chow, 1959)

$$T = \frac{3A}{2y} \qquad [3.253]$$

Substituting [3.252] in [3.253] results in

$$\begin{aligned} T &= \frac{3}{2}\frac{A}{\sigma_1 A^{\sigma_2}} \\ &= \frac{3}{2}\frac{A^{1-\sigma_2}}{\sigma_1} \\ &= \gamma_1 A^{\gamma_2} \end{aligned} \qquad [3.254]$$

where γ_1 and γ_2 are empirical fitting constants with $\gamma_1 = 3/2\sigma_1$ and $\gamma_2 = 1 - \sigma_2$. Equation [3.254] is only applicable to furrows with parabolic cross sections. These parameters are considered constants for any given furrow given the assumption of a prismatic channel. The parameters γ_1 and γ_2 in [3.254] will reduce to 1 and 1.5, respectively, when the case of borders is considered ($T = 1$, given a unit width of flow). Equation [3.252] is utilized in the finite element formulations of the zero-inertia and the steady and unsteady hydrodynamic models.

b. Uniform Flow Equations

The uniform flow equations (Equations [2.11], [2.12], and [2.13]) that were reviewed in the previous chapter can be written in this general form

$$S_f = \frac{Q^2}{k_1 A^2 R^{k_2}}$$

$$= \frac{Q^2}{k_1 A^2 \left(\frac{A}{P}\right)^{k_2}}$$

$$= \frac{Q^2 P^{k_2}}{k_1 A^{2+k_2}} \qquad [3.255]$$

where the parameters k_1 and k_2 are shown in Table 1, A is the cross-sectional area of flow, R is the hydraulic radius, P is the wetted perimeter, Q is the flow rate, and S_f is the friction slope. The number of dependent variables in [3.255] can be reduced by mathematically relating the wetted perimeter, P, and the cross-sectional area of flow, A, in furrows. Again, if a power curve is selected, the following relationship can be established

$$P = \beta_1 A^{\beta_2} \qquad [3.256]$$

where β_1 and β_2 are empirical fitting constants. Substituting Equation [3.256] in [3.255] results in

$$S_f = \frac{Q^2 \beta_1^{k_2} A^{k_2 \beta_2}}{k_1 A^{2+k_2}}$$

$$= \frac{Q^2}{k_1 \left(\beta_1^{-k_2}\right) A^{2+k_2-k_2\beta_2}}$$

$$= \frac{Q^2}{k_1 \rho_1 A^{\rho_2}} \qquad [3.257]$$

Table 1. Summary of the parameters k_1 and k_2 in the uniform flow equations.

Equation	Parameter k_1	Parameter k_2
Manning's	$1/n^2$	4/3
Chezy's	C^2	1
Darcy-Weisbach's	$8g/f$	1

Note:
n is Manning's roughness coefficient,
C is Chezy's roughness coefficient,
f is Darcy-Weisbach's roughness coefficient, and
g is the acceleration due to gravity.

where ρ_1 and ρ_2 are empirical fitting constants that are controlled by the hydraulic section of the furrow or border. The parameters ρ_1 and ρ_2 will vary based on the selected uniform flow equation. The above parameters are held constant for any given irrigation event. The parameters β_1 and β_2 reduce to 1 and 0, respectively, when the case of borders is considered given a unit width of flow. This reduces the parameters ρ_1 and ρ_2 to 1 and $2+k_2$, respectively.

c. Infiltration Functions

Any of the empirical infiltration functions that were reviewed in the previous chapter can be utilized to estimate infiltration in the finite element development of the motion equations as applied to surface irrigation problems. However, many scientists preferred to use the Kostiakov-Lewis infiltration equation in their developments. Although this infiltration function neglects to account for the effect of wetted perimeter changes on infiltration, it was reported in many studies that the Kostiakov-Lewis function produces good results especially when the definition of the parameters a, k, and f_0 was based on flow rates typical of the normal irrigating conditions. The Kostiakov-Lewis function has the form

$$Z = k\tau^a + f_0\tau \qquad [3.258]$$

or

$$I = \frac{\partial Z}{\partial \tau} = ak\tau^{a-1} + f_0 \qquad [3.259]$$

where a, k, and f_0 are fitting constants, Z is the cumulative infiltration, I is the infiltration rate, and τ is the opportunity time, to be differentiated from the time t.

IV. RESULTS AND ANALYSIS

The Galerkin formulation of the finite element method was used to solve the complete and simplified forms of the hydrodynamic model using both linear and quadratic one-dimensional elements. The Galerkin formulation was first applied to both the continuity and momentum (in its complete or simplified forms) equations with respect to the space coordinate for a fixed instant of time. This results in a system of first-order ordinary differential equations in the time domain. Then, a finite difference approximation in the time domain was applied to the final system of equations to generate a numerical solution. The direct stiffness procedure was utilized in building the global systems of equations at various time steps. The final system of equations were then modified to incorporate the boundary conditions of the advance, ponding, depletion, and recession phases. The dimensions of the total system of equations remain unchanged after the application of these boundary conditions at any given time step. The finite element Galerkin formulation was then utilized in building a general computer model. The latter model can be used in the analysis of water flow conditions in surface irrigation systems. Currently, only the kinematic wave finite element analysis is fully operational in the present version of the computer model.

The finite element Galerkin formulation that was applied in the development of the computer model will be discussed in this chapter, together with the analysis of the results from applying this model to the simulation of flow conditions in surface irrigation systems.

A. Finite Element Model Formulation

The finite element Galerkin formulation of the complete and simplified forms of the Hydrodynamic model was developed in the Chapter III using both linear and quadratic elements. This development produced a general system of equations for each of the latter elements (refer to Equations [3.218] and [3.233]). The generalized system of equations reduces to either the complete hydrodynamic model, the steady hydrodynamic model (where the time dependent term in the momentum equation is assumed negligible), the zero-inertia model, or the kinematic wave model through the selection of the appropriate coefficients which result in the respective model. The system of equations for the various elements are then assembled into a global system of equations using the direct stiffness procedure. This latter represents a system of nonlinear ordinary differential equations. The mean value theorem for differentiation is then applied to change this ordinary system of equation into a system of nonlinear algebraic equations. The next step includes the modification of this system of equations to incorporate the various boundary conditions. Then, the resultant system of equations is solved numerically using the Gauss elimination. These steps will be described in further detail in the following sub-sections using the linear element. It is important at this point to remember that the procedure that is followed in accomplishing the various tasks that were highlighted in this paragraph is similar for both linear and quadratic elements. The only difference between the two is reflected in the original equations of each element.

1. Assembling System of Equations

The finite element solution starts with one two-nodal element during the first time step (Figure 6). This element has two nodes: an upper node at the

flow inlet boundary and a lower node at the tip of the advancing front. Since the lower node is selected at the tip of the advancing front, both the flow rate (Q) and the cross-sectional area of flow (A) are zero at that node. The only unknowns during the first time step are the length of the element, which represents the distance that the moving front has advanced during the time step Δt_1, and the cross-sectional area of flow (A) at the upper boundary. This is true for all models except the kinematic wave model where the cross-sectional area of flow (A) at the upper boundary of the first element is determined from the uniform flow relationship depicted in Equation [3.254]. As time progresses, an element is added to the system of equations during each subsequent time step (Figure 7). The advance phase (Figure 8a) is concluded once the advancing front reaches the end of the furrow or border. After the completion of the advance phase, the number of elements remains the same after subsequent time steps during the ponding (Figure 8b) and depletion (Figure 9a) phases. After the cutoff time of water inflow the number of elements is reduced starting from the upstream end of the furrow or border as the cross-sectional area of flow approaches zero at various nodes. The solution process is concluded once the receding front reaches the end of the furrow or border at the end of the recession phase (Figure 9b).

In order to demonstrate the direct stiffness procedure which is implemented in assembling the global system of equations, four examples will be presented in this section using linear elements. These examples will pertain to the full hydrodynamic, steady hydrodynamic (hydrodynamic II), zero-inertia, and kinematic wave models, respectively. These examples are based on the discussion that was presented in Section B-5 of Chapter III. The number of elements that will be used in each of these examples is 3 which results in four nodes. The global capacitance matrix ($[C_G]$), stiffness matrix ($[K_G]$), and force vector ($\{F_G\}$) in Equation [3.232] which has the form

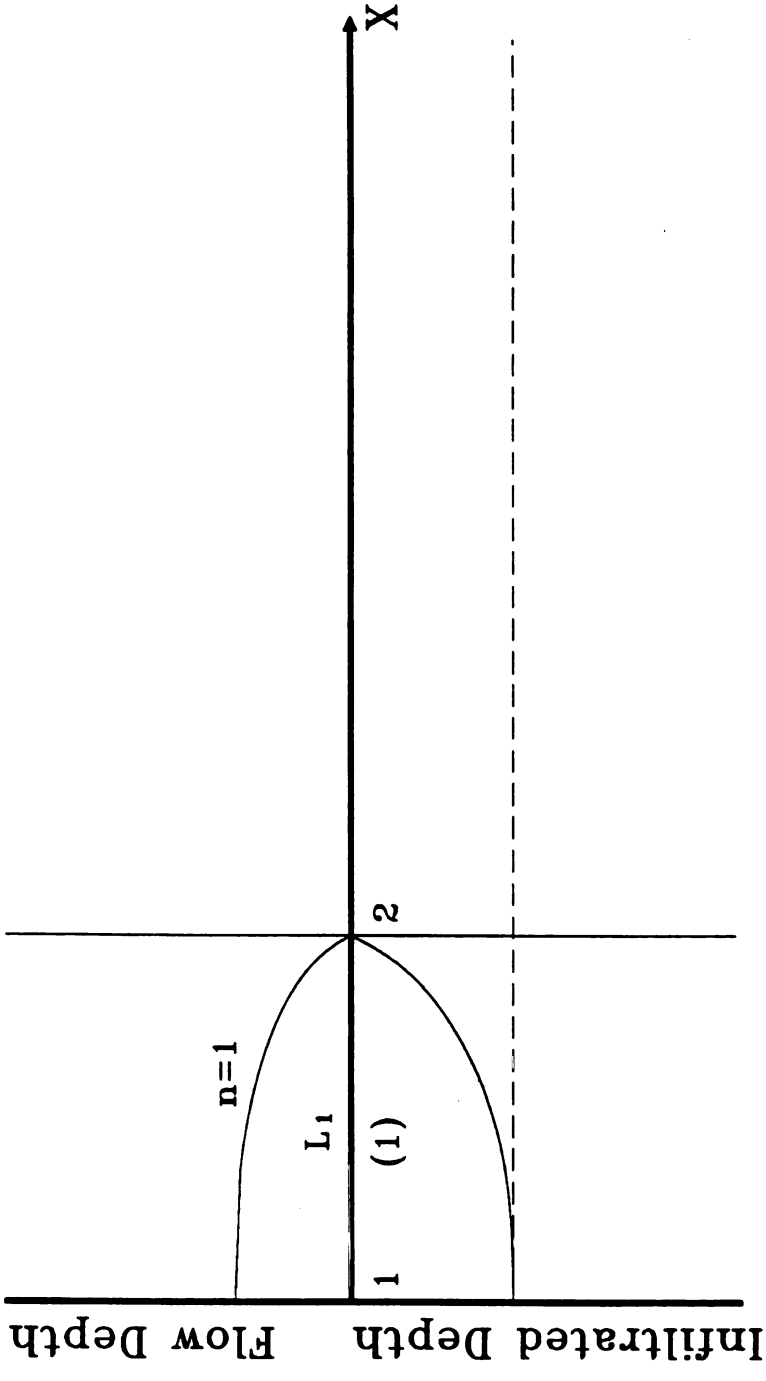


Figure 6. Schematic diagram of surface flow and infiltration during the first time step of the linear one-dimensional finite element solution.

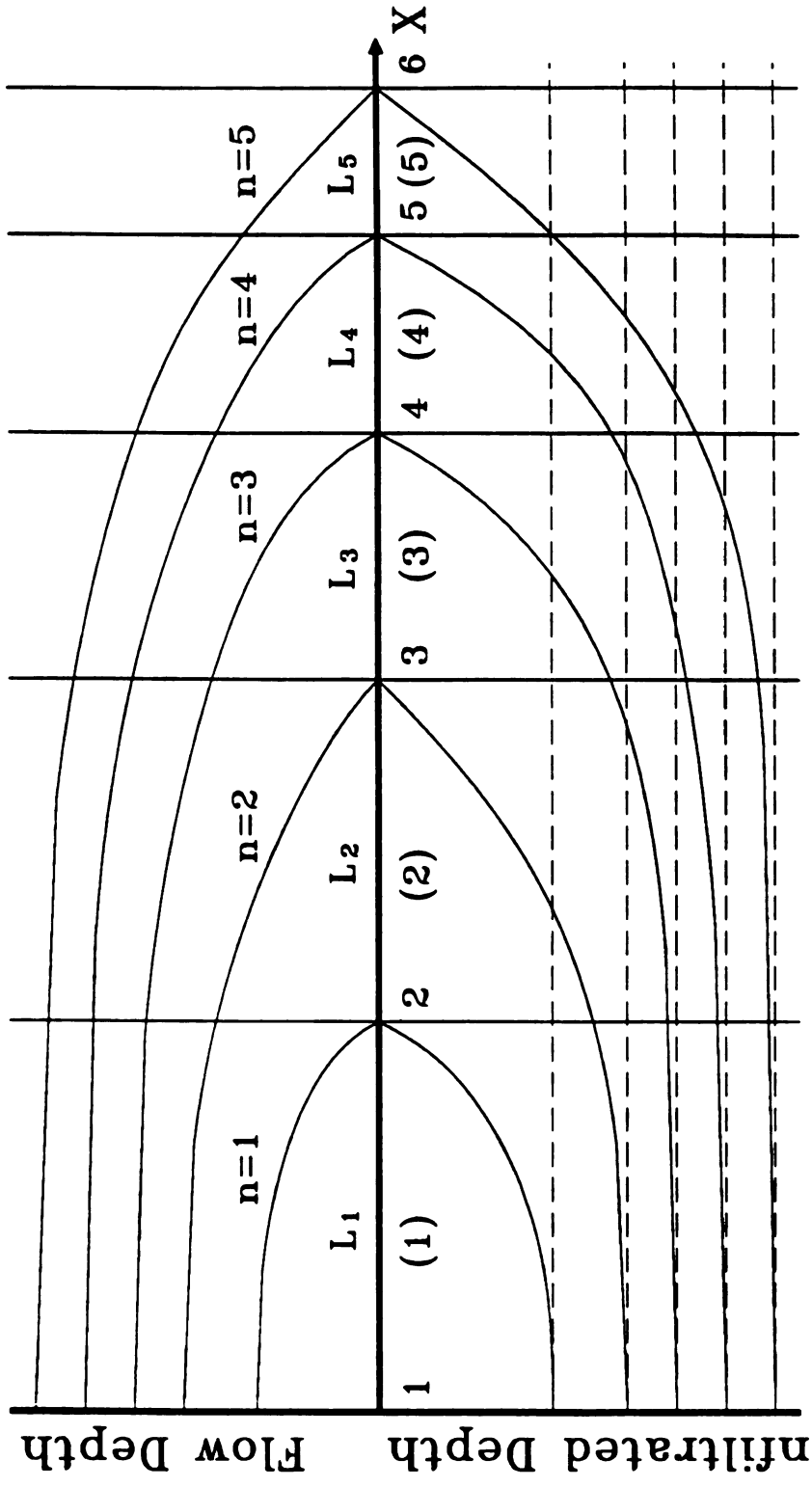


Figure 7. Schematic progression of surface flow and infiltration over linear one-dimensional finite element grid with a constant time step.

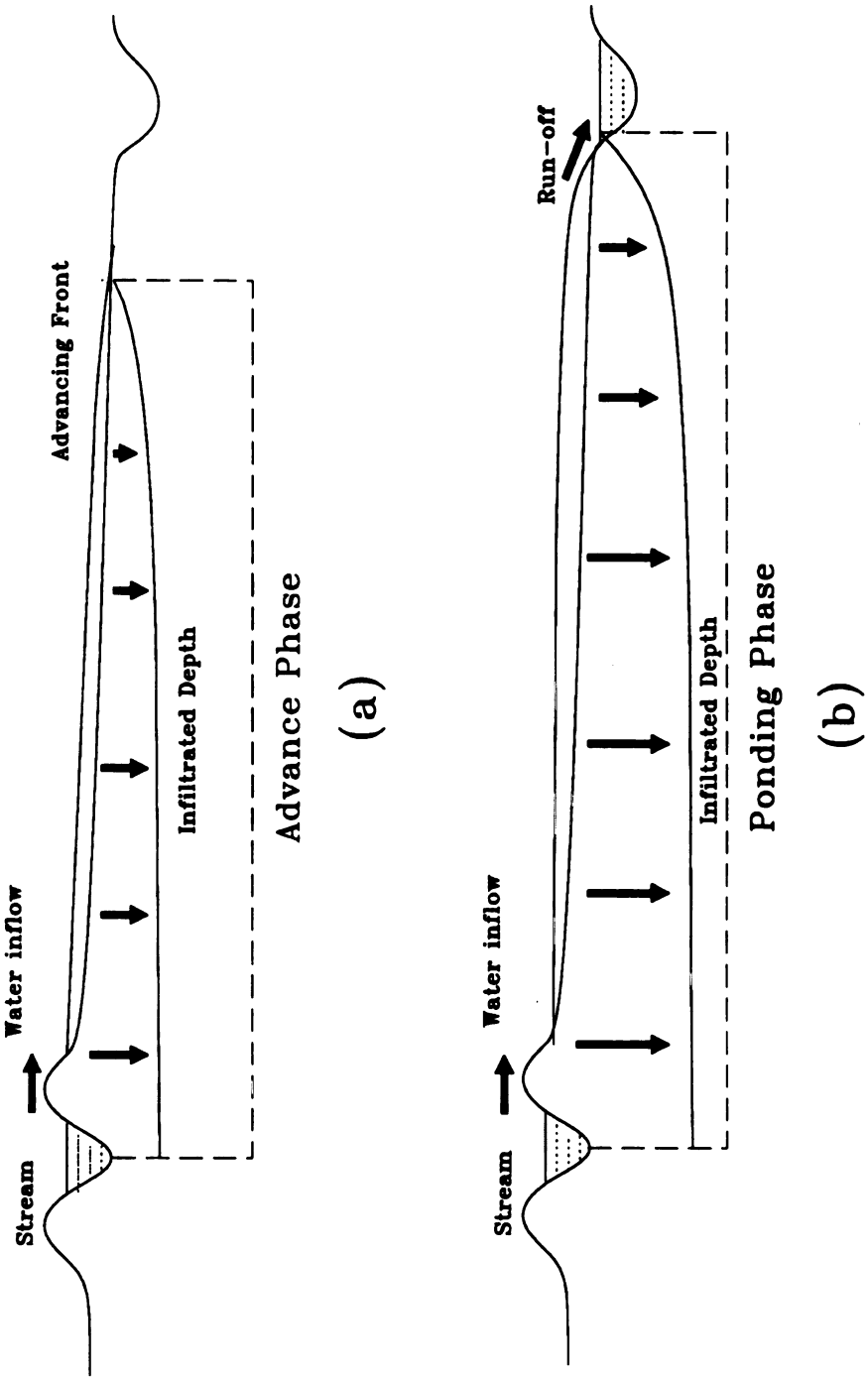


Figure 8. Schematic diagram of flow phases in surface irrigation systems: (a) advance phase, and (b) ponding phase.

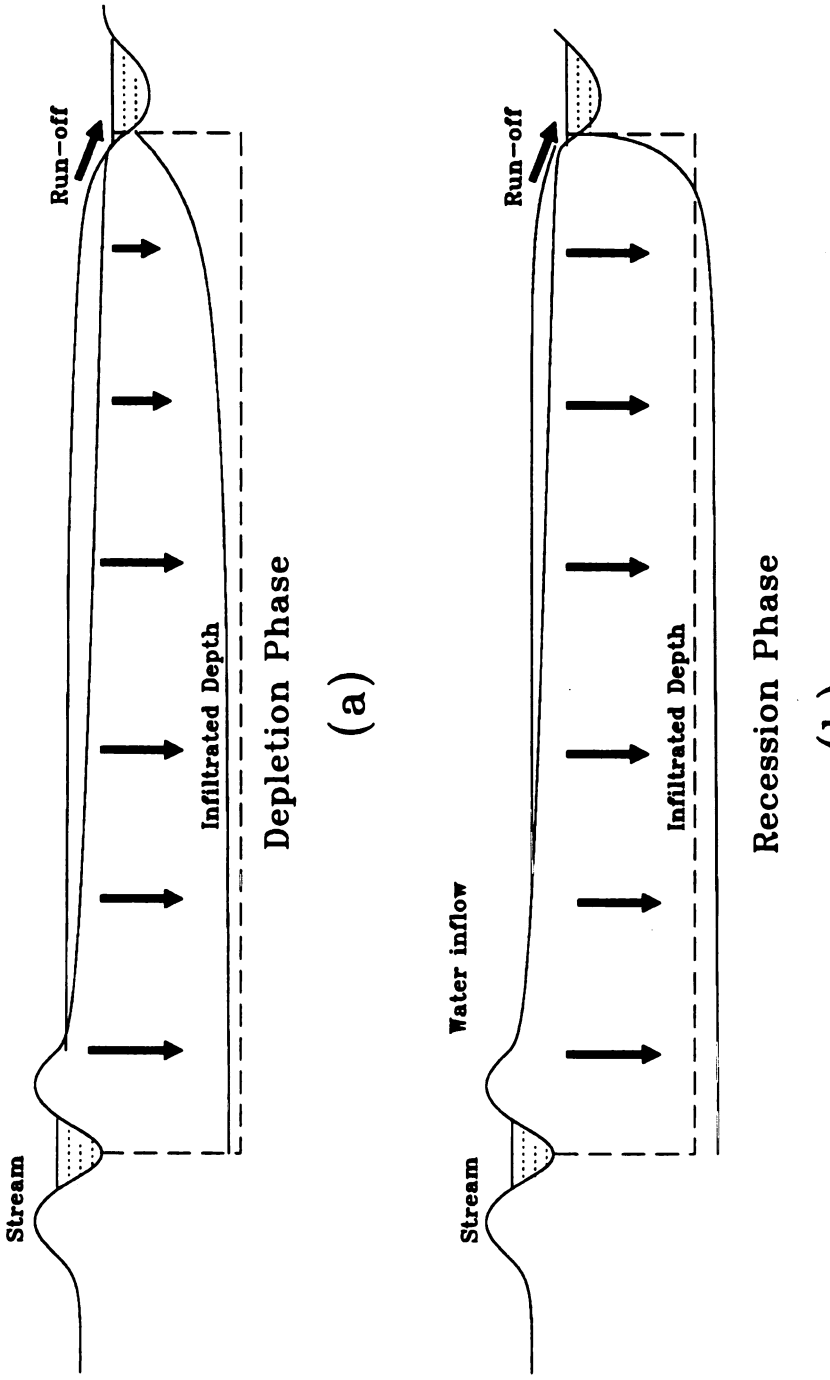


Figure 9. Schematic diagram of flow phases in surface irrigation systems: (a) depletion phase, and (b) recession phase.

$$\{R_G\} = [C_G]\{\dot{\Phi}\} + [K_G]\{\Phi\} - \{F_G\}$$

will have the form

Example 1: Full Hydrodynamic Model

$$[C_G] = \frac{1}{2}\begin{bmatrix} L_1 & 0 & 0 & 0 & & & & \\ 0 & \frac{L_1}{gA_1} & 0 & 0 & & & & \\ 0 & 0 & L_1+L_2 & 0 & 0 & 0 & & \\ 0 & 0 & 0 & \frac{L_1+L_2}{gA_2} & 0 & 0 & & \\ & & 0 & 0 & L_2+L_3 & 0 & 0 & 0 \\ & & 0 & 0 & 0 & \frac{L_2+L_3}{gA_3} & 0 & 0 \\ & & & & 0 & 0 & L_3 & 0 \\ & & & & 0 & 0 & 0 & \frac{L_3}{gA_4} \end{bmatrix} \qquad [4.1]$$

$$[F_G] = \begin{Bmatrix} -\frac{I_1 L_1}{2} & & & \\ \frac{L_1}{2}(S_{0_1} - S_{f_1}) & & & \\ -\frac{I_2 L_1}{2} & -\frac{I_2 L_2}{2} & & \\ \frac{L_1}{2}(S_{0_2} - S_{f_2}) & +\frac{L_2}{2}(S_{0_2} - S_{f_2}) & & \\ & -\frac{I_3 L_2}{2} & -\frac{I_3 L_3}{2} & \\ & \frac{L_2}{2}(S_{0_3} - S_{f_3}) & +\frac{L_3}{2}(S_{0_3} - S_{f_3}) & \\ & & -\frac{I_4 L_3}{2} & \\ & & \frac{L_3}{2}(S_{0_4} - S_{f_4}) & \end{Bmatrix} \qquad [4.2]$$

$$[K_G]=\frac{1}{2}\begin{bmatrix}
0 & -1+\alpha & 0 & 1+\alpha & & & & \\
(-1+\alpha)\left(\frac{1}{T_1}-\frac{Q_1^2}{gA_1^3}\right) & (-1+\alpha)\left(\frac{2Q_1}{gA_1^2}\right) & (1-\alpha)\left(\frac{1}{T_2}-\frac{Q_2^2}{gA_2^3}\right) & (1-\alpha)\left(\frac{2Q_2}{gA_2^2}\right) & & & & \\
0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha & & \\
(-1-\alpha)\left(\frac{1}{T_1}-\frac{Q_1^2}{gA_1^3}\right) & (-1-\alpha)\left(\frac{2Q_1}{gA_1^2}\right) & 2\alpha\left(\frac{1}{T_2}-\frac{Q_2^2}{gA_2^3}\right) & 2\alpha\left(\frac{2Q_2}{gA_2^2}\right) & (1-\alpha)\left(\frac{1}{T_3}-\frac{Q_3^2}{gA_3^3}\right) & (1-\alpha)\left(\frac{2Q_3}{gA_3^2}\right) & & \\
 & & 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha \\
 & & (-1-\alpha)\left(\frac{1}{T_2}-\frac{Q_2^2}{gA_2^3}\right) & (-1-\alpha)\left(\frac{2Q_2}{gA_2^2}\right) & 2\alpha\left(\frac{1}{T_3}-\frac{Q_3^2}{gA_3^3}\right) & 2\alpha\left(\frac{2Q_3}{gA_3^2}\right) & (1-\alpha)\left(\frac{1}{T_4}-\frac{Q_4^2}{gA_4^3}\right) & (1-\alpha)\left(\frac{2Q_4}{gA_4^2}\right) \\
 & & & & 0 & -1-\alpha & 0 & 1+\alpha \\
 & & & & (-1-\alpha)\left(\frac{1}{T_3}-\frac{Q_3^2}{gA_3^3}\right) & (-1-\alpha)\left(\frac{2Q_3}{gA_3^2}\right) & (1+\alpha)\left(\frac{1}{T_4}-\frac{Q_4^2}{gA_4^3}\right) & (1+\alpha)\left(\frac{2Q_4}{gA_4^2}\right)
\end{bmatrix} \quad [4.3]$$

$$\{\dot{\Phi}\}=\begin{Bmatrix}\partial A_1/\partial t\\ \partial Q_1/\partial t\\ \partial A_2/\partial t\\ \partial Q_2/\partial t\\ \partial A_3/\partial t\\ \partial Q_3/\partial t\\ \partial A_4/\partial t\\ \partial Q_4/\partial t\end{Bmatrix}=\begin{Bmatrix}\dot{A}_1\\ \dot{Q}_1\\ \dot{A}_2\\ \dot{Q}_2\\ \dot{A}_3\\ \dot{Q}_3\\ \dot{A}_4\\ \dot{Q}_4\end{Bmatrix} \quad [4.4] \qquad \text{and} \qquad \{\Phi\}=\begin{Bmatrix}A_1\\ Q_1\\ A_2\\ Q_2\\ A_3\\ Q_3\\ A_4\\ Q_4\end{Bmatrix} \quad [4.5]$$

Example 2: Steady Hydrodynamic Model (Hydrodynamic II)

$$[C_G]=\frac{1}{2}\begin{bmatrix} L_1 & 0 & 0 & 0 & & & & \\ 0 & 0 & 0 & 0 & & & & \\ 0 & 0 & L_1+L_2 & 0 & 0 & 0 & & \\ 0 & 0 & 0 & 0 & 0 & 0 & & \\ & & 0 & 0 & L_2+L_3 & 0 & 0 & 0 \\ & & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & 0 & 0 & L_3 & 0 \\ & & & & 0 & 0 & 0 & 0 \end{bmatrix} \qquad [4.6]$$

$$[F_G]=\begin{Bmatrix} \frac{-I_1L_1}{2} & & & \\ \frac{L_1}{2}(S_{0_1}-S_{f_1}) & & & \\ -\frac{I_2L_1}{2} & -\frac{I_2L_2}{2} & & \\ \frac{L_1}{2}(S_{0_2}-S_{f_2}) & +\frac{L_2}{2}(S_{0_2}-S_{f_2}) & & \\ & -\frac{I_3L_2}{2} & -\frac{I_3L_3}{2} & \\ & \frac{L_2}{2}(S_{0_3}-S_{f_3}) & +\frac{L_3}{2}(S_{0_3}-S_{f_3}) & \\ & & -\frac{I_4L_3}{2} & \\ & & \frac{L_3}{2}(S_{0_4}-S_{f_4}) & \end{Bmatrix} \qquad [4.7]$$

$$[K_G]=\frac{1}{2}\begin{bmatrix}
0 & -1+\alpha & 0 & 1+\alpha & & & & \\
(-1+\alpha)\left(\frac{1}{T_1}-\frac{Q_1^2}{gA_1^3}\right) & (-1+\alpha)\left(\frac{2Q_1}{gA_1^2}\right) & (1-\alpha)\left(\frac{1}{T_2}-\frac{Q_2^2}{gA_2^3}\right) & (1-\alpha)\left(\frac{2Q_2}{gA_2^2}\right) & & & & \\
0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha & & \\
(-1-\alpha)\left(\frac{1}{T_1}-\frac{Q_1^2}{gA_1^3}\right) & (-1-\alpha)\left(\frac{2Q_1}{gA_1^2}\right) & 2\alpha\left(\frac{1}{T_2}-\frac{Q_2^2}{gA_2^3}\right) & 2\alpha\left(\frac{2Q_2}{gA_2^2}\right) & (1-\alpha)\left(\frac{1}{T_3}-\frac{Q_3^2}{gA_3^3}\right) & (1-\alpha)\left(\frac{2Q_3}{gA_3^2}\right) & & \\
 & & 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha \\
 & & (-1-\alpha)\left(\frac{1}{T_2}-\frac{Q_2^2}{gA_2^3}\right) & (-1-\alpha)\left(\frac{2Q_2}{gA_2^2}\right) & 2\alpha\left(\frac{1}{T_3}-\frac{Q_3^2}{gA_3^3}\right) & 2\alpha\left(\frac{2Q_3}{gA_3^2}\right) & (1-\alpha)\left(\frac{1}{T_4}-\frac{Q_4^2}{gA_4^3}\right) & (1-\alpha)\left(\frac{2Q_4}{gA_4^2}\right) \\
 & & & & 0 & -1-\alpha & 0 & 1+\alpha \\
 & & & & (-1-\alpha)\left(\frac{1}{T_3}-\frac{Q_3^2}{gA_3^3}\right) & (-1-\alpha)\left(\frac{2Q_3}{gA_3^2}\right) & (1+\alpha)\left(\frac{1}{T_4}-\frac{Q_4^2}{gA_4^3}\right) & (1+\alpha)\left(\frac{2Q_4}{gA_4^2}\right)
\end{bmatrix} \qquad [4.8]$$

$$\{\Phi\}=\begin{Bmatrix}\partial A_1/\partial t\\ \partial Q_1/\partial t\\ \partial A_2/\partial t\\ \partial Q_2/\partial t\\ \partial A_3/\partial t\\ \partial Q_3/\partial t\\ \partial A_4/\partial t\\ \partial Q_4/\partial t\end{Bmatrix}=\begin{Bmatrix}\dot{A}_1\\ \dot{Q}_1\\ \dot{A}_2\\ \dot{Q}_2\\ \dot{A}_3\\ \dot{Q}_3\\ \dot{A}_4\\ \dot{Q}_4\end{Bmatrix} \qquad [4.9]$$

and

$$\{\Phi\}=\begin{Bmatrix}A_1\\ Q_1\\ A_2\\ Q_2\\ A_3\\ Q_3\\ A_4\\ Q_4\end{Bmatrix} \qquad [4.10]$$

Example 3: Zero-Inertia Model

$$[C_G]=\frac{1}{2}\begin{bmatrix} L_1 & 0 & 0 & 0 & & & & \\ 0 & 0 & 0 & 0 & & & & \\ 0 & 0 & L_1+L_2 & 0 & 0 & 0 & & \\ 0 & 0 & 0 & 0 & 0 & 0 & & \\ & & 0 & 0 & L_2+L_3 & 0 & 0 & 0 \\ & & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & 0 & 0 & L_3 & 0 \\ & & & & 0 & 0 & 0 & 0 \end{bmatrix} \qquad [4.11]$$

$$[F_G]=\frac{1}{2}\begin{Bmatrix} -I_1L_1 & & \\ L_1S_{0_1} & & \\ -I_2L_1 & -I_2L_2 & \\ L_1S_{0_2} & +L_2S_{0_2} & \\ & -I_3L_2 & -I_3L_3 \\ & L_2S_{0_3} & +L_3S_{0_3} \\ & & -I_4L_3 \\ & & L_3S_{0_4} \end{Bmatrix} \qquad [4.12]$$

$$[K_G]=\frac{1}{2}\begin{bmatrix} 0 & -1+\alpha & 0 & 1+\alpha & & & & \\ \frac{-1+\alpha}{T_1} & \frac{L_1n^2Q_1}{\rho_1A_1^{\prime 2}} & \frac{1-\alpha}{T_2} & 0 & & & & \\ 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha & & \\ \frac{-1-\alpha}{T_1} & 0 & \frac{2\alpha}{T_2} & (L_1+L_2)\frac{n^2Q_2}{\rho_1A_2^{\prime 2}} & \frac{1-\alpha}{T_3} & 0 & & \\ & & 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha \\ & & \frac{-1-\alpha}{T_2} & 0 & \frac{2\alpha}{T_3} & (L_2+L_3)\frac{n^2Q_3}{\rho_1A_3^{\prime 2}} & \frac{1-\alpha}{T_4} & 0 \\ & & & & 0 & -1-\alpha & 0 & 1+\alpha \\ & & & & \frac{-1-\alpha}{T_3} & 0 & \frac{1+\alpha}{T_4} & \frac{L_3n^2Q_4}{\rho_1A_4^{\prime 2}} \end{bmatrix} \qquad [4.13]$$

Example 4: Kinematic Wave Model

$$[C_G]=\frac{1}{2}\begin{bmatrix} L_1 & 0 & 0 & 0 & & & & \\ 0 & 0 & 0 & 0 & & & & \\ 0 & 0 & L_1+L_2 & 0 & 0 & 0 & & \\ 0 & 0 & 0 & 0 & 0 & 0 & & \\ & & 0 & 0 & L_2+L_3 & 0 & 0 & 0 \\ & & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & 0 & 0 & L_3 & 0 \\ & & & & 0 & 0 & 0 & 0 \end{bmatrix} \qquad [4.14]$$

$$[F_G]=\begin{Bmatrix} 0 & \\ 0 & \\ 0 & +0 \\ 0 & +0 \\ 0 & +0 \\ 0 & +0 \\ & 0 \\ & 0 \end{Bmatrix} \qquad [4.15]$$

$$[K_G]=\frac{1}{2}\begin{bmatrix} 0 & -1+\alpha & 0 & 1+\alpha & & & & \\ \frac{1}{n}\rho_1 A_1^{\rho_2-2} S_{0_1} & -1 & 0 & 0 & & & & \\ 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha & & \\ 0 & 0 & \frac{1}{n}\rho_1 A_2^{\rho_2-2} S_{0_2} & -1 & 0 & 0 & & \\ & & 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha \\ & & 0 & 0 & \frac{1}{n}\rho_1 A_3^{\rho_2-2} S_{0_3} & -1 & 0 & 0 \\ & & & & 0 & -1-\alpha & 0 & 1+\alpha \\ & & & & 0 & 0 & \frac{1}{n}\rho_1 A_4^{\rho_2-2} S_{0_4} & -1 \end{bmatrix} \qquad [4.16]$$

2. Incorporating Boundary Conditions

Most of the boundary conditions in the one-dimensional surface irrigation problem involve boundaries with known-values. These are referred to as boundary conditions of the first kind or Dirichlet boundary conditions. Generally, there are five of these boundary conditions in sloping border and furrow irrigation systems. The first is the upstream boundary during the advance phase (Figure 8a) where the inlet flow to the furrow, or the unit inlet flow to the border, is known. This represents the case where the flow rate at node 1 (Q_1) is known. During the advance phase, another known boundary exists at the tip of the advancing front. At this boundary, both the flow rate and the cross-sectional area of flow at that node are zero. Once the advancing front reaches the end of the furrow or border, two boundary conditions may occur. The first involves the case when tail water is draining from the end of the field. At this instance, a flow is assumed to be uniform at the downstream node of the last element. The other boundary could involve the case where a barrier at the downstream node of the last element stops the forward flow of water. This boundary translates to a flow rate of zero at the downstream node of the last element while the cross-sectional area of flow increases as the water ponding phase (Figure 8b) starts. The last boundary represents the upstream boundary as the receding edge moves downstream during the recession phase (Figure 9b) of flow. Both the flow rate and cross-sectional area of flow at the upper node of the upstream element of the receding front are zero.

As for the recession phase, the upstream node that has a cross-sectional area of flow approaching zero dictates that the receding front has reached that node which would mean that the last boundary that was described in the last paragraph is applicable. Then, the receding front moves to the upper node of

the next downstream element if the same conditions occurs. This process continues on until the receding front reaches the upper node of the last downstream element.

The various boundary conditions are implemented in the solution process of the finite element Galerkin formulation so as to result in a well posed problem. These known boundary conditions are implemented after the global system of equations is assembled using the direct stiffness procedure of the finite element method as was discussed in the previous section. The global system of equations at each time step is modified to incorporate the various known boundary conditions starting with the initial conditions where the global system of equations contains the contribution of only one element. The same procedure is then repeated at subsequent time steps until the solution process of the various phases of flow is concluded.

The general form of the global system of equations was developed in Section B-7 of Chapter III after implementing the mean value theorem for differentiation. This resulted in Equation [3.246] which has the form

$$([C_G]+\theta\Delta t[K_G])\{\Phi\}_b=([C_G]-(1-\theta)\Delta t[K_G])\{\Phi\}_a+\Delta t((1-\theta)\{F_G\}_a+\theta\{F_G\}_b) \qquad [4.16]$$

In order to implement known boundary conditions, the system of equations above ([4.16]) should be modified at each time step. However, the modification should not be done before determining the matrices $[A_G]$ and $[P_G]$ and the vector $\{F_G^*\}$ in the equation

$$[A_G]\{\Phi\}_b=[P_G]\{\Phi\}_a+\{F_G^*\}=\{S_G\} \qquad [4.18]$$

where

$$[A_G] = ([C_G] + \theta \Delta t [K_G]),$$

$$[P_G] = ([C_G] - (1-\theta)\Delta t [K_G]), \text{ and}$$

$$\{F_G^*\} = \Delta t((1-\theta)\{F_G\}_a + \theta\{F_G\}_b).$$

If the number of equations is n and Φ_k is the known value (k is *1,2,3,...,* or n), the modification of the system of equations in [4.18] is accomplished using the following steps:

1. Subtract the product $A_{i,k}\Phi_{bk}$ with $i = 1,2,....,n$ from the corresponding coefficient in the vector $\{F_G^*\}$.
2. Replace the coefficients in row k and column k of the matrix $[A_G]$ by zeros.
3. Replace the coefficient $A_{k,k}$ in the $[A_G]$ matrix by one.
4. Add the product $P_{i,k}\Phi_{ak}$ with $i = 1,2,....,n$ to the corresponding coefficient in the vector $\{F_G^*\}$.
5. Replace the coefficients in row k and column k of the matrix $[P_G]$ by zeros.
6. Replace the coefficient in row k of the $\{F_G^*\}$ vector by Φ_{bk}.

The nodal values Φ_{bk} and Φ_{ak} represent Φ_k at times b and a, respectively. This would allow the known boundary conditions to be specified as inflow or outflow hydrograph.

3. Numerical Solution

After assembling the system of equations using the direct stiffness procedure as was discussed in Section A-1 of this chapter, the system of equations is solved iteratively at each time step. The iterative solution procedure is a necessity since the algebraic equations that make up the global system of equations, which resulted from implementing the mean value theorem for differentiation, is nonlinear.

The general finite element Galerkin formulation of the complete and simplified forms of the hydrodynamic model which were developed in Chapter III using both linear and quadratic elements has some particular characteristics that are specific to the selected model. The difference is primarily between the full hydrodynamic model on the one hand and the three simplified forms of the hydrodynamic model (namely the steady hydrodynamic, zero-inertia, and kinematic wave models) on the other hand. The complete hydrodynamic model is based on two unsteady nonlinear partial differential equations. The simplified hydrodynamic models are based on one unsteady partial differential equation (continuity equation) and one steady partial differential equation (simplified momentum equation). Therefore, the mean value theorem of differentiation applies to both system of equations in the case of full hydrodynamic model while it applies only to the continuity equation in the case of the simplified hydrodynamic models. This would then translate into the fact that the ordinary differential equations with odd numbers in equation [3.235], which represent the contribution of the continuity equation to the final global system of equations, should be transformed to algebraic equations using the mean value theorem of differentiation. However, the equations with even numbers in [3.235] would need the same procedure only for the case of the complete hydrodynamic model. In other words, the mean value theorem for differentiation is not needed in the latter case when considering the simplified hydrodynamic models. Instead, these equations should have the form

$$[K_G]\{\Phi\}_b = \{F_G\}_b \qquad [4.19]$$

In order to satisfy both [4.16] and [4.19] for the complete and simplified hydrodynamic models, the following equation is developed:

$$([C_G] + c_1[K_G])\{\Phi\}_b = ([C_G] - c_2[K_G])\{\Phi\}_a + c_2\{F_G\}_a + c_1\{F_G\}_b \qquad [4.20]$$

where the parameters c_1 and c_2 are defined in Tables 2 and 3.

The Gauss elimination routine was used in this development for the solution of the non-linear system of simultaneous algebraic equations which emerged from the global system of ordinary differential equations based on the discussion earlier in this section. The Gauss elimination routine (GAUSSBND) implements the solution of the system of equations in two steps. The first involves the forward elimination which reduces the set of algebraic equations to an upper triangular system. The next phase utilizes the backward substitution to produce a solution to the unknowns in the system of equations. GAUSSBND implements partial pivoting, a step that involves the switching of rows to make the largest element the pivot element. This step is accomplished without exchanging the rows physically, but by keeping track of the order of equations in an array. The partial pivoting makes the routine applicable to the solution of sparse and ill-conditioned system of equations.

The developed Gauss elimination routine was modified to solve the final system of equations in a banded form since the resultant system of equations is banded with bandwidths of 4 and 6 for the linear and quadratic elements (refer to Section B-6 of chapter III), respectively. For this reason, the system of equations is assembled in banded form using the direct stiffness procedure. This step reduces the computer storage requirements of the $[A_G]$, $[P_G]$, $[C_G]$, and $[K_G]$ matrices in Equations [4.17] and [4.19]. The dimensions of these matrices are defined in Table 5 when these matrices are stored in full and banded forms. The advantages of this approach are demonstrated in Figures 10, 11, 12, and 13 for both linear and quadratic elements.

Table 2. Definition of the parameters c_1 and c_2 in Equation [4.20] for equations with odd numbers in the global system of equations.

Model	Parameter c_1	Parameter c_2
Complete Hydrodynamic Model (Hydrodynamic Model I)	$\theta\Delta t$	$(1-\theta)\Delta t$
Steady Hydrodynamic Model (Hydrodynamic Model II)	$\theta\Delta t$	$(1-\theta)\Delta t$
Zero-Inertia Model	$\theta\Delta t$	$(1-\theta)\Delta t$
Kinematic wave Model	$\theta\Delta t$	$(1-\theta)\Delta t$

Table 3. Definition of the parameters c_1 and c_2 in Equation [4.20] for equations with even numbers in the global system of equations.

Model	Parameter c_1	Parameter c_2
Complete Hydrodynamic Model (Hydrodynamic Model I)	$\theta\Delta t$	$(1-\theta)\Delta t$
Steady Hydrodynamic Model (Hydrodynamic Model II)	1	0
Zero-Inertia Model	1	0
Kinematic wave Model	1	0

Table 4. Dimensions of the matrices $[C_G]$, $[K_G]$, $[A_G]$, $[P_G]$, and $[X_G]$ in full and banded forms.

Matrix	Number of Columns, c			
	Linear Element ($n_e = 2$ and $n = e + 1$)		Quadratic Element ($n_e = 3$ and $n = 2e + 1$)	
	Full Matrix	Banded Matrix	Full Matrix	Banded Matrix
$[C_G]$	$2n$	$4n_e - 1 = 7$	$2n$	$4n_e - 1 = 11$
$[K_G]$	$2n$	$4n_e - 1 = 7$	$2n$	$4n_e - 1 = 11$
$[A_G]$	$2n$	$4n_e - 1 = 7$	$2n$	$4n_e - 1 = 11$
$[P_G]$	$2n$	$4n_e - 1 = 7$	$2n$	$4n_e - 1 = 11$
$[X_G]$	$2n$	$6n_e - 1 = 11$	$2n$	$6n_e - 1 = 17$

r : number of rows = $2n$
c : number of columns
e : number of elements
n : number of nodes = $(n_e - 1)e + 1$
n_e: number of nodes per element
$[X_G]$: the matrix that corresponds to $[A_G]$ in Equation [4.18] once the GAUSSBND subprogram is called

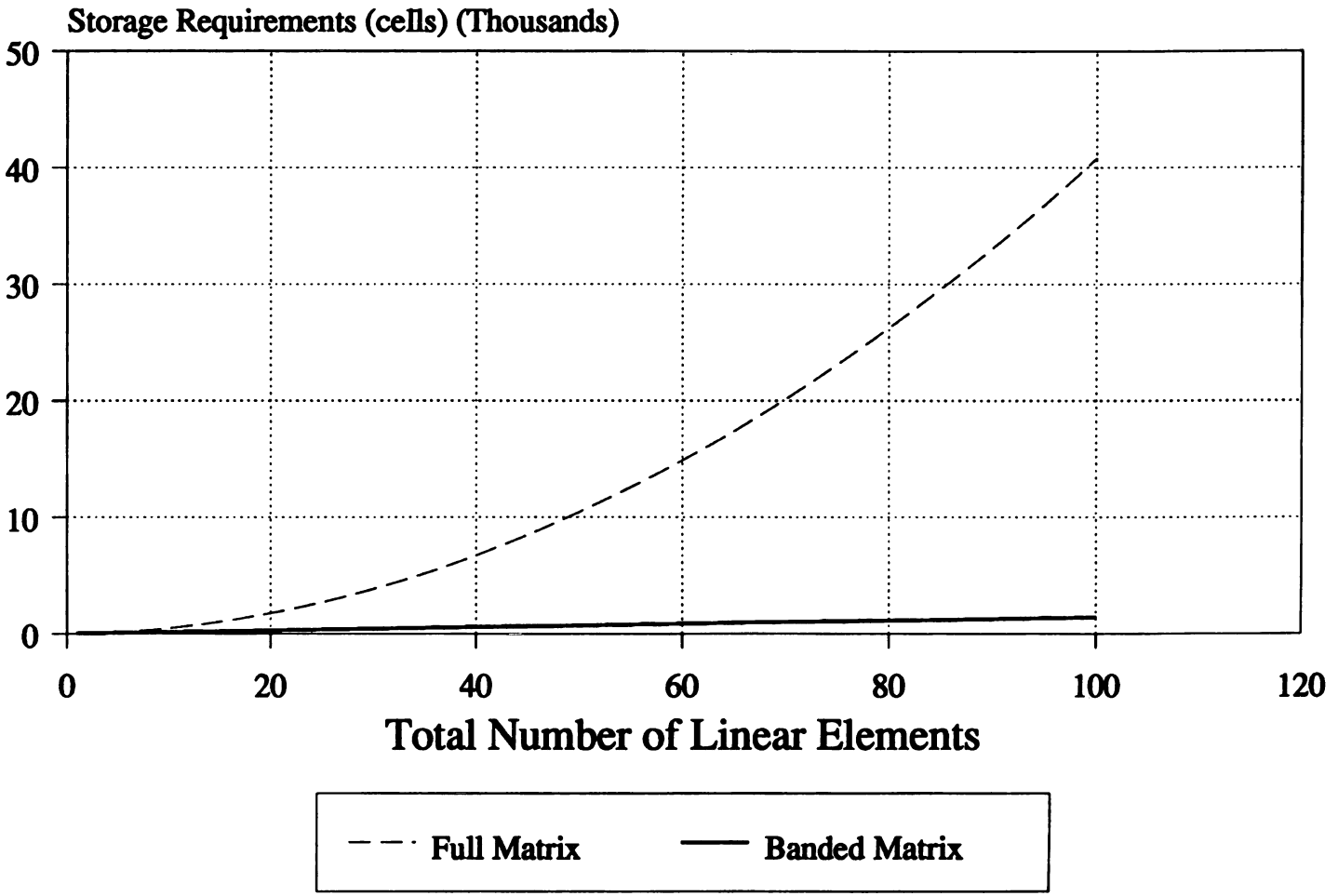


Figure 10. Linear plot of storage requirements of the global stiffness or capacitance matrices versus total number of linear elements.

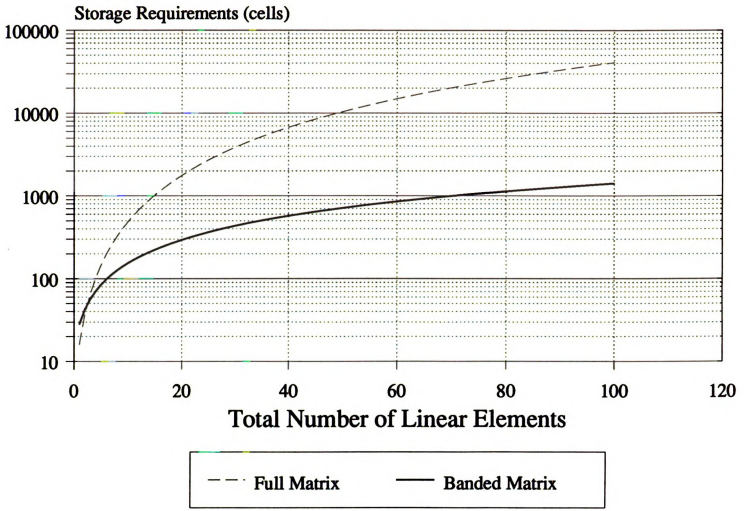


Figure 11. **Semi-log plot of storage requirements of the global stiffness or capacitance matrices versus total number of linear elements.**

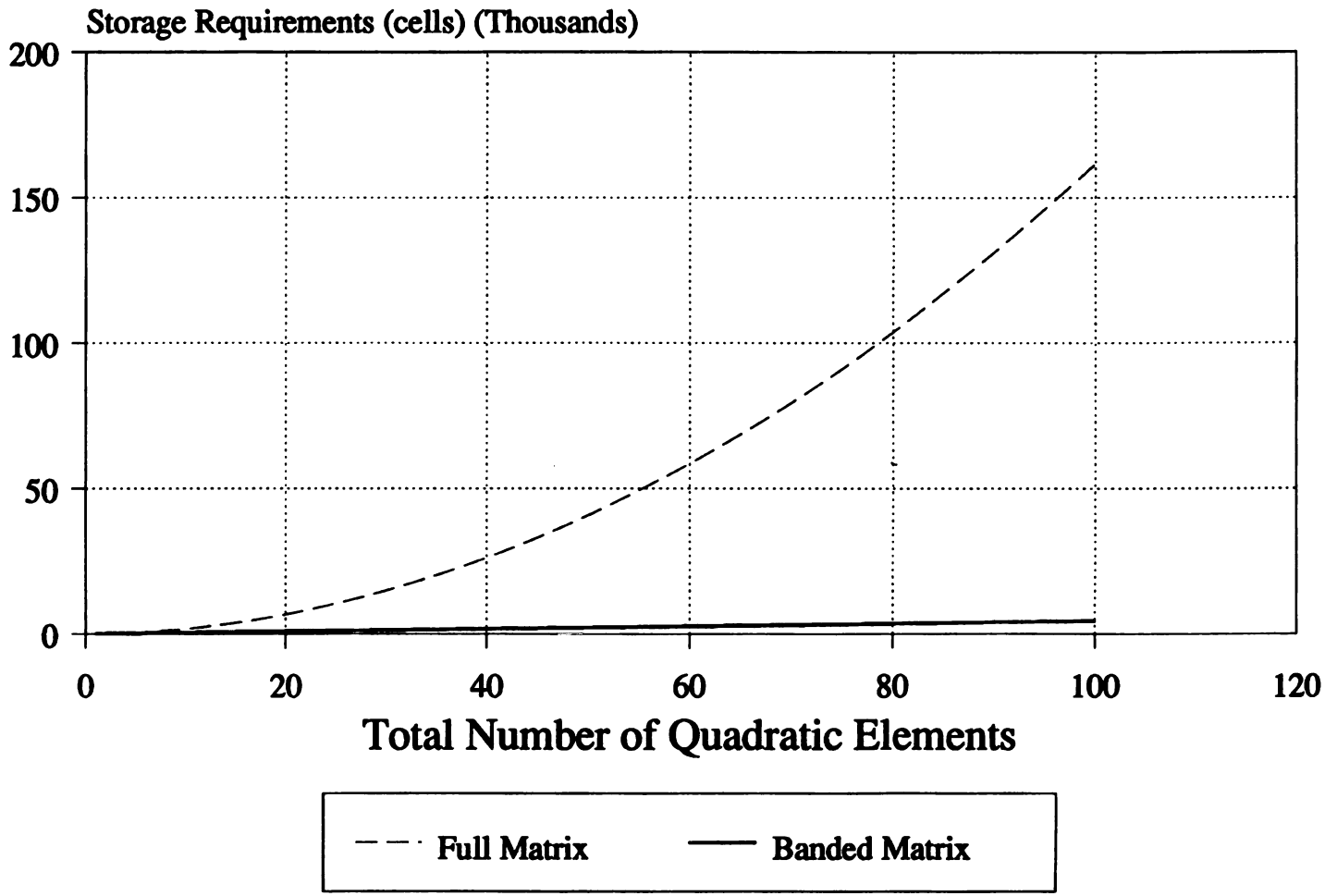


Figure 12. Linear plot of storage requirements of the global stiffness or capacitance matrices versus total number of quadratic elements.

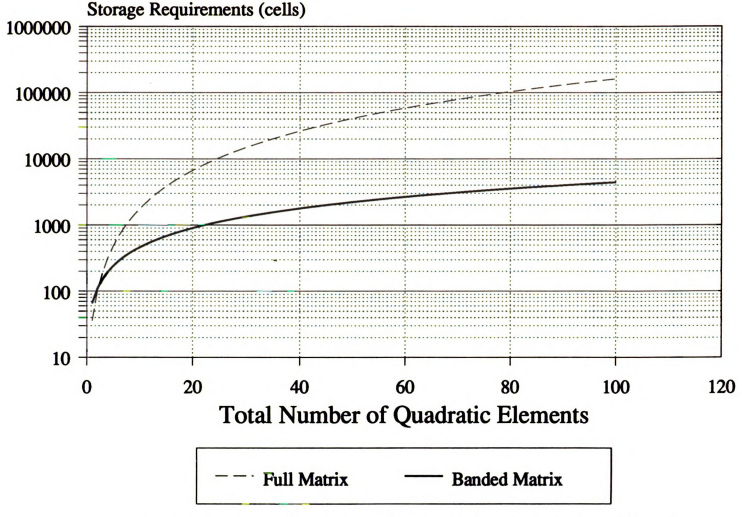


Figure 13. Semi-log plot of storage requirements of the global stiffness or capacitance matrices versus total number of quadratic elements.

In order to demonstrate the assembling of the global system of equations in a banded format, the four examples that were presented earlier will be presented in this section in a banded form. The global capacitance ($[C_G]$) and stiffness ($[K_G]$) matrices of Equation [3.232] will have the form

Example 1: Full Hydrodynamic Model

$$[C_G]=\frac{1}{2}\begin{bmatrix} * & * & * & L_1 & 0 & 0 & 0 \\ * & * & 0 & \frac{L_1}{gA_1} & 0 & 0 & \\ * & 0 & 0 & L_1+L_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{L_1+L_2}{gA_2} & 0 & 0 & \\ & 0 & 0 & L_2+L_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{L_2+L_3}{gA_3} & 0 & 0 & * \\ & 0 & 0 & L_3 & 0 & * & * \\ 0 & 0 & 0 & \frac{L_3}{gA_4} & * & * & * \end{bmatrix} \qquad [4.21]$$

$$[K_G]=\frac{1}{2}\begin{bmatrix} * & * & * & 0 & -1+\alpha & 0 & 1+\alpha \\ * & * & (-1+\alpha)\left(\frac{1}{T_1}-\frac{Q_1^2}{gA_1^3}\right) & (-1+\alpha)\left(\frac{2Q_1}{gA_1^2}\right) & (1-\alpha)\left(\frac{1}{T_2}-\frac{Q_2^2}{gA_2^3}\right) & (1-\alpha)\left(\frac{2Q_2}{gA_2^2}\right) & \\ * & 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha \\ (-1-\alpha)\left(\frac{1}{T_1}-\frac{Q_1^2}{gA_1^3}\right) & (-1-\alpha)\left(\frac{2Q_1}{gA_1^2}\right) & 2\alpha\left(\frac{1}{T_2}-\frac{Q_2^2}{gA_2^3}\right) & 2\alpha\left(\frac{2Q_2}{gA_2^2}\right) & (1-\alpha)\left(\frac{1}{T_3}-\frac{Q_3^2}{gA_3^3}\right) & (1-\alpha)\left(\frac{2Q_3}{gA_3^2}\right) & \\ & 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha \\ (-1-\alpha)\left(\frac{1}{T_2}-\frac{Q_2^2}{gA_2^3}\right) & (-1-\alpha)\left(\frac{2Q_2}{gA_2^2}\right) & 2\alpha\left(\frac{1}{T_3}-\frac{Q_3^2}{gA_3^3}\right) & 2\alpha\left(\frac{2Q_3}{gA_3^2}\right) & (1-\alpha)\left(\frac{1}{T_4}-\frac{Q_4^2}{gA_4^3}\right) & (1-\alpha)\left(\frac{2Q_4}{gA_4^2}\right) & * \\ & 0 & -1-\alpha & 0 & 1+\alpha & * & * \\ (-1-\alpha)\left(\frac{1}{T_3}-\frac{Q_3^2}{gA_3^3}\right) & (-1-\alpha)\left(\frac{2Q_3}{gA_3^2}\right) & (1+\alpha)\left(\frac{1}{T_4}-\frac{Q_4^2}{gA_4^3}\right) & (1+\alpha)\left(\frac{2Q_4}{gA_4^2}\right) & * & * & * \end{bmatrix}$$

[4.22]

Example 2: Steady Hydrodynamic Model (Hydrodynamic II)

$$[C_G]=\frac{1}{2}\begin{bmatrix} * & * & * & L_1 & 0 & 0 & 0 \\ * & * & 0 & 0 & 0 & 0 & \\ * & 0 & 0 & L_1+L_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \\ & 0 & 0 & L_2+L_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & * \\ & 0 & 0 & L_3 & 0 & * & * \\ 0 & 0 & 0 & 0 & * & * & * \end{bmatrix} \qquad [4.23]$$

$$[K_e]=\frac{1}{2}\begin{bmatrix} * & * & * & 0 & -1+\alpha & 0 & 1+\alpha \\ * & * & (-1+\alpha)\left(\frac{1}{T_1}-\frac{Q_1^2}{gA_1^3}\right) & (-1+\alpha)\left(\frac{2Q_1}{gA_1^2}\right) & (1-\alpha)\left(\frac{1}{T_2}-\frac{Q_2^2}{gA_2^3}\right) & (1-\alpha)\left(\frac{2Q_2}{gA_2^2}\right) & \\ * & 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha \\ (-1-\alpha)\left(\frac{1}{T_1}-\frac{Q_1^2}{gA_1^3}\right) & (-1-\alpha)\left(\frac{2Q_1}{gA_1^2}\right) & 2\alpha\left(\frac{1}{T_2}-\frac{Q_2^2}{gA_2^3}\right) & 2\alpha\left(\frac{2Q_2}{gA_2^2}\right) & (1-\alpha)\left(\frac{1}{T_3}-\frac{Q_3^2}{gA_3^3}\right) & (1-\alpha)\left(\frac{2Q_3}{gA_3^2}\right) & \\ & 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha \\ (-1-\alpha)\left(\frac{1}{T_2}-\frac{Q_2^2}{gA_2^3}\right) & (-1-\alpha)\left(\frac{2Q_2}{gA_2^2}\right) & 2\alpha\left(\frac{1}{T_3}-\frac{Q_3^2}{gA_3^3}\right) & 2\alpha\left(\frac{2Q_3}{gA_3^2}\right) & (1-\alpha)\left(\frac{1}{T_4}-\frac{Q_4^2}{gA_4^3}\right) & (1-\alpha)\left(\frac{2Q_4}{gA_4^2}\right) & * \\ & 0 & -1-\alpha & 0 & 1+\alpha & * & * \\ (-1-\alpha)\left(\frac{1}{T_3}-\frac{Q_3^2}{gA_3^3}\right) & (-1-\alpha)\left(\frac{2Q_3}{gA_3^2}\right) & (1+\alpha)\left(\frac{1}{T_4}-\frac{Q_4^2}{gA_4^3}\right) & (1+\alpha)\left(\frac{2Q_4}{gA_4^2}\right) & * & * & * \end{bmatrix}$$

[4.24]

Example 3: Zero-Inertia Model

$$[C_G]=\frac{1}{2}\begin{bmatrix} * & * & * & L_1 & 0 & 0 & 0 \\ * & * & 0 & 0 & 0 & 0 & \\ * & 0 & 0 & L_1+L_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \\ & 0 & 0 & L_2+L_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & * \\ & 0 & 0 & L_3 & 0 & * & * \\ 0 & 0 & 0 & 0 & * & * & * \end{bmatrix} \qquad [4.25]$$

$$[K_G]=\frac{1}{2}\begin{bmatrix} * & * & * & 0 & -1+\alpha & 0 & 1+\alpha \\ * & * & \frac{-1+\alpha}{T_1} & \frac{L_1 n^2 Q_1}{\rho_1 A_1^{\rho 2}} & \frac{1-\alpha}{T_2} & 0 & \\ * & 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha \\ \frac{-1-\alpha}{T_1} & 0 & \frac{2\alpha}{T_2} & (L_1+L_2)\frac{n^2 Q_2}{\rho_1 A_2^{\rho 2}} & \frac{1-\alpha}{T_3} & 0 & \\ & 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha \\ \frac{-1-\alpha}{T_2} & 0 & \frac{2\alpha}{T_3} & (L_2+L_3)\frac{n^2 Q_3}{\rho_1 A_3^{\rho 2}} & \frac{1-\alpha}{T_4} & 0 & * \\ & 0 & -1-\alpha & 0 & 1+\alpha & * & * \\ \frac{-1-\alpha}{T_3} & 0 & \frac{1+\alpha}{T_4} & \frac{L_3 n^2 Q_4}{\rho_1 A_4^{\rho 2}} & * & * & * \end{bmatrix} \qquad [4.26]$$

Example 4: Kinematic Wave Model

$$[C_G]=\frac{1}{2}\begin{bmatrix} * & * & * & L_1 & 0 & 0 & 0 \\ * & * & 0 & 0 & 0 & 0 & \\ * & 0 & 0 & L_1+L_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \\ & 0 & 0 & L_2+L_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & * \\ & 0 & 0 & L_3 & 0 & * & * \\ 0 & 0 & 0 & 0 & * & * & * \end{bmatrix} \qquad [4.27]$$

$$[K_G]=\frac{1}{2}\begin{bmatrix} * & * & * & 0 & -1+\alpha & 0 & 1+\alpha \\ * & * & \frac{1}{n}\rho_1 A_1^{\rho_2-2} S_{0_1} & -1 & 0 & 0 & \\ * & 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha \\ 0 & 0 & \frac{1}{n}\rho_1 A_2^{\rho_2-2} S_{0_2} & -1 & 0 & 0 & \\ & 0 & -1-\alpha & 0 & 2\alpha & 0 & 1-\alpha \\ 0 & 0 & \frac{1}{n}\rho_1 A_3^{\rho_2-2} S_{0_3} & -1 & 0 & 0 & * \\ & 0 & -1-\alpha & 0 & 1+\alpha & * & * \\ 0 & 0 & \frac{1}{n}\rho_1 A_4^{\rho_2-2} S_{0_4} & -1 & * & * & * \end{bmatrix} \qquad [4.28]$$

B. Finite Element Computer Model

A general one-dimensional surface irrigation computer model (FE-SURFDSGN) was developed based on the finite element Galerkin formulation of the complete and simplified hydrodynamic equations. The model was developed based on the methodology of Chapter III. It utilizes the finite element Galerkin formulation for performing the analysis of the surface irrigation problem using either linear or quadratic elements. The model was developed to run on any IBM-compatible microcomputer with a Random Access Memory (RAM) of 512 Kbytes or more and an MS-DOS version 2.00 or higher.

1. Model Components

FE-SURFDSGN was developed in a modular format. It has a main program that addresses a series of routines. These routines vary in size and complexity. Some of these routines are devoted solely to input and output while others deal with the setting up of the global systems of equations using the direct stiffness procedure of the finite element method and the numerical solution of the developed system of equations at each time step. The program allows the user to produce data files that include the data for both the advance and recession curves of flow in surface irrigation. FE-SURFDSGN has various levels of tabular output which is also controlled by the user. It has a companion graphics routine which, if selected, produces a graphical output of the advance and recession curves in addition to the plot of actual field measurements. FE-SURFDSGN was developed and compiled in Power Basic which is the product of Spectra Publishing Company. The size of the program is very small compared to the number of various functions and options that it has. The size of the listing of FE-SURFDSGN doesn't exceed 90 Kbytes which

also includes the listing of the graphics routine. The listing of both the FE-SURFDSGN program and the graphics routine are presented in Appendices A and B, respectively.

The model can simulate the various phases of flow in furrows and borders. These include the advance, ponding, depletion, and recession phases. This analysis is performed based on either the standard or nonstandard finite element Galerkin formulation and using either linear or quadratic elements. The analysis can be carried out based on any combination of the above options and using the finite element formulation of the complete hydrodynamic (hydrodynamic model I), steady hydrodynamic (hydrodynamic model II), zero-inertia, or kinematic wave model.

2. Data Input

There are three screens of data input to FE-SURFDSGN. The first input screen (Figure 14) has the following entries:

Irrigation Method: This input allows the user to select the irrigation method which could either be furrow (F) or border (B).

Method of Solution: This entry allows FE-SURFDSGN to perform the analysis based on the finite element formulation of the complete hydrodynamic (H), steady hydrodynamic (S), zero-inertia (Z), or kinematic wave model (K).

Type of Element: The user has to select the type of element which could either be linear (L) or quadratic (Q).

Level of Printing: The level of printing can be selected by entering an integer in the range of 0 to 3. By selecting 0, the model would only print a general summary of the simulation process. On the other hand,

extensive output is produced when the level of printing is selected at (3). This output would include a printout of the intermediate steps of computation which might amount to pages and pages of printout. This level of printing is only needed for debugging FE-SURFDSGN.

Output Device: The last entry on the first screen would allow the user to choose the output device which could be the screen ((S)), a temporary file SURFDSGN.OUT ((F)), or the printer ((P)).

The second screen (Figure 15) allows the user to select many specific parameters which affect the speed and accuracy of the solution. These parameters include the following:

Time Step: The time step (δt) should be entered in minutes. FE-SURFDSGN utilizes constant time steps to solve the time-dependent surface irrigation problem. However, the model can be modified to use variable time steps if the user sees an advantage in implementing such a change.

Maximum Number of Iterations: This entry represents the maximum number of iterations that the model is allowed to perform before convergence is reached at various time steps. If the maximum number of time steps has been reached and the solution did not converge to the desired accuracy that is specified by the user, the program proceeds to the analysis of the subsequent time step.

Allowable error: This error represents the maximum allowable error for the accumulated deviation between the results of the previous and current iteration at all nodes.

Time Weighting Coefficient: The time weighting coefficient (θ) is a number in the range of 0 to 1 (refer to Section B-7 of Chapter III).

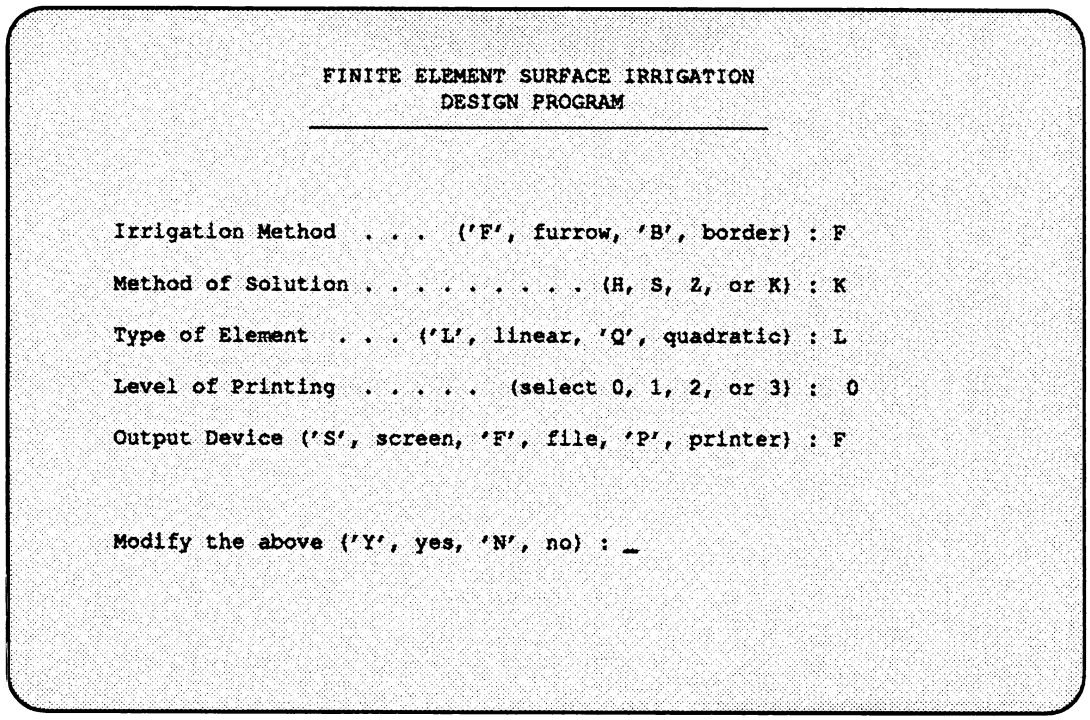


Figure 14. First screen of input data into FE-SURFDSGN.

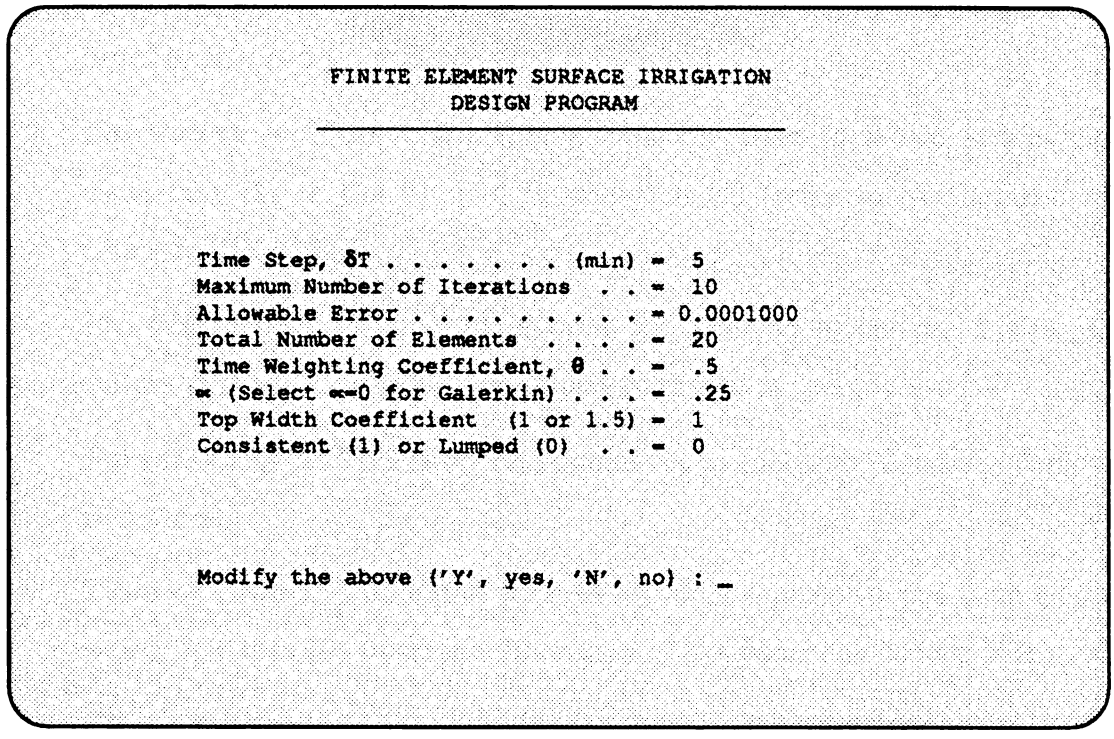


Figure 15. Second screen of input data into FE-SURFDSGN.

Space Weighting Coefficient: The space weighting coefficient (α) is a number in the range of *0* to *1* (refer to Section B-4 of Chapter III). Selecting α at *0* would result in the standard finite element Galerkin formulation. However, if $0 < \alpha \leq 1$ is selected, the nonstandard finite element Galerkin formulation results.

Top Width Coefficient: The coefficient *3/2* in Equation [3.254] is included sometimes in the coefficient σ_1. If this was the case, a value of *1* should be selected. Otherwise, a value of *1.5* should be keyed in as an input to this entry.

Consistent or Lumped: This allows the user to select either the consistent (1) or the lumped formulation (0) (refer to Section B-5 of Chapter III).

The last screen (Figure 16) allows the user to enter the hydraulic parameters of the furrow or border. These parameters include the following:

Furrow Length: This entry represents the length of the border or furrow in meters.

Time of Cutoff: The cutoff time in minutes represents the time when water flow into the upstream boundary of the furrow or border is turned off.

Inlet Flow Rate: The inlet flow rate into the furrow or the unit inlet flow rate into the border should be entered in liters/minute. This entry is currently considered as the average inflow rate at the upper boundary of the furrow or border. However, the model can be modified to handle an inflow hydrograph. This step could be accomplished through the definition of inlet flow rates at various time steps starting with time zero until water inflow is turned off.

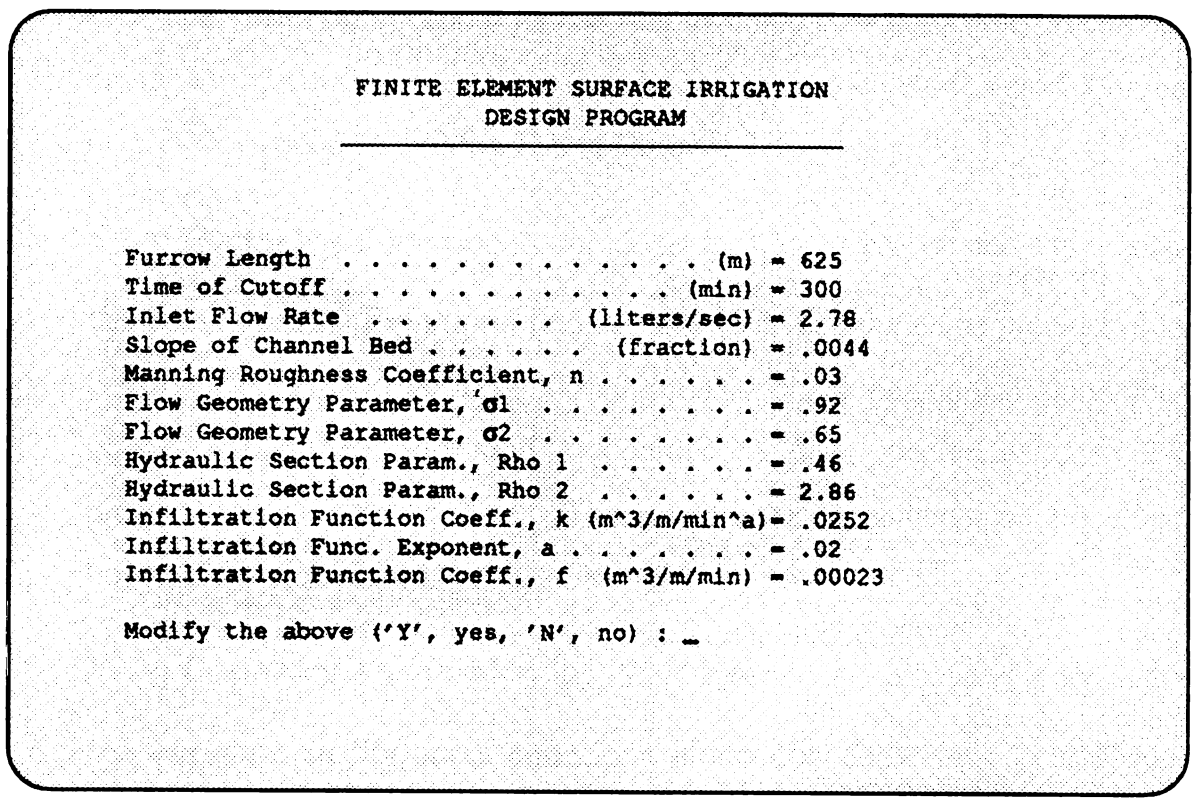


Figure 16. Third screen of input data into FE-SURFDSGN.

Slope of Channel Bed: This entry represents the slope of channel bed which should be entered as a fraction. The program currently defines the slope at all nodes to be the same. However, this could also be changed in FE-SURFDSGN to allow for the set up of an array that contains the slope of channel bed at various nodes.

Manning's Roughness Coefficient: This entry pertains to the roughness coefficient in the Manning's equation. The program currently uses the Manning equation as the sole uniform flow equation. However, various uniform flow equations may be incorporated into the model based on the discussion in Section B-8 of Chapter III.

Flow Geometry Parameter, σ_1: This hydraulic parameter represents the coefficient in Equations [3.252] and [3.254] that were presented in Section B-8 of Chapter III. These equations are power functions that correlate the flow rate and top width of flow, respectively, to the cross-sectional area of flow. The parameter σ_1 reduces to *1.5* when borders are considered.

Flow Geometry Parameter, σ_2: This hydraulic parameter represents the exponent in the same equations above (i.e, Equations [3.252] and [3.254]). Again, this parameter reduces to *1* when the user chooses to analyze borders.

Hydraulic Section Parameters, ρ_1 ***and*** ρ_2: These parameters are empirical fitting constants which are controlled by the hydraulic section of the furrow as shown in Equation [3.257] (refer to Section B-8 of Chapter III). These parameters reduce to *1* and *3.333*, respectively, when the case of borders is considered.

Infiltration Function Coefficient, k: This is the first coefficient in the Kostiakov-Lewis equation (see Equation [3.258] in Section B-8 of Chapter III). This input should be entered in $m^3/m/min^a$. The Kostiakov-Lewis function is currently the only available infiltration function in the developed computer model. However, other infiltration functions can be easily incorporated into FE-SURFDSGN.

Infiltration Function Exponent, a: This parameter represents the exponent in the Kostiakov-Lewis equation.

Infiltration Function Coefficient, f: This is the second coefficient in the Kostiakov-Lewis equation. This input should be entered in $m^3/m/min$.

3. Model Output

The developed finite element surface irrigation model displays a summary output which represent both the flow rate and the cross-sectional area of flow at all nodes at various time steps starting from initial conditions until the conclusion of the recession phase. Also, FE-SURFDSGN produces two data files which can subsequently be used to plot the simulated advance and recession curves of flow in surface irrigation. The program reports to the user the execution time of the computer after the completion of the simulation run for all phases of flow.

As was discussed earlier, FE-SURFDSGN has a companion graphics routine which produces a graphical output of the advance and recession curves in addition to the plot of actual field measurements for those curves. The utility of the graphics routine of FE-SURFDSGN is demonstrated in the following section where comparisons between simulated model data and actual field measurements are made.

FE-SURFDSGN has various levels of tabular output which is also controlled by the user. The level of printing can be selected by entering an integer in the range of 0 to 3. By selecting 0, the model would only print a general summary of the simulation process. However, extensive outputs are possible by selecting numbers in the range of 1 to 3 (refer to the Data Input section). The level of data output becomes more and more extensive when numbers closer to the upper range are selected. The latter outputs would include printouts of the intermediate steps of computation which amount to pages and pages of printout.

C. Results and Comparison

A computer model (FE-SURFDSGN) was developed based on the finite element Galerkin formulation of the complete and simplified forms of the hydrodynamic model. The model was originally developed to simulate the advance phase of water in border and furrow irrigation systems. The model was then extended to simulate the ponding, depletion, and recession phases of flow conditions in border and furrow irrigation systems. This was accomplished through incorporating the proper boundary conditions that correspond to these phases of flow into the finite element Galerkin formulation as discussed in Section A-2 of this chapter.

This section includes a summary of the comparisons that were made between the simulated model data and actual field measurements. A summary of the actual field measurements will be presented first, together with the input parameters that were utilized in running FE-SURFDSGN as well as the sources of these data. Then, plots of actual field data and predicted

data using the developed computer model will be presented. Finally, a sensitivity analysis will be presented to show the model's sensitivity to various input parameters.

1. Actual Field Data

In order to validate any surface irrigation model, actual field measurements are a necessity. The input data used for running FE-SURFDSGN which was based on the Galerkin formulation of the complete and simplified forms of the hydrodynamic model are presented in Table 5. These data were reported by Elliott et al. (1982b). The data were originally collected from furrow irrigation evaluations at three Colorado locations during the Summer of 1979 by Colorado State University researchers (Elliott, 1980). The three sites belonged to three farms that were privately owned. Walker and Skogerboe (1987) reported that this study involved six furrows (three groups of two furrows each) at each site during the 1979 irrigation season. The reader is referred to the publications by Elliott (1980) and Elliott et al. (1982b) for a detailed description of the Colorado study. Additional input data for model testing were used from four different Utah and Idaho tests. These data were collected by the researchers in the Department of Agricultural and Irrigation Engineering at Utah State University from two locations in Utah and Idaho. These data are described in Table 6 and were taken from Walker and Humpherys (1983) and Walker and Skogerboe (1987).

The observed advance data for the Colorado study were taken from the journal article by Elliott et al. (1982b) where those of the Utah and Idaho studies were taken from the text by Walker and Skogerboe (1987). The observed recession data for the Colorado study data were taken from the research study by Oweis (1983).

Table 5. General information and furrow evaluation data for the Colorado study sites that were used in model testing (source: Elliott et al., 1982b).

Parameter	Benson Farm		Matchett Farm		Printz Farm	
Soil Type	Clay loam		Loam to clay loam		Loamy sand	
Crop	Corn		Corn		Corn	
Furrow Length (m)	625		425		350	
Spacing Between Wetted Furrows (m)	1.524		0.762		1.524	
Irrigation Event, Group #, Furrow #	1, 1, 5	5, 2, 1	2, 3, 5	1, 4, 5	8, 2, 3	1, 1, 1
Time of Cutoff (min)	690	619	1364	1478.5	171	248
Inlet Flow, Q_0 (lps)	2.78	1.17	0.92	0.85	2.77	4.81
Slope, S_0 (m/m)	0.0044	0.0044	0.0095	0.0092	0.0025	0.0023
Manning's Roughness, n	.03	.02	.02	.03	.02	.03
Furrow Geometry Parameter, σ_1	0.92	0.72	2.18	.87	1.13	1.78
Furrow Geometry Parameter, σ_2	0.65	0.64	0.79	0.56	0.75	0.72
Hydraulic Section Parameter, ρ_1	0.46	0.34	1.35	0.30	0.73	0.92
Hydraulic Section Parameter, ρ_2	2.86	2.84	3.00	2.73	2.98	2.91
Infiltration Function Coefficient, k ($m^3/m/min^a$)	0.0252	0.0173	0.0033	0.0011	0.0161	0.0078
Infiltration Function Exponent, a	0.02	0.01	0.40	0.48	0.02	0.40
Infiltration Function Coefficient, f_0 ($m^3/m/min$)	0.00023	0.00008	0.00003	0.00003	0.00040	0.0014

Table 6. Additional data for model testing (source: Walker and Skogerboe, 1987).

Parameter	Flowell Nonwheel Furrow	Flowell Wheel Furrow	Kimberly Nonwheel Furrow	Kimberly Wheel Furrow
Soil Type	Sandy loam	Sandy loam	Silty-clay loam	silty-clay loam
Furrow Length (m)	250	360	360	360
Time of Cutoff (min)	350	400	400	200
Inlet Flow, Q_0 (lps)	2.0	2.0	0.8	1.5
Slope, S_0 (m/m)	0.008	0.008	0.0104	0.0104
Manning's Roughness, n	0.04	0.04	0.04	0.04
Furrow Geometry Parameter, σ_1	0.782	0.782	0.962	0.962
Furrow Geometry Parameter, σ_2	0.536	0.536	0.6046	0.6046
Hydraulic Section Parameter, ρ_1	0.3269	0.3269	0.6644	0.6644
Hydraulic Section Parameter, ρ_2	2.734	2.734	2.734	2.734
Infiltration Function Coefficient, k ($m^3/m/min^a$)	0.00217	0.00280	0.00701	0.00884
Infiltration Function Exponent, a	0.673	0.534	0.533	0.212
Infiltration Function Coefficient, f_0 ($m^3/m/min$)	0.00022	0.00022	0.00017	0.00017

2. Model Runs

FE-SURFDSGN was developed as a general computer model for simulating the flow conditions in borders and furrows based on the methodology that was discussed in the previous chapter. The various models that are available include the hydrodynamic and kinematic-wave models as well as the zero-inertia model. Currently, only the kinematic wave finite element analysis is fully operational in the present version of FE-SURFDSGN. There are still some problems in predicting the rate of advance in both the hydrodynamic and zero-inertia models. More work is being done to complete the development of these options in the computer model so that the general finite element development is operational for the complete and simplified forms of the hydrodynamic equations.

In order to demonstrate the effectiveness of the finite element formulation in the numerical solution of the hydrodynamic equations, various simulation runs were conducted using the kinematic-wave model. Comparing model results to actual field measurements demonstrate the effectiveness of the presented methodology and the utility of the nonstandard Galerkin formulation of the complete and simplified forms of the hydrodynamic equations as applied to the analysis of furrow and border irrigation systems.

The finite element model generated an advance curve for each column of the input data in Tables 4 and 5. These curves are shown in Figures 17 through 25 together with the plot of the measured advance data as reported by Elliott et al. (1982b) and Walker and Skogerboe (1987). The parameters α and θ were selected at *0.25* and *0.5*, respectively, for all these runs. The selection of these parameters was based on the sensitivity analysis that will be presented in the following subsection. The time step, Δt, was selected at *5* minutes for all these runs.

In general, it is obvious from these runs (Figures 17 through 25) that model predictions are very reasonable. This is apparent where the simulated rate of advance is very consistent with actual field measurements in almost all cases. The model predictions are good indicators of the effectiveness of the presented methodology in this research study.

As to the speed of the developed model, the latter prediction runs took approximately *50* seconds on average on a 386 IBM compatible machine. However, the average execution time could be cut at least in half if the length of the time step, Δt, is doubled. The accuracy of the simulated model results will not be affected even if Δt was doubled or quadrupled as will be demonstrated again in the subsequent subsection on sensitivity analysis.

FE-SURFDSGN was also utilized in simulating the complete irrigation cycle for the furrow tests from the Colorado study that were presented in Table 4. The irrigation cycle includes the advance, ponding, depletion, and recession phases of flow. The results of these simulation runs are presented in Figures 26 through 31. The plot of the measured advance and recession data were taken from Elliott et al. (1982b) and Oweis (1983), respectively. These runs were conducted using a Δt of *10* minutes. The parameters α and θ were selected at *0.25* and *0.5*, respectively.

Again, it is apparent from Figures 26 through 31 that the simulated rates of advance and recession are very consistent with actual field measurements. This shows that FE-SURFDSGN can be used as an effective tool for simulating the irrigation cycle in furrows and borders including the advance, ponding, depletion, and recession phases of flow.

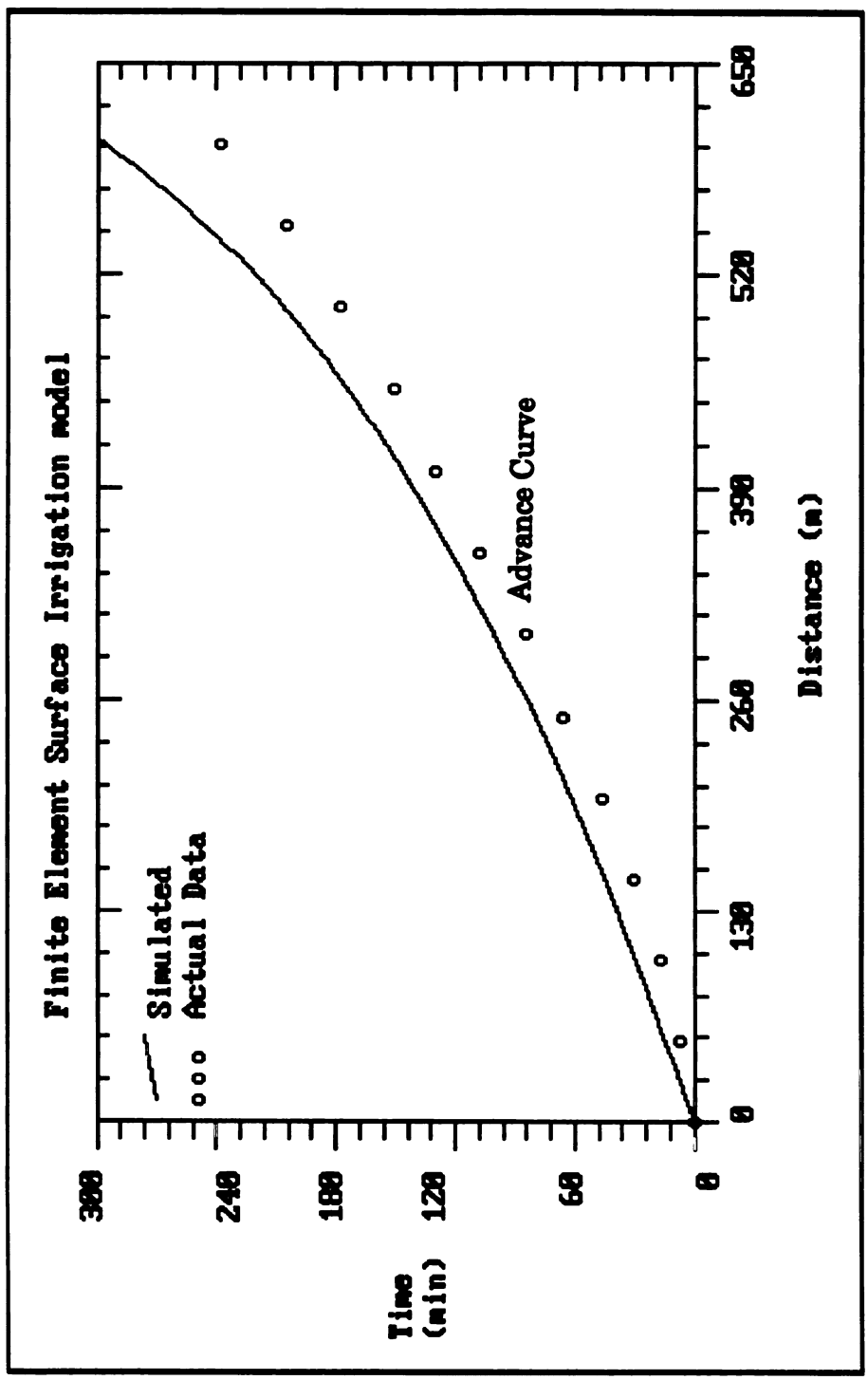


Figure 17. Predicted and observed advance for irrigation 1, group 1, furrow 5 on Benson farm using the kinematic-wave finite element model (source of actual data: Elliott et al., 1982b).

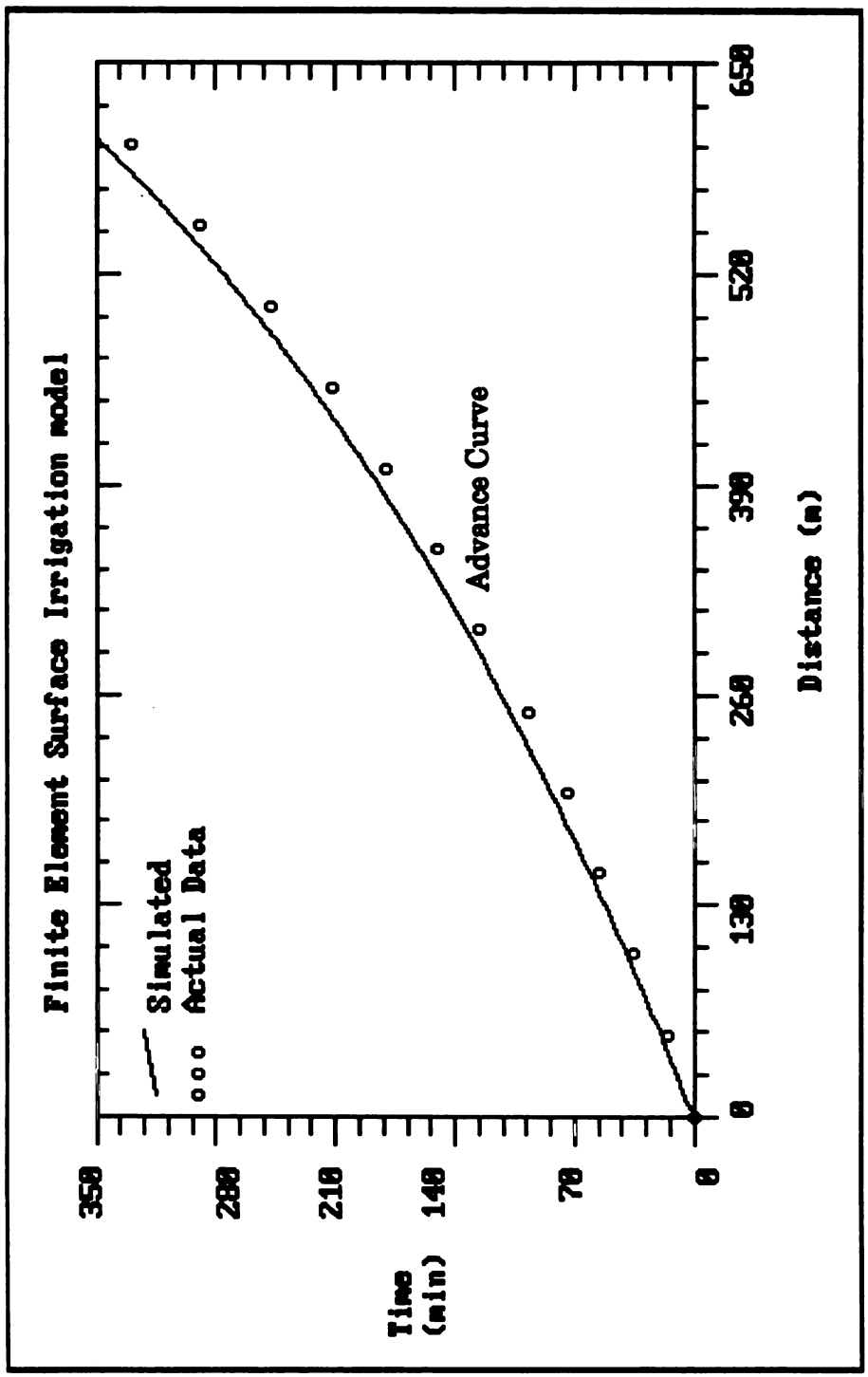


Figure 18. Predicted and observed advance for irrigation 5, group 2, furrow 1 on Benson farm using the kinematic-wave finite element model (source of actual data: Elliott et al., 1982b).

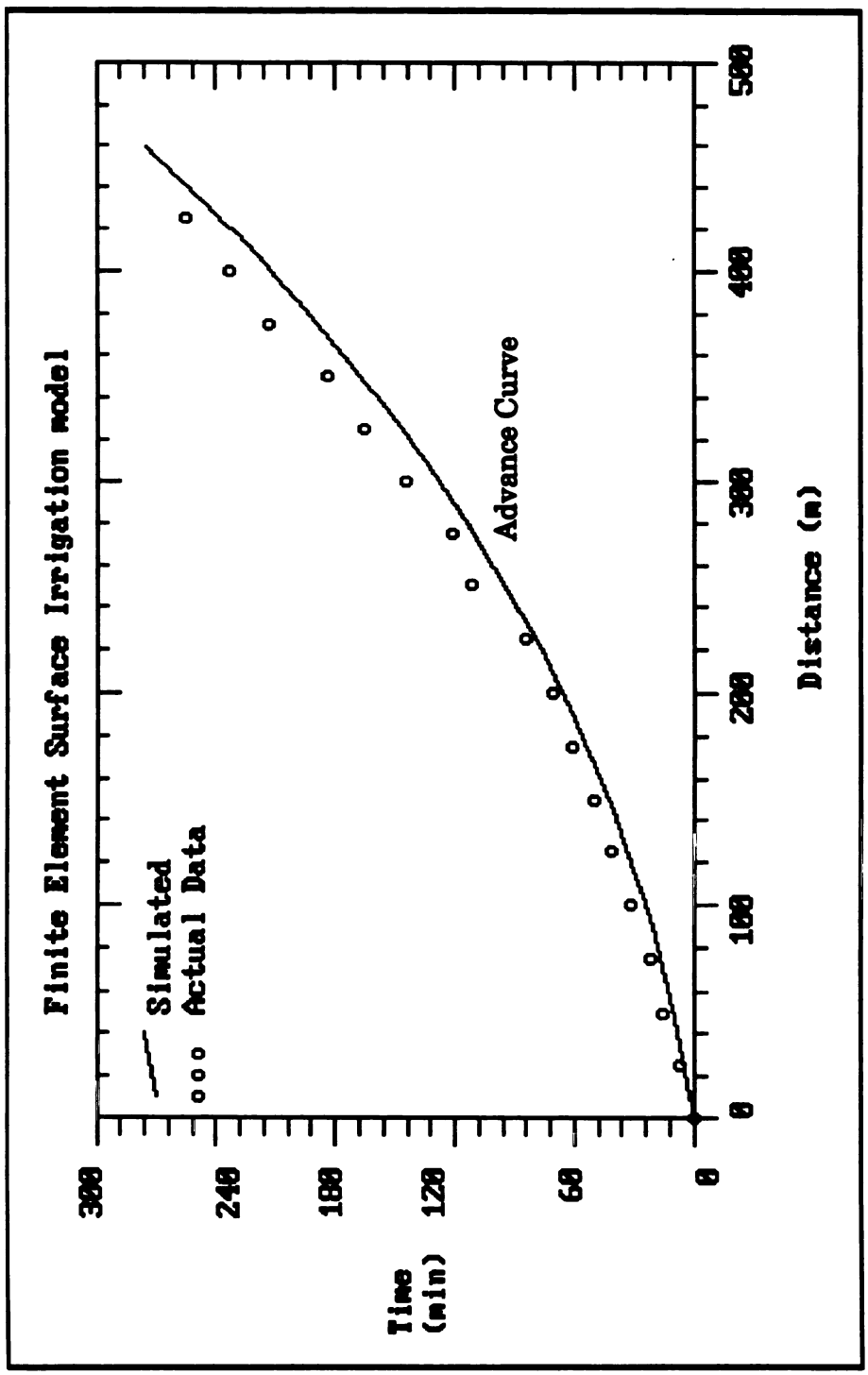


Figure 19. Predicted and observed advance for irrigation 2, group 3, furrow 5 on Matchett farm using the kinematic-wave finite element model (source of actual data: Elliott et al., 1982b).

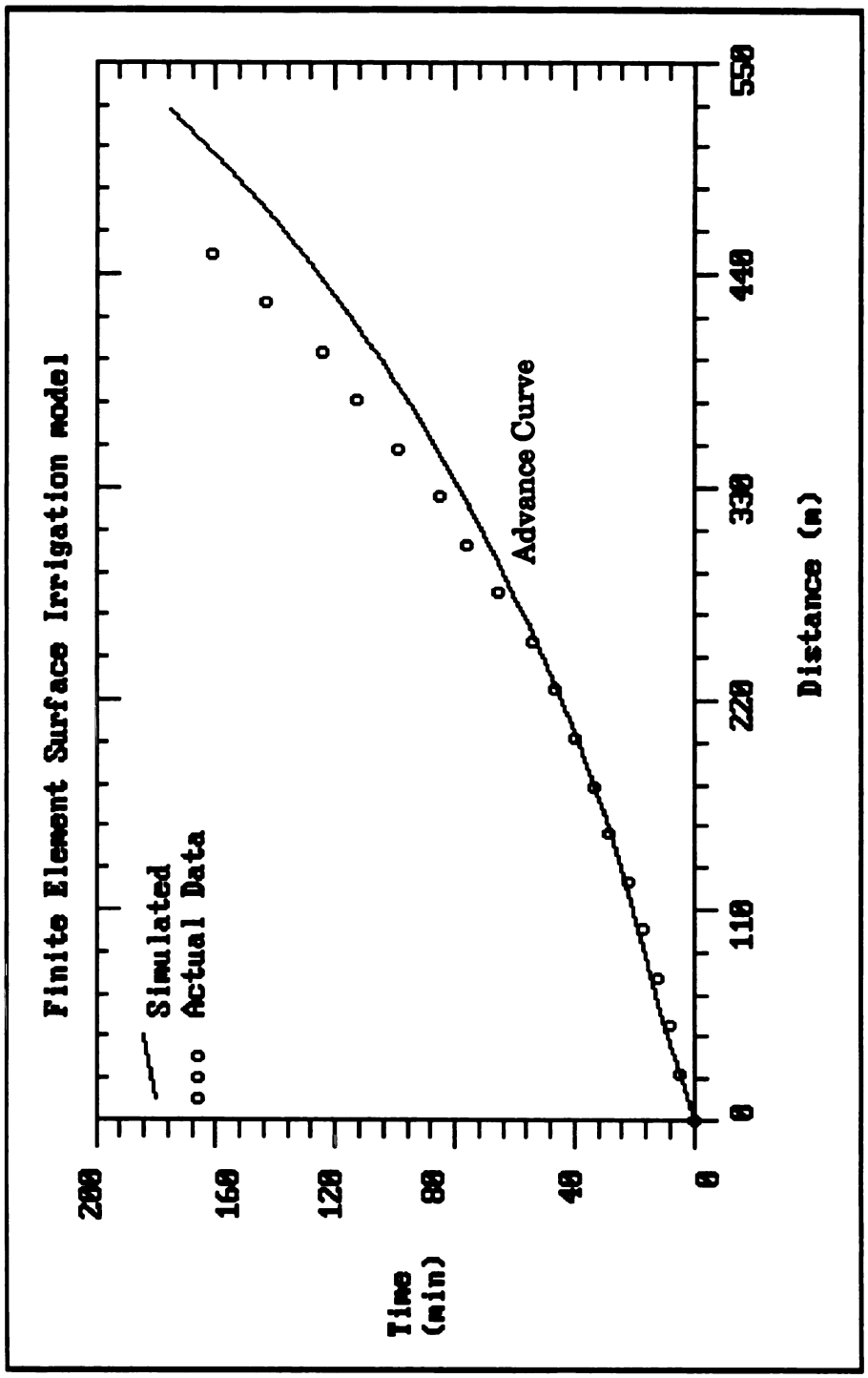


Figure 20. Predicted and observed advance for irrigation 1, group 4, furrow 5 on Matchett farm using the kinematic-wave finite element model (source of actual data: Elliott et al., 1982b).

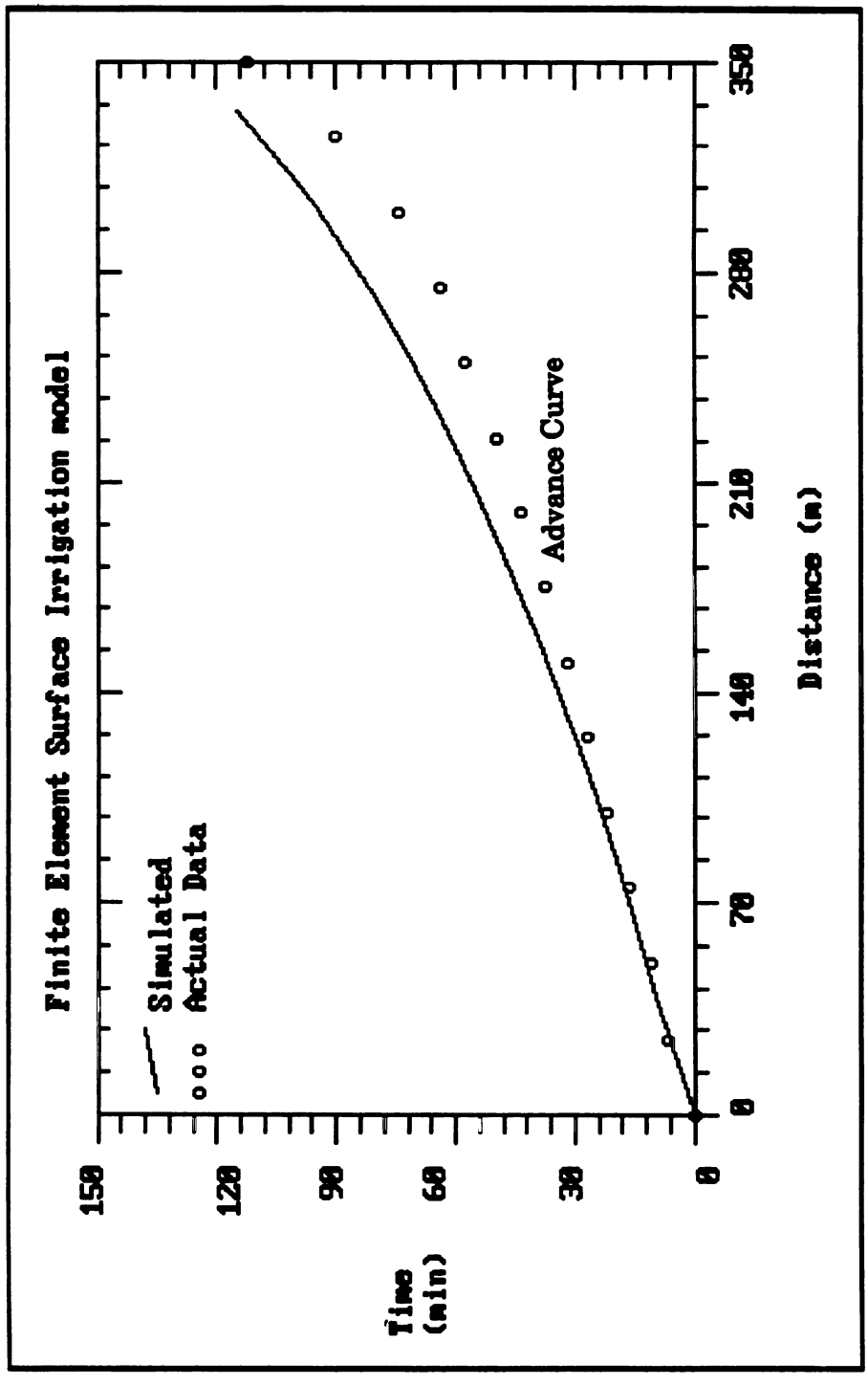


Figure 21. Predicted and observed advance for irrigation 8, group 2, furrow 3 on Printz farm using the kinematic-wave finite element model (source of actual data: Elliott et al., 1982b).

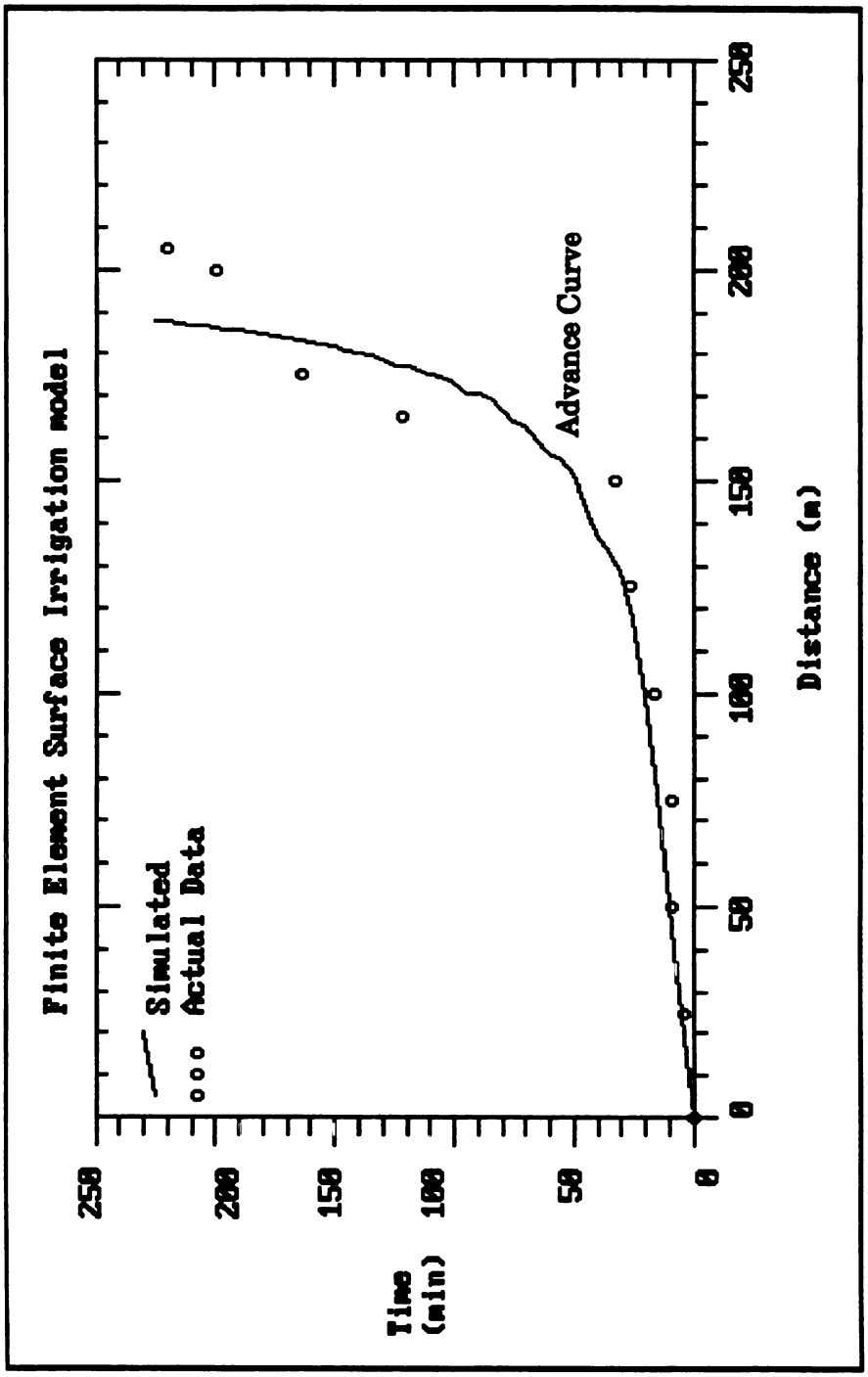


Figure 22. Predicted and observed advance for irrigation 1, group 1, furrow 1 on Printz farm using the kinematic-wave finite element model (source of actual data: Walker and Skogerboe, 1987).

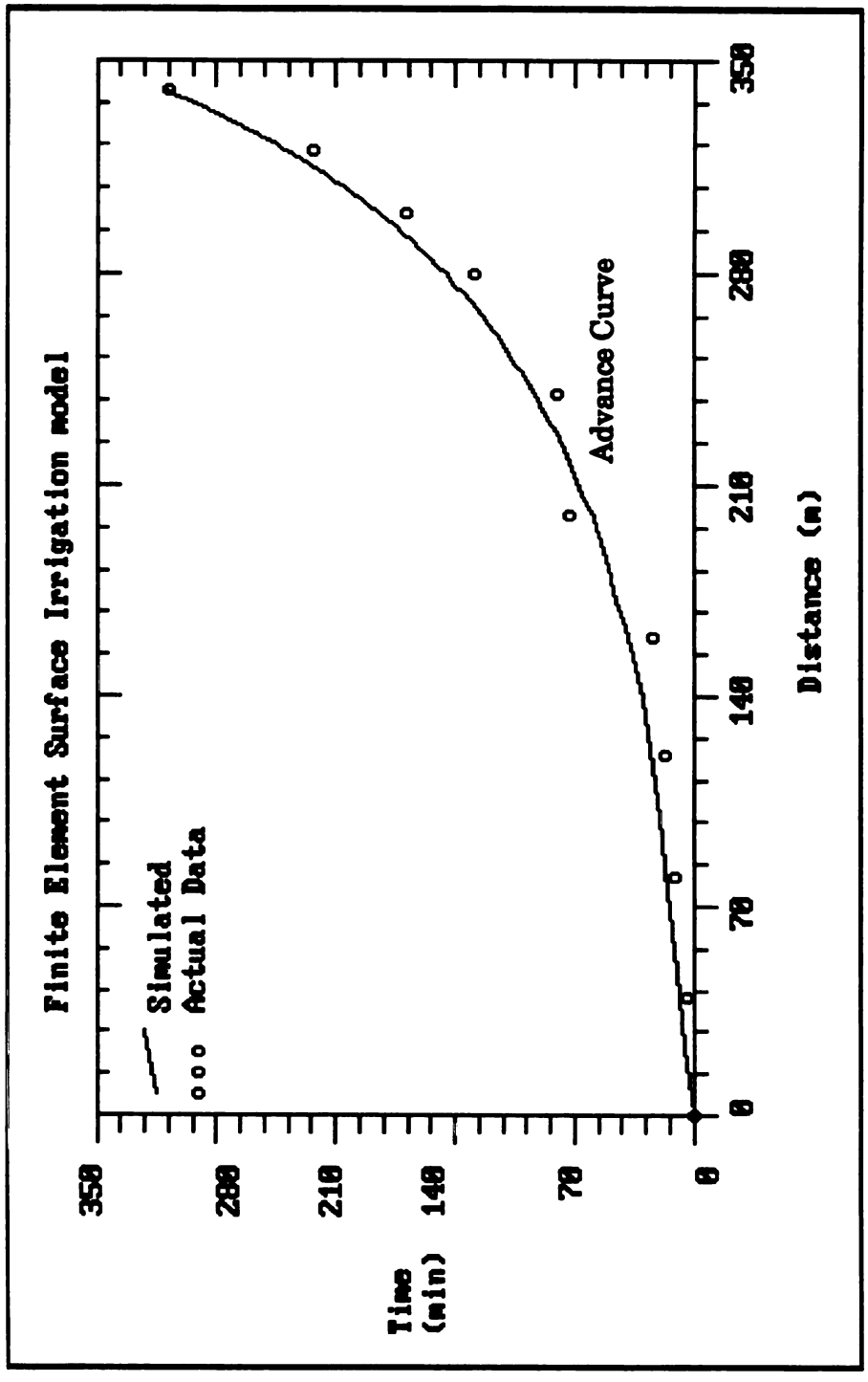


Figure 23. Predicted and observed advance for irrigation on Flowell continuous wheel furrow using the kinematic-wave finite element model (source of actual data: Walker and Skogerboe, 1987).

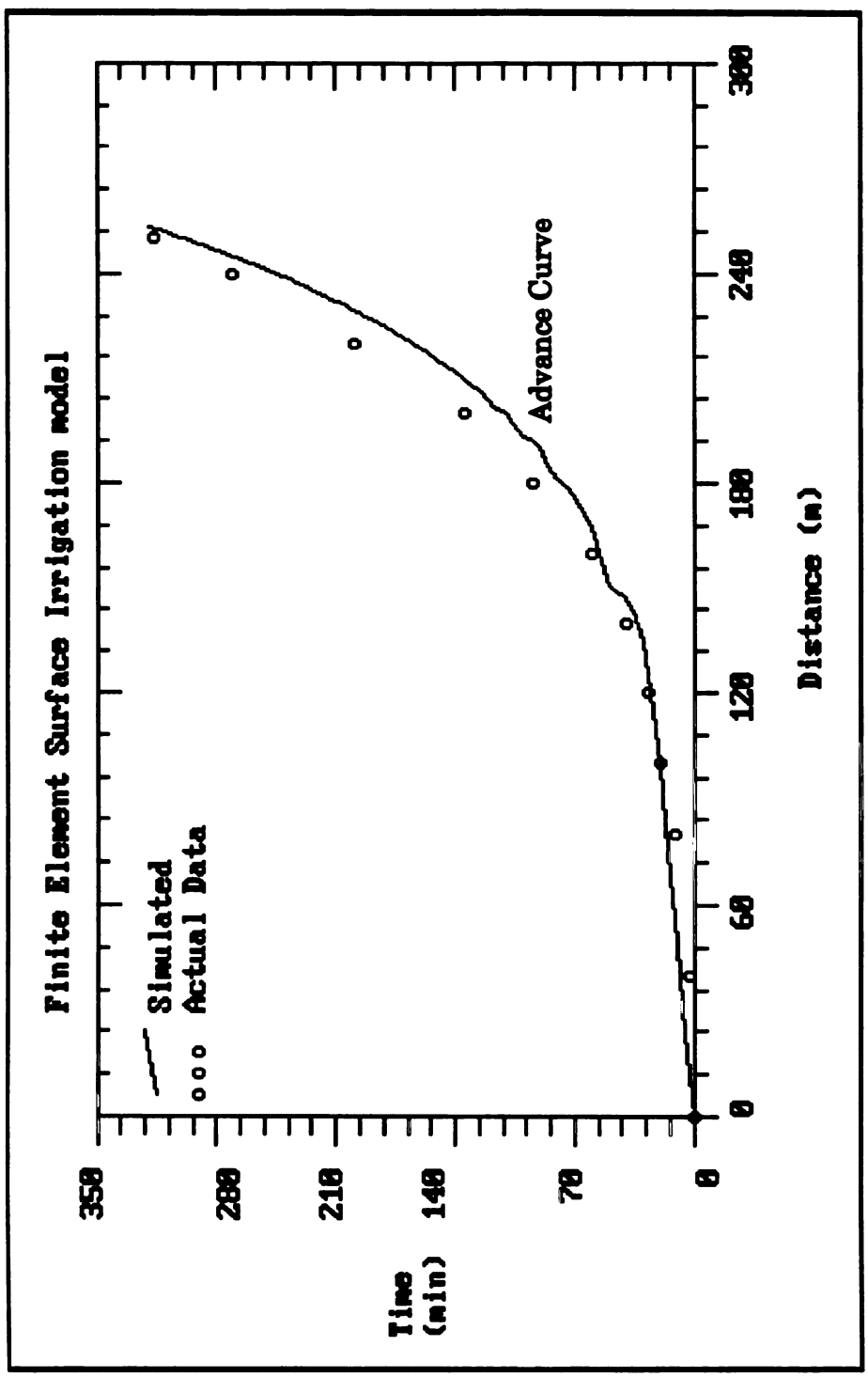


Figure 24. Predicted and observed advance for irrigation on Flowell continuous nonwheel furrow using the kinematic-wave finite element model (source of actual data: Walker and Skogerboe, 1987).

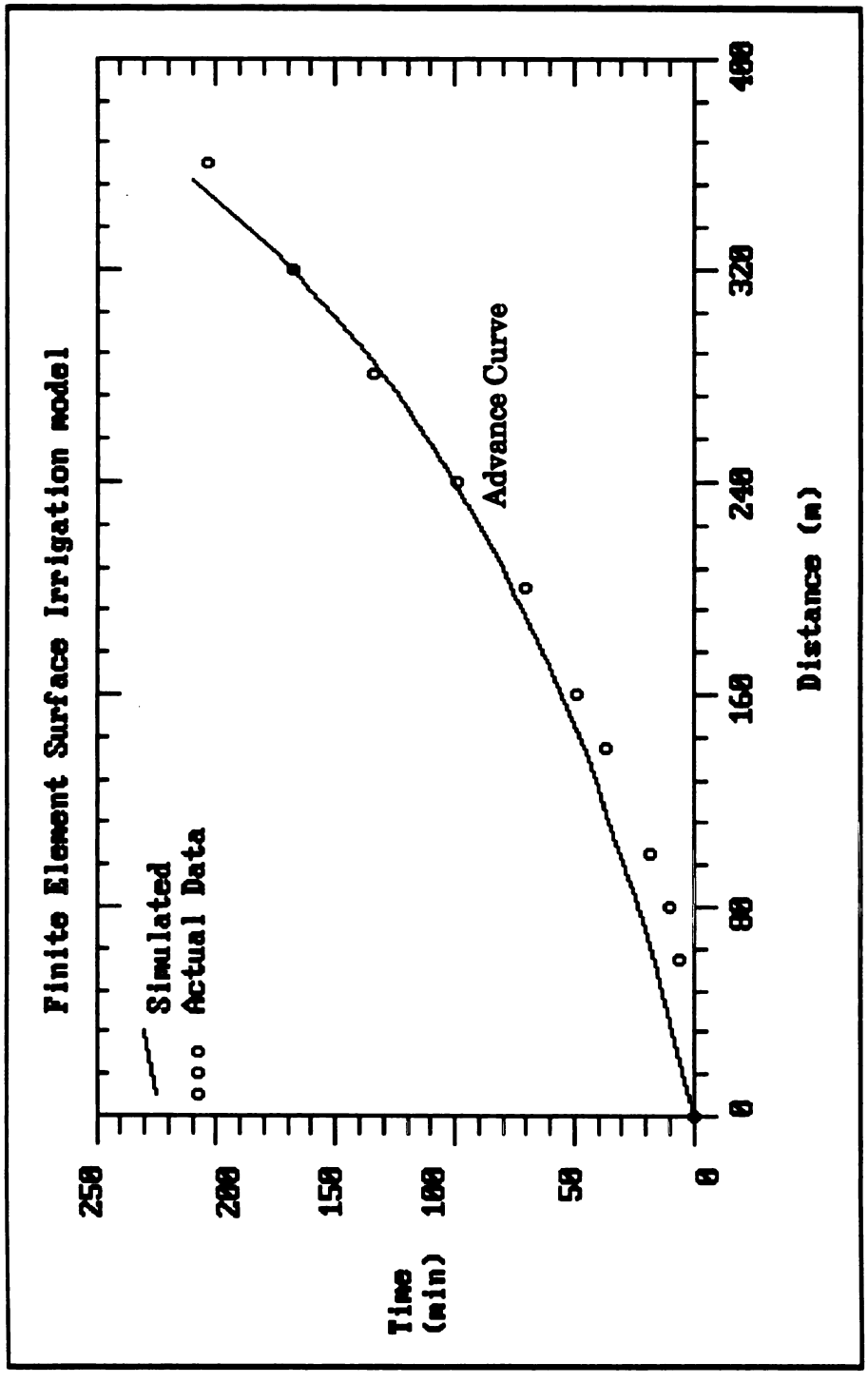


Figure 25. Predicted and observed advance for irrigation on Kimberly continuous flow wheel furrow using the kinematic-wave finite element model (source of actual data: Walker and Skogerboe, 1987).

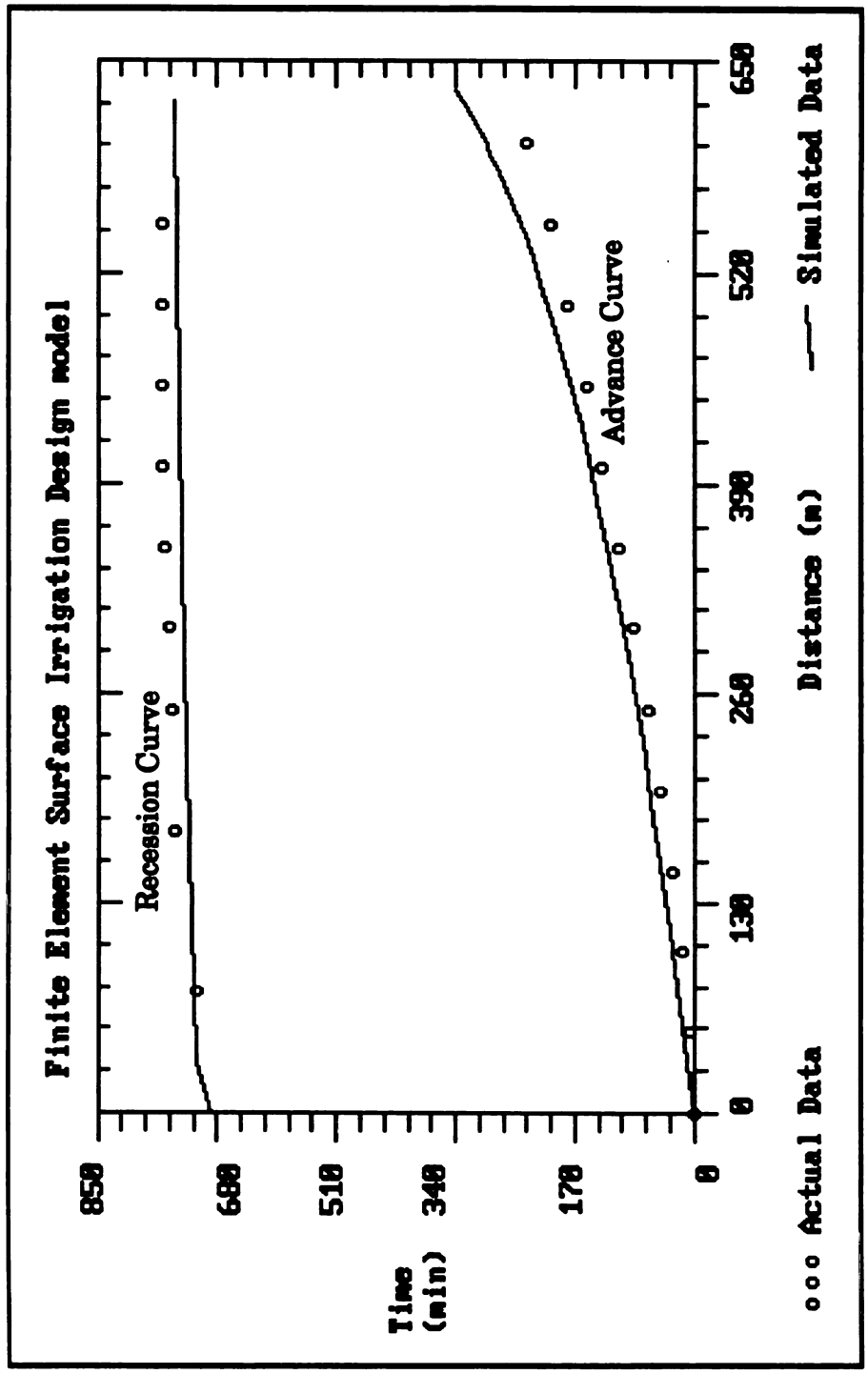


Figure 26. Predicted and observed irrigation data for irrigation 1, group 1, furrow 5 on Benson farm using the kinematic-wave finite element model (source of actual data: Elliott et al., 1982b, advance data; Oweis, 1983, recession data).

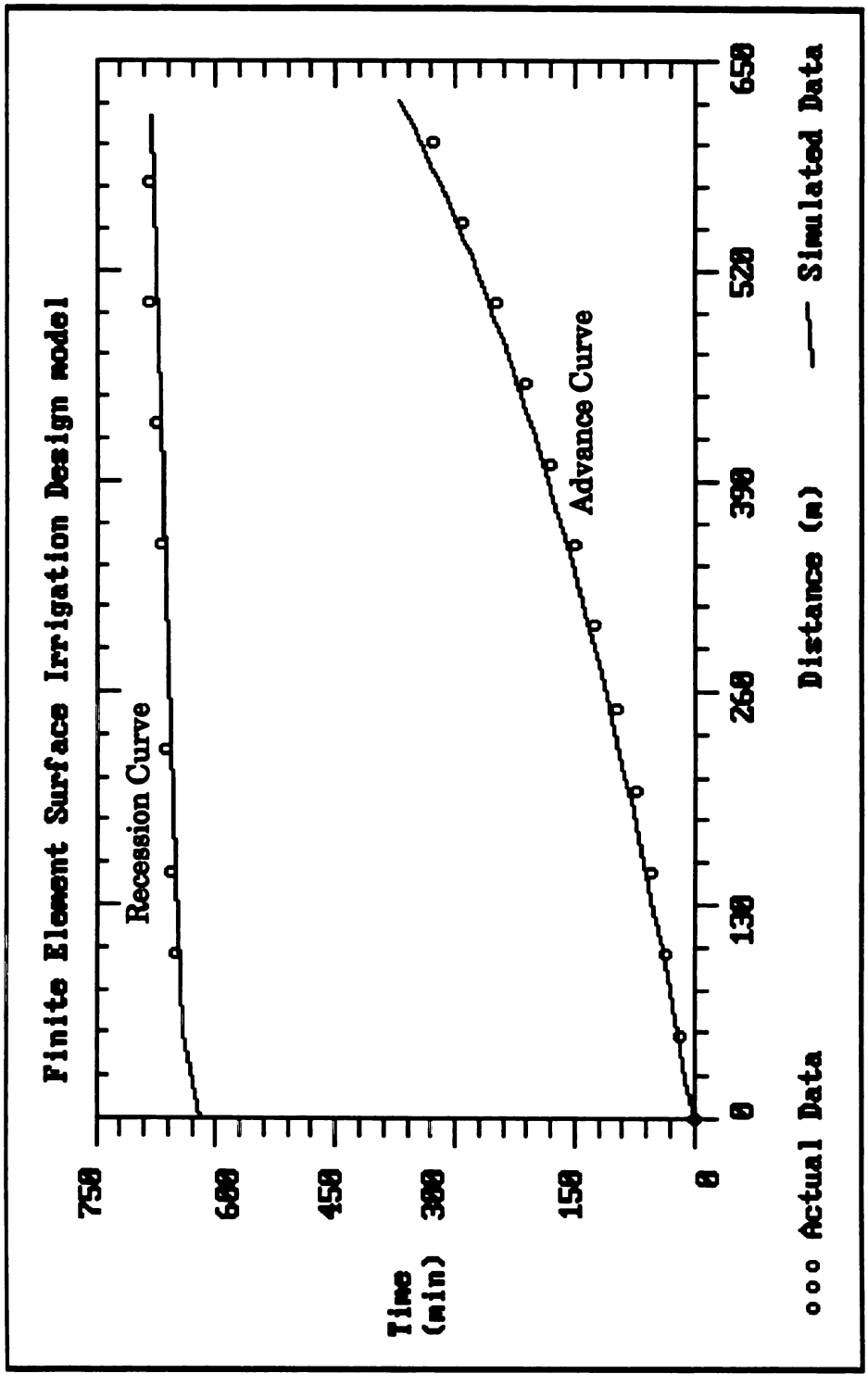


Figure 27. Predicted and observed irrigation data for irrigation 5, group 2, furrow 1 on Benson farm using the kinematic-wave finite element model (source of actual data: Elliott et al., 1982b, advance data; Oweis, 1983, recession data).

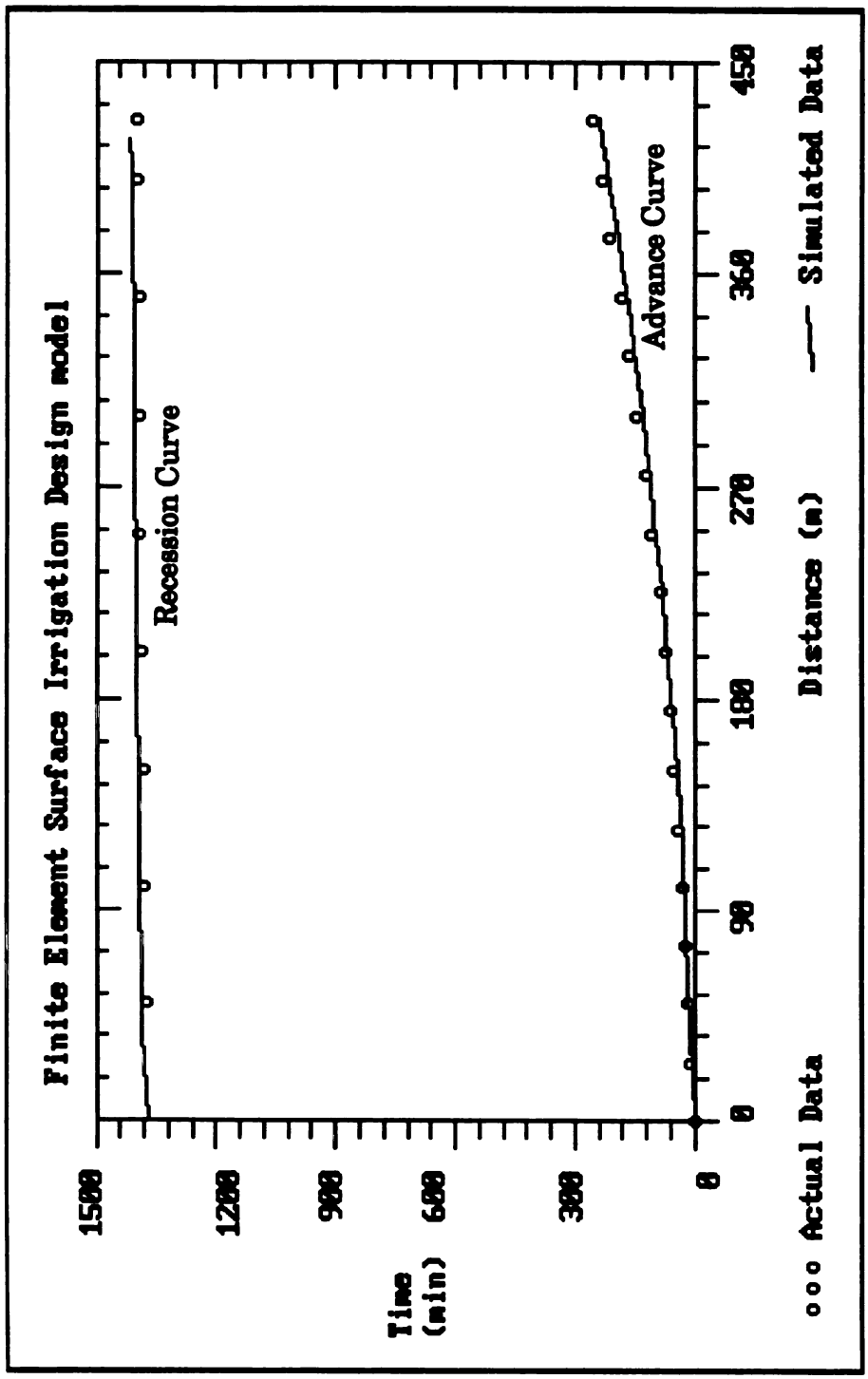


Figure 28. Predicted and observed irrigation data for irrigation 2, group 3, furrow 5 on Matchett farm using the kinematic-wave finite element model (source of actual data: Elliott et al., 1982b, advance data; Oweis, 1983, recession data).

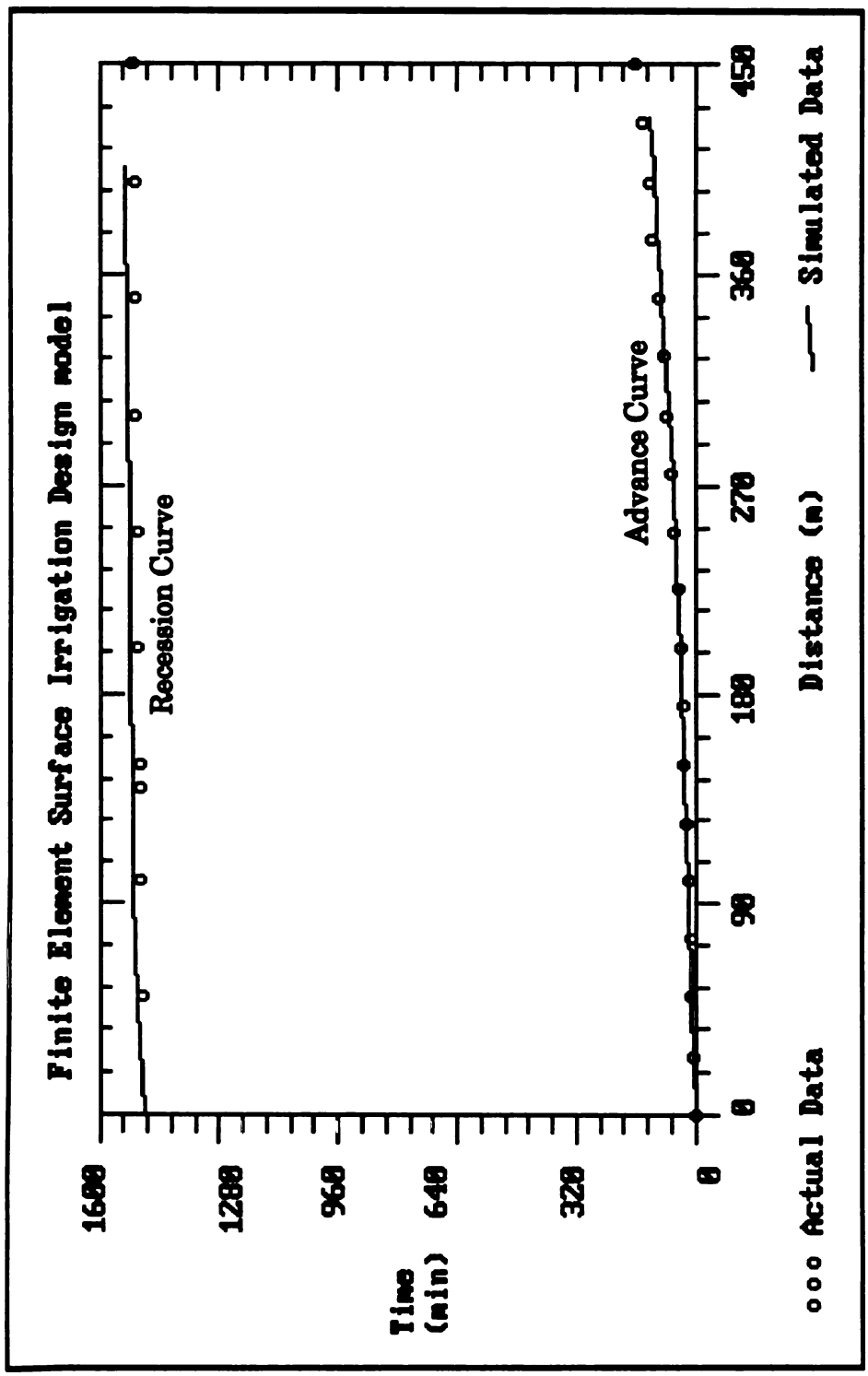


Figure 29. Predicted and observed irrigation data for irrigation 1, group 4, furrow 5 on Matchett farm using the kinematic-wave finite element model (source of actual data: Elliott et al., 1982b, advance data; Oweis, 1983, recession data).

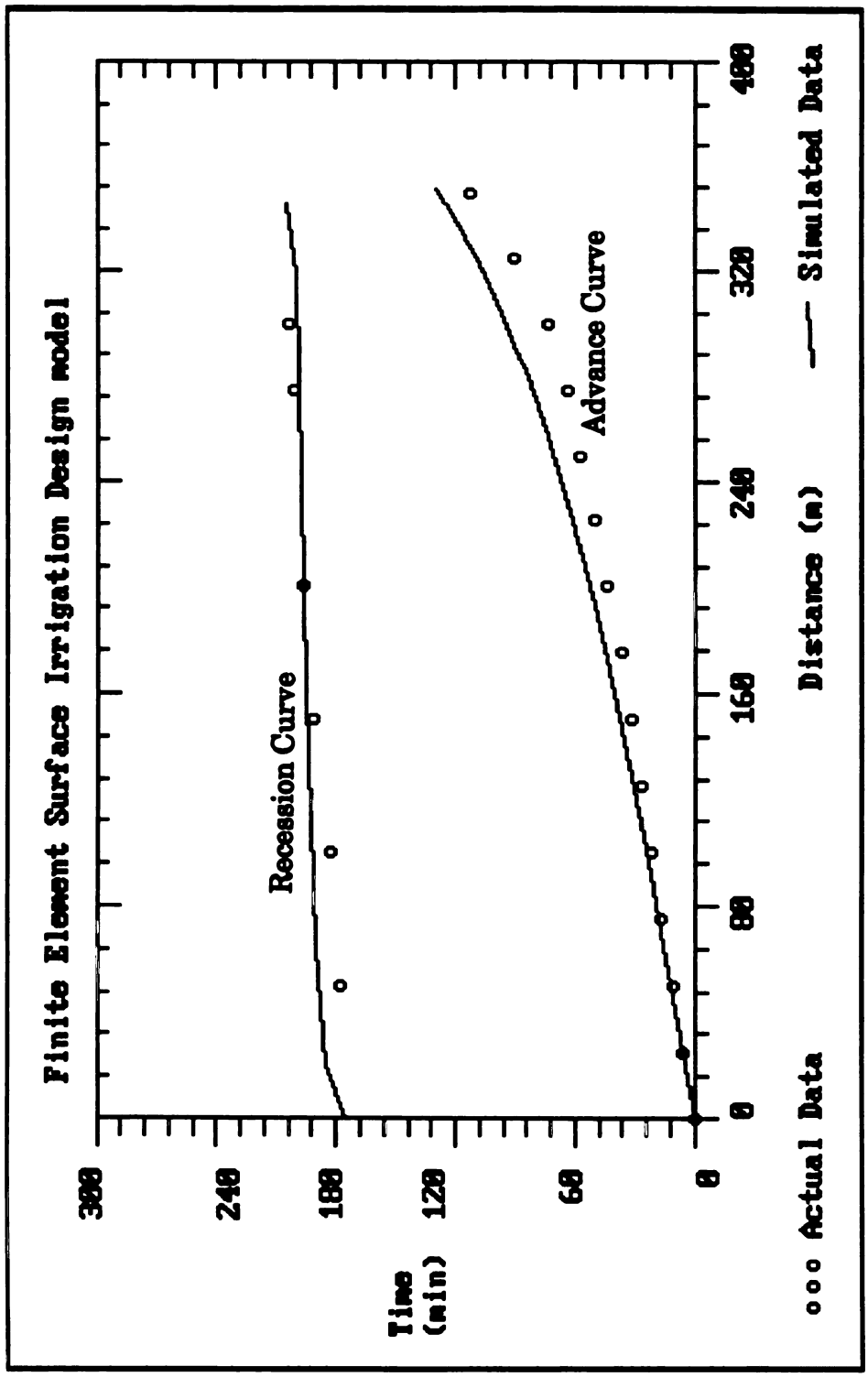


Figure 30. Predicted and observed irrigation data for irrigation 8, group 2, furrow 3 on Printz farm using the kinematic-wave finite element model (source of actual data: Elliott et al., 1982b, advance data; Oweis, 1983, recession data).

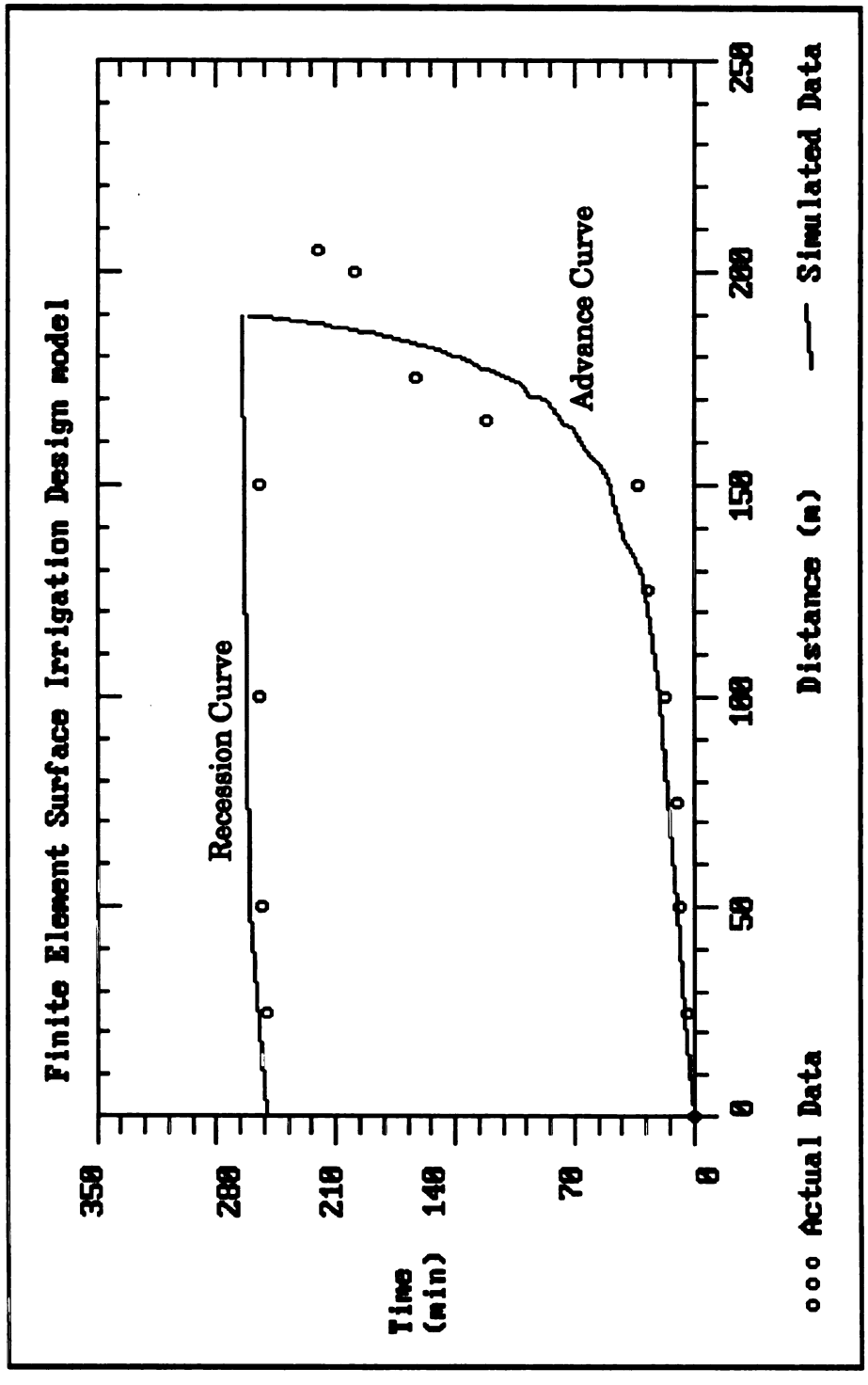


Figure 31. Predicted and observed irrigation data for irrigation 1, group 1, furrow 1 on Printz farm using the kinematic-wave finite element model (source of actual data: Elliott et al., 1982b, advance data; Oweis, 1983, recession data).

3. Sensitivity Analysis

FE-SURFDSGN was utilized to carry out simulation runs for investigating the sensitivity of the model to various physical input parameters. The sensitivity of the model to the selected time step (Δt), time weighting coefficient (θ), and the parameter α (a parameter that dictates if the standard or nonstandard Galerkin finite element formulations are utilized) were also studied. The linear finite element formulation of the kinematic-wave model was used for this purpose. The expected trends should be representative of the complete and simplified forms of the hydrodynamic equations. This is especially true when considering the sensitivity of the model to various physical input parameters.

The first phase was to study the effect of the selected time step, Δt, on model predictions as related to accuracy and numerical oscillations. The various time steps selected in this sensitivity analysis were *2.5*, *5*, *10*, *15*, *20*, *30*, *40*, and *50* minutes. On the other hand, the selected physical input data were those that correspond to irrigation 5, group 2, and furrow 1 of the Benson farm (refer to Table 5). The results that reveal the sensitivity of the model to the selected time steps are presented in Table 7. These results indicate that selecting a Δt as high as *20* or *30* minutes would produce results that are reasonably accurate. This would certainly shorten the execution time which is as low as *5* seconds on a 386 IBM compatible microcomputer.

The sensitivity of the model to the selection of various values of the weighting coefficient θ and α was studied next. The results from these runs are presented in Table 8 and 9 for θ and α, respectively. A conservative time step was used in all these runs. The Δt was chosen at *10* minutes even though values as high as *20* or *30* minutes could have been selected.

Table 7. Rate of advance predictions for several time steps using the kinematic wave model of FE-SURFDSGN.

Advance Time	Time Step (*min*)							
(*min*)	2.5	5	10	15	20	30	40	50
10	26.76	25.91	23.93					
20	52.61	52.66	50.92		46.40			
30	77.83	77.41	77.72	75.21		67.65		
40	102.40	102.29	102.44		98.82		87.80	
50	126.34	126.04	125.57					106.94
60	149.65	149.49	148.66	149.77	149.65	144.19		
70	172.37	172.12	171.87					
80	194.49	194.31	194.18		194.80		187.28	
90	216.03	215.81	215.42	214.53		216.67		
100	237.00	236.81	236.31		235.41			228.27
110	257.42	257.22	256.99					
120	277.29	277.09	276.91	276.94	275.36	278.44	279.24	
130	296.62	296.43	296.07					
140	315.43	315.23	314.90		314.89			
150	333.72	333.54	333.35	332.76		331.56		337.75
160	351.50	351.31	351.11		351.32		354.17	
170	368.79	368.62	368.30					
180	385.59	385.41	385.19	385.15	384.61	383.00		
190	401.91	401.75	401.60					
200	417.77	417.61	417.39		416.80		414.85	422.66
210	433.17	433.01	432.79	432.62		433.27		
220	448.12	447.98	447.85		447.94			
230	462.64	462.49	462.37					
240	476.72	476.60	476.41	476.27	476.41	477.15	473.66	
250	490.39	490.26	490.14					487.21
260	503.65	503.53	503.47		502.91			
270	516.50	516.40	516.28	516.28		515.31		
280	528.97	528.87	528.77		528.67		530.39	
290	541.05	540.97	540.94					
300	552.76	552.68	552.64	552.41	552.81	552.14		549.87
310	564.11	564.04	563.97					
320	575.10	575.04	575.05		574.81		576.68	
# of Elements	128	64	32	22	16	11	8	7
Exec. Time (*sec*)	415.8	104.7	28.1	15.3	8.7	4.4	2.5	2.0

Table 8. Rate of advance predictions for several time weighting coefficients using the kinematic wave model of FE-SURFDSGN.

Advance Time	Time Weighting Coefficient, θ						
(*min*)	0.40	0.45	0.50	0.55	0.60	0.65	0.70
10	21.91	22.99	23.93	24.76	25.50	26.16	26.75
20	51.40	51.21	50.92	49.83	48.63	47.38	46.11
30	82.82	80.11	77.72	74.94	72.54	70.35	68.47
40	110.63	105.77	102.44	99.00	96.14	93.40	90.88
50	133.90	128.34	125.57	122.00	118.78	115.43	112.25
60	155.66	151.50	148.66	144.58	140.80	136.89	133.20
70	182.69	176.42	171.87	166.82	162.43	157.99	153.77
80	212.90	199.99	194.18	188.44	183.53	178.54	173.79
90	238.92	220.77	215.42	209.39	204.02	198.53	193.31
100	260.60	241.48	236.31	229.81	223.99	218.05	212.37
110	278.37	263.78	256.99	249.77	243.50	237.09	230.98
120	309.78	284.91	276.91	269.19	262.47	255.66	249.13
130	354.04	303.34	296.07	288.08	280.95	273.76	266.84
140	398.02	322.03	314.90	306.47	298.96	291.40	284.11
150	435.98	342.37	333.35	324.39	316.51	308.60	300.96
160	468.43	360.95	351.11	341.82	333.59	325.36	317.39
170	495.91	376.97	368.30	358.77	350.23	341.69	333.41
180	518.84	394.54	385.19	375.26	366.42	357.60	349.03
190	537.71	413.46	401.60	391.31	382.19	373.10	364.25
200	553.03	428.93	417.39	406.91	397.53	388.19	379.09
210	599.93	442.81	432.79	422.07	412.45	402.89	393.56
220	674.94	461.38	447.85	436.81	426.98	417.20	407.65
230		479.59	462.37	451.13	441.10	431.13	421.38
240		491.66	476.41	465.05	454.83	444.68	434.75
250		506.62	490.14	478.57	468.19	457.88	447.77
260		531.53	503.47	491.70	481.17	470.71	460.46
270		554.35	516.28	504.44	493.79	483.20	472.81
280		568.53	528.77	516.82	506.05	495.34	484.83
290		576.67	540.94	528.83	517.96	507.15	496.53
300		587.92	552.64	540.49	529.53	518.64	507.92
310		616.11	563.97	551.80	540.77	529.80	519.01
320		643.11	575.05	562.77	551.68	540.65	529.79
Execution Time (sec)	45.6	51.0	28.1	20.9	21.6	27.6	27.6

Table 9. Rate of advance predictions for several α weighting coefficients using the kinematic wave model of FE-SURFDSGN.

Advance Time (*min*)	α Weighting Coefficient						
	0	0.10	0.20	0.25	0.30	0.40	0.50
10	23.93	23.93	23.93	23.93	23.93	23.93	23.93
20	47.86	49.24	50.40	50.92	51.40	52.27	53.02
30	75.21	76.24	77.24	77.72	78.19	79.11	80.01
40	100.83	101.55	102.15	102.44	102.73	103.32	103.98
50	122.84	124.39	125.25	125.57	125.86	126.37	126.82
60	144.66	146.77	148.16	148.66	149.09	149.83	150.40
70	169.11	170.30	171.38	171.87	172.32	173.15	173.81
80	192.97	193.22	193.83	194.18	194.54	195.25	195.90
90	213.56	214.26	215.06	215.42	215.78	216.43	217.07
100	232.82	234.62	235.83	236.31	236.73	237.46	238.10
110	253.43	255.38	256.53	256.99	257.40	258.14	258.80
120	274.30	275.61	276.51	276.91	277.28	277.98	278.66
130	293.95	294.67	295.62	296.07	296.46	297.17	297.87
140	312.89	313.33	314.41	314.90	315.32	316.05	316.74
150	331.07	331.91	332.90	333.35	333.75	334.46	335.15
160	347.78	349.69	350.68	351.11	351.50	352.21	352.92
170	364.57	366.68	367.82	368.30	368.73	369.45	370.18
180	382.95	383.60	384.70	385.19	385.63	386.33	387.05
190	400.40	400.20	401.15	401.60	402.02	402.70	403.43
200	414.93	415.89	416.93	417.39	417.82	418.53	419.28
210	429.01	431.10	432.28	432.79	433.24	433.94	434.70
220	444.88	446.28	447.36	447.85	448.29	448.96	449.72
230	460.16	460.88	461.91	462.37	462.81	463.48	464.26
240	473.87	474.75	475.91	476.41	476.87	477.56	478.35
250	487.88	488.48	489.64	490.14	490.61	491.28	492.07
260	501.46	501.95	503.00	503.47	503.92	504.57	505.36
270	513.06	514.66	515.77	516.28	516.75	517.43	518.24
280	525.13	527.03	528.23	528.77	529.25	529.92	530.74
290	538.94	539.35	540.43	540.94	541.40	542.05	542.87
300	551.13	551.06	552.14	552.64	553.11	553.77	554.60
310	561.25	562.23	563.44	563.97	564.44	565.14	565.98
320	572.08	573.37	574.53	575.05	575.51	576.17	577.01
Execution Time (*sec*)	45.7	33.9	29.4	28.1	28.5	33.9	34.2

The results in Table 8 show that an optimum value of *0.5* should be chosen for the time weighting coefficient, θ. This would result in the highest accuracy and the least numerical oscillations. The numerical oscillations have shown to be substantial for $\theta < 0.5$ while the accuracy is also lower. Selecting θ at values in the range of *0.5* to *0.7* produced results as stable as $\theta = 0.5$ but not necessarily as accurate. This was observed in the runs that were presented in Table 8 and in other simulation runs with different input parameters.

On the other hand, it was observed that selecting α at *0.25* would produce optimum results both in terms of numerical accuracy and stability. Even though the results in Table 9 show similar results using the various levels of α that were selected (i.e. *0*, *0.10*, *0.20*, *0.25*, *0.30*, *0.40*, and *0.50*), high levels of numerical oscillations were observed in other simulation runs when α was set at *0* which corresponds to the standard Galerkin formulation of the finite element method. Numerical oscillations were never observed in any of the simulation runs when α was selected within the range of 0.1 to 0.4. These conclusions are consistent with those established by Lapidus and Pinder (1982) and Allen et al. (1988).

The next step was to investigate the sensitivity of FE-SURFDSGN to physical input parameters. The data of irrigation 5, group 2, and furrow 1 of the Benson farm were again used for this purpose (Table 5). These input parameters were designated as the reference data. Variations of these parameters were then established on both sides of the latter data. The reference and formulated input data are summarized in Table 10. These data were used to investigate the sensitivity of the model to variations in physical parameters. The runs were conducted by selecting all of the reference input data except for the input parameter that was varied during the respective run. The results of these simulation runs are presented in Figures 32 through 39.

Table 10. Input data to investigate the sensitivity of the model to the change of various physical parameters.

Parameter	Low	Medium Low	Reference	Medium High	High
Inlet Flow, Q_0 (*lps*)	.60	1.00	1.17	1.35	1.75
Slope, S_0 (m/m)	0.0022	0.0033	.0044	0.0066	0.0088
Manning's Roughness, n	.005	.01	.02	.03	.04
Hydraulic Section Parameter, ρ_1	0.26	0.30	.34	.38	.42
Hydraulic Section Parameter, ρ_2	2.44	2.64	2.84	3.04	3.24
Infiltration Function Coefficient, k ($m^3/m/min^a$)	0.0050	0.0100	0.0173	.0300	0.0400
Infiltration Function Exponent, a	0.005	0.0075	0.01	0.02	0.04
Infiltration Function Coefficient, f_0 ($m^3/m/min$)	0.00002	0.00006	0.00008	0.00020	0.00040

Furrow Length :	*625 m*
Time of Cutoff :	*619 min*
Time Step, Δt :	*10 min*
Time Weighting Coefficient, θ :	*0.50*
Weighting Coefficient, α :	*0.25*
Type of Element :	Linear

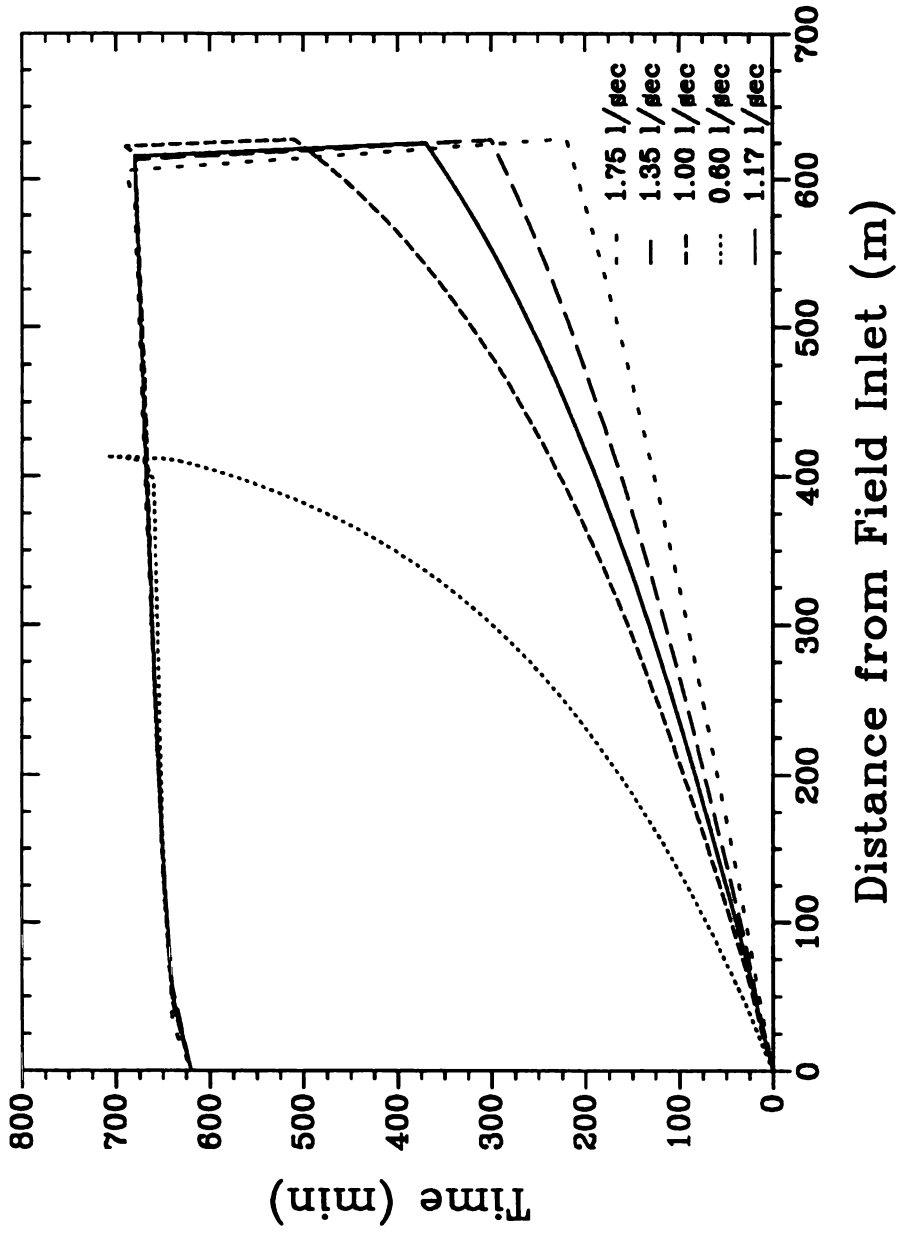


Figure 32. Sensitivity of simulated advance and recession curves of FE-SURFDSGN kinematic wave model to inflow rate, Q_0, at field inlet.

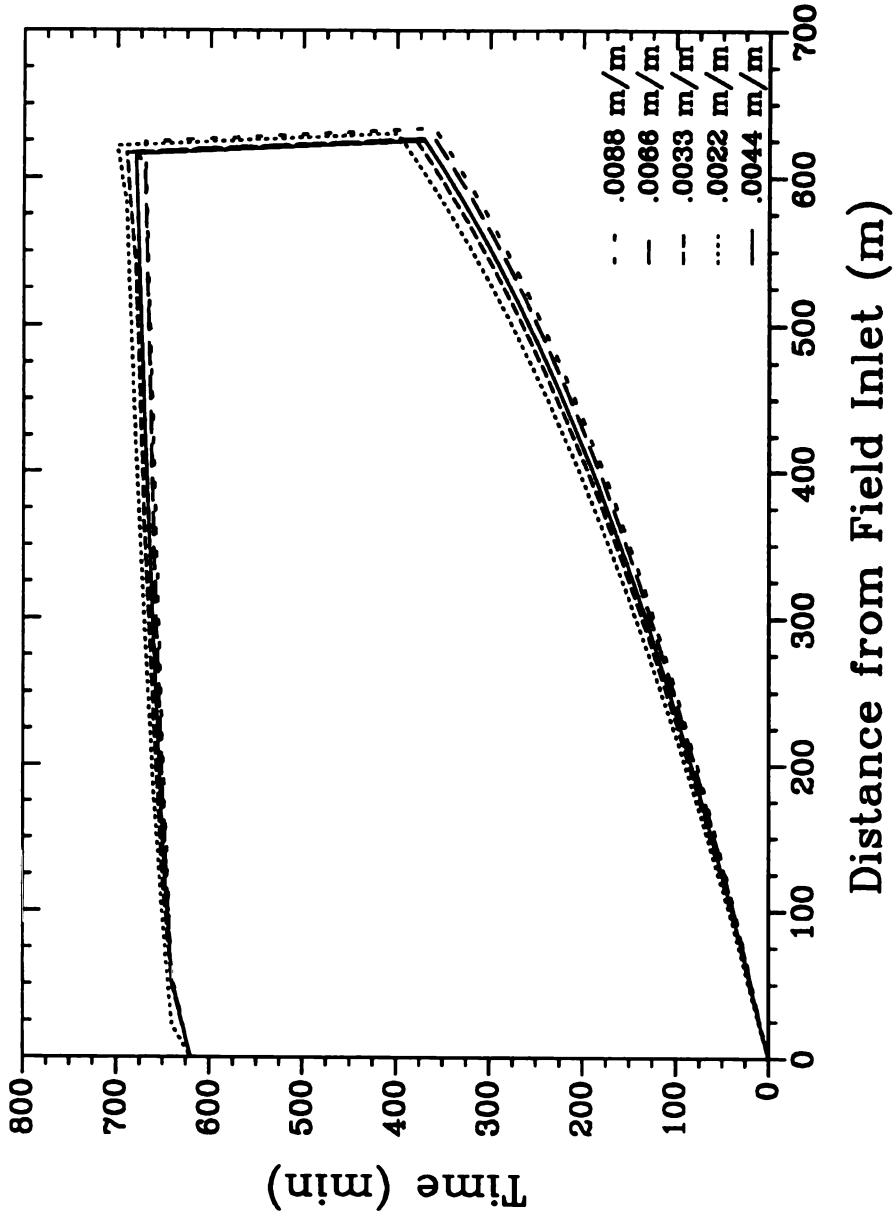


Figure 33. Sensitivity of simulated advance and recession curves of FE-SURFDSGN kinematic wave model to field slope, S_o.

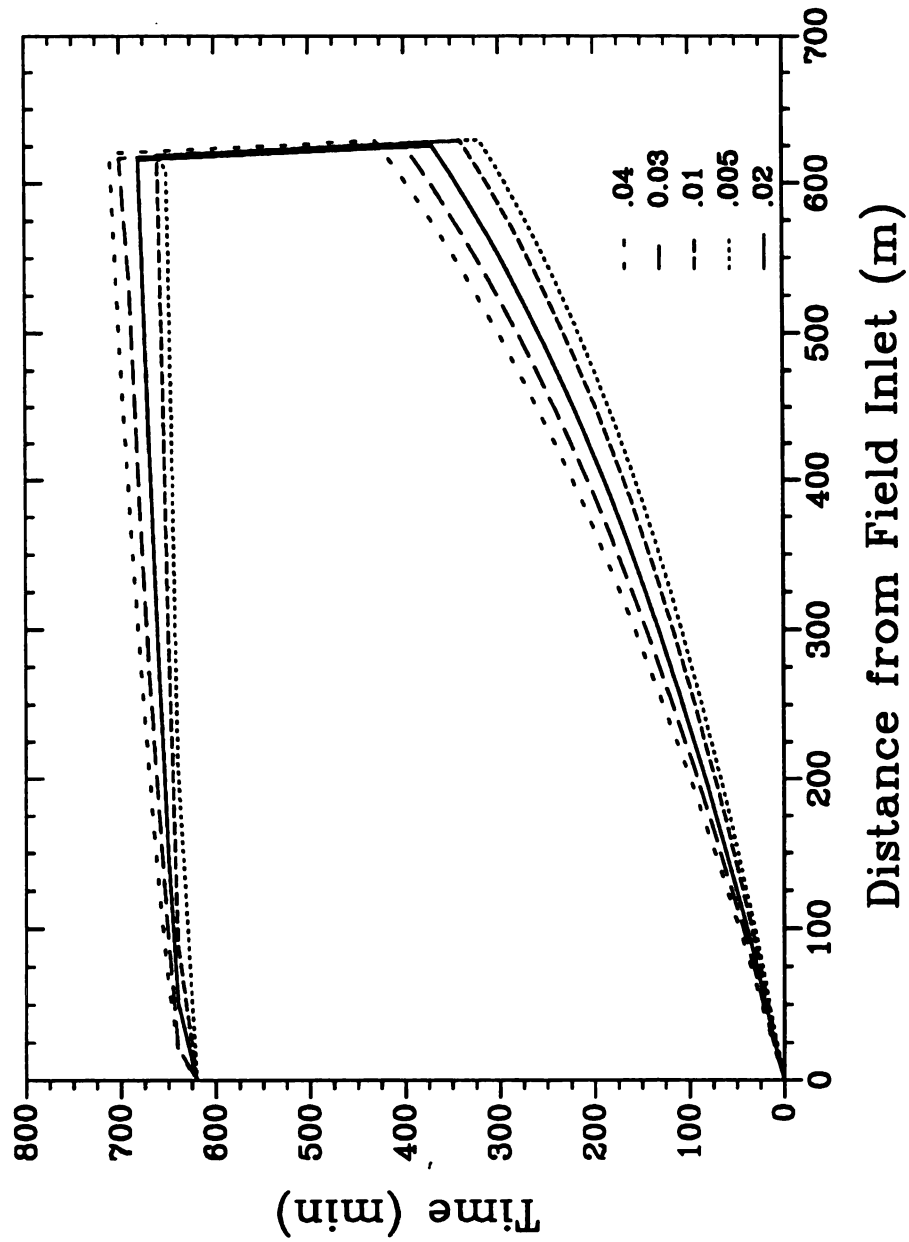


Figure 34. Sensitivity of simulated advance and recession curves of FE-SURFDSGN kinematic wave model to Manning's coefficient, *n*.

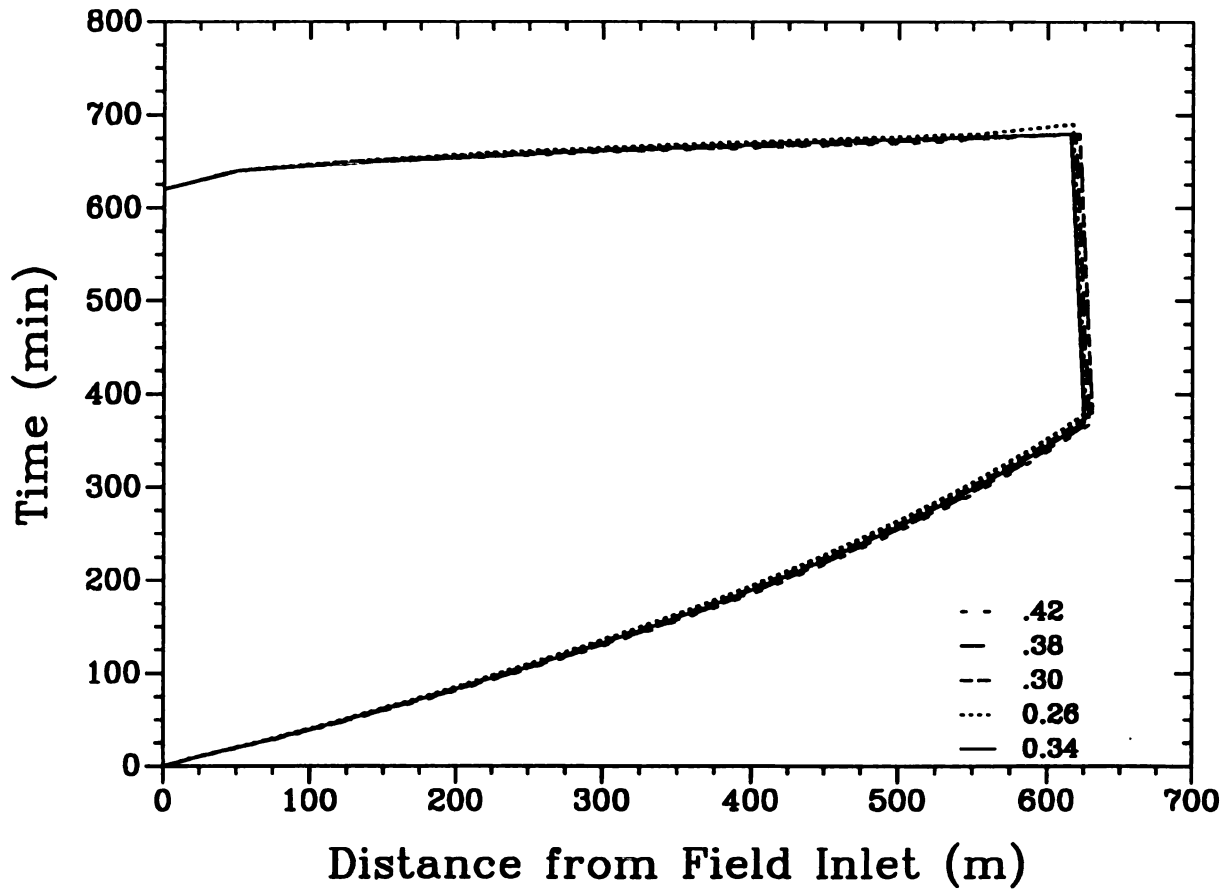


Figure 35. Sensitivity of simulated advance and recession curves of FE-SURFDSGN kinematic wave model to the hydraulic section parameter, ρ_1.

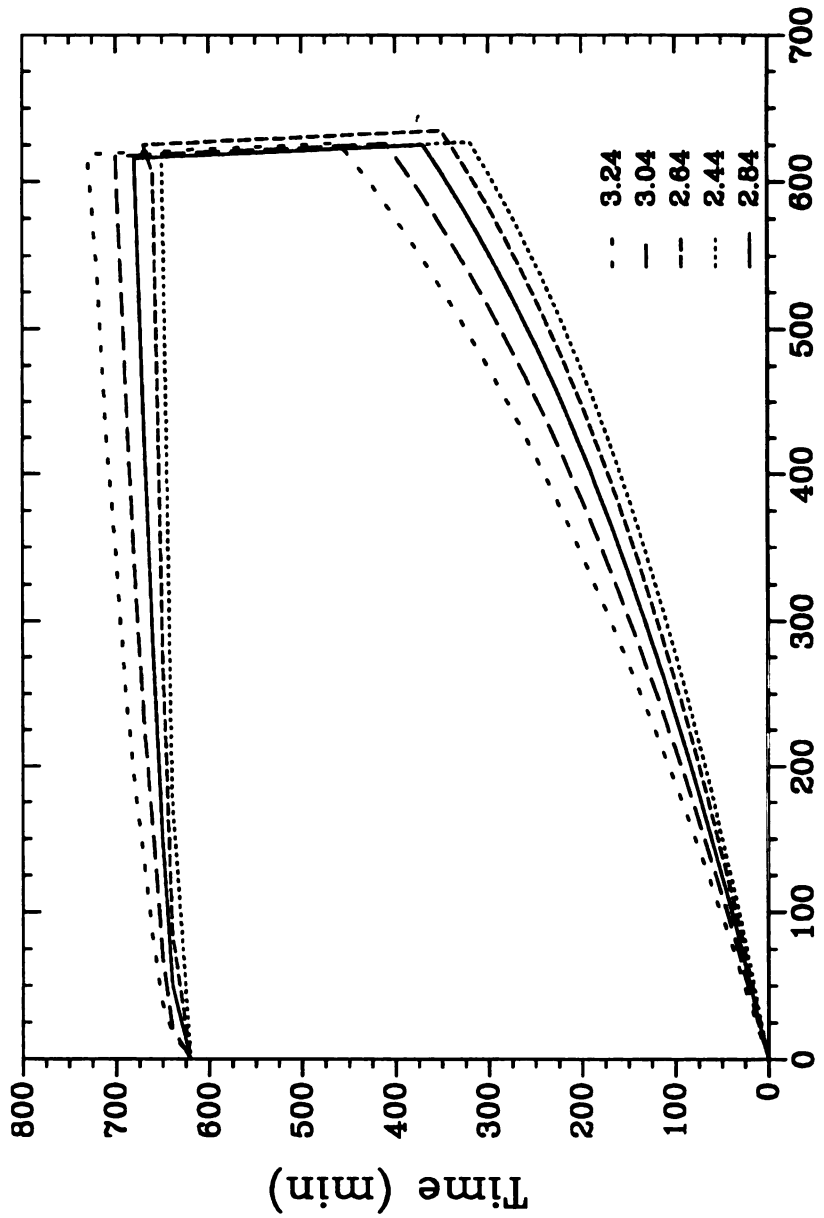


Figure 36. Sensitivity of simulated advance and recession curves of FE-SURFDSGN kinematic wave model to the hydraulic section parameter, ρ_2.

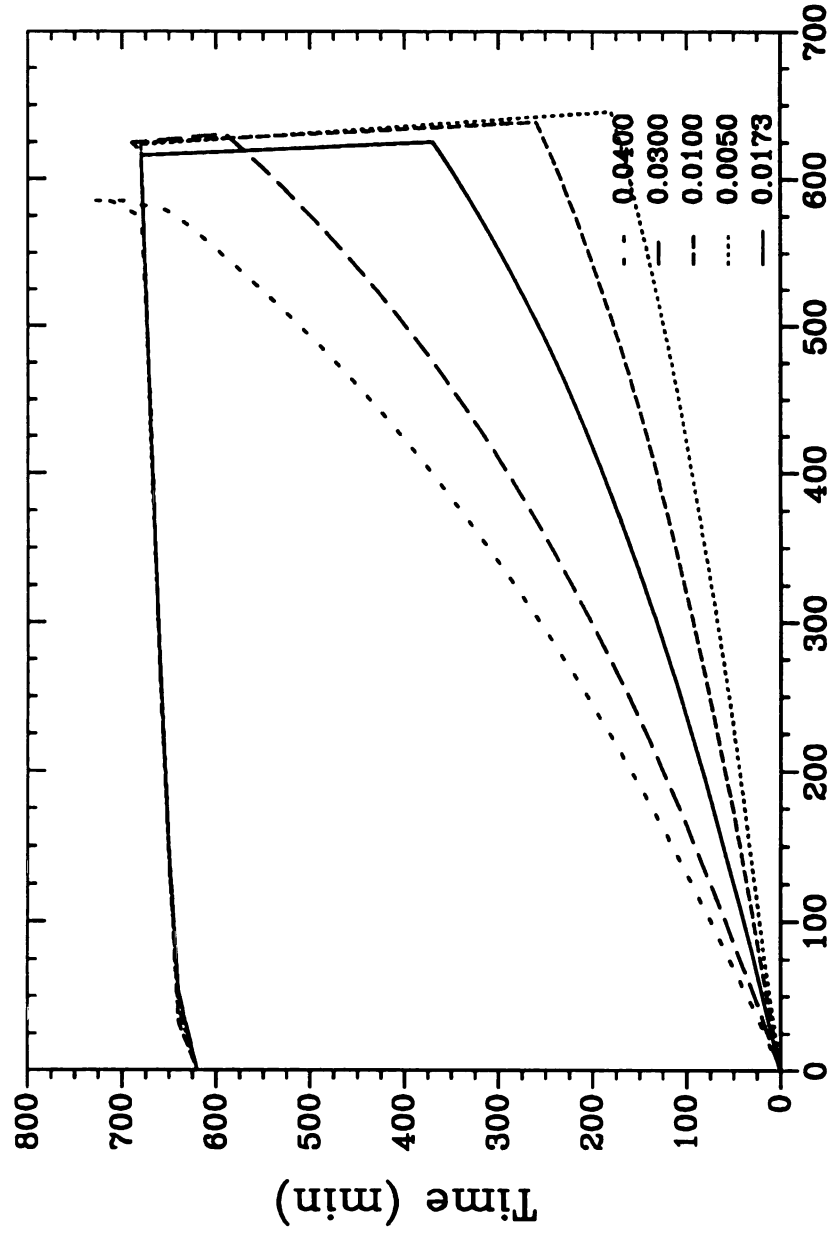


Figure 37. Sensitivity of simulated advance and recession curves of FE-SURFDSGN kinematic wave model to the infiltration function coefficient, k.

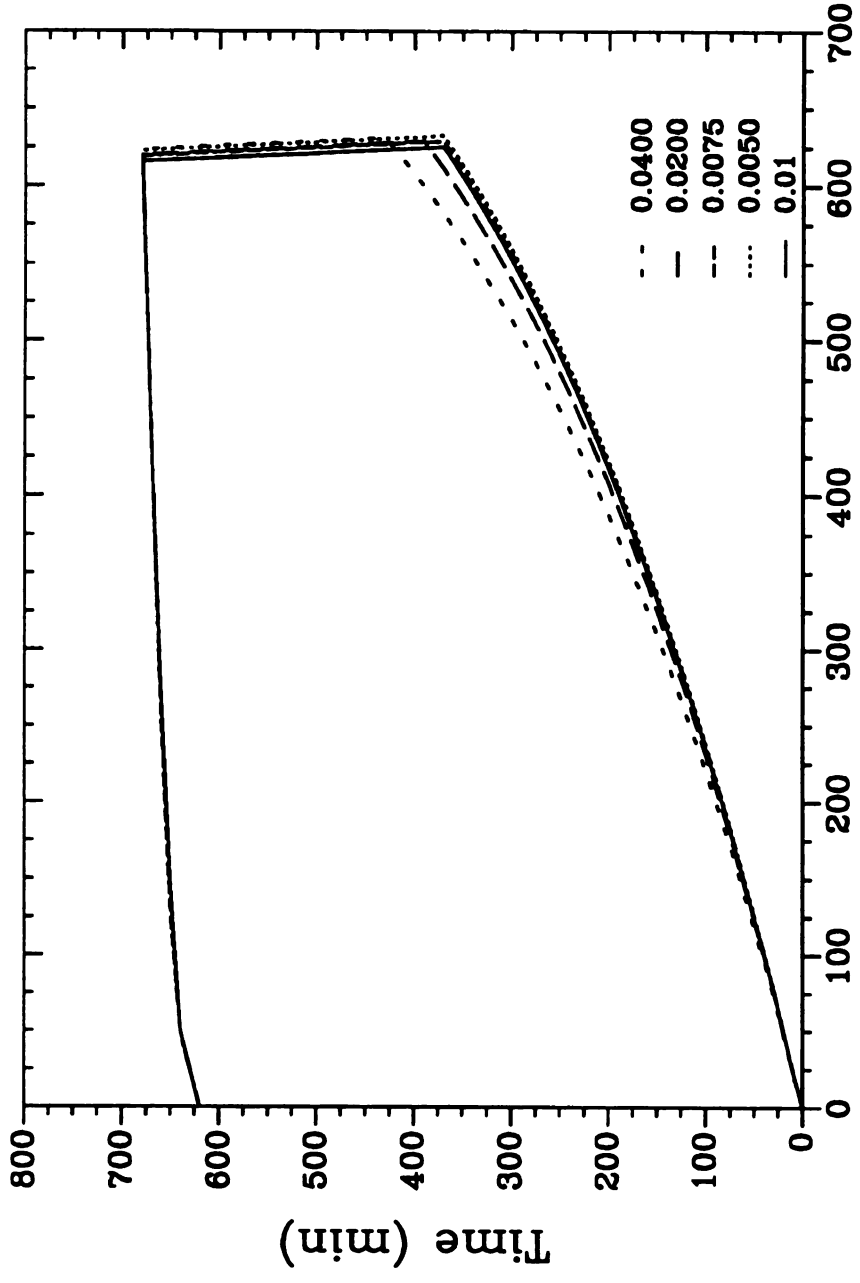


Figure 38. Sensitivity of simulated advance and recession curves of FE-SURFDSGN kinematic wave model to the infiltration function exponent, *a*.

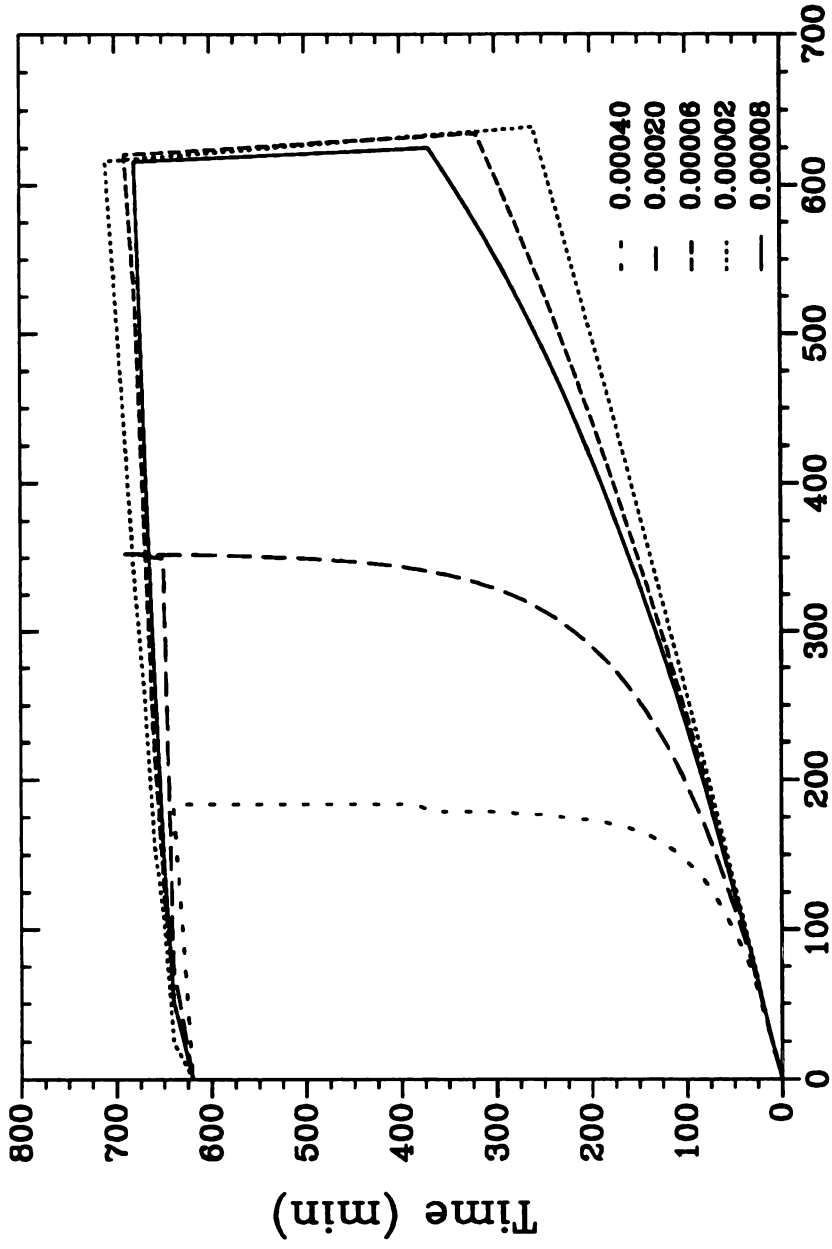


Figure 39. Sensitivity of simulated advance and recession curves of FE-SURFDSGN kinematic wave model to the infiltration function coefficient, f_o.

The sensitivity of the model to the inflow rate at field inlet, Q_0, is very clear in Figure 32. By comparing the predicted advance and recession curves of the reference data (the curve that is represented by a solid line in Figure 32 as well as Figures 33 through 39) to the other curves, it is clear that as the inflow rate increases the predicted advance curve becomes more linear while the predicted recession curve is not affected to any extent. It is also obvious from these curves that as the inflow rate increases from *0.6* to *1.75* l/sec, the advance rate becomes more sensitive to Q_0. When the average inflow rate for a selected run is low, the rate of advance diminishes.

The sensitivity of the model to the slope of the furrow is illustrated in Figure 33. Even though the variation in the slope between various runs was not considerable, it is obvious that there are some variations in both the advance and the recession trajectories. The trend seems to be comparable to that observed in the case of the inflow rate shown in Figure 32. One can also observe that as the slope, S_0, increases the predicted rates of advance and recession become less sensitive to the change in slope. This trend becomes clear when a comparison is done among the advance and recession curves of *0.0044*, *0.0066*, and *0.0088* m/m runs. The sensitivity of the model to the variation in the slope should not be of any concern since accurate field slope measurements could be easily obtained.

Figure 33 illustrates the sensitivity of the model to the Manning's roughness coefficient, n. By examining the advance and recession curves in this figure, it is clear that these curves are sensitive to the variations in the Manning's coefficient. This could be of some concern since it is not easy to assess the Manning's coefficient very accurately in the field. The curves in

Figure 33 depict that there is an inverse relationship between the rates of advance and recession, on the one hand, and the Manning's roughness coefficient on the other.

The sensitivity of the model to the hydraulic section parameters, ρ_1 and ρ_2, is shown in Figures 35 and 36. While the predicted advance and recession curves appear not sensitive to ρ_1, these curves are sensitive to the variations in the hydraulic section parameter, ρ_2. The curves in Figure 36 reveal a similar trend as that depicted in Figure 34 where the sensitivity of the model to the Manning's coefficient is presented. However, it seems that the predicted advance and recession curves become less sensitive to the change in ρ_2 when the latter parameter increases.

The sensitivity of the model to the three infiltration parameters k, a, and f_0 is illustrated in Figures 37, 38, and 39, respectively. The predicted rate of advance curve seems to be very sensitive to the infiltration function coefficients, k and f_0, but less sensitive to the infiltration function exponent, a. The predicted recession curve, however, shows moderate sensitivity only to the infiltration function coefficient, f_0, with no sensitivity to either the infiltration coefficient, k, or the exponent, a. The curves in Figures 37, 38, and 39 show the inverse relationship between the rate of advance and the infiltration parameters. These figures also depict that as the infiltration function parameters decrease the predicted rate of advance becomes less sensitive to the changes in these parameters.

D. Discussion

From the results presented in this chapter, it can be concluded that the finite element formulation of the hydrodynamic equations can be successfully used in the hydraulic analysis of flow conditions in sloping furrows and borders. It was demonstrated that the one-dimensional linear finite element formulation produced excellent predictions of the advance and recession trajectories for almost all the simulation runs conducted under the available field tests. Besides, the computer execution times of the developed computer model (FE-SURFDSGN) were very reasonable even when very small time steps were selected. Although the kinematic wave model applications were the only runs that were presented in this manuscript, similar results are expected from the other models that are not complete in FE-SURFDSGN at the present time. These analyses should be undertaken some time in the future to confirm this claim.

As to the type of element used, it was observed that the quadratic finite element formulations of the complete and simplified forms of the hydrodynamic equations did not produce the expected results. Because of the higher order of the quadratic element, it seems that whatever was gained from the more appropriate approximation of the element to both surface and subsurface profiles of flow was lost due to the unstable behavior of the problem. This was partially expected since higher order elements were known to exhibit this kind of behavior in time-dependent problems.

The development of the nonstandard finite element Galerkin formulation was prepared after realizing that the standard Galerkin formulation of the complete and simplified forms of the hydrodynamic equations had many problems of numerical instabilities. These problems were avoided in the linear finite element developments through the application of the principle of

upstream weighting, a step that was accomplished using non-linear shape functions instead of the usual linear functions. On the other hand, it was observed that the standard finite element formulation in time, or the so-called consistent formulation, was highly unstable. This necessitated the application of yet another nonstandard finite element Galerkin formulation in time for both linear and quadratic finite element formulations. The latter nonstandard Galerkin formulation was referred to in the literature as the lumped formulation.

The problems of numerical instability of the standard finite element Galerkin formulation could be attributed to the nature of the hydrodynamic equations. These two first-order partial differential equations which represent the equations of continuity and momentum are both hyperbolic. Numerical solutions of hyperbolic differential equations are known to exhibit more numerical instabilities compared to parabolic and elliptic partial differential equations (Allen et al., 1975).

The goals of this research have been realized by accomplishing the following five fundamental objectives. The approaches utilized to achieve these objectives are delineated below.

Objective 1. Develop a finite element solution procedure of the Saint-Venant equations for the hydraulic analysis of surface irrigation systems.

The approach that was followed under Objective 1 was to develop a finite element Galerkin formulation of the hydrodynamic equations using linear one-dimensional elements. The finite element development resulted in a general system of first-order differential equations for each individual element in the space domain. The system of ordinary differential equations included the derivatives of the unknowns (cross-sectional area of flow, A, and flow rate,

Q) with respect to time. This system incorporated the contributions of both the continuity and momentum equations which constitute the hydrodynamic or so-called Saint-Venant equations. Each of the hydrodynamic equations resulted in a system of ordinary differential equations, and the resultant two systems were combined into one general system of elemental equations. The mean value theorem for differentiation was then applied to transform the global system of ordinary differential equations into a system of algebraic equations. The global system of algebraic equations should be solved iteratively over time. The elemental equations should be assembled into a global system of equations at various time steps using the direct stiffness procedure. After deriving the standard finite element Galerkin formulation of the hydrodynamic equations using linear elements, the same development was repeated using quadratic one-dimensional elements. The latter development was an attempt to investigate if such an approach would yield faster and more accurate results. By examining the shape of both the advance and recession fronts, it was felt that quadratic elements would model both fronts more adequately compared to linear elements. The final step under Objective 1 was to repeat the previous developments using nonstandard finite element Galerkin formulations of the hydrodynamic equations. The one-dimensional linear and quadratic element developments were repeated under this step. This action was deemed necessary after realizing that the standard Galerkin formulations produced numerical oscillations. The latter problem could be attributed to the unsteady nature of the problem and the presence of sharp advancing fronts during the advance phase.

Objective 2. Create a general solution approach that accommodates the available mathematical models of the Saint-Venant equations in the analysis of surface irrigation systems.

The approach that was followed under Objective 2 was to repeat all the developments of Objective 1 for the simplified forms of the hydrodynamic equations. These models included the hydrodynamic II (the model with a steady momentum equation), zero-inertia, and kinematic wave models. The next step was to derive a general finite element representation that would accommodate the complete as well as the simplified forms of the hydrodynamic model. The general model has different coefficients which vary based on the selected model. By choosing the coefficients of the desired model, the solution process would accomplish the analysis of flow conditions in surface irrigation based on the corresponding model of the hydrodynamic equations. The general development was repeated using quadratic elements. The general linear and quadratic finite element developments could be implemented in developing computer models that simulate the flow conditions in surface irrigation systems. Such models would be independent of the selected form of the hydrodynamic equations. The general finite element formulations using linear and quadratic elements are distinct when considering the elemental equations. However, the numerical solution of the resultant system of equations is independent of the selected type of element after the elemental equations are assembled into a global system of equations using the direct stiffness procedure.

Objective 3. Develop an approach to easily incorporate the varying boundary conditions of the advance, ponding, depletion, and recession phases of surface irrigation into the solution process with minimal arbitrary or experimental parameters.

The approach that was followed under Objective 3 was to develop a procedure for incorporating the appropriate boundary conditions under varying physical phases of flow in an irrigation cycle into the final system of

equations. Prior to the application of this step, the global system of equations would have been assembled using the direct stiffness procedure. Also, the mean value theorem for differentiation would have been applied to transfer the system of ordinary differential equations into a system of algebraic equations. Since most of the boundary conditions in the one-dimensional surface irrigation problem involve boundaries with known-values, a six-step process was devised to incorporate the different boundary conditions. The developed procedure allows the known boundary conditions to be specified as inflow and outflow hydrographs. The global system of equations would be modified at each time step to incorporate the various known boundary conditions starting with the initial conditions where the global system of equations incorporates the contribution of only one element. The same procedure would then be repeated at subsequent time steps until the solution process of the various phases of flow is concluded. The dimensions of the total system of equations would be kept the same at any instance in time.

Objective 4. Develop a finite element computer model for the hydraulic analysis of flow conditions in border and furrow irrigation systems.

The approach that was followed under Objective 4 was to implement the finite element mathematical development of the motion equations in building a computer model that can be utilized in simulating the advance, ponding, depletion, and recession phases of flow in both furrow and border irrigation systems. The analysis could be performed based on either the standard or nonstandard finite element Galerkin formulations and using either linear or quadratic elements. Any combination of the above options could be selected and the finite element analysis could be performed using the complete hydrodynamic (hydrodynamic model I), steady hydrodynamic (hydrodynamic model II), zero-inertia, or kinematic wave models. Currently, the kinematic

wave model is the only model that is fully operational in the present version of the program. The developed computer model has a companion graphics routine which produces a graphical output of the advance and recession curves in addition to the plot of actual field measurements. The computer model was prepared in a modular format and was developed to run on any IBM-compatible microcomputer with a Random Access Memory (RAM) of 512 Kbytes or more and an MS-DOS version 2.00 or higher. The program is very concise in size compared to the number of functions and options that it embodies. It makes use of the banded form of the global matrices that result from the finite element solution. It directly stores the non-symmetrical square matrices in a banded form. The program has a routine that solves the global system of algebraic equations based on the method of Gaussian elimination. The latter routine was developed to solve matrices that are stored in a banded form. This drastically reduces the execution time of the program and makes the simulation run time highly efficient.

Objective 5. Evaluate the predications of the finite element model using actual field measurements from some existing surface irrigation systems.

The approach that was followed under Objective 5 was to compare the results obtained from running the developed finite element computer model to those reported from actual field measurements for some existing surface irrigation systems. Actual field measurements were taken from furrow irrigation evaluations at three Colorado locations and two different locations in Utah and Idaho. The data were originally collected by Colorado State University researchers and the researchers in the Department of Agricultural and Irrigation Engineering at Utah State University, respectively. The actual data were taken from Elliott et al. (1982b), Walker and Humpherys (1983), Walker and Skogerboe (1987), and Oweis (1983). These data included the

physical input parameters as well as the actual measurements of both the advance and recession phases of flow. The developed graphics routine was used to display both simulated and actual data of the various flow phases of irrigation on the same graph. The graphical display included plots of actual and predicted advance and recession trajectories of flow. The results indicated that the developed finite element model simulates the hydraulic analysis of flow conditions in surface irrigation systems fairly well, under the investigated conditions.

V. CONCLUSIONS AND RECOMMENDATIONS

A. Conclusions

Based on the results and discussion presented in this study, the following conclusions can be made:

1. A general finite element formulation was successfully developed for the numerical solution of the complete and simplified forms of the hydrodynamic equations as applied to the hydraulic analysis of surface irrigation systems. This formulation allows for the implementation of the various forms of the Saint-Venant equations without the need for modifying the solution process.

2. The developed computer model (FE-SURFDSGN) based on the finite element formulation of the hydrodynamic equations has proven to be an effective tool in the hydraulic analysis of flow conditions in furrow and border irrigation systems. Even though not all the forms of the hydrodynamic equations are fully operational in FE-SURFDSGN at the present time, it was shown that the model produces excellent results when compared to actual field measurements for existing systems in Colorado, Idaho, and Utah. These results were obtained by using the finite element kinematic wave model with linear elements. Other models are expected to produce results at least as good as those presented using the kinematic wave model, if not better.

3. FE-SURFDSGN model predictions appear to be very consistent with actual field measurements for all phases of flow under the studied cases. The varying boundary conditions of the advance, ponding, depletion, and

recession phases of surface irrigation were easily implemented into the solution process by modifying the system of equations after it had been assembled at various time steps.

4. The standard finite element Galerkin formulation using linear elements was shown to have some problems with respect to numerical instabilities. However, the nonstandard Galerkin formulation resulted in stable solutions for the various simulation runs conducted.

5. Even though the model was developed to run on any IBM compatible microcomputer, the average time for completing a simulation run was still approximately *50* seconds on a 386 machine. The selected time steps under those runs were very conservative in almost all cases. It was shown that good accuracy could be achieved by using larger time steps (*20* or *30* minutes), resulting in execution times as low as 5 seconds. Time steps as high as *20* minutes were very feasible for all the simulation runs conducted.

6. Due to the banded form of the global square matrices, it was possible to analyze systems with as much as 200 elements on any IBM compatible microcomputer with 512 Kbytes or more of memory. This system of equations would contain around 400 equations to be solved simultaneously at various time steps starting from initial conditions. Since the system was assembled without storing any zeros outside the bandwidth and was solved using a modified form of the Gauss elimination method, the execution time for solving these systems of simultaneous equations was a fraction of the time that would have otherwise been needed.

7. The input data necessary for running the model was kept to a bare minimum. All the input parameters required by the model are consistent with those needed with any other numerical surface irrigation model.

B. Recommendations

1. Further work should be devoted towards completing the finite element formulation of the remaining forms of the hydrodynamic equations using linear elements.

2. More simulation runs should be conducted using the various models of the hydrodynamic equations to compare the accuracy of these models using actual field observations. These runs would make it feasible to assess the accuracy of the various models and to estimate the trade offs in using any of the forms of the hydrodynamic equations.

3. The speed and accuracy of FE-SURFDSGN should be compared to existing finite difference surface irrigation models to establish the advantages and disadvantages of numerical methods as applied to the solution of flow conditions in surface irrigation systems.

4. FE-SURFDSGN should be modified to allow for the simulation of surge irrigations. Such a modification could be easily accomplished in FE-SURFDSGN since the finite element development allows for the incorporation of various kinds of boundary conditions including an inflow hydrograph at field inlet.

5. Devote more time to investigate the finite element formulation of the simplified and complete forms of the hydrodynamic equations using quadratic elements. Since the amount of time that was devoted to developing and debugging the options of FE-SURFDSGN that correspond to quadratic elements was not extensive by any means, further work is needed to investigate whether these elements would work and produce results comparable to, or even better than, those observed with linear element solutions.

VI. LIST OF REFERENCES

Abramowitz, M. and I.A. Stegun. 1964. *Handbook of Mathematical Functions and Formulas, Graphs, and Mathematical Tables*. National Bureau of Standards.

Akanbi, A.A. and N.D. Katopodes. 1988. Model for flood propagation on initially dry land. *Journal of Hydraulic Engineering*, ASCE, 114(7):689-706.

Allen, M.B, I. Herrera, and G.F. Pinder. 1988. *Numerical Modeling in Science and Engineering*. John Wiley and Sons, New York, NY.

Aweis, T.Y. 1983. Surge flow furrow irrigation hydraulics with zero inertia. Thesis submitted to Utah State University, Logan, Utah, in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Engineering.

Bajwa, R.S., W.M.Crosswhite, and J.E. Hostetler. 1987. Agricultural irrigation and water supply. Agriculture Information Bulletin 532, Economic Research Service, U.S. Department of Agriculture, Washington, D.C.

Bali, K. and W.W. Wallender. 1987. Water application under varying soil and intake opportunity time. *Transactions of the ASAE* 30(2):442-448.

Bassett, D.L., D.D. Fangmeier, and T. Strelkoff. 1980. Hydraulics of surface irrigation. In: *Design and Operation of Farm Irrigation Systems*. ASAE Monogr. 3, St. Joseph, MI. pp. 447-498.

Bassett, D.L. 1972. Mathematical model of water advance in border irrigation. *Transactions of the ASAE* 15(5):992-995.

Bassett, D.L. and D.W. Fitzsimmons. 1976. Simulating overland flow in border irrigation. *Transactions of the ASAE* 19(4):666-671.

Bautista, E. and W.W. Wallender. 1985. Spatial variability of infiltration in furrows. *Transactions of the ASAE* 28(6):1846-1851,1855.

Binder, R.C. 1973. *Fluid Mechanics*, fifth edition. Prentice-Hall, Englewood Cliffs, N.J.

Bishop, A.A., W.R. Walker, N.L. Allen, and G.J. Poole. 1981. Furrow advance rates under surge flow systems. *Journal of the Irrigation and Drainage Division*, ASCE, 107(IR3):257-264.

Blair, A.W. and E.T. Smerdon. 1987. Modeling surge irrigation infiltration. *Journal of Irrigation and Drainage Engineering*, ASCE, 113(4):497-515.

Blair, A.W. and T.L. Trout. 1989. Recirculating furrow infiltrometer design guide. Technical Report CRWR 223, Center for Research in Water Resources, Austin, Texas 78758-4497.

Bouwer, H. 1957. Infiltration patterns for surface irrigation. *Agricultural Engineering* 38(9):662-669, 676.

Brakensiek, D.L., A.L. Heath, and G.H. Comer. 1966. Numerical techniques for small watershed routing. Agricultural Research Service 41-113, United States Department of Agriculture, Washington, D.C.

Brakensiek, D.L., W.J. Rawls, and W.R. Hamon. 1979. Application of an infiltrometer system for describing infiltration into soils. *Transactions of the ASAE* 22:320-325.

Brakensiek, D.L. 1966. Hydrodynamics of overland flow and nonprismatic channels. *Transactions of the ASAE* 9:119-122.

Brakensiek, D.L. 1967. Kinematic flood routing. *Transactions of the ASAE* 10(3):340-343.

Bralts, V.F. 1983. Hydraulic design and field evaluation of drip irrigation submain units. Thesis submitted to Michigan State University, East Lansing, Michigan, in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Agricultural Engineering.

Bralts, V.F. and L.J. Segerlind. 1985. Finite element analysis of drip irrigation submain units. *Transactions of the ASAE* 28(3):809-814.

Brater, E.F. and H.W. King. 1976. *Handbook of Hydraulics*, sixth edition. McGraw-Hill Book Company, New York, NY.

Brutsaert, W. 1971. De Saint-Venant equations experimentally verified. *Journal of the Hydraulics Division*, ASCE, 97(HY9): 1387-1401.

Chen, Cheng-lung. 1966. Discussion of "A solution of the irrigation advance problems," by O. Wilke and E.T. Smerdon (1965). *Journal of the Irrigation and Drainage Division*, ASCE, 92(IR2):97-101.

Chen, Cheng-lung. 1970. Surface irrigation using kinematic-wave method. *Journal of the Irrigation and Drainage Division*, ASCE, 96(IR1):39-46.

Chen, Cheng-lung and V.E.Hansen. 1966. Theory and characteristics of overland flow. *Transactions of the ASAE* 9:20-26.

Chow, V.T. 1959. *Open-Channel Hydraulics*. McGraw-Hill Book Company, New York. pp. 523-553.

Christiansen, J.E., A.A. Bishop, F.W. Kiefer Jr., and Yu-Si Fok. 1966. Evaluation of intake rate constants as related to advance of water in surface irrigation. *Transactions of the ASAE* 9(5):671-674.

Clemmens, A.J. 1978. Discussion of "Dimensionless solutions of border-irrigation advance," by N.D. Katopodes and T. Strelkoff (1977). *Journal of the Irrigation and Drainage Division*, ASCE, 104(IR3):339-341.

Clemmens, A.J. 1979. Verification of the zero-inertia model for border irrigation. *Transactions of the ASAE* 22(6):1306-1309.

Clemmens, A.J. 1981. Evaluation of infiltration measurements for border irrigation. *Agricultural Water Management* 3(4):251-267.

Clemmens, A.J. and D.D. Fangmeier. 1978. Discussion of "Border-irrigation hydraulics with zero inertia," by T. Strelkoff and N.D. Katopodes (1977). *Journal of the Irrigation and Drainage Division*, ASCE, 104(IR3):337-339.

Clemmens, A.J. and T. Strelkoff. 1979. Dimensionless advance for level-basin irrigation. *Journal of the Irrigation and Drainage Division*, ASCE, 105(IR3):259-273.

Davis, J.R. and A.W. Fry. 1963. Measurement of infiltration rates in irrigated furrows. *Transactions of the ASAE* 6(4):318-319.

Desai, C.S. 1979. *Elementary Finite Element Method.* Prentice-Hall, Englewood Cliffs, N.J. pp.211-223,299-331.

Elliott, R.L., W.R. Walker, and G.V. Skogerboe. 1982a. Furrow irrigation infiltration parameters from advance rate and zero-inertia theory. ASAE Paper No. 82-2103., ASAE, St. Joseph, MI 49085.

Elliott, R.L., W.R. Walker, and G.V. Skogerboe. 1982b. Zero-inertia modeling of furrow irrigation advance. *Journal of the Irrigation and Drainage Division*, ASCE, 108(IR3):179-195.

Elliott, R.L., W.R. Walker, and G.V. Skogerboe. 1983a. Furrow irrigation advance rates: A dimensionless approach. *Transactions of the ASAE* 26(6):1722-1725,1731.

Elliott, R.L., W.R. Walker, and G.V. Skogerboe. 1983b. Infiltration parameters from furrow irrigation advance data. *Transactions of the ASAE* 26(6):1726-1731.

Elliott, R.L. (compiler). 1980. Furrow irrigation field evaluation data. Department of Agricultural and Chemical Engineering, Colorado State University, Fort Collins, Colorado.

Elliott, R.L. and W.R. Walker. 1980. Furrow irrigation infiltration and advance functions. ASAE Paper No. 80-2075., ASAE, St. Joseph, MI 49085.

Elliott, R.L. and W.R. Walker. 1982. Field evaluation of furrow infiltration and advance functions. *Transactions of the ASAE* 25(2):396-400.

Escoffier, F.E. and M.B. Boyd. 1962. Stability ascpects of flow in open channels. *Journal of the Hydraulics Division*, ASCE, 88(HY6): 145-166.

Fangmeier, D.D. and M.K. Ramsey. 1978. Intake characteristics of irrigation furrows. *Transactions of the ASAE* 21(4):696-700,705.

Fangmeier, D.D. and T. Strelkoff. 1979. Mathematical models and border irrigation design. *Transactions of the ASAE* 22(1):93-99.

Fok, Yu-Si. 1967. Infiltration equation in exponential forms. *Journal of the Irrigation and Drainage Division*, ASCE, 93(IR4):125-135.

Fok, Yu-Si and A.A. Bishop. 1965. Analysis of water advance in surface irrigation. *Journal of the Irrigation and Drainage Division*, ASCE, 91(IR1):99-117.

Fonteh, M.F. and T. Podmore. 1989. A furrow irrigation model with spatially varying infiltration. ASAE Paper No. 89-2534, ASAE, St. Joseph, MI 49085.

Guymon, G.L. 1972. Application of the finite element method for the simulation of surface water transport problems. Institute of Water Resources Report No. IWR-21, University of Alaska, Anchorage, Alaska.

Hall, W.A. 1956. Estimating irrigation border flow. *Agricultural Engineering* 37(4):263-265.

Hart, W.E., D.L. Bassett, and T. Strelkoff. 1968. Surface irrigation hydraulics-kinematics. *Journal of the Irrigation and Drainage Division*, ASCE, 94(IR4):419-440.

Heatwole, C.D., V.O. Shanholtz, and B.B. Ross. 1982. Finite element model to describe overland flow on an infiltrating watershed. *Transactions of the ASAE* 25(3):630-637.

Henderson, F.M. 1966. *Open Channel Flow*. Macmillan Publishing Company, New York, NY.

Henderson, F.M. and R.A. Wooding. 1965. Overland flow and groundwater flow from a steady rainfall of finite duration. *Journal of Geophysical Research*, American Geophysical Union, 69(8):1531-1540.

Hillel, D. 1980. *Applications of Soil Physics*. Academic Press, New York, NY.

Hromadka II, T.V. and J.J. DeVries. 1988. Kinematic wave routing and computational error. *Journal of Hydraulic Engineering*, ASCE, 114(2):207-217.

Hu, K.K., S.E. Adeff, and S.Y. Wang. 1989. A new methodology for formulating finite element hydrodynamic models. In: *Finite Element Analysis in Fluids*. Proceedings of the Seventh International Conference on Finite Element Methods in Flow Problems, The University of Alabama in Huntsville, Alabama. pp. 605-610.

Izadi, B. and W.W. Wallender. 1985. Furrow hydraulic characteristics and infiltration. *Transactions of the ASAE* 28(6):1901-1908.

Izuno, F.T., T.H. Podmore, and H.R. Duke. 1985. Infiltration under surge irrigation. *Transactions of the ASAE* 28(2):517-521.

Izuno, F.T. and T.H. Podmore. 1985. Kinematic wave model for surge irrigation research in furrows. *Transactions of the ASAE* 28(4):1145-1150.

Izuno, F.T. and T.H. Podmore. 1986. Surge irrigation management. *Agricultural Water Management* 11:279-291.

James L.G. 1988. *Principles of Farm Irrigation System Design*. John Wiley and Sons, New York, NY.

Jaynes, D.B. 1986. Simple model of border irrigation. *Journal of Irrigation and Drainage Engineering*, ASCE, 112(2):172-183.

Judah, O.M., V.O. Shanholtz, and D.N. Contractor. 1975. Finite element simulation of flood hydrographs. *Transactions of the ASAE* 18:518-522.

Kaneko, N. 1989. A periodic analysis of shallow water equation. In: *Finite Element Analysis in Fluids*. Proceedings of the Seventh International Conference on Finite Element Methods in Flow Problems, The University of Alabama in Huntsville, Alabama. pp. 635-640.

Kashiyama, K. and M. Kawahara. 1989. Finite element analysis of shallow water flow using adaptive mesh refinements. In: *Finite Element Analysis in Fluids*. Proceedings of the Seventh International Conference on Finite Element Methods in Flow Problems, The University of Alabama in Huntsville, Alabama. pp. 683-688.

Katopodes, N.D., J. Tang, and A.J. Clemmens. 1977b. Estimation of surface irrigation parameters. *Journal of Irrigation and Drainage Engineering*, ASCE, 116(5):676-696.

Katopodes, N.D. 1974. Shallow water equations at low Froude numbers. Thesis submitted to the University of California, at Davis, in partial fulfillment of the requirements for the degree of Masters of Science.

Katopodes, N.D. 1984. A dissipative Galerkin scheme for open-channel flow. *Journal of Hydraulic Engineering*, ASCE, 110(4):450-466.

Katopodes, N.D. and D.R. Schamber. 1983. Applicability of dam-break flood wave models. *Journal of Hydraulic Engineering*, ASCE, 109(5):702-721.

Katopodes, N.D. and T. Strelkoff. 1977a. Hydrodynamics of border irrigation-complete model. *Journal of the Irrigation and Drainage Division*, ASCE, 103(IR3):309-324.

Katopodes, N.D. and T. Strelkoff. 1977b. Dimensionless solutions of border-irrigation advance. *Journal of Irrigation and Drainage Division*, ASCE, 103(IR4):401-417.

Kay, M. 1986. *Surface Irrigation: Systems and Practice*. Cranfield Press, England, United Kingdom.

Kaytal, A.K., G.S. Campbell, and L.G. King. 1987. Improved infiltration function for furrow and border irrigation. ASAE Paper No. 87-2640, ASAE, St. Joseph, MI 49085.

Kemper, W.D., B.J. Ruffing, and J.A. Bondurant. 1982. Furrow intake rates and water management. *Transactions of the ASAE* 25(2):333-339, 343.

Kemper, W.D., T.J. Trout, A.S. Humpherys, and M.S. Bullock. 1988. Mechanisms by which surge irrigation reduces furrow infiltration rates in a silty loam soil. *Transactions of the ASAE* 31(3):821-829.

Kincaid, D.C., D.F. Heermann, and E.G. Kruse. 1972. Hydrodynamics of border irrigation advance. *Transactions of the ASAE* 15(4):674-680.

Kruger, W.E. and D.L. Bassett. 1965. Unsteady flow of water over a porous bed having constant infiltration. *Transactions of the ASAE* 8(1):60-62.

Kunze, R. J. and H. M. Kar-Kuri. 1983. Gravitational flow in infiltration. Proceedings of the National Conference on Advances in Infiltration, American Society of Agricultural Engineers, St. Joseph, Michigan. pp. 14-23.

Kunze, R.J. and W.H. Shayya. 1990. FINDIT: A new 3-directional infiltration model with graphics. ASAE Paper No. 90-2545, ASAE, St. Joseph, MI 49085.

Lapidus, L. and G.F. Pinder. 1982. *Numerical Solution of partial Differential equations in Science and Engineering*. John Wiley and Sons, New York, NY.

Lewis, M.R. and W.E. Milne. 1938. Analysis of border irrigation. *Agricultural Engineering* 19:267-272.

Liggett, J.A. and D.A. Woolhiser. 1967. Difference solutions of the shallow-water equation. *Journal of the Engineering Mechanics Division*, ASCE, 93(EM2):39-71.

Liggett, J.A. and J.A. Cunge. 1975. Numerical methods of solution of the unsteady flow equations. In: *Unsteady Flow in Open Channels*, Volume I. K. Mahmood, K. (Ed.) and V. Yevjevich (Assoc. Ed.), Water Resources Pub., Ft. Collins, Colorado. pp. 89-182.

Lighthill, M.J. and G.B. Whitham. 1955. On kinematic waves, 1. Flood movements in long rivers. Proc. Roy. Soc. (London), Series A, 229:281-316.

Martin, C.S. and D.C. Wiggert. 1975. Discussion of "Simulation accuracies of gradually varied flow," by J.P. Jolly and V. Yevjevich (1974). *Journal of the Hydraulic Division*, ASCE, 101(HY7):1021-1024.

Miller, W.A. and J.A. Cunge. 1975. Simplified equations of unsteady flow. In: *Unsteady Flow in Open Channels*, Volume I. K. Mahmood, K. (Editor) and V. Yevjevich (Assoc. Editor), Water Resources Publications, Ft. Collins, Colorado.

Miller, W.A. and V. Yevjevich (Editors). 1975. *Unsteady Flow in Open Channels*, Volume III. Water Resources Publications, Ft. Collins, Colorado.

Morgali, J.R. and R.K. Linsley. 1965. Computer analysis of overland flow. *Journal of the Hydraulics Division*, ASCE, 91(HY3):81-100.

Norum, D.I. and D.M. Gray. 1970. Infiltration equations from rate-of-advance data. *Journal of the Irrigation and Drainage Division*, ASCE, 96(IR2):111-119.

Parlange, J.Y., J.L. Starr, and R. Haverkamp. 1987. Research needs in infiltration theory. ASAE Paper No. 87-2537, ASAE, St. Joseph, MI 49085.

Philip, J.R. 1957a. The theory of infiltration: 4. Sorptivity and algebraic infiltration equations. *Soil Science* 84:257-264.

Philip, J.R. 1957b. The theory of infiltration: 5. The influence of initial moisture content. *Soil Science* 84:329-337.

Philip, J.R. and D.A. Farrell. 1964. General solution of the infiltration advance problem in irrigation hydraulics. *Journal of Geophysical Research*, American Geophysical Union, 69:621-631.

Ponce V.M., R. Li, and D.B. Simons. 1978. Applicability of kinematic and diffusion models. *Journal of Hydraulics Division*, ASCE, 104(HY3):353-360.

Ponce V.M. 1987. Diffusion wave modeling of catchment dynamics. *Journal of Hydraulic Engineering*, ASCE, 112(8):716-727.

Ponce V.M. and F.D. Theurer. 1982. Accuracy criteria in diffusion routing. *Journal of Hydraulics Division*, ASCE, 104(HY6):747-757.

Potter, M.C. and D.C. Wiggert. 1991. *Mechanics of Fluids*. Prentice Hall, Englewood Cliffs, NJ.

Price, R. K. 1974. Comparison of four numerical methods for flood routing. *Journal of the Hydraulics Division*, ASCE, 100(HY7): 879-899.

Ragan, R.M. 1966. Laboratory Evaluation of a numerical flooding routing technique for channels subject to lateral inflow. *Water Resources Research* 2(1):111-121.

Rayej, M. and W.W. Wallender. 1985. Furrow irrigation simulation time reduction. *Journal of Irrigation and Drainage Engineering*, ASCE, 111(2):134-146.

Ross, B.B., D.N. Contractor, and V.O. Shanholtz. 1977. Finite element simulation of overland and channel flow. *Transactions of the ASAE* 28(4):705-712.

Ross, B.B., D.N. Contractor, and V.O. Shanholtz. 1979. A finite element model of overland and channel flow for assessing the hydrologic impact on landuse change. *Journal of Hydrology* 4:11-30.

Ross, B.B., D.N. Contractor, and V.O. Shanholtz. 1980. A one-dimensional finite element structure for modeling hydrology of small upland watersheds. Proceedings of the Hydrologic Transport Modeling Symposium, ASAE Publication No. 4-80:42-59.

Sakkas, J.G. and T. Strelkoff. 1974. Hydrodynamics of surface irrigation-advance phase. *Journal of the Irrigation and Drainage Division*, ASCE, 100(IR1):31-48.

Samani, Z.A., W.R. Walker, and L.S. Willardson. 1985. Infiltration under surge flow irrigation. *Transactions of the ASAE* 28(5):1539-1542.

Schmitz, G.H. and G.J. Seus. 1991. Mathematical zero-inertia modeling of surface irrigation: Advance in borders. *Journal of Irrigation and Drainage Engineering*, ASCE, 116(5):603-615.

Schwankl, L.J. and W.W. Wallender. 1988. Zero inertia furrow modeling with variable infiltration and hydraulic characteristics. *Transactions of the ASAE* 31(5):1470-1475.

Segerlind, L.J. 1984. *Applied Finite Element Analysis*. John Wiley and Sons, New York, NY.

Sherman, B. and V.P. Singh. 1978. A kinematic model for surface irrigation. *Water Resources Research* 14(2):357-364.

Sherman, B. and V.P. Singh. 1982. A kinematic model for surface irrigation: An extension. *Water Resources Research* 18(3):659-667.

Singh, P. and H.S. Chauhan. 1972. Shape factors in irrigation water advance equation. *Journal of the Irrigation and Drainage Division*, ASCE, 98(IR3):443-458.

Singh, V.P. 1975. Hybrid formulation of kinematic wave models of watershed runoff. *Journal of Hydrology*, Elsvier Scientific Publishing Company, Amsterdam, The Netherlands 27:33-50.

Singh, V.P. 1976. A note on the step error of some finite-difference schemes used to solve kinematic wave equations. *Journal of Hydrology*, Elsvier Scientific Publishing Company, Amsterdam, The Netherlands 30(3):247-255.

Singh, V.P. and D.A. Woolhiser. 1976. A nonlinear kinematic wave model for watershed surface runoff. *Journal of Hydrology*, Elsvier Scientific Publishing Company, Amsterdam, The Netherlands 31:221-243.

Sirjani, F. and W.W. Wallender. 1989. Stochastic infiltration from advance in furrows. *Transactions of the ASAE* 32(2):649-654.

Smith, R.E. 1972. Border irrigation advance and ephemeral flood waves. *Journal of the Irrigation and Drainage Division*, ASCE, 98(IR2):289-307.

Streeter V.L. and E.B. Wylie. 1967. *Hydraulic Transients*. McGraw-Hill Book Company, New York, NY.

Streeter V.L. and E.B. Wylie. 1979. *Fluid Mechanics*, seventh edition. McGraw-Hill Book Company, New York, NY.

Strelkoff, T. 1969. One-dimensional equations of open-channel flow. *Journal of the Hydraulics Division*, ASCE, 95(HY3):861-876.

Strelkoff, T. 1970. Numerical solution of the Saint-Venant equations. *Journal of the Hydraulics Division*, ASCE, 96(HY1):223-252.

Strelkoff, T. 1972. Prediction of increases in surface-irrigation efficiency. Presented at the ASCE National Water Resources Engineering Meeting held at Altlanta, Georgia.

Strelkoff, T. 1977. Algebraic computation of flow in border irrigation. *Journal of the Irrigation and Drainage Division*, ASCE, 103(IR3):357-377.

Strelkoff, T. and A.J. Clemmens. 1981. Dimensionless stream advance in sloping borders. *Journal of the Irrigation and Drainage Division*, ASCE, 107(IR4):361-382.

Strelkoff, T. and F. Souza. 1984. Modeling effect of depth on furrow infiltration. *Journal of the Irrigation and Drainage Division*, ASCE, 110(IR4):375-387.

Strelkoff, T. and N.D. Katopodes. 1977a. Border-irrigation hydraulics with zero inertia. *Journal of the Irrigation and Drainage Division*, ASCE, 103(IR3):325-342.

Strelkoff, T. and N.D. Katopodes. 1977b. End depth under zero-inertia conditions. *Journal of the Hydraulics Division*, ASCE, 103(HY7):699-711.

Swain, E.D. and D.A. Chin. 1990. Model of flow in regulated open-channel networks. *Journal of Irrigation and Drainage Engineering*, ASCE, 116(4):537-556.

Taylor, C. G. Al-Mashidani, and J.M. Davis. 1974. A finite element approach to watershed runoff. *Journal of Hydrology* 21(3):231-246.

Tinney, E.R. and D.L. Bassett. 1961. Terminal shape of a shallow liquid front. *Journal of the Hydraulics Division*, ASCE, 87(HY5):117-133.

Trout, T.J. 1986. Flow velocity and wetted perimeter effect on furrow infiltration. ASAE Paper No. 86-2573, ASAE, St. Joseph, MI 49085.

Trout, T.J. and B.E. Mackey. 1985. Furrow inflow and infiltration variability. ASAE Paper No. 85-2588, ASAE, St. Joseph, MI 49085.

Trout, T.J. and B.E. Mackey. 1988a. Furrow inflow and infiltration variability. *Transactions of the ASAE* 31(2):531-537.

Trout, T.J. and B.E. Mackey. 1988b. Inflow-outflow infiltration measurement accuracy. *Journal of Irrigation and Drainage Engineering*, ASCE, 114(2):256-265.

U. S. Soil Conservation Service. 1974. Border irrigation. Chapter 4, section 15, *National Engineering Handbook*, United States Department of Agriculture, Soil Conservation Service, Washington, D.C.

U. S. Soil Conservation Service. 1984. Furrow irrigation. Chapter 5, section 15, *National Engineering Handbook*, United States Department of Agriculture, Soil Conservation Service, Washington, D.C.

Vieux, B.E. 1988. Finite element analysis of hydrologic response areas using geographic information systems. Thesis submitted to Michigan State University, East Lansing, Michigan, in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Agricultural Engineering.

Walker, W.R. and A.S. Humpherys. 1983. Kinematic-wave furrow irrigation model. *Journal of the Irrigation and Drainage Division*, ASCE, 109(IR4):377-392.

Walker, W.R. and G.V. Skogerboe. 1987. *Surface Irrigation Theory and Practice*. Prentice-Hall, Englewood Cliffs, N.J.

Walker, W.R. and J.D. Busman. 1990. Real-time estimation of furrow infiltration. *Journal of Irrigation and Drainage Engineering*, ASCE, 116(3):299-318.

Walker, W.R. and T.S. Lee. 1981. Kinematic-wave approximation of surged furrow advance. ASAE Paper No. 81-2544, ASAE, St. Joseph, MI 49085.

Wallender, W.W. and M. Rayej. 1985. Zero-inertia surge model with wet-dry advance. *Transactions of the ASAE* 28(5):1530-1534.

White, F.M. 1979. *Fluid Mechanics*, second edition. McGraw-Hill Book Company, New York, NY.

Wilke, O. and E.T. Smerdon. 1965. A solution of the irrigation advance problem. *Journal of the Irrigation and Drainage Division*, ASCE, 91(IR3):23-34.

Wooding, R.A. 1965a. A hydraulic model for the catchment-stream problem, 1. Kinematic wave theory. *Journal of Hydrology* 3(3):254-267.

Wooding, R.A. 1965b. A hydraulic model for the catchment-stream problem, 2. Numerical solutions. *Journal of Hydrology* 3(3):268-282.

Woolhiser, D.A. 1969. Overland flow on a converging surface. *Transactions of the ASAE* 2(4):460-462.

Woolhiser, D.A. and J.A. Liggett. 1967. Unsteady, one-dimensional flow over a plane-the rising hydrograph. *Water Resources Research* 3(3):753-771.

Wylie, E.B. and V.L. Streeter. 1983. *Fluid Transients*. FEB Press, Ann Arbor, MI 48106.

Yen, B.C. 1973. Open-channel flow equations revisited. *Journal of the Engineering Mechanics Division*, ASCE, 99(EM5):979-1009.

APPENDICES

APPENDIX A

FE-SURFDSGN Computer Model Listing

```
REM ************************************************************************
REM *                                                                      *
REM *                 FINITE ELEMENT SURFACE IRRIGATION                    *
REM *                          DESIGN PROGRAM                              *
REM *                                                                      *
REM *----------------------------------------------------------------------*
REM *                        Program SURFDSGN.BAS                          *
REM ************************************************************************
REM *                                                                      *
REM *                             Developed By                             *
REM *                                                                      *
REM *                           Walid H. Shayya                            *
REM *                Department of Agricultural Engineering                *
REM *                      Michigan State University                       *
REM *                                                                      *
REM *                             July 30, 1991                            *
REM *                                                                      *
REM ************************************************************************

COMMON NGphFil$

%DimAry = 302
%DimCol = 7
%DimCo2 = 11
%DimAryh = 151

DIM C(%DimAry,1),S(%DimAry)

dim x(%DimAry,%DimCo2),y(%DimAry,%DimCol),z(%DimAry,%DimCol)
dim phi(%DimAry,1),Kmatrx(%DimAry,%DimCol),Cmatrx(%DimAry,%DimCol)
dim templ(%DimAry,%DimCol),Force(%DimAry,1),ForceP(%DimAry,1)
dim Fstar(%DimAry,1),phil(%DimAry,1),pfx(%DimAry,1)
dim Amatrx(%DimAry,%DimCo2),Pmatrx(%DimAry,%DimCol)

dim PrevPhi(%DimAry,1)

dim PFplusFs(%DimAry,1)

dim tempF1(%DimAry,1),tempF2(%DimAry,1)

dim Length(%DimAryh,1),Infil(%DimAryh,1),TofOpp(%DimAryh,1)
dim S0(%DimAryh,1),TopWidth(%DimAryh,1),InfilP(%DimAryh,1)
dim Fsubi(%DimAryh,1),Fsubj(%DimAryh,1),Fsubk(%DimAryh,1)
dim Coef1(%DimAryh,3),Coef2(%DimAryh,3),Coef3(%DimAryh,3)
dim Coef4(%DimAryh,3),Coef5(%DimAryh,3)

dim Czi(%DimAryh,1),Czj(%DimAryh,1),Czk(%DimAryh,1)

dim ElmtKmtx(6,6),ElmtCmtx(6,6),TempK(6,6),TempC(6,6),NofPhi(%DimAryh,1)
dim ATmp$(11)

DIM GrphXpnt(4), XX%(10), YY%(10), CC$(10), AA$(10)

GOSUB InitialScrn

'+---------------------------------------------------------------------+
'                          DEFINITION OF VARIABLES
'                          =======================
' NumElem%     :  Total number of elements
' NumNode%     :  Total number of nodes
' NP%          :  Total number of unknowns (2 * NumNode%)
' NumElmNode% :  Number of nodes per element (2 for Linear Element and
```

```
'                3 for Quadratic Element)
' NumBandW%    :  Band width of the global stiffness and capacitace
'                matrices
' NumofPhi%    :  Number of known Phi values
' phi(i,1)     :  Vector of unkowns Q and A
' phi1(i,1)    :  Vector of unkowns Q and A at time t - δT
' Force(i,1)   :  Force vector
' ForceP(i,1)  :  Force vector of the previous time step
' Fstar(i,1)   :  Combination of the vectors Force(i,1) and ForceP(i,1)
' Kmatrx(i,i)  :  Global stiffness matrix [K]
' Cmatrx(i,i)  :  Global capacitance matrix [C]
' Amatrx(i,i)  :  [C] + Θ.δT.[K]  = [A]
' Pmatrx(i,i)  :  [C] - (1 - Θ).δT.[K]  = [P]
' ElmtKmtx(6,6): Element stiffness matrix
' ElmtCmtx(6,6): Element capacitance matrix
' NofPhi(10,1): Number representing known Phis as boundary conditions
' Coef(i,5)    :  Coefficients in the element [C] and [K] matrices
' VARTheta     :  Is the parameters that determines the model for the
'                 finite difference solution in time where
'                        Θ = 0 for forward difference
'                        Θ = 1/2 for central difference
'                        Θ = 1 for backward difference
'                        Θ = 2/3 for Galerkin
' ALPHA        :  A coefficient that determines if the formulation is
'                 either a finite element Galerkin formulation or non-
'                 Galerkin formulation.  If Alpha = 0 then the solution
'                 is Galerkin.  Otherwise, the solution is asymetric.
'-----------------------------------------------------------------------
' Qin          :  Inlet flow rate
' Rho1 & Rho2  :  Characteristics of the hydraulic section of the furrow
'                 or border where
'                          2  1.33        μ2
'                         A  R      = μ1 A
'                         Rho1 = μ1
'                         Rho2 = μ2
'                 For the case of the border irrigation, the values of
'                 Rho1 and Rho2 are 1 and 3.33, respectively.
' Sigma1 and
'   Sigma2     :  Empirical fitting constants controlled by the
'                 characteristics of the hydraulic section of the furrow
'                 or border where
'                                      σ2
'                         y       = σ1 A
'                         Sigma1 = σ1
'                         Sigma2 = σ2
'                 For the case of the border irrigation, the values of
'                 Sigma1 and Sigma2 are 1.5 and 1.0, respectively.
' Gravity      :  Gravitational acceleration (9.81 m/sec^2)
' IrrMethod$   :  Irrigation method withsolving the surface irrigation
'                        "F" for Furrow irrigation
'                        "B" for ,Border Irrigation
' Method$      :  Numerical method for solving the surface irrigation
'                 problem where the variable Method$ equals
'                        "H" for Hydrodynamic Model I (Continuity and
'                             Unsteady momentum equations)
'                        "S" for Hydrodynamic Model II (Continuity and
'                             steady momentum equations)
'                        "Z" for Zero Inertia Model
'                        "K" for Kinematic Wave Model
' Phase%       :  An integer that represents the phase of flow where
'                 "1" - Advance
'                 "2" - Ponding
' PhaseRec%    :  An Integer to denote if the recession phase has
'                 started (1) or not yet (0)
```

```
' FDorFE%      :  Determines if Ai & Qi to be used in matrices for
'                 calculating coefficients c1, c2, c3, c4, and c5 (FD)
'                 instead of Ai, Aj, Ak, Qi, Qj, and Qk (FE) where
'                 "1" selects the finite difference and
'                 "0" selects the finite element approach
' CONSISorLUMP%: Determines if the consistent or the lumped finite
'                 element formulations for the change of Phi with respect
'                 to time ought to be used where
'                 "1" selects the consistent formulation and
'                 "0" selects the lumped formulation
' TypElem$      :  Specifies the type of element where the variable
'                 TypElem$ equals
'                        "L" for linear element
'                        "Q" for quadratic element
' FGNam$        :  Represents the name of the configuration file
' LevPrt$       :  Specifies the level of print out which varies from
'                 0 to 2.  Specifying 0 produces no print out and 2
'                 extensive print out.
' SelPrtOpt$    :  A variable to select the output device.  In order to
'                 select the output device, you ought to enter the
'                 following for the variable SelPrtOpt$
'                        "S" for screen
'                        "P" for printer
'                        "F" for the data file "FESIDP.OUT"
' DeltaT        :  Time step, δT
' TotalTime     :  Total elapsed time since time 0
' TotalLength   :  Accumulated length of flow for the advance phase
'                 and constant thereafter
' NumStep%      :  Number of elapsed time steps
' Length(i,1)   :  Length of element i
' Infil(i,1)    :  Infiltration depth at individual nodes (I).  The
'                 Kostiakov - Lewis relation will be used for
'                 determining I (m3/sec/m) where
'                                  (a-1)
'                        I = a k τ      + f
'                                        0
' InfilP(i,1)   :  Infiltration depth at previous time step.
' kofInf        :  The coefficient k in the infiltration equation
' fsub0         :  The coefficient f  in the infiltration equation
'                                   0
' aofInf        :  The exponent a in the infiltration function
' TofOpp(i,1)   :  Time of opportunity τ in the infiltration function
'                 at individual nodes
' S0(i,1)       :  Slope of furrow or border at the individual node (So)
' ManngN        :  Manning's roughness coefficient (n)
' CoefTW        :  Top width coefficient which could either be 1.5 or 1
' Fsubi(i,1), Fsubj(i,1), and Fsubk(i,1) :  Coefficients f , f , and f
'                                                         i   j       k
'                 in the force vector at nodes i, j, and k, respectively.
'+---------------------------------------------------------------------+

GOSUB HEADING

GOSUB INITIAL

FGNam$ = "SURFDSGN.CFG"

FilIsIn% = FNExists%(FGNam$)
IF FilIsIn% = True% THEN
  OPEN "I",#2,FGNam$
  INPUT #2,IrrMethod$
  INPUT #2,Method$
  INPUT #2,TypElem$
  INPUT #2,SelPrtOpt$
  INPUT #2,iprint%
```

```
  INPUT #2,VARTheta
  INPUT #2,ALPHA
  INPUT #2,DeltaT
  INPUT #2,numiter%
  INPUT #2,AllError
  INPUT #2,TotNumStep%
  INPUT #2,CONSISorLUMP%
  INPUT #2,CoefTW
  INPUT #2,FurLength
  INPUT #2,TimCut
  INPUT #2,Qin
  INPUT #2,kofInf
  INPUT #2,fsub0
  INPUT #2,Slope
  INPUT #2,aofInf
  INPUT #2,ManngN
  INPUT #2,Rho1
  INPUT #2,Rho2
  INPUT #2,Sigma1
  INPUT #2,Sigma2
  CLOSE #2
END IF

IrrMethod$ = UCASE$(IrrMethod$)
Method$ = UCASE$(Method$)
TypElem$ = UCASE$(TypElem$)
SelPrtOpt$ = UCASE$(SelPrtOpt$)

LOCATE 8,8
PRINT "Irrigation Method  . . .  ('F', furrow, 'B', border) : ";IrrMethod$
LOCATE 10,8
PRINT "Method of Solution . . . . . (H, S, Z, or K) : ";Method$
LOCATE 12,8
PRINT "Type of Element  . . . ('L', linear, 'Q', quadratic) : ";TypElem$
LOCATE 14,8
PRINT "Level of Printing  . . . . .  (select 0, 1, 2, or 3) : ";iprint%
LOCATE 16,8
PRINT "Output Device ('S', screen, 'F', file, 'P', printer) : ";
PRINT SelPrtOpt$

Reiterate1:

SelStrg$ = "YyNn"
Strg$ = "Modify the above ('Y', yes, 'N', no) : "
RowStrg% = 20
ColStrg% = 8
call SelStrgEntry
Ans1$ = SelOpt$

Ans1$ = UCASE$(Ans1$)

IF Ans1$ = "Y" THEN
  GOSUB HEADING
  SelStrg$ = "FfBb"
  Strg$ = "Irrigation Method  . . .  ('F', furrow, 'B', border) : "
  RowStrg% = 8
  ColStrg% = 8
  call SelStrgEntry
  IrrMethod$ = SelOpt$
  SelStrg$ = "HSZKhszk"
  Strg$ = "Method of Solution  . . . . . (select H, S, Z, or K) : "
  RowStrg% = 10
  ColStrg% = 8
  call SelStrgEntry
  Method$ = SelOpt$
```

```
  SelStrg$ = "LQlq"
  Strg$ = "Type of Element  . . . ('L', linear, 'Q', quadratic) : "
  RowStrg% = 12
  ColStrg% = 8
  call SelStrgEntry
  TypElem$ = SelOpt$

  SelStrg$ = "0123"
  Strg$ = "Level of Printing . . . . . (select 0, 1, 2, or 3) : "
  RowStrg% = 14
  ColStrg% = 8
  call SelStrgEntry
  LevPrt$ = SelOpt$
  iprint% = VAL(LevPrt$)

  SelStrg$ = "PpSsFf"
  Strg$ = "Output Device ('S', screen, 'F', file, 'P', printer) : "
  RowStrg% = 16
  ColStrg% = 8
  call SelStrgEntry
  SelPrtOpt$ = SelOpt$

  GOTO Reiterate1
end if

IrrMethod$ = UCASE$(IrrMethod$)
Method$ = UCASE$(Method$)
TypElem$ = UCASE$(TypElem$)
SelPrtOpt$ = UCASE$(SelPrtOpt$)

IF SelPrtOpt$ = "S" THEN
  OPEN "SCRN:" FOR OUTPUT AS #1
ELSEIF SelPrtOpt$ = "P" THEN
  OPEN "LPT1:" FOR OUTPUT AS #1
ELSEIF SelPrtOpt$ = "F" THEN
  OPEN "SURFDSGN.OUT" FOR OUTPUT AS #1
END IF

GOSUB HEADING

LOCATE 9,14
PRINT "Time Step, δT . . . . . (min) = ";DeltaT
LOCATE 10,14
PRINT "Maximum Number of Iterations  . . = ";numiter%
LOCATE 11,14
PRINT "Allowable Error . . . . . = ";
PRINT USING "#.#######";AllError
LOCATE 12,14
PRINT "Maximum Number of time Steps  . . = ";TotNumStep%
LOCATE 13,14
PRINT "Time Weighting Coefficient, Θ . . = ";VARTheta
LOCATE 14,14
PRINT "α (Select α=0 for Galerkin) . . . = ";ALPHA
LOCATE 15,14
PRINT "Top Width Coefficient  (1 or 1.5) = ";CoefTW
LOCATE 16,14
PRINT "Consistent (1) or Lumped (0)  . . = ";CONSISorLUMP%

Reiterate2:

SelStrg$ = "YyNn"
Strg$ = "Modify the above ('Y', yes, 'N', no) : "
RowStrg% = 21
ColStrg% = 14
call SelStrgEntry
Ans2$ = SelOpt$

Ans2$ = UCASE$(Ans2$)
```

```
IF Ans2$ = "Y" THEN
  GOSUB HEADING
  LOCATE 9,14
  PRINT "Time Step, δT . . . . . (min) = ";
  INPUT "",DeltaT
  LOCATE 10,14
  PRINT "Maximum Number of Iterations  . . = ";
  INPUT "",numiter%
  LOCATE 11,14
  PRINT "Allowable Error . . . . . = ";
  INPUT "",AllError
  LOCATE 12,14
  PRINT "Maximum Number of Time Steps  . . = ";
  INPUT "",TotNumStep%
  LOCATE 13,14
  PRINT "Time Weighting Coefficient, Θ . . = ";
  INPUT "",VARTheta
  LOCATE 14,14
  PRINT "α (Select α=0 for Galerkin) . . . = ";
  INPUT "",ALPHA
  LOCATE 15,14
  PRINT "Top Width Coefficient  (1 or 1.5) = ";
  INPUT "",CoefTW
  LOCATE 16,14
  PRINT "Consistent (1) or Lumped (0)  . . = ";
  INPUT "",CONSISorLUMP%
  GOTO Reiterate2
END IF

freqprint% = 1
Gravity = 9.81

GOSUB HEADING

LOCATE 8,8
PRINT "Furrow Length  . . . . . (m) = ";FurLength
LOCATE 9,8
PRINT "Time of Cutoff . . . . . (min) = ";TimCut
LOCATE 10,8
PRINT "Inlet Flow Rate  . . . . . (liters/sec) = ";Qin
LOCATE 11,8
PRINT "Slope of Channel Bed . . . . . (fraction) = ";Slope
LOCATE 12,8
PRINT "Manning Roughness Coefficient, n . . . . . = ";ManngN
LOCATE 13,8
PRINT "Flow Geometry Parameter, σ1  . . . . . = ";Sigma1
LOCATE 14,8
PRINT "Flow Geometry Parameter, σ2  . . . . . = ";Sigma2
LOCATE 15,8
PRINT "Hydraulic Section Param., Rho 1  . . . . . = ";Rho1
LOCATE 16,8
PRINT "Hydraulic Section Param., Rho 2  . . . . . = ";Rho2
LOCATE 17,8
PRINT "Infiltration Function Coeff., k (m^3/m/min^a)= ";kofInf
LOCATE 18,8
PRINT "Infiltration Func. Exponent, a . . . . . = ";aofInf
LOCATE 19,8
PRINT "Infiltration Function Coeff., f  (m^3/m/min) = ";fsub0

Reiterate3:

SelStrg$ = "YyNn"
Strg$ = "Modify the above ('Y', yes, 'N', no) : "
RowStrg% = 21
ColStrg% = 8
call SelStrgEntry
Ans3$ = SelOpt$
```

```
Ans3$ = UCASE$(Ans3$)

IF Ans3$ = "Y" THEN
  GOSUB HEADING
  LOCATE 8,8
  PRINT "Furrow Length  . . . . . . . . . . . . . . (m) = ";
  INPUT "",FurLength
  LOCATE 9,8
  PRINT "Time of Cutoff . . . . . . . . . . . . . (min) = ";
  INPUT "",TimCut
  LOCATE 10,8
  PRINT "Inlet Flow Rate  . . . . . . . .  (liters/sec) = ";
  INPUT "",Qin
  LOCATE 11,8
  PRINT "Slope of Channel Bed . . . . . .  (fraction) = ";
  INPUT "",Slope
  LOCATE 12,8
  PRINT "Manning Roughness Coefficient, n . . . . . . . = ";
  INPUT "",ManngN
  LOCATE 13,8
  PRINT "Flow Geometry Parameter, σ1  . . . . . . . . . = ";
  INPUT "",Sigma1
  LOCATE 14,8
  PRINT "Flow Geometry Parameter, σ2  . . . . . . . . . = ";
  INPUT "",Sigma2
  LOCATE 15,8
  PRINT "Hydraulic Section Param., Rho 1  . . . . . . . = ";
  INPUT "",Rho1
  LOCATE 16,8
  PRINT "Hydraulic Section Param., Rho 2  . . . . . . . = ";
  INPUT "",Rho2
  LOCATE 17,8
  PRINT "Infiltration Function Coeff., k (m^3/m/min^a)= ";
  INPUT "",kofInf
  LOCATE 18,8
  PRINT "Infiltration Func. Exponent, a . . . . . . . . = ";
  INPUT "",aofInf
  LOCATE 19,8
  PRINT "Infiltration Function Coeff., f  (m^3/m/min) = ";
  INPUT "",fsub0
  GOTO Reiterate3
END IF

OPEN "O",#2,FGNam$
WRITE #2,IrrMethod$
WRITE #2,Method$
WRITE #2,TypElem$
WRITE #2,SelPrtOpt$
WRITE #2,iprint%

WRITE #2,VARTheta
WRITE #2,ALPHA

WRITE #2,DeltaT
WRITE #2,numiter%
WRITE #2,AllError
WRITE #2,TotNumStep%
WRITE #2,CONSISorLUMP%
WRITE #2,CoefTW

WRITE #2,FurLength
WRITE #2,TimCut
WRITE #2,Qin
WRITE #2,kofInf
WRITE #2,fsub0
```

```
WRITE #2,Slope
WRITE #2,aofInf
WRITE #2,ManngN
WRITE #2,Rho1
WRITE #2,Rho2
WRITE #2,Sigma1
WRITE #2,Sigma2
CLOSE #2
Qin = Qin / 1000
NumStep% = 0
GOSUB HEADING
LOCATE 10,8
PRINT "Enter File Name of Advance and Recession Data (No extension) : ";
INPUT "",NGphFil$
XXX = TIMER
if iprint% >= 0 AND SelPrtOpt$ <> "S" THEN
  ATmp$(1)=STRING$ (5," ")+DATE$
  ATmp$(1)=ATmp$(1)+STRING$ (41," ")+TIME$
  ATmp$(2)=STRING$ (69,"*")
  ATmp$(3)="*"+STRING$ (67," ")+"*"
  ATmp$(4)="*"+STRING$ (25," ")+"   Output from   "+STRING$ (25," ")+"*"
  ATmp$(5)="*"+STRING$ (3," ")
  ATmp$(5)=ATmp$(5)+"FINITE ELEMENT SURFACE IRRIGATION DESIGN PROGRAM"
  ATmp$(5)=ATmp$(5)+", FE-SURFDSGN"+STRING$ (3," ")+"*"
  ATmp$(6)=ATmp$(3)
  ATmp$(7)=ATmp$(2)
  ATmp$(8)="*"+STRING$ (6," ")
  ATmp$(8)=ATmp$(8)+"Developed By : Walid H. Shayya"+STRING$ (31," ")+"*"
  ATmp$(9)="*"+STRING$ (21," ")
  ATmp$(9)=ATmp$(9)+"Department of Agricultural Engineering        *"
  ATmp$(10)="*"+STRING$ (21," ")
  ATmp$(10)=ATmp$(10)+"Michigan State University                    *"
  ATmp$(11)=ATmp$(2)
  for icnt% = 1 to 11
    print #1, ATmp$(icnt%)
  next icnt%
  print #1,""
  print #1,""
  print #1,""
end if
if iprint% >= 0 AND SelPrtOpt$ <> "S" THEN
  PRINT #1,"Output File for Recession and Advance Data . . . . . : ";NGphFil$;
  PRINT #1,".PRG"
  PRINT #1,"Irrigation Method  . . .  ('F', furrow, 'B', border) : ";IrrMethod$
  PRINT #1,"Method of Solution . . . . . (H, S, Z, or K) : ";Method$
  PRINT #1,"Type of Element  . . . ('L', linear, 'Q', quadratic) : ";TypElem$
  PRINT #1,"Level of Printing  . . . . .  (select 0, 1, 2, or 3) : ";iprint%
  PRINT #1,"Output Device ('S', screen, 'F', file, 'P', printer) : ";
  PRINT #1, SelPrtOpt$
  PRINT #1,
  PRINT #1,"Time Step, δT . . . . . (min) = ";DeltaT
  PRINT #1,"Maximum Number of Iterations  . . = ";numiter%
  PRINT #1,"Allowable Error . . . . . = ";
  PRINT #1, USING "#.#######";AllError
  PRINT #1,"Maximum Number of Time Steps  . . = ";TotNumStep%
  PRINT #1,"Time Weighting Coefficient, Θ . . = ";VARTheta
  PRINT #1,"α (Select α=0 for Galerkin) . . . = ";ALPHA
  PRINT #1,"Top Width Coefficient  (1 or 1.5) = ";CoefTW
  PRINT #1,"Consistent (1) or Lumped (0)  . . = ";CONSISorLUMP%
  PRINT #1,
  PRINT #1,"Furrow Length  . . . . . (m) = ";FurLength
  PRINT #1,"Time of Cutoff . . . . . (min) = ";TimCut
```

```
  PRINT #1,"Inlet Flow Rate  . . . . . . .  (liters/sec) = ";Qin*1000
  PRINT #1,"Slope of Channel Bed . . . . . .  (fraction) = ";Slope
  PRINT #1,"Manning Roughness Coefficient, n . . . . . = ";ManngN
  PRINT #1,"Hydraulic Section Param., σ1 . . . . . = ";Sigma1
  PRINT #1,"Hydraulic Section Param., σ2 . . . . . = ";Sigma2
  PRINT #1,"Hydraulic Section Param., Rho 1  . . . . . = ";Rho1
  PRINT #1,"Hydraulic Section Param., Rho 2  . . . . . = ";Rho2
  PRINT #1,"Infiltration Function Coeff., k (m^3/m/min^a)= ";kofInf
  PRINT #1,"Infiltration Func. Exponent, a . . . . . = ";aofInf
  PRINT #1,"Infiltration Function Coeff., f  (m^3/m/min) = ";fsub0
  PRINT #1, ""
end if

IF TypElem$ = "L" THEN
  NumElmNode% = 2
ELSE
  NumElmNode% = 3
END IF
NumBandW% = NumElmNode% * 2
NClm% = 2*NumBandW%-1

cls

GraphFil1$ = NGphFil$ + ".PRG"
GraphFil2$ = NGphFil$ + ".REC"

open "O",#2,GraphFil1$
open "O",#3,GraphFil2$

TotalTime = 0
TotalLength = 0
write #2, TotalTime,TotalLength

GOSUB GetBasicElmntMtx

if iprint% >= 2 then
  call matrixprt(ElmtCmtx(),NumElmNode2%,NumElmNode2%,". . . .  [c(e)]  . . . .
")
  call matrixprt(ElmtKmtx(),NumElmNode2%,NumElmNode2%,". . . .  [k(e)]  . . . .
")
end if

if iprint% >= 0 then
  print #1,STRING$(78,"*")
end if

IF SelPrtOpt$ = "F" or SelPrtOpt$ = "P" THEN
  GOSUB HEADING
END IF

NumElem% = 1
Phase% = 1
PhaseRec% = 0
STPExc% = 0 ' A flag that stops program execution

DO
  IF SelPrtOpt$ = "F" or SelPrtOpt$ = "P" THEN
    locate 10,14
    print "Completed Time Steps  . . . . . : ";NumStep%
    locate 12,14
    print "Completed Time of Current Simulation Run (min): ";TotalTime
  END IF

  IF Phase% = 1 AND PhaseRec% = 0 THEN
    IF Method$ = "K" OR NumElem% = 1 THEN
      NumofPhi% = 4
      NofPhi(1,1) = 0.5
      NofPhi(2,1) = 1
      NofPhi(3,1) = 2 + (NumElmNode%-2) + (NumElem%-1)*(NumElmNode%-1)
      NofPhi(4,1) = NofPhi(3,1)-0.5
    ELSE
```

```
    NumofPhi% = 3
    NofPhi(1,1) = 1
    NofPhi(2,1) = 2 + (NumElmNode%-2) + (NumElem%-1)*(NumElmNode%-1)
    NofPhi(3,1) = NofPhi(2,1)-0.5
  END IF
ELSEIF Phase% = 2 AND PhaseRec% = 0 THEN
  IF Method$ = "K" THEN
    NumofPhi% = 2
    NofPhi(1,1) = 0.5
    NofPhi(2,1) = 1
  ELSE
    NumofPhi% = 1
    NofPhi(1,1) = 1
  END IF
END IF

IF PhaseRec% = 1 THEN
  PrPhi% = 0
  IF Phase% = 1 THEN
    PrPhi% = 2
    NofPhi(1,1) = 2 + (NumElmNode%-2) + (NumElem%-1)*(NumElmNode%-1)
    NofPhi(2,1) = NofPhi(1,1)-0.5
  END IF
  NumofPhi% = NumNodRec%*2 + PrPhi%
  FOR jjt% = 1 to NumNodRec%
    NofPhi(jjt%*2-1+PrPhi%,1) = jjt%-.5
    NofPhi(jjt%*2+PrPhi%,1) = jjt%
  NEXT jjt%
END IF

IF NumElmNode% = 3 THEN
  NumNode% = NumElem%*2+1
ELSEIF NumElmNode% = 2 THEN
  NumNode% = NumElem%+1
END IF

NP% = NumNode% * 2

IF Phase% = 1 THEN
  FOR SPct% = 1 TO NumNode%
    S0(SPct%,1) = Slope
  NEXT SPct%
END IF

IF NumElmNode% = 3 THEN
  LocL% = NumElem%*4+1
  LocQ% = NumElem%*4-2
ELSEIF NumElmNode% = 2 THEN
  LocL% = NumElem%*2+1
  LocQ% = NumElem%*2
END IF

IF Phase% = 1 AND PhaseRec% = 0 THEN
  IF NumElem% = 1 THEN
    Phi(2,1) = Qin
    Phi(1,1) = ( (Phi(2,1)^2*ManngN^2) / (Rho1*S0(1,1)) )^(1/Rho2)
    PhiSet = 0.02*Phi(1,1)
    IF NumElmNode% = 3 THEN
      Phi(4,1) = Qin*VARTheta
      Phi(3,1) = ( (Phi(4,1)^2*ManngN^2) / (Rho1*S0(1,1)) )^(1/Rho2)
    END IF
  ELSE
    IF NumElmNode% = 3 THEN
      Phi(LocL%-4,1) = Phi(LocL%-8,1)
      Phi(LocL%-3,1) = Phi(LocL%-7,1)
      Phi(LocL%-2,1) = Phi(LocL%-6,1)
      Phi(LocL%-1,1) = Phi(LocL%-5,1)
    ELSE
```

```
      Phi(LocL%-2,1) = Phi(LocL%-4,1)
      Phi(LocL%-1,1) = Phi(LocL%-3,1)
    END IF
  END IF
ELSEIF Phase% = 1 AND PhaseRec% = 1 THEN
  IF NumElmNode% = 3 THEN
    Phi(LocL%-4,1) = Phi(LocL%-8,1)
    Phi(LocL%-3,1) = Phi(LocL%-7,1)
    Phi(LocL%-2,1) = Phi(LocL%-6,1)
    Phi(LocL%-1,1) = Phi(LocL%-5,1)
  ELSE
    Phi(LocL%-2,1) = Phi(LocL%-4,1)
    Phi(LocL%-1,1) = Phi(LocL%-3,1)
  END IF
END IF

IF Phase%= 1 AND PhaseRec% = 0 THEN
  FOR ICT% = 1 TO NumNode% - 1
    IF NumElmNode% = 3 AND ICT% = NumNode%-1 THEN
      TofOpp(ICT%,1) = DeltaT*VARTheta
    ELSE
      TofOpp(ICT%,1) = DeltaT + TofOpp(ICT%,1)
    END IF
  NEXT ICT%
ELSEIF Phase% = 2 AND PhaseRec% = 0 THEN
  FOR ICT% = 1 TO NumNode%
    TofOpp(ICT%,1) = DeltaT + TofOpp(ICT%,1)
  NEXT ICT%
ELSEIF PhaseRec% = 1 THEN
  FOR ICT% = NumNodRec%+1 TO NumNode%
    TofOpp(ICT%,1) = DeltaT + TofOpp(ICT%,1)
  NEXT ICT%
END IF

if iprint% >= 2 then
  PRINT #1, "Number of elements      : ";NumElem%
  PRINT #1, "Number of nodes/element : ";NumElmNode%
  PRINT #1, "Band Width              : ";NumBandW%
  PRINT #1, "Total number of nodes   : ";NumNode%
end if

iloop% = 0

FOR JTmp% = 1 TO NP%
  PrevPhi(JTmp%,1) = 0
NEXT JTmp%

DO
    iloop% = iloop% + 1
  IF SelPrtOpt$ = "F" or SelPrtOpt$ = "P" THEN
    locate 14,14
    print "Completed Iterations Within Current Time Step : ";iloop%-1
  END IF

  IF Phase% = 1 THEN
    IF NumElmNode%=3 then
      TmpI = kofInf*TofOpp(NumElem%*2-1,1)^aofInf
      TmpI = TmpI+ fsub0*TofOpp(NumElem%*2-1,1)
      TmpJ = kofInf*TofOpp(NumElem%*2,1)^aofInf
      TmpJ = TmpI+ fsub0*TofOpp(NumElem%*2,1)
      TmpJ = TmpJ + Phi(LocQ%+1,1)
    ELSE
      TmpI = kofInf*TofOpp(NumElem%,1)^aofInf
      TmpI = TmpI + fsub0*TofOpp(NumElem%,1)
      TmpI = VarTheta * TmpI + Phi(LocQ%-1,1)
    END IF
    IF NumElmNode%=3 then
      Length(NumElem%,1)=(VARTheta*Phi(LocQ%,1)*60*DeltaT)/TmpI
```

```
  ELSE
    Length(NumElem%,1)=(VARTheta*Phi(LocQ%,1)*60*DeltaT*(1-APLHA))/TmpI
  END IF
  IF Length(NumElem%,1)<0 THEN Length(NumElem%,1)=0
  PRINT #1,"Iter #";
  PRINT #1, USING "###";iloop%;
  PRINT #1,SPACE$(57);
  PRINT #1,USING"#####.### m";Length(NumElem%,1)
ELSE
  PRINT #1,"Iter #";
  PRINT #1, USING "###";iloop%
END IF

FOR ICT% = 1 TO NP%
  FOR JCT% = 1 TO NClm%
    Kmatrx(ICT%,JCT%) = 0
    Cmatrx(ICT%,JCT%) = 0
    Amatrx(ICT%,JCT%) = 0
  NEXT JCT%
  FOR JCT% = 1 TO NClm%+NumBandW%
    Amatrx(ICT%,JCT%) = 0
  NEXT JCT%
  Force(ICT%,1) = 0
NEXT ICT%

for ict% = 1 to NP%
  Pfx(ict%,1) = Phi(ict%,1)
  if Pfx(ict%,1) < 0 THEN Pfx(ict%,1)=0
next ict%

IF NumElmNode% = 3 THEN
  LocL% = NumElem%*4+1
ELSEIF NumElmNode% = 2 THEN
  LocL% = NumElem%*2+1
END IF

IF Phase% = 1 THEN
  Pfx(LocL%,1) = 0
  Pfx(LocL%+1,1) = 0
END IF

if iprint% >= 3 then
  call matrixprt(Length(),NumElem%,1,". . . .  {L}  . . . . ")
  call matrixprt(TofOpp(),NumNode%,1,". . . .  {τ}  . . . . ")
  call matrixprt(S0(),NumNode%,1,". . . .  {S0}  . . . . ")
  call matrixprt(Phi(),NP%,1,". . . .  {Phi}  . . . . ")
  call matrixprt(ForceP(),NP%,1,". . . .  {Fa}  . . . . ")
end if

CALL GetElmntMtxCoef

if iprint% >= 3 then
  call matrixprt(Fsubi(),NumElem%,1,". . . .  {Fsubi}  . . . . ")
  call matrixprt(Fsubj(),NumElem%,1,". . . .  {Fsubj}  . . . . ")
  call matrixprt(Fsubk(),NumElem%,1,". . . .  {Fsubk}  . . . . ")
  call matrixprt(TopWidth(),NumNode%,1,". . . .  {T}  . . . . ")
end if

CALL BuildGlobalMtx

if iprint% >= 3 then
  call matrixprt(Cmatrx(),NP%,NClm%,". . . .  [C]  . . . . ")
  call matrixprt(Kmatrx(),NP%,NClm%,". . . .  [K]  . . . . ")
  call matrixprt(Force(),NP%,1,". . . .  {F}  . . . . ")
  call matrixprt(Phi(),NP%,1,". . . .  [Phi]  . . . . ")
end if

GOSUB SolveTimeStep1

CALL ModfyGlobMtx
```

```
  if iprint% >= 3 then
    call matrixprt(Amatrx(),NP%,NClm%,". . . .  [A]  . . . . ")
    call matrixprt(Pmatrx(),NP%,NClm%,". . . .  [P]  . . . . ")
    call matrixprt(Fstar(),NP%,1,". . . .  (F*)  . . . . ")
  end if

  call matrixVectmult(Pmatrx(),NP%,NumBandW%,phi1(),Temp1())

  call matrixadd(temp1(),NP%,1,Fstar(),NP%,1,PFplusFs(),NP%,1)

  call GAUSSBND(Amatrx(),NP%,NumBandW%,PFplusFs(),Phi())

  if iprint% >= 0 then
    IF iloop% = 1 then
      print #1,"   A1      Q1      A2      Q2      A3";
      IF Phase% = 1 THEN
        print #1,USING "      Q3      A4      Q4   Elem### Length ";NumElem%
      ELSE
        print #1,"      Q3      A4      Q4"
      END IF
      if NumNode%>4 then
        print #1,"   A5      Q5      A6      Q6      A7";
        print #1,"      Q7      A8      Q8 ..etc."
      end if
    end if
    call VectorTprt(Phi(),NP%,1)
  end if

  DiffError = 0

  FOR JTmp% = 1 TO NP% STEP 1
    DiffError = DiffError + ABS(PrevPhi(JTmp%,1)-Phi(JTmp%,1))
    PrevPhi(JTmp%,1) = Phi(JTmp%,1)
  NEXT JTmp%
  AllowErr = AllError+numiter%*AllError/10
LOOP UNTIL (iloop%>2 AND DiffError<=AllowErr) OR (iloop%>numiter%)

TotalTime = TotalTime + DeltaT
NumStep% = NumStep% + 1
IF Phase% = 1 THEN
  TotalLength = TotalLength + Length(NumElem%,1)
  write #2, TotalTime,TotalLength
END IF

for j =1 to NP%
  Phi1(j,1) = Phi(j,1)
  ForceP(j,1) = Force(j,1)
next j

FOR ICT% = 1 TO NumNode%
  InfilP(ICT%,1) = Infil(ICT%,1)
NEXT ICT%

if iprint% >= 0 then
  print #1,"   A1      Q1      A2      Q2      ";
  print #1,"A3      Q3      A4      Q4 ==>Time Step:";
  print #1, USING "###";NumStep%
  if NumNode%>4 then
    print #1,"   A5      Q5      A6      Q6      ";
    print #1,"A7      Q7      A8      Q8  "
  end if
  print #1,STRING$(78,"+")
  PRINT #1,SPACE$(25);"Simulated Time :";TotalTime;"min"
  print #1,STRING$(78,"*")
end if
```

```
  IF Phase% = 1 THEN
    IF TotalLength<FurLength THEN
      NumElem% = NumElem% + 1
    ELSE
      Phase% = 2
      CLOSE #2
    END IF
  END IF

  IF PhaseRec% = 0 THEN
    IF TotalTime>=TimCut THEN
      PhaseRec% = 1
      NumNodRec% = NumNodRec% + 1
      Phi(1,1) = 0
      Phi(2,1) = 0
      write #3, TotalTime,TotalRec
    END IF
  END IF

  IF PhaseRec% = 1 THEN
    TotalRec = 0
    NumNodRec% = 1
    j = 2
    WHILE j <= NumElem%
      IF Phi(j*2-1,1) <=PhiSet THEN

        Phi(j*2-1,1) = 0
        Phi(j*2,1) = 0
        NumNodRec% = NumNodRec% + 1
        TotalRec = TotalRec + Length(j-1,1)
      ELSE
        j = NumElem%
      END IF
      j = j + 1
    WEND
    IF TotalRec <> 0 THEN
      write #3, TotalTime,TotalRec
    END IF
    IF NumNodRec% >= NumElem% THEN STPExc% = 1
  END IF

  IF NumStep%>=TotNumStep% THEN
    STPExc% = 1
  END IF
LOOP UNTIL STPExc% = 1

YYY = TIMER
IF SelPrtOpt$ = "F" or SelPrtOpt$ = "P" THEN
  locate 10,14
  print "Completed Time Steps  . . . . . . . . . . . . : ";NumStep%
  locate 12,14
  print "Completed Time of Current Simulation Run (min): ";TotalTime
  locate 16,14
  PRINT USING "Time of Execution for this Run :#####.## ";YYY-XXX;
  PRINT "sec"
  LOCATE 21,1
  PRINT
END IF
PRINT #1, ""
PRINT #1, USING "Time of Execution for this Run : ######.## ";YYY-XXX;
PRINT #1, "sec"
PRINT #1, USING "                                  : ######.## ";(YYY-XXX)/60;
PRINT #1, "min"
CLOSE
LOCATE 23,40
PRINT "Press any key to see next screen ..."
WHILE INKEY$ =""
```

```
WEND
GOSUB HEADING
LOCATE 10,8
SelStrg$ = "YyNn"
Strg$ = "Would you like to run the graphics routine ('Y', yes, 'N', no) ? "
RowStrg% = 10
ColStrg% = 8
call SelStrgEntry
AnsG$ = SelOpt$
AnsG$ = UCASE$(AnsG$)
IF AnsG$ = "Y" THEN
  CHAIN "SURFGRPH.EXE"
END IF
CLS
END

InitialScrn:
  REM ************************************************************************
  REM * A subroutine to display the first screen.                             *
  REM ************************************************************************
  KEY OFF
  XX%(1) = 5: XX%(2) = 23: XX%(3) = 23: XX%(4) = 22: XX%(5) = 10
  YY%(1) = 2: YY%(2) = 4: YY%(3) = 8: YY%(4) = 10: YY%(5) = 12
  XX%(6) = 17: YY%(6) = 14
  CC$(1) = CHR$(201): CC$(2) = CHR$(205): CC$(3) = CHR$(187)
  CC$(4) = CHR$(186): CC$(5) = CHR$(204): CC$(6) = CHR$(185)
  CC$(7) = CHR$(200): CC$(8) = CHR$(188)
  A1$ = CC$(1) + STRING$(54, CC$(2)) + CC$(3)
  A2$ = CC$(5) + STRING$(54, CC$(2)) + CC$(6)
  A3$ = CC$(7) + STRING$(54, CC$(2)) + CC$(8)
  AA$(1) = " FINITE ELEMENT SURFACE IRRIGATION DESIGN MODEL "
  AA$(2) = "VERSION 1.00"
  AA$(3) = "Developed by"
  AA$(4) = "Walid H. Shayya"
  AA$(5) = "Department of Agricultural Engineering"
  AA$(6) = "Michigan State University"

  CLS
  LOCATE 4, 1

  PRINT SPACE$(12); A1$
  FOR Iloop% = 1 TO 5
    PRINT SPACE$(12); CC$(4); SPACE$(54); CC$(4)
  NEXT Iloop%
  PRINT SPACE$(12); A2$
  FOR Iloop% = 1 TO 9
    PRINT SPACE$(12); CC$(4); SPACE$(54); CC$(4)
  NEXT Iloop%
  PRINT SPACE$(12); A3$
  FOR Iloop% = 1 TO 6
    LOCATE YY%(Iloop%) + 4, XX%(Iloop%) + 12
    PRINT AA$(Iloop%)
  NEXT Iloop%

  LOCATE 23, 48
  COLOR 15, 0
  PRINT "Press any key to continue."
  COLOR 7, 0

  WHILE INKEY$ = "": WEND
  RETURN

HEADING:
  rem ************************************************************************
  rem *   A subroutine to print page heading.                                  *
  rem ************************************************************************
  cls
  LOCATE 2,24
```

```
  PRINT "FINITE ELEMENT SURFACE IRRIGATION"
  LOCATE 3,24
  PRINT "          DESIGN PROGRAM"
  LOCATE 4,23
  PRINT STRING$(35,196)
  return

SUB SelStrgEntry
  rem ************************************************************************
  rem *   A subprogram for entering one charcter input to a selected   *
  rem * string variable.                                                     *
  rem ************************************************************************
  SHARED SelOpt$,SelStrg$,Strg$,RowStrg%,ColStrg%
  LOCAL BCount%

  BCount% = 0
  SelOpt$ = ""
  WHILE INSTR(1,SelOpt$,ANY SelStrg$) = 0
    locate RowStrg%,ColStrg%
    IF BCount% <> 0 THEN BEEP
    PRINT Strg$;
    SelOpt$ = INPUT$(1)
    PRINT SelOpt$
    BCount% = 1
  WEND
  END SUB


SolveTimeStep1:
  rem ************************************************************************
  rem *  A subroutine to build the ordinary differential equation in   *
  rem *  time.  It will construct the following system of equations:   *
  rem *                                                                *
  rem *                 [A] {φ}  =  [P] {φ}  +  {F*}                   *
  rem *                       b           a                            *
  rem *  where,                                                        *
  rem *                         [A]  =  [C] + Θ.δT.[K]                 *
  rem *                         [P]  =  [C] - (1 - Θ).δT.[K]           *
  rem *                         {F*} =  δT.(1 - Θ){F} + δT.Θ.{F}       *
  rem *                                            a          b       *
  rem ************************************************************************

  Param1 = VARTheta * DeltaT * 60
  Param2 = (VARTheta - 1) * DeltaT * 60
  Param3 = - Param2
  Param1M = 1
  Param2M = 0
  Param3M = 0

  IF Method$ = "H" THEN
    call matrixNumult(Kmatrx(),NP%,NClm%,Param1,temp1())
  else
    call matrixNumultMod(Kmatrx(),NP%,NClm%,Param1,Param1M,temp1())
  end if

  if iprint% >= 3 then
    call matrixprt(temp1(),NP%,NClm%,". . . . .  Θ.δT.[K]  . . . . . ")
  end if

  call matrixadd(temp1(),NP%,NClm%,Cmatrx(),NP%,NClm%,Amatrx(),NP%,NClm%)
  if iprint% >= 3 then
    call matrixprt(Amatrx(),NP%,NClm%,". .  [A] = [C] + Θ.δT.[K]  . .")
  end if
```

```
  IF Method$ = "H" THEN
    call matrixNumult(Kmatrx(),NP%,NClm%,Param2,templ())
  else
    call matrixNumultMod(Kmatrx(),NP%,NClm%,Param2,Param2M,templ())
  end if
  if iprint% >= 3 then
    call matrixprt(templ(),NP%,NClm%,". . . .  - (1 - Θ).δT.[K]  . . . . ")
  end if

  call matrixadd(templ(),NP%,NClm%,Cmatrx(),NP%,NClm%,Pmatrx(),NP%,NClm%)
  if iprint% >= 3 then
    call matrixprt(Pmatrx(),NP%,NClm%,". .  [P] = [C] - (1 - Θ).δT.[K]  . .")
  end if

  IF Method$ = "H" THEN
    call matrixNumult(ForceP(),NP%,1,Param3,TempF1())
  else
    call matrixNumultMod(ForceP(),NP%,1,Param3,Param3M,TempF1())
  end if

  if iprint% >= 3 then
    call matrixprt(TempF1(),NP%,1,". . . .  (1 - Θ).δT.{F}a  . . . . ")
  end if

  IF Method$ = "H" THEN
    call matrixNumult(Force(),NP%,1,Param1,TempF2())
  else
    call matrixNumultMod(Force(),NP%,1,Param1,Param1M,TempF2())
  end if

  if iprint% >= 3 then
    call matrixprt(TempF2(),NP%,1,". . . .  Θ.δT.{F}b  . . . . ")
  end if

  call matrixadd(TempF1(),NP%,1,TempF2(),NP%,1,Fstar(),NP%,1)
  if iprint% >= 3 then
    call matrixprt(Fstar(),NP%,1,". {F*} = δT.(1-Θ){Fa} + δT.Θ.{Fb} .")
  end if
return

GetBasicElmntMtx:
  rem ************************************************************************
  rem *    A subroutine for getting the constant coefficients of the   *
  rem * element stiffness and the capacitance matrices.  This          *
  rem * subroutine works for both linear and quadratic elements.       *
  rem ************************************************************************

  NumElmNode2% = NumElmNode% * 2

  IF NumElmNode2% = 6 THEN
    IF CONSISorLUMP% = 1 THEN
      RESTORE 100
      FOR Icount% = 1 TO NumElmNode2%
        FOR Jcount% = 1 TO NumElmNode2%
          READ ElmtCmtx(Icount%,Jcount%)
        NEXT Jcount%
      NEXT Icount%
    ELSE
      RESTORE 125
      FOR Icount% = 1 TO NumElmNode2%
        FOR Jcount% = 1 TO NumElmNode2%
          READ ElmtCmtx(Icount%,Jcount%)
        NEXT Jcount%
      NEXT Icount%
    END IF
    RESTORE 150
    FOR Icount% = 1 TO NumElmNode2%
      FOR Jcount% = 1 TO NumElmNode2%
        READ ElmtKmtx(Icount%,Jcount%)
```

```
      ElmtKmtx(Icount%,Jcount%) = ElmtKmtx(Icount%,Jcount%) / 24
    NEXT Jcount%
  NEXT Icount%
ELSEIF NumElmNode2% = 4 THEN
  IF CONSISorLUMP% = 1 THEN
    RESTORE 200
    FOR Icount% = 1 TO NumElmNode2%
      FOR Jcount% = 1 TO NumElmNode2%
        READ ElmtCmtx(Icount%,Jcount%)
      NEXT Jcount%
    NEXT Icount%
  ELSE
    RESTORE 225
    FOR Icount% = 1 TO NumElmNode2%
      FOR Jcount% = 1 TO NumElmNode2%
        READ ElmtCmtx(Icount%,Jcount%)
      NEXT Jcount%
    NEXT Icount%
  END IF
  RESTORE 250
  FOR Icount% = 1 TO NumElmNode2%
    FOR Jcount% = 1 TO NumElmNode2%
      READ ElmtKmtx(Icount%,Jcount%)
    NEXT Jcount%
  NEXT Icount%
  REM ** Non-Galerkin Finite Element Formulation except if ALPHA=0
  ElmtKmtx(1,2) = ElmtKmtx(1,2) + ALPHA / 2
  ElmtKmtx(1,4) = ElmtKmtx(1,4) - ALPHA / 2
  ElmtKmtx(3,2) = ElmtKmtx(3,2) - ALPHA / 2
  ElmtKmtx(3,4) = ElmtKmtx(3,4) + ALPHA / 2
END IF

100 DATA  4, 0, 2, 0,-1, 0
    DATA  0, 0, 0, 0, 0, 0
    DATA  2, 0,16, 0, 2, 0
    DATA  0, 0, 0, 0, 0, 0
    DATA -1, 0, 2, 0, 4, 0
    DATA  0, 0, 0, 0, 0, 0

125 DATA  5, 0, 0, 0, 0, 0
    DATA  0, 0, 0, 0, 0, 0
    DATA  0, 0,20, 0, 0, 0
    DATA  0, 0, 0, 0, 0, 0
    DATA  0, 0, 0, 0, 5, 0
    DATA  0, 0, 0, 0, 0, 0

150 DATA  0,-12,  0, 16,  0, -4
    DATA  0,  0,  0,  0,  0,  0
    DATA  0,-16,  0,  0,  0, 16
    DATA  0,  0,  0,  0,  0,  0
    DATA  0,  4,  0,-16,  0, 12
    DATA  0,  0,  0,  0,  0,  0

200 DATA  2, 0, 1, 0
    DATA  0, 0, 0, 0
    DATA  1, 0, 2, 0
    DATA  0, 0, 0, 0

225 DATA  3, 0, 0, 0
    DATA  0, 0, 0, 0
    DATA  0, 0, 3, 0
    DATA  0, 0, 0, 0

250 DATA  0,-.5,  0, .5
    DATA  0,  0,  0,  0
    DATA  0,-.5,  0, .5
    DATA  0,  0,  0,  0
RETURN
```

```
sub GetElmntMtxCoef
  rem ************************************************************************
  rem *    A subroutine for determining the coefficients C, C2, C3, and*
  rem * C4 of the element stiffness and the capacitance matrices. Also,*
  rem * the coefficients fi, fj, and fk for the force vector are       *
  rem * determined.  This subroutine works for both linear and         *
  rem * quadratic elements.                                           *
  rem ************************************************************************
  SHARED NumElem%,Gravity,Phi(),NumElmNode%,Method$,Sigma1,Sigma2
  SHARED NumNode%,Fsubi(),Fsubj(),Fsubk(),ManngN,S0(),Rho1,Rho2,Length()
  SHARED TopWidth(),Coef1(),Coef2(),Coef3(),Coef4(),Coef5(),CONSISorLUMP%
  SHARED Pfx(),Czi(),Czj(),Czk(),CoefTW
  LOCAL Count%,AreaC%,FlowC%,TempV1,TempV2,TempV3,Sfi,Sfj,Sfk,NodeC%,itr%
  LOCAL ct%,Expnt
  IF NumElmNode% = 2 THEN
    TempV1 = 2
    TempV2 = 2
    TempV3 = 2
  ELSEIF NumElmNode% = 3 THEN
    TempV1 = 6
    TempV2 = 2
    TempV3 = 4
  END IF
  IF Method$ = "Z" THEN
    Expnt = 1
  ELSE
    Expnt = 2
  END IF
  FOR Count% = 1 TO NumElem%
    REM ************************************************************************
    REM * Determine the coefficients c , c , c , c , c , f , f , & f *
    REM *                              1   2   3   4   5   i   j     k*
    REM * of the individual element matrices for all the methods.    *
    REM ************************************************************************
    IF NumElmNode% = 2 THEN
      AreaC% = Count%*2-1
      FlowC% = Count%*2
      NodeC% = Count%
    ELSEIF NumElmNode% = 3 THEN
      AreaC% = Count%*4-3
      FlowC% = Count%*4-2
      NodeC% = Count%*2-1
    END IF
    IF Method$ <> "K" THEN
      REM ************************************************************************
      REM * If kinematic wave, the force vector of the 2nd, 4th, &   *
      rem * 6th equations is placed in the stiffness matrix.        *
      rem ************************************************************************
      FOR itr% = 1 TO NumElmNode%
        ct% = (itr%-1)*2
        IF Method$ = "H" AND Pfx(AreaC%+ct%,1)<>0 THEN
          Coef1(Count%,itr%) = 1 / (Gravity * Pfx(AreaC%+ct%,1))
        ELSE
          Coef1(Count%,itr%) = 0
        END IF
        IF (Method$ = "H" OR Method$ = "S") AND (Pfx(AreaC%+ct%,1)<>0) THEN
          Coef3(Count%,itr%)=Pfx(FlowC%+ct%,1)^2/(Gravity*Pfx(AreaC%+ct%,1)^3)
        ELSE
          Coef3(Count%,itr%) = 0
        END IF
        IF Pfx(AreaC%+ct%,1)<>0 THEN
          TopWidth(NodeC%+itr%-1,1)=(CoefTW*Pfx(AreaC%+ct%,1)^(1-Sigma2))/Sigma1
          Coef2(Count%,itr%) = 1 / TopWidth(NodeC%+itr%-1,1)
```

```
        ELSE
          TopWidth(NodeC%+itr%-1,1) = 0
          Coef2(Count%,itr%) = 0
        END IF
        IF Pfx(AreaC%+ct%,1)<>0 AND Method$ <> "Z" THEN
          Coef4(Count%,itr%)=2*Pfx(FlowC%+ct%,1)/(Gravity*Pfx(AreaC%+ct%,1)^2)
        ELSE
          Coef4(Count%,itr%) = 0
        END IF
        Coef5(Count%,itr%)=Coef2(Count%,itr%)-Coef3(Count%,itr%)
      NEXT itr%
      IF Pfx(AreaC%,1) <> 0 THEN
        Sfi = (Pfx(FlowC%,1)^Expnt*ManngN^2)/(Rho1*(Pfx(AreaC%,1))^Rho2)
      ELSE
        Sfi = 0
      END IF

      IF Pfx(AreaC%+2,1) <> 0 THEN
        Sfj = (Pfx(FlowC%+2,1)^Expnt*ManngN^2)/(Rho1*(Pfx(AreaC%+2,1))^Rho2)
      ELSE
        Sfj = 0
      END IF
      IF Method$ = "Z" THEN
        Czi(Count%,1) = Sfi * Length(Count%,1)/TempV1
        Czj(Count%,1) = Sfj * Length(Count%,1)/TempV3
        Sfi = 0
        Sfj = 0
      END IF
      Fsubi(Count%,1) = Length(Count%,1)*(S0(NodeC%,1)-Sfi)/TempV1
      Fsubj(Count%,1) = Length(Count%,1)*(S0(NodeC%+1,1)-Sfj)/TempV1
      IF NumElmNode% = 3 THEN
        IF Pfx(AreaC%+4,1) <> 0 THEN
          Sfk = (Pfx(FlowC%+4,1)^Expnt*ManngN^2)/(Rho1*(Pfx(AreaC%+4,1))^Rho2)
        ELSE
          Sfk = 0
        END IF
        IF Method$ = "Z" THEN
          Czk(Count%,1) = Sfk * Length(Count%,1)/TempV1
          Sfk = 0
        END IF
        Fsubk(Count%,1) = Length(Count%,1)*(S0(NodeC%+2,1)-Sfk)/TempV1
      END IF
    ELSE
      Sfi = Rho1*(Pfx(AreaC%,1))^(Rho2-2)*S0(NodeC%,1)
      Sfj = Rho1*(Pfx(AreaC%+2,1))^(Rho2-2)*S0(NodeC%+1,1)
      Fsubi(Count%,1) = (Sfi^.5)/(ManngN*TempV2)
      Fsubj(Count%,1) = (Sfj^.5)/(ManngN*TempV3)
      IF NumElmNode% = 3 THEN
        Sfk = Rho1*(Pfx(AreaC%+4,1))^(Rho2-2)*S0(NodeC%+2,1)
        Fsubk(Count%,1) = (Sfk^.5)/(ManngN*TempV2)
      END IF
    END IF
  NEXT Count%
  END SUB

sub BuildGlobalMtx
  rem ************************************************************************
  rem *    A subroutine for building the global stiffness and                *
  rem * capacitance matrices, and the global force vector.                   *
  rem ************************************************************************
  SHARED NumElem%,Coef(),NumElmNode%,Kmatrx(),Cmatrx(),ElmtCmtx(),Force()
  SHARED Length(),ElmtKmtx(),TempK(),TempC(),Method$,ManngN,Infil()
  SHARED kofInf,fsub0,aofInf,TofOpp(),Fsubi(),Fsubj(),Fsubk(),CONSISorLUMP%
```

```
SHARED Coef1(),Coef2(),Coef3(),Coef4(),Coef5(),DeltaT,FDorFE%,InfilP()
SHARED Czi(),Czj(),Czk(),NumBandW%
SHARED ALPHA,Phase%

LOCAL NumElmNode2%,Count%,Count2%,LocatC%,LocatN%,Ict%,Jct%,TempV1
LOCAL TempV2,TempV3,TempV4,TempV5

NumElmNode2% = NumElmNode% * 2

IF NumElmNode% = 2 THEN
  TempV1 = 2
ELSEIF NumElmNode% = 3 THEN
  TempV1 = 6
END IF

IF NumElem% = 1 OR Phase%<>1 THEN
  UpLmt% = NumElem%
ELSE
  UpLmt% = NumElem% - 1
END IF

FOR Count% = 1 TO UpLmt%
  REM ************************************************************************
  REM * Construct the individual element matrices for all the                *
  REM * different models.                                                    *
  REM ************************************************************************
  IF NumElmNode2% = 6 THEN
    LocatC% = Count%*4-4
    LocatN% = Count%*2-2
  ELSEIF NumElmNode2% = 4 THEN
    LocatC% = Count%*2-2
    LocatN% = Count%-1
  END IF
  FOR Ict% = 1 TO NumElmNode2%
    FOR Jct% = 1 TO NumElmNode2%
      TempK(Ict%,Jct%) = ElmtKmtx(Ict%,Jct%)
      TempC(Ict%,Jct%) = ElmtCmtx(Ict%,Jct%)
    NEXT Jct%
  NEXT Ict%
  IF Method$ = "H" THEN
    IF CONSISorLUMP% = 1 THEN
      IF FDorFE%=1 THEN
        IF NumElmNode2% = 6 THEN
          TempC(2,2) = 4 * Coef1(Count%,1)
          TempC(2,4) = 2 * Coef1(Count%,2)
          TempC(2,6) = - Coef1(Count%,3)
          TempC(4,2) = 2 * Coef1(Count%,1)
          TempC(4,4) = 16 * Coef1(Count%,2)
          TempC(4,6) = 2 * Coef1(Count%,3)
          TempC(6,2) = - Coef1(Count%,1)
          TempC(6,4) = 2 * Coef1(Count%,2)
          TempC(6,6) = 4 * Coef1(Count%,3)
        ELSEIF NumElmNode2% = 4 THEN
          TempC(2,2) = 2 * Coef1(Count%,1)
          TempC(2,4) = Coef1(Count%,2)
          TempC(4,2) = Coef1(Count%,1)
          TempC(4,4) = 2 * Coef1(Count%,2)
        END IF
      ELSE
        IF NumElmNode2% = 6 THEN
          TempC(2,2) = 4 * Coef1(Count%,1)
          TempC(2,4) = 2 * Coef1(Count%,1)
          TempC(2,6) = - Coef1(Count%,1)
          TempC(4,2) = 2 * Coef1(Count%,1)
          TempC(4,4) = 16 * Coef1(Count%,1)
          TempC(4,6) = 2 * Coef1(Count%,1)
          TempC(6,2) = - Coef1(Count%,1)
```

```
        TempC(6,4) = 2 * Coef1(Count%,1)
        TempC(6,6) = 4 * Coef1(Count%,1)
      ELSEIF NumElmNode2% = 4 THEN
        TempC(2,2) = 2 * Coef1(Count%,1)
        TempC(2,4) = Coef1(Count%,1)
        TempC(4,2) = Coef1(Count%,1)
        TempC(4,4) = 2 * Coef1(Count%,1)
      END IF
    END IF
  ELSE
    IF FDorFE%=1 THEN
      IF NumElmNode2% = 6 THEN
        TempC(2,2) = 5 * Coef1(Count%,1)
        TempC(4,4) = 20 * Coef1(Count%,2)
        TempC(6,6) = 5 * Coef1(Count%,3)
      ELSEIF NumElmNode2% = 4 THEN
        TempC(2,2) = 3 * Coef1(Count%,1)
        TempC(4,4) = 3 * Coef1(Count%,2)
      END IF
    ELSE
      IF NumElmNode2% = 6 THEN
        TempC(2,2) = 5 * Coef1(Count%,1)
        TempC(4,4) = 20 * Coef1(Count%,1)
        TempC(6,6) = 5 * Coef1(Count%,1)
      ELSEIF NumElmNode2% = 4 THEN
        TempC(2,2) = 3 * Coef1(Count%,1)
        TempC(4,4) = 3 * Coef1(Count%,1)
      END IF
    END IF
  END IF
END IF
IF Method$ = "K" THEN
  IF NumElmNode2% = 6 THEN
    REM ************************************************************
    REM * If kinematic wave, the force vector of the 2nd, 4th, & *
    rem * 6th equations is placed in the stiffness matrix.       *
    rem ************************************************************
    TempK(2,2) = .5
    TempK(2,1) = -Fsubi(COUNT%,1)
    TempK(4,4) = 1
    TempK(4,3) = -4*Fsubj(COUNT%,1)
    TempK(6,6) = .5
    TempK(6,5) = -Fsubk(COUNT%,1)
  ELSEIF NumElmNode2% = 4 THEN
    TempK(2,2) = .5
    TempK(2,1) = -Fsubi(COUNT%,1)
    TempK(4,4) = .5
    TempK(4,3) = -Fsubj(COUNT%,1)
  END IF
ELSEIF Method$ = "Z" THEN
  IF FDorFE%=1 THEN
    IF NumElmNode2% = 6 THEN
      TempK(2,1) = - Coef2(Count%,1) / 2
      TempK(2,3) = 2 * Coef2(Count%,2) / 3
      TempK(2,5) = - Coef2(Count%,3) / 6
      TempK(4,1) = - 2 * Coef2(Count%,1) / 3
      TempK(4,5) = 2 * Coef2(Count%,2) / 3
      TempK(6,1) = Coef2(Count%,1) / 6
      TempK(6,3) = - 2 * Coef2(Count%,2) / 3
      TempK(6,5) = Coef2(Count%,3) / 2
    ELSEIF NumElmNode2% = 4 THEN
      TempK(2,1) = Coef2(Count%,1) * (-.5 + ALPHA/2)
      TempK(2,3) = Coef2(Count%,2) * (.5 - ALPHA/2)
      TempK(4,1) = Coef2(Count%,1) * (-.5 - ALPHA/2)
```

```
      TempK(4,3) = Coef2(Count%,2) * (.5 + ALPHA/2)
    END IF
  ELSE
    IF NumElmNode2% = 6 THEN
      TempK(2,1) = - Coef2(Count%,1) / 2
      TempK(2,3) = 2 * Coef2(Count%,1) / 3
      TempK(2,5) = - Coef2(Count%,1) / 6
      TempK(4,1) = - 2 * Coef2(Count%,1) / 3
      TempK(4,5) = 2 * Coef2(Count%,1) / 3
      TempK(6,1) = Coef2(Count%,1) / 6
      TempK(6,3) = - 2 * Coef2(Count%,1) / 3
      TempK(6,5) = Coef2(Count%,1) / 2
    ELSEIF NumElmNode2% = 4 THEN
      TempK(2,1) = Coef2(Count%,1) * (-.5 + ALPHA/2)
      TempK(2,3) = Coef2(Count%,1) * ( .5 - ALPHA/2)
      TempK(4,1) = Coef2(Count%,1) * (-.5 - ALPHA/2)
      TempK(4,3) = Coef2(Count%,1) * ( .5 + ALPHA/2)
    END IF
  END IF
  IF NumElmNode2% = 6 THEN
    TempK(2,2) = Czi(COUNT%,1)
    TempK(4,4) = 4*Czj(COUNT%,1)
    TempK(6,6) = Czk(COUNT%,1)
  ELSEIF NumElmNode2% = 4 THEN
    TempK(2,2) = Czi(COUNT%,1)
    TempK(4,4) = Czj(COUNT%,1)
  END IF
ELSE
  IF FDorFE%=1 THEN
    IF NumElmNode2% = 6 THEN
      TempK(2,1) = - Coef5(Count%,1) / 2
      TempK(2,2) = - Coef4(Count%,1) / 2
      TempK(2,3) = 2 * Coef5(Count%,2) / 3
      TempK(2,4) = 2 * Coef4(Count%,2) / 3
      TempK(2,5) = - Coef5(Count%,3) / 6
      TempK(2,6) = - Coef4(Count%,3) / 6
      TempK(4,1) = - 2 * Coef5(Count%,1) / 3
      TempK(4,2) = - 2 * Coef4(Count%,1) / 3
      TempK(4,5) = 2 * Coef5(Count%,2) / 3
      TempK(4,6) = 2 * Coef4(Count%,2) / 3
      TempK(6,1) = Coef5(Count%,1) / 6
      TempK(6,2) = Coef4(Count%,1) / 6
      TempK(6,3) = - 2 * Coef5(Count%,2) / 3
      TempK(6,4) = - 2 * Coef4(Count%,2) / 3
      TempK(6,5) = Coef5(Count%,3) / 2
      TempK(6,6) = Coef4(Count%,3) / 2
    ELSEIF NumElmNode2% = 4 THEN
      TempK(2,1) = Coef5(Count%,1) * (-.5 + ALPHA/2)
      TempK(2,2) = Coef4(Count%,1) * (-.5 + ALPHA/2)
      TempK(2,3) = Coef5(Count%,2) * (.5 - ALPHA/2)
      TempK(2,4) = Coef4(Count%,2) * (.5 - ALPHA/2)
      TempK(4,1) = Coef5(Count%,1) * (-.5 - ALPHA/2)
      TempK(4,2) = Coef4(Count%,1) * (-.5 - ALPHA/2)
      TempK(4,3) = Coef5(Count%,2) * (.5 + ALPHA/2)
      TempK(4,4) = Coef4(Count%,2) * (.5 + ALPHA/2)
    END IF
  ELSE
    IF NumElmNode2% = 6 THEN
      TempK(2,1) = - Coef5(Count%,1) / 2
      TempK(2,2) = - Coef4(Count%,1) / 2
      TempK(2,3) = 2 * Coef5(Count%,1) / 3
      TempK(2,4) = 2 * Coef4(Count%,1) / 3
      TempK(2,5) = - Coef5(Count%,1) / 6
      TempK(2,6) = - Coef4(Count%,1) / 6
```

```
        TempK(4,1) = - 2 * Coef5(Count%,1) / 3
        TempK(4,2) = - 2 * Coef4(Count%,1) / 3
        TempK(4,5) = 2 * Coef5(Count%,1) / 3
        TempK(4,6) = 2 * Coef4(Count%,1) / 3
        TempK(6,1) = Coef5(Count%,1) / 6
        TempK(6,2) = Coef4(Count%,1) / 6
        TempK(6,3) = - 2 * Coef5(Count%,1) / 3
        TempK(6,4) = - 2 * Coef4(Count%,1) / 3
        TempK(6,5) = Coef5(Count%,1) / 2
        TempK(6,6) = Coef4(Count%,1) / 2
      ELSEIF NumElmNode2% = 4 THEN
        TempK(2,1) = Coef5(Count%,1) * (-.5 + ALPHA/2)
        TempK(2,2) = Coef4(Count%,1) * (-.5 + ALPHA/2)
        TempK(2,3) = Coef5(Count%,1) * ( .5 - ALPHA/2)
        TempK(2,4) = Coef4(Count%,1) * ( .5 - ALPHA/2)
        TempK(4,1) = Coef5(Count%,1) * (-.5 - ALPHA/2)
        TempK(4,2) = Coef4(Count%,1) * (-.5 - ALPHA/2)
        TempK(4,3) = Coef5(Count%,1) * ( .5 + ALPHA/2)
        TempK(4,4) = Coef4(Count%,1) * ( .5 + ALPHA/2)
      END IF
    END IF
  END IF

  REM ************************************************************************
  REM * Construct the global stiffness and capacitance matrices, and *
  REM * build the global force vector.                                       *
  REM ************************************************************************
  FOR Ict% = 1 TO NumElmNode2%
    FOR Jct% = 1 TO NumElmNode2%
      IRow% = Ict%+LocatC%
      JCol% = Jct%+LocatC%+NumBandW%-IRow%
      Kmatrx(IRow%,JCol%) = TempK(Ict%,Jct%) + Kmatrx(IRow%,JCol%)
      IF TempC(Ict%,Jct%) <> 0 THEN
        IF NumElmNode2% = 6 THEN
          TempV2 = TempC(Ict%,Jct%) * Length(Count%,1) / 30
        ELSEIF NumElmNode2% = 4 THEN
          TempV2 = TempC(Ict%,Jct%) * Length(Count%,1) / 6
        END IF
        Cmatrx(IRow%,JCol%) = TempV2 + Cmatrx(IRow%,JCol%)
      END IF
    NEXT Jct%
  NEXT Ict%
  FOR COUNT2% = 1 TO NumElmNode%
    TempV3=kofInf*TofOpp(LocatN%+COUNT2%,1)^aofInf
    TempV3=(TempV3+TofOpp(LocatN%+COUNT2%,1)*fsub0)
    Infil(LocatN%+COUNT2%,1) = TempV3
    TempV4=(TempV3-InfilP(LocatN%+COUNT2%,1))/(DeltaT*60)
    TempV5 = - Length(Count%,1)*TempV4/TempV1
    If COUNT2% = 2 AND NumElmNode% = 3 THEN
      TempV5 = TempV5 * 4
    END If
    Force(LocatC%+Count2%*2-1,1)=TempV5+Force(LocatC%+Count2%*2-1,1)
  NEXT COUNT2%
  IF Method$ <> "K" THEN
    Force(LocatC%+2,1) = Force(LocatC%+2,1)+Fsubi(COUNT%,1)
    IF NumElmNode% = 2 THEN
      Force(LocatC%+4,1) = Force(LocatC%+4,1)+Fsubj(COUNT%,1)
    ELSEIF NumElmNode% = 3 THEN
      Force(LocatC%+4,1) = Force(LocatC%+4,1)+4*Fsubj(COUNT%,1)
      Force(LocatC%+6,1) = Force(LocatC%+6,1)+Fsubk(COUNT%,1)
    END IF
  END IF
NEXT Count%
END SUB
```

```
sub ModfyGlobMtx
  rem ************************************************************************
  rem *    A subroutine for modifying the global  system of equations  *
  rem * for known Phi boundary conditions.  This subroutine modifies   *
  rem * the [A] and [P] matrices by deleting rows and columns.  It is  *
  rem * always assumed that the unknown to be modified is the flow Q   *
  rem * which represents the second unknown at each node.              *
  rem * However, if the Area of flow at the node has a fixed value as  *
  rem * a boundary condition, then this can still be incorporated into *
  rem * the final system of equations using this subroutine.  This step*
  rem * is accomplished by entering the number of node minus 0.5       *
  rem * as the node number to be modified.  In other words, the        *
  rem * first value in the array NofPhi(i,1) will be                   *
  rem *         NofPhi(1,1) = Node Number of 1st boundary - .5         *
  rem ************************************************************************
  SHARED NumElem%,Amatrx(),Pmatrx(),Method$,NumNode%,Phi(),Phi1()
  SHARED NumofPhi%,NumBandW%,Fstar(),NP%,NofPhi()
  LOCAL Ict%,Jct%,PrvN%,KPhiN%,TT1%,TT2%,TT3%,TT4%

  FOR Ict% = 1 TO NumofPhi%
    KPhiN% = NofPhi(Ict%,1) * 2
    PrvN% = KPhiN% - 1
    FOR Jct% = KPhiN%+1 TO NumBandW%+PrvN%
      TT1% = NumBandW% - Jct% + KPhiN%
      TT2% = NumBandW% + Jct% - KPhiN%
      TT3% = NumBandW% - PrvN% + KPhiN%
      TT4% = NumBandW% + PrvN% - KPhiN%
      IF Jct% <= NP% THEN
        Fstar(Jct%,1) = Fstar(Jct%,1)-Amatrx(Jct%,TT1%)*Phi(KPhiN%,1)
        Fstar(Jct%,1) = Fstar(Jct%,1)+Pmatrx(Jct%,TT1%)*Phi1(KPhiN%,1)
        Amatrx(KPhiN%,TT2%) = 0.
        Amatrx(Jct%,TT1%) = 0.
        Pmatrx(KPhiN%,TT2%) = 0.
        Pmatrx(Jct%,TT1%) = 0.
      END IF
      IF PrvN% > 0 THEN
        Fstar(PrvN%,1) = Fstar(PrvN%,1)-Amatrx(PrvN%,TT3%)*Phi(KPhiN%,1)
        Fstar(PrvN%,1) = Fstar(PrvN%,1)+Pmatrx(PrvN%,TT3%)*Phi1(KPhiN%,1)
        Amatrx(PrvN%,TT3%) = 0.
        Amatrx(KPhiN%,TT4%) = 0.
        Pmatrx(PrvN%,TT3%) = 0.
        Pmatrx(KPhiN%,TT4%) = 0.
        PrvN% = PrvN% - 1
      END IF
    NEXT Jct%
    Amatrx(KPhiN%,NumBandW%) = 1.
    Pmatrx(KPhiN%,NumBandW%) = 0.
    Fstar(KPhiN%,1) = Phi(KPhiN%,1)
  NEXT Ict%
  end sub

sub matrixNumult(x(2),rx%,cx%,var1,z(2))
  rem ************************************************************************
  rem * A subroutine for matrix multiplication by a constant.          *
  rem ************************************************************************
  local ii%,jj%

  for ii% = 1 to rx%
    for jj% = 1 to cx%
      z(ii%,jj%) = 0
    next jj%
  next ii%
```

```
  for ii% = 1 to rx%
    for jj% = 1 to cx%
      z(ii%,jj%) = var1 * x(ii%,jj%)
    next jj%
  next ii%
  end sub

sub matrixNumultMod(x(2),rx%,cx%,var1,var2,z(2))
  rem ************************************************************************
  rem * A subroutine for matrix multiplication by two constants.  The       *
  rem * for the unsteady part of the problem while the second is for         *
  rem * the steady part of the problem when other than full dynamic         *
  rem * equations are used.                                                 *
  rem ************************************************************************
  local ii%,jj%

  for ii% = 1 to rx%
    for jj% = 1 to cx%
      z(ii%,jj%) = 0
    next jj%
  next ii%

  for ii% = 1 to rx%
    for jj% = 1 to cx%
      if int(ii%/2)*2 = ii% then
        z(ii%,jj%) = var2 * x(ii%,jj%)
      else
        z(ii%,jj%) = var1 * x(ii%,jj%)
      end if
    next jj%
  next ii%
  end sub

sub matrixprt(x(2),rx%,cx%,var$)
  rem ************************************************************************
  rem * A subroutine for printing matrices.                                  *
  rem ************************************************************************
  local ii%,jj%

  PRINT #1,
  PRINT #1,
  PRINT #1, space$(20);var$
  PRINT #1,
  for ii% = 1 to rx%
    for jj% =1 to cx%
      PRINT #1, using "##.#### ";x(ii%,jj%);
    next jj%
    PRINT #1,
  next ii%
  PRINT #1,
  PRINT #1,
  end sub

sub VectorTprt(x(2),rx%,cx%)
  rem ************************************************************************
  rem * A subroutine for printing matrices transposes.                       *
  rem ************************************************************************
  local ii%,jj%

  for ii% = 1 to cx%
    for jj% =1 to rx%
      PRINT #1, using "##.#### ";x(jj%,ii%);
      if int(jj%/8)*8 = jj% then PRINT #1,
    next jj%
    PRINT #1,
  next ii%
  end sub
```

```
sub matrixadd(x(2),rx%,cx%,y(2),ry%,cy%,z(2),rz%,cz%)
  rem ************************************************************************
  rem * A subroutine for matrix addition.                                    *
  rem ************************************************************************
  local ii%,jj%,kk%

  if cx%<>cy% or rx%<>ry% then
    PRINT #1, "Matrices can't be added !!!!"
    goto quitadd
  end if

  rz% = rx%
  cz% = cy%

  for ii% = 1 to rx%
    for jj% = 1 to cx%
      z(ii%,jj%) = 0
    next jj%
  next ii%

  for ii% = 1 to rx%
    for jj% = 1 to cx%
      z(ii%,jj%) = x(ii%,jj%) + y(ii%,jj%)
    next jj%
  next ii%

  quitadd:
  end sub

SUB GAUSSBND(X(2),NEQU%,BndWdth%,C(2),Z(2))
  REM ************************************************************************
  REM *  A subprogram that implements the Gaussian Elimination              *
  REM *  procedure to the solution of a system of equations.  This           *
  REM *  subprogram takes the bandwidth into account when solving the        *
  REM *  system of equations.  However, if your system is not banded,        *
  REM *  you should use the total dimension of the matrix as the             *
  REM *  bandwidth.                                                          *
  REM ************************************************************************

  LOCAL UPLMT%,MXMUM,PVT%,K%,I%,J%,DUM,DUM%,II%,Factor,SumOfX
  SHARED S()

  for ii% = 1 to NEQU%
    Z(ii%,1) = 0
  next ii%

  UPLMT2G% = 3 * BndWdth% - 1
  FOR I% = 1 TO NEQU%
    S(I%) = ABS(X(I%,1))
    FOR J% = 2 TO 2 * BndWdth% - 1
      IF ABS(X(I%,J%)) > S(I%) THEN
        S(I%) = ABS(X(I%,J%))
      END IF
    NEXT J%
  NEXT I%

  FOR K% = 1 TO NEQU%-1
    PVT% = K%
    MXMUM = ABS(X(K%,BndWdth%)/S(K%))
    UPLMT% = K% + BndWdth% - 1

    IF UPLMT% > NEQU% THEN UPLMT% = NEQU%

    FOR II% = K% + 1 TO UPLMT%
      IF BndWdth%+K%-II% > 0 THEN
        DUM = ABS(X(II%,BndWdth%+K%-II%)/S(II%))
        IF DUM > MXMUM THEN
          MXMUM = DUM
          PVT% = II%
```

```
      END IF
    END IF
  NEXT II%

  IF PVT% <> K% THEN
    FOR III% = BndWdth% TO UPLMT2G%
      DUM = X(PVT%,III%-PVT%+K%)
      X(PVT%,III%-PVT%+K%) = X(K%,III%)
      X(K%,III%) = DUM
    NEXT III%
    DUM = C(PVT%,1)
    C(PVT%,1) = C(K%,1)
    C(K%,1) = DUM
    DUM = S(PVT%)
    S(PVT%) = S(K%)
    S(K%) = DUM
  END IF

  UPLMT2% = 3*BndWdth%-1
  IF UPLMT2% > NEQU% THEN UPLMT2% = NEQU%
  FOR I% = K% + 1 TO UPLMT%
    Factor = X(I%,BndWdth%-I%+K%)/X(K%,BndWdth%)
    FOR J% = BndWdth%+1 TO UPLMT2G%
      X(I%,J%-I%+K%)=X(I%,J%-I%+K%) - Factor*X(K%,J%)
    NEXT J%
    C(I%,1) = C(I%,1) - Factor * C(K%,1)
  NEXT I%
NEXT K%

Z(NEQU%,1) = C(NEQU%,1) / X(NEQU%,BndWdth%)
FOR I% = NEQU%-1 TO 1 STEP -1
  SumOfX = 0
  FOR J% = I% + 1 TO NEQU%
    IF BndWdth%+J%-I% <= UPLMT2G% THEN
      SumOfX = SumOfX + X(I%,BndWdth%+J%-I%) * Z(J%,1)
    END IF
  NEXT J%
  Z(I%,1) = (C(I%,1) - SumOfX)/ X(I%,BndWdth%)
NEXT I%
END SUB

sub matrixVectmult(X(2),NEQU%,BndWdth%,C(2),Z(2))
  rem ************************************************************************
  rem * A subroutine for matrix multiplication.  This routine would    *
  rem * only be usable to mutiply a matrix [X] by a vector {C}.  The   *
  rem * matrix should be banded and the band width should be given.    *
  rem * Moreover, the the matrix should be based in a banded form with *
  rem * the dimensions of [X] as follows: NEQU% rows and 2*BndWdth%-1  *
  rem * columns.  If the data is not passed as such, errors will occur.*
  rem * The subroutine doesn't have any error checks.                 *
  rem ************************************************************************

  LOCAL I%,J%,II%

  for ii% = 1 to NEQU%
    Z(ii%,1) = 0
  next ii%

  FOR I% = 1 TO NEQU%
    FOR J% = 1 TO 2*BndWdth%-1
      IF -BndWdth%+J%+I% > 0 AND -BndWdth%+J%+I%<=NEQU% THEN
        Z(I%,1) = Z(I%,1) + X(I%,J%) * C(-BndWdth%+J%+I%,1)
      END IF
    NEXT J%
  NEXT I%
  END SUB
```

```
INITIAL:
  REM ************************************************************************
  REM * A Subroutine to initialize parameters.                               *
  REM ************************************************************************
  FDorFE% = 1          '<====1 is finite difference, 0 is finite element==
  False%= 0
  True% = 1
  IrrMethod$ = "F"
  Method$ = "K"
  TypElem$ = "L"
  SelPrtOpt$ = "F"
  iprint% = 0

  VARTheta    = .50       '   <=====================================
  ALPHA       = .25

  DeltaT      = 5         '   <==minutes============================
  numiter%    = 10        '   <=====================================
  AllError    = 0.0005
  TotNumStep%= 20
  CONSISorLUMP% = 0    '<====1 is consistent, 0 lumped====
  CoefTW = 1.0 '<====== Coefficient of Top Width (possible values are 1 & 1.5)

  FurLength = 625
  TimCut = 300  'min
  Qin    = 2.78 'liters/sec
  kofInf = 0.0252
  fsub0  = 0.00023
  Slope = 0.0044
  aofInf = 0.02
  ManngN = 0.03

  IF IrrMethod$ = "B" THEN
    Rho1   = 1
    Rho2   = 3.3333333
    Sigma1 = 1.5
    Sigma2 = 1
  ELSE
    Rho1   = 0.46
    Rho2   = 2.86
    Sigma1 = 0.92
    Sigma2 = 0.65
  END IF
  RETURN
DEF FNExists%(FilNam$)
  REM ************************************************************************
  REM *  The function Exists% returns a non zero integer value if the  *
  REM * file specified by FilNam$ is on the current disk drive.        *
  REM ************************************************************************

  Shared False%,True%
  LOCAL ExistF%

  ON ERROR GOTO FileError
  ExistF% = True%                      'initial

  OPEN FilNam$ FOR INPUT AS #9
  IF ERR=0 THEN CLOSE #9
  GOTO Finish

  FileError:
    ExistF% = FALSE%                   ' File doesn't exist
    RESUME NEXT

  Finish:
  ON ERROR GOTO 0
  FNExists%=ExistF%
END DEF
```

APPENDIX B

FE-SURFDSGN Graphics Routine Listing

```
REM ************************************************************************
REM *                                                                      *
REM *                        Program SURFGRPH.BAS                          *
REM *                                                                      *
REM ************************************************************************
REM *                                                                      *
REM *                             Developed By                             *
REM *                                                                      *
REM *                            Walid H. Shayya                           *
REM *                                                                      *
REM *                      Michigan State University                       *
REM *                                                                      *
REM *                             July 30, 1990                            *
REM *                                                                      *
REM ************************************************************************

COMMON NGphFil$

%MaxNpts = 200

DIM Varbl1(%MaxNpts), Varbl2(%MaxNpts)
DIM XPOS%(6), YPOS%(11), XPnts(%MaxNpts), YPnts(%MaxNpts)
DIM Length(%MaxNpts,3), Time(%MaxNpts,3)
DIM GrphXpnt(4), XX%(10), YY%(10), CC$(10), AA$(10)

SCREEN 0

IF NGphFil$ ="" THEN
  GOSUB InitialScrn
END IF

CLS
'ON ERROR GOTO Trap

NumOfPrb% = 1

GOSUB HEADING

LOCATE 8, 10
IF NGphFil$ ="" THEN
  INPUT "Enter the name of file to plot (no extension) : ", NGphFil$
END IF
FilPrg1$ = NGphFil$ + ".PRG"
FilPrg2$ = NGphFil$ + ".REC"

FilIn$ =  NGphFil$ + ".OUT"

CheckEnt$ = ""
GOSUB GetFilNam

GOSUB LdScrnFl

IF CheckEnt$ = "R" OR CheckEnt$ = "B" THEN
  NumOfPrb% = NumOfPrb% + 1
  GOSUB LdScrnFl1 'open file for data output
END IF

IF CheckEnt$ = "A" OR CheckEnt$ = "B" THEN
  NumOfPrb% = NumOfPrb% + 1
  GOSUB OpnFil 'open file for data output
END IF

GOSUB Initial
```

```
GOSUB HEADING

GOSUB GetScren

REM ************************************************************************
REM * Prepare Screen for output.
REM ************************************************************************
CALL PrntScrn

GOSUB PrepDGrph

CALL FindRange2(MaxLength / 10)

CALL FindRange1(MaxTime / 10)

CALL LabelAxis

GOSUB GetDat

a$ = INPUT$(1)
SCREEN 0
END

InitialScrn:
  REM ************************************************************************
  REM * A subroutine to display the first screen.
  REM ************************************************************************
  KEY OFF
  XX%(1) = 5: XX%(2) = 23: XX%(3) = 23: XX%(4) = 22: XX%(5) = 10
  YY%(1) = 2: YY%(2) = 4: YY%(3) = 8: YY%(4) = 10: YY%(5) = 12
  XX%(6) = 17: YY%(6) = 14
  CC$(1) = CHR$(201): CC$(2) = CHR$(205): CC$(3) = CHR$(187)
  CC$(4) = CHR$(186): CC$(5) = CHR$(204): CC$(6) = CHR$(185)
  CC$(7) = CHR$(200): CC$(8) = CHR$(188)
  A1$ = CC$(1) + STRING$(54, CC$(2)) + CC$(3)
  A2$ = CC$(5) + STRING$(54, CC$(2)) + CC$(6)
  A3$ = CC$(7) + STRING$(54, CC$(2)) + CC$(8)
  AA$(1) = " FINITE ELEMENT SURFACE IRRIGATION DESIGN MODEL "
  AA$(2) = "VERSION 1.00"
  AA$(3) = "Developed by"
  AA$(4) = "Walid H. Shayya"
  AA$(5) = "Department of Agricultural Engineering"
  AA$(6) = "Michigan State University"

  CLS
  LOCATE 4, 1

  PRINT SPACE$(12); A1$
  FOR Iloop% = 1 TO 5
    PRINT SPACE$(12); CC$(4); SPACE$(54); CC$(4)
  NEXT Iloop%
  PRINT SPACE$(12); A2$
  FOR Iloop% = 1 TO 9
    PRINT SPACE$(12); CC$(4); SPACE$(54); CC$(4)
  NEXT Iloop%
  PRINT SPACE$(12); A3$
  FOR Iloop% = 1 TO 6
    LOCATE YY%(Iloop%) + 4, XX%(Iloop%) + 12
    PRINT AA$(Iloop%)
  NEXT Iloop%

  LOCATE 23, 48
  COLOR 15, 0
  PRINT "Press any key to continue."
  COLOR 7, 0

  WHILE INKEY$ = "": WEND
  RETURN
```

```
HEADING:
  REM ************************************************************************
  REM *                            Heading Number 1
  REM ************************************************************************
  CLS
  LOCATE 2, 16
  PRINT " FINITE ELEMENT SURFACE IRRIGATION DESIGN MODEL "
  LOCATE 3, 16
  PRINT "                 GRAPHICS ROUTINE"
  LOCATE 4, 16
  PRINT STRING$(48, 196)
  RETURN

GetScren:
  REM ************************************************************************
  REM *                         Check the screen number.                     *
  REM ************************************************************************
  LOCATE 8, 10
  PRINT "Please select the number of the graphics screen from the following :"
  LOCATE 11, 10: PRINT "Press "; : COLOR 0, 7: PRINT " 2 "; : COLOR 7, 0
  PRINT " for a high resolution ,graphics screen (640x200 pixels),"
  LOCATE 13, 16: COLOR 0, 7: PRINT " 9 "; : COLOR 7, 0
  PRINT " for an enhanced resolution graphics screen (640x350 ";
  LOCATE 14, 20
  PRINT "pixels), or"
  LOCATE 16, 16: COLOR 0, 7: PRINT " S "; : COLOR 7, 0
  PRINT " for special screen number 10 (640x200 pixels) : ";
  WHILE ScnNum$ = "": ScnNum$ = INKEY$: WEND
  ScnNum$ = UCASE$(ScnNum$): PRINT ScnNum$
  IF ScnNum$ <> "2" AND ScnNum$ <> "9" AND ScnNum$ <> "S" THEN
    BEEP: ScnNum$ = "": LOCATE 23, 42: PRINT "Press either "; : COLOR 0, 7
    PRINT "2"; : COLOR 7, 0: PRINT ", "; : COLOR 0, 7:
    PRINT "9"; : COLOR 7, 0: PRINT " or "; : COLOR 0, 7:
    PRINT "S"; : COLOR 7, 0
    PRINT " to proceed."
    GOTO GetScren
  END IF
  RETURN

GetFilNam:
  REM ************************************************************************
  REM * A subroutine to check if to plot any additional data files.
  REM ************************************************************************
  LOCATE 11, 10
  PRINT "Would you like to plot any additional data files ?";
  LOCATE 14, 10: PRINT "Press "; : COLOR 0, 7: PRINT " R "; : COLOR 7, 0
  PRINT " for simulated recession curve,"
  LOCATE 16, 16: COLOR 0, 7: PRINT " A "; : COLOR 7, 0
  PRINT " for a plot of actual advance data curve, or"
  LOCATE 18, 16: COLOR 0, 7: PRINT " B "; : COLOR 7, 0
  PRINT " for a plot of both curves, or"
  LOCATE 20, 16: COLOR 0, 7: PRINT " N "; : COLOR 7, 0
  PRINT " for a plot of neither curve : ";
  WHILE CheckEnt$ = "": CheckEnt$ = INKEY$: WEND
  CheckEnt$ = UCASE$(CheckEnt$): PRINT CheckEnt$
  IF CheckEnt$ <> "R" AND CheckEnt$ <> "A" AND CheckEnt$ <> "B" _
           AND CheckEnt$ <> "N" THEN
    BEEP: CheckEnt$ = "": LOCATE 23, 40: PRINT "Press either "; : COLOR 0, 7
    PRINT "R"; : COLOR 7, 0: PRINT ", "; : COLOR 0, 7: PRINT "A";
    COLOR 7, 0: PRINT ", "; : COLOR 0, 7: PRINT "B";
    COLOR 7, 0: PRINT ", or "; : COLOR 0, 7: PRINT "N";
    COLOR 7, 0
    PRINT " to proceed."
```

```
    GOTO GetFilNam
  END IF
  RETURN

OpnFil:
  REM ************************************************************************
  REM * Open file and initialize numbers.
  REM ************************************************************************
  OPEN "I", #1, FilIn$
  NumOfPts%(NumOfPrb%) = 0
  IILP% = 1
  WHILE NOT EOF(1)
    INPUT #1, Time(IILP%,NumOfPrb%), Length(IILP%,NumOfPrb%)
    IILP% = IILP% + 1
    NumOfPts%(NumOfPrb%) = NumOfPts%(NumOfPrb%) + 1
  WEND
  CLOSE #1
  RETURN

PrepDGrph:
  REM ************************************************************************
  REM *  Subroutine to prepare for doing graph.
  REM ************************************************************************
  MaxLength = 0
  MaxTime = 0

  FOR J = 1 TO NumOfPrb%
    FOR I = 1 TO NumOfPts%(J)
      IF Length(I,J) > MaxLength THEN MaxLength = Length(I,J)
      IF Time(I,J) > MaxTime THEN MaxTime = Time(I,J)
    NEXT I
  NEXT J
  RETURN

GetDat:
  REM ************************************************************************
  REM * A subroutine initialize the array.
  REM ************************************************************************
  FOR Jlop% = 1 TO NumOfPrb%
    FOR Ilop% = 1 TO NumOfPts%(Jlop%)
      Varbl2(Ilop%) = Length(Ilop%,Jlop%)
      Varbl1(Ilop%) = Time(Ilop%,Jlop%)
      IF Varbl2(Ilop%) < 0 THEN Varbl2(Ilop%) = 0
      IF Varbl1(Ilop%) < 0 THEN Varbl1(Ilop%) = 0
    NEXT Ilop%
    CALL PrepPoints(Varbl2(), Varbl1(), Jlop%)
  NEXT Jlop%
  RETURN

LdScrnFl:
  REM ************************************************************************
  REM * Subroutine to load file containing saved screen for file 1.
  REM ************************************************************************
  OPEN "I", #1, FilPrg1$
  NumOfPts%(1) = 0
  IILP% = 1
  WHILE NOT EOF(1)
    INPUT #1, Time(IILP%,1), Length(IILP%,1)
    IILP% = IILP% + 1
    NumOfPts%(1) = NumOfPts%(1) + 1
  WEND
  CLOSE #1
  RETURN
```

```
LdScrnFl1:
  REM ************************************************************************
  REM * Subroutine to load file containing saved screen for file 2.
  REM ************************************************************************
  OPEN "I", #1, FilPrg2$
  NumOfPts%(2) = 0
  IILP% = 1
  WHILE NOT EOF(1)
    INPUT #1, Time(IILP%,2), Length(IILP%,2)
    IILP% = IILP% + 1
    NumOfPts%(2) = NumOfPts%(2) + 1
  WEND
  CLOSE #1
  RETURN
Initial:
  REM ************************************************************************
  REM * Subroutine to Initialize.
  REM ************************************************************************
  XPos%(1) = 21:  YPos%(1) = 13
  XPos%(2) = 17:  YPos%(2) = 25
  XPos%(3) = 13:  YPos%(3) = 38
  XPos%(4) = 10:  YPos%(4) = 50
  XPos%(5) = 7:   YPos%(5) = 63
  XPos%(6) = 3:   YPos%(6) = 75
  GrphXpnt(1) = 4.0
  GrphXpnt(2) = 8.0
  GrphXpnt(3) = 12.0
  GrphXpnt(4) = 16.0
  RETURN
Trap:
  REM ************************************************************************
  REM * A subroutine to handle error.
  REM ************************************************************************
  LOCATE 24, 1
  PRINT SPC(79);
  LOCATE 23, 35
  PRINT "Error Number "; ERR; " has occurred !";
  LOCATE 24, 35
  PRINT "    Press any key to continue.";
  RESUME Qtrap
  RETURN
Qtrap:
  WHILE INKEY$ = "": WEND
  SCREEN 0
  CLS
  END
SUB DoGrph (X1, Y1, X2, Y2)
  REM ************************************************************************
  REM * This subprogram recieves 2 points (x1,y1) and (x2,y2) and
  REM * connects the line between these two points.
  REM ************************************************************************
  LINE (X1, Y1)-(X2, Y2)
END SUB
SUB DoGrph2 (X1, Y1, X2, Y2)
  REM ************************************************************************
  REM * This subprogram recieves 2 points (x1,y1) and (x2,y2) and
  REM * connects the line between these two points.
  REM ************************************************************************
  SHARED Numbr%
  IF Numbr% = 2 THEN
    LINE (X1, Y1)-(X2, Y2), , , 8888
  ELSEIF Numbr% = 3 THEN
```

```
    LINE (X1, Y1)-(X2, Y2), , , 1111
  ELSE
    LINE (X1, Y1)-(X2, Y2)
  END IF
END SUB

SUB DoGrph1 (X1, Y1)
  REM ************************************************************************
  REM * This subprogram draws a point (x1,y1).
  REM ************************************************************************
  Circle (X1, Y1),.5
END SUB

SUB FindRange2 (Var2)
  REM ************************************************************************
  REM * A subroutine to find the Range 1.
  REM ************************************************************************
  SHARED Range1, Form1$
  'LOCAL II
  FOR II = 90 TO 10 STEP -5
    IF Var2 > II THEN
      Range1 = II + 5
      Form1$ = "####"
      GOTO QSB
    END IF
  NEXT II
  IF Var2 > 8 THEN
    Range1 = 10
    Form1$ = "####"
    GOTO QSB
  END IF
  FOR II = 10 TO 2 STEP -2
    IF Var2 > II THEN
      Range1 = II + 2
      Form1$ = "##.#"
      GOTO QSB
    END IF
  NEXT II
  FOR II = 2 TO 1 STEP -1
    IF Var2 > II THEN
      Range1 = II + 1
      Form1$ = "##.#"
      GOTO QSB
    END IF
  NEXT II
  FOR II = 1 TO 0 STEP -.5
    IF Var2 > II THEN
      Range1 = II + .5
      Form1$ = "##.#"
      GOTO QSB
    END IF
  NEXT II
  Range1 = .5
  Form1$ = "##.#"
QSB:
END SUB

SUB FindRange1 (Var1)
  REM ************************************************************************
  REM * A subroutine to find the Range 2.
  REM ************************************************************************
  SHARED Range2, Form2$, CheckEnt$
  'LOCAL II
  FOR II = 190 TO 10 STEP -5
    IF CheckEnt$="A" THEN
      IF Var1 > II THEN
```

```
        Range2 = II + 5
        Form2$ = "####"
        GOTO QSB2
      END IF
    ELSE
      IF Var1 > II - 5 THEN
        Range2 = II + 5
        Form2$ = "####"
        GOTO QSB2
      END IF
    END IF
  NEXT II
  IF Var1 > 8 THEN
    Range2 = 10
    Form2$ = "####"
    GOTO QSB2
  END IF
  FOR II = 10 TO 2 STEP -2
    IF Var1 > II THEN
      Range2 = II + 2
      Form2$ = "##.#"
      GOTO QSB2
    END IF
  NEXT II
  FOR II = 2 TO 1 STEP -1
    IF Var1 > II THEN
      Range2 = II + 1
      Form2$ = "##.#"
      GOTO QSB2
    END IF
  NEXT II
  FOR II = 1 TO 0 STEP -.5
    IF Var1 > II THEN
      Range2 = II + .5
      Form2$ = "##.#"
      GOTO QSB2
    END IF
  NEXT II
  Range2 = .5
  Form2$ = "##.#"
QSB2:
END SUB

SUB LabelAxis
  REM ************************************************************************
  REM * A subroutine to label X and Y axis.
  REM ************************************************************************
  SHARED Range1, Form1$, Direct$, XPOS%(), YPOS%(), Bound$, Range2, Form2$
  'LOCAL VertPos%, TLabel1%, TLabel2%, HorizPos%, f1$, f2$, Varb1%

  f2$ = Form1$
  f1$ = Form2$
  FOR TLabel1% = 0 TO 5
    LOCATE XPos%(TLabel1% + 1), 9
    PRINT USING f1$; Range2 * (TLabel1% * 2);
  NEXT TLabel1%
  FOR TLabel2% = 0 TO 5
    LOCATE 22, YPos%(TLabel2% + 1)
    PRINT USING f2$; Range1 * (TLabel2% * 2);
  NEXT TLabel2%

END SUB
```

```
SUB PrepPoints (XPnts(), YPnts(), NumOfIter%)
  REM ************************************************************************
  REM * A subroutine to draw points.
  REM ************************************************************************
  SHARED Range1, Range2, NumOfPts%(), NumOfPrb%, CheckEnt$
  'LOCAL Pont%, r1, r2
  r1 = Range1
  r2 = Range2
  FOR Pont% = 1 TO NumOfPts%(NumOfIter%)
    XPnts(Pont%) = XPnts(Pont%) / r1 * 10
    YPnts(Pont%) = YPnts(Pont%) / r2 * 10
  NEXT Pont%
  IF NumOfIter% = 1 THEN
    FOR Pont% = 2 TO NumOfPts%(NumOfIter%)
      CALL DoGrph(XPnts(Pont% - 1), YPnts(Pont% - 1), XPnts(Pont%),
YPnts(Pont%))
    NEXT Pont%
  ELSEIF NumOfIter% = 2 AND (CheckEnt$ = "R" OR CheckEnt$ = "B") THEN
    FOR Pont% = 2 TO NumOfPts%(NumOfIter%)
      CALL DoGrph(XPnts(Pont% - 1), YPnts(Pont% - 1), XPnts(Pont%),
YPnts(Pont%))
    NEXT Pont%
  ELSE
    FOR Pont% = 1 TO NumOfPts%(NumOfIter%)
      CALL DoGrph1(XPnts(Pont%), YPnts(Pont%))
    NEXT Pont%
    IF CheckEnt$ = "A" THEN
      FOR ILP = 2 to 6 step 2
        Circle (ILP, 83),.5
      NEXT ILP
      LOCATE 6, 22
      PRINT "Actual Data"
    ELSEIF CheckEnt$ = "B" THEN
      FOR ILP = 2 to 6 step 2
        Circle (ILP-20, -20),.5
      NEXT ILP
      LOCATE 24, 9
      PRINT "Actual Data";
    END IF
  END IF
END SUB

SUB PrntScrn
  REM ************************************************************************
  REM * A subroutine to plot the X and Y axis.
  REM ************************************************************************
  SHARED GrphXpnt(), ScnNum$, CheckEnt$
  'LOCAL VertPos%, TLabel1%, TLabel2%, HorizPos%, f1$, f2$, Varbl%

  IF ScnNum$ = "2" THEN
    SCREEN 2
  ELSEIF ScnNum$ = "9" THEN
    SCREEN 9
  ELSE
    SCREEN 10
  END IF

  WINDOW (-24, -28)-(105, 115)

  LINE (-24, -28)-(105, 115), , B
  LINE (-23.75, -28)-(104.75, 115), , B

  LINE (0, 0)-(100, 100), , B

  FOR VertPos% = 0 TO 100 STEP 20
    LINE (0, VertPos%)-(-2.5, VertPos%)
    LINE (100, VertPos%)-(97.5, VertPos%)
  NEXT VertPos%
```

```
  FOR ICycle% = 1 TO 5
    FOR JCycle% = 1 TO 4
      LOCAT2 = GrphXpnt(JCycle%) + (ICycle% - 1) * 20
      LINE (0, LOCAT2)-(-1.5, LOCAT2)
      LINE (100, LOCAT2)-(98.5, LOCAT2)
    NEXT JCycle%
  NEXT ICycle%

  FOR HorizPos% = 0 TO 100 STEP 20
    LINE (HorizPos%, 0)-(HorizPos%, -4)
    LINE (HorizPos%, 100)-(HorizPos%, 96)
  NEXT HorizPos%
  FOR ICycle% = 1 TO 5
    FOR JCycle% = 1 TO 4
      LOCAT2 = GrphXpnt(JCycle%) + (ICycle% - 1) * 20
      LINE (LOCAT2, 0)-(LOCAT2, -2)
      LINE (LOCAT2, 100)-(LOCAT2, 98)
    NEXT JCycle%
  NEXT ICycle%

  LOCATE 24, 41
  PRINT "Distance (m)";
  LOCATE 12, 3
  PRINT "Time"
  LOCATE 13, 3
  PRINT "(min)";

  LOCATE 2, 18
  PRINT "Finite Element Surface Irrigation Design model";

  IF CheckEnt$="A" OR CheckEnt$="N" THEN
    LINE (2, 90)-(8, 92)
    LOCATE 5, 22
    PRINT "Simulated Data"
  ELSEIF CheckEnt$ = "B" THEN
    LINE (71, -20)-(77, -19)
    LOCATE 24, 65
    PRINT "Simulated Data";
  ELSEIF CheckEnt$ = "R" THEN
    LINE (-22, -20)-(-16, -19)
    LOCATE 24, 7
    PRINT "Simulated Data";
  END IF
END SUB
```