DISSERTATION

GIS-BASED UPLAND EROSION MODELING, GEOVISUALIZATION AND GRID SIZE EFFECTS ON EROSION SIMULATIONS WITH CASC2D-SED

Submitted by

Rosalía Rojas Sánchez

Department of Civil Engineering

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Summer 2002

UMI Number: 3064017

UMI®

UMI Microform 3064017

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

COLORADO STATE UNIVERSITY

June 5, 2002

WE HEREBY RECOMMEND THAT THE DISSERTATION PREPARED UNDER OUR SUPERVISION BY ROSALIA ROJAS SANCHEZ ENTITLED GIS-BASED UPLAND EROSION MODELING, GEOVISUALIZATION AND GRID SIZE EFFECTS ON EROSION SIMULATIONS WITH CASC2D-SED BE ACCEPTED AS FULFILLLING IN PART REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Committee on Graduate Work

____

____

____

____
Advisor

____
Department Head

ABSTRACT OF DISSERTATION

GIS-BASED UPLAND EROSION MODELING, GEOVISUALIZATION AND GRID SIZE EFFECTS ON EROSION SIMULATIONS WITH CASC2D-SED

The recent development of two-dimensional models for the simulation of upland erosion and sediment yield from watersheds facilitates the analysis of washload in rivers and assists in the planning of soil conservation systems.

The spatial resolution in Digital Elevation Models influences the landscape representation. In runoff models, change in the physical variables translate into changes in the hydrological model outputs. Process-based upland erosion models are affected by both the change in landscape representation and the change in the hydrological model. The effects of grid cell size on a 2-dimensional, physically-based erosion model have not yet been investigated.

The objectives of this research are to: (1) extend the development of the CASC2D-SED upland erosion model and implement a suitable channel sediment routing algorithm; (2) couple the model time-series output grids with a GIS that will allow the geovisualization of sediment transport dynamics by size fraction and the spatial erosion and deposition patterns; and (3) investigate the effects of grid resolution on the simulated values of gross erosion, sediment yield, spatial distribution of net erosion and sediment delivery ratios.

The physically-based model CASC2D-SED calculates runoff and upland erosion and simulates spatial erosion rates in a watershed on a grid basis. This two-dimensional model is extended in this study to include interception, revised upland erosion and

sediment routing, and a channel sediment routing algorithm. In the channels, the Engelund and Hansen transport equation is used to approximate the sediment transport capacity by size fraction. While erosion in channels is not allowed, deposition may occur. The time-series output grids from CASC2D-SED are coupled with GIS for their automatic display and creation of a MPEG movie. The extensively monitored 20.5 km^2 Goodwin Creek watershed, MS, is used to test the new CASC2D-SED model and to study the effects of grid cell size on erosion simulations.

The calibration and validation runs of CASC2D-SED show that the model appropriately predicts hydrographs and sediment graphs at the basin outlet and other internal locations. Washload is simulated to be transported through the system by streamflow with very little deposition, carried primarily in suspension while coarser fractions move as bed-material with local deposition. CASC2D-SED predicts sediment graphs by size fraction at the basin outlet. Sediment yields are predicted within ±50% of the observed data. Spatially, CASC2D-SED predicts areas of net erosion on steep slopes, depending on the land use and soil type. Zones of deposition were simulated in valley bottoms or forested areas. Geovisualization and time-series animation of the erosion dynamics and distribution of net erosion represents a form of model validation and facilitates model testing at internal locations.

Goodwin Creek GIS data at 30-m grid cell spacing was resampled to up to 330-m. The input grid cell size affected the CASC2D-SED erosion predictions and at increasingly larger grid cell sizes (x), equilibrium discharge and sediment yield decrease approximately with the inverse of the grid size while gross erosion decreases with $x^{-1.5}$.

Estimates of sediment delivery ratio increase with grid cell size, and values simulated for the cases of 30-m and 90-m fall within field observations.

From the results of the calibration and validation runs of CASC2D-SED on Goodwin Creek, MS, it is concluded that a grid cell size smaller that 150-m should be used to identify regions vulnerable to erosion on a watershed for the implementation of a suitable erosion control measure. For the prediction of sediment delivery ratios closer to field observations, a grid size smaller than 90-m is preferable.

Rosalía Rojas Sánchez
Civil Engineering Department
Colorado State University
Fort Collins, CO 80523
Summer 2002

ACKNOWLEDGMENTS

I would like to take this opportunity to express my appreciation and gratitude to my advisor, Dr. Pierre Julien, for his constant guidance and encouragement. I would also like to thank Dr. Jorge Ramírez, Dr. Denis Dean and Dr. Brian Bledsoe for serving on my committee and for their useful insight and support.

I am grateful to Dr. Billy E. Johnson, Dr. David Woolhiser, and Dr. Oli Sveinsson for their invaluable input. I am especially grateful to Gerhard, for his tremendous support through difficult times, his guidance, and undying encouragement. My friends, colleagues, and instructors -from everywhere-, who have helped me accomplish this work. I am deeply grateful to them. And finally, my heartfelt gratitude to my loving parents and brother and sisters for their unending moral support.

Financial support was granted through the Instituto Nacional de Investigaciones Agrarias in Spain and through the Center for Geosciences.

De nuevo,

a mis padres

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION 1

1.1. PROBLEM STATEMENT 1

1.2. RESEARCH OBJECTIVES 4

1.3. METHODOLOGY AND APPROACH 5

CHAPTER 2 LITERATURE REVIEW 7

2.1. SOIL EROSION MODELS 8

2.1.1. Rationale and Limitations 8

2.1.2. Review of Existing Erosion Models 10

2.1.3. Soil Erosion Modeling and GIS 13

2.2. GEOVISUALIZATION AND TIME-SERIES ANIMATION. 15

2.3. CELL SIZE EFFECTS 18

2.3.1. On Landscape Representation 18

2.3.2. On Simulated Flows 22

2.3.3. On Soil Erosion Prediction 26

2.4. SUMMARY 29

CHAPTER 3 THE REVISED CASC2D-SED MODEL 30

3.1. MODEL DEVELOPMENT 31

3.2. CASC2D-SED DESCRIPTION 33

3.3. FLOW ROUTING ALGORITHM 35

3.3.1. Precipitation 35

3.3.2. Interception 35

3.3.3. Infiltration 36

3.3.4. Overland Flow 37

3.3.5. Channel Flow 39

3.4. SEDIMENT ROUTING ALGORITHM 40

3.4.1. New Sediment Transport Approximation 40

3.4.2. Revised Upland Erosion and Sediment Transport 47

3.4.3. Implementation of Channel Sediment Transport 50

3.4.4. Suspended Sediment Settling 52

3.5. CASC2D-SED GRIDS TIME-SERIES VISUALIZATION 55

3.6. SUMMARY 56

CHAPTER 4 STUDY SITE DATA SET 58

4.1. GOODWIN CREEK EXPERIMENTAL WATERSHED DESCRIPTION 59
4.2. DEM PRE-PROCESSING AND RESAMPLING 67
4.2.1. Elevations and Slopes 70
4.2.2. Channel Network and Watershed Delimitation 73
4.3. SOIL TYPE AND LAND USE GRIDS RESAMPLING 75
4.4. SUMMARY 77

CHAPTER 5 MODEL CALIBRATION AND VALIDATION 78

5.1. CASC2D-SED SET UP 79
5.1.1. Watershed Definition 79
5.1.2. Precipitation Events 79
5.1.3. Parameter Set 80
5.2. RESULTS 81
5.2.1. Calibration Event 81
5.2.2. Validation Events 94
5.3. SUMMARY 99

CHAPTER 6 MODELING PREDICTIONS AT DIFFERENT GRID SIZES 101

6.1. CONCEPTUAL EQUILIBRIUM UNIT SEDIMENT DISCHARGE 102
6.2. MODEL APPLICATION AT DIFFERENT SPATIAL RESOLUTIONS 105
6.2.1. Method 105
6.2.2. Results 107
6.3. SUMMARY 119

CHAPTER 7 SUMMARY AND CONCLUSIONS 121

7.1. SUMMARY 121
7.2. CONCLUSIONS 123

CITED REFERENCES 125

APPENDIX I: CASC2D-SED CODE I-1

APPENDIX II: GOODWIN CREEK RESAMPLED GRIDS II-1

APPENDIX III: CALIBRATION AND VALIDATION RESULTS III-1

LIST OF FIGURES

Figure 3-1. CASC2D-SED flowchart and revised/new code (in bold letters) ______ 32

Figure 3-2. Topographical representation of overland flow and channel routing schemes ______ 34

Figure 3-3. Sediment transport capacity and supply curves (Julien, 1995) ______ 41

Figure 3-4. Schematic of upland sediment transport (Johnson, 1997) ______ 43

Figure 3-5. Overland erosion and sediment routing flowchart ______ 44

Figure 3-6. Schematic of channel sediment transport ______ 45

Figure 3-7. Channel sediment routing flowchart ______ 46

Figure 4-1. Goodwin Creek location ______ 60

Figure 4-2. DEM, soils and land use maps of the Goodwin Creek watershed (30-m) _ 63

Figure 4-3. Goodwin Creek Watershed rain gauge locations ______ 64

Figure 4-4. Goodwin Creek watershed and studied sub-basins ______ 65

Figure 4-5. Channel network and link numbers (30-m) ______ 66

Figure 4-6. Resample of an even and odd number of cells ______ 68

Figure 4-7. Relative frequency of slope intervals for the different resolutions ______ 69

Figure 4-8. Slope cumulative distribution and frequencies for 30-, 90-, 150-, 210-, 270-, and 330-m resolution ______ 71

Figure 4-9. Statistical mean and standard deviation and maximum slope angles for 30-, 90-, 150-, 210-, 270-, and 330-m resolution ______ 71

Figure 4-10. Goodwin Creek slope angle values and distribution ______ 72

Figure 4-11. Percentage of the area covered by each soil type at Goodwin Creek ______ 76

Figure 4-12. Percentage of the area covered by each land use type at Goodwin Creek _ 76

Figure 5-1. Simulated rainfall minus interception depth, infiltration depth, water depth and sediment concentration at t = 14 min. (calibration event) ______ 82

Figure 5-2. Simulated excess rainfall, infiltration depth, water depth and sediment concentration at t = 83 min. (calibration event) ______ 82

Figure 5-3 Simulated excess rainfall, infiltration depth, water depth and sediment concentration at t = 129 min. (calibration event) 83

Figure 5-4. Simulated excess rainfall, infiltration depth, water depth and sediment concentration at t = 306 min. (calibration event) 83

Figure 5-5. Calibrated hydrographs for the calibration event 1 84

Figure 5-6. Calibrated sediment graphs for the calibration event 1 86

Figure 5-7. Maximum flux-averaged concentration by size fraction and total simulated in the calibration run (time 240 min.) 87

Figure 5-8. Sediment graph at the outlet by size fraction 87

Figure 5-9. Transport of sand, silt and clay fraction in suspension or deposition mode simulated at t = 173 min (calibration event) 88

Figure 5-10. Observed vs. simulated sediment yield at the outlet (station 1) and at internal locations (stations 4, 6, 7, 8, and 14) for event 1 90

Figure 5-11. Three-dimensional view of the net erosion (tons/ha) at the end of the event 1 simulation (90-m) 91

Figure 5-12. Elevation and net erosion at cross section A-A' 93

Figure 5-13. Observed vs. simulated a) discharge volume, b) peak discharge and c) time to peak for the calibration and validation events at all stations 95

Figure 5-14. Observed vs. simulated sediment yield at the outlet (station 1) and at internal locations (stations 4, 6, 7, 8, and 14) for a) event 2 and b) event 3 97

Figure 5-15. Sediment rating curves at the outlet (station 1) and at two internal nested locations (stations 4 and 7) for the calibration and validation events 98

Figure 6-1. a) Peak discharge and time to peak and b) percentage of runoff and infiltration for different grid sizes before model calibration 107

Figure 6-2. a) Peak discharge and time to peak and b) percentage of runoff and infiltration for different grid sizes after model calibration 107

Figure 6-3. Normalized equilibrium sediment discharge at the watershed outlet for different normalized grid cell values and rainfall intensities 108

Figure 6-4 Normalized averaged gross erosion and sediment yield for the uniform conditions case 111

Figure 6-5. Normalized averaged gross erosion and sediment yield for the distributed conditions case 111

Figure 6-6. Percentage of the watershed with net erosion, deposition, and zero net erosion for the case a) of the uniform conditions (15 mm/h rainfall intensity) and b) distributed conditions (event 1) 113

Figure 6-7. Uniform case net erosion [tons/ha] at 30-, 90-,150-, 210-, 270- and 330-m 114

Figure 6-8. Distributed case net erosion [tons/ha] at 30-, 90-, 150-, 210-, 270- and 330-m 115

Figure 6-9. Sediment delivery ratio of Goodwin creek watershed for a) total sediment load, b) sand load, c) silt load and d) clay load as a function of grid size and rainfall intensity for the uniform case 116

Figure 6-10. Sediment delivery ratio of Goodwin creek watershed for a) total sediment load, b) sand load, c) silt load and d) clay load as a function of grid size for the distributed case 117

Figure 6-11. Boyce observed sediment delivery ratio values and simulated ones with CASC2D-SED 119

LIST OF TABLES

Table 3-1. Particle mean diameter and fall velocity ____ 54

Table 4-1. Goodwin Creek soil descriptions (Blackmarr, 1995) ____ 61

Table 4-2. Land use/cover in Goodwin Creek watershed ____ 62

Table 4-3. Selected stream gauges location ____ 65

Table 4-4. Averaged channel depths, and widths ____ 66

Table 4-5. DEM statistics for 30-, 90-, 150-, 210-, 270-, and 330-m resolution ____ 70

Table 4-6. Computed and documented (Blackmarr, 1995) basin and sub-basins areas ____ 73

Table 4-7. Channel minimum contributing threshold area, stream length and drainage density for each of the grid cell sizes ____ 74

Table 5-1. Rainfall events used for calibration (event 1) and validation (event 2 and 3) ____ 80

Table 5-2. Soil infiltration and erosion parameter values for the calibration run ____ 81

Table 5-3. Land use parameter values for the calibration run ____ 81

Table 5-4. Percentages of eroded sediment remaining in the overland and channel cells and leaving the watershed ____ 89

Table 6-1. Soil parameter values for an impervious silty loam soil ____ 105

Table 6-2. Land use parameter values for pasture ____ 105

LIST OF SYMBOLS

Symbol		Description
A	=	channel flow cross-sectional area $[L^2]$
$BMvol_i$	=	volume of size fraction i found in the bed-material $[L^3]$
C	=	cropping management factor in the USLE equation [--]
CD	=	drag coefficient [--]
C_i	=	suspended size fraction i concentration by volume $[L^3\ L^{-3}]$
Cwi	=	sediment concentration by weight of size fraction i $[M^3\ M^{-3}]$
d_m	=	distance from element (j,k) to m-th rain gauge located at (jrg,krg) [L]
ds_i	=	size fraction i diameter [L]
dt	=	time step [T]
e	=	excess rainfall rate (i-f) $[L\ T^{-1}]$
E_{30}	=	gross erosion simulated at 30-m $[M\ L^{-2}]$
E_x	=	gross erosion simulated at x-m $[M\ L^{-2}]$
f	=	infiltration rate $[L\ T^{-1}]$
F	=	total infiltrated depth [L]
g	=	gravitational acceleration $[L\ T^{-2}]$
G	=	specific gravity of sediment [--]
h	=	water depth [L]
H_f	=	capillary pressure head at the wetting front [L]
i	=	rainfall intensity $[L\ T^{-1}]$
I	=	interception depth [L]
$i^t(j,k)$	=	rainfall intensity in element (j,k) at time t $[L\ T^{-1}]$

K	=	erodibility factor in the USLE equation [--]
K_s	=	saturated hydraulic conductivity $[L\ T^{-1}]$
M_d	=	soil moisture deficit [--]
n	=	Manning roughness coefficient [--]
N	=	number of cells defining a basin
NRG	=	total number of rain gauges
P	=	conservation practice factor in the USLE equation [--]
P_i	=	percentage of fraction i in the parent material
Q	=	flow discharge $[L^3\ T^{-1}]$
q	=	unit flow discharge $[L^2\ T^{-1}]$
q_l	=	lateral inflow rate per unit length $[L^2\ T^{-1}]$
q_s	=	sediment unit discharge $[L^2\ T^{-1}]$
qs_{ADVi}	=	volume of size fraction i transported by advection $[L^3\ T^{-1}]$
Qs_{BMi}	=	volume of size fraction i transported from the bed-material $[L^3]$
Qs_{EROSi}	=	volume of size fraction i eroded from the parent material $[L^3]$
Qs_{SUSi}	=	sediment size fraction, i, transported by advection in channels $[L^3]$
q_x	=	unit flow rate in the x-direction $[L^2\ T^{-1}]$
q_y	=	unit flow rate in the y-direction $[L^2\ T^{-1}]$
R_h	=	channel hydraulic radius [L]
SDR	=	sediment delivery ratio [--]
S_f	=	friction slope $[L\ L^{-1}]$
$S_{f(x,y)}$	=	friction slopes in the x- and y-direction, respectively $[L\ L^{-1}]$
So	=	surface slope $[L\ L^{-1}]$

$S_{o(x,y)}$ = bed slopes in the x- and y-direction, respectively [L L^{-1}]

$SusVol_i$ = Size fraction i suspended volume [L^3]

t = time [T]

totXSScap = total excess capacity [L^3]

u,v = average velocities in the respective x- and y- directions [L T^{-1}]

V = depth-averaged velocity [L T^{-1}]

x = grid cell size [L]

$XSScap_i$ = excess capacity to carry size fraction i channel bed-material [L^3]

Y_{30} = sediment yield simulated at 30-m [M L^{-2}]

Y_x = sediment yield simulated at x-m [M L^{-2}]

φ = total soil porosity [--]

θ_e = effective porosity [--]

θ_i = soil initial moisture content [--]

θ_r = residual saturation [--]

ρ = fluid density [M L^{-3}]

ρ_s = particle density [M L^{-3}]

ω_s = particle settling velocity [L T^{-1}]

CHAPTER 1

INTRODUCTION

1.1. PROBLEM STATEMENT

Erosion and sedimentation embody the processes of detachment, transport, and deposition of soil particles. Erosion and subsequent deposition can be major problems. Erosion reduces productivity of cropland and sediment degrades water quality and may carry soil-absorbed polluting chemicals. Deposition in irrigation canals, stream channels and reservoirs reduces the capacity of the structure and requires costly removal (Foster *et al.*, 1980).

The processes controlling sediment detachment, transport and deposition are complex and interactive. This complexity results in the term "erosion process" in erosion models. The difficulty in the observation and measurement of the erosion processes during a runoff and erosion event due to small temporal and spatial scales makes necessary the use of an erosion model in the prediction of erosion and deposition in a watershed. The use of an erosion model and "after the fact" erosion observations are the best tools in resource management (Lane *et al.*, 1988).

Ideally, an erosion model should represent the essential mechanisms controlling erosion, and the model parameters should be directly related to measurable physical properties (Lane *et al.*, 1988). A two-dimensional (2-D) model is more appropriate than a one-dimensional (1-D) model when identifying those areas more sensitive to soil erosion (Hong and Mostaghimi, 1997).

Over the last two decades, the dramatic increase in computer power has led to significant developments in the way that hydrological research and operations are conducted. In this context, Geographical Information Systems (GIS) applications in environmental modeling have proliferated to take advantage of the spatial data representation capabilities, and GIS systems have been linked to process-based models (Gurnell and Montgomery, 1998). In hydrology, 2-D physical models are replacing simple 1-D representation of surface flow processes in catchment models. These add to the physical realism of existing models but at the same time require more field data collection to generate the necessary validation, calibration and parameterization data. It also needs better computational tools and techniques. The space, time and dimensional resolution at which such models may be applied has reached a stage where typical simulations can involve many thousands of grid cells, simulated for hundreds of thousands of time steps in up to three dimensions (Bates and Lane, 1998). In erosion models, 2-D studies enable the identification of vulnerable regions within a watershed, thus facilitating the improvement in the planning of soil conservation systems.

Digital Elevation Models (DEMs) have facilitated various hydrologic applications such as watershed hydrologic simulations, delineation of stream patterns, terrain catchment partitioning and erosion/deposition predictions. While GIS provides a rapidly expanding tool-kit that will allow hydrological scientists and engineers to address new questions, the development of such applications will inevitably raise a variety of interesting new problems (Gurnell and Montgomery, 1998). With the availability of high resolution DEMs, one such issue is the optimum resolution at which data is input in hydrological models.

In a raster map, spatial resolution increases as the size of grid cells decreases. There are some trade-offs when using a small resolution digital model versus using one with coarser resolution. Computational time, computer storage, level of accuracy and data manipulation are the most important factors that will determine the cell size selection. For a given area, a change to grid cells half the current size requires as much as four times the storage space, depending on the type of data and the storage techniques used (ESRI, 1994). The selection of the cell size will depend on the established objectives and the resources availability to model the watershed system.

It has been observed how the cell size in a DEM affects the landscape representation like slopes, slope lengths, and the extracted channel network in a basin (Zhang and Montgomery, 1994; Bruneau *et al.*, 1995; Yu, 1997; Wolock and McCabe, 2000). The change of these physical variables will translate into a change on hydrological model outputs. The cell size in distributed hydrological models will have a direct effect on the information content and the accuracy of the simulated output. In general, researchers agree that the finer cell size gives more accurate results. However, there is no consensus on the selected cell size and the recommended one depends in each study on the used model, initial conditions and assumptions, magnitude of the rainfall event and watershed and objective of the study.

Investigation of the effect of the input DEM resolution on the prediction of erosion has been very limited until now. Studies of how a change of the grid cell size affects the upland erosion prediction are almost non-existent (Julien and Frenete, 1987; Kienzle, 1996; Molnar and Julien, 1998). These studies are limited to the application of the Universal Soil Loss Equation (USLE), which estimates average annual soil loss but fails to predict deposition. The USLE was originally developed for agricultural fields and its application to landscape scale erosion modeling is often inappropriate (Foster and Wischmeier, 1974; Moore and Wilson, 1992; Mitasova *et al.*, 1996).

The selection of the grid cell size affects empirical erosion models such as the USLE through a change of the landscape characteristics. Process-based erosion models will be affected by both the change in landscape representation and the change in the hydrological model. The effects of grid cell size on the prediction of erosion and deposition rates when using a 2-dimensional, physically-based hydrological model have not yet been investigated.

1.2. Research Objectives

The objectives of this research are:

- Extended development of the CASC2D-SED upland erosion and sediment transport and implementation of a suitable channel sediment routing algorithm

- Coupling of the model time-series output grids with a GIS that will allow the geovisualization of sediment transport dynamics by size fraction and the spatial erosion and deposition patterns.
- Investigation of the effects of grid resolution on the simulated values of gross erosion, sediment yield, spatial distribution of net erosion and sediment delivery ratios.

1.3. METHODOLOGY AND APPROACH

The 2-D physically-based computer model CASC2D-SED is a hydrological model designed to simulate the erosion rates in a watershed in a grid basis. This model is used in this study to investigate the effects of the grid cell size on the prediction of the sediment production in a watershed.

Data are taken from the 20.5-km^2 Goodwin Creek watershed (MS). Extensively monitored by the ARS National Sediment Laboratory (NSL), this watershed has been chosen because it has a complete database compiling precipitation, runoff, channel cross-section and sediment discharge.

The current CASC2D-SED version of upland sediment transport and deposition (Johnson, 1997) will be modified to include the transport of sediment by advection. To calibrate and validate the model, a channel sediment transport module will be implemented in CASC2D-SED. Eroded sediment on the hillslope will be simply routed in the channels by size fraction using the Engelund and Hansen (1967) equation. While erosion is not going to be allowed in the channel, deposition may occur.

Coupling of the CASC2D-SED model output with GIS will aid in the testing phase of the model. Output grids time-series visualization of the simulation of the sediment transport dynamics and geovisualization of the net erosion spatial distribution patterns will be used as a form of model validation. The program Arc/Info[1] is going to be used for input data preparation, analysis and visualization of the time-series output grids. Results are to be presented for a given event and size fraction in a animated MPEG file.

Goodwin Creek GIS data at 30-m will be resampled to up to 330-m grid cell size to study the effects of spatial resolution on simulated erosion rates in a watershed. First, the gross erosion and sediment yield produced at the Goodwin Creek Watershed at a given resolution will be quantified for uniform impervious soils and uniform land use parameters using three different constant uniform rainfall intensities. Later, a real event will be simulated and the differences in hydrological and sediment outputs presented for different grid cell sizes. Results will be presented in terms of gross erosion, sediment yield, spatial distribution of net erosion and sediment delivery ratios.

---

[1] Arc/Info® is a registered trademark of the Environmental Systems Research Institute

CHAPTER 2

LITERATURE REVIEW

"The need for estimates of sediment yield are ubiquitous throughout water resources analyses, modelling, and engineering as sediment is a major pollutant, a transporter of pollutants, and sedimentation rates and amounts determine the performance and life of reservoirs, canals, drainage channels, harbors, and other downstream structures and improvements."

Leonard J. Lane et al. (1997)

According to the objectives, the following topics are reviewed in this chapter: a) soil erosion modeling, including the motivation and limitations, existing erosion models, and the importance of GIS to soil erosion modeling, b) geovisualization and time series animation techniques are described and their advantages as powerful data exploration techniques are enumerated and, finally, c) the last section revisits the findings of the grid cell size effects on the landscape representation, simulated flows and soil erosion predictions.

2.1. Soil Erosion Models

2.1.1. Rationale and Limitations

Soil erosion and sedimentation by water involves the processes of detachment, transportation, and deposition of sediment by raindrop impact and flowing water (Foster and Meyer, 1977; Wischmeier and Smith, 1978; Julien, 1995). Erosion and sedimentation represent on-site and off-site problems. On-site, erosion can degrade the productivity of the soil necessary for crop and food production (Pimentel *et al.*, 1976 and 1995; Meyer *et al.*, 1984; Kimberlin and Moldenhauer, 1977). Off-site, sediment deposited in stream channels, reservoirs and other hydraulic structures reduces their capacity and needs to be removed. Sediment is a pollutant by itself and may carry soil-absorbed pollutants (Stone, 2000; Lane et al, 1983; Hadley and Ongley, 1989;). Soil conservation specialists have for many years attempted to estimate soil loss from individual fields or slopes to determine land use practices which will ensure long-term productivity of the soil (Walling, 1982; Hadley, 1986; Hadley and Mizuyama, 1993;

Walling and Webb, 1996; Walling and Probst, 1997; Summer *et al.*, 1998). The ultimate objective of research on soil erosion is to solve erosion problems by adopting suitable conservation measures (Painter, 1981). Location of the sediment source areas, volume of sediment eroded, transported and delivered and rate of sediment delivery and transport after treatment are some of the factors that should be quantified before deciding what erosion control treatment is best suited for a problem (Madej, 1981). In this context, erosion models and, more specifically, 2-D erosion models provide the necessary tool to predict excessive soil loss and help in the implementation of an erosion control strategy (Julien and Frenette, 1987; Sharma *et al.*, 1996; Doe *et al.*, 1996; Marston and Dolanb, 1999).

The measurement of erosion rates is one of the limitations of erosion modeling. While direct surface measurements are preferable, these are often difficult to obtain from a logistical, observational and instrumental point of view. Many land surfaces change only imperceptibly and only highly accurate measurements over long periods of time may produce meaningful results (Campbell, 1981).

A second limitation comes at the erosion modeling stage. Since the erosion components are subsidiary to a hydrologic model, the difficulties encountered when modeling rainfall-runoff relationships are magnified when modeling erosion and sediment production (Bennet, 1974; Clarke, 1994; Rojas and Woolhiser, 2000). Because the calculation of sediment-transport rate in most cases is dependent on the prior estimation of the hydraulics of overland flow, any errors in the predictions of the hydraulic will propagate through the erosion estimation process (Wainwright and Parsons, 1998).

Nevertheless, due to the complexity of the natural systems, watershed erosion modeling represents a powerful tool to predict the consequences of natural as well as man-induced environmental changes and impacts on the sediment dynamics.

2.1.2. Review of Existing Erosion Models

Soil erosion computer models use mathematical expressions to represent the relationships between various factors and processes occurring on the landscape. These factors generally include topography, meteorological variables, soil properties, and land use and land cover features. One classification of models distinguishes between theoretical or physically based models and empirical models. However, most erosion models are of a hybrid type including both theoretical and empirical components (Haan *et al.*, 1994). The emphasis in erosion research on strictly empirically based models is declining. The trend in erosion prediction technology is toward the development of process-based simulation models (Morgan 1980; Nearing *et al.*, 1990). Many physically based upland erosion models in the past have used lumped or semi-distributed techniques for their hydrologic modeling component.

Most of the erosion models are based on the Universal Soil Loss Equation (USLE) (e.g. AGNPS (Young *et al.*, 1989), ANSWERS (Beasley *et al.*, 1989), EPIC (Sharpley and Williams, 1990), and SWAT (Arnold *et al.*, 1996)), on partitioning of the watershed into planes and channel elements (i.e. KINEROS (Woolhiser *et al.* 1990), and EUROSEM (Morgan, 1990)) or they are not intended for basin-scale use (i.e. CREAMS (Knisel, 1995)). An extensive description and discussion on these models can be found in Singh (1995), and Doe *et al.* (1999).

The USLE (Wischmeier and Smith, 1978) is the most widely used and accepted empirical soil erosion model. It was developed for sheet and rill erosion based on a large set of experimental data from agricultural plots. Yet, the equation was derived on single agricultural plots and is only valid when applied to an area up to 1 ha. The USLE equation takes into account slope length (L factor), steepness (S factor), climate (R factor), soils (K factor), cropping (C factor) and management (P factor). This model was specifically designed and tested to predict the average annual soil movement from a given field plot under specified land use and management conditions. The USLE has been enhanced during the past 30 years by a number of researchers. MUSLE (Williams, 1975), RUSLE (Renard *et al.*, 1996; Stone *et al.*, 2000), ANSWERS (Beasley *et al.*, 1989) and USPED (Mitasova *et al.*, 1996) are based on the USLE and represent an improvement of the former. The use of the USLE and its derivatives is limited to the estimation of gross erosion, and lack the capability to compute deposition along hill slopes, depressions, valleys or in channels. Moreover, the fact that erosion can occur only along a flow line without the influence of the water flow itself restricts direct application of the USLE to complex terrain within GIS. The history of the development of the USLE and its modifications can be found in Peterson *et al.* (1979), Lane *et al.* (1992) and Renard *et al.* (1997).

Developed by the USDA, the WEPP model (Flanagan and Nearing, 1995) is intended to replace the USLE family of models and expand the capabilities for erosion prediction in a variety of landscapes and settings. This model is a physically based, distributed parameter, single-event simulation erosion prediction model. Processes within the model include erosion, sediment transport and deposition across the landscape and in channels via a transport equation.

The KINEROS model (Woolhiser et al, 1990; Smith *et al.*, 1995) is a single-event, physically based model that uses the Smith/Parlange infiltration model and the kinematic wave approximation to route overland flow and sediment. The EUROSEM model (Morgan, 1990) is a single-event, physically based model for predicting soil erosion by water from fields and small catchments. There are several sediment transport equation options implemented in both models. The representation of a catchment by a cascade of plane and channel elements in the WEPP, KINEROS, and EUROSEM models necessitates lumping of some parameters for small areas. This represents a drawback when representing large watersheds.

SHESED (Wicks and Bathurst, 1996) is based in the European raster SHE model. It simulates soil erosion by raindrop impact, leaf drip and sheet overland flow. Eroded material is transported across the landscape via overland flow. Erosion in channels is modeled as bed erosion. The sediment routine can handle the transport of fine and coarse material. The SHESED model has been applied successfully in Europe and the USA. The LISEM model (De Roo, 1996a; De Roo *et al.*, 1996a, 1996b) is a physically based hydrological and soil erosion model that is completely incorporated in a raster GIS. It can simulate splash erosion and rill and inter-rill erosion. The transport capacity of overland flow is modeled as a function of the unit stream power (Govers, 1990) and depends on the value of the median grain size.

Developed at CSU, CASC2D-SED (Julien and Sagahfian, 1991; Sagahfian, 1993; Julien *et al.*, 1995; Ogden, 1997a, 1997b; Johnson, 1997; Ogden, 1998; Johnson *et al.*,

2000) is a physically-based, distributed, raster, hydrologic and soil erosion model that simulates the hydrologic response of a watershed subject to a given rainfall field. Input rainfall is allowed to vary in space and time. CASC2D-SED can simulate the upland sediment production and deposition by size fraction at any point in a watershed. The sediment is transported from one cell to the next using the 2-D diffusive wave overland flow routine scheme present in CASC2D. For a full description of the CASC2D-SED model, see Chapter 3.

2.1.3. Soil Erosion Modeling and GIS

The ability to represent elevation in terms of topographic surfaces is central to geomorphological analyses and thus to the importance of representing topography using DEMs. It is through the distribution of soil that the land surface changes over the long term and so the ability to link sediment transfer with DEM changes (Schmidt, 2000). The redistribution of sediment will drive the long-term landscape change, which in turn will affect the hydrological processes acting within and over individual hillslopes (Brooks and McDonnel, 2000).

Soil erosion is influenced – among other factors – by the spatial heterogeneity in topography, vegetation, soil properties, and land use. All too often, however, predicative soil erosion models do not examine the problem in a spatial context. This is where a Geographical Information System (GIS) becomes a valuable tool. A GIS is "an organized collection of computer hardware, software, geographic data, and personnel designed to efficiently capture, store, update, manipulate, analyze, and display all forms of geographically referenced data" (ESRI, 1994). Therefore, a GIS is a useful technique for coping with the vast number of spatial data and the relation between data from various sources in the erosion modeling process. The advantages of linking soil erosion models with a GIS include the following:

(1) The possibility of rapidly producing input data to simulate different scenarios. A GIS provides an important spatial/analytical function performing the time-consuming georeferencing and spatial overlays to develop the model input data at various spatial scales (Sharma *et al.*, 1996).

(2) The ability to use very large catchments with many pixels, so the catchment can be simulated with more detail (De Roo, 1996b).

(3) The facility of displaying the model outputs (visualization). Visualization can be used to display and animate sequences of model output images across time and space. Therefore, visualization enables object to be viewed from all external perspectives, and to invoke insight into data through manipulable visual representations (Tim, 1996).

There are different strategies for linking erosion models with GIS, ranging from loosely coupled to tightly coupled arrangements. Pullar and Springer (2000) categorize three levels of integration:

(1) Loose coupling: the GIS system and the erosion model are separated, and the files must be transferred back and forth externally between the GIS and the model.

(2) Tight coupling: the GIS (typically) provides the shared interface to move the spatial data between the GIS and the separated modeling program.

(3) Fully integrated: the model is fully integrated as a component in the host GIS application.

Most of the current integrations of soil erosion models with GIS are examples for level 1 and 2 types. Category 3 type linkages are problematic because of the lack of an efficient temporal dimension in most GIS systems (Kaden, 1993; Doe, 1999).

Erosion potential prediction is becoming a more and more widely applied GIS operation. Several examples for the integration of GIS with erosion models can be found in the literature: De Roo *et al.* (1989) combined ANSWERS with GIS technology; later, De Roo integrated LISEM (De Roo, 1996a,b; De Roo *et al.*, 1996a, 1996b) and LISFLOOD (2000) with a GIS; Mitchell *et al.* (1993) linked AGNPS with GIS. Other integration examples are described in Mallants and Feyen (1989), Lin (1991), Dryer and Frhlich (1994), De Jong (1994), Mitasova *et al.* (1996), Harrison and Doe (1997), Adler *et al.* (1997), Warren *et al.* (2000), Wijesekera and Samarakoon (2001)

2.2. GEOVISUALIZATION AND TIME-SERIES ANIMATION.

The ever-changing natural world involves the interaction between many environmental components. It is becoming increasingly important to understand the processes behind these changes to be able to better manage the anthropogenic effects on the environment. The exploration of complex spatio-temporal environmental data demands creative methods of analysis and assessment (Lane and Richards, 2001). Thus, the importance of the visualization of all environmental data to facilitate improved understanding of the data and better predictions through modeling.

Visualization of geographical information has been termed geovisualization (Thurston, 2001). **Geovisualization** is a powerful data exploration technique, exploiting the ability of current computing technology to analyze and display dynamically large amounts of information (Edsall *et al.*, 2000). Geovisualization involves the use of computer graphics to stimulate the human visual system to recognize patterns that would not otherwise be obvious (APOALA project, 2001). Geovisualization refers not only to a set of graphical images but also to the interactive process of visual thinking and interaction with the images (Dibiase *et al.*, 1992a, 1992b).

Scientific research can be described as a process in which a scientist receives an insight into an investigated phenomenon, verifies a derived hypothesis and finally communicates the results. Geovisualization is useful for exploration analysis (MacEachren and Kraak, 1997, 2001; Uhlenküken *et al.*, 2000; Blaser *et al.*, 2000; Steiner, 2001) and the communication of knowledge about spatial information (Kraak and Ormeling, 1996). Lane and Richards (2001) list visualization as an alternative type of model assessment allowing the modeler to reproduce in time and space processes that can be observed through experimentation or in the file, but not necessarily readily measured. They further state that while convectional empirical validation relies on measurement, visualization relies on observation and perception.

Visualization techniques are applied to the scientific research in two ways: a) as visual thinking or cognitive visualization, for exploration and verification of spatial data, and b) as visual communication, where the scientist communicates with other people to present and discuss results (Dransch, 2000).

Visualization techniques have been used widely in a variety of cases, i.e. Kreuseler (2000) used virtual 3-D scenes to explore oceanographic ecosystems and Fuhrmann (2000) as a system for visualizing hydrological data.

Time-series animation is a visualization technique ideally suited for displaying temporal geographic data. Animation leads to faster and easier comprehension of spatial patterns, trends and rates of change of data: "Just as a picture is worth a thousand words, an animation is worth much more than the input static images" (Acevedo and Masuoka, 1997). Dykes (1997) states that while dynamic maps are particularly appropriate for exploratory analysis, short-term, slightly different, views of a data set are an essential part of the analytical process.

Frame-based raster animations are produced by showing a series of images one after the other in temporal order (Peterson, 1995). When creating a time-series animation one must consider the following issues:

(1) Number of data frames: the greater the number of frames, the better the representation of the process. Correspondingly, the more the number of frames, the bigger the animated file will be.

(2) Starting and ending time of the representation: to capture only the most significant sequence of the animation.

(3) Animation display speed: this will depend on many factors such as understanding of human visual perception and cognition, the intended purpose or objective of the animation, and the rate of change within the data itself.

(4) Media where the animation will be played: on computer screens, recorded to video tape, or put on the World Wide Web (Dykes, 1997; Cartwright, 1997).

Time-series animation techniques has been used, among others, for predictions from an orographic precipitation model (Hay and Knapp, 1996); representation of model uncertainties (Ehlschlaeger *et al.*, 1997; Davis and Keller, 1997; Evans, 1997); the visualization of urban growth (Acevedo, 1997); animation of time-varying three-dimensional groundwater model output (Predmore, 1999); support of development and evaluation of a soil erosion model (Mitasova *et al.*, 1996; Mitas *et al.*, 1997; Mitasova *et al.*, 2000); the properties of spatial and temporal periodicity (Edsall *et al.*, 2000); fluxes of materials in the coastal zone (Morris, 2000); and the representation of statistical sociological data (Adrienko *et al.*, 2000).

2.3. Cell Size Effects

2.3.1. On Landscape Representation

Topography defines the movement of water in a watershed and therefore it greatly controls fluxes of energy, nutrient distribution and mass movement in a watershed. Landscape representation will affect the hydrologic response of a watershed at any scale (Schmidt, 2000) and thus it becomes an important factor in hydrological modeling. Characteristics such as slope angle and upslope drainage area will affect the water flow-path geometries, flow velocities and channel network (Thieken, 1999). The hillslope form and soil properties will dominate the runoff production at the hillslope scale. On the basin scale, the hydrograph is influenced by the catchment height distribution, length and form of the basin, and parameters describing the drainage network.

Topography is one of the factors affecting the hydrologic response of a watershed and so inaccuracies in the topographic model will propagate into errors in the prediction of the hydrologic output (Lagacherie *et al.*, 1996; Lee and Chu, 1996). One of such inaccuracies derives from cell aggregation in a raster model.

Resampling is the process of determining new values for cells in an output grid that result from applying some geometric transformation to an input grid. Spatial data aggregation or resampling to coarser scales is often used in the environmental sciences to "scale-up" from local to regional or global scales when modeling a system (Bian and Butler, 1999). The aggregated data is referred to as having a coarser resolution or, in the specific case of raster data, as having a larger grid spacing or larger grid cell size.

Arc/Info (ESRI, 1982-2001) GRID module utilizes resampling for geometric transformations and for converting grids to other resolutions (ESRI, 1994). Some techniques have been developed to determine the output value depending on where the point falls relative to the center of cells of the input grid and the value associated with these points (Richards, 1993). The three processes for determining output values in GRID are nearest neighbor, bilinear interpolation, and cubic convolution.

The nearest neighbor is the resampling technique of choice for categorical data as it does not alter the value of the input cells. Once the location of the cell's center on the output grid is located on the input grid, the nearest neighbor assignment will determine the location of the closest cell center on the input grid, identify the value that is associated with the cell, and assign that value to the cell that the output cell center is associated with. The bilinear interpolation and the cubic convolution uses respectively the 4 and 16 nearest input cell centers and their values to determine the output cell value. The new value for the output cell is a weighted average determined by the value of the four or 16 nearest input cell centers and their relative position or weighted distance from the location of the center of the output cell in the input grid. The last two methods should not be used on categorical data since the categories will not be maintained in the output grid. The bilinear and cubic interpolation methods also tend to smooth the data.

Zhang and Montgomery (1994) found that the spacing of the original data used to construct a DEM effectively limits the resolution of the DEM. For that purpose, they obtained digital elevation data from low altitude aerial photographs using a stereo digitizer at a density about every 10 m. Then, they gridded spot elevation data at scales of 2, 4, 10, 30, and 90-m using the GRID module of Arc/Info. In their study, they found that decreasing the cell size beyond the resolution of the original survey does not improve the accuracy of the DEM but potentially can introduce interpolation errors. In their case study, cell sizes smaller than 10-m gave only marginal improvement in slope representation.

Kienzle (1996) studied the effect of the grid size on the division of a catchment in South Africa into individual terrain units (valley bottom, footslope, midslope, scarp and crest). The simulation of terrain units with DTMs was shown successful with the 100- and 50-m grids, both spatially and proportionally. The 250-m grid gave good proportional results, but performed weaker with respect to spatial representation. The simulation with the 400-m grid failed to represent terrain units.

Terrain slope values are usually computed from DEM gridded data by using elevations of the four immediately adjacent pixels (Ritter's algorithm, 1987) or from the eight surrounding elevations in a 3x3 window (Sharpnack and Akin's method, 1969). Slope is an important variable affecting processes such as water flow routing and erosion rate predictions in a watershed. Slope mapping can be affected by the used computational algorithm, DEM resolution, topographic complexity and DEM data quality.

Zhang and Montgomery (1994) found that the percent of a catchment steeper than a given slope systematically decreases as the DEM cell size increases and that this effect is larger for steeper terrain. They concluded that since the slope of a grid cell represents the average slope for the area covered by the cell, an increase in the cell size will decrease the ability of the model to describe the characteristics of the steeper and more dissected topography. Wolock and McCabe (2000) agree in that the smoothing effects are greatest on DEMs with the most amount of terrain information, the ones with short terrain features and steep slopes.

Analyzing six independently created DEMs from 1/10,000 contour data (20- to 100-m cell size), Bruneau *et al.* (1995) found that the relative frequency of low slope angles decreased for increasing cell sizes and vice-versa. Observed minimum and maximum elevation values were higher and lower respectively as the cell size increased.

Yu (1997) examined the effect of DEM grid cell size on the land surface representation. He found that the range of slopes increases as the cell size decreased and that the slopes distributions for cell sizes less than 300-ft did not differ from the distribution for 300-ft.

In general, the following relation has been observed between DEM spatial resolution and the computed slope angles (Kienzle, 1996; Gao, 1997; Chang and Tsai, 1991; Molnar, 1997):

(1) Derived slope values decrease with lower grid resolutions. This effect of DEM resolution is more apparent along topographic discontinuities such as valleys, ridges, and minor landforms.

(2) Slope mean and standard deviation decrease with increasing grid cell size.

(3) Coarsening of the DEM resolution causes the disappearance of extremely large gradients and the reduction in gradient range for simple and complex terrain units alike. This effect results in a change in slope distributions with decreased skewness and values aggregated around a lower mean.

DEM spatial resolution also affects the derived channel network. In general, the number of channel links, total channel length, link slope and drainage density decrease with increasing grid size (Garbrecht and Martz, 1994; Thieken, 1999; Wang, 2000). This decrease is the result of:

(1) The inability of a coarse grid to reproduce the channel sinuosity and

(2) The channel and drainage area capturing effect, i.e. the DEM grid can no longer resolve the separation between channels or drainage boundaries.

2.3.2. On Simulated Flows

The results of hydrological models are not independent of the DEM grid resolution used in the model formulation. In distributed models, the cell size will have a direct effect on the information content and the accuracy of the simulated output. In general, researchers agree in that the finer cell size gives more accurate results. However, there is no consensus in the selected cell size and the recommended one depends in each study on the used hydrological model, initial conditions and assumptions, magnitude of the rainfall event and watershed and objective of the study.

The cell size effect on the output of the a hydrological model has been most studied for the TOPMODEL framework (Beven and Kirkby, 1979). Quinn (1991), Wolock and Price (1994), Zhang and Montgomery (1994), Bruneau *et al.* (1995) and Saulnier *et al.* (1997) studied the effect of grid cell size to the computed topographic characteristics, wetness index and outflow.

The topographic index (α / tan β) is an important component of many physically based geomorphic and hydrologic models. It reflects the spatial distribution of soil moisture, surface saturation, and runoff generation processes. The variable α represents the specific catchment area, which is the area upslope from a specified contour segment divided by the length of the contour segment. *Tan* β is the slope. TOPMODEL relies on such a topographic index. Due to increased specific catchment area and decreased slope values with increasing cell sizes, the topographic index increases with coarser resolution DEMs (Zhang and Montgomery (1994), Wolock and Price (1994), Bruneau (1995), Saulnier (1997),Wolock and McCabe (2000)). As a consequence, the magnitude of the peak runoff rate and therefore the runoff volume predicted are higher for the coarser resolution. Zhang and Montgomery (1994) found that this change was larger for the case of the watershed with steeper slopes and that the effect of the cell size on computed hydrographs depends on both rainfall intensity and initial base flow. They concluded (page 1026) that "the runoff processes are governed neither by the finest, nor by the coarser resolution" and they suggest that "the most appropriate DEM cell size for topographically driven hydrologic models is somewhat finer than the hillslope scale identifiable in the field". While a 10-m cell size represented a significant improvement over 30-m or coarser cell sizes, finer cell sizes provided relatively little additional resolution.

Yu (1997) examined the effect of DEM grid cell size on the land surface representation and hydrologic simulation using the Basin-Scale Hydrologic Model (BSHM). In this model, an algorithm with unidirectional flow is used in the calculation of the spatial distribution of the runoff travel time and the prediction of simulated hydrographs for different cell sizes. In his study, infiltration along the overland flow path was not considered. Using a digital elevation generator (DEG) in BSHM, Yu produced DEMs of a 555-mi^2 basin with a regular grid ranging from 3600 to 120 feet. He found that the resolution of patterns of calculated runoff travel time improves as the cell size decreases. For cell sizes smaller than 300-ft, the frequency distribution of travel times and peak flows were almost identical. They considered the 600-ft spacing to be efficient in terms of CPU time and the quality of the simulation.

KINEROS (Woolhiser *et al.*, 1990) is a event-based, distributed model that calculates surface runoff and erosion in small agricultural an urban watersheds. Using the KINEROS model, Thieken *et al.* (1999) observed that watershed configurations generated with the same contributing threshold area but from different DEMs produce an increase in runoff volume as well as in peak flow rate for a given storm as cell size increases. The time to peak was observed to decrease as the DEM resolution was increased. They were able to correlate drainage density and areal mean lengths with outflow depth and conclude that both descriptors strongly influence the hydrological modeling results.

Using a spatially-explicit, variable source area hydrology model, Kuo *et al.* (1999) studied the effect of different cell sizes on soil water content and the sensitivity of the model to aggregation of specific types of input data by increasing the grid cell size. Their model works in areas of impermeable sloping sub-soils at shallow depth and includes unsaturated flow. They tested elevation, soil and land use digital models with grid cell sizes ranging from 10- to 600-m. They found that the spatial variability of soil type and land use were not affected by the grid cell size. They also found that while the spatial variability of the slope decreased slightly, the Laplacian (or curvature of the landscape) decreased greatly. This resulted in higher soil water content and higher evaporation rates for large grid cell sizes. The runoff in a wet year was not affected by the grid cell size but for a dry year it was greatest for smallest grid cell size. Kuo *et al.* (1999) and Beven (1995) agree that the aggregation approach using averaged parameter values toward macroscale modeling is inadequate for representing hydrological processes at large scale.

Although CASC2D has been applied to a wide range of grid sizes, Ogden (1997b) has found that grid cell sizes smaller that 200-m produce more robust calibrations. Ogden (1997a) further states that the selection of the grid cell size will depend both on the existing relevant data and the computational effort required. Ogden (1997b) estimates that approximately 250 bytes of memory is required for each grid cell cell, thus decreasing the grid cell size will increase the computational time. At the same time "the higher the resolution, the more accurate the solution will be" (Ogden, 1997b, page 6). Grid cell size affects the computational time step in that the latter is shorter for finer grid cell sizes.

Molnár (1997) applied the CASC2D model to two basins in Mississippi of approximately 21 km^2 (Goodwin Creek) and 560 km^2 (Hickahala-Senatobia). The effects of grid cell size were evaluated using grid cell sizes ranging from 127- to 380-m in Goodwin Creek and from 30- to 914-m in Hickahala-Senatobia. Using the nearest neighbor technique, the aggregation process led to an increase in the percentage of channel cells as compared to the total number of cells. Due to the representation of the watershed at larger grid cell sizes, runoff from upland regions reached the channels more rapidly and transmitted more quickly to the outlet. As a result, coarser resolutions resulted in higher runoff rates and shorter times to peak. Molnár found that to compensate for larger grid cell sizes, overland and channel roughness coefficient had to be increased. Molnár concludes that the selected grid cell size should small enough to preserve essential characteristics governing rainfall-runoff processes, but large enough to be used effectively and efficiently.

2.3.3. On Soil Erosion Prediction

Investigation of the effect of the input DEM on the prediction of erosion has been very limited until now. Studies of grid cell size effects the erosion/deposition prediction are almost non-existent and they are limited to the application of the USLE equation. However, the USLE was originally developed to predict long term average annual erosion and thus, it is not designed to predict event erosion well. Kinnell (2000) points out how procedures available to determine the slope length factor for a cell does not enable the impact of variations in upslope runoff on cell erosion to be determined adequately.

Julien and Frenette (1987) performed studies on the Chaudiere basin (5830 km^2) in Canada in order to examine the applicability of the USLE to large areas. They were able to extend the applicability of the USLE to the large watershed by applying a correction factor. This correction factor was a function of the grid cell size. The same method was latter applied by Julien and González del Tanago (1991) in Spain, validating the use of the grid size factor. In both studies, it was concluded that unbiased estimates of soil erosion losses are obtained for grid cell sizes less than about 350 m.

Kienzle (1996) used the RUSLE (Revised USLE) to study the effect of grid cell size on the estimation of soil loss potential. It was determined that the soil loss was very sensitive to slope. Mean annual erosion values increased up to 67% when computed with the 400- and 50-m resolution grids while this increment was much larger (430%) when the maximum erosion rates were compared.

A fourth study is done by Molnár and Julien (1998) relating cell size and erosion prediction using the USLE in two basins (21 km^2 and 304 km^2). In this study, it is concluded that at grid cell sizes exceeding 100-m, a correction factor must be included in the calculations. With the use of this correction factor, the USLE can be applied directly at macroscales.

The effect of grid cell size produced when modeling rainfall-runoff processes are magnified when modeling erosion and sediment production (Rudra *et al.*, 1998; Wainwright and Parsons, 1998). One of the sources of error in the prediction of erosional / depositional processes is the accuracy of the input elevation data. DEM errors impact terrain attributes (such as slopes and slope lengths) that affect the erosion prediction the most.

The selection of the grid cell size will also depend on the established objectives and the resources availability to model the watershed system. Using the same mathematical representations of physical processes for runoff and sediment transport, Hong and Mostaghimi (1997) concluded that a 2-D model provides better representation of spatial distribution of flow depths and sediment concentration that the 1-D model. However, the 2-D model provided little improvement when simulating the total runoff volume and sediment yield at the outlet. It becomes apparent that either the 1-D or 2-D models could be used, based on the specific objectives. When using a 2-D model the same type of philosophy should be applied in the selection of the model resolution.

Holmes *et al.* (2000) suggested that although the global (average) error in a 30-m USGS DEM is small, local error values can be large and even small amounts of elevation errors can greatly affect derivative calculations. In this study, it is shown that errors in a 30-m DEM affects the prediction of the area prone to potential debris flow. Holmes *et al.* (2000) and Mitasova *et al.* (2000) agree that the error propagation is most dramatic in valley bottoms (low lands) and along streamlines. Mitasova *et al.* (2000) conclude that the computation of net erosion/ deposition as a change in transport capacity is very sensitive to artifacts in the DEM.

Molnár and Julien (1997) used the USLE equation for estimating soil loss at different grid resolutions (from 30-m to 2-km grid cell size). They found that large grid cell sizes tend to underestimate soil losses because of a decrease in the terrain slopes. They proposed a correction factor to compensate for the underestimation of soil loss when applying the USLE at macroscales.

2.4. Summary

From the literature review we have learned that:

(1) Watershed erosion modeling provides the necessary tool to predict excessive soil loss or deposition and helps in the implementation of an erosion control strategy. CASC2D-SED is a raster, physically based hydrological and soil erosion model that allows the prediction of the location of erosion source areas. This model predicts erosion rates and deposition on an event basis, taking into account the spatial distribution of the hydrological and erosional model variables.

(2) GIS is an increasingly important tool in environmental models allowing rapid data pre-processing, data input, and analyses and display of results. Applied to scientific research, geovisualization is a powerful data exploration technique utilizing the ability of the computing technology to analyze and display dynamically large amount of information

(3) In raster models, the grid cell size affects the landscape representation by affecting primarily the terrain slopes and the channel network definition. With increasing grid cell size, hydrological simulations are affected by the lost in information content and accuracy of the simulated output. Process-based erosion models, depending on both landscape representation and hydrological variables, have a magnified dependency on the input data spatial resolution.

CHAPTER 3

THE REVISED CASC2D-SED MODEL

"The future of watershed models will be shaped by increasing societal demand for integrated environmental management; growing need for globalization by incorporation of biological, chemical, and physical aspects of the hydrologic cycle; rapid advances in remote sensing and satellite technology, geographical information systems (GIS) database management systems (DBMS), and expert systems; enhanced role of models in planning and decision making; mounting pressure on transformation of models to user-friendly levels; and clearer statement of reliability and risk associated with model results."

Vijay P. Singh, 1995

The CASC2D-SED program governing equations including the newly implemented interception process are reviewed in this chapter. The revised upland erosion and sediment transport and the implementation of the channel sediment routing are described as well.

In the second part, the implementation of the CASC2D-SED output time-series grids visualization in Arc/Info® is explained.

CASC2D-SED source code and subroutine list, and input/output files are listed in Appendix I. AML programs for the output grids time-series animation are also found in Appendix I.

3.1. MODEL DEVELOPMENT

The CASCade 2 Dimensional SEDimentation (CASC2D-SED) model originally began with a two-dimensional overland flow routing algorithm developed and written in APL by Prof. P.Y. Julien at Colorado State University. The overland flow routing module was converted from APL to FORTRAN by Dr. Bahram Saghafian, then at Colorado State University, with the addition of Green & Ampt infiltration, detention storage and diffusive-wave channel routing (Julien and Saghafian, 1991; Saghafian, 1992; Julien *et al.*, 1995).

CASC2D was incorporated as part of the GRASS GIS for hydrologic simulations as r.hydro.CASC2D (Sagahfian, 1993; Sagahfian and Ogden, 1996). Later on, Ogden (1994) added implicit channel routing option to CASC2D. Since 1995, CASC2D has been reformulated with the addition of continuous simulation capabilities and other hydrological components such as interception, initial depths, evapotranspiration and redistribution (Ogden, 1997a). This last version is incorporated in the Watershed Modeling System Interface developed by Brigham Young University (2001) and is known as CASC2D for WMS. CASC2D/WMS development and integration is funded by the U.S. Army Corps of Engineers, Waterways Experiment Station. The integrated version of CASC2D-SED in WMS interface may be used only by the Corps of Engineers and in educational institutions.

The upland erosion module was added by Dr. Billy Johnson (1997) based on previous work done by Kilinc (1972) and Kilinc and Richardson (1973) at Colorado State University and was called CASC2D-SED. Figure shows the CASC2D-SED model flowchart. The new or revised part of the model is shown in bold letters.

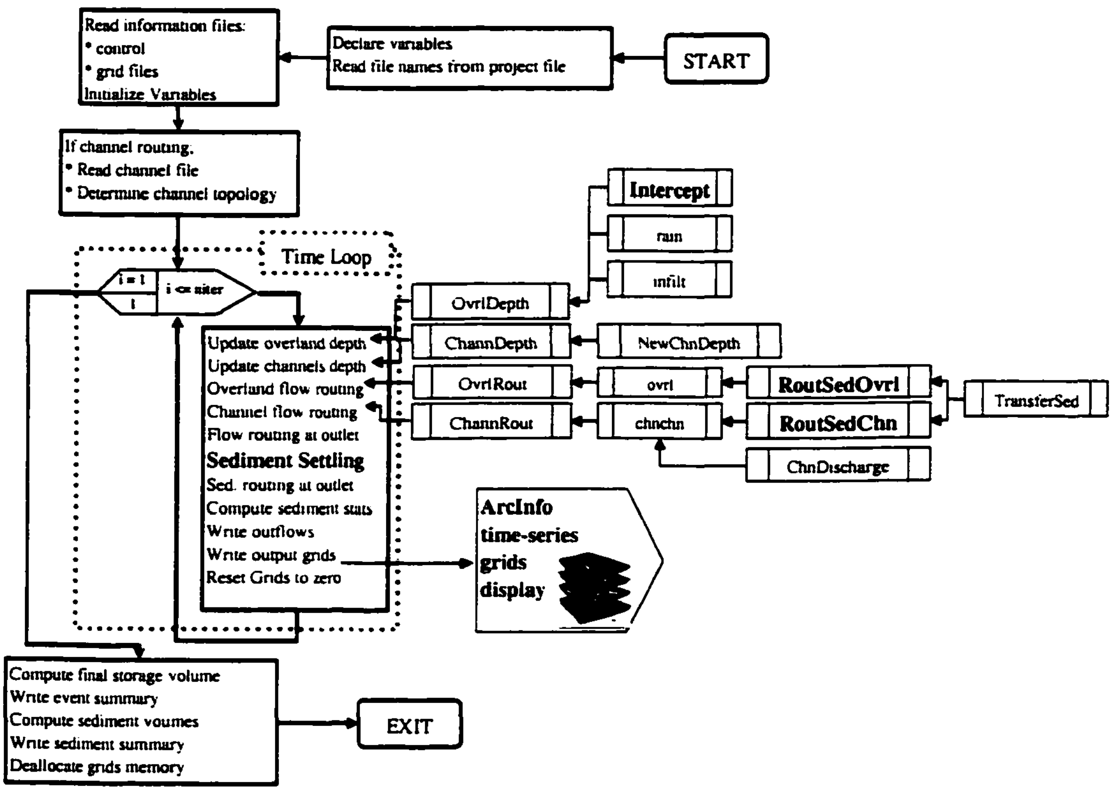


Figure 3-1. CASC2D-SED flowchart and revised/new code (in bold letters)

3.2. CASC2D-SED DESCRIPTION

CASC2D has been developed to determine the runoff hydrograph generated from any temporally-spatially varied rainfall event. When using the erosion/sedimentation module of CASC2D, sediment rates can be predicted at any location as well.

For a given rainfall event, once the initial losses have been subtracted from rainfall, water begins to infiltrate. This step requires the adoption of an infiltration scheme that can predict the portion of the rainfall that drains into the ground. The Green & Ampt (1911) infiltration equation accommodates spatial and temporal variabilities due to changes in the rainfall and/or soils properties, and takes into account the accumulated infiltration.

Using a Hortonian overland flow process (Brass, 1990), when the precipitation rate exceeds the infiltration rate, the excess rainfall will accumulate as surface water and begin to flow. In CASC2D, overland flow is routed into the channels using a diffusive wave approximation in two dimensions. In channels, the water is routed using a 1-D diffusive wave equation.

The erosion/sedimentation rates are calculated as a function of the hydraulic properties of the flow, the physical properties of the soil and the surface characteristics. The modified Kilinc-Richardson equation (Julien, 1995) is used in CASC2D-SED to determine the upland sediment transport by grain size (silt, clay, and sand) from one cell into the next one in two orthogonal directions.

CASC2D solves the equations of conservation of mass, energy and linear momentum to estimate watershed runoff for a given rainfall input. The overland flow routing formulation is based on an explicit 2-D finite difference (FD) technique. The channel formulation is based on an explicit 1-D FD technique.

In the next section, the governing equations in the overland flow and channel routing schemes are described. The equations used in estimating precipitation, interception and infiltration rates are also included. Finally, the processes of sediment transport on overland and channel cells are described.

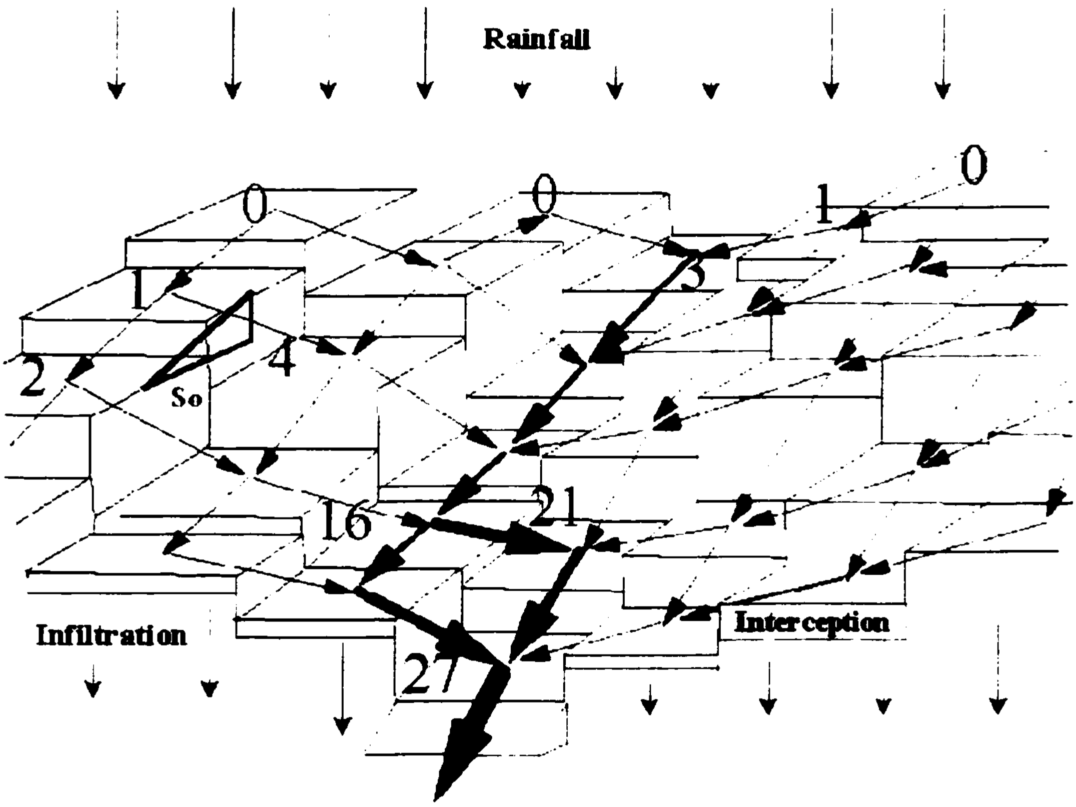


Figure 3-2. Topographical representation of overland flow and channel routing schemes

3.3. Flow Routing Algorithm

3.3.1. Precipitation

When more than one rain gauge is used in the estimation of the precipitation, an interpolation scheme based on the inverse distance squared approximates the distribution of rainfall intensity over the watershed:

$$i^t(j,k) = \frac{\sum_{m=1}^{NRG} \frac{i_m^t(jrg,krg)}{d_m^2}}{\sum_{m=1}^{NRG} \frac{1}{d_m^2}} \qquad [3\text{-}1]$$

where,

$i^t(j,k)$ = rainfall intensity in element (j,k) at time t [m s^{-1}]

$i_m^t(jrg,krg)$ = rainfall intensity recorded by the m-th rainfall gauge located at (jrg,krg) [m s^{-1}]

d_m = distance from element (j,k) to m-th rain gauge located at (jrg,krg) [m]

NRG = total number of rain gauges

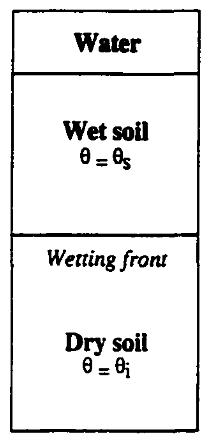
3.3.2. Interception

As rain falls on a vegetated surface, part of it is held on the foliage by surface tension forces. Rather than reaching eventually the ground, this portion of the rainfall evaporates directly and does not take part in ultimate runoff and thus, it is usually termed interception loss (Eagleson, 1970). Chow (1988) defines as retention the part of the surface storage held for a long period of time and depleted by evaporation.

Because intercepted water does not reach the soil surface, it has no part in infiltration. Accordingly, the interception loss is subtracted from the rainfall before infiltration is calculated. In CASC2D-SED, the rainfall rate is reduced until the interception depth (I) has been satisfied. For a given grid cell inside the basin, if the total rain falling during the first time increment (dt) is greater than I, the rainfall rate is reduced by I/dt. If the rainfall depth is less than I, the rainfall rate is set to zero and the remainder of the interception is removed from the rainfall in the following time increments. Measured values of interception depth for a number of vegetative covers are found in Woolhiser *et al.* (1990) and Brass (1990).

3.3.3. Infiltration

The Green and Ampt equation (1911) provides the primary relationship for infiltration within the CASC2D computer model. The Green and Ampt infiltration scheme gained considerable attention partially due to the ever growing trend of physically-based hydrological modeling (Philip, 1983). To accurately account for the physical process involved with surface flow, CASC2D uses the Green and Ampt approximation for soil infiltration. Specifically, this relationship is utilized within the model's infiltration scheme to determine the depth and rate of soil infiltration as a component of the resulting overland flow.



The Green-Ampt model assumes piston flow with a sharp wetting front between the infiltration zone and soil at the initial water content. The wet zone increases in length as infiltration progresses (Bras, 1990).

Neglecting the level of ponding on the surface, the general equation showing the Green-Ampt relationship can be expressed as (Bras, 1990):

$$f = K_s * \left(1 + \frac{H_f M_d}{F}\right) \qquad [3\text{-}2]$$

where:

f = infiltration rate [m s^{-1}]
K_s = saturated hydraulic conductivity [m s^{-1}]
H_f = capillary pressure head at the wetting front [m]
M_d = soil moisture deficit = $(\theta_e - \theta_i)$
  θ_e = effective porosity = $(\varphi - \theta_r)$
    φ = total soil porosity
    θ_r = residual saturation
  θ_i = soil initial moisture content
F = total infiltrated depth [m]

To apply this calculation to the entire watershed area, four separate physical characteristics must be known and provided as input to the model. These characteristics are hydraulic conductivity, capillary pressure, effective soil porosity, and initial soil moisture content. Their numerical values can be approximated from the experimental data by Rawls *et al.* (1983) depending on the soil texture.

3.3.4. Overland Flow

The governing equations for overland flow with the CASC2D Model are based primarily on the de-Saint Venant Equations of continuity and momentum. Using these formulations, CASC2D was designed around an explicit finite difference, diffusive-wave method to route overland flow. The general form for these equations, as shown in Julien and Saghafian (1991), are commonly expressed in partial differential form as:

Continuity:

$$\frac{\partial h}{\partial t} + \frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} = e \qquad [3\text{-}3]$$

Momentum:

x-direction

$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=g(S_{ox}-S_{fx}-\frac{\partial h}{\partial x}) \quad [3\text{-}4]$$

y-direction

$$\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}=g(S_{oy}-S_{fy}-\frac{\partial h}{\partial y}) \quad [3\text{-}5]$$

where:

h	=	surface flow depth [m]
q_x	=	unit flow rate in the x-direction [$m^2\ s^{-1}$]
q_y	=	unit flow rate in the y-direction [$m^2\ s^{-1}$]
e	=	excess rainfall rate (i-f) [$m\ s^{-1}$]
i	=	rainfall intensity [$m\ s^{-1}$]
f	=	infiltration rate [$m\ s^{-1}$]
x,y	=	Cartesian spatial coordinates [m]
t	=	time [s]
$S_{o(x,y)}$	=	bed slopes in the x- and y-direction, respectively [$m\ m^{-1}$]
$S_{f(x,y)}$	=	friction slopes in the x- and y-direction, respectively [$m\ m^{-1}$]
u,v	=	average velocities in the respective x- and y- directions [$m\ s^{-1}$]
g	=	gravitational acceleration [$m\ s^{-2}$]

Equations 3.4 and 3.5 show the relationship between the net forces per unit mass in each direction and the acceleration of flow in relation to that given direction. Thus, the forces along a given axis are shown on the right side of the equation, while the local and convective acceleration is given by the left-hand side of the equation. The simplified diffusive approximation for Equations 3.4 and 3.5 assumes that the net forces acting along the given axis of interest are approximately zero. Thus, the resulting diffusive wave approximation can be descried by the following equations.

$$S_{fx}=S_{ox}-\frac{\partial h}{\partial x} \quad [3\text{-}6]$$

$$S_{fy}=S_{oy}-\frac{\partial h}{\partial y} \quad [3\text{-}7]$$

The key advantage that is provided in using the diffusive form of the momentum equations is the ability to account for backwater effects observed during overland and channel flow events.

Using the three equations given for continuity and momentum, a resistance law can be established. This equation relates flow rate to depth and other given flow parameters such as surface roughness. The defined resistance law can be derived for either the x or y-directions as:

$$q_{x,y} = \alpha_{x,y} h^{\beta} \qquad [3\text{-}8]$$

In this form $\alpha_{x,y}$ and β are flow regime parameters that vary depending on whether turbulent or laminar conditions exist. CASC2D assumes turbulent conditions for the entire watershed and the Manning approximation for $\alpha_{x,y}$ and β are determined to be:

$$\alpha_{x,y} = \frac{S_{f(x,y)}^{1/2}}{n} \qquad [3\text{-}9]$$

$$\beta = \frac{5}{3} \qquad [3\text{-}10]$$

Where n is the Manning roughness coefficient, or surface roughness. This coefficient can be estimated from the land use map using the values provided by Woolhiser (1975).

3.3.5. Channel Flow

The channel routing scheme employed by CASC2D is capable of processing completely unsteady hydraulic scenarios. This is achieved by the use of a one-dimensional diffusive channel flow equation (Julien and Saghafian, 1991). The governing equations for the channel flow routing process are similar to those for overland flow, with one significant exception to note. The equations used in channel flow routing are defined by a finite width established for a given channel section. The one-dimensional continuity relationship can be expressed by the following equation (Julien and Saghafian, 1991):

$$\frac{\partial A}{\partial t}+\frac{\partial Q}{\partial x}=q_l \qquad [3\text{-}11]$$

where:

A = channel flow cross-sectional area [m^2]
Q = total channel discharge [$m^3\ s^{-1}$]
q_l = lateral inflow rate per unit length (into or out of the channel) [$m^2\ s^{-1}$]

Once again, by assuming the flow within the channel is completely turbulent, the model utilizes Manning's equation to ascertain a value for channel flow equation (Julien and Saghafian, 1991).

$$Q=\frac{1}{n}AR^{\frac{2}{3}}S_f^{\frac{1}{2}} \qquad [3\text{-}12]$$

where:

R = hydraulic radius [m]
S_f = friction slope [$m\ m^{-1}$]
n = Manning roughness coefficient

3.4. SEDIMENT ROUTING ALGORITHM

3.4.1. New Sediment Transport Approximation

Once a soil particle erodes, it becomes part of the flow and is transported downstream. A particle moving past a control point must have eroded somewhere in the watershed upstream of the cross section, and it must be transported by the flow from the point of detachment to the point of interest (Einstein, 1950). Each of these two requirements may control the sediment rate at the cross section: the availability of sediment in the watershed and the transport capacity of the stream. Typically, the finer material, which is easily carried in large amounts by the flow, has limited availability in the watershed. The coarse material is much more difficult to move by flows, so its rate of movement is limited by the transport capacity of the flow (Haan, 1994). Julien (1995) uses Figure 3-3 to explain the concept of supply and transport limited capacity.

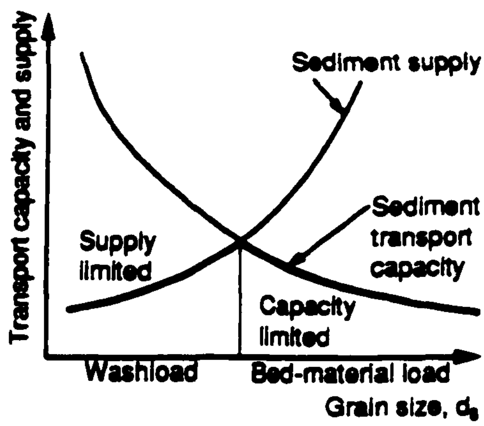


Figure 3-3. Sediment transport capacity and supply curves (Julien, 1995)

It is often assumed that the washload travels through a system by streamflow with very little deposition and is carried primarily in suspension. Coarser fractions tend to move as bed-material load. In addition, particles moving in suspension at one place may be moving as bed-material further downstream. The total sediment discharge or total sediment load, consists of both bed-material load and washload. Generally, total sediment load can only be estimated if the washload is estimated by measurements.

experiment, or upland sediment yield equation because most sediment transport methods can only determine bed-material load (Haan, 1994).

Streamwise velocity of sediment particles always lags behind the velocity of the surrounding fluid. This lag is small for the case of silt particles in suspension and large for the case of sands and gravels (Francis, 1973). Thus, it is important to predict the movement of the individual sizes found in the flow (Borah, 1982).

The overland sediment transport routine in CASC2D-SED has been extended to simulate the described transport processes. In the overland, CASC2D-SED uses the modified Kilinc and Richardson (1973) transport capacity, which depends on flow discharge, terrain slope and soil and land use characteristics. Small particles such as clay and silt move mostly in suspension while the sand fraction moves as bed-material. This is accomplished by using the particle settling concept and the transport of suspended material by advective processes. With a large settling velocity, the eroded sand fraction brought into suspension rapidly settles while the small fractions such as clay tend to remain in suspension due to its low settling velocity. Once it is in suspension, the suspended size fraction moves by advection. The advective fluxes describe the transport of sediments imparted by velocity currents (Julien, 1995). This implies that sediment will move with the fluid even for capacity limited conditions. The excess transport capacity is defined as the flow capacity to move the bed-material and produce erosion of the parent material. First, the excess transport capacity is used to move the sediment by size fraction according to its percentage in the bed-material. Then, if there is any remaining transport capacity once the suspended and bed-material have been transported, the soil is eroded proportionally to the percentage of the corresponding size fraction in the parent material. See Figure 3-4 for overland erosion and sediment transport processes flowchart.

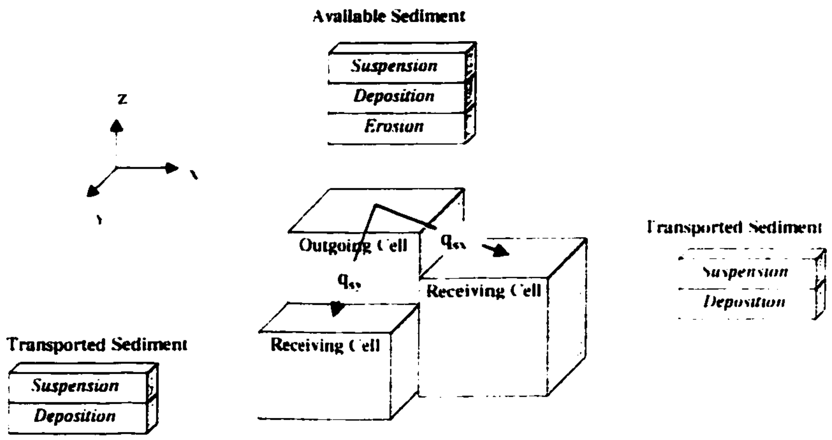


Figure 3-4. Schematic of upland sediment transport (Johnson, 1997)

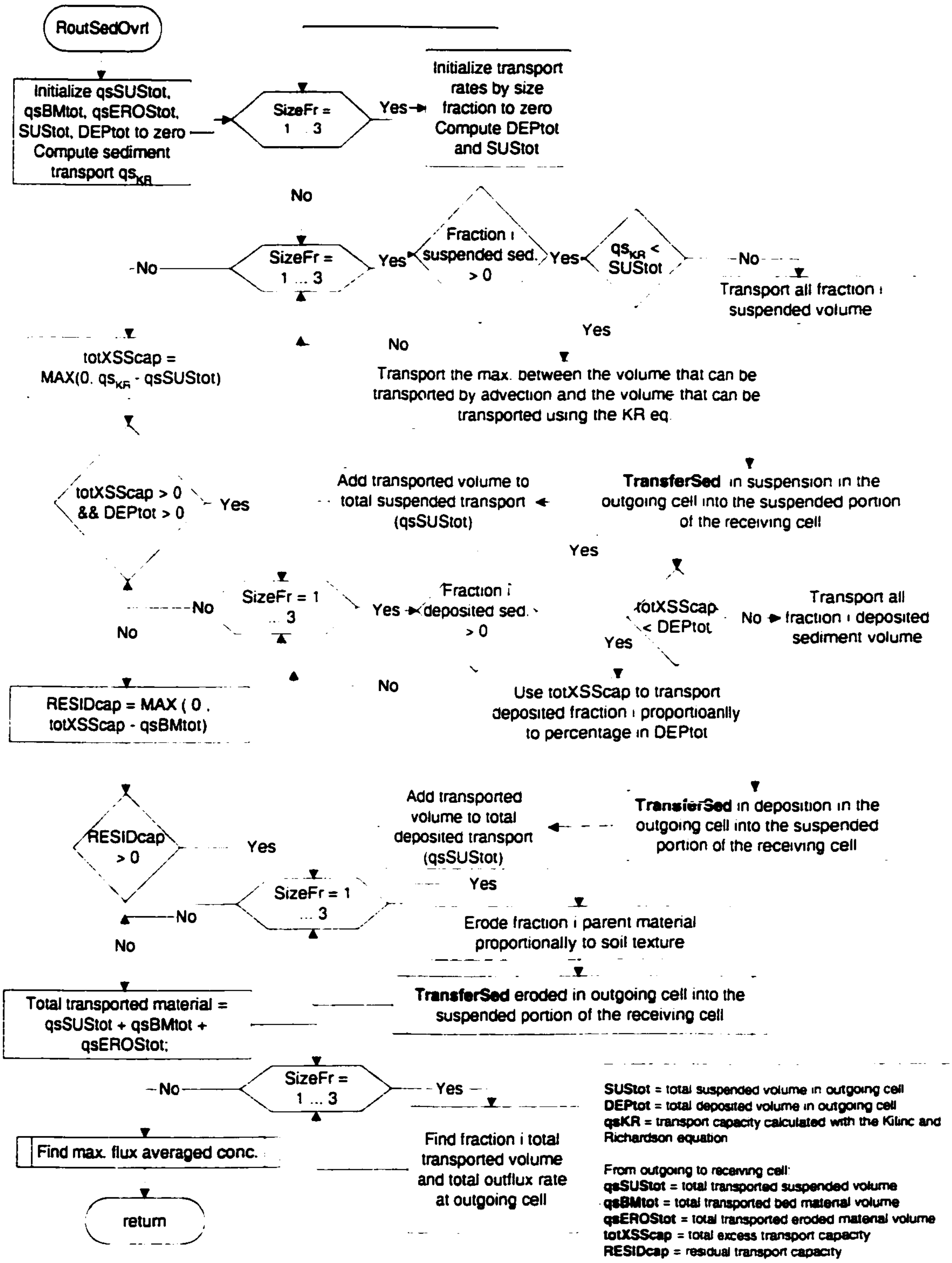


Figure 3-5. Overland erosion and sediment routing flowchart

Sediment by size fraction is routed in the channels using the Engelund and Hansen (1967) equation in 1-D. This formulation depends on hydraulic parameters (hydraulic radius, flow velocity and friction slope) and particle characteristics (specific gravity and particle diameter). For each of those fractions, a transport capacity is calculated using the Engelund and Hansen equation. For a particular size fraction, once the amount of sediment carried in suspension by advective processes has been subtracted from the transport capacity, the bed-material is transported using the excess transport capacity. The amount of transported bed-material will be the minimum between the amount that can be carried by advective processes and the excess transport capacity. Channels are not allowed to erode in CASC2D-SED and thus, the remaining transport capacity is not used. The differential treatment between the transport of the fine material as washload and the sand as bedload is accomplished through different transport capacities (as calculated with the Engelund and Hansen equation) and the differences in the settling velocities for each size fraction.

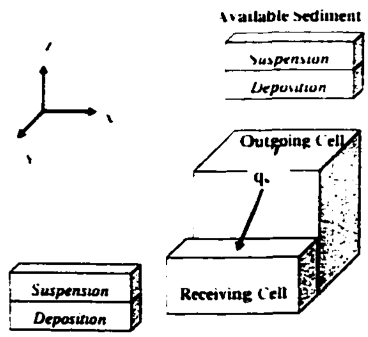


Figure 3-6. Schematic of channel sediment transport

The transported material (from suspension, bed-material or parent material) is transferred from the outgoing cell into the suspended portion of the receiving cell. Once the sediment has been routed for all the overland and channel cells in the watershed in the x- and y-direction, the sediment in suspension is allowed to settle. For a given sediment size fraction at a given cell, the volume of suspended sediment that will settle depends on the total volume of that fraction in suspension, on the particle settling velocity and the water depth at the given cell. See Figure 3-6 for channel sediment routing processes flowchart.

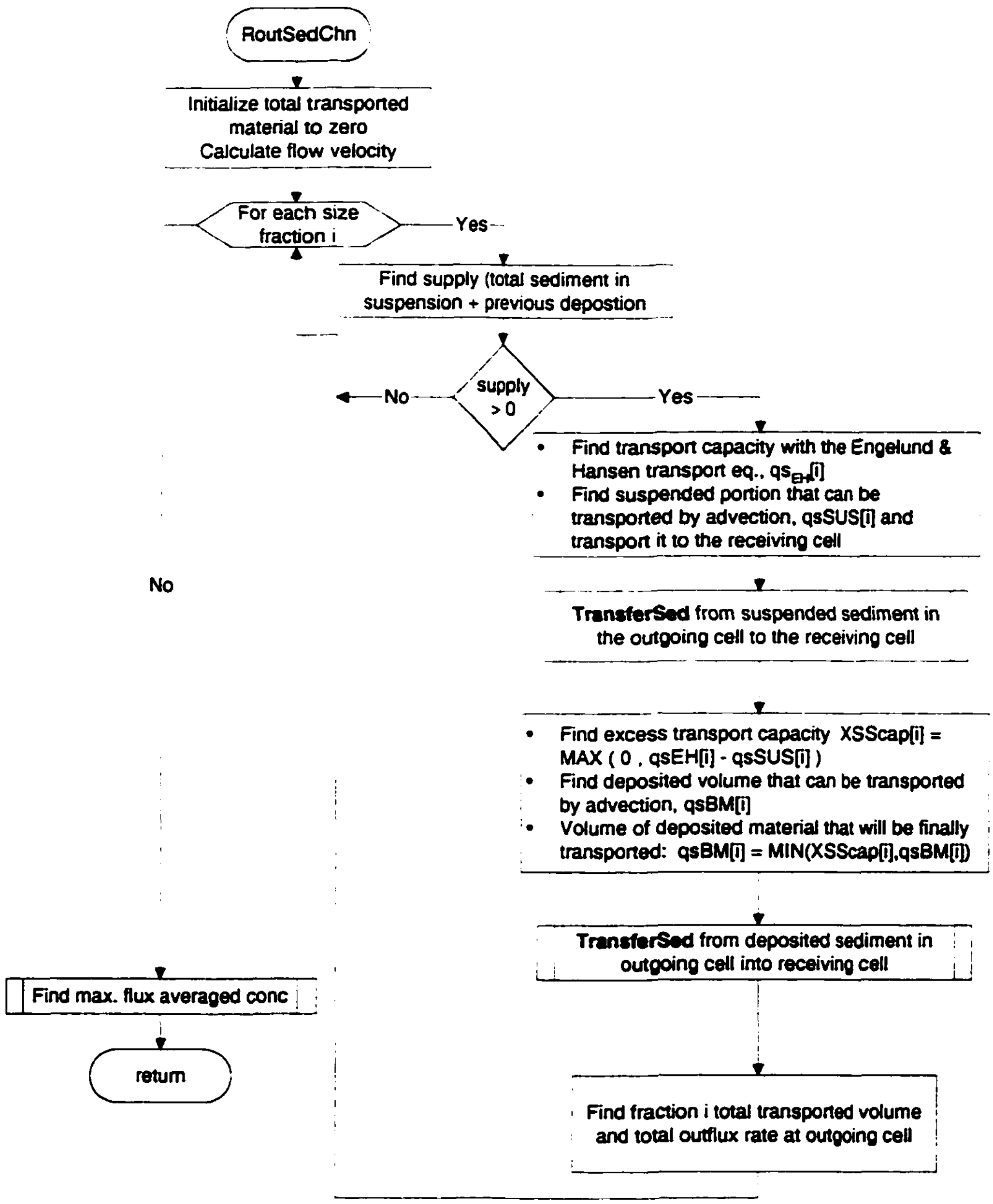


Figure 3-7. Channel sediment routing flowchart

3.4.2. Revised Upland Erosion and Sediment Transport

Julien and Simons (1984) derived a general relationship supported by dimensional analysis that can be written as a power function of slope and discharge. The unit upland sediment discharge from sheet and rill erosion can be written in the form (Julien and Simons, 1984):

$$q_s = \alpha \, S_o^{\beta} q^{\gamma} \qquad [3\text{-}13]$$

where:

So	=	surface slope [m m^{-1}]
q	=	unit discharge [m^2 s^{-1}]
q_s	=	sediment unit discharge [m^2 s^{-1}]
α, β, γ	=	coefficients

Several empirical equations were transformed to evaluate the β and γ exponents. Among these equations are those of Musgrave (1957), Li, Shen and Simons (1973), and Kilinc (1972). The values of the exponents β and γ range typically between $1.2 < \beta < 1.9$ and $1.4 < \gamma < 2.4$ (Julien and Simons, 1984).

In 1972, Kilinc studied experimentally and analytically the mechanics of soil erosion from overland flow generated by simulated rainfall. The main objectives of his research were to study the most important factors affecting soil erosion, and to develop a soil loss prediction equation. Experiments were conducted at the rainfall-runoff facilities at the Engineering Research Center of the Colorado State University. Data was collected for sediment concentration, surface velocity of overland flow, water discharge, water temperature, infiltration rate, bulk density of surface soil, slope, intensity of rainfall, and rill geometry. Some of the conclusions derived from the study of Kilinc were: a) the Reynolds number and slope were the most important parameters in sediment transport prediction equations, b) the Reynolds number, rainfall excess, and water discharge each had the same significance and influence on sediment transport from overland flow, c) sediment discharge increases with the square of water discharge, q, and 5/3 power of the slope. This model was comparable to that used by Meyer and Wischmeier (1969).

The equation of Kilinc and Richardson (1973) is recommended by Julien (1995) for the estimation of sheet and rill erosion in bare soils:

$$q_s = e^{11.727} S_o^{1.664} q^{2.035} \qquad [3\text{-}14]$$

where:

q_s = unit sediment discharge [lb ft^{-1} s^{-1}]
S_o = terrain slope [m/m]
q = unit flow discharge [ft^2 s^{-1}]

In the more general case of erosion from sheet flow, modifications to the last equation reflect the influence of soil type, vegetation, and practice factor using the USLE factors K, C, and P as (Julien, 1995):

$$q_s = 23210\, S_o^{1.66} q^{2.035} \frac{K}{0.15} C\, P \qquad [3\text{-}15]$$

where:

q_s = unit sediment discharge [tons m^{-1} s^{-1}]
q = unit flow discharge [m^2 s^{-1}]
K = dimensionless erodibility factor in the USLE equation
C = cropping management factor in the USLE equation
P = conservation practice factor in the USLE equation

The total upland erosion is calculated as:

$$Au = \int_{time} \int_{width} q_s * dx * dt \qquad [3\text{-}16]$$

For a grid of cell size, x, and for a time interval, dt, the total volume (in m^3) of sediment coming from a cell is calculated as:

$$Q_{sKR} = 58390 * S_o^{1.664} * q^{2.035} * K * C * P * x * dt \quad [3\text{-}17]$$

The rate of mass transport carried by advection, qs_{ADVi} [$m^3\ s^{-1}$] is obtained from the product of sediment concentration and the velocity component (Julien, 1995):

$$\frac{qs_{ADV_i}}{A} = V * C_i \quad [3\text{-}18]$$

where:

qs_{ADVi} = size fraction i sediment transport [$m^3\ s^{-1}$]
A = flow area [m^2]
V = average flow velocity [$m\ s^{-1}$]
C_i = suspended size fraction i concentration [$m^3\ m^{-3}$]

Simplifying and integrating for a time step dt, the volume (in m^3) of suspended sediment size fraction, i, that can be transported by advection (Qs_{SUSi}) is:

$$Qs_{SUS_i} = SusVol_i * \frac{V * dt}{x} \quad [3\text{-}19]$$

where:

$SusVol_i$ = Size fraction i suspended volume [m^3]
x = grid cell size [m]

the volume of suspended sediment that will be finally transported is found as:

$$Qs_{SUS_i} = MAX\left(Qs_{SUS_i} \ ;\ Qs_{KR} * \frac{SusVol_i}{\sum_{i=1}^{3} SusVol_i} \right) \quad if \quad Qs_{KR} < \sum_{i=1}^{3} SusVol_i \quad [3\text{-}20]$$

$$Qs_{SUS_i} = SusVol_i \quad otherwise$$

The total excess capacity (totXSScap) of the flow to carry sediments from the bed-material is calculated as:

$$totXSScap = MAX(0\ ;\ Qs_{KR} - \sum_{i=1}^{3} Qs_{SUS_i}) \qquad [3\text{-}21]$$

And the volume (in m^3) of size fraction i transported from the bed-material, Qs_{BMi}, is found as:

$$Qs_{BM_i} = totXSScap * \frac{BMvol_i}{\sum_{i=1}^{3} BMvol_i} \qquad if\ \ totXSScap < \sum_{i=1}^{3} BMvol_i$$
$$Qs_{BM_i} = BMvol_i \qquad otherwise \qquad [3\text{-}22]$$

where $BMvol_i$ is the volume of size fraction i found in the bed-material

Once the sediment in suspension and the bed-material are transported and provided that there is still remaining capacity left, the soil is eroded proportionally to the percentage of fraction i in the parent material, P_i:

$$Qs_{EROS_i} = (totXSScap - \sum_{i=1}^{3} Qs_{BM_i}) * P_i \qquad [3\text{-}23]$$

where Qs_{EROSi} is the volume (in m^3) of size fraction i eroded from the parent material.

3.4.3. Implementation of Channel Sediment Transport

The eroded material in the upland portion of the watershed is transported to the outlet through the channels. Erosion is not allowed to occur in the channels. Thus, transport might be supply limited. Deposition of material is allowed. The Engelund and Hansen (1967) equation is used to calculate the sediment transport capacity in the channels. This equation predicts transport capacity depending on the size fraction diameter and it can be used to estimate the transport capacity of the sand, silt and clay fractions coming from the upland areas.

Engelund and Hansen applied Bagnold's stream power concept and the similarity principle to obtain the sediment concentration by weight as follows (Julien, 1995):

$$Cw_i = 0.05 * \left(\frac{G}{G-1}\right) * \frac{V * S_f}{\sqrt{(G-1) * g * ds_i}} * \sqrt{\frac{R_h * S_f}{(G-1) * ds_i}}$$

where:

G	=	specific gravity of sediment [--]
V	=	channel depth-averaged velocity [m/s]
S_f	=	channel friction slope [m/m]
g	=	gravitational acceleration [m2/s]
ds_i	=	size fraction i diameter [m]
R_h	=	channel hydraulic radius [m]

and the volume of size fraction, i, that can be transported during a time interval, dt, is estimated as:

$$Qs_{EH_i} = \frac{Q * Cw_i * dt}{2.65} \qquad [3\text{-}24]$$

where Q [$m^3\ s^{-1}$] is the flow discharge in the channel and Cwi is the sediment concentration by weight of size fraction i.

The volume (in m^3) of suspended sediment size fraction, i, that is transported in the channels by advection ($Qs_{SUS}i$) is:

$$Qs_{SUS_i} = SusVol_i * \frac{V * dt}{x} \qquad [3\text{-}25]$$

The excess capacity to carry size fraction i channel bed-material, $XSScap_i$, is found as:

$$XSScap_i = MAX(0 \; ; \; Qs_{EH_i} - Qs_{SUS_i}) \qquad [3\text{-}26]$$

while the volume of fraction, i, bed-material that can be transported by advection in the channel is proportional to:

$$BMvol_i * \frac{V * dt}{x} \qquad [3\text{-}27]$$

where $BMvol_i$ is the volume of size fraction i found in the bed-material. The volume of size i bed-material that will be finally transported from the bed-material, Qs_{BMi}, is found to be the minimum between the excess capacity and the volume of the bed-material that can be carried by advection for that size fraction:

$$Qs_{BMi} = MIN(XSScap_i \; : BMvol_i \; * \frac{V * dt}{x}) \qquad [3\text{-}28]$$

In the case in which there is still remaining transport capacity, this will not be used, as channels are not currently allowed to erode in CASC2D-SED.

3.4.4. Suspended Sediment Settling

The former deposition of sediments in CASC2D-SED used the trap efficiency concept (Johnson, 1997) to simulate deposition of the sediment. The trap efficiency is defined as the percentage of sediment fraction i that settles within a given distance, depending on the particle settling velocity, flow velocity and water depth (Julien, 1995). The trap efficiency has been substituted by the sediment settling concept in the new CASC2D-SED. Two are the reasons for this change. First, for the very low water depths and velocities simulated in overland cells, the trap efficiency calculated for the sand and silt fractions was for most of the simulations equal to one. This is, all the sand and silt fractions moved as bed load, independently of the silt fraction having a much smaller settling velocity. The second reason is that for no flow conditions, for example in concave areas or sinks, the sediment remained in suspension, even after all the water had infiltrated at the end of the simulation. Using the simple concept of sediment settling once all the sediment has been routed from one cell to the next at each time step, these two inconsistencies were finally resolved. Sand, silt and clay fractions may deposit proportional to their settling velocities, thus allowing for a differential treatment of sediment routing by size fraction. The suspended sediment may deposit for no flow conditions, and once all the surface water has been infiltrated, there is no sediment in suspension left.

A particle falling in turbulent free water moves in response to the difference in the submerged weight of the particle and the drag of the fluid on the particle. At steady state, the forces in equilibrium are described by:

$$CD\left[\frac{\pi d^2}{4}\right]\left[\frac{\rho\, \omega_s^2}{2}\right] = \frac{\pi d^3}{6}(\rho_s - \rho)g \qquad [3\text{-}29]$$

where:

CD	=	drag coefficient, normally a function of the Reynolds number, Re
d	=	particle diameter
ρ_s	=	particle density
ρ	=	fluid density
ω_s	=	particle settling velocity
g	=	acceleration of gravity

Stokes showed that the drag for spheres depends on the Reynold's number:

$$CD = {}^{24}\!/_{\mathrm{Re}} \qquad [3\text{-}30]$$

where Re depends on the kinematic viscosity, ν:

$$Re = \omega_s d / \upsilon \qquad [3\text{-}31]$$

For the Stokes' range (i.e. Re < 0.5). Equation 3-30 can be simplified to yield (Haan, 1994; Julien, 1995):

$$\omega_s = \frac{1}{18}\left[\frac{d^2 g}{\upsilon}(G-1)\right] \qquad [3\text{-}32]$$

where G is the specific gravity of the particles. The settling velocity of large particles (> 0.2 mm and Re > 0.5), can be estimated with the following relationship (Julien, 1995):

$$\omega_s = \frac{8\,\upsilon}{d}\left\{\left[1 + 0.0139\, d^3 \frac{(G-1)\,g}{\upsilon^2}\right]^{0.5} - 1\right\} \qquad [3\text{-}33]$$

Assuming, medium sediment particle diameter for the sand, silt and clay fractions, and clear water at 20 °C the following fall velocities are approximated (Julien, 1995):

Table 3-1. Particle mean diameter and fall velocity

Size fraction	d [mm]	ω_s [m/s]
Sand	0.35	0.036
Silt	0.016	2.20E-04
Clay	0.001	8.60E-07

After routing the sediment from the outgoing into the receiving cell, the sediment in suspension is allowed to settle. CASC2D-SED assumes a discrete particle settling type in which particles tend to fall independently of each other as discrete particles. The percentage of suspended sediment that will deposit for an increment of time, dt, depends on the particle settling velocity, ω_i, and the cell water depth, h:

$$\text{PercSett}_i = \omega_i \frac{dt}{h} \quad \text{if} \quad h > \omega_i * dt$$
$$\text{PercSett}_i = 1 \quad \text{otherwise} \qquad [3\text{-}34]$$

The volume of suspended sediment that settles in a given time increment is subtracted from the suspended sediment portion and added to the deposited one.

3.5. CASC2D-SED Grids Time-Series Visualization

CASC2D-SED generates a series of Arc/Info® ASCII grids at specified time intervals during simulation. These grids show the interpolated excess rainfall intensity, water depth in overland and channel cells, and infiltrated rainfall volume. When the erosion option is selected, suspended sediment volume and concentration, net erosion depth, sediment flux by size fraction, and maximum flux-averaged sediment flux by size fraction can be displayed.

Currently, CASC2D-SED is loosely coupled with Arc/Info to display the outputs as a sequence of grids. Arc/Info's Arc Macro Language (AML) is used to convert the CASC2D-SED output (in ASCII format) to Arc/Info grid format and display them sequentially in a layout in ArcPlot (see Appendix I).It is possible to create layouts with 1, 2, 3, 4 and 6 thematic grids, their corresponding titles and legends and simulation time. The grids are reclassified according to the remap tables defined by the user. For the sequential display, the user may select the start and ending grid to display, as well as the time to delay the display between two sequential layouts.

MPEG is a compression standard for audio, video, and data designed to produce reasonable quality images and sound at low bit rates. The Berkeley MPEG-1 video encoder[2] is a free software that can be used in the generation of a MPEG video from image files. From the ArcPlot display, an image file is created for each layout. This image is later copied several times and finally fed into the Berkeley encoder to create a MPEG file.

3.6. Summary

In this chapter:

(1) The interception process has been implemented in CASC2D-SED adding for physical realism of the model.

(2) The upland erosion routing scheme has been revised to include sediment transport by advective processes. Suspended sediment may move as washload even in capacity limited conditions. In supply limited conditions, the suspended sediment is transported by size fraction proportionally to the amount of this size fraction in suspension. If the transport capacity is greater than the volume of suspended sediment, the previously deposited sediment is transported by size fraction, again, proportionally to the amount of the given size fraction in the deposited sediment. Subsequent excess transport capacity corresponds to erosion of the parent material with size fractions in proportion to the soil texture.

(3) The sediment routing by size fraction in channels has been added. The eroded material in the upland portion of the watershed is transported to the outlet through the channels. Currently, channels are not allowed to erode, thus, transport might be

---

[2] http://bmrc.berkeley.edu/frame/research/mpeg/

supply limited. Deposition of material is allowed. Suspended sediment moves downstream as washload by advection implying that sediment will move with the fluid even when the transport rate is lower or higher than the transport capacity. This means a supply limited transport in some cases and a capacity limited transport conditions in others. Like for the routing of the sediment in the overland portion, the transport capacity is fulfilled, first, with the suspended sediment volume and, if needed, with the previously deposited sediment. The excess transport capacity is defined as the flow capacity to move the bed-material once the sediment transported by advection is subtracted. For each size fraction, the amount of bed-material that will be transported is limited by the amount that can be transport by the flow (advection) or by the excess transport capacity. In the case in which there is still remaining transport capacity, this will not be used, as the channels are not currently allowed to erode in CASC2D-SED. The fine sediment fraction has larger transport capacity than the coarser fraction and a smaller settling velocity making the finer fractions to move downstream as washload more rapidly than the sand fraction.

(4) The former trap efficiency has been traded with the sediment settling concept. This allows deposition of the sediment for no flow conditions. It also allows for a differential treatment in the silt and sand fractions routing in the upland portion of the basin. The low settling velocities corresponding to the finer sizes (silt and clay) make them to stay in suspension and so they move as washload with the flow.

(5) The program time-series output grids may be displayed automatically in the ArcPlot module of Arc/Info® with the aid of the AML programming language. An MPEG file may be generated with the use of the freeware Berkeley encoder.

CHAPTER 4

STUDY SITE DATA SET

"We must keep in mind that all models are simplifications or abstractions of reality and all models are to some extent wrong. In fact, if they aren't simpler in some sense than the real-world object, they aren't useful! For this reason we neglect certain aspects of the problem because they are considered to be unimportant. These simplifications should be based on sound physical reasoning or strong empirical evidence obtained from field studies or appropriate material models [...]. Because of difficulties and the cost of measurements we simply cannot provide a detailed, three-dimensional description of the surface microtopography, the hydraulic characteristics of the soils, and the underlying geologic materials."

David A. Woolhiser, 1996

The extensively monitored Goodwin Creek watershed is selected for the CASC2D-SED model application. This watershed has a large database[3] compiling precipitation, runoff, sediment, and GIS data. In the first section, this watershed is described.

Goodwin Creek digital elevation model (DEM), land use and soil maps are available at 30-m resolution (see Figure 5-1). These raster maps at 30-m resolution are gridded at different scales to obtain coarser spatial resolution grids at 90-, 150-, 210-, 270- and 330-m. The DEM is resampled using the bilinear interpolation approximation provided in the Arc/Info GRID module. The nearest neighbor has been the method of choice for resampling the soils and land use maps. The effects of the grid cell size on Goodwin Creek watershed representation are analyzed. DEM, slopes, soils, and land use resampled maps are found in Appendix II.

4.1. Goodwin Creek Experimental Watershed Description

Goodwin Creek is located in Panola County, Mississippi (see Figure 4-1). It is a tributary of Long Creek, which flows into the Yocona River, one of the main rivers of the Yazoo River Basin. The watershed is operated by the National Sedimentation Laboratory (NSL), and it is organized and instrumented for conducting extensive research on upstream erosion, instream sediment transport, and watershed hydrology (Shields *et al.*, 1995; Alonso, 1995; Alonso *et al.*, 1996; Kunhle *et al.*, 1996; Kunhle and Willis, 1998).

---

[3] http://www.sedlab.olemiss.edu/cwp_unit/Goodwin.html

The watershed has a database compiling runoff, sediment, and precipitation from 1981 until 1996. This database is available at the NSL web site

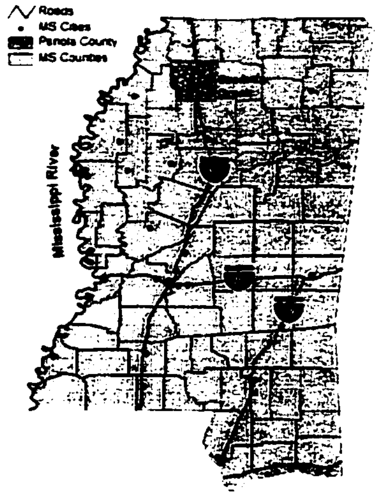


Figure 4-1. Goodwin Creek location

The watershed flows approximately from northeast to southwest, draining a total area of 8.26 square miles (21.4 km^2), with the outlet at latitude 89 54' 50" and longitude 34 13' 55". The terrain elevation ranges from 233 feet to 420 feet (71 m. to 128 m.) above mean sea level, with an average channel slope of 0.004 in Goodwin Creek. The Digital Elevation Model (DEM) of Goodwin Creek is found in Figure 4-2.

In Goodwin Creek, two major soil associations are mapped. The Collins-Falaya-Grenada-Calloway association is mapped in the terrace and flood plain locations. These are silty soils, poorly to moderately well drained and include much of the cultivated area in the watershed. The Loring-Grenada-Memphis association has developed on the loess ridges and hillsides. These are well to moderately well drained soils on gently sloping to very steep surfaces and include most of the pasture and wooded area in the watershed. Table 4-1contains the soil characteristics table for Goodwin Creek (Blackmarr, 1995) and the soils map is found in Figure 4-2.

Table 4-1. Goodwin Creek soil descriptions (Blackmarr, 1995)

Soil Series	Description
Calloway (Ca)	Fine-silty, mixed, thermic Glossaquic Fragiudalfs; soils are somewhat poorly drained, strongly acid or medium acid silt loam soils formed in deposits of loess in upland positions of low relief (terraces). A fragipan is present generally at a depth of 16 inches.
Collins (Cm)	Coarse-silty, mixed, acid, thermic Aquic Udifluvents; soils are moderately well drained, strongly to medium acid, that have formed in silty alluvium on nearly level bottom lands. These silt loam soils occur primarily along the stream in the bottom area and are the location of much of the cultivation in the watershed. Cotton is the predominant crop but has been supplanted somewhat in recent years by soybeans.
Falaya (Fa)	Coarse-silty, mixed, acid, thermic Aeric Fluvaquents; soil consists of somewhat poorly drained, strongly to very strongly acid silt loam soils that developed in silty alluvium on nearly level bottom land. Most of the Falaya is cultivated.
Grenada (Gr)	Fine-silty, mixed, thermic Glossic Fragiudalfs; soil consists of moderately well drained, strongly to very strongly acid silt loam soils that have developed in thick loess deposits on uplands or terraces. A fragipan is present at a depth of about 24 inches.
Gullied Land (Gu)	Land consists of areas that are severely eroded, severely gullied, or both. The surface soil and much of the subsurface soil has been washed away. Most of this is land that was cleared, cultivated and later abandoned. It is now in trees, idle or pastured. It is unsuited for cultivation.
Loring (Lo)	Fine-silty, mixed, thermic Typic Fragiudalfs, soil series is moderately well drained to well drained, strongly to very strongly acid silt loam soils that developed in thick loess on uplands. A fragipan has formed at a depth of about 30 inches.
Memphis (Ml)	Fine-silty, mixed, thermic Typic Hapludalfs, soil series consists of well drained, strongly to very strongly acid silt loam soils that developed in thick loess on uplands. In Goodwin Creek, this soil occurs as a mixture with the Natchez and Guin or the Loring. This series has no fragipan within the characterization depth; it is predominantly wooded.
Mixed Alluvial Land (Mx)	Land is poorly drained to excessively drained, strongly acid silt loam and coarse sand; no uniformity in the arrangement, depth, color, or thickness of the soil layers. The soil is doughty and very low in organic-matter content and in natural fertility. It is in cultivation (row crops), pasture and trees (hardwoods).

The land use and management practices influence the rate and amount of sediment delivered to streams from the upland. They range from timbered areas to row crops. The Goodwin Creek watershed is largely free of land management activities with 13 percent of its total area being under cultivation and the rest in idle, pasture and forestland. Periodic acquisition of aerial photography and satellite data contributes to complete aerial coverage of land use and surface conditions. Land Use / Land Cover (LULC) in Goodwin Creek are classified as shown in Table 4-2 (Blackmarr, 1995).

Table 4-2. Land use/cover in Goodwin Creek watershed

Land Use / Land Cover	Description
Cultivated Land	Divided into three categories: cotton, soybeans and small grain. The field classification is based upon visual confirmation of the crop or by asking the land owner. Types of crops are cotton, soybeans, corn, and small grain
Pasture	Classified on the up-keep of the land, the presence of cattle, the presence of fences, and/or asking the land owner.
Idle Land	Classified on the up-keep of the land, if overgrown with scrub vegetation, the absence of cattle, no fences present, and/or asking the land owner
Forest	Classified on the age of the trees, an approximation of age is based on tree height and width which is usually seven years and older
Planted Forest	Classified on the age of the trees; as with forest, an approximation of age is based on tree height and width. The range for the classification is from newly planted to seven years old

In this study, the LULC has been further reclassified as forest (includes planted forest), pasture (includes idle land), water and cultivated. The land use map is presented in Figure 4-2.

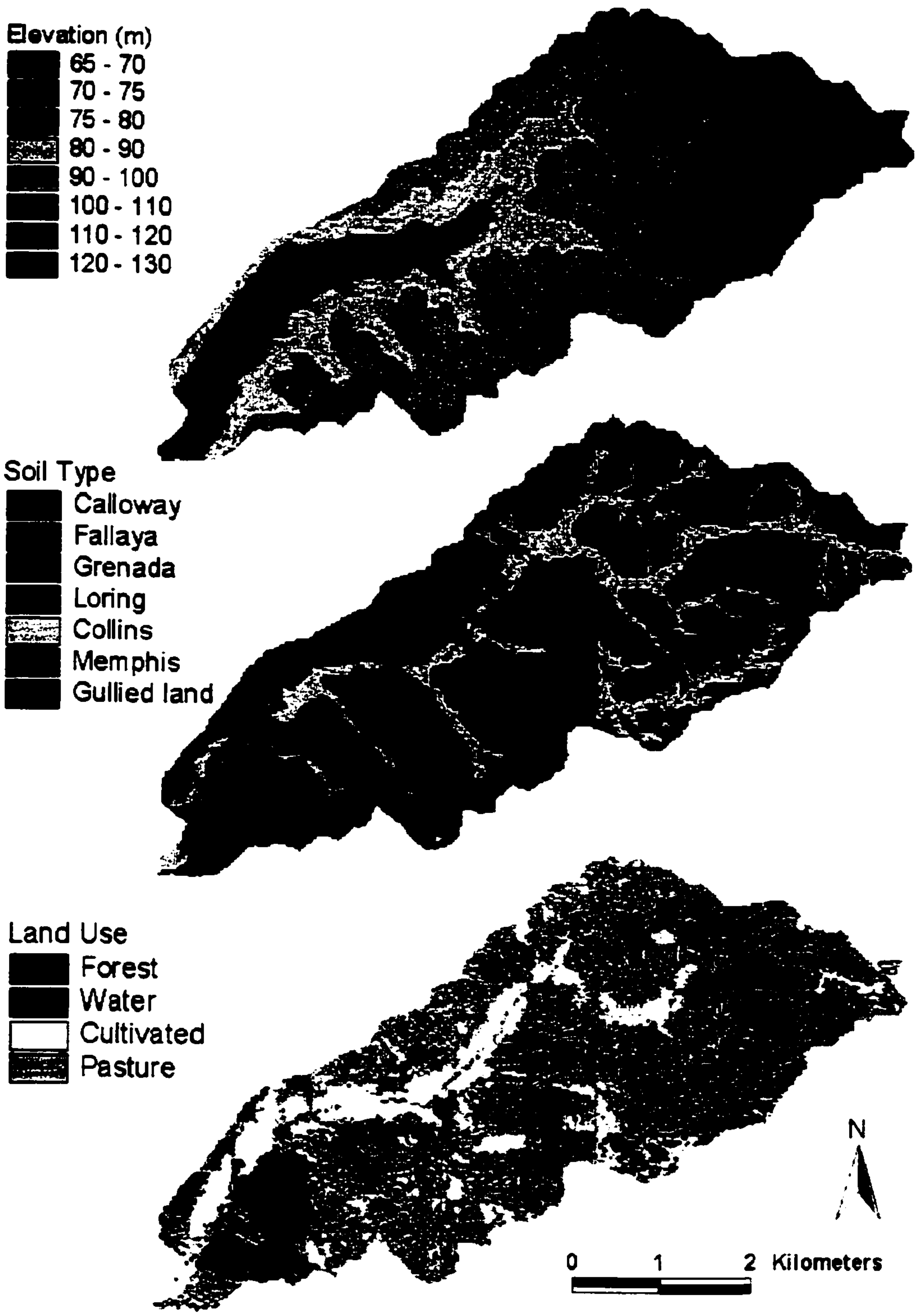


Figure 4-2. DEM, soils and land use maps of the Goodwin Creek watershed (30-m)

The climate of the Goodwin Creek watershed is humid, hot in summer and mild in winter. The average annual rainfall during 1982-1992 from all storms was 56.7 inches (1440 mm), and the mean annual runoff measured at the watershed outlet was 5.7 inches (145 mm) per year. Thirty two standard recording rain gauges are uniformly located within and just outside the watershed. Figure 4-3 shows the rain gauges location as well as the rain gauge data used in the present study.

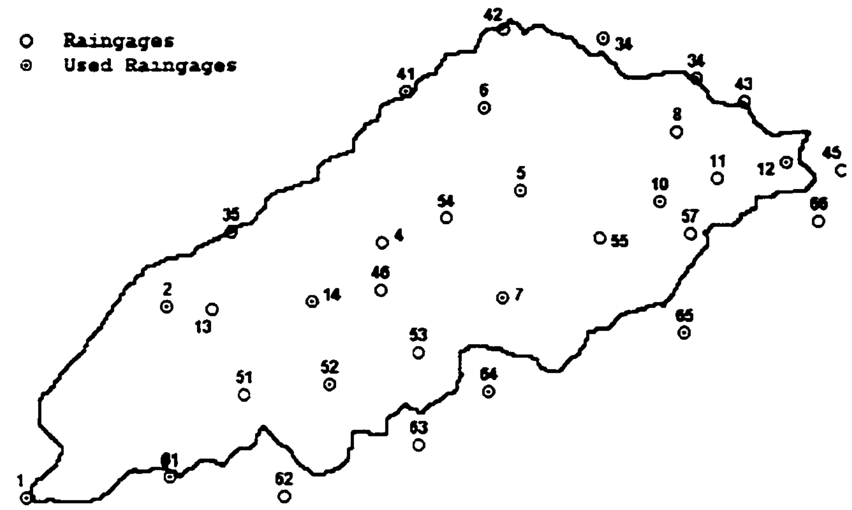


Figure 4-3. Goodwin Creek Watershed rain gauge locations

The Goodwin Creek Watershed is divided into fourteen nested sub-catchments with a flow-measuring flume constructed at each of the drainage outlets. The drainage areas above these stream-gaging sites range from 0.63 to 8.26 square miles. Figure 4-4 shows the streamflow measuring stations and corresponding sub-catchments that have been used in this study. The station coordinates are listed in Table 4-3.

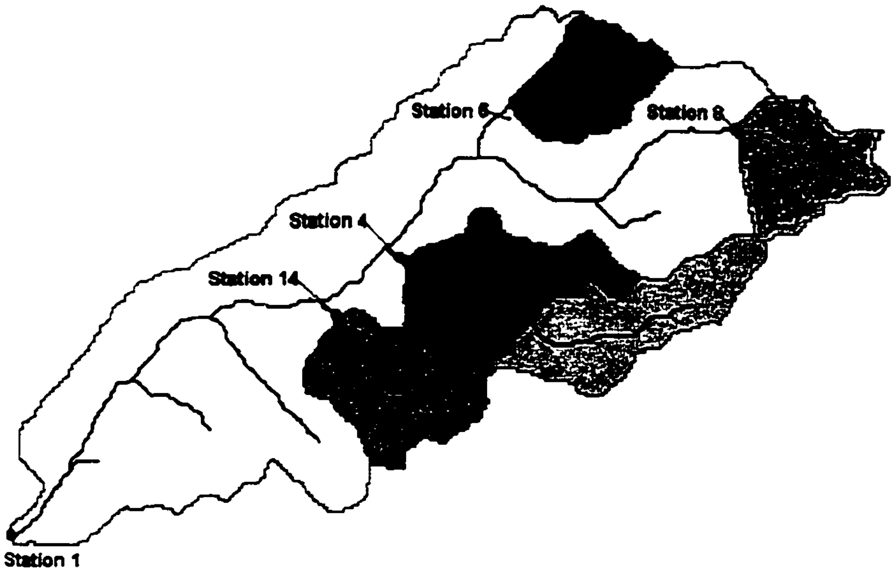


Figure 4-4. Goodwin Creek watershed and studied sub-basins

Table 4-3. Selected stream gauges location

	Geographical		UTM	
Station	Latitude	Longitude	X	Y
1	34 13 56.063	89 54 51.000	231568.712	3791553.743
4	34 15 27.986	89 52 25.712	235367.347	3794280.517
6	34 16 16.082	89 51 44.665	236459.330	3795732.931
7	34 15 10.342	89 51 34.479	236662.950	3793699.882
8	34 16 09.930	89 50 21.643	238577.958	3795483.805
14	34 15 07.040	89 52 53.252	234644.397	3793655.021

Channel cross section data has been compiled by the ARS-NSL from 1978 until 1988. Average depth and width of each of the surveyed channel links are presented in

Table 4-4 (Backmarr, 1995). The link numbers in correspond to the ones shown in Figure 4-5.

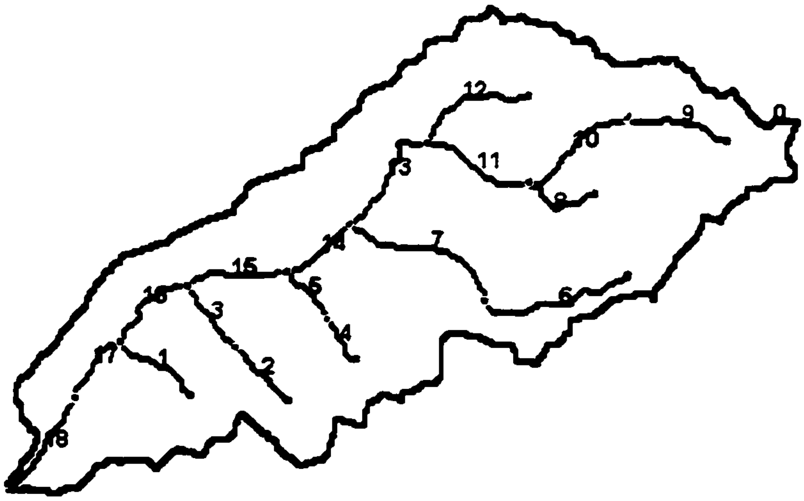


Figure 4-5. Channel network and link numbers (30-m)

Table 4-4. Averaged channel depths, and widths

Channel Link	Width [m]	Depth [m]
1	25	3.5
2	20	3
3	22	3.5
4	27	4.3
5	28	3.1
6	30	3.4
7	30	3.55
8	22	4.1
9	29.4	4.2
10	26	4.35
11	30	4.4
12	22	4.4
13	27	4.3
14	30	4.5
15	30	4.7
16	50	5
17	48	5
18	34	6.05

High erosion rates have been observed to occur, with rill formation in the upland areas that turn into gullies along channel banks (Johnson, 1997; Johnson, 2000). Dr. Johnson observed bank and bed erosion along the main stem and tributary channels as well as deposition in the milder sloped fields.

4.2. DEM Pre-processing and Resampling

Digital elevation models (DEMs) and their derivatives such as slope, flow direction and flow accumulation maps are used as an input to hydrologic and nonpoint source modeling. The depressions which are frequently present in DEMs may represent the actual topography, but are often the result of errors. Creating a depression-free surface is commonly required prior to deriving flow direction, flow accumulation, flow network, and watershed boundary maps. The 30-m DEM as provided by the ARS-NSL is edited with this purpose using the TOPAZ program (Garbrecht and Martz, 1997a, 1997b). TOPAZ was chosen over ArcInfo because a) it performs better in minimizing the area modified in the process of creating a depressionless surface (Srivastava, 2000), and b) it always produces a connected channel network. TOPAZ outperforms ArcInfo particularly in flatter topography zones of the watershed.

The purpose of **DEM pre-processing** is to create a depressionless surface. With this purpose, first, the elevation data is smoothed using an equal weight, single pass 3x3 box filter. Then, the sinks or pits in the DEM are resolved by filling them. Normally, filling of the sinks involve less than 2% of the cells and, on the average, these minimal adjustments are smaller than the root-mean square error of the data determined by the U.S. Geological Survey (Tarboton *et al.*, 1989). Flat areas (including the ones as a result of depression fillings) are modified to produce the most likely drainage paths and to produce a fully connected drainage network. TOPAZ relief modification uses incremental elevations of 2/100,000 of a DEM elevation unit (1 meter) to build gradients over flat surfaces that drains flow away from rising terrain and towards the nearest outlet flat surface (Garbrecht and Martz, 1997c, 1999; Martz and Garbrecht, 1998).

Resampling in Arc/Info® GRID is the process of determining new values for cells in an output grid that result from applying some geometric transformation to an input grid (ESRI, 1994). The three resampling techniques used in GRID are nearest neighbor assignment, bilinear interpolation, and cubic convolution. The bilinear interpolation identifies the four nearest input cell centers to the location of the center of an output cell on the input grid and assigns a new value for the output cell as a weighted average. This average is determined by the value of the four nearest input cell centers and their relative position or weighted distance from the location of the center of the output cell in the input grid. The bilinear interpolation is used when aggregating continuous data type and has been used in this study to resample the DEM.

The entire DEM has been resampled with the nearest neighbor and bilinear interpolation to investigate the differences between these two methods when resampling continuous data type. Figure 4-7 shows the differences in computed slope angles using these techniques when an odd and even number of cells are aggregated (see Figure 4-6). There are no differences in slope values distribution between both methods for the odd resampling case and a slight one in the case of even resampling.

In order to preserve as much as possible the terrain slope, the DEM is resampled using an odd number of cells with the bilinear interpolation method. This is equivalent to resample the watershed using the nearest neighbor technique.

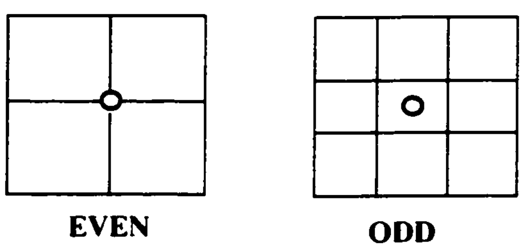


Figure 4-6. Resample of an even and odd number of cells

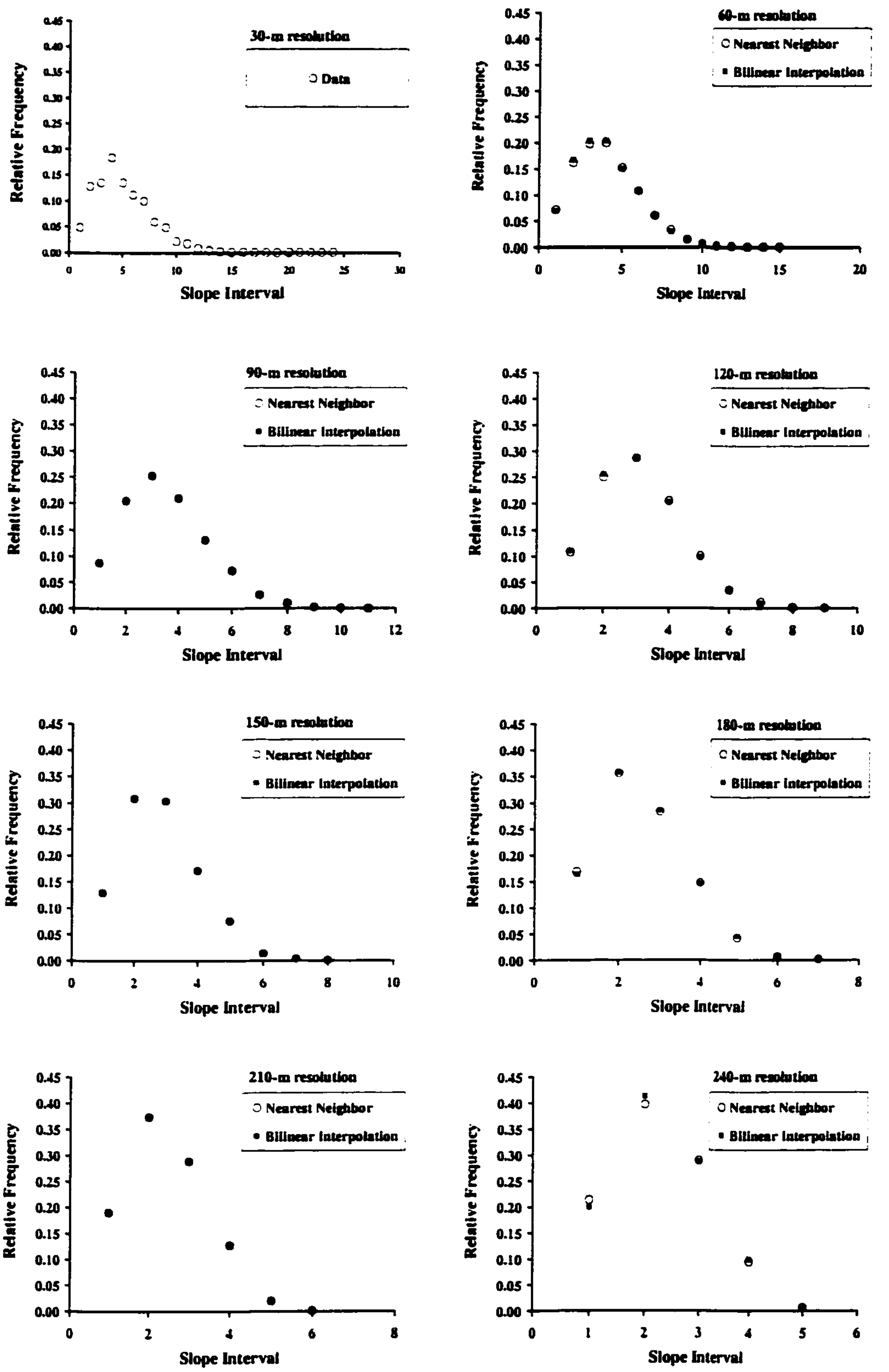


Figure 4-7. Relative frequency of slope intervals for the different resolutions

Using the 30-m spatial resolution grid, the Goodwin Creek basin was resampled at 5 different spatial resolutions: 90-, 150-, 210-, 270- and 330-m.

4.2.1. Elevations and Slopes

Aggregation of the cells from 30- to 330-m causes smoothing of the basin's relief (see Table 4-5). While the mean and minimum elevations are mostly maintained for any of the resolutions, there is a reduction in the elevation range of about 8 m. from the smallest resolution (30-m) to the coarsest resolution (330-m).

Table 4-5. DEM statistics for 30-, 90-, 150-, 210-, 270-, and 330-m resolution

Grid Size [m]	Minimum Z [m]	Maximum Z [m]	Range (ΔZ) [m]	Mean Z [m]	Std.Dev [m]
30	69	126	57	95.58	12.42
90	69	124	55	95.51	12.33
150	69	123	54	95.30	12.13
210	69	123	54	95.00	11.86
270	70	121	51	95.19	11.67
330	71	120	49	94.93	11.26

ΔZ = Maximum. Z - Minimun Z

Resampling of the elevation map from 30 to 330-m resolution causes a change in the slope distributions as well (see Figure 4-8). The steepest slopes occur for the finest resolution and they vanish for coarser resolutions. As the cell size increases, the slope average and standard deviation decrease (see Figure 4-9). The average of the slope changes with grid cell size (x) according to the next equation: $\bar{S}_x = 17.9x^{0.407}$

The minimum slope value tends to increase slightly with increasing cell size while the maximum value decreases more significantly for coarser resolutions. Spatial distributions of slope values computed from Goodwin Creek DEM at different resolutions are shown in Figure 4-10.

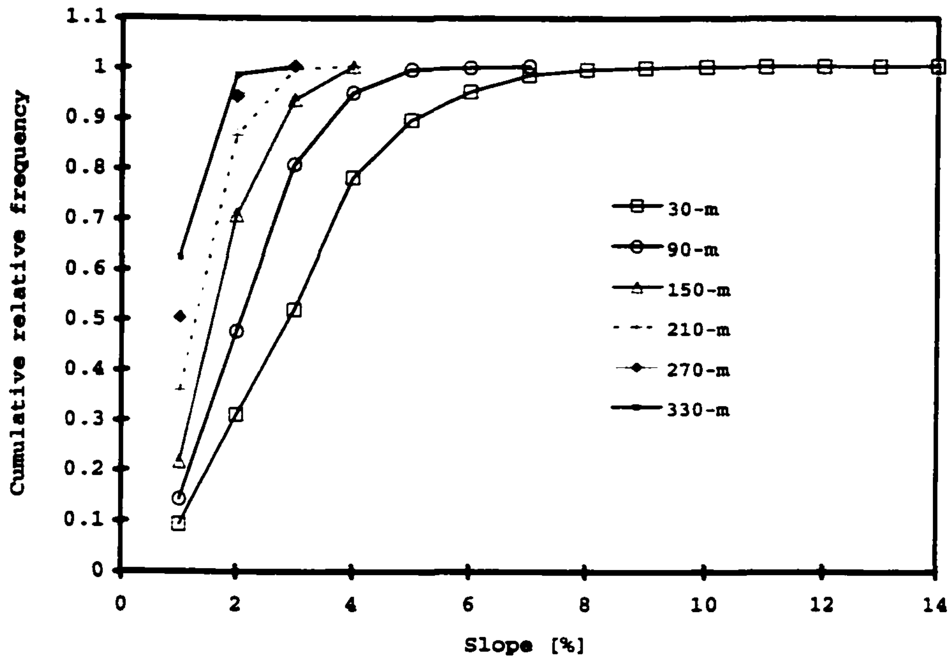


Figure 4-8. Slope cumulative distribution and frequencies for 30-, 90-, 150-, 210-, 270-, and 330-m resolution

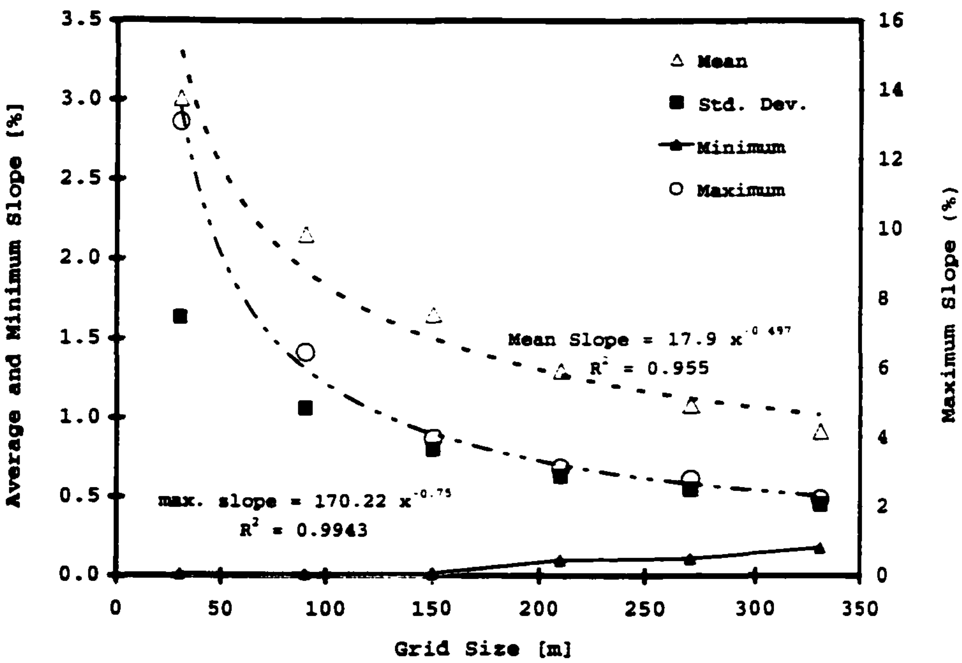


Figure 4-9. Statistical mean and standard deviation and maximum slope angles for 30-, 90-, 150-, 210-, 270-, and 330-m resolution

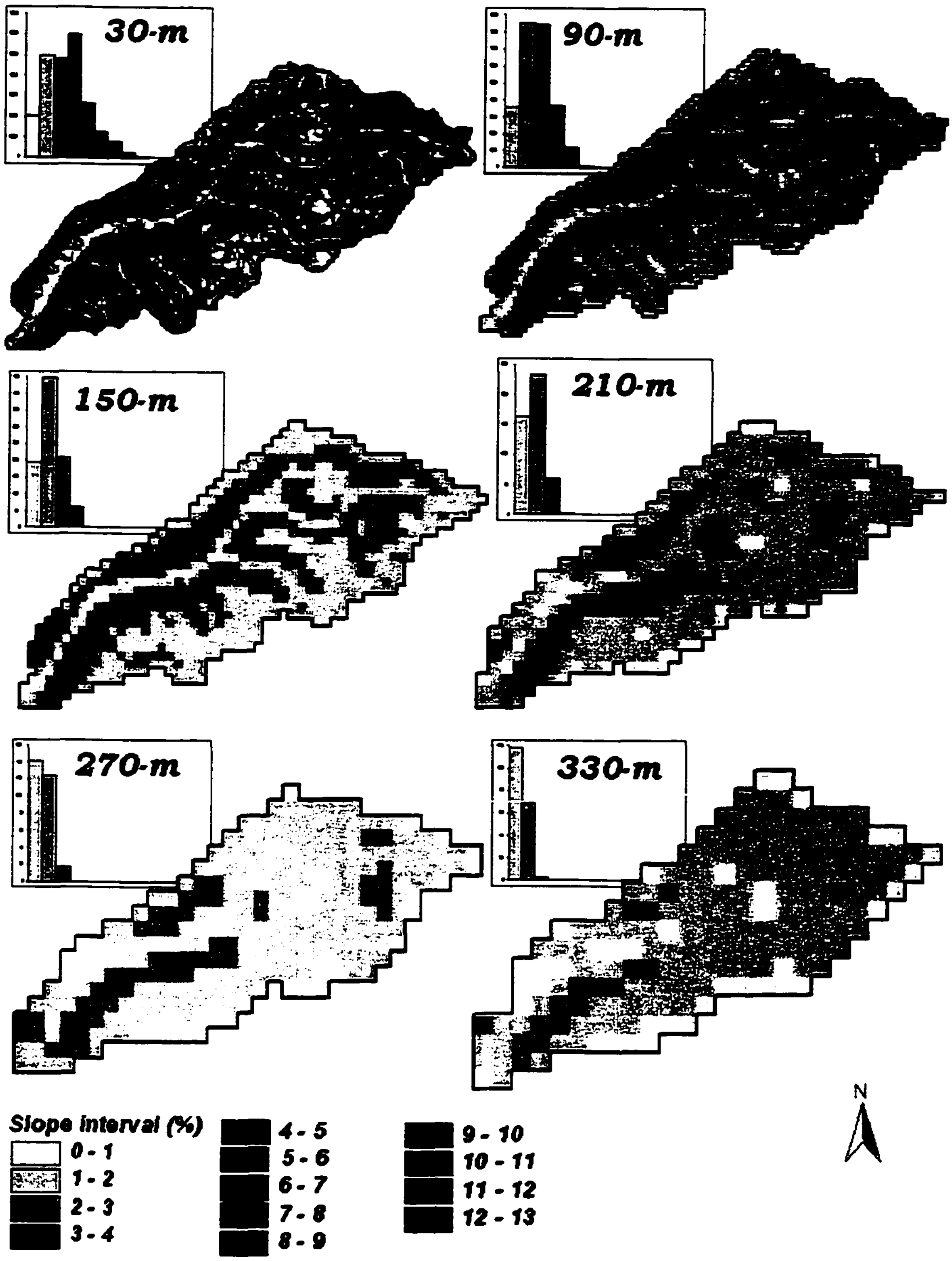


Figure 4-10. Goodwin Creek slope angle values and distribution

4.2.2. Channel Network and Watershed Delimitation

The channel network and watershed are delimited from the resampled, smoothed DEM at 30-m spatial resolution by first determining the flow direction in each cell. TOPAZ uses the D-8 method to establish the drainage direction in one of eight principal directions of the raster grid cell (Garbretch and Martz, 1997b). Using the flow direction grid, the flow accumulation grid is obtained by counting the number of pixels that drain through each cell in the DEM. The largest accumulation of drainage area are located in the valleys along the streams. The streams are defined as those pixels with a total drainage area greater than a support area threshold. With the DEM, channel network and the basin and sub-basins outlet location (see Figure 4-4), the watershed and sub-watersheds are delimited by identifying those pixels that eventually drain through the corresponding outlet pixel. For all the spatial resolutions, the basins' outlets have been located on a stream line, as close as possible to the position described by the NSL (see Table 4-3). The basin and sub-basins configuration (maps) are found in Appendix II for 30-, 90-, 150-, 210-, 270-, and 330-m spatial resolution.

Because sub-basins areas differ between resolutions, results are going to be compared by unit area (i.e. hydrographs in mm/h and sedigraphs in tons/ha/day).

Table 4-6. Computed and documented (Blackmarr, 1995) basin and sub-basins areas

Resolution [m]	Station 14 [has]	Station 4 [has]	Sub-basin 7 [has]	Station 8 [has]	Station 6 [has]	Station 1 [has]
30	165.02	357.82	162.98	137.56	111.49	2065.59
90	151.46	365.88	166.07	135.85	145.86	2071.98
150	138.53	345.81	185.36	140.42	141.07	2052.00
210	145.53	332.08	157.05	129.27	119.71	2028.60
270	132.36	374.10	272.78	137.75	152.52	2055.78
330	108.90	402.93	141.57	141.57	196.02	2069.10
Documented	*162.6*	*356.5*	*162.5*	*155.5*	*120.5*	*2145.25*

In this study, it was assumed that channels were initiated by a constant contributing threshold area (CSA). The channel network in Goodwin Creek has been derived from the 30-m DEM for a minimum flow accumulation of 450 cell (40.5 has). The resulting network is very similar to the blueline drainage network shown on the USGS 7.5-min topographic quadrangles of the study watershed (see Figure 4-5). The channel network has been defined for other resolutions for a minimum drainage area of approximately 40.5 ha. The minimum number of cells and corresponding minimum drainage area are listed in Table 4-7 for all spatial resolutions. The corresponding channel networks are shown Appendix II.

Table 4-7. Channel minimum contributing threshold area, stream length and drainage density for each of the grid cell sizes

Cell Size [m]	**Minimum cell number**[1]	**CSA** [ha]	**Stream Length** [km]	**Drainage Density** [km/km^2]
30	450	40.5	19.800	0.959
90	50	40.5	18.678	0.902
150	18	40.5	16.454	0.802
210	9	39.69	15.955	0.786
270	5	36.45	15.462	0.752
330	4	43.56	14.095	0.681

[1] Minimum flow accumulation for channel definition.

The contributing threshold area controls the extent of the watershed configuration and therefore determines the drainage density. In Goodwin Creek, the drainage density decreases from 0.959 to 0.681 [km/km^2] as grid cell size increases (see Table 4-7). This grid size dependency is due to vanishing of the shorter channel links and the inability of the resampled DEM to reproduce drainage features such as channel sinuosity as the grid cell size increases.

Resampling of Goodwin Creek watershed to 330-m causes not only shortening of the channel links but it results in channel and drainage area capturing. This is, the DEM grid can no longer resolve the separation between channels or drainage boundaries (Garbrecht and Martz, 1994). The number of channels and network pattern departs considerably from the initial reference values.

4.3. Soil Type and Land Use Grids Resampling

The Nearest Neighbor is the resampling technique of choice for categorical data since it does not alter the value of the input cells (ESRI, 1994). Both soil type and land use are categorical data and the nearest neighbor has been the applied technique for resampling the 30-m resolution grids to 90-, 150-, 210-, 270-, and 330-m.

Figure 4-11 and Figure 4-12 show the extent of each soil type and land use for each of the resolutions. In general, these extents remain practically the same for the different resolutions at the basin scale. At different sub-basins within the watershed, some of the values might be much different from the rest (See Appendix II)

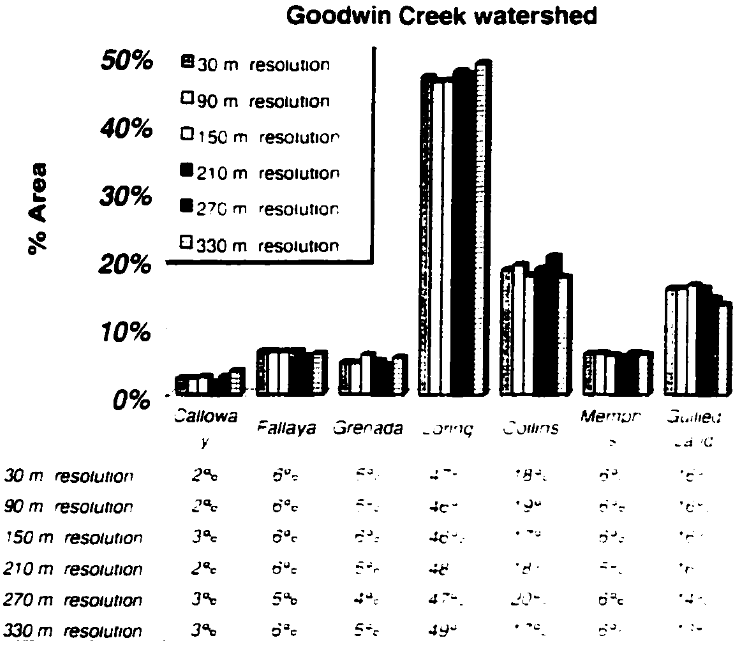


Figure 4-11. Percentage of the area covered by each soil type at Goodwin Creek

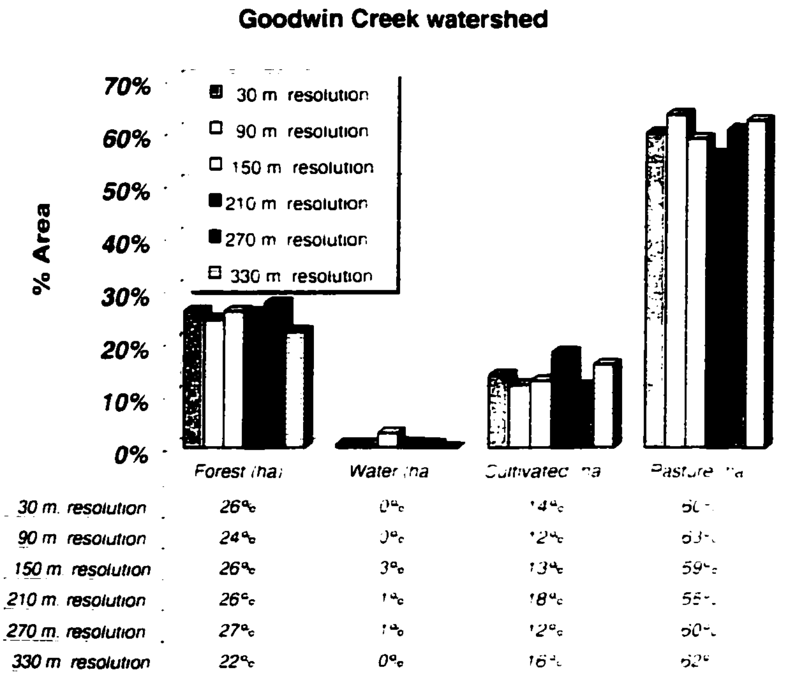


Figure 4-12. Percentage of the area covered by each land use type at Goodwin Creek

4.4. SUMMARY

The Goodwin Creek watershed has been selected as study site due to its extensive database compiling precipitation, runoff, sediment, and GIS data. The DEM, soil type and land use grids have been resampled from 30-m to 90-, 150-, 210-, 270-, and 330-m resolution. As expected, it has been observed that increasing the grid size:

(1) has affected the watershed representation by affecting primarily the computed slope distribution. Mean slope, slope standard deviation, and slope range decrease with increasing grid cell size.

(2) has affected the channel network definition as well. Drainage density decreases as grid cell size increases due to vanishing of the shorter channel links and the inability of the resampled DEM to reproduce drainage features such as channel sinuosity. Resampling of the basin's DEM to 330-m results in channel and drainage area capturing.

(3) did not change significantly aerial extents of the different soil types and land use at the basin scale. This difference is more significant for the smaller sub-basins.

CHAPTER 5

MODEL CALIBRATION AND VALIDATION

"In a thorough scientific test of physically based runoff models, it has appeared desirable to include events with a range of initial conditions, time pattern of rainfall, and total rainfall (and runoff) amounts. This ensures that the ability of the model to handle small- and large-scale spatial variability will be tested. However, it is important to appreciate the very sensitive nature of surface runoff modeling for small events (where the runoff is a small proportion of the rainfall) so that we don't have unrealistic expectations"

David A. Woolhiser, 1996

The CASC2D-SED model is used to predict discharge and erosion rates at Goodwin Creek. The basin is defined at 90-m spatial resolution. The model is manually calibrated for one event and later validated for two other events. Calibration and validation hydrographs, sedigraphs and sediment yield are shown at the basin outlet and at other internal locations. Spatially distributed net erosion is shown for the calibration event.

5.1. CASC2D-SED Set Up

5.1.1. Watershed Definition

The Goodwin Creek Watershed was used to calibrate and validate the model. Input DEM, soil type and land use maps are resampled at 90-m spatial resolution. The watershed characteristics are defined in Chapter 4.

5.1.2. Precipitation Events

CASC2D-SED was calibrated for the storm event of October 17, 1981 (event 1). This event began at 9:19 p.m. and had a total rainfall duration of 4.8 hours with very little rainfall preceding this event. Precipitation data was taken from sixteen rain gauges that are located within and just outside the watershed (see Figure 4-3). The total rainfall depth for this event varied from 66 to 78.7 mm with an average value of 73.6 mm. The average rainfall intensity was 14.7 mm/h with a maximum of 51.6 mm/h.

For the validation, two different events were chosen. The rainfall event on September 20, 1983 (event 2) had very little rainfall preceding the event, and an averaged rainfall depth twice as much as the calibration event. On the contrary, the rainfall event on August 28, 1982 (event 3) was preceded by wet conditions. The mean rainfall depth for event 3 (66.7 mm) was closer to the calibration event but the rainfall intensities were the largest of the three events.

Table 5-1 shows the summary data for the precipitation events, while the antecedent rainfall daily depths are shown in Appendix III. The input 'rainfall file' for event 1 is found also in Appendix III.

Table 5-1. Rainfall events used for calibration (event 1) and validation (event 2 and 3)

Rainfall event	Event 1 17-Oct-81	Event 2 20-Sep-83	Event 3 28-Aug-82
Rainfall duration [hr]	4.8	9.8	6
Mean rainfall depth [mm]	73.6	147.5	61.7
[1]Rainfall depth range [mm]	66.0 -78.7	135.4 - 154.9	39.1 - 91.7
Mean rainfall intensity [mm/h]	14.7	10.1	10.3
[1]Rainfall intensity range [mm/h]	0 - 51.6	0 - 65.5	0 - 90.3

[1]At any rain gauge for the rainfall event

5.1.3. Parameter Set

The model was calibrated for event 1 depending on soil type (see Table 4-1) and the LULC (see Table 4-2) and according to the range of possible values found in classical sources (Rawls *et al.*, 1983; Saxton *et al.*, 1986; Wishmeier and Smith, 1978; USDA, 1975; Woolhiser, 1975, Woolhiser *et al.*,1990). The parameter set values used in the calibration run are shown in Table 5-2 and Table 5-3. Because there is no data on the soil initial moisture a uniform value was used. Event 1 parameter set and simulation control variables are found in the 'control file' in Appendix III.

Table 5-2. Soil infiltration and erosion parameter values for the calibration run

Soil series	Drainage condition	Ks[a] [cm/s]	G[a] [cm]	Md [cm^3/cm^3]	K[b]	% Sand[c]	% Silt[c]
Calloway	Poor	0.336	22	0.29	0.4	25	55
Fallaya	Poor	0.307	14	0. 29	1	25	55
Grenada	Moderate	0.355	17	0. 29	0.2	30	60
Loring	Mod/well	0.365	22	0. 29	0.4	25	55
Collins	Mod/well	0.346	18	0. 29	0.2	30	60
Memphis	Well	0.432	22	0. 29	0.1	30	60
Gullied Land	Poor	0.384	15	0. 29	0.1	25	55

[a]Rawls *et al.* (1983); [b]Wischmeier and Smith (1978); [c]USDA (1975) texture triangle

Table 5-3. Land use parameter values for the calibration run

Land Cover	Roughness[a]	Interception[b] [mm]	C[c]	P[c]
Forest	0.25	3	0.0036	1
Water	0.01	0	0	1
Cultivated	0.15	1	0.18	1
Pasture	0.2	1.5	0.072	1

[a]Woolhiser (1975); [b]Woolhiser (1995) ; [c]Wischmeier and Smith (1978)

5.2. Results

5.2.1. Calibration Event

At the outlet (station 1), CASC2D-SED was able to correctly predict peak flow and time to peak, slightly overestimating the flows at recession. At internal locations, the rising and falling limbs of the hydrograph and the time to peak flow were correctly simulated. Peak flows were underestimated for the smallest sub-basins (stations 6, 8 and 14). For the bigger sub-basin (station 4) the peak flow was simulated properly. Simulated values of rainfall rates, water depth, infiltrated depth, and sediment concentration are shown in Figure 5-1 through Figure 5-4 . Observed and simulated values of the flow at the outlet and other internal locations are shown in Figure 5-5.

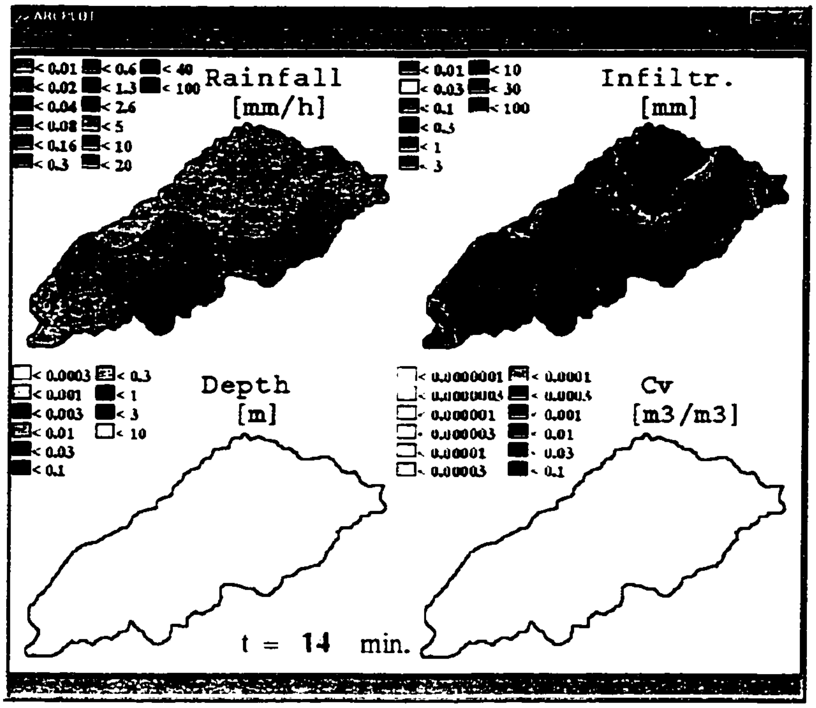


Figure 5-1. Simulated rainfall minus interception depth, infiltration depth, water depth and sediment concentration at t = 14 min. (calibration event)

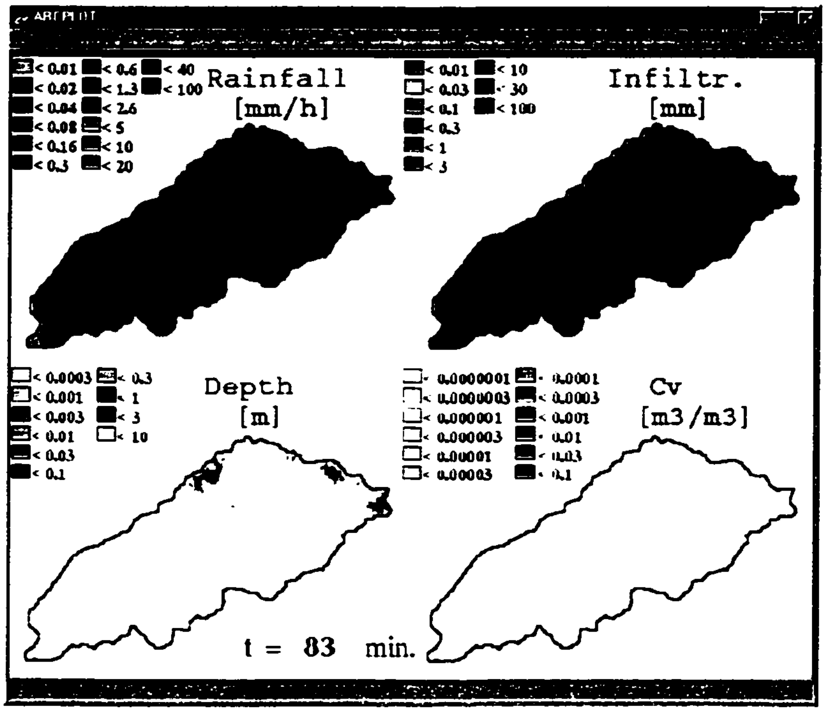


Figure 5-2. Simulated excess rainfall, infiltration depth, water depth and sediment concentration at t = 83 min. (calibration event)

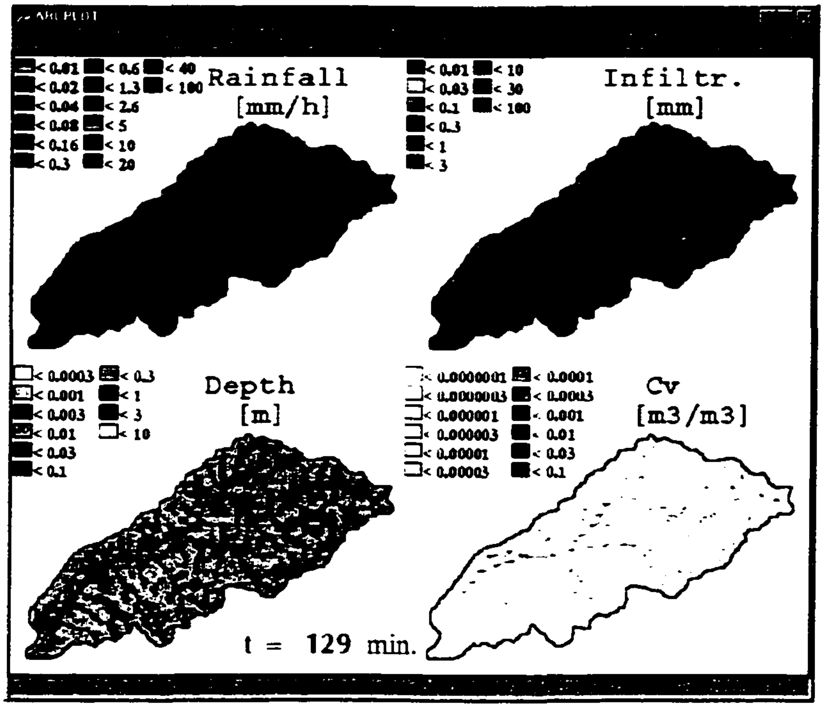


Figure 5-3 Simulated excess rainfall, infiltration depth, water depth and sediment concentration at t = 129 min. (calibration event)

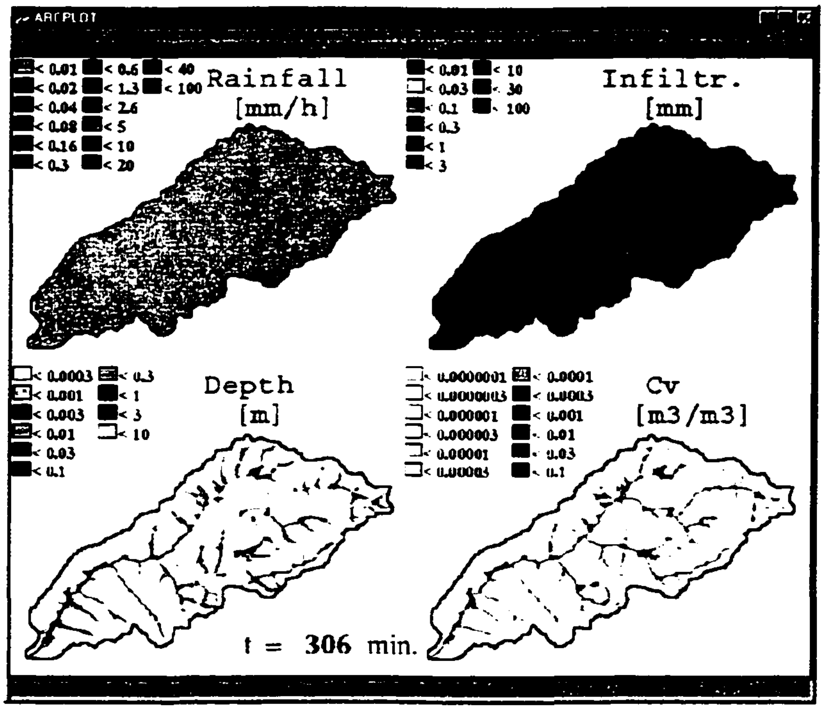


Figure 5-4. Simulated excess rainfall, infiltration depth, water depth and sediment concentration at t = 306 min. (calibration event)

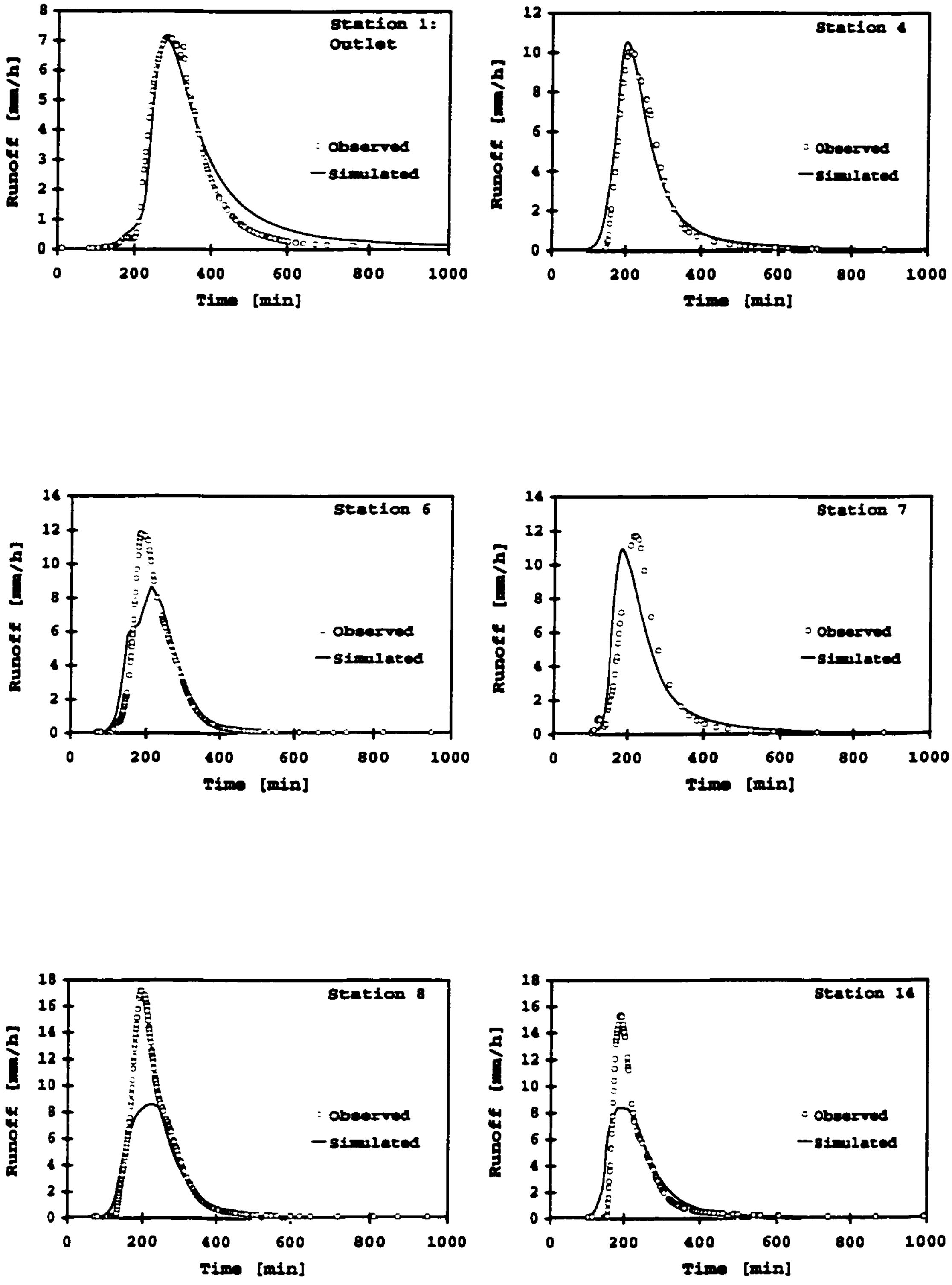


Figure 5-5. Calibrated hydrographs for the calibration event 1

The simulated sediment graphs were calibrated to match those observed at the outlet and at other internal locations. Results are shown in Figure 5-6. At the outlet, it was possible to match volumes and the shape of the sediment graph. The simulated sediment graph was delayed with respect to the observed one. At internal locations, the sediment graph rising limb was matched with observations. The falling limb was correctly predicted only for the largest sub-basins. In general, CASC2D-SED fails to predict instantaneous pulses of washload observed at internal locations.

At the beginning, only the washload fractions are found in the simulated sediment graph. Later, with increasing water discharge, the sand will add on to the total sediment load. The effect of differences in settling rates and transport capacities for each size fraction result in different percentages of the total eroded material to leave the watershed at the outlet. Although the clay fraction represented only 10-15% of the parent material, it is the fraction that corresponds to 44% of the total eroded material leaving the watershed. This fraction is transported essentially as washload. The silt fraction at the outlet represents 52% of the total sediment yield, while the sand, transported primarily as bed material, represents only 4%. These percentages of sand, silt and clay are comparable to observations, yet there are no data available to validate these results.

Figure 5-7 shows the simulated maximum flux-averaged concentration (ratio of sediment discharge to discharge) by size fraction at simulation time $t = 240$ min. The sediment graph by size fraction at the outlet is shown in Figure 5-8. Figure 5-9 shows the transported sediment either in suspension or deposition for each of the three size fractions at simulation time $t = 173$ min. While sand is transported mostly as bed-material load, clay is transported as suspended load. Silts behave as in-between case.

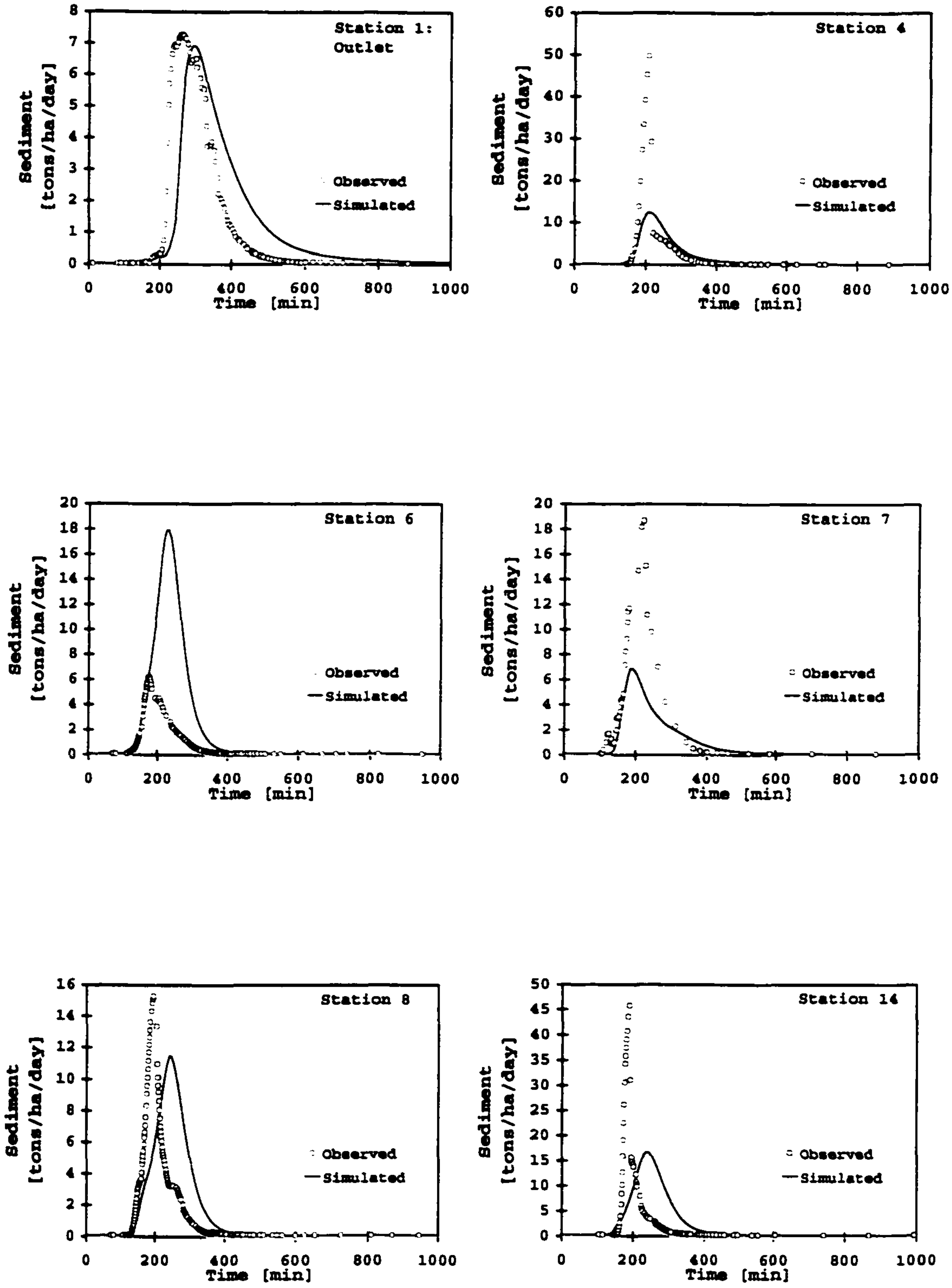


Figure 5-6. Calibrated sediment graphs for the calibration event 1

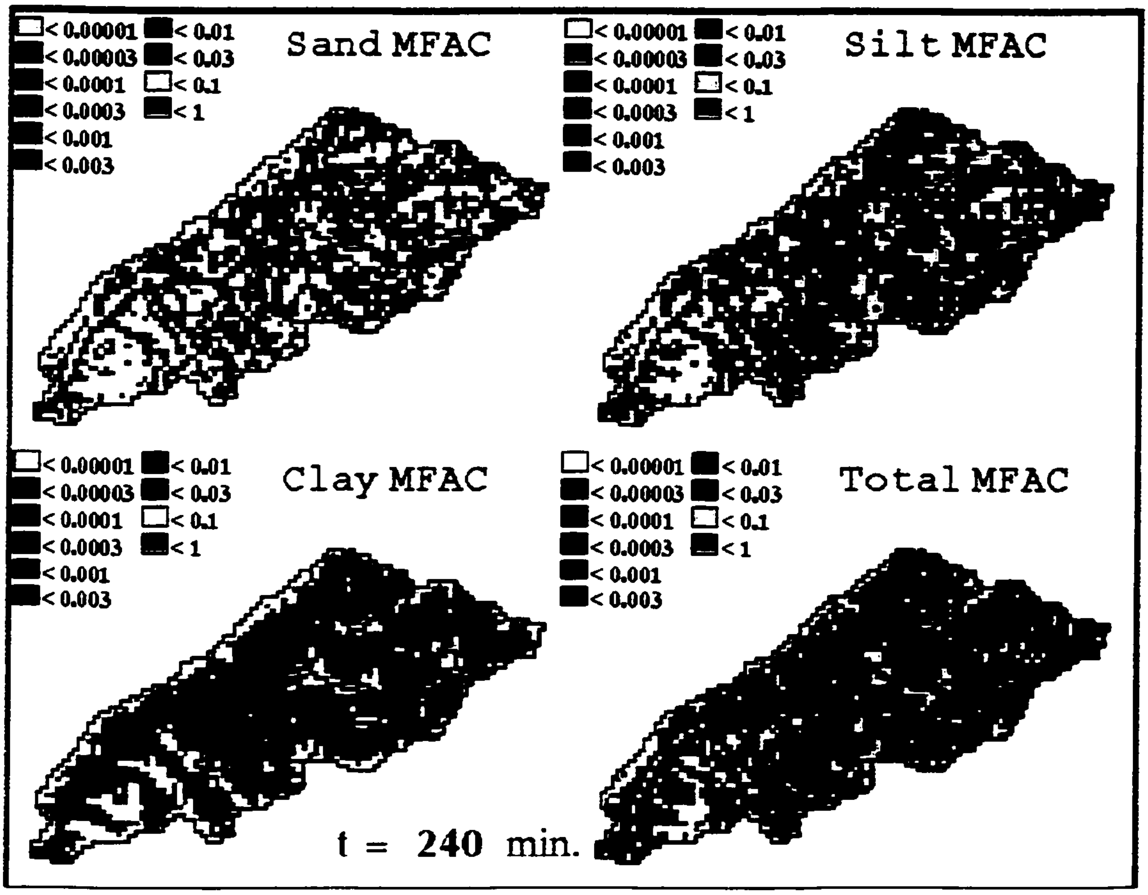


Figure 5-7. Maximum flux-averaged concentration by size fraction and total simulated in the calibration run (time 240 min.)

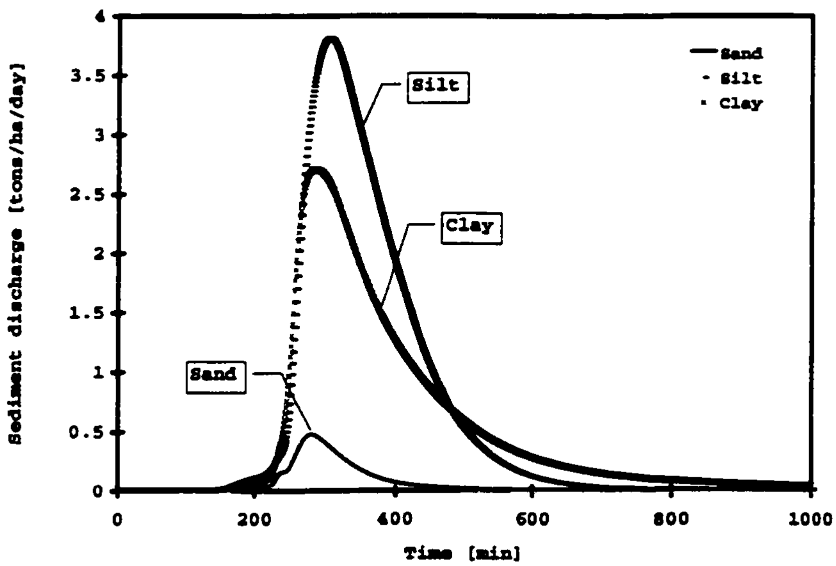


Figure 5-8. Sediment graph at the outlet by size fraction

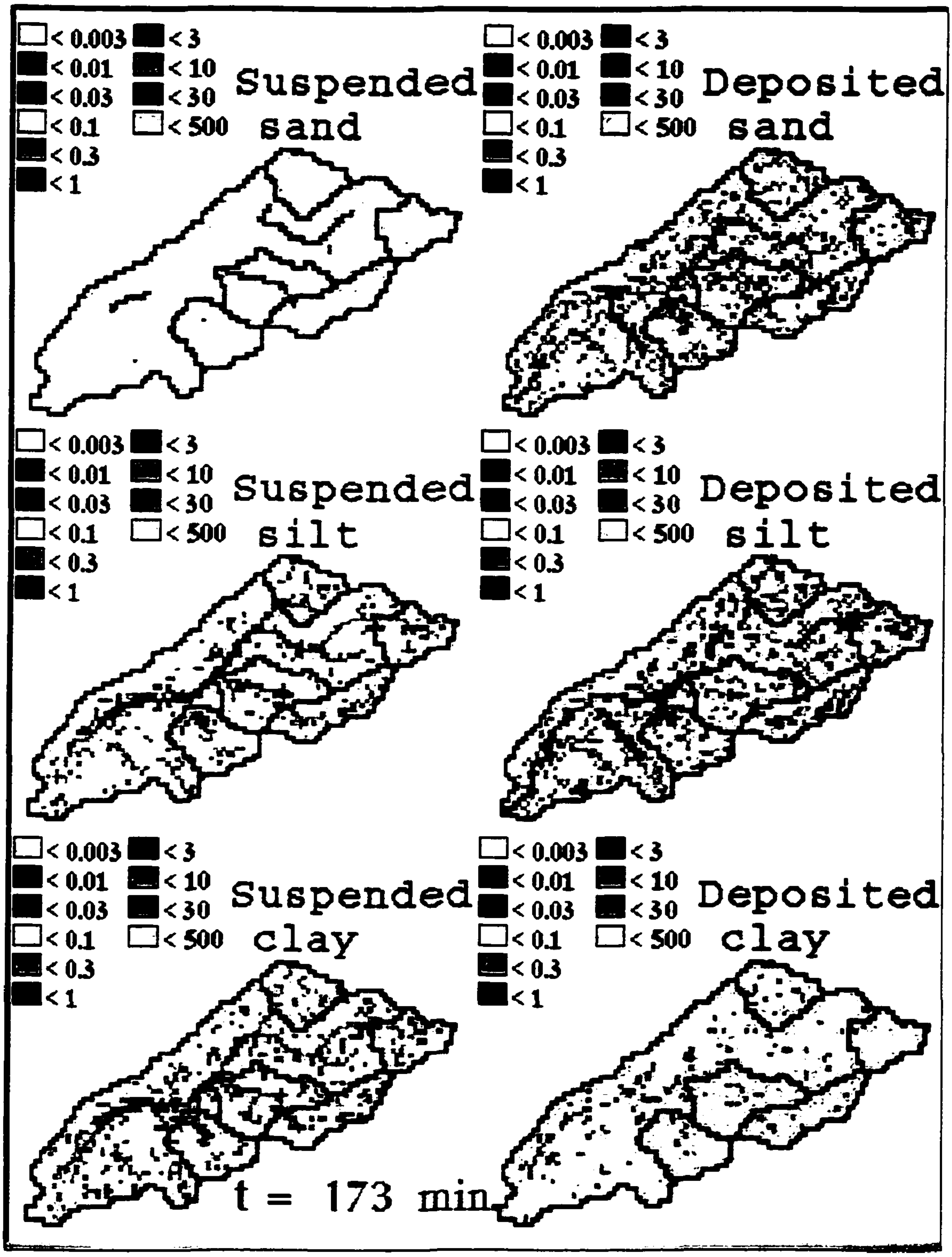


Figure 5-9. Transport of sand, silt and clay fraction in suspension or deposition mode simulated at t = 173 min (calibration event)

Spatially, there is practically no suspended sediment remaining at the end of the simulation but for a small amount of clay remaining in water sinks. For about half of the total eroded sediment deposits in the overland and about 15% deposits in channels (see Table 5-4). The percentages of the total sand, silt and clay depositing on the overland are 68%, 26% and 5% respectively. This is due to the differences in the settling velocities of each of the size fractions. There is almost no deposition of clay in channels while 26.6% of the eroded sand and 13.1% of the eroded silt deposit in the channels. The effect of both settling rates and transport capacity for each size fraction result in about 5.4%, 32.4% and 93.5% of the total eroded sand, silt and clay respectively, to leave the watershed at the outlet.

Table 5-4. Percentages of eroded sediment remaining in the overland and channel cells and leaving the watershed

Size Fraction	Remaining Overland	Remaining Channels	Yield
[1]Sand [%]	68.0	26.6	5.4
[2]Silt [%]	54.5	13.1	32.4
[3]Clay [%]	4.9	0.3	93.5
[4]Total [%]	49.7	14.7	35.0

Percentage of the total eroded [1]sand, [2]silt and [3]clay. [4] Percentage of total eroded material

In general, at internal gauging locations, the sediment volume was predicted with a difference of ±50% from observations (see Figure 5-10). The range of ±200% is generally accepted by sedimentation engineers as being acceptable when comparing computed sediment yields versus actual sediment yields (Johnson, 1997). At station 6, the sediment volume was overpredicted due to the large erosion rates simulated in a zone of cultivated land in sub-basin 6 close to the channel network.

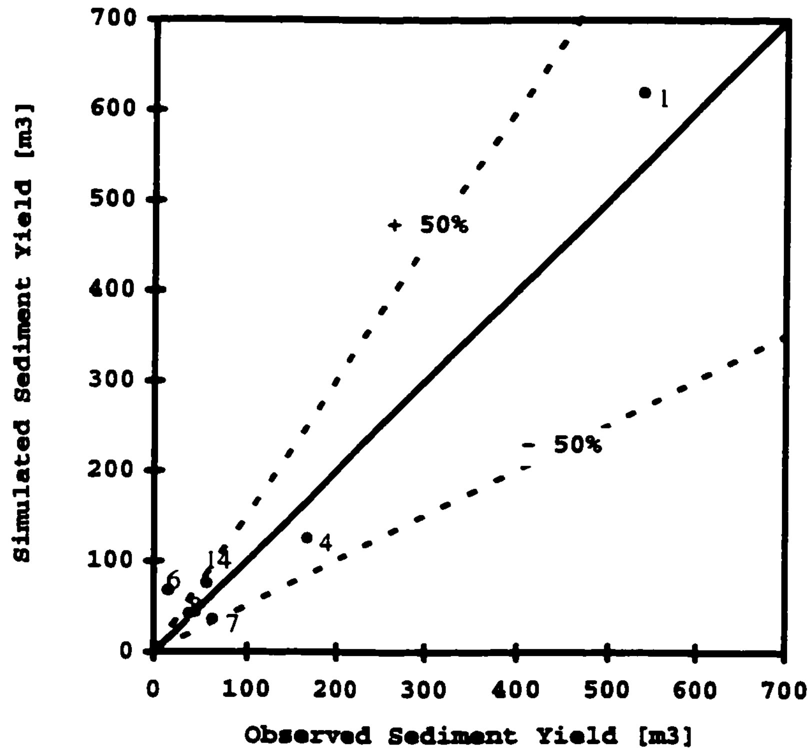


Figure 5-10. Observed vs. simulated sediment yield at the outlet (station 1) and at internal locations (stations 4, 6, 7, 8, and 14) for event 1

Spatially, net erosion values simulated for the calibration event are shown in Figure 5-11 in 3-dimensions, using the watershed DEM as the 3-d surface where the net erosion values are draped. Warm colors represent erosion while cold colors represent deposition. Using the Cs-137 technique, it has been observed (Norton, 1986; Martz and de Jong, 1987; Vesseth, 1990; Sutherland, 1991; Busacca *et al.*, 1993) that, generally, convex ridge tops and mid-slope knobs have the most erosion, while concave shapes with convergent runoff exhibit more sedimentation. It has been also observed that midslopes have either zones of erosion or deposition, suggesting they are zones of soil transport (Busacca *et al.*, 1993) and that phases of net aggradation alternate with phases of volumetric loss (Campbell, 1981). These observations are discernible in Figure 5-11.

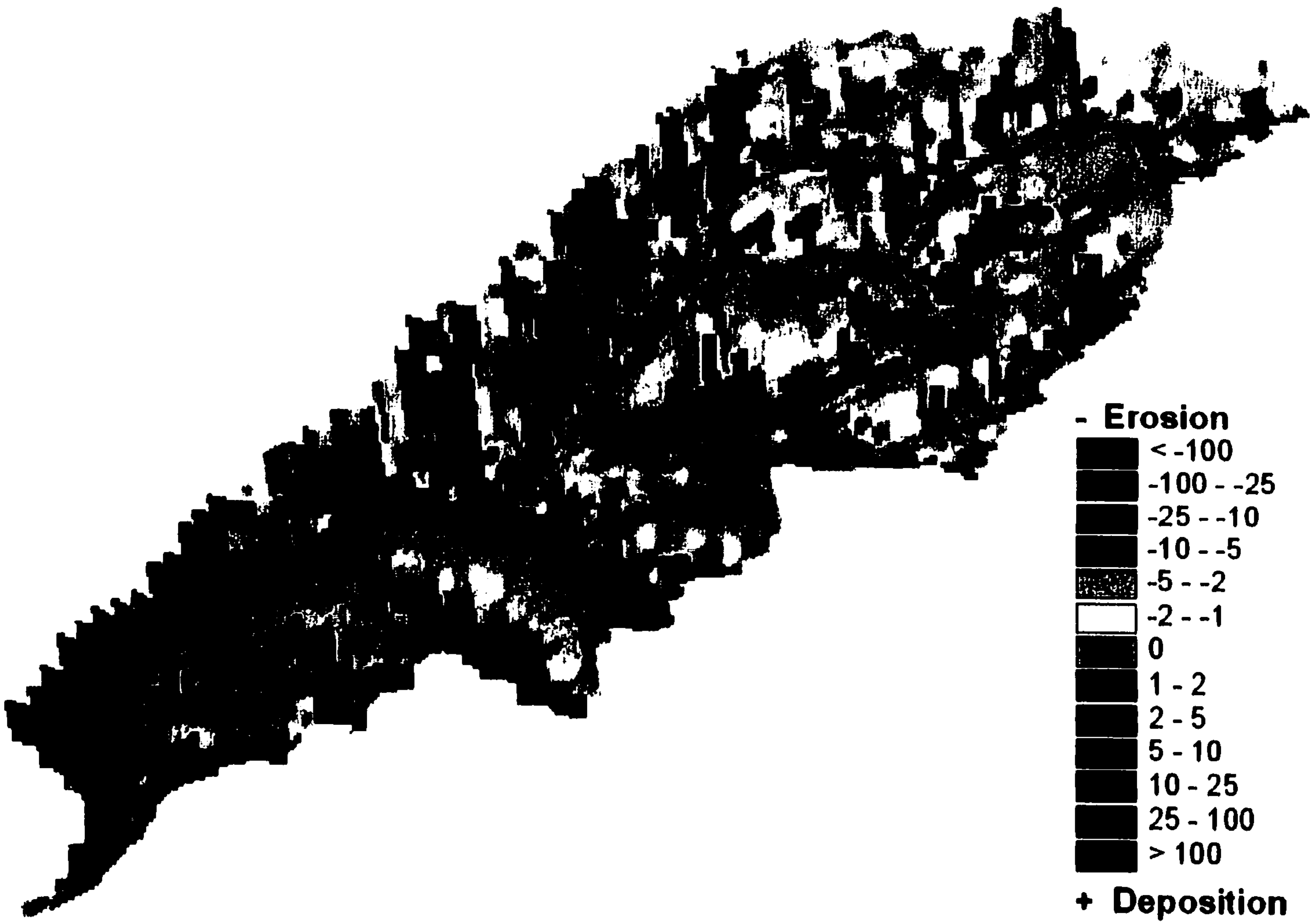


Figure 5-11. Three-dimensional view of the net erosion (tons/ha) at the end of the event 1 simulation (90-m)

The cited observations were made on agricultural lands, thus not taking into consideration land use or soil type variabilities. Taking all the factors into account, zones of net soil gain and loss may further change. Figure 5-12 shows the relationship between topography, land use and soil movement rates in the north-south direction at a cross section. In general, zones of steep slopes associated with agricultural lands or pasture are the ones presenting a net soil loss. Deposition occurs mostly in flatter or concave areas or in areas with forest land. Net erosion is observed at the watershed boundary and convex surfaces (in general, on ridges). Erosion rates at these locations are not as large as the ones predicted by equations taking only into account landscape representation (example, USLE) since predicted flow depths are small. Observed high rates of erosion in elevated or exposed areas may include aeolian transport (Sutherland, 1991) which CASC2D-SED does not simulate.

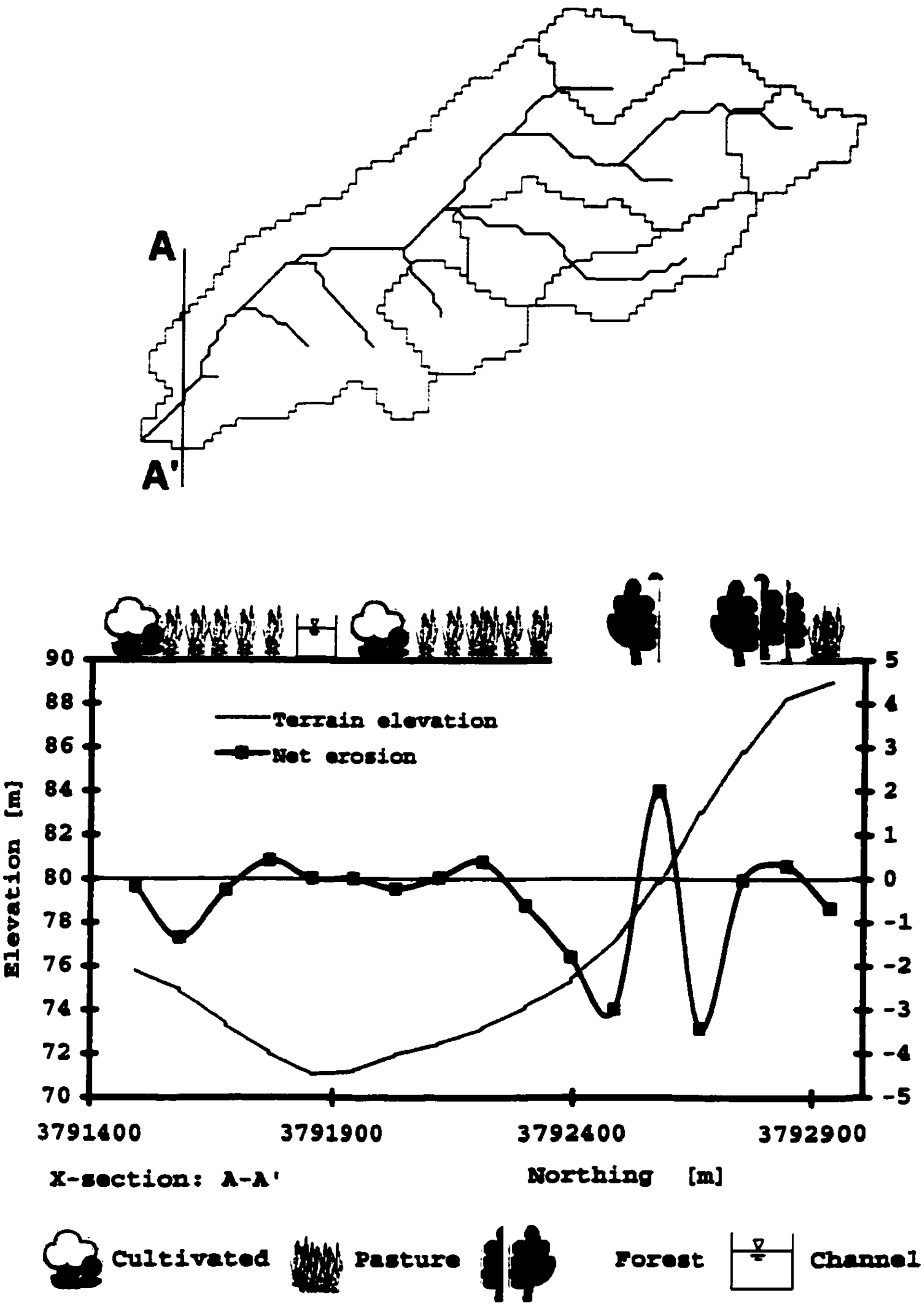


Figure 5-12. Elevation and net erosion at cross section A-A'

5.2.2. Validation Events

Two different events were used to validate the model. Event 2 has a peak discharge about three times the one observed in event 1, very dry initial conditions and a late peak. Event 3 has larger rainfall intensity rates and distribution, and wet antecedent conditions. In order to match observed and simulated hydrographs, only the initial soil moisture was changed. For event 2, the initial moisture deficit was decreased while this value was increased in the case of event 3. The observed and simulated hydrographs at the outlet and other internal locations for both events are shown in Appendix III. Figure 5-13 shows the summary of the hydrological variables for both the calibration and validation runs at all stream gauges. At the outlet, outflow volumes, peak flows and times to peak are very closed to those observed. At internal locations, outflow volumes and peak rates are in general underpredited, while the hydrographs timings are very close to those observed.

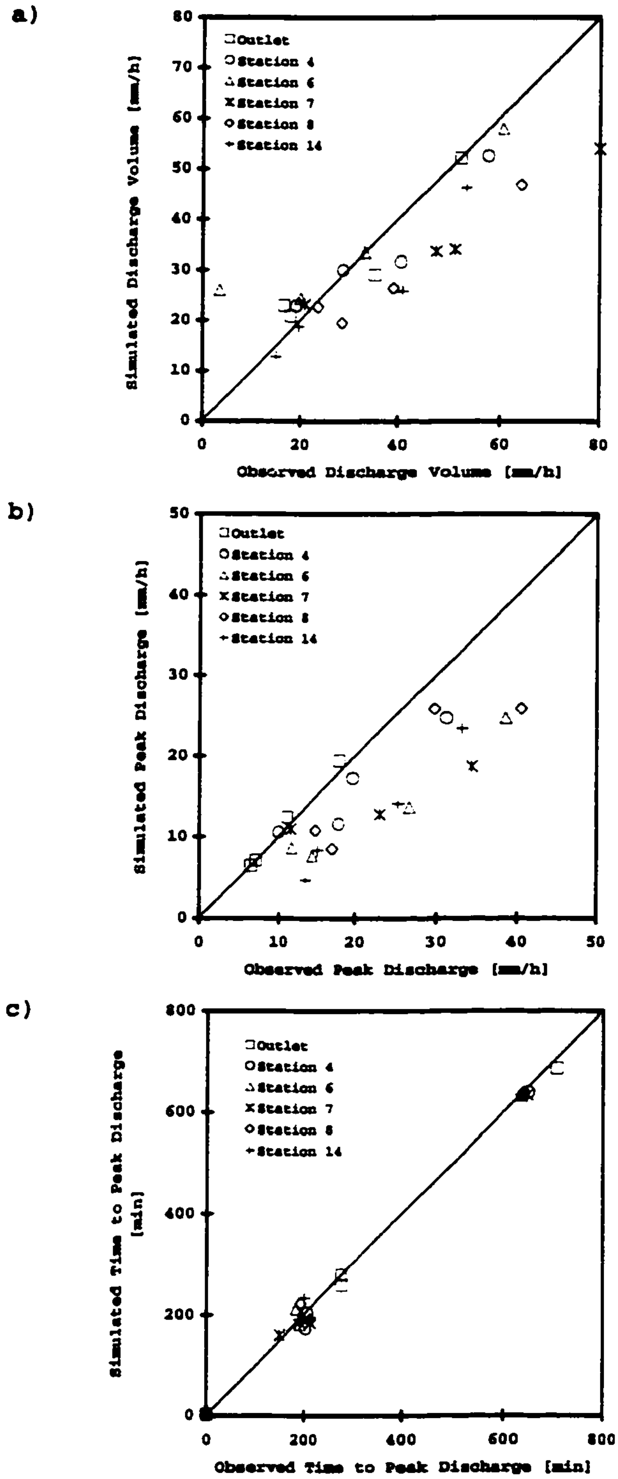


Figure 5-13. Observed vs. simulated a) discharge volume, b) peak discharge and c) time to peak for the calibration and validation events at all stations

Since the cover factor, C, is a parameter that depends on the crop conditions, it was changed a certain percentage below (for event 2) and above (event 3) the one used in the calibration run. The observed and simulated sedigraphs at the outlet and other internal locations for both events are shown in Appendix III. Figure 5-14 shows the observed vs. simulated values of the sediment yield for events 2 and 3. In both cases, the majority of the points lie within the ±50% frame, over- or underestimating for the cases of the small sub-basins.

Figure 5-15 shows the historical observed rating curves at the outlet and at stations 4 and 7. These curves show a difference of a couple of orders of magnitude between the observed flow rate and the observed sediment discharge. The simulated rating curves for the calibrated and validated events, superimposed over the observed values, lie within the range of variability of the observed ones.

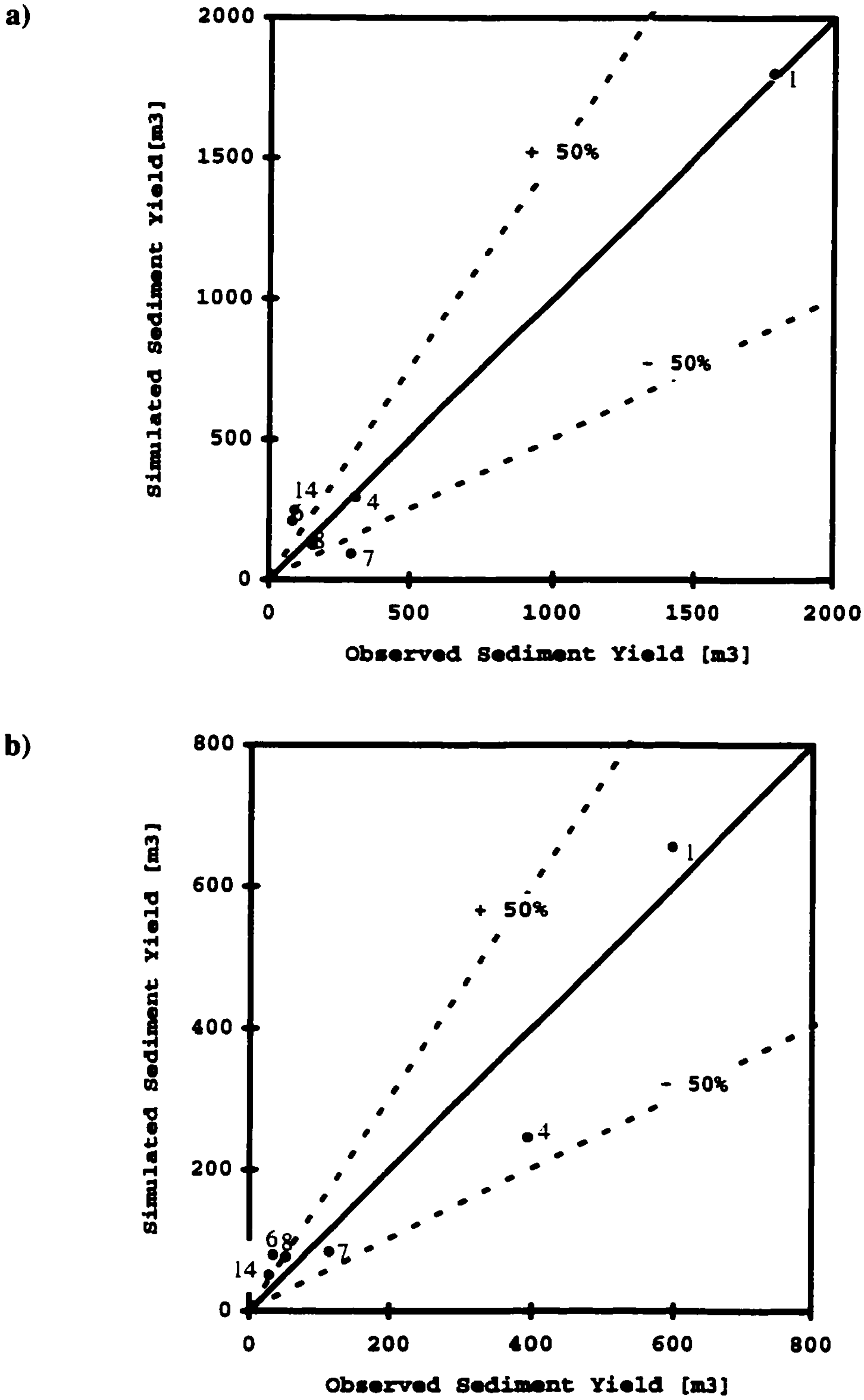


Figure 5-14. Observed vs. simulated sediment yield at the outlet (station 1) and at internal locations (stations 4, 6, 7, 8, and 14) for a) event 2 and b) event 3

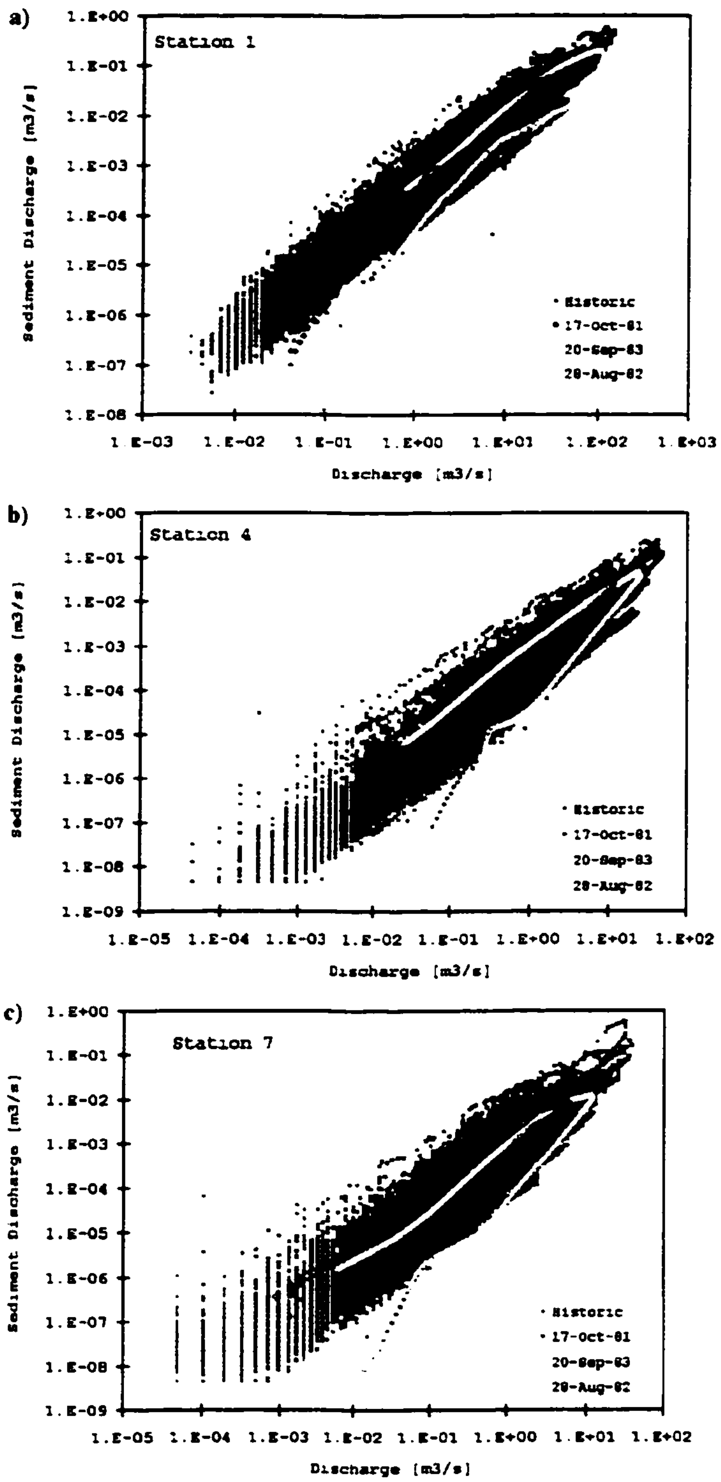


Figure 5-15. Sediment rating curves at the outlet (station 1) and at two internal nested locations (stations 4 and 7) for the calibration and validation events

5.3. SUMMARY

The model CASC2D-SED was calibrated and validated on Goodwin Creek, Mississippi, using rainfall data from 16 meteorological stations and 6 stream and sediment gaging stations. The following has been shown in this chapter:

(1) CASC2D-SED is able to simulate the hydrological response of a watershed subject to a spatially and temporally variable rainfall field. At the outlet, simulated values of peak discharge, discharge volume and time to peak discharge are very close to the observed values.

(2) CASC2D-SED is able to simulate the sediment graph at the basin outlet. Sediment yield at the outlet and at internal locations was predicted within a range of ±50% of the observed values. Observed rating curves show how for the simulated outflows range, the sediment flows vary within two orders of magnitude. Simulated rating curves fall inside the observed ones.

(3) The effect of differences in settling rates and transport capacities for each size fraction result in different percentages of the total eroded material to leave the watershed at the outlet. The clay fraction, representing the smallest percentage in the parent (source) material, made up most of the sediment yield at the outlet. This fraction is carried mostly as washload, with little deposition in the watershed or the channels. The sand fraction, carried primarily as bed load material, has the largest deposition rates and thus a small percentage of this fraction is found at the outlet.

(4) CASC2D-SED predicts erosion and deposition zones in a watershed. Visual inspection of the net erosion map show that there is erosion in convex and steep areas and deposition in concave or flat areas. In a basin cross section, it can be observed

that zones of steep slopes associated with agricultural lands or pasture are the ones presenting the largest net soil loss. Deposition occurs mostly in flatter or concave areas or in areas with forest land.

(5) Geovisualization and animation of the CASC2D-SED output grids proved helpful in the testing phase of the model. These GIS techniques are used as a form of model validation of the sediment transport dynamics and net erosion patterns where measured data are not available.

CHAPTER 6

MODELING PREDICTIONS AT DIFFERENT GRID SIZES

"In an era that emphasizes global scaled research, data aggregation is widely practiced primarily for "scaling up" environmental analyses or models from local to regional or global scales. Spatial data available at finer resolutions need to be aggregated to represent the spatial characteristics at corresponding scales.

The aggregation process may alter the statistical and spatial characteristics of the data. When aggregated data are used as input to analyses or models, the output of these analyses or models may be affected."

Ling Bian and Rachael Butler, 1999

In this chapter, differences in model predictions due to differences in spatial data aggregation in input grids are studied. As a result of decreasing slope angles defining the terrain at larger spatial resolutions, we expect the predicted erosion rates to decrease, both at the outlet and locally. The first part of this chapter shows that, theoretically, for decreasing averaged slopes, the equilibrium sediment discharge decreases.

First, to isolate the topographic effect on simulated erosion with increasing grid cell size, temporally and spatially uniform rainfall events, uniform impervious soils and uniform land use type are used. Later, the same type of results is shown for a real event case, with spatial and temporal rainfall distribution, and spatial soil type and LULC distribution.

In the USLE equation the length factor, L, is the ratio of field soil loss to the corresponding loss from 72.6-ft (22.1-m) length. In our study, 30-m is chosen to be the reference slope length. The average gross erosion (tons/ha) and sediment yield (tons/ha) are going to be normalized by the corresponding value at 30-m spatial resolution.

6.1. CONCEPTUAL EQUILIBRIUM UNIT SEDIMENT DISCHARGE

Julien and Simons (1984) developed a general sediment transport capacity relationship supported by dimensional analysis. The recommended formula can be written as a power function of slope and discharge, the most significant parameters pointed out by their study:

$$q_s = \alpha S^{\beta} q^{\gamma} \qquad [6\text{-}1]$$

where S is the terrain slope and q is the unit discharge. α, β and γ are empirical exponents. The exponent β varies between 1.2 and 1.9 while the exponent γ varies between 1.4 and 2.4.

In particular, the Kilinc and Richardson equation ($\beta = 1.66$, $\gamma = 2.035$) reveals that predicted erosion rates strongly depend on the hydraulics of the system and the topography. These two factors will be modified with grid aggregation and therefore the prediction of soil erosion will change. Errors in the prediction of the unit discharge or slopes will produce augmented errors in the estimation of the net erosion rates.

Goodwin Creek defined at different spatial resolutions (see CHAPTER 4) is used to examine the effects of grid aggregation on the equilibrium sediment discharge. Assuming that this basin is a square watershed at x-meter grid cell size with N number of cells and of uniform slope $\overline{So}$ [m/m] equal to the watershed averaged slope for the given watershed spatial resolution, the length of the watershed is estimated to be:

$$L = \sqrt{N * x^2} \qquad [6\text{-}2]$$

Assuming a uniform constant rainfall intensity, i, impervious condition and uniform LULC and soil type, we can estimate the average equilibrium unit sediment discharge, $\overline{q_s}$, for each grid cell size, x, using a first order approximation to the Taylor Series expansion as:

$$\overline{q_s} = \alpha\,(i * L)^{2.035}\,\overline{So}^{1.66} \quad \text{or} \quad \frac{\overline{q_s}}{\alpha} = (i * \sqrt{N} * x)^{2.035}\,\overline{So}^{1.66} \qquad [6\text{-}3]$$

where α is a constant that depends on the soil and land cover.

The mean slopes have been described in section 4.2.1 and they decrease with increasing grid cell size according to the next relationship:

$$\overline{S} = 17.9 * x^{-0.497} \qquad [6\text{-}4]$$

The number of cells defining Goodwin Creek decreases with increasing grid cell size according to the following expression:

$$N = 2E + 07 * x^{-2.01} \qquad [6\text{-}5]$$

Substituting Equations 6-4 and 6-5 into Equation 6-3, we get for the averaged conditions that:

$$\frac{q_s}{\alpha} = 4.14E9 * i^{2.035} * x^{-0.86} \qquad [6\text{-}6]$$

For a given rainfall intensity, i, we can relate the equilibrium unit sediment discharge at x-m grid cell size, qs_x, to a reference equilibrium sediment discharge at 30-m, qs_{30} as follows:

$$\frac{qs_x}{qs_{30}} = \left(\frac{x}{30}\right)^{-0.86} \qquad [6\text{-}7]$$

This is, on the average, as grid cell size, x, increases, the equilibrium unit sediment discharge is expected to decrease.

6.2. MODEL APPLICATION AT DIFFERENT SPATIAL RESOLUTIONS

6.2.1. Method

Uniform Conditions: Spatial and temporal rainfall distributions must be considered as possible sources of error. This error will produce variability in the computed flow (Ogden and Julien, 1993). Thus, temporally and spatially uniform rainfall events have been selected as input to the CASC2D model for testing purposes.

The simulations are carried out using three different constant intensities: 5, 10 and 15mm/h. The rainfall duration was 1200 minutes so that the hydrograph equilibrium conditions are reached for each of the resolutions. The simulation duration was long enough (2500 minutes) to allow for the basin to drain at all resolutions.

The soil parameters are spatially uniform and correspond to the averaged values for a silty-loam soil, the dominant soil type in the watershed. The infiltration and erosion parameter values used in this case are shown in Table 6-1.

Table 6-1. Soil parameter values for an impervious silty loam soil

SoilType	Drainage condition	Ks [cm/s]	G [cm]	Md [cm³/cm³]	K[a]	% Sand[b]	% Silt[b]
Silty loam	Moderate	0	0	0	0.2	20	65

[a]Wischmeier and Smith (1978); [b]USDA (1975) texture triangle

The land use parameters are uniform and correspond to a pasture cover, which represents the largest percentage in the basin. The parameter values used are shown in Table 6-2.

Table 6-2. Land use parameter values for pasture

Land Cover	Roughness[a]	Interception[b] [mm]	C[b]	P[b]
Pasture	0.2	1	0.02	1

[a]Woolhiser (1975); [b]Woolhiser (1990); [c]Wischmeier and Smith (1978)

The channels width and depth are described in (Section 4.1). The roughness coefficient for all the channels at all resolutions are kept constant (n = 0.035).

Real Event: The calibration event of October 17, 1981 at 30-m was used to test the effects of grid cell size on the model erosion predictions when spatial variations in soil, land use and rainfall are taken into account.

The model was run at spatial resolution larger than 30-m using the same hydrological parameters. As a result of grid aggregation, peak discharges decrease and times to peak are delayed for the larger grid cell sizes (see Figure 6-1a). The runoff volumes decrease with increasing grid cell size because a larger portion of the water is infiltrated (see Figure 6-1b). Smaller average slopes and progressive vanishing of the larger slopes for the larger grid cell sizes result in a larger opportunity for the water to infiltrate. This in turn reduces the flow unit discharge. As already shown in the last section, reducing the flow unit discharge will induce a reduction in the transport capacity as calculated by the Kilinc and Richardson equation. This is, the reduction in the slope with increasing grid cell sizes will affect the transport capacity both directly, through the slope, S, term, and indirectly, through the unit discharge term, q.

The model is then calibrated at spatial resolutions greater than 30-m to get similar peak discharges and runoff volumes at all spatial resolutions by increasing the infiltration rates through the saturated hydraulic conductivity. It is reasonable to assume that flatter slopes would lead into more saturated areas in the watershed, and thus, the moisture deficit has been reduced for increasing grid cell sizes. Finally, the channel roughness coefficient has been increased with increasing grid cell sizes to cause a delay in the hydrograph peak. After calibration, the new peak discharges, time to peak, and percentages of runoff and infiltration volumes are shown in Figure 6-2a,b.

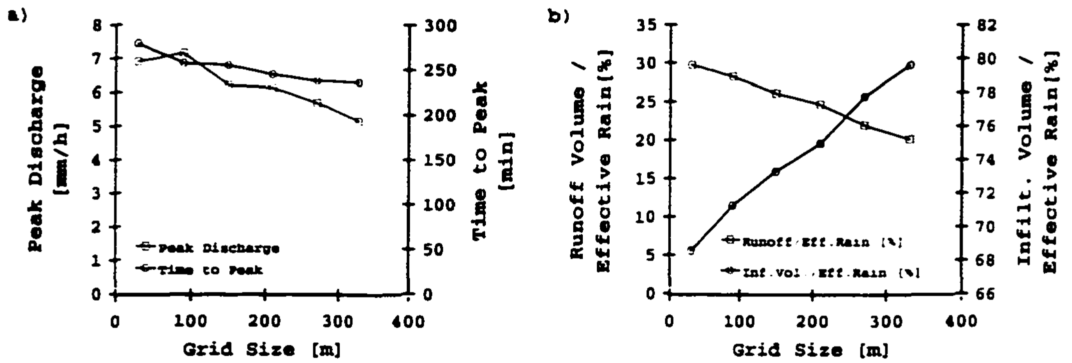


Figure 6-1. a) Peak discharge and time to peak and b) percentage of runoff and infiltration for different grid sizes before model calibration

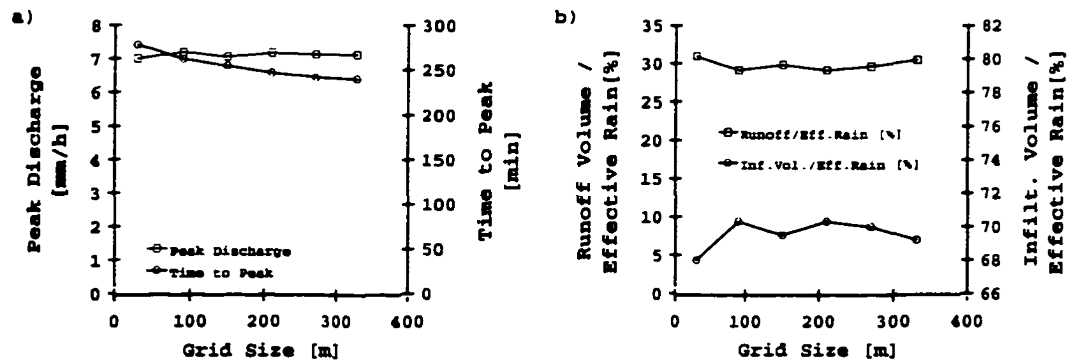


Figure 6-2. a) Peak discharge and time to peak and b) percentage of runoff and infiltration for different grid sizes after model calibration

6.2.2. Results

Equilibrium Unit Sediment discharge: In the case of uniform rainfall conditions, a regression line has been fitted to the simulated equilibrium sediment discharge values normalized by the respective value at 30-m spatial resolution for each rainfall intensity.

This equation is similar to the empirical one (Equation 6-7) obtained for the averaged conditions.

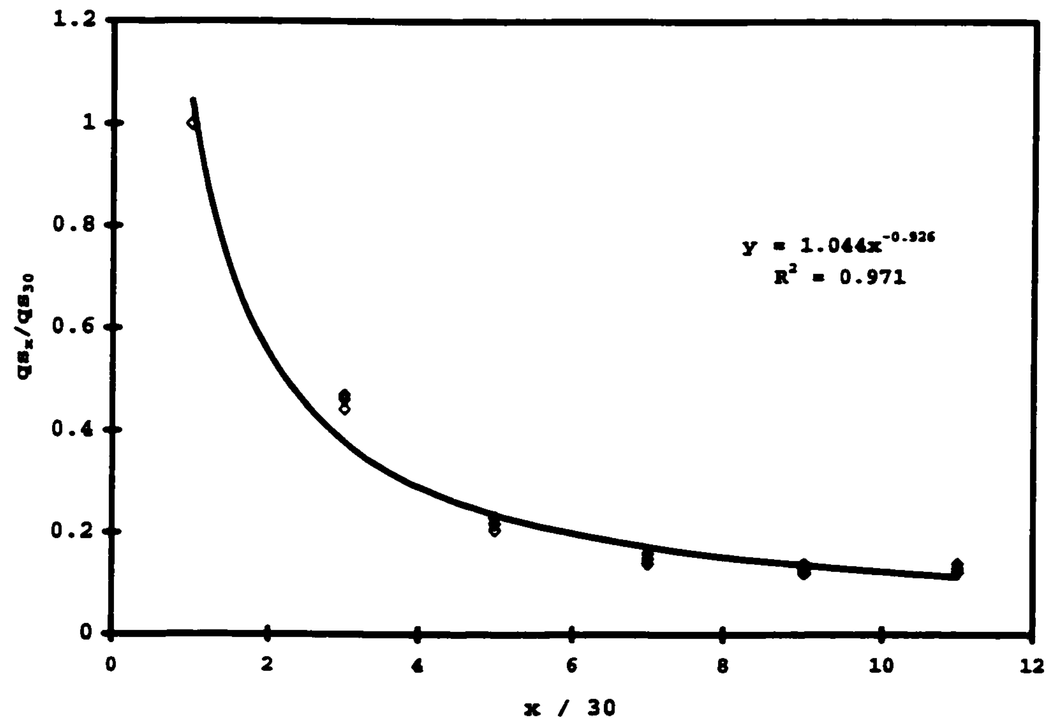


Figure 6-3. Normalized equilibrium sediment discharge at the watershed outlet for different normalized grid cell values and rainfall intensities

Average gross erosion and average sediment yield: Average gross erosion is defined as the volume of parent material per unit area [tons/ha] that has been scoured in the basin, without taking into account deposition. Average sediment yield is defined as the amount of sediment per unit area [tons/ha] leaving the watershed. Both quantities have been normalized at all spatial resolutions by the respective amount obtained at the 30-m simulations. Figure 6-4 shows the normalized average gross erosion and sediment yield for the case of uniform rainfall (with 5, 10 and 15 mm/h intensity), soil type and LULC. Figure 6-5 shows the same results for the distributed conditions case. As expected, simulated erosion values decrease with grid cell size with the strongest dependency found for the case of the averaged gross erosion.

The normalized averaged gross erosion in the case of the uniform and the distributed conditions are practically identical. The next approximate regression line is fitted to the model simulated values in both cases:

$$\frac{\overline{E}_x}{\overline{E}_{30}} = 170 * x^{-1.5} \qquad [6\text{-}8]$$

where

E_x = gross erosion simulated at x-m [tons ha^{-1}]
E_{30} = gross erosion simulated at 30-m [tons ha^{-1}]
x = grid cell size [m]

For the averaged conditions, if the slope decreases with the inverse of the squared root of the cell size (Equation 6.4), then the unit discharge should decrease with the inverse of the cubic root of the cell size as shown in the next expression:

$$\left.\begin{array}{l} \dfrac{\overline{Ex}}{\overline{E}_{30}} \propto \left(\left(\dfrac{x}{30}\right)^{-0.5}\right)^{1.66} * \left(\left(\dfrac{x}{30}\right)^{a}\right)^{2.035} \\ \dfrac{\overline{E}_x}{\overline{E}_{30}} \propto x^{-1.5} \end{array}\right\} \Rightarrow a \cong -1/3 \qquad [6\text{-}9]$$

Thus, on the average, the unit discharge also depends on the grid cell size although the effect of the changes in slope values with grid cell sizes in the transport equation will be larger.

The averaged sediment yield takes into account not only the amount of sediment eroded in the watershed but also deposition. Thus, a change in grid cell size will affect less the model sediment yield predictions than it does the gross erosion (see Figure 6-4 and Figure 6-5). For the smaller spatial resolution models, sediment deposits on concave areas or areas of low slope. As the grid spatial resolution increases, a) the convexities and concavities of the terrain tend to disappear and thus the rates of local erosion and deposition, respectively, decrease and b) slope lengths become shorter and the channels become less sinuous which causes a faster routing of the sediment to the outlet.

The normalized averaged sediment yield is affected by the spatial variability of the rainfall, soils and land use input. The regression line fitted for the uniform and spatially variable cases are (also shown in Figure 6-4 and Figure 6-5):

$$\left[\frac{Y_x}{Y_{30}}\right]_{\text{UniformConditions}} = 32.2 * x^{-0.995}$$

$$\left[\frac{Y_x}{Y_{30}}\right]_{\text{Spatial variability}} = 50.8 * x^{-1.153}$$

This is, spatial variabilities will increase the difference in the sediment yield predictions between the model at certain spatial resolution, x-m, and the reference one, 30-m. Resampling of the soil type and land use grids to coarser resolution affects the sediment yield prediction by affecting deposition patterns

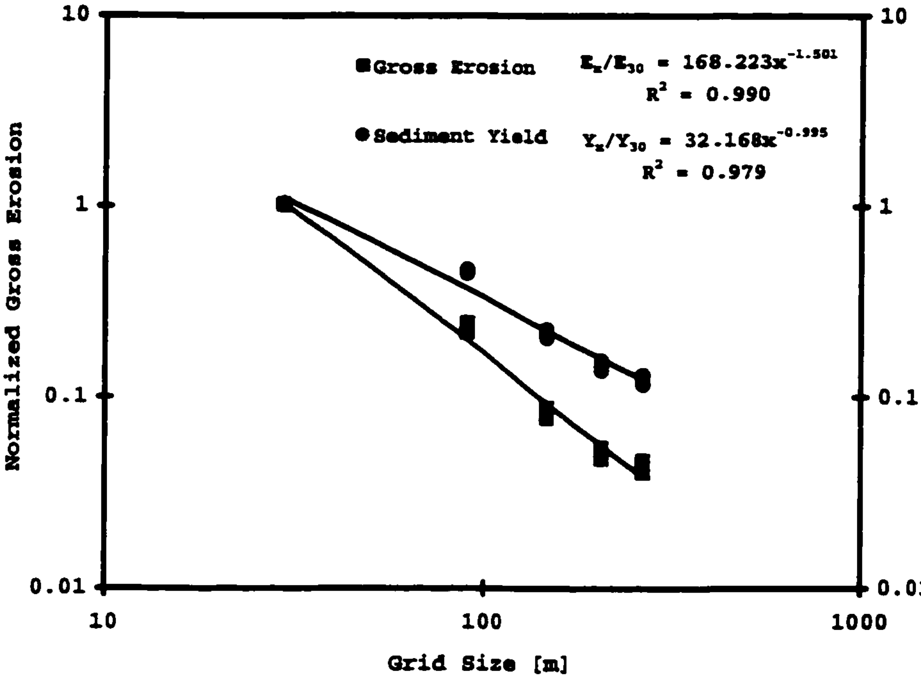


Figure 6-4 Normalized averaged gross erosion and sediment yield for the uniform conditions case

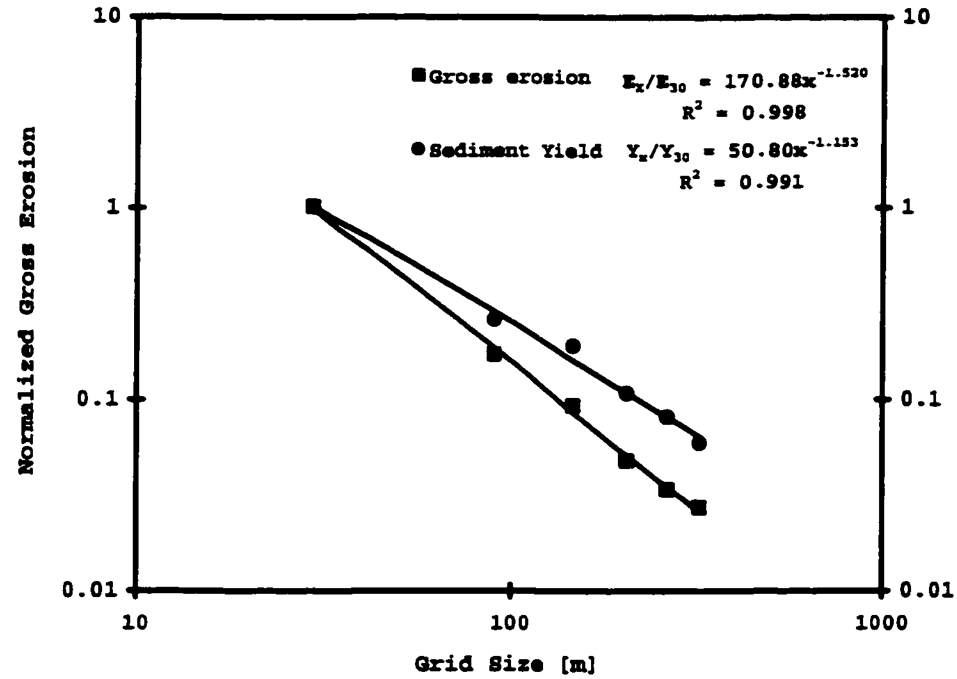


Figure 6-5. Normalized averaged gross erosion and sediment yield for the distributed conditions case

Net erosion volumes: Net erosion values are the result of adding deposited volumes (positive quantity) and scoured volumes (negative quantity) of sediment at a given grid cell. Spatially, net erosion volumes change depending on the input grid spatial resolution as shown in Figure 6-7 (uniform conditions with 15 mm/h rainfall intensity) and Figure 6-8 (distributed conditions). The net erosion pattern also change with the distributions of soil and land use (compare Figure 6-7 and Figure 6-8).

Visual inspection of the net erosion maps indicates that the 30-m resolution grid is the one simulating net erosion values closer to what it would be expected while the 330-m does not perform as well. Figure 6-6a represents the percentage of the watershed area with either net erosion, deposition or zero net erosion as seen in Figure 6-7. Figure 6-6b represents the percentage of the watershed area with either net erosion, deposition or zero net erosion in Figure 6-8. In Figure 6-6a, the 30-m grid is the one has the largest area covered by net deposition (28%) and the smallest area in which the net erosion is zero (5%) while in the 330-m grid the opposite happens (10% with net deposition and 25% with no net erosion). The largest percentage of net erosion has a maximum on the 150-m grid. In Figure 6-6b, the trends are similar to those in the uniform conditions case, where the 30-m grid has the largest percentage of the area covered by net deposition and the coarsest resolutions have the largest percentage of the area in which the net erosion is zero. The 90-m grid has maximum percentage of the area covered by net erosion

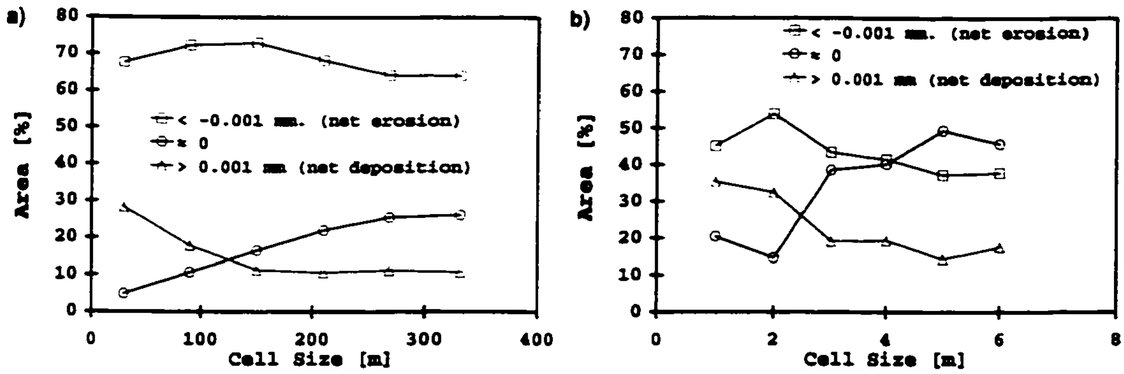


Figure 6-6. Percentage of the watershed with net erosion, deposition, and zero net erosion for the case a) of the uniform conditions (15 mm/h rainfall intensity) and b) distributed conditions (event 1)

Location of the sediment source and deposition areas, and volume of sediment eroded transported and delivered are some of the factors that should be quantified before deciding what erosion control treatment is best suited for a problem. In this case, it seems advisable to recommend that simulations with the smaller grid cell sizes are carried out.

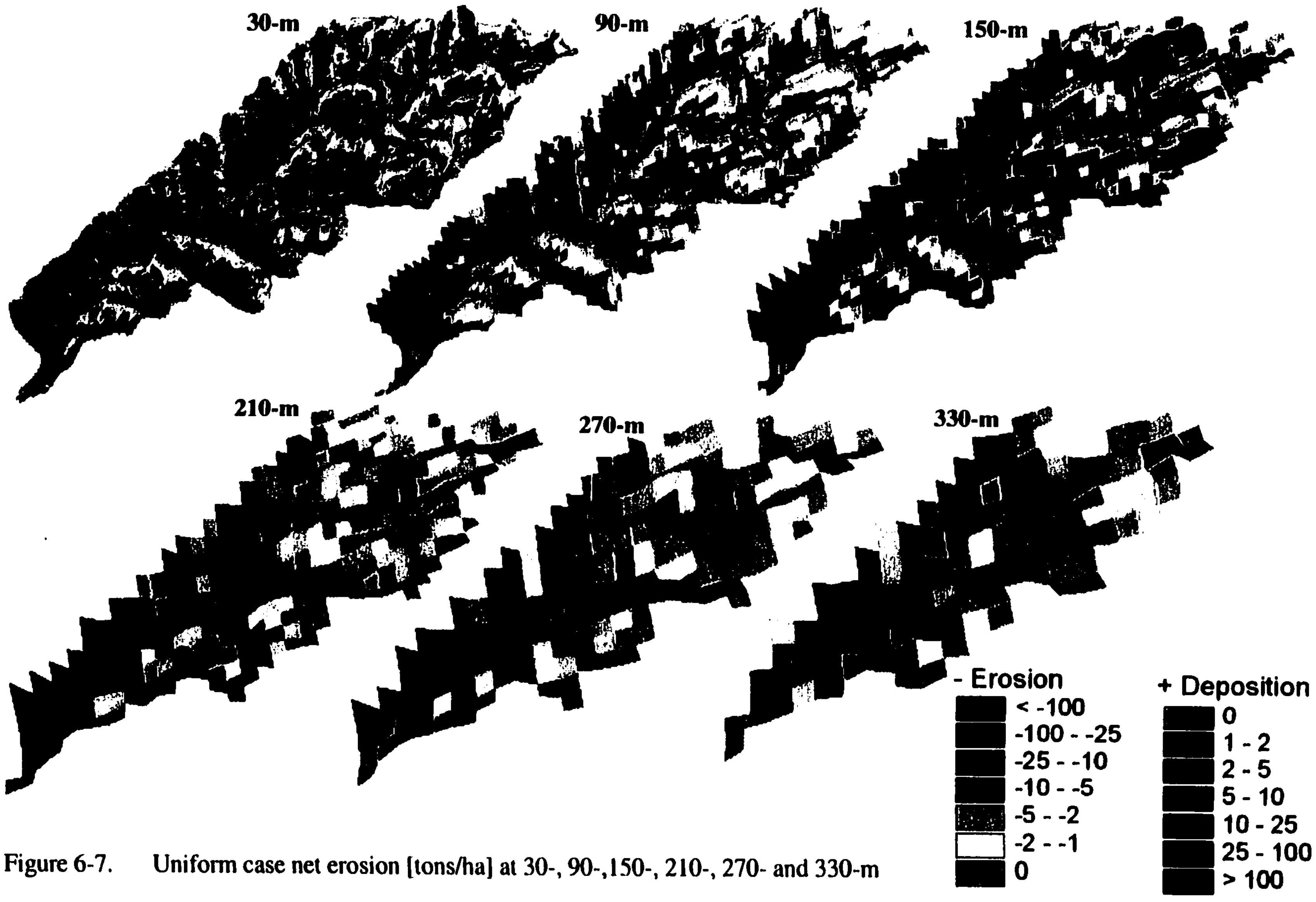


Figure 6-7. Uniform case net erosion [tons/ha] at 30-, 90-,150-, 210-, 270- and 330-m

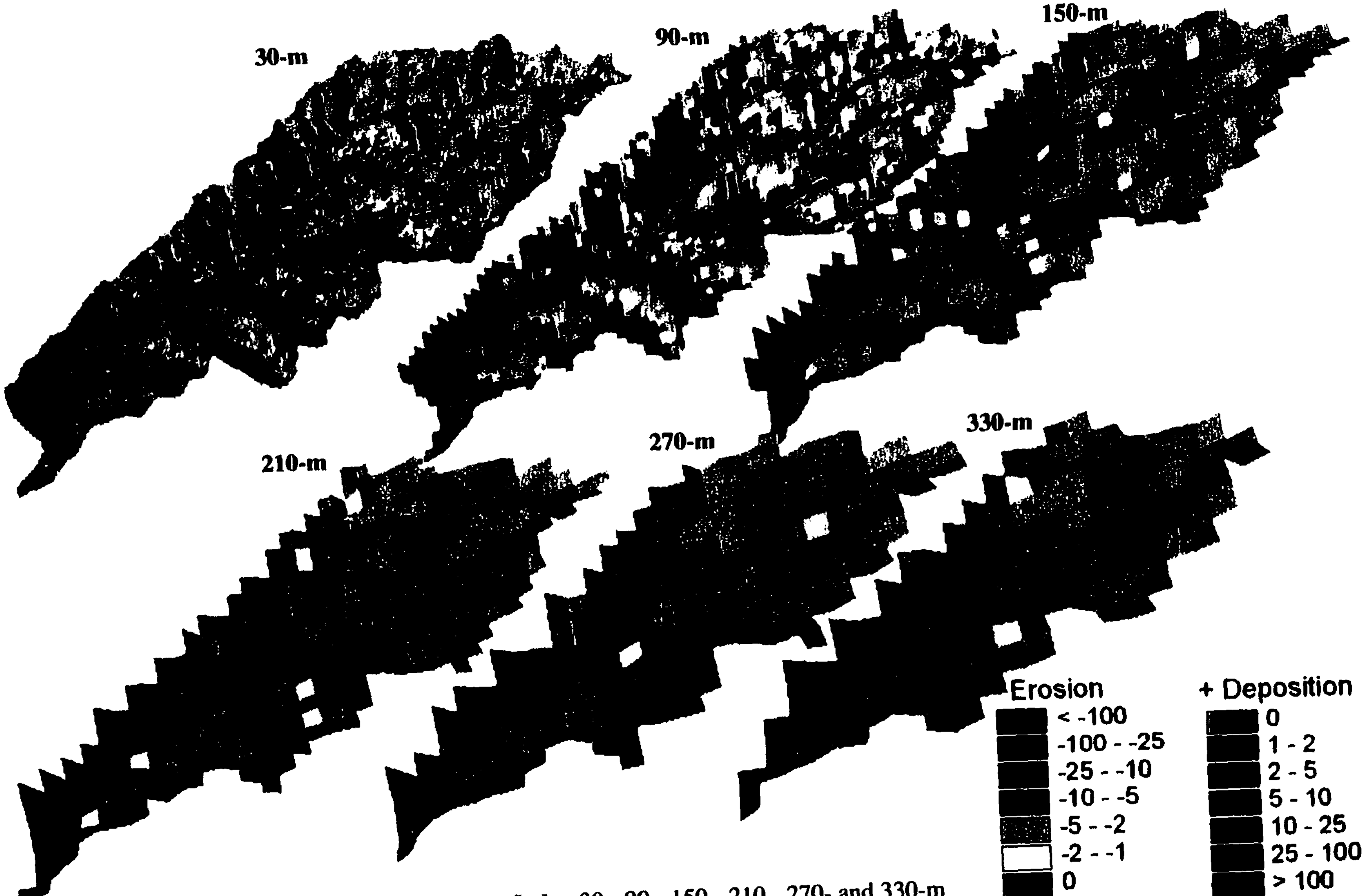


Figure 6-8. Distributed case net erosion [tons/ha] at 30-, 90-, 150-, 210-, 270- and 330-m

Sediment delivery ratio (SDR): The SDR denotes the ratio of the sediment yield at a given stream cross section to the gross erosion from the watershed upstream from the measuring point. The SDR depends primarily on the drainage area of the upstream watershed (Julien, 1995). Figure 6-9a and Figure 6-10a show the SDR obtained for the case of the uniform conditions and for the case of the distributed ones, respectively. Figure 6-9b,c,d and Figure 6-10b,c,d show the SDR for the particular case of the sand, silt and clay fractions, respectively.

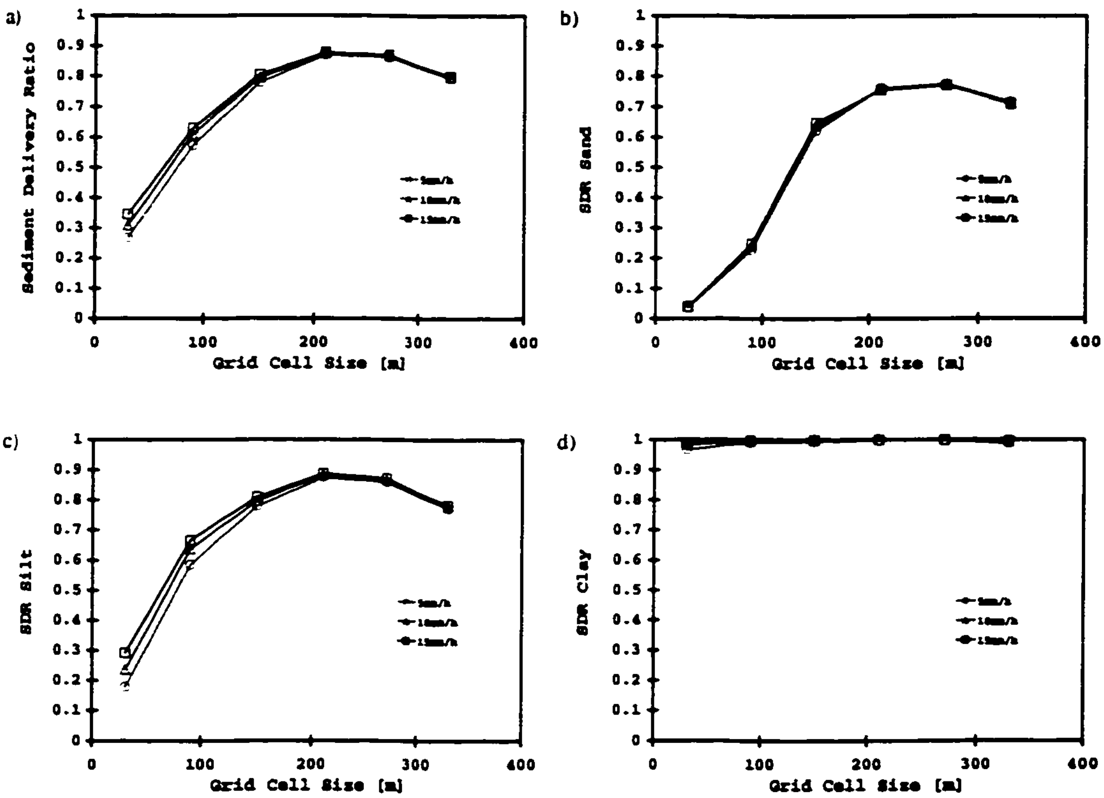


Figure 6-9. Sediment delivery ratio of Goodwin creek watershed for a) total sediment load, b) sand load, c) silt load and d) clay load as a function of grid size and rainfall intensity for the uniform case

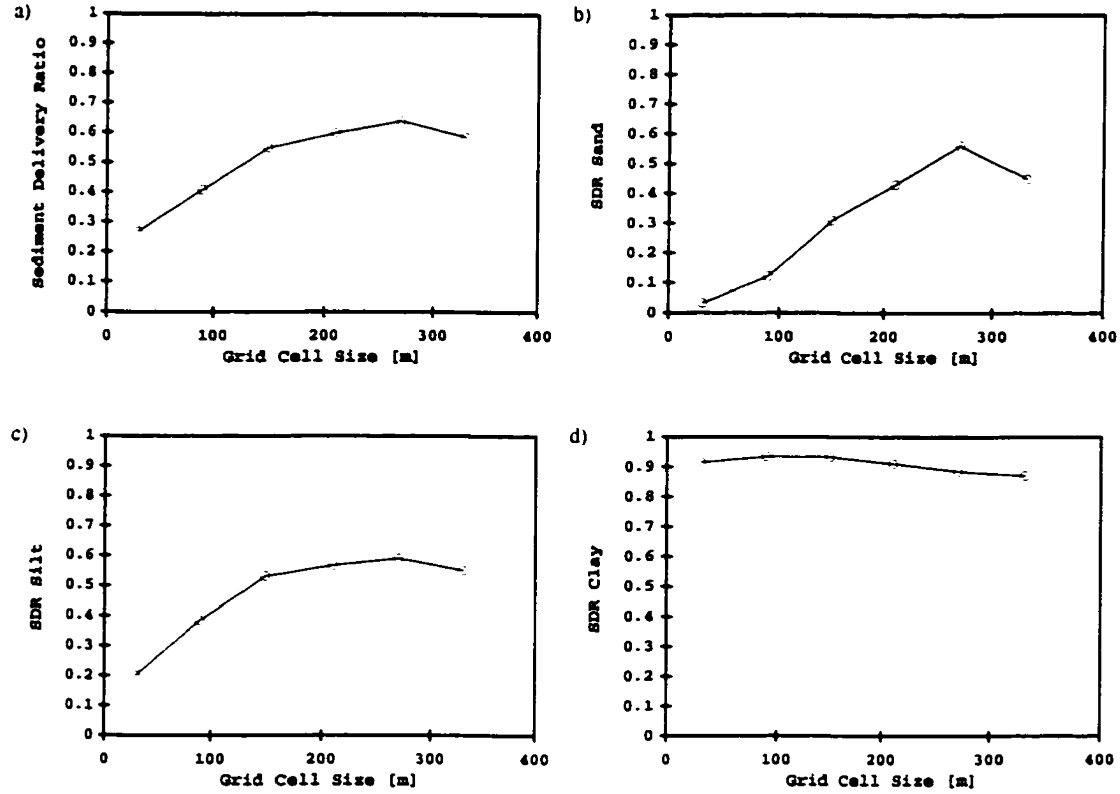


Figure 6-10. Sediment delivery ratio of Goodwin creek watershed for a) total sediment load, b) sand load, c) silt load and d) clay load as a function of grid size for the distributed case

The distribution of the SDR of sand and clay does not change much with rainfall intensity (see Figure 6-9b,d). The reason why this happens is that the sand fraction moves mostly as bed material and the clay as suspended load due to their large and very small settling velocities, respectively. In the case of the silt (Figure 6-9c) the rainfall intensity or, equivalently, the water discharge affects its transport rate. With larger water depths for the larger rainfall intensity, larger fraction of the silt will move as suspended sediment and thus, for the same spatial resolution, the fraction of silt leaving the basin increases for larger rainfall intensities. The small differences in the SDR with precipitation intensity are due to differences in the transport of the silt fraction.

Basically, the SDR for the distributed conditions case follows the same pattern as the one for the uniform conditions although the range of variation of the SDR with grid cell size is smaller in the first case.

Sediment delivery ratios increase with grid cell size up to the 210-m grid in the case of the uniform conditions. As shown in Figure 6-9, for grid sizes 210-m or larger, the net soil loss area starts decreasing and the area of zero net erosion increases while the area of net deposition is maintained. This induces a reduction in the SDR values for grid sizes larger than 210-m. In the case of the distributed conditions, SDRs increase up to the 270-m grid (see Figure 6-10). It should be noted that the sum of areas with net erosion and the areas of zero net erosion has a maximum at the 210-m grid in the case of the uniform conditions and 270-m grid in the case of the distributed conditions.

According to Boyce (1975), the SDR for a drainage basin of 20.5 km^2 (7.9 mi^2) has been observed to vary between 0.1 and 0.5. Figure 6-11 shows the SDR values of the simulated event 1 (from 30- to 330-m) and events 2 and 3 (at 90-m) superposed with the Boyce data. Falling within the range of observed data, the SDRs simulated for the case of the 30- and 90-m grid cell size are comparable to field observations. SDRs for larger grid cell sizes greater are overpredicted.

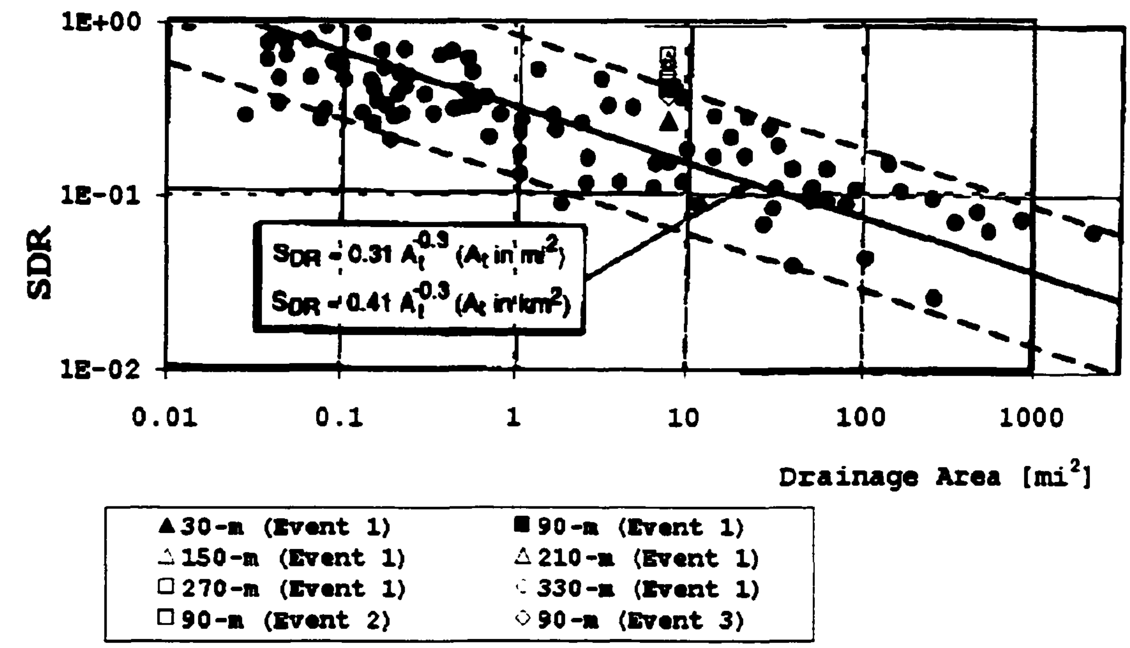


Figure 6-11. Boyce observed sediment delivery ratio values and simulated ones with CASC2D-SED

6.3. SUMMARY

Differences in model erosion predictions due to differences in spatial data aggregation in input grids have been studied for two different cases. In the uniform conditions case, rainfall intensity, soil type and land use type are maintained uniform. In the distributed conditions case, an observed rainfall event, and distributed soil and land use types have been used. The following has been shown:

(1) Empirical and simulated equilibrium sediment discharge at spatial resolution x is related to the value simulated at 30-m spatial resolution approximately according to the following equation:

$$\frac{qs_x}{qs_{30}} = \left(\frac{x}{30}\right)^{-0.9}$$

(2) Gross erosion depends, among others, on terrain slope and unit discharge, with the stronger dependency on the first one. The normalized gross erosion decreases with $x^{-1.5}$, with x representing the grid cell size.

(3) Predicted sediment yield values are less sensitive to the model grid cell size than gross erosion predictions because it takes into account not only erosion but also deposition rates. The normalized sediment yield decreases approximately with the inverse of the grid cell size. Resampling of the soil type and land use grids affects simulated averaged sediment yield.

(4) Spatially, net erosion volumes change depending on the input grid cell size. The area covered with net deposition decreases while the area with zero net soil erosion increases. These values affect in turn the sediment delivery ratio (SDR). Simulated values for the smaller grid cell sizes (30-m to 150-m) represent erosion and deposition patterns more realistically than the coarser resolution grids.

(5) SDRs increase with increasing grid cell size. SDRs are not greatly affected by the rainfall intensity. Differences in the SDR for the smaller spatial resolutions are due to differences in the silt SDR. The SDR simulated for the case of the 30- and 90-m grid cell size for the uniform conditions and the distributed conditions fall within observed SDR range by Boyce (1975) for the Goodwin Creek drainage area.

CHAPTER 7

SUMMARY AND CONCLUSIONS

7.1. SUMMARY

In summary, the CASC2D-SED model was reviewed in chapter 3. The interception process has been included in the current version of the model. The upland erosion scheme has been modified to include sediment routing by advection in capacity limited transport conditions. The sediment routing by size fraction has been added in channels. In channels, suspended sediment is transported by advection processes regardless of the transport capacity. In hillslopes and channels, the transport capacity is first fulfilled with the suspended sediment volume and, if needed, with the previously deposited sediment. After moving the suspended and previously deposited sediment, any remaining transport capacity will be used to erode the upland cells. The newly implemented sediment settling concept allows for a differential treatment of silt and sand fractions in upland cells and allows sediment deposition in impoundments. As a result, silt and clay fractions are transported as washload while sand moves as bed-material load. Finally, in Chapter 3, the output time-series grids from CASC2D-SED have been coupled with Arc/Info® for their automatic display and, if desired, the creation of an MPEG movie.

In Chapter 4, the extensively monitored Goodwin Creek watershed has been selected as the study site to test the new CASC2D-SED erosion model. The raster GIS input data is resampled from 30-m up to 330-m to study the effects of grid cell size on upland erosion simulations.

In chapter 5, the CASC2D-SED model is calibrated for one event and validated for two other events. Resulting hydrographs and sediment graphs are obtained at the outlet and at other internal locations. The sediment transport at the basin outlet is shown by size fractions for the calibration event. Observed and simulated values of sediment yield are obtained for all the gaging stations and events. Geovisualization of the erosion dynamics shows sediment transport by size fraction and displays spatial erosion and deposition patterns for the calibration event.

In chapter 6, the effects of grid cell size on upland erosion rates are presented for two different cases. In the first case, the precipitation is uniform with constant rainfall intensity, the soils are impervious and there is only one soil and land use type. In the second case, the rainfall is spatially and temporally distributed and the soils and land use are spatially distributed. The effects of grid resolution on simulated erosion rates are presented in terms of gross erosion, sediment yield, spatial distribution of net erosion and sediment delivery ratios.

7.2. CONCLUSIONS

With reference to the objectives stated in Chapter 1, the results of the calibration and validation runs of CASC2D-SED on Goodwin Creek, MS, defined from 30-m to up to 330-m lead to the following conclusions:

(1) **a.** CASC2D-SED is able to simulate the hydrological response of a watershed subject to a spatially and temporally variable rainfall field. Simulated hydrographs and sediment graphs are very close to the observed ones. Predicted sediment yields vary within a range of ±50% of the observed values. This is in agreement with observed rating curves in the basin.

b. The differences in settling velocities and transport capacities for each size fraction result in different percentages of eroded material leaving the watershed at the outlet. The clay fraction, representing the smallest percentage in the parent material, made up most of the sediment yield at the outlet. This fraction is carried mostly as washload, with little deposition in the watershed or the channels. The sand fraction, carried primarily as bed-material load, has the largest deposition rates and thus a small percentage of this fraction is found at the outlet.

(2) Geovisualization of the CASC2D-SED output grids in terms of MPEG movies shows the time sequence of spatial distribution of upland erosion or sedimentation, suspended sediment concentration, sediment flux by size fraction and net aggradation / degradation on the watershed. This three-dimensional representation shows that the simulated zones of erosion are generally coupled with steep slopes, depending on the land use or soil type. Zones of deposition were simulated in valley bottoms or forested areas.

(3) **a.** Resampling the Goodwin Creek watershed from 30-m to up to 330-m affects the basin representation by changing the slope distribution and the channel network definition. Resampling of the basin's DEM to 330-m results in channel and drainage area capturing. The area percentage for land use and soil type do not change significantly with resampling.

b. The following general conclusions are derived from the study of the grid cell size effect on the upland erosion predictions. For an increasing grid cell size (x):

a. Equilibrium sediment discharge decreases approximately with the inverse of the grid cell size. Gross erosion depends, among other factors, on terrain slope and unit discharge, with the stronger dependency on the first one. The normalized gross erosion decreases with $x^{-1.5}$. Taking into account not only erosion but also deposition, the predicted sediment yield values are less sensitive to the model grid cell size than the gross erosion predictions. The normalized sediment yield decreases approximately with the inverse of the grid cell size. Resampling of the soil type and land use grids affects simulated averaged sediment yield

b. The area covered with net deposition decreases while the area with negligible soil erosion increases with grid cell size. These values affect in turn the sediment delivery ratio (SDR). Simulated values for the smaller grid cell sizes (30-m to 150-m) represent erosion and deposition patterns more realistically than the coarser grids.

c. The sediment delivery ratios increase for increasing coarser spatial resolutions and are not greatly affected by rainfall intensity. Differences in SDR for the smaller spatial resolutions are due to differences in the silt fraction. The clay SDR is close to one, indicating that most of the eroded clay leave the basin as washload. The SDR simulated for the case of the 30- and 90-m grid cell size for the uniform conditions and the distributed conditions fall within observed SDR range by Boyce (1975) for the Goodwin Creek drainage area

The selected grid cell size for the erosion model should be able to reasonably represent the landscape and enable the identification of erosion-vulnerable regions in a watershed for the implementation of suitable erosion control measures. One may conclude that grid cell sizes smaller than 150-m when modeling erosion with CASC2D-SED are preferable because: (1) they represent erosion and deposition patterns more realistically than the coarser resolution grids: and (2) the values of the sediment delivery ratios are closer to field measurements. Larger grid cell sizes are discouraged due to distortions in the landscape representation and the coarse resolution at which the location of erosion and deposition areas are simulated.

CITED REFERENCES

Acevedo, W. and Masuoka, P. (1997). "Time-series animation techniques for visualizing urban growth." *Computers & Geosciences*, 23(4), 423-435.

Adler, M.-J., Stancalie, G., and Dragoi, E. (1997). "Analysis of erosion in a catchment affected by a landslide." *Human impact on erosion and sedimentation*, D. E. Walling and J.-L. Probst, eds., IAHS, 135-144.

Alonso, C. V. (1996). "Hydrologic Research on the USDA Goodwin Creek Experimental Watershed, Northern Mississippi." *Procs. 16th AGU Hydrology Days*, Colorado State University, 25-36.

Alonso, C. V., Bingner, R. L., and Kuhnle, R. A. (1995). "Sediment Yield From Watersheds With Eroding Channels." *Water Resources Engineering : Proceedings of the First International Conference*, Am. Soc. of Civil Eng., 1183-1187.

Andrienko, N., Andrienko, G., and Gatalsky, P. (2000). "Mapping spatio-temporal data for exploratory analysis." *GeoInformatics*, 3.

APOALA project (2001). "Geographic Visualization (geovisualization)." <http://www.geovista.psu.edu/grants/apoala/vis.htm> (July 2001).

Arnold, J. G. and others (1996). "SWAT: Soil and Water Assessment Tool -- User's manual." USDA-ARS, Temple, TX.

Bates, P. D. and Lane, S. N. (1998). "Preface: High resolution flow modelling in hydrology and geomorphology." *Hydrological processes*, 12(8), 1129-1130.

Beasley, D. B., Huggins, L. F., and Monke, E. J. (1989). "ANSWERS: a model for watershed planning." *Transactions of the ASAE*, 23(4), 938-944.

Bennett, J. P. (1974). "Concepts of mathematical modeling of sediment yield." *Water Resources Research*, 10(3), 485-492.

Beven, K. (1995). "Linking parameters across scales: subgrid parameterizations and scale dependent hydrological models." *Hydrological processes*, 9, 507-525.

Beven, K. and Kirby, M. J. (1979). "A physically based, variable contributing area model of basin hydrology." *Hydrological Sci. Bull.*, 24, 43-69.

Bian, L. and Butler, R. (1999). "Comparing Effects of Aggregation Methods on Statistical and Spatial Properties of Simulated Spatial Data ." *Photogram. eng. and remote sensing* , 65(1), 73-84.

Blackmarr, W. A. (1995). "Documentation of Hydrologic, Geomorphic, and Sediment transport Measurements on the Goodwin Creek Experimental Watershed, Northern

Mississippi, for the Period 1982-1993 - Preliminary Release." *Research Report No. 3 (CD-ROM),* U.S, Dept. of Agriculture. Agric. Research Service..

Blaser, A. D., Sester, M., and Egenhofer, M. J. (2000). "Visualization in an early stage of the problem-solving process in GIS." *Computers & Geosciences*, 26(1), 57-66.

Borah, D. K., Alonso, C. V., and Prasad, S. N. (1982). "Routing graded sediment in streams: formulations." *J. Hydraulic Division*, 108(12), 1486-1503.

Boyce, R. (1975). "Sediment Routing With Sediment Delivery Ratios." *Procs. Present and Prospective Technology for Predicting Sediment Yields and Sources. USDA-ARS-S-40.* ARS - USDA, Washington, D.C. 61-65.

Bras, R. L. (1990). *Hydrology, an introduction to hydrologic science*, Addison -Wesley.

Brigham Young University (2001). "Environmental Modeling Research Laboratory: WMS." <http://emrl.byu.edu/wms.htm> (July 2001).

Brooks, S. M. and McDonnell, R. A. (2000). "Research advances in geocomputation for hydrological and geomorphological modeling towards the twenty-first century." *Hydrological processes*, 14(11-12), 1899-1907.

Bruneau, P., Gascuel-Odoux, C., Tobin, P., Merot, Ph., and Beven, K. (1995). "Sensitivity to space and time resolution of a hydrological model using digital elevation data." *Hydrological processes*, 9(1), 69-81.

Busacca, A. J., Cook, C. A., and Mulla, D. J. (1995). "Comparing landscape scale estimation of soil erosion in the palouse using Cs-137 and RUSLE." *J. Soil and Water Conservation*, 48(4), 361-367.

Campbell, I. A. (1981). "Spatial and Temporal Variations in Erosion Measurements." *Procs. Erosion and Sediment Transport Measurement*, IAHS-AISH Publication No. 133, 448-456.

Cartwright, W. (1997). "New media and their application to the production of map products ." *Computers & Geosciences*, 23(4), 447-456.

Chang, K. and Tsai, B. (1991). "The effect of DEM resolution on slope and aspect mapping." *Cartography and Geographic Information Systems*, 18(1), 69-77.

Chow, V. T., Maidment, D. R., and Mays, L. W. (1988). *Applied hydrology*, McGraw-Hill.

Clarke, R. T. (1994). *Statistical Modelling in Hydrology*, John Wiley & Sons.

Davis, T. J. and Keller, C. P. (1997). "Modeling and visualizing multiple spatial uncertainties ." *Computers & Geosciences*, 23(4), 397-408.

De Jong, S. M. (1994). "Applications of reflective remote sensing for land degradation studies in a Mediterranean environment." *The Netherlands Geographical Studies Vol. 177.*

De Roo, A. P. J. (1996a). "The LISEM project: an introduction." *Hydrological processes*, 10(8), 1021-1025.

De Roo, A. P. J. (1996b). "Soil Erosion Assessment using GIS." *Geographical information systems in hydrology*. Kluwer Academic Publishers. Dordrecht ; Boston, 339-356.

De Roo, A. P. J., Hazelhoff, L., and Burroughs, E. R. (1989). "Soil erosion modeling using ANSWERS and Geographical information systems." *Earth surface processes and landforms*, 14, 517-532.

De Roo, A. P. J., Offermans, R. J. E., and Cremers, N. H. D. T. (1996a). "LISEM: a single-event, physically based hydrological and soil erosion model for drainage basins: II. Sensitivity analysis, validation and application." *Hydrological processes*, 10(8), 1119-1126.

De Roo, A. P. J., Wesseling, C. G., and Ritsema, C. J. (1996b). "LISEM: a single-event physically based hydrological and soil erosion model for drainage basins: I. Theory, input and output." *Hydrological processes*. 10(8), 1107-1117.

De Roo, A. P. J., Wesseling, C. G., and Van Deursen, W. P. A. (2000). "Physically based river basin modelling within a GIS: the LISFLOOD model." *Hydrological processes*, 14(11-12), 1981-1992.

DiBiase, D., MacEachren, A. M., Krygier, J. B., and Reeves, C. (1992a). "Animation and the role of map design in scientific visualization." *Cartography and Geographic Information Systems*, 19(4), 201-214.

DiBiase, D., MacEachren, A. M., Krygier, J. B., and Reeves, C. (1992b). "Animation and the role of map design in scientific visualization." *Cartography and Geographic Information Systems*, 19(4), 265-266.

Doe, W. W., Jones, D. S., and Warren, S. D. (1999). "The Soil Erosion model guide for the military land managers: Analysis of erosion models for natural and cultural resources applications." Center for Ecological Management of Military Lands. Colorado State University, Fort Collins, CO.

Doe, W. W., Saghafian, B., and Julien, P. Y. (1996). "Land-use impact on watershed response: the integration of two-dimensional hydrological modelling and geographical information systems." *Hydrological processes*, 10, 1503-1511.

Dransch, D. (2000). "The use of different media in visualizing spatial data." *Computers & Geosciences*, 26, 5-9.

Dryer, D. and Frhlich, J. (1994). "A GIS-Based Soil Erosion Model for Two Investigation Areas in the High Rhine Valley and the Swiss Jura Plateau (NW Switzerland)." *Procs. Fifth European Conference on Geographical Information Systems.*, 1032-1041.

Dykes, J. A. (1997). "Exploring spatial data representation with dynamic graphics." *Computers & Geosciences*, 23(4), 345-370.

Eagleson, P. S. (1970). *Dynamic hydrology*, McGraw-Hill.

Edsall, R. M., Harrower, M., and Mennis, J. L. (2000). "Tools for visualizing properties of spatial and temporal periodicity in geographic data." *Computers & Geosciences*, 26(1), 109-118.

Ehlschlaeger, C. R., Shortridge, A. M., and Goodchild, M. F. (1997). "Visualizing spatial data uncertainty using animation." *Computers & Geosciences*, 23(4), 387-395.

Einstein, H. A. (1950). "The bed-load function for sediment transportation in open channel flows ." U.S. Dept. of Agriculture, Washington.

Engelund, F. and Hansen, E. (1967). *A monograph on sediment transport in alluvial streams*, Teknisk Forlag, Copenhagen (Denmark).

ESRI (1982-2001). "ArcInfo" Environmental Systems Research Institute, Inc., Redlands, California.

ESRI (1994). *Cell-based modeling with GRID*, Environmental systems research institute, inc., Redlands, CA.

Evans, B. J. (1997). "Dynamic display of spatial data-reliability: does it benefit the map user?" *Computers & Geosciences*, 23(4), 409-422.

Flanagan, D. C. and Nearing, M. A. (1995). "USDA-Water Erosion Prediction Project: Hillslope Profile and Watershed Model Documentation ." *Report No. 10*, NSERL .

Foster, G. R. and Meyer, L. D. (1977). "Soil Erosion and Sedimentation by Water -- an Overview." *Procs. National Symposium on Soil Erosion and Sedimentation by Water*, Am. Soc. of Agric. Eng., St. Joseph, Michigan, 1-13.

Foster, G. R., Neibling, W. H., Davis, S. S., and Alberts, E. E. (1980). "Modeling the Particle Segregation During Deposition by Overland Flow." *Procs. Hydrologic Transport Modeling*, St. Joseph, Mich., 184-205.

Foster, G. R. and Wischmeier, W. H. (1974). "Evaluating irregular slopes for soil loss prediction." *Transactions of the ASAE*, 17(2), 305-309.

Francis J. R. D. (1973). "Experiments on the Motion of Solitary Grains Along the Bed of a Water Stream." *Procs. Royal Soc. London* , 443-471.

Fuhrmann, S. (2000). "Designing a visualization system for hydrological data." *Computers & Geosciences*, 26(1), 11-19.

Gao, J. (1997). "Resolution and accuracy of terrain representation by grid DEMs at a micro-scale." *Int. J. Geographical Information Science*, 11(2), 199-212.

Garbrecht, J. and Martz, L. W. (1997c). "The assignment of drainage direction over flat surfaces in raster digital elevation models." *J. Hydrology*, 193, 204-213.

Garbrecht, J. and Martz, L. W. (1999). "Digital Elevation Model Issues in Water Resources Modeling." *Procs. 1999 ESRI User Conference*.

Garbrecht, J. and Martz, L. W. (1994). "Grid Size Dependency of parameters extracted from digital elevation models." *Computers & Geosciences*, 20(1), 85-87.

Garbrecht, J. and Martz, L. W. (1997a). "TOPAZ v1.22: an automated digital landscape analysis tool for topographic evaluation, drainage identification, watershed segmentation and subcatchment parameterization: overview." *ARS Publication*, U. S. Department of Agriculture, Agricultural Research Service.

Garbrecht, J. and Martz, L. W. (1997b). "TOPAZ v1.22: an automated digital landscape analysis tool for topographic evaluation, drainage identification, watershed segmentation and subcatchment parameterization: user manual." *ARS Publication No. GRL 97-4*, U. S. Department of Agriculture, Agricultural Research Service.

Govers, G. (1990). "Empirical relationships for the transport capacity of overland flow." *Erosion, Transport and deposition processes*, D. E. Walling, A. Yair, and S. Berkowicz, eds., IAHS, 45-63.

Green, W. H. and Ampt, G. A. (1911). "Studies on soil physics - Part I. The flow of air and water through soils." *The Journal of agricultural Science*, 4, 1-24.

Gurnell, A. M. and Montgomery, D. R. (1998). "Preface: hydrological applications of GIS." *Hydrological processes*, 12, 821-822.

Haan, C. T., Barfield, B. J., and Hayes, J. C. (1994). *Design hydrology and sedimentology for small catchments*, Academic Press.

Haan, C. T., Johnson, H. P., and Brakensiek, D. L. (1982). *Hydrologic modeling of small watersheds.*

Hadley, R. F. (1986). *Drainage basin sediment delivery*, IAHS.

Hadley, R. F. and Mizuyama, T. (1993). *Sediment problems: strategies for monitoring, prediction, and control*, IAHS.

Hadley, R. F. and Ongley, E. D. (1989). *Sediment and the environment* , IAHS.

Harrison, J. S. and Doe, W. W. (1997). "Erosion modeling in pinon canyon maneuver site using the Universal Soil Loss Equation and the GIS system -- GRASS." Center for ecological management of military lands. Colorado State University, Fort Collins, CO.

Hay, L. and Knapp, L. (1996). "Integrating a GIS, a scientific visualization system and an orographic precipitation model." *Application of geographic information systems in hydrology and water resources management*, K. Kovar and H. P. Nachtnebel, eds., 123-131.

Holmes, K. W., Chadwick, O. A., and Kyriakidis, P. C. (2000). "Error in a USGS 30-meter digital elevation model and its impact on terrain modeling ." *J. Hydrology*, 233, 154-173.

Hong, S. and Mostaghimi, S. (1997). "Comparison of 1-D and 2-D Modeling of Overland Runoff and Sediment Transport." *Water Resources Bulletin*, 33(5), 1103-1116.

Johnson, B. E. (1997). "Development of a storm event based two-dimensional upland erosion model." PhD thesis, Colorado State University, Fort Collins, CO.

Johnson, B. E., Julien, P. Y., Molnar, D. K., and Watson, C. C. (2000). "The two-dimensional-upland erosion model CASC2D-SED." *J. of the AWRA*, 36(1), 31-42.

Julien, P. Y. (1995). *Erosion and Sedimentation*. Cambridge University Press, Cambridge, New York.

Julien, P. Y. and Frenette, M. (1987). "Macroscale analysis of upland erosion." *Hydrological sci. J. - Journal des sci. hydrol.*, 32(3), 347-358.

Julien, P. Y. and González del Tanago, M. (1991). "Spatially varied soil erosion under different climates." *Hydrological sci. J. - Journal des sci. hydrol.*, 36(6), 511-524.

Julien, P. Y. and Saghafian, B. (1991). "CASC2D users manual - A two dimensional watershed rainfall-runoff model." *Civil Eng. Report, CER90-91PYJ-BS-12*, Colorado State University, Fort Collins, Fort Collins, CO.

Julien, P. Y., Saghafian, B., and Ogden, F. L. (1995). "Raster-Based hydrologic modeling of spatially-varied surface runoff." *Water resources bulletin*, 31(3), 523-536.

Julien, P. Y. and Simons, D. B. (1984). "Analysis of sediment transport equations for rainfall erosion." *Civil Eng. report: CER83-84PYJ-DBS52*, Colorado State University, Fort Collins, CO.

Kaden, S. O. (1993). "GIS in Water-Related Environmental Planning and Management: Problems and Solutions." *Procs. Application of Geographic Information Systems in Hydrology and Water Resources Management*, IAHS-AISH Publication No. 211, 385-397.

Kienzle, S. W. (1996). "Using DTMs and GIS to define input variables for hydrological and geomorphological analysis." *Application of geographic information systems in hydrology and water resources management*, K. Kovar and H. P. Nachtnebel, eds., 183-190.

Kilinc, M. Y. (1972). "Mechanics of soil erosion from overland flow generated by simulated rainfall." PhD thesis, Colorado State University, Fort Collins, CO.

Kilinc, M. Y. and Richardson, E. V. (1973). "Mechanics of soil erosion from overland flow generated by simulated rainfall." *Hydrology Papers No. 63*, Colorado State University, Fort Collins, CO.

Kimberlin, L. W. and Moldenhauer, W. C. (1977). "Predicting Soil Erosion." *Procs. National Symposium on Soil Erosion and Sedimentation by Water*, American Society of Civil Engineers, St. Joseph, Michigan, 31-42.

Kinnell, P. I. A. (2000). "AGNPS-M: applying the USLE-M within the agricultural non-point source pollution model." *Environmental modelling & software*, 15, 331-341.

Knisel, W. G. (1995). "CREAMS: A field-scale model for Chemicals, Runoff, and Erosion from Agricultural Management Systems." *Report No. 26*, U.S. Dept. of Agriculture. Agric. Research Service.

Kraak, M.-J. and Ormeling, F. (1996). *Cartography : visualization of spatial data* , Longman, UK.

Kreuseler, M. (2000). "Visualization of geographically related multidimensional data in virtual 3D scenes." *Computers & Geosciences*, 26(1), 101-108.

Kuhnle, R. A., Bingner, R. L., Foster, G. R., and Grissinger, E. H. (1996). "Effect of land use changes on sediment transport in Goodwin Creek." *Water Resources Research*, 32(10), 3189-3196.

Kuhnle, R. A. and Willis, J. C. (1998). "Statistics of Sediment Transport in Goodwin Creek." *Journal of Hydraulic Engineering* , 124(11), 1109-1114.

Kuo, W.-L., Steenhuis, T. S., McCulloch, C. E., Mohler, C. L., Weinstein, D. A., DeGloria, S. D., and Swaney, D. P. (1999). "Effects of grid size on runoff and soil moisture for a variable-source-area hydrology model." *Water Resources Research*, 35(11), 3419-3428.

Lagacherie, P., Moussa, R., Cormary, D., and Molenat, J. (1996). "Effects of DEM data source and sampling pattern on topographical parameters and on a topography-based hydrological model." *Application of geographic information systems in hydrology and water resources management*, K. Kovar and H. P. Nachtnebel, eds., 191-199.

Lane, L. J. and others (1992). "Development and application of modern soil erosion prediction technology -- The USDA experience." *Australian journal of soil research*, 30, 893-912.

Lane, L. J., Hakonson, T., and Foster, G. R. (1983). "Watershed Erosion and Sediment Yield Affecting Contaminant Transport." *Procs. Symposium on Environmental Research on Actinide Elements*, 193-223.

Lane, L. J., Hernández, M., and Nichols, M. (1997). "Processes controlling sediment yield from watersheds as functions of spatial scale." *Environmental modelling & software*, 12(4), 355-369.

Lane, L. J., Nichols, M. H., and Paige, G. B. (1995). "Modeling Erosion on Hillslopes: Concepts, Theory, and Data." *Procs. International Congress on Modelling and Simulation (MODSIM'95)*, 1-7.

Lane, L. J., Shirley, E. D., and Singh, V. P. (1988). "Modelling erosion on hillslopes." *Modelling geomorphological systems*, M.G. Anderson, ed., John Wiley & Sons, 287-308.

Lane, S. N. and Richards, K. S. (2001). "The 'Validation' of hydrodynamic models: some critical perspectives." *Model validation : perspectives in hydrological science* , M. G. Anderson and P. D. Bates, eds., J. Wiley, Chichester ; New York, 413-438.

Lee, J. and Chu, C.-J. (1996). "Spatial structures of digital terrain models and hydrological feature extraction." *Application of geographic information systems in hydrology and water resources management*, K. Kovar and H. P. Nachtnebel, eds., 201-206.

Li, R. M., Shen, H. W., and Simons, D. B. (1973). "Mechanics of Soil Erosion by Overland Flow." *Procs. 15th Congress IAHR*, Proc. Paper A-54.

Li, R. M., Stevens, M. A., and Simons, D. B. (1976). "Solutions to Green-Ampt infiltration equations." *J. Irrigation and Drainage*, 102(No. IR2), 239-248.

Lin, C. J. (1991). "An application of geographic information systems to soil erosion hazards investigation on forest land." PhD thesis, State University of New York, Buffalo, NY.

MacEachren, A. M. and Kraak, M.-J. (1997). "Exploratory cartographic visualization: advancing the agenda." *Computers & Geosciences*, 23(4), 335-343.

MacEachren, A. M. and Kraak, M.-J. (2001). "Research Challenges in Geovisualization." *Cartography and Geographic Information Science*, 28(1).

Madej, M. A. (1981). "Impacts and management of steepland erosion: a review." *J. Hydrology (NZ)*, 20(1), 108-111.

Mallants, D. and Feyen, J. (1989). "Hydrological systems modeling of the rainfall-runoff conversion using a digital terrain model." *Landscape*, 4, 277-290.

Marston, R. A. and Dolanb, L. S. (1999). "Effectiveness of sediment control structures relative to spatial patterns of upland soil loss in an arid watershed, Wyoming." *Geomorphology*, 31(1-4), 313-323.

Martz, L. W. and de Jong, E. (1987). "Using cesium-137 to assess the variability of net soil erosion and its association with topography in a canadian prairie landscape." *Catena*, 14, 439-451.

Martz, L. W. and Garbrecht, J. (1998). "The treatment of flat areas and depressions in automated drainage analysis of raster digital elevation models." *Hydrological processes*, 12(6), 843-855.

Meyer, L. D., Rhoton, F. E., and Renard, K. G. (1984). "Soil Erosion - Soil Productivity. A Look to the Future." *Procs. National Symposium on Soil Erosion and Soil Productivity*, St. Joseph, Michigan, 266-277.

Meyer, L. D. and Wischmeier, W. H. (1969). "Mathematical simulation of the process of soil erosion by water." *Transactions of the ASAE*, 12, 754-758.

Mitas, L., Brown, W. M., and Mitasova, H. (1997). "Role of dynamic cartography in simulations of landscape processes based on multivariate fields." *Computers & Geosciences*, 23(4), 347-446.

Mitasova, H., Jofierka, J., Zlocha, M., and Iverson, L. R. (1996). "Modelling topographic potential for erosion and deposition using GIS." *Int. J. geographical information systems*, 10(5), 629-641.

Mitasova, H., Mitas, L., Brown, W. M., and Johnston, D. M. (2000). "DEM resolution, topographic analysis and erosion modeling (Incomplete draft)." <http://www2.gis.uiuc.edu:2280/modviz/irwin/irwin1.html> (Aug. 2000).

Mitchell, J. K., Engel, B. A., Srinivasan, R., and Wang, S. S. Y. (1993). "Validation of AGNPS for small watersheds using an integrated AGNPS/GIS system." *Geographic Information Systems and Water Resources*, 89-100.

Molnar, D. K. (1997). "Grid size selection for 2-D hydrologic modeling of large watersheds." PhD thesis, Colorado State University, Fort Collins, CO.

Molnar, D. K. and Julien, P. Y. (1998). "Estimation of Upland erosion using GIS." *Computers & Geosciences*, 24(2), 183-192.

Moore, I. D. and Wilson, J. P. (1992). "Length-slope factors for the revised universal soil loss equation: simplified method of estimation." *J. Soil and Water Conservation*, 47(5), 423-428.

Morgan, R. P. C. (1980). "Implications." *Soil Erosion*, M. J. Kirkby and R. P. C.

Morgan, eds., John Wiley & Sons, 253-301.

Morgan, R. P. C., Quinton, J. N., and Rickson, R. J. (1990). "EUROSEM: A user guide, version 2." , Silsoe College, Cranfield University, United Kingdom.

Morris, K., Hill, D., and Moore, A. (2000). "Mapping the environment through three-dimensional space and time." *Computers, environment and urban systems*, 24(5), 435-450.

Musgrave, G. W. (1957). "The qualitative evaluation of factors in water erosion: a first approximation." *J. Soil and Water Conservation*, 2(3), 1331-1338.

Nearing, M. A., Lane, L. J., Alberts, E. E., and Laflen, J. M. (1990). "Prediction technology for soil erosion by water: status and research needs." *Soil Sci. Soc. Amer. J.*, 54, 1702-1711.

Norton, D. (1986). "Erosion-sedimentation in a closed drainage basin in northwest Indiana." *Soil Sci. Soc. Amer. J.*, 50(1), 209-213.

Ogden, F. L. (1998). "A brief description of the Hydrologic Model CASC2D." <http://www.engr.uconn.edu/~ogden/casc2d/casc2_desc.htm> (Aug. 1999).

Ogden, F. L. (1997a). "CASC2D reference manual. Version 1.17." University of Connecticut, Storrs, CT.

Ogden, F. L. (1994). "De-St Venant Channel Routing in Distributed Hydrologic Modeling." *Procs. Hydraulic Engineering '94, ASCE Hydraulics Specialty Conference*, 492-496.

Ogden, F. L. (1997b). "Primer: Using WMS for CASC2D Data Development." Brinham Young University, Provo, UT.

Painter, D. J. (1981). "Steeplands erosion processes and their assessment." *Erosion and sediment transport in pacific rim steeplands*, T. R. H. Davies and A. Pierce, eds., IAHS, Washington, D.C. 2-20.

Peterson, A. E. and Swan, J. B. (1979). "Universal Soil Loss Equation: Past, present and future." *SSSA Special Publication No. 8*, Soil Science Society of America.

Peterson, M. P. (1995). *Interactive and animated cartography*, Prentice Hall, Englewood Cliffs, NJ.

Philip, J. R. (1983). "Infiltration in One, Two, and Three Dimensions." *Procs. National Conference on Advances in Infiltration*, American Society of Agricultural Engineers, St. Joseph, Mich., 1-13.

Pimentel, D. and others (1976). "Land degradation: Effects on food and energy resources." *Science*, 194, 149-155.

Pimentel, D., Harvey, C., Resosudarmo, P., and Blair, R. (1995). "Environmental and economic loss of soil erosion and conservation benefits." *Science*, 276, 1117-1123.

Predmore, S. K. (1999). "Animating Time Series Data Using ArcView 3D Analyst." *Procs. 1999 ESRI User Conference*.

Pullar, D. and Springer, D. (2000). "Towards integrating GIS and catchment models." *Environmental modelling & software*, 15, 451-459.

Quinn, P., Beven, K., Chevallier, P., and Planchon, O. (1991). "The prediction of hillslope flow paths for distributed hydrological modeling using digital terrain models." *Hydrological processes*, 5, 59-79.

Rawls, W. J., Brakensiek, D. L., and Miller, N. (1983). "Green-Ampt infiltration parameters from soils data." *Journal of Hydraulic Engineering* , 109(1), 62-69.

Renard, K. G. and others (1997). "Predicting soil erosion by water: A guide to conservation planning with the Revised Universal Soil Loss Equation (RUSLE)." *USDA handbook No. 703*, U.S. Department of Agriculture.

Renard, K. G., Lane, L. J., Foster, G. R., and Laflen, J. M. (1996). "Soil loss estimation." *Soil erosion, conservation, and rehabilitation*, M. Agassi, ed., Marcel Dekker, Inc., New York, 169-202.

Richards, J. (1993). *Remote sensing digital image analysis: an introduction*, Springer-Verlag.

Ritter, P. (1987). "A vector-based slope and aspect generation algorithm." *Photogram. eng. and remote sensing* , 53, 1109-1111.

Rojas, R. and Woolhiser, D. A. (2000). "Erosion Parameters Identifiability in the KINEROS Model." *Procs. the Debris Flow Disaster of December 1999 in Venezuela*, Dynamics of Debris Flow section (CD-ROM).

Rudra, R. P., Dickinson, W. T., and Wall, G. J. (1998). "Problems regarding the use of soil erosion models." *Modelling soil erosion by water*, J. Boardman and D. T. Favis-Mortlock, eds., Springer-Verlag, Berlin Heidelberg, 175-189.

Saghafian, B. (1992). "Hydrologic analysis of watershed response to spatially varied infiltration." PhD thesis, Colorado State University, Fort Collins , CO.

Saghafian, B. (1993). "Implementation of a Distributed Hydrologic Model Within Geographic Resources Analysis Support System (GRASS)." *Procs. Second International Conference on Integrating Environmental Models and GIS*.

Saghafian, B. and Ogden, F. L. (1996). "r.hydro.CASC2D." <http://www.baylor.edu/grass/gdp/html_grass5/html/r.hydro.CASC2D.html> (July 2001).

Saulnier, G.-M., Obled, C., and Beven, K. (1997). "Analytical compensation between DTM grid resolution and effective values of saturated hydraulic conductivity within the TOPMODEL framework." *Hydrological processes*, 11(9), 1331-1346.

Saxton, K. E., Rawls, W. J., Romberger, J. S., and Papendick, R. I. (1986). "Estimating generalized soil-water characteristics from texture." *Soil Sci. Soc. Amer. J.*, 50(4), 1031-1036.

Schmidt, J., Hennrich, K., and Dikau, R. (2000). "Scales and similarities in runoff processes with respect to geomorphometry." *Hydrological processes*, 14(11-12), 1963-1979.

Sharma, K. D., Menenti, M., and Huygen, A. V. (1996). "Modeling spatial sediment delivery in an arid region using thematic mapper data and GIS." *Transactions of the ASAE*, 39(2), 551-557.

Sharpley, A. N. and Williams, J. R. (1990). "EPIC - Erosion/Productivity Impact Calculator: Model documentation." *Technical bulletin No. 1768*, USDA-ARS.

Sharpnack, D. A. and Akin, G. (1969). "An Algorithm for Computing Slope and Aspect from Elevations." *Photogram. Eng.*, 35(3), 247-248.

Shields, F. D., Knight, S. S., and Cooper, C. M. (1995). "Rehabilitation of watersheds with incising channels." *Water resources bulletin*, 31(6), 971-981.

Singh, V. P. (1995a). *Computer models of watershed hydrology*, Water Resources Publications, Colorado.

Singh, V. P. (1995b). "Watershed modeling." *Computer models of watershed hydrology*, V. P. Singh, ed., Water Resources Publications, Colorado, 1-22.

Smith, R. E., Goodrich, D. C., Woolhiser, D. A., and Unkrich, C. L. (1995). "KINEROS - A kinematic runoff and erosion model." *Computer models of watershed hydrology*, V. P. Singh, ed., Water Resources Publications, Colorado, 697-732.

Spah, J. A. (2000). "Rainfall runoff and the effects of initial soil moisture associated with the Little Washita River Watershed, Oklahoma ." MSc thesis, Colorado State University, Fort Collins, CO.

Srivastava, A. (2000). "Comparison of two algorithms for removing depressions and delineating flow networks from grid digital elevation models." MSc thesis, Virginia Tech, Blacksburg, VA.

Steiner, E. B., MacEachren, A. M., and Guo, D. (2001). "Developing and assessing light-weight data-driven exploratory geovisualization tools for the web." <http://www.geovista.psu.edu/publications/Beijing01/SteinerICA01/flash_db2.htm> (July 2001).

Stone, M. (2000). *The role of erosion and sediment transport in nutrient and contaminant transfer* , Wallingford, Oxfordshire, UK .

Summer, W., Klaghofer, E., and Zhang, W. (1998). *Modelling soil erosion, sediment transport and closely related hydrological process*. IAHS.

Sutherland, R. A. (1991). "Caesium-137 and sediment budgeting within a partially closed drainage basin." *Zeitschrift für Geomorphologie. Annals of geomorphology*, 35(1), 47-63.

Tarboton, D. G., Bras, R. L., and Rodriguez-Iturbe, I. (1989). "Scaling and elevation in river networks." *Water Resources Research*, 25(9), 2037-2051.

Thieken, A. H., Lücke, A., Diekkrüger, B., and Richter, O. (1999). "Scaling input data by GIS for hydrological modelling." *Hydrological processes*, 13(4), 611-630.

Thurston, J. (2001). "Geo-Visualization: Current Issues / Future Potentials ." <http://www.giscafe.com/GISVision/Review/GeoVisualization.html> (July 2001).

Tim, U. S. (1996). "Emerging technologies for hydrologic and water quality modeling research." *Transactions of the ASAE*, 39(2), 465-476.

Uhlenküken, C., Schmidt, B., and Streit, U. (2000). "Visual exploration of high-dimensional spatial data: requirements and deficits." *Computers & Geosciences*, 26(1), 77-85.

Vesseth, R. (1990). "STEEP - Radioactive Fallout Provides Estimator for Soil Erosion." *Direct seed tillage handbook*, Washington State University. http://pnwsteep.wsu.edu/tillagehandbook/chapter1/011190.htm.

Vieux, B. E. (1993). "Aggregation and smoothing effects on surface runoff modeling." *ASCE - Journal of Computing in Civil Engineering*, 7(3), 310-338.

Wainwright, J. and Parsons, A. J. (1998). "Sensitivity of sediment-transport equations to errors in hydraulic models of overland flow." *Modelling soil erosion by water*, J. Boardman and D. T. Favis-Mortlock, eds., Springer-Verlag, Berlin Heidelberg, 271-284.

Walling, D. E. (1982). *Recent developments in the explanation and prediction of erosion and sediment yield*, IAHS, Exeter, UK.

Walling, D. E. and Probst, J.-L. (1997). *Human impact on erosion and sedimentation*, IAHS, Wallingford, Oxfordshire, UK .

Walling, D. E. and Webb, B. E. (1996). *Erosion and sediment yield : global and regional perspectives* , IAHS, Wallingford, Oxfordshire, UK .

Wang, M., Hjelmfelt, A. T., and Garbrecht, J. (2000). "DEM aggregation for watershed modeling." *J. of the AWRA*, 36(3), 579-584.

Warren, S. D., Mitasova, H., Gebhart, D. L., Hohmann, M. G., Landsberger, S., and Iskander, F. Y. (2000). "Validation of enhancements to the Universal Soil Loss Equation topographic factor using 137Cesium." Center for ecological management of military lands. Colorado State University.

Wicks, J. M. and Bathurst, J. C. (1996). "SHESED: a physically based, distributed erosion and sediment yield component for the SHE hydrological modelling system." *Journal of hydrology*, 175(1-4), 213-238.

Wijesekera, S. and Samarakoon, L. (2001). "Extraction of Parameters and Modelling Soil Erosion Using GIS in a Grid Environment." *Procs. 22nd Asian Conference on Remote Sensing*, Singapore.

Williams, J. R. (1975). "Sediment yield prediction with universal equation using runoff energy factor." *Agricultural Research Service report ARS-S-40*, U.S. Department of Agriculture.

(1978). "Predicting rainfall erosion losses -- A guide to conservation planning." *USDA handbook No. 537*, U.S. Department of Agriculture in cooperation with Purdue Agricultural Experiment Station.

Wolock, D. M. and McCabe, G. J. (2000). "Differences in topographic characteristics computed from 100- and 1000-m resolution digital elevation model data." *Hydrological processes*, 14(6), 987-1002.

Wolock, D. M. and Price, C. V. (1994). "Effects of digital elevation model map scale and data resolution on a topography-based watershed model." *Water resources research*, 30(11), 3041-3052.

Woolhiser, D. A. (1996). "Search for physically based runoff model - A hydrologic El Dorado?" *J. Hydraul. Eng.*, 122(3), 122-129.

Woolhiser, D. A. (1975). "Simulation of unsteady overland flow." *Unsteady flow in open channels*, K. Mahmood and V. Yevjevich, eds., Fort Collins, CO.

Woolhiser, D. A., Smith, R. E., Sharif, H. O., and Goodrich, D. C. (1990). "KINEROS, a kinematic runoff and erosion model: documentation and user manual." *ARS-77*, U.S. Dept. of Agriculture, Agricultural Research Service.

Young, R. A., Onstad, C. A., Bosch, D. D., and Anderson, W. P. (1989). "AGNPS: A nonpoint source pollution model for evaluating agricultural watersheds." *Journal of soil and water conservation*, 44, 168-173.

Yu, Z. (1997). "Grid-spacing effect on watershed hydrologic simulations." *Hydrological science and technology*, 13(1-4), 75-85.

Zhang, W. and Montgomery, D. R. (1994). "Digital elevation model grid size, landscape representation, and hydrologic simulations." *Water resources research*, 30(4), 1019-1028.

APPENDIX I

I.1 CASC2D-SED Sub-routines List ---------- *I-2*

I.2 CASC2D-SED Code ---------- *I-4*

I.3 CASC2D-SED Input/Output Files ---------- *I-82*

I.3.1 Input Files ---------- I-82

I.3.2 Output Files---------- I-83

I.4 Time-Series Grids Display ---------- *I-86*

I.4.1 AML programs---------- I-86

I.4.2 DS.aml parameter file example ---------- I-106

I.1 CASC2D-SED SUB-ROUTINES LIST

PROGRAM	CALLS	LOCATION
main()	ReadProjectFile(argv[])	ReadProjectFile.c
	ReadControlFile	ReadControlFile.c
	ReadGridHeaders()	ReadGrids.c
	GridsMemAlloc()	MemAlloc.c
	ReadInputGrids()	ReadGrids.c
	InitializeVars()	InitializeVars.c
	ReadChannlFile	ReadChannFile.c
	ChannToplg()	ChannToplg.c
	CompInitialVol()	CompVolumes.c
	OvrlDepth()	OvrlDepth.c
	ChannDepth()	ChannDepth.c
	OvrlRout()	OvrlRout.c
	ChannRout()	ChannRout.c
	RoutOutlet()	RoutOutlet.c
	Settling()	Settling.c
	RoutSedOut()	RoutSedOut.c
	SedStats()	SedStats.c
	WriteOutflow()	WriteOutflow.c
	WriteGrids()	WriteGrids.c
	Reset()	Reset.c
	CompFinalVol()	CompVolumes.c
	WriteSummFlow()	WriteSummFlow.c
	SedVolumes()	SedVolumes.c
	WriteSummSed()	WriteSummSed.c
	GridsMemFree()	MemFree.c
	RunTime()	RunTime.c
ReadControlFile()	floatMemAlloc2d()	MemAlloc.c
	intMemAlloc2d()	MemAlloc.c
GridsMemAlloc()	intMemAlloc2d()	MemAlloc.c
	floatMemAlloc2d()	MemAlloc.c
OvrlDepth()	intercept()	intercept.c
	rain()	rain.c
	infilt()	infilt.c
rain()	intercept()	intercept.c

ChannDepth()	NewChannDepth()	Fcts_Chann.c
OvrlRout()	ovrl	OvrlRout.c
ovrl()	RoutSedOvrl()	RoutSedOvrl.c
RoutSedOvrl()	TransferSed()	Fcts_sed.c
ChannRout()	chnchn	ChannRout.c
chnchn()	chnDischarge() RoutSedChn()	Fcts_Chann.c RoutSedChn.c
RoutSedChn()	TransferSed()	Fcts_sed.c
RoutOutlet()	chnDischarge()	Fcts_Chann.c
SedStats()	RoutSedChn() RoutSedOvrl()	RoutSedChn.c RoutSedOvrl.c
WriteGrids()	write2dTS() write3dTS()	WriteGrids.c WriteGrids.c
GridsMemFree()	iMemFree2d() fMemFree2d()	MemFree.c MemFree.c

I.2 CASC2D-SED CODE

File	Page
All.h	I-5
CASC2D-SED_sed.c	I-9
ReadProjectFile.c	I-12
ReadControlFile.c	I-16
ReadGrids.c	I-19
MemAlloc.c	I-22
InitializeVars.c	I-24
ChannToplg.c	I-28
CompVolumes.c	I-32
OvrlDepth.c	I-34
infilt.c	I-37
intercept.c	I-38
rain.c	I-39
ChannDepth.c	I-40
OvrlRout.c	I-42
RoutSedOvrl.c	I-45
ChannRout.c	I-47
RoutSedChn.c	I-50
RoutOutlet.c	I-52
Settling.c	I-54
RoutSedOut.c	I-55
SedStats.c	I-56
WriteOutflows.c	I-58
WriteGrids.c	I-60
Reset.c	I-64
WriteSummFlow.c	I-65
SedVolumes.c	I-67
WriteSummSed.c	I-68
MemFree.c	I-74
RunTime.c	I-75
Fcts_chann.c	I-76
Fcts_misc.c	I-78
Fcts_sed.c	I-79
Fcts_stats.c	I-80

```
                    /**************************/
                    /*      INCLUDE FILES     */
                    /**************************/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include <float.h>
#include <time.h>
#include <malloc.h>

                  /**********************************/
                  /*      FUNCTION PROTOTYPES       */
                  /**********************************/

extern void ReadProjectFile(char ProjectName[80]);

extern void ReadGridHeaders();

extern void GridsMemAlloc();

int** intMemAlloc2d(int NoRows, int NoCols);

float** floatMemAlloc2d(int NoRows, int NoCols);

float** floatMemAlloc3d(int layer, int NoRows, int NoCols);

extern void ReadControlFile();

extern void ReadChannFile();

extern void ReadInputGrids();

extern void InitializeVars();

extern void ChannToplg();

extern void CompInitialVol();

extern void OvrlDepth( );

extern void intercept(int j, int k);

extern void rain(int j,int k);

extern void infilt(int j, int k);

extern void Settling();

extern void OvrlRout();

extern void ovrl( int j,int k, int jj, int kk);

extern void RoutSedOvrl(float dqq, float sf,int j,int k,
                                                  int jj,int kk);

extern void ChannDepth();

extern float newChnDepth(float wch,float dch,float sfactor,
                         int j, int k,float addedVolume);

extern void ChannRout();
```

```
extern void chnchn(int ic,int l);

float chnDischarge(float hchan,float hh,float wch,float dch,
        float stordep,float rmanch,float a,float sf,float sfactor);

extern void RoutSedChn(float dq,float sf,int jfrom,int kfrom,
                       int jto,int kto,float wch,float hchan);

extern float KRcap(float KRcoeff,float discharge,float width,
                   float sf, int jfrom, int kfrom);

extern float EHcap(int SizeFr,float width,float depth,
                   float sf,float discharge);

extern void TransferSed(int SizeFr,float **Source[],
                        float VolToTransfer,
                        int jfrom,int kfrom,int jto,int kto);

extern void findMFAC(int jfrom,int kfrom,float totQSout,
                     float discharge,float qs[4]);

extern void RoutOutlet();

extern void RoutSedOut();

extern void WriteOutflow();

extern void SedStats();

extern void WriteGrids();

extern void write2dTS(char *name, char *write_name,FILE *write_name_fptr
                float **ParamValue, float UnitsConv);

extern void write3dTS(char *name, char *write_name,FILE *write_name_fptr
                int layer, float **ParamValue[]);

extern void Reset();

extern void CompFinalVol();

extern void SedVolumes();

extern void WriteSummFlow();

extern void WriteSummSed();

extern void GridsMemFree();

void fMemFree2d(float** matrix2d, int NoRows, int NoCols);

void iMemFree2d(int** matrix2d, int NoRows, int NoCols);

extern void RunTime(clock_t finish);

extern float MAX(float one,float theOther);

extern float MIN(float one,float theOther);

extern struct gstats gridstats(float **inputgrid);
```

```
/**********************************/
/*        GLOBAL CONSTANTS        */
/**********************************/

#define g 9.81       /* Gravitational acceleration [m2/s]           */
#define ROs 2.65     /* Mass density of solid particles [tons/m3]   */
#define KRov 58390   /* Kilinc Constant in the overland transp.eq.  */
#define KRch 58390   /* Kilinc Constant in the channel transp.eq.   */
#define G 2.65       /* Specific gravity [adimensional]             */


/**********************************/
/*        GLOBAL VARIABLES        */
/**********************************/

extern char
      project_file[80],control_file[80],  rain_file[80],chn_file[80],
      header[300],elev_file[80],shap_file[80],soil_file[80],
      iman_file[80],sdep_file[80],node_file[80],link_file[80],
      Summ_file[80],dis_out_file[80],sed_out_file[80],
      rname[80],write_rname[80],vinfname[80],write_vinfname[80],
      dname[80],cname[80],susname[80],erosdepname[80],
      sandfluxname[80],siltfluxname[80],clayfluxname[80],
      totalfluxname[80],sandMFACname[80],siltMFACname[80],
      clayMFACname[80],totalMFACname[80];

extern int
      m,n,jout,kout,ncells,nman,nsoil,
      indexsdep,indexinf,indexeros,chancheck,
      nchan_link,nchan_node[100], maxlink,ichn[50][100][3],
      iter,niter,nitrn,nprn,nplt,irain,nrg,nread,
      ifcount,icount,ipcount,
      indexdis,ndis,**iq,unitsQ,indexsed,nsed,**ised,unitsQs,
      nodatavalue, **link,**node,**isoil, **iman, **ishp;

extern float
      w,dt,sout,wchout,dchout,sfactorout,rmanout,sovout,
      raincount,crain,*xrg,*yrg,*areaQ, *areaQs,
      xllcorner, yllcorner,*pman, *cfactor, *pfactor, *retention,
      elconv,**e,**ret,**sdep,**vinf,**pinf,vin,vout,vinftot,
      init_ov_vol,init_ch_vol,final_ch_vol,final_ov_vol,
      chp[50][100][7],**h,**hch,*rrg,**rint,**rtot,
      **dqov,**dqch,qout,qoutov,qpeak,tpeak,*q,
      *qsed,**qss, **qovs[4],**vols[4],**ssoil[4],**SedFluxOut[4],
      **MFAC[4],**MaxFluxCout,**totnetv,**totsus,**SusC,
      tot_eroded[4],sus_ov[4],dep_ov[4],tot_ov[4],
      sus_ch[4],dep_ch[4],tot_ch[4],sed_out[4];

extern double
      amaxdepth,amindepth,amaxcdepth,amincdepth, amaxvinf,aminvinf,
      amaxnetv,aminnetv,amaxFluxCoutOv,amaxFluxCoutCh,amaxrain,
      aminrain,amaxSusCh,amaxSusOv, amaxSusCch, amaxSusCov,
      ds[4],ws[4],elapsedTime;

struct gstats {
    double min;
    double max;
    float mean;
    float stdev;};

clock_t startTime;
```

```
extern FILE *project_file_fptr;
extern FILE *shap_file_fptr;
extern FILE *sdep_file_fptr;
extern FILE *elev_file_fptr;
extern FILE *rain_file_fptr;
extern FILE *soil_file_fptr;
extern FILE *control_file_fptr;
extern FILE *chn_file_fptr;
extern FILE *node_file_fptr;
extern FILE *link_file_fptr;
extern FILE *Summ_file_fptr;
extern FILE *iman_file_fptr;
extern FILE *dis_out_file_fptr;
extern FILE *sed_out_file_fptr;
extern FILE *debug;
```

```
#include "all.h"

      /**********************************/
      /*  GLOBAL VARIABLES DECLARATION  */
      /**********************************/

char  project_file[80],control_file[80],  rain_file[80],chn_file[80],
      header[300],elev_file[80],shap_file[80],soil_file[80],
      iman_file[80],sdep_file[80],node_file[80],link_file[80],
      Summ_file[80],dis_out_file[80],sed_out_file[80],
      rname[80],write_rname[80],vinfname[80],write_vinfname[80],
      dname[80],cname[80],susname[80],erosdepname[80],
      sandfluxname[80],siltfluxname[80],clayfluxname[80],
      totalfluxname[80],sandMFACname[80],siltMFACname[80],
      clayMFACname[80],totalMFACname[80];


int   m,n,jout,kout,ncells,nman,nsoil,
      indexsdep,indexinf,indexeros,chancheck,
      nchan_link,nchan_node[100], maxlink,ichn[50][100][3],
      iter,niter,nitrn,nprn,nplt,irain,nrg,nread,
      ifcount,icount,ipcount,
      indexdis,ndis,**iq,unitsQ,indexsed,nsed,**ised,unitsQs,
      nodatavalue, **link,**node,**isoil, **iman, **ishp;

float w,dt,sout,wchout,dchout,sfactorout,rmanout,sovout,
      raincount,crain,*xrg,*yrg,*areaQ, *areaQs,
      xllcorner, yllcorner,*pman, *cfactor, *pfactor, *retention,
      elconv,**e,**ret,**sdep,**vinf,**pinf,vin,vout,vinftot,
      init_ov_vol,init_ch_vol,final_ch_vol,final_ov_vol,
      chp[50][100][7],**h,**hch,*rrg,**rint,**rtot,
      **dqov,**dqch,qout,qoutov,qpeak,tpeak,*q,
      *qsed,**qss, **qovs[4],**vols[4],**ssoil[4],**SedFluxOut[4],
      **MFAC[4],**MaxFluxCout,**totnetv,**totsus, **SusC,
      tot_eroded[4],sus_ov[4],dep_ov[4],tot_ov[4],
      sus_ch[4],dep_ch[4],tot_ch[4],sed_out[4];

double amaxdepth,amindepth,amaxcdepth,amincdepth, amaxvinf,aminvinf,
      amaxnetv,aminnetv,amaxFluxCoutOv,amaxFluxCoutCh,amaxrain,
      aminrain,amaxSusCh,amaxSusOv, amaxSusCch, amaxSusCov,
      ds[4],ws[4],elapsedTime;

clock_t startTime;

FILE *project_file_fptr = NULL;
FILE *shap_file_fptr = NULL;
FILE *sdep_file_fptr = NULL;
FILE *elev_file_fptr = NULL;
FILE *rain_file_fptr = NULL;
FILE *soil_file_fptr = NULL;
FILE *control_file_fptr = NULL;
FILE *chn_file_fptr = NULL;
FILE *node_file_fptr = NULL;
FILE *link_file_fptr = NULL;
FILE *Summ_file_fptr = NULL;
FILE *iman_file_fptr = NULL;
FILE *dis_out_file_fptr = NULL;
FILE *sed_out_file_fptr = NULL;
FILE *debug = NULL;
```

```
       /******************************/
       /*   CASC2D_SED MAIN PROGRAM   */
       /******************************/

int main(argc,argv)
  int argc;
  char *argv[];
  {
      /*******************************************/
      /*          Read and initialize data       */
      /*******************************************/

    /* Time CASC2D running time                                        */
    startTime = clock();

    /* Reads the project files names                                   */
    ReadProjectFile(argv[1]);

    /* Reading in the CONTROL Information                              */
    ReadControlFile();

    /* Reads the input grids headers                                   */
    ReadGridHeaders();

    /* Dynamically allocates memory to the input & output grids        */
    GridsMemAlloc();

    /* Reading Input Grids                                             */
    ReadInputGrids();

    /* Initializing Variables                                          */
    InitializeVars();

    /* Determines channel network topology and variables               */
    if(chancheck == 1)
    {
      ReadChannFile();
      ChannToplg();
    }

    /* Computing Initial Storage Volume for Overland and Channels */
    if(indexsdep == 1) CompInitialVol();
```

```
    /*********************************************/
    /*            Event time loop               */
    /*********************************************/

  for(iter=1;iter<=niter;iter++)
  {

    /* Updating overland and channels water depth               */
    OvrlDepth( );
    if(chancheck == 1)ChannDepth();

    /* Overland and channel flow routing                        */
    OvrlRout();
    if(chancheck == 1) ChannRout();

    /* Flow routing at the outlet                               */
    RoutOutlet();

    /* Sediment settling for this water depth and time dt       */
    if (indexeros == 1) Settling();

    /* Sediment routing at the outlet, total erosion/deposition */
    /* volumes, sediment max./min values and paticle settling   */
    if (indexeros == 1)
    {
      if(qout != 0) RoutSedOut();
      SedStats();
    }

    /* Write outflow to screen and files                        */
    WriteOutflow();

    /* Write output grids                                       */
    WriteGrids();

    /* Reset grids to zero                                      */
    Reset();

  } /* End of Time Loop */

      /*********************************************/
      /* Computes final volumes & writes results   */
      /*********************************************/

  /* Compute final water volumes on the overland and channels   */
  CompFinalVol();

  /* Writing summary file                                       */
  WriteSummFlow();

  /* Compute final sed. volumes on the overland and channels    */
  if(indexeros == 1)
  {
    SedVolumes();
    WriteSummSed();
  }

  /* Deallocate memory                                          */
  GridsMemFree();

  /* Computes casc2d running time                               */
  RunTime(clock());

} /* End of Main Program                                        */
```

```
/********************************/
/*     ReadProjectFile.c        */
/********************************/

#include "all.h"

extern void ReadProjectFile(char ProjectName[80])
{

  char  file_id[80];
  int ch;

  /* Initialize the output grid names to the null element          */
  dname[0] = '\0';
  rname[0] = '\0';
  vinfname[0] = '\0';
  cname[0] = '\0';
  susname[0] = '\0';
  erosdepname[0] = '\0';
  sandfluxname[0] = '\0';
  siltfluxname[0] = '\0';
  clayfluxname[0] = '\0';
  totalfluxname[0] = '\0';
  sandMFACname[0] = '\0';
  siltMFACname[0] = '\0';
  clayMFACname[0] = '\0';
  totalMFACname[0] = '\0';
  sed_out_file[0] = '\0';

  strncpy(project_file,ProjectName,80);

  printf("Project File            %s \n",project_file);

  if((project_file_fptr=fopen(project_file,"r"))==NULL)
  {printf("Can't open Project File : %s \n",project_file);
   exit(EXIT_FAILURE);}

  /* Reading in the Data Filenames and Opening Files              */

  printf("Starting to read the Project File \n");

  do{

  ch=fscanf(project_file_fptr,"%s",file_id);

  if(ch!=EOF)
  {
    printf("FILE_ID = %s \n",&file_id);

    if(strcmp(file_id,"ELEVATION") == 0)
    {
      ch=fscanf(project_file_fptr,"%s",elev_file);
      if((elev_file_fptr=fopen(elev_file,"r"))==NULL)
      {printf("Can't open Elevation File : %s \n",elev_file);
       exit(EXIT_FAILURE);}
      printf("ELEVATION          %s \n",elev_file);
    }
    if(strcmp(file_id,"PRECIP") == 0)
    {
      ch=fscanf(project_file_fptr,"%s",rain_file);
      if((rain_file_fptr=fopen(rain_file,"r"))==NULL)
      {printf("Can't open Rain File : %s \n",rain_file);
       exit(EXIT_FAILURE);}
```

```
   printf("PRECIP              %s \n",rain_file);
  }
  if(strcmp(file_id,"SOIL") == 0)
  {
   ch=fscanf(project_file_fptr,"%s",soil_file);
   if((soil_file_fptr=fopen(soil_file,"r"))==NULL)
   {printf("Can't open Soil File : %s \n",soil_file);
   exit(EXIT_FAILURE);}
   printf("SOIL                %s \n",soil_file);
  }
  if(strcmp(file_id,"LANDUSE") == 0)
  {
   ch=fscanf(project_file_fptr,"%s",iman_file);
   if((iman_file_fptr=fopen(iman_file,"r"))==NULL)
   {printf("Can't open Landuse File : %s \n",iman_file);
   exit(EXIT_FAILURE);}
   printf("LANDUSE             %s \n",iman_file);
  }
  if(strcmp(file_id,"CONTROL") == 0)
  {
   ch=fscanf(project_file_fptr,"%s",control_file);
   if((control_file_fptr=fopen(control_file,"r"))==NULL)
   {printf("Can't open control File : %s \n",control_file);
   exit(EXIT_FAILURE);}
   printf("CONTROL              %s \n",control_file);
  }
  if(strcmp(file_id,"CHANNEL") == 0)
  {
   ch=fscanf(project_file_fptr,"%s",chn_file);
   if((chn_file_fptr=fopen(chn_file,"r"))==NULL)
   {printf("Can't open Channel File : %s \n",chn_file);
   exit(EXIT_FAILURE);}
   printf("CHANNEL             %s \n",chn_file);
  }
  if(strcmp(file_id,"NODE") == 0)
  {
   ch=fscanf(project_file_fptr,"%s",node_file);
   if((node_file_fptr=fopen(node_file,"r"))==NULL)
   {printf("Can't open Node File : %s \n",node_file);
   exit(EXIT_FAILURE);}
   printf("NODE             %s \n",node_file);
  }
  if(strcmp(file_id,"LINK") == 0)
  {
   ch=fscanf(project_file_fptr,"%s",link_file);
   if((link_file_fptr=fopen(link_file,"r"))==NULL)
   {printf("Can't open Link File : %s \n",link_file);
   exit(EXIT_FAILURE);}
   printf("LINK             %s \n",link_file);
  }
  if(strcmp(file_id,"MASK") == 0)
  {
   ch=fscanf(project_file_fptr,"%s",shap_file);
   if((shap_file_fptr=fopen(shap_file,"r"))==NULL)
   {printf("Can't open Shape File : %s \n",shap_file);
   exit(EXIT_FAILURE);}
   printf("MASK                %s \n",shap_file);
  }
  if(strcmp(file_id,"STORAGE_DEPTH") == 0)
  {
   ch=fscanf(project_file_fptr,"%s",sdep_file);
   if((sdep_file_fptr=fopen(sdep_file,"r"))==NULL)
   {printf("Can't open Storage Depth File : %s \n",sdep_file);
   exit(EXIT_FAILURE);}
```

```
  printf("STORAGE_DEPTH                 %s \n",sdep_file);
}
if(strcmp(file_id,"SUMMARY") == 0)
{
  ch=fscanf(project_file_fptr,"%s",Summ_file);
  if((Summ_file_fptr=fopen(Summ_file,"w"))==NULL)
  {printf("Can't open Output PRN File : %s \n",Summ_file);
  exit(EXIT_FAILURE);}
  printf("SUMMARY          %s \n",Summ_file);
}
if(strcmp(file_id,"DISCHARGE_OUT") == 0)
{
  ch=fscanf(project_file_fptr,"%s",dis_out_file);
  if((dis_out_file_fptr=fopen(dis_out_file,"w"))==NULL)
  {printf("Can't open Discharge File : %s \n",dis_out_file);
  exit(EXIT_FAILURE);}
  printf("DISCHARGE_OUT       %s \n",dis_out_file);
}
if(strcmp(file_id,"SEDIMENT_OUT") == 0)
{
  ch=fscanf(project_file_fptr,"%s",sed_out_file);
  if((sed_out_file_fptr=fopen(sed_out_file,"w"))==NULL)
  {printf("Can't open Sediment File : %s \n",sed_out_file);
  exit(EXIT_FAILURE);}
  printf("SEDIMENT_OUT        %s \n",sed_out_file);
}

/* Reading in the time-series grid names                          */
if(strcmp(file_id,"WATER_DEPTH") == 0)
{
  ch=fscanf(project_file_fptr,"%s",dname);
  printf("WATER_DEPTH                %s \n",&dname);
}
if(strcmp(file_id,"RAINFALL") == 0)
{
  ch=fscanf(project_file_fptr,"%s",rname);
  printf("RNAME                 %s \n",&rname);
}
if(strcmp(file_id,"INF_DEPTH") == 0)
{
  ch=fscanf(project_file_fptr,"%s",vinfname);
  printf("INF_DEPTH             %s \n",&vinfname);
}
if(strcmp(file_id,"SAND_FLUX_OUT") == 0)
{
  ch=fscanf(project_file_fptr,"%s",sandfluxname);
  printf("SAND_FLUX_OUT       %s \n",&sandfluxname);
}
if(strcmp(file_id,"SILT_FLUX_OUT") == 0)
{
  ch=fscanf(project_file_fptr,"%s",siltfluxname);
  printf("SILT_FLUX_OUT       %s \n",&siltfluxname);
}
if(strcmp(file_id,"CLAY_FLUX_OUT") == 0)
{
  ch=fscanf(project_file_fptr,"%s",clayfluxname);
  printf("CLAY_FLUX_OUT       %s \n",&clayfluxname);
}
if(strcmp(file_id,"TOTAL_FLUX_OUT") == 0)
{
  ch=fscanf(project_file_fptr,"%s",totalfluxname);
  printf("TOTAL_FLUX_OUT      %s \n",&totalfluxname);
}
if(strcmp(file_id,"SAND_MFAC_OUT") == 0)
```

```
    {
      ch=fscanf(project_file_fptr,"%s",sandMFACname);
      printf("SAND_MFAC_OUT       %s \n",&sandMFACname);
    }
    if(strcmp(file_id,"SILT_MFAC_OUT") == 0)
    {
      ch=fscanf(project_file_fptr,"%s",siltMFACname);
      printf("SILT_MFAC_OUT       %s \n",&siltMFACname);
    }
    if(strcmp(file_id,"CLAY_MFAC_OUT") == 0)
    {
      ch=fscanf(project_file_fptr,"%s",clayMFACname);
      printf("CLAY_MFAC_OUT       %s \n",&clayMFACname);
    }
    if(strcmp(file_id,"TOTAL_MFAC_OUT") == 0)
    {
      ch=fscanf(project_file_fptr,"%s",totalMFACname);
      printf("TOTAL_MFAC_OUT      %s \n",&totalMFACname);
    }
    if(strcmp(file_id,"SED_CONC") == 0)
    {
      ch=fscanf(project_file_fptr,"%s",cname);
      printf("SED_CONC                %s \n",&cname);
    }
    if(strcmp(file_id,"SUS_SED") == 0)
    {
      ch=fscanf(project_file_fptr,"%s",susname);
      printf("SUS_SED             %s \n",&susname);
    }
    if(strcmp(file_id,"EROS_DEP") == 0)
    {
      ch=fscanf(project_file_fptr,"%s",erosdepname);
      printf("EROS_DEP                %s \n",&erosdepname);
    }
  }

  }while(ch != EOF);

  if((debug =fopen("debug.txt","w"))==NULL)
  {printf("Can't open debug File \n");
  exit(EXIT_FAILURE);}


  printf("Successfully Read all data from the Project File \n");
  fclose(project_file_fptr);

}
```

```
                /*******************************/
                /*      ReadControlFile.c      */
                /*******************************/

#include "all.h"

extern void ReadControlFile()
{
  int i,j,l;

  printf("Reading CONTROL file \n");

  fprintf(Summ_file_fptr,"\nCONTROL Input Data \n");
  fprintf(Summ_file_fptr,  "================== \n");
  fprintf(Summ_file_fptr,  "Control file : %s\n", control_file);

  fscanf(control_file_fptr,"%f %ld %ld %ld %ld",
        &dt,&niter,&nitrn,&nprn,&nplt);

  fprintf(Summ_file_fptr,
  "DT = %2.2f sec NITER= %ld NITRN = %ld NPRN = %ld NPLT = %ld\n\n",
                             dt,niter,nitrn,nprn,nplt);

  fprintf(Summ_file_fptr,"Overland outlet cell: \n");
  fscanf(control_file_fptr,"%ld %ld %f",&jout,&kout,&sovout);
  fprintf(Summ_file_fptr,"JOUT = %ld KOUT = %ld SOVOUT = %g \n\n",
                                          jout,kout,sovout);

  fscanf(control_file_fptr,"%i",&chancheck);
  fprintf(Summ_file_fptr,"CHANCHECK= %i \n",chancheck);

  fscanf(control_file_fptr,"%f",&elconv);
  fprintf(Summ_file_fptr,"ELCONV = %.2f\n",elconv);

  fprintf(Summ_file_fptr,"\nRainfall Data \n");
  fprintf(Summ_file_fptr,  "============= \n");

  fscanf(control_file_fptr,"%ld ",&irain);
  fprintf(Summ_file_fptr,"IRAIN = %ld \n",irain);

  if(irain == 0)
  {
   fscanf(control_file_fptr,"%f ",&crain);
   fprintf(Summ_file_fptr,"CRAIN = %.3f (mm/h)\n",crain);
  }
  else
  {
   fprintf(Summ_file_fptr,"Rainfall file: %s \n",rain_file);
   fscanf(control_file_fptr,"%ld %ld ",&nrg,&nread);
   fprintf(Summ_file_fptr,"NRG = %ld NREAD = %ld \n",nrg,nread);

   xrg = (float*) malloc((nrg+1)*sizeof(float));
   yrg = (float*) malloc((nrg+1)*sizeof(float));
   rrg = (float*) malloc((nrg+1)*sizeof(float));

   for(l=1;l<=nrg;l++)
   {
     fscanf(control_file_fptr,"%f %f ",&xrg[l],&yrg[l]);
     fprintf(Summ_file_fptr,"XRG[%ld] = %8.6f\tYRG[%ld] = %8.6f \n",
                                       l,xrg[l],l,yrg[l]);
   }
  }
```

```
fprintf(Summ_file_fptr,"\nLand Use Parameters \n");
fprintf(Summ_file_fptr,  "=================== \n");

/* Allocate memory for the variables derived from the landuse    */
pman = (float*) malloc((nman+1)*sizeof(float));
retention = (float*) malloc((nman+1)*sizeof(float));
cfactor = (float*) malloc((nman+1)*sizeof(float));
pfactor = (float*) malloc((nman+1)*sizeof(float));

fscanf(control_file_fptr," %ld %ld ",&nman,&indexsdep);
fprintf(Summ_file_fptr,"NMAN = %ld INDEXSDEP = %ld \n\n",
                                         nman,indexsdep);
fprintf(Summ_file_fptr,"\tIndex\tManning    Intercept.\tCusle\tPusle\n
fprintf(Summ_file_fptr,"\t-----\t-- n --    -- [mm] --\t-----\t-----\n

for(i=1;i<=nman;i++)
{
 fscanf(control_file_fptr,"%f %f %f %f",
        &pman[i], &retention[i],&cfactor[i],&pfactor[i]);
 fprintf(Summ_file_fptr,
    "\t%3ld%11.3f\t%10.3f\t%5.3f\t%5.3f\n",
                 i,pman[i],retention[i],cfactor[i],pfactor[i]);
}

fprintf(Summ_file_fptr,"\nSoil type Parameters \n");
fprintf(Summ_file_fptr,  "==================== \n");

fscanf(control_file_fptr,"%i %i %i",&indexinf,&indexeros,&nsoil);
fprintf(Summ_file_fptr,"INDEXINF= %i INDEXEROS= %i NSOIL= %i \n",
                               indexinf,indexeros,nsoil);

fprintf(Summ_file_fptr,
  "\n        Index    Ks        G          Md        %%Sand      %%Silt
fprintf(Summ_file_fptr,
  "        -----  [cm/h]    [cm]       -----      -----      -----     --
if(indexinf == 1)
{
 pinf = floatMemAlloc2d(nsoil,8);

 /* Read values from the CONTROL file */
 for(i=1;i<=nsoil;i++)
 {
   fprintf(Summ_file_fptr,"%10i",i);
   for(j=1;j<=7;j++)
   {
     fscanf(control_file_fptr,"%g ",&pinf[i][j]);
     fprintf(Summ_file_fptr,"%10.3f",pinf[i][j]);
     /* Converts Ks: cm/h -> m/s                                  */
     if(j ==1) pinf[i][j] = pinf[i][j]/3600/100;
     /* Converts G:  cm -> m                                      */
     if(j ==2) pinf[i][j] = pinf[i][j]/100;
   }

   fprintf(Summ_file_fptr,"\n");
 }
}

fprintf(Summ_file_fptr,"\nInternal Gages Data \n");
fprintf(Summ_file_fptr,  "------------------- \n");

fscanf(control_file_fptr,"%ld %ld %i",&indexdis,&ndis, &unitsQ);
fprintf(Summ_file_fptr,
                 "INDEXDIS = %ld  NDIS = %ld  Q_units = %i \n",
```

```
                                                indexdis,ndis, unitsQ);

  if(indexdis == 1)
  {
    fprintf(Summ_file_fptr,"\n        Gage       Row       Column      Are
    fprintf(Summ_file_fptr,"        ----       ---       ------     [has]\
    iq = intMemAlloc2d(ndis,2);
    areaQ = (float*) malloc((ndis+1)*sizeof(float));

    for (i=1;i<=ndis;i++)
    {
     fprintf(Summ_file_fptr,"        %2i",i);
     for (j=1;j<=2;j++)
     {
       fscanf(control_file_fptr,"%ld ",&iq[i][j]);
       fprintf(Summ_file_fptr,"%10i",iq[i][j]);
     }
     fscanf(control_file_fptr,"%f", &areaQ[i]);
     fprintf(Summ_file_fptr,"%14.3f\n", areaQ[i]);
    }
  }


  fprintf(Summ_file_fptr,"\nInternal Sediment Gages Data \n");
  fprintf(Summ_file_fptr, "---------------------------- \n");

  fscanf(control_file_fptr,"%ld %ld %i",&indexsed,&nsed, &unitsQs);
  fprintf(Summ_file_fptr,
                    "INDEXSED = %ld  NSED = %ld Qs_Units = %i\n",
                                     indexsed,nsed, unitsQs);

  if(indexsed == 1)
  {
    fprintf(Summ_file_fptr,"\n        Gage       Row       Column      Are
    fprintf(Summ_file_fptr,"        ----       ---       ------     [has]\
    ised = intMemAlloc2d(nsed,2);
    areaQs = (float*) malloc((nsed+1)*sizeof(float));

    for(i=1;i<=nsed;i++)
    {
     fprintf(Summ_file_fptr,"        %2i",i);
     for (j=1;j<=2;j++)
     {
       fscanf(control_file_fptr,"%ld ",&ised[i][j]);
       fprintf(Summ_file_fptr,"%10i",ised[i][j]);
     }
     fscanf(control_file_fptr,"%f", &areaQs[i]);
     fprintf(Summ_file_fptr,"%14.3f\n",areaQs[i]);
    }
  }

  printf("Successfully Read All CONTROL Information \n");
  fclose(control_file_fptr);

  /* Allocate memory for the vector that holds the flow          */
  /* dicharge and sediment discharge at each of those locations   */
  q =  (float*) malloc((ndis+1)*sizeof(float));
  qsed = (float*) malloc((nsed+1)*sizeof(float));

}
```

```
            /*******************************/
            /*         ReadGrids.c         */
            /*******************************/

#include "all.h"

      /*********************************/
      /*    FCT: ReadGridHeaders       */
      /*********************************/

extern void ReadGridHeaders()
{
  /* Variables that will hold the ASCII grids header information  */
  char nrows_label[15], ncols_label[15], xllcorner_label[15],
    yllcorner_label[15], cellsize_label[15], nodatavalue_label[15],
    h_label[7][80];
  int  head,i;
  float h_value[7];

  /* Read the shape ASCII grid header to get project info         */
  /* the "header" variable will be used to hold the header info   */

  fprintf(Summ_file_fptr,"\nASCII grid files header  \n");
  fprintf(Summ_file_fptr,"------------------------- \n");
  fscanf(shap_file_fptr,"%s %i",&ncols_label, &n);
  fscanf(shap_file_fptr,"%s %i",&nrows_label, &m);
  fscanf(shap_file_fptr,"%s %f",&xllcorner_label, &xllcorner);
  fscanf(shap_file_fptr,"%s %f",&yllcorner_label, &yllcorner);
  fscanf(shap_file_fptr,"%s %f",&cellsize_label, &w);
  fscanf(shap_file_fptr,"%s %i",&nodatavalue_label, &nodatavalue);

  /* Do not use the xxxx_label variables because I want to        */
  /* make sure that the user entered them in the right order.     */

  head  = sprintf( header,        "ncols\t\t%i\n", n );
  head += sprintf( header + head, "nrows\t\t%i\n", m  );
  head += sprintf( header + head, "xllcorner\t%.2f\n", xllcorner);
  head += sprintf( header + head, "yllcorner\t%.2f\n", yllcorner);
  head += sprintf( header + head, "cellsize\t%.2f\n", w);
  head += sprintf( header + head, "NODATA_value\t%i\n",
                                                 nodatavalue);
  fprintf(Summ_file_fptr,header);

  /* Need to read the rest of the input grids headers             */
  for (i=1;i<=6;i++)
   {
     if(elev_file_fptr)
       fscanf(elev_file_fptr,"%s %f",&h_label[i], &h_value[i]);
     else
     {
       printf(" Elevation grid doesn't exist. Pease, define one\n");
       exit(EXIT_FAILURE);
     }

     if(nsoil > 1)
     {
       if(soil_file_fptr)
         fscanf(soil_file_fptr,"%s %f",&h_label[i], &h_value[i]);
       else
       {
         printf("\n\nYou have defined more than one soil type");
         printf("\nYou need to define a soil type grid\n\n");
         exit(EXIT_FAILURE);
       }
```

```
      }

      if(nman > 1)
      {
        if(iman_file_fptr)
          fscanf(iman_file_fptr,"%s %f",&h_label[i], &h_value[i]);
        else
        {
          printf("\n\nYou have defined more than one land use type");
          printf("\nYou need to define a land use type grid\n\n");
          exit(EXIT_FAILURE);
        }
      }

      if(chancheck == 1)
      {
        if(node_file_fptr)
          fscanf(node_file_fptr,"%s %f",&h_label[i], &h_value[i]);
        else
        {
          printf("\n\nYou have selected the channel routing option");
          printf("\nYou need to define a node number grid\n\n");
          exit(EXIT_FAILURE);
        }

        if(link_file_fptr)
          fscanf(link_file_fptr,"%s %f",&h_label[i], &h_value[i]);
        else
        {
          printf("\n\nYou have selected the channel routing option");
          printf("\nYou need to define a link number grid\n\n");
          exit(EXIT_FAILURE);
        }

      }

      if(indexsdep == 1)
      {
        if(sdep_file_fptr)
          fscanf(sdep_file_fptr,"%s %f",&h_label[i], &h_value[i]);
        else
        {
          printf("\n\nYou have selected the retention depth option");
          printf("\nYou need to define a retention depth grid\n\n");
          exit(EXIT_FAILURE);
        }
      }
    }

}

       /************************************/
       /*          ReadInputGrids          */
       /************************************/

extern void ReadInputGrids()
{
  int j,k;

  ncells = 0;

  for(j=1;j<=m;j++)
  {
    for(k=1;k<=n;k++)
```

```
          {
            fscanf(shap_file_fptr,"%ld ",&ishp[j][k]);

            if(ishp[j][k] != nodatavalue) ncells++;

            fscanf(elev_file_fptr,"%f ",&e[j][k]);
            e[j][k] = e[j][k] / elconv;

            if(nsoil > 0)
            {
              if(nsoil == 1)
              {
                isoil[j][k] = 1;
              }
              else
              {
                fscanf(soil_file_fptr,"%ld ",&isoil[j][k]);
              }
            }

            if(nman == 1)
            {
              iman[j][k] = 1;
            }
            else
            {
              fscanf(iman_file_fptr,"%ld ",&iman[j][k]);
            }

            if(chancheck == 1)
            {
              fscanf(node_file_fptr,"%ld",&node[j][k]);
              fscanf(link_file_fptr,"%ld",&link[j][k]);
            }

            if(indexsdep == 1) fscanf(sdep_file_fptr,"%f",&sdep[j][k]);
          }
        }

        /* Input ASCII grids .... read                                    */
        fprintf(Summ_file_fptr,
                          "\nThe following Grids have been read: \n");
        fprintf(Summ_file_fptr," ------------------------------ \n");
        fprintf(Summ_file_fptr,"Elevation Grid: %s \n", elev_file);
        fprintf(Summ_file_fptr,"Shape Grid: \t%s \n", shap_file);
        fprintf(Summ_file_fptr,"Soils Grid: \t%s \n", soil_file);
        fprintf(Summ_file_fptr,"Landuse Grid: \t%s \n", iman_file);
        fprintf(Summ_file_fptr,"Node Grid: \t%s \n", node_file);
        fprintf(Summ_file_fptr,"Link Grid: \t%s \n\n", link_file);
        printf("Successfully Read Input Grid Data \n");
        fprintf(Summ_file_fptr,"Number of cells in basin: %i\n", ncells);
        fprintf(Summ_file_fptr,"Basin area: %.3f [ha]\n", ncells*w*w/10000);

        fclose(elev_file_fptr);
        fclose(shap_file_fptr);
        if(nsoil > 1) fclose(soil_file_fptr);
        if(nman > 1) fclose(iman_file_fptr);
        if(chancheck == 1)
        {
          fclose(node_file_fptr);
          fclose(link_file_fptr);
        }
        if(indexsdep == 1) fclose(sdep_file_fptr);
      }
```

```
/*******************************/
/*          MemAlloc.c         */
/*******************************/

#include "all.h"

extern void GridsMemAlloc()
{
  int SizeFr;

  /* Allocate memory for input matrices                                    */
  ishp = intMemAlloc2d(m,n);
  isoil = intMemAlloc2d(m,n);
  iman  = intMemAlloc2d(m,n);
  e = floatMemAlloc2d(m,n);
  ret = floatMemAlloc2d(m,n);

  if(chancheck == 1)
  {
    link = intMemAlloc2d(m,n);
    node = intMemAlloc2d(m,n);
  }

  sdep = floatMemAlloc2d(m,n);

  /* Allocate memory for output matrices                                   */

  if(indexinf == 1) vinf = floatMemAlloc2d(m,n);

  h = floatMemAlloc2d(m,n);
  hch = floatMemAlloc2d(m,n);

  if(indexeros == 1)
  {
    qss = floatMemAlloc2d(m,n);
    MaxFluxCout = floatMemAlloc2d(m,n);
    totsus = floatMemAlloc2d(m,n);
    SusC = floatMemAlloc2d(m,n);
    totnetv = floatMemAlloc2d(m,n);

    /* Allocate memory for 3-D matrices                                    */
    for (SizeFr=0;SizeFr<4;SizeFr++)
    {
      qovs[SizeFr] = floatMemAlloc2d(m,n);
      vols[SizeFr] = floatMemAlloc2d(m,n);
      ssoil[SizeFr] = floatMemAlloc2d(m,n);
      SedFluxOut[SizeFr] = floatMemAlloc2d(m,n);
      MFAC[SizeFr] = floatMemAlloc2d(m,n);
    }
  }

  dqov = floatMemAlloc2d(m,n);
  dqch = floatMemAlloc2d(m,n);
  rtot = floatMemAlloc2d(m,n);
  rint = floatMemAlloc2d(m,n);
}
```

```
      /***********************************/
      /*        intMemAlloc2d            */
      /***********************************/

/* Allocates memory for a integer 2D-grid of NoRows*NoCols */
int** intMemAlloc2d(int NoRows, int NoCols)
{
  int k;
  int** matrix2d;

  matrix2d = (int **)malloc((NoRows+1)*sizeof(int *));

  for(k=0; k<(NoRows+1); k++)
  {
    *(matrix2d + k) =(int*)malloc((NoCols+1)*sizeof(int));
  }

  return matrix2d;
}

      /***********************************/
      /*        floatMemAlloc2d          */
      /***********************************/

/* Allocates memory for a floating 2D-grid of NoRows*NoCols */
float** floatMemAlloc2d(int NoRows, int NoCols)
{
  int k;
  float** matrix2d;

  matrix2d = (float **)malloc((NoRows+1)*sizeof(float *));

  for(k=0; k<(NoRows+1); k++)
  {
    *(matrix2d + k) =(float*)malloc((NoCols+1)*sizeof(float));
  }

  return matrix2d;
}
```

```
/*******************************/
/*      InitializeVars.c       */
/*******************************/

#include "all.h"

extern void InitializeVars()
{
  int SizeFr,i,j,k;

  /* Grid values                                                    */
  for(j=1;j<=m;j++)
  {
    for(k=1;k<=n;k++)
    {
      if(ishp[j][k] != nodatavalue)
      {
        /* retention depth was entered in mm.                          */
        ret[j][k] = retention[iman[j][k]] / 1000;
        dqov[j][k] = 0.0;
        dqch[j][k] = 0.0;
        vinf[j][k] = 0.0;
        rtot[j][k] = 0.0;
        MaxFluxCout[j][k] = 0.0;
        qss[j][k] = 0.0;
        rint[j][k] = 0.0;

        for (SizeFr=1;SizeFr<=3;SizeFr++)
        {
          vols[SizeFr][j][k] = 0.0;
          ssoil[SizeFr][j][k] = 0.0;
          qovs[SizeFr][j][k] = 0.0;
          SedFluxOut[SizeFr][j][k] = 0.0;
          MFAC[SizeFr][j][k] = 0.0;
        }
      }
    }
  }

  /* Total volumes                                                  */
  vin = 0.0;
  vout = 0.0;
  vinftot = 0.0;

  /* Counters                                                       */
  raincount = 0;
  icount = 1;
  ipcount = 1;
  ifcount = 1;

  /* Variables that hold min. and max. values                       */
  amaxdepth = 9e-30;
  amindepth = 9e30;
  amaxcdepth = 9e-30;
  amincdepth = 9e30;
  amaxvinf = 9e-30;
  aminvinf = 9e30;
  amaxnetv = 9e-30;
  aminnetv = 9e30;
  amaxFluxCoutOv = 9e-30;
  amaxFluxCoutCh = 9e-30;
  amaxrain = 9e-30;
  aminrain = 9e30;
```

```
amaxSusOv = 9e-30;
amaxSusCh = 9e-30;
amaxSusCov = 9e-30;
amaxSusCch = 9e-30;

/* Sediment volumes by size fraction (sand/silt/clay)              */
for (SizeFr=1;SizeFr<=3;SizeFr++)
{
  tot_eroded[SizeFr] = 0.0; /* Eroded during the simulation       */
  sus_ov[SizeFr] = 0.0;   /* Suspended remaining on overl.        */
  dep_ov[SizeFr] = 0.0;   /* Deposited remaining on overl.        */
  tot_ov[SizeFr] = 0.0;   /* Susp.+ Dep. remaining on overl.      */
  sus_ch[SizeFr] = 0.0;   /* Suspended remaining in channels      */
  dep_ch[SizeFr] = 0.0;   /* Deposited remaining in channels      */
  tot_ch[SizeFr] = 0.0;   /* Susp.+ Dep. remaining in channels    */
  sed_out[SizeFr] = 0.0;
}

/* Water volumes at the beggining & end of the  simulation          */
init_ov_vol = 0.0;
init_ch_vol = 0.0;
final_ov_vol = 0.0;
final_ch_vol = 0.0;

/* Sediment characts. for medium grain sizes at 20 degrees C        */
ds[1] = 0.00035;          /* Sand median diameter [m]   0.00035 */
ds[2] = 0.000016 ;        /* Silt median diameter [m]   0.000016*/
ds[3] = 0.000001;         /* Clay median diameter [m]   0.000001*/
ws[1] = 0.036;            /* Sand fall veleocity (m/s)  0.036   */
ws[2] = 2.2e-4;           /* Silt fall velocity (m/s)   2.2e-4  */
ws[3] = 8.6e-7;           /* Clay fall velocity (m/s)   8.6e-7  */

/* Initializing ICHN Array.  NOTE: it is assumed that the max.      */
/* number of nodes at any link is 100                               */
for(i=1;i<=maxlink;i++)
  for(k=1;k<=2;k++)
    for(j=1;j<=99;j++)
      ichn[i][j][k] = 0;
```

```
  /* Initializing channel and overland water depths                  */
  for(j=1;j<=m;j++)
  {
    for(k=1;k<=n;k++)
    {
      hch[j][k] = 0.0;

      if(indexsdep == 1)
      {
        h[j][k] = sdep[j][k];

        if(sdep[j][k] > 0.0 && chancheck == 1 && link[j][k] > 0)
        {
          hch[j][k] = sdep[j][k];

          if(sdep[j][k] > chp[link[j][k]][node[j][k]][3])
          {
            h[j][k] = sdep[j][k] - chp[link[j][k]][node[j][k]][3];
          }
          else
          {
            h[j][k] = 0.0;
          }
        }
      }
      else
      {
        h[j][k] = 0.0;
        sdep[j][k] = 0.0;
      }
    }
  }
}
```

```
/*******************************/
/*        ReadChannFile.c      */
/*******************************/

#include "all.h"

extern void ReadChannFile()
{
  char ch;
  int j,k;

  printf("Reading Channel Data \n");

  do{
   ch=fscanf(chn_file_fptr,"%ld",&nchan_link);
   ch=fscanf(chn_file_fptr,"%ld",&nchan_node[nchan_link]);

   for(k=1;k<=nchan_node[nchan_link];k++)
   {
     for(j=1;j<=6;j++)
     {
       ch=fscanf(chn_file_fptr,"%f ",&chp[nchan_link][k][j]);
     }
   }

  } while(ch!=EOF);

  maxlink = nchan_link;

  /* Writing Channel Information to Output Summary File           */

/*  fprintf(Summ_file_fptr,"\n Channel Input Data \n");
  fprintf(Summ_file_fptr," ------------------ \n");
  for(i=1;i<=maxlink;i++)
  {
    for(k=1;k<=nchan_node[i];k++)
    {
      for(j=1;j<=6;j++)
      {
        fprintf(Summ_file_fptr,"CHP[%ld][%ld][%ld] = %f ",
                                  i,k,j,chp[i][k][j]);
      }
      fprintf(Summ_file_fptr,"\n");
    }
  }
*/

  printf("Successfully Read Channel Data \n");
  fclose(chn_file_fptr);

}
```

```
/******************************/
/*        ChannToplg.c       */
/******************************/

#include "all.h"

extern void ChannToplg()
{

  int maxchn = 1;
  int downstream[50],upstream1[50],upstream2[50],
      ic,l,rr,cc,ll,rc,i,j,k, p,q,ro,co,outlink,outnode;
  float elev;

  outlink = link[jout][kout];
  outnode = node[jout][kout];
  /* Determining Row and Column Locations                                   */

  for(j=1;j<=m;j++)
  {
    for(k=1;k<=n;k++)
    {
      if(link[j][k] !=0 && node[j][k] !=0)
      {
        ichn[link[j][k]][node[j][k]][1] = j;
        ichn[link[j][k]][node[j][k]][2] = k;

        if(maxchn < node[j][k]) maxchn = node[j][k];
      }
    }
  }

  /* NOTE: because the last node of one link and the first node         */
  /* of the downstream link is the same, the number of nodes of         */
  /* one link is the last numbered node in the link plus one            */
  maxchn = maxchn + 2;  /* Maximum number of nodes + 1                  */

  /* Determining Downstream and Upstream Linkages                       */
  for(i=1;i<=maxlink;i++)
  {
    downstream[i] = 0;
    upstream1[i] = 0;
    upstream2[i] = 0;
  }

  for(i=1;i<=maxlink;i++)
  {
    rr = ichn[i][1][1];
    cc = ichn[i][1][2];

    /* Checking North                                                   */

    if(rr-1 >= 1)
    {
      ll = link[rr-1][cc];

      if(ll !=0 && ll != i)
      {
        if(upstream1[i] == 0)
        {
          upstream1[i] = ll;
        }
        else
```

```
      {
        upstream2[i] = ll;
      }
      downstream[ll] = i;
    }
  }

  /* Checking South                                          */

  if(rr+1 <= m)
  {
    ll = link[rr+1][cc];

    if(ll !=0 && ll != i)
    {
      if(upstream1[i] == 0)
      {
        upstream1[i] = ll;
      }
      else
      {
        upstream2[i] = ll;
      }
      downstream[ll] = i;
    }
  }

  /* Checking West                                           */

  if(cc-1 >= 1)
  {
    ll = link[rr][cc-1];

    if(ll !=0 && ll != i)
    {
      if(upstream1[i] == 0)
      {
        upstream1[i] = ll;
      }
      else
      {
        upstream2[i] = ll;
      }
      downstream[ll] = i;
    }
  }

  /* Checking East                                           */

  if(cc+1 <= n)
  {
    ll = link[rr][cc+1];

    if(ll !=0 && ll != i)
    {
      if(upstream1[i] == 0)
      {
        upstream1[i] = ll;
      }
      else
      {
        upstream2[i] = ll;
      }
```

```
      downstream[ll] = i;
    }
  }
}

/*Adding the downstream Row and Column to the end of every Link */

for(i=1;i<=maxlink;i++)
{
  rc = 0;

  for(j=1;j<=maxchn;j++)
  {
    if(i != outlink)
    {
      if(ichn[i][j][1] == 0 && rc == 0)
      {
        if(ichn[i][j-1][1] > 0)
        {
          ll = downstream[i];

          ichn[i][j][1] = ichn[ll][1][1];
          ichn[i][j][2] = ichn[ll][1][2];
        }
        rc = 1;
      }
    }
  }
}

/* Adding a -1 to the end of every link except for the outlet    */
/* link                                                          */

for(i=1;i<=maxlink;i++)
{
  for(k=1;k<=2;k++)
  {
    for(j=1;j<=maxchn;j++)
    {
      if(i != outlink)
      {
        if(ichn[i][j][k] == 0)
        {
          if(ichn[i][j-1][k] > 0) ichn[i][j][k] = -1;
        }
      }
    }
  }
}

/* Writing channel network info and topology to summary file    */

printf("Number of Channels           = %ld \n",maxlink);
printf("Maximum Length of Channels = %ld \n",maxchn);

fprintf(Summ_file_fptr,"\nChannel Linkage Information\n");
fprintf(Summ_file_fptr,"---------------------------\n\n");

fprintf(Summ_file_fptr,"Maximum Number of Links = %ld \n",
                                             maxlink);
fprintf(Summ_file_fptr,"Maximum Number of Nodes = %ld \n\n",
                                             maxchn);

for(i=1;i<=maxlink;i++)
```

```
  {
    for(k=1;k<=2;k++)
    {
      for(j=1;j<=maxchn;j++)
      {
        if(j<maxchn)
        {
          fprintf(Summ_file_fptr,"%5ld",ichn[i][j][k]);
        }
        else
        {
          fprintf(Summ_file_fptr,"%5ld \n",ichn[i][j][k]);
        }
      }
    }

    fprintf(Summ_file_fptr,"\n");
  }

  fprintf(Summ_file_fptr,
          " Link     Downstream    Upstream1   Upstream2 \n\n");

  for(i=1;i<=maxlink;i++)
  {
    fprintf(Summ_file_fptr,"%4ld %10ld  %10ld  %10ld \n",
                     i,downstream[i],upstream1[i],upstream2[i]);

    printf("%4ld %10ld  %10ld  %10ld \n",
                     i,downstream[i],upstream1[i],upstream2[i]);
  }

  /* Mask grid value equal to 2 if channel going through it        */
  for(ic=1;ic<=maxlink;ic++)
  {
    for(l=1;l<=maxchn;l++)
    {
      j=ichn[ic][l][1];
      k=ichn[ic][l][2];

      if(j > 0 && chancheck == 1)
      {
       ishp[j][k] = 2;
      }
    }
  }


  /* Channel characteristics at the outlet                          */
  wchout = chp[outlink][outnode][2];
  dchout = chp[outlink][outnode][3];
  rmanout = chp[outlink][outnode][5];
  sfactorout = chp[outlink][outnode][6];
  for(p=-1; p<=1; p++)
  {
    for(q=-1; q<=1; q++)
    {
      ro = jout + p;
      co = kout + q;
      if(ro>0 && ro<=m && co>0 && co<=n && ro!=jout && co!=kout)
        if(ishp[ro][co] == 2 ) elev = e[ro][co];
    }
  }
  sout = (float)(fabs(elev - e[jout][kout] )/ w);
}
```

```
/********************************/
/*        CompVolumes.c        */
/********************************/

#include "all.h"

extern void CompInitialVol()

{
  int j,k;
  float wch, dch, sslope, tw_ch, area_ch, vol_ch;

  for(j=1;j<=m;j++)
  {
    for(k=1;k<=n;k++)
    {
      if(ishp[j][k] != nodatavalue)
      {
        if(chancheck == 1 && link[j][k] > 0)
        {
          wch = chp[link[j][k]][node[j][k]][2];
          dch = chp[link[j][k]][node[j][k]][3];
          sslope = chp[link[j][k]][node[j][k]][4];

          if(hch[j][k] < dch)
          {
           tw_ch = wch + 2*hch[j][k]*sslope;
           area_ch = (tw_ch + wch)/2 * hch[j][k];
           vol_ch = area_ch * w * chp[link[j][k]][node[j][k]][6];
          }
          else
          {
           tw_ch = wch + 2*dch*sslope;
           area_ch = (tw_ch + wch)/2 * dch;
           area_ch = area_ch + (hch[j][k] - dch)*tw_ch;
           vol_ch = area_ch * w * chp[link[j][k]][node[j][k]][6];
          }

          init_ch_vol = init_ch_vol + vol_ch;
          tw_ch = wch + 2*dch*sslope;
          init_ov_vol = init_ov_vol + (w-tw_ch)*w*h[j][k];
        }
        else
        {
         init_ov_vol = init_ov_vol + h[j][k]*w*w;
        }
      }
    }
  }
}
```

```
        /*******************************/
         /* COMPUTE FINAL STORAGE VOLUME */
        /*******************************/
extern void CompFinalVol()
{
  int j,k;
  float wch, dch, sslope, tw_ch, area_ch, vol_ch;

  for(j=1;j<=m;j++)
  {
    for(k=1;k<=n;k++)
    {
      if(ishp[j][k] != nodatavalue)
      {
        if(chancheck == 1 && link[j][k] > 0)
        {
          wch = chp[link[j][k]][node[j][k]][2];
          dch = chp[link[j][k]][node[j][k]][3];
          sslope = chp[link[j][k]][node[j][k]][4];

          if(hch[j][k] < dch)
          {
           tw_ch = wch + 2*hch[j][k]*sslope;
           area_ch = (tw_ch + wch)/2 * hch[j][k];
           vol_ch = area_ch * w * chp[link[j][k]][node[j][k]][6];
          }
          else
          {
           tw_ch = wch + 2*dch*sslope;
           area_ch = (tw_ch + wch)/2 * dch;
           area_ch = area_ch + (hch[j][k] - dch)*tw_ch;
           vol_ch = area_ch * w * chp[link[j][k]][node[j][k]][6];
          }

          final_ch_vol = final_ch_vol + vol_ch;
          tw_ch = wch + 2*dch*sslope;
          final_ov_vol = final_ov_vol + (w-tw_ch)*w*h[j][k];
        }
        else
        {
         final_ov_vol = final_ov_vol + h[j][k]*w*w;
        }
      }
    }
  }
}
```

```
/*******************************/
/*        OvrlDepth.c         */
/*******************************/

#include "all.h"

extern void OvrlDepth( )
{
  int j,k,l,rindex,icall;
  float hov;

  /*******************************/
  /*    Read rainfall intensity  */
  /*******************************/

  icall = 0;
  rindex = 1;

  if(iter > nitrn) rindex = 0;
  if(iter <= nitrn && irain == 1)
  {
    icall=1;

    raincount = raincount + 1;

    if(raincount == nread+1) raincount = 1;

    if(raincount == 1)
    {
      for(l=1;l<=nrg;l++)
      {
        fscanf(rain_file_fptr,"%f ",&rrg[l]);
      }
    }
  }

  /*********************************************/
  /*    Updating overland depth (water balance) */
  /*********************************************/

  /* Applying the Rainfall to each Grid Cell within the Watershed */

  for(j=1;j<=m;j++)
  {
    for(k=1;k<=n;k++)
    {
      if(ishp[j][k] != nodatavalue)
      {
        if(irain == 0)  /* Uniform Rainfall */
        {
          /* Convert uniform rainfall intensity: mm/h -> m/s      */

          rint[j][k] = (float)(crain/3600.0/1000.0);

          /* Rainfall rate reduced until intercept. is satisfied  */

          if (ret[j][k] != 0) intercept(j,k);
        }
        else      /* Spatially Distributed Rainfall */
        {
          if(icall == 1)
```

```
   {
     rain(j,k);
   }
 }

 if(iter > nitrn) rint[j][k] = 0.0;

 /* Determining the Min. and Max. Rainfall Intensity       */

 if(rint[j][k] <= aminrain) aminrain=rint[j][k];
 if(rint[j][k] >= amaxrain) amaxrain=rint[j][k];

 /* Determines total rainfall depth */

 rtot[j][k] = rtot[j][k] + rint[j][k]*dt;

 /* Computing the Overland Depth due to the Overland Flow  */

 hov = dqov[j][k] * dt / (w*w);

 /* Computing the Total Overland Depth due to the          */
 /* overland flow, previous overland depth, plus rainfall  */
 /* minus  interception                                    */

 hov = hov + h[j][k] + rindex * rint[j][k] * dt;

 h[j][k] = hov;

 /* When HOV < 0, then a negative depth situation occurs. */

 if(hov < 0.0)
 {
   if(hov > -0.0001) hov = 0;
   else
   {

     fprintf(Summ_file_fptr,
       "Negative Depth (HOV = %f) in the Overland Plane \n",
                                                       hov);
     fprintf(Summ_file_fptr,
            "Grid Cell : Row = %ld   Column = %ld \n",j,k);
     fprintf(Summ_file_fptr,
                  "dqov[%ld][%ld] * dt / (w*w) = %f \n",
                            j,k,dqov[j][k]*dt/(w*w));
     fprintf(Summ_file_fptr,"rint[%ld][%ld] = %f \n",
                                   j,k,rint[j][k]);
     fprintf(Summ_file_fptr,"h[%ld][%ld] = %f \n",
                                     j,k,h[j][k]);
     RunTime(clock());

     GridsMemFree();

     exit(0);
   }
 }

 /* Calling the Infiltration Subroutine. HOV will be       */
 /* modified by subtracting  the infiltration losses.      */
 /* NOTE : If the infiltration volume that can be  lost    */
 /* for DT is greater than the amount of surface           */
 /* volume present, then HOV will be reduced to zero.      */
```

```
          if(indexinf == 1) infilt(j,k);

          /* Determining the Minimum and Maximum Overland Depth      */
          /* and  the Minimum and Maximum Infiltration Depth         */

          if(h[j][k] < amindepth) amindepth = h[j][k];
          if(h[j][k] > amaxdepth) amaxdepth = h[j][k];

          if(vinf[j][k] < aminvinf) aminvinf = vinf[j][k];
          if(vinf[j][k] > amaxvinf) amaxvinf = vinf[j][k];

          /* Once used in the water balance calculation, this        */
          /* variable is reset to zero. It will be recalculated      */
          /* in the next time step in the ov. flow routing           */

          dqov[j][k] = 0.0;

          if(iter == niter)  /* Last computation step */
          {
            /* Total Volume of Rainfall minus interception;          */

            vin = vin + rtot[j][k] * w * w;

            /* Total Volume of Infiltration                          */

            vinftot = vinftot + vinf[j][k] * w * w;
          }
        }
      }
    }
}
```

```
/********************************/
/*          infilt.c            */
/********************************/

#include "all.h"

extern void infilt(int j, int k)
{

  int iinf;

  float hydcon, cs, smd, p1, p2, rinf;

  iinf = isoil[j][k];
  hydcon = pinf[iinf][1];
  cs = pinf[iinf][2];
  smd = pinf[iinf][3];

  /* Infiltration rate                                                 */

  p1 = (float)(hydcon*dt - 2.0*vinf[j][k]);
  p2 = hydcon*(vinf[j][k] + cs*smd);
  rinf = (float)((p1 + sqrt(pow(p1,2.0) + 8.0*p2*dt))/(2.0*dt));

  /* When the Rate of Infiltration is Greater Than HOV/DT,             */
  /* then all the water on the overland cell is assumed to             */
  /* be infiltrated and the overland depth is reduced to zero.         */

  if (h[j][k]/dt <= rinf)
  {
    rinf = h[j][k]/dt;
    h[j][k] = 0.0;
  }
  else
  {
    h[j][k] = h[j][k] - rinf*dt;
  }

  /* Accumulated volume of infiltrated water                           */

  vinf[j][k] = vinf[j][k] + rinf*dt;
}
```

```
/******************************/
/*        intercept.c        */
/******************************/

#include "all.h"

extern void intercept(int j, int k)
{

  if (rint[j][k] * dt >= ret[j][k])
  {
    rint[j][k] = rint[j][k] - ret[j][k] / dt;
    ret[j][k] = 0;
  }
  else
  {
    ret[j][k] = ret[j][k] - rint[j][k] * dt;
    rint[j][k] = 0;
  }

}
```

```
/*******************************/
/*            rain.c           */
/*******************************/

#include "all.h"

extern void rain(int j,int k)
{
  float totdist = 0.0;
  float totrain = 0.0;

  float xc,yc,dist,xul,yul;
  int l;

  rint[j][k] = -999.0;

  xul = xllcorner;
  yul = yllcorner + m * w;

  if(nrg == 1)
  {
    rint[j][k] = rrg[1];
  }
  else
  {
    for(l=1;l<=nrg;l++)
    {
      yc = (float)(yul - j * w + w / 2.0);
      xc = (float)(xul + k * w - w / 2.0);

      dist = (float)(sqrt(pow((yc-yrg[l]),2.0) +
                                        pow((xc-xrg[l]),2.0)));
      if(dist < 1e-5)
      {
        rint[j][k] = rrg[l];
      }
      else
      {
        totdist = (float)(totdist + 1.0/pow(dist,2.0));
        totrain = (float)(totrain + rrg[l]/pow(dist,2.0));
      }
    }
  }

  if(rint[j][k] == -999.0)
  {
    rint[j][k] = totrain / totdist;
  }

  /* Changing Units from inches/hour to meters/second */

  rint[j][k] = (float)(rint[j][k] * 0.0254 / 3600.);

  /* Rainfall rate is reduced until interception is satisfied */

  if (ret[j][k] != 0)
  {
    intercept(j,k);
  }

}
```

```
        /*******************************/
        /*        ChannDepth.c         */
        /*******************************/

#include "all.h"

extern void ChannDepth()
{
  int ic,j,l,k,jj;
  float wch, dch,sfactor, sdep_ov, inflowVol,vol_ov_in;

  for(ic=1;ic<=maxlink;ic++)
  {
   for(l=1;l<=nchan_node[ic];l++)
   {
     wch = chp[ic][l][2];
     dch = chp[ic][l][3];
     sfactor = chp[ic][l][6];

     j = ichn[ic][l][1];
     k = ichn[ic][l][2];
     jj = ichn[ic][l+1][1];

     /* Channel depths are updated at each link and node except   */
     /* for the last node of the link because the last node of    */
     /* one link is the same as the first node of the next link   */
     /* In the outlet link, we update depth in all the nodes      */
     /* NOTE: To understand the condition of (j > 0 && jj >= 0)   */
     /*   look at the CASC2D summary output file, where it says : */
     /*   Channel Linkage Information                             */

     if(j > 0 && jj >= 0)
     {
       /* Find new channel depth after adding inflow volume      */

       inflowVol = dqch[j][k]*dt;

       /* ... and the volume coming from the overland (vol_ov_in) */

       if(sdep[j][k] > dch)
         sdep_ov = sdep[j][k] - dch;
       else
         sdep_ov = 0.0;

       vol_ov_in = 0;

       if(h[j][k] > sdep_ov)
       {
         vol_ov_in = (h[j][k] - sdep_ov)*w*w;
         h[j][k] = sdep_ov;
       }

       hch[j][k] = newChnDepth(wch,dch,sfactor,j,k,
                          (inflowVol + vol_ov_in));

       /* Determining the Minimum and the Maximum Channel Depths  */

       if(hch[j][k] < amincdepth) amincdepth = hch[j][k];
       if(hch[j][k] > amaxcdepth) amaxcdepth = hch[j][k];

       /* Negative Depth in the Channel --> EXIT program          */

       if (hch[j][k] < 0.0)
```

```
          {
            if (hch[j][k] < 0.0 && hch[j][k] > -0.001)
              hch[j][k] = 0.0;
            else
              {
                fprintf(Summ_file_fptr,
                 "Channel Negative Depth:row = %ld column = %ld \n",
                                                                  j,k);
                fprintf(Summ_file_fptr,
                 "Depth at link = %ld, node = %ld is %.8f meters\n",
                                                      ic,l, hch[j][k]);
                RunTime(clock());

                GridsMemFree();

                exit(0);
              }
          }

          dqch[j][k] = 0.0;

        }  /* end of : if(j > 0 && jj >= 0)  */
      }  /* end of : for(l=1;l<=nchan_node[ic];l++)   */
    } /* end of : for(ic=1;ic<=maxlink;ic++)  */
}
```

```
/********************************/
/*           OvrlRout.c         */
/********************************/

/**********************************/
/*        FUNCT: OvrlRout         */
/**********************************/

#include "all.h"

extern void OvrlRout()
{

  int j,k,jj,kk,l;

  for(j=1;j<=m;j++)
  {
   for(k=1;k<=n;k++)
   {
     if(ishp[j][k] != nodatavalue)
     {
       for(l=-1;l<=0;l++)
       {
         jj=j+l+1;
         kk=k-l;

         if(jj <= m && kk <=n && ishp[jj][kk] != nodatavalue)
         {
           ovrl(j,k,jj,kk);
         }
       }
     }
   }
  }
}
```

```
    /**********************************/
    /*        FUNCT: Ovrl             */
    /**********************************/

extern void ovrl( int j,int k, int jj, int kk)
{
  int jfrom,kfrom,jto,kto;
  float a=1.0;
  float vel = 0.0;
  float so,sf,dhdx,hh,rman,alfa,dqq,stordepth;

  so = (e[j][k] - e[jj][kk])/w;

  dhdx = (h[jj][kk] - h[j][k])/w;

  sf = so - dhdx +(float)(1e-30);

  hh = h[j][k];

  rman = pman[iman[j][k]];

  if(chancheck == 1 && link[j][k] > 0)
  {
    if(sdep[j][k] > chp[link[j][k]][node[j][k]][3])
    {
      stordepth = sdep[j][k] - chp[link[j][k]][node[j][k]][3];
    }
    else
    {
      stordepth = 0.0;
    }
  }
  else
  {
    stordepth = sdep[j][k];
  }

  if(sf < 0)
  {
      hh = h[jj][kk];
      rman = pman[iman[jj][kk]];

      if(chancheck == 1 && link[jj][kk] > 0)
      {
        if(sdep[jj][kk] > chp[link[jj][kk]][node[jj][kk]][3])
        {
          stordepth =
            sdep[jj][kk] - chp[link[jj][kk]][node[jj][kk]][3];
        }
        else
        {
          stordepth = 0.0;
        }
      }
      else
      {
        stordepth = sdep[jj][kk];
      }
  }

  if(hh >= stordepth)
  {
    alfa = (float)((pow(fabs(sf),0.5))/rman);
```

```
/*  Note : The variable "a" represents the sign of the         */
/*  Friction Slope (Sf) Computing Overland Flow                */

if(sf >= 0) a = 1.0;

if(sf < 0) a = -1.0;

dqq = (float)(a*w*alfa*pow((hh-stordepth),1.667));

dqov[j][k] = dqov[j][k] - dqq;

dqov[jj][kk] = dqov[jj][kk] + dqq;

/* Compute overland sediment flow and erosion/deposition      */

if(indexeros == 1 && dqq != 0.0)
{
  if(a > 0)          /* (J,K) to (JJ,KK) */
  {
    jfrom = j;
    kfrom = k;
    jto = jj;
    kto = kk;
  }
  else                 /* (JJ-KK) to (J,K) */
  {
    jfrom = jj;
    kfrom = kk;
    jto = j;
    kto = k;
  }

  if (ishp[jfrom][kfrom] == 1)
  {
    RoutSedOvrl(dqq,sf,jfrom,kfrom,jto,kto);
  }
}

} /* End of HH >= STORDEPTH */

}   /* End of OVRL */
```

```
#include "all.h"

extern void RoutSedOvrl(float dqq, float sf,int jfrom,int kfrom,
                                              int jto,int kto)
{
  float V, qsSUS[4],qsADV[4],qsSUStot,SUStot, DEPtot,qsBM[4],
        totXSScap,capacity[4],RESIDcap, qsBMtot,
        qsEROS[4],qsEROStot,percent[4],qs[4],qsKR;

  int SizeFr,SoilType;

  V = (float)(fabs(dqq)/(w * h[jfrom][kfrom]));

  qsSUStot = 0;
  qsBMtot = 0;
  qsEROStot = 0;
  SUStot = 0;
  DEPtot = 0;

  SoilType = isoil[jfrom][kfrom];

  qsKR = KRcap(KRov,dqq,w,sf,jfrom,kfrom);

  for(SizeFr=1;SizeFr<=3;SizeFr++)
  {
    qsSUS[SizeFr] = 0;
    qsBM[SizeFr] = 0 ;
    qsEROS[SizeFr] = 0;
    qs[SizeFr] = 0;
    SUStot += qovs[SizeFr][jfrom][kfrom];
    DEPtot += vols[SizeFr][jfrom][kfrom];

  }

  for(SizeFr=1;SizeFr<=3;SizeFr++)
  {
    if(qovs[SizeFr][jfrom][kfrom] > 0)
    {
      if(qsKR < SUStot)
      {
        capacity[SizeFr] = qsKR * qovs[SizeFr][jfrom][kfrom]/SUStot;
        qsADV[SizeFr] = qovs[SizeFr][jfrom][kfrom] * V * dt / w;
        qsSUS[SizeFr] = MAX(capacity[SizeFr],qsADV[SizeFr]);
      }
      else
      {
        qsSUS[SizeFr] = qovs[SizeFr][jfrom][kfrom];
      }

      TransferSed(SizeFr,qovs,qsSUS[SizeFr],jfrom,kfrom,jto,kto);

      qsSUStot += qsSUS[SizeFr];
    }
  }
```

```
    totXSScap = MAX(0,qsKR - qsSUStot);

    if(totXSScap > 0 && DEPtot > 0)
    {
      for(SizeFr=1;SizeFr<=3;SizeFr++)
      {
        if(vols[SizeFr][jfrom][kfrom] > 0)
        {
          if(totXSScap < DEPtot)
          {
            qsBM[SizeFr] =
                       totXSScap * vols[SizeFr][jfrom][kfrom] / DEPtot;
          }
          else
          {
            qsBM[SizeFr] = vols[SizeFr][jfrom][kfrom];
          }

          TransferSed(SizeFr,vols,qsBM[SizeFr],jfrom,kfrom,jto,kto);

          qsBMtot += qsBM[SizeFr];
        }
      }
    }


    RESIDcap = MAX(0,totXSScap - qsBMtot);

    if(RESIDcap >0)
    {
      for(SizeFr=1;SizeFr<=3;SizeFr++)
      {
        percent[SizeFr] = pinf[SoilType][SizeFr + 3];

        qsEROS[SizeFr] = RESIDcap * percent[SizeFr];

        qsEROStot += qsEROS[SizeFr];

        TransferSed(SizeFr,ssoil,qsEROS[SizeFr],jfrom,kfrom,jto,kto);
      }
    }

    qss[jfrom][kfrom] += qsSUStot + qsBMtot + qsEROStot;

    for(SizeFr=1;SizeFr<=3;SizeFr++)
    {
      qs[SizeFr] = qsSUS[SizeFr] + qsBM[SizeFr] + qsEROS[SizeFr];

      SedFluxOut[SizeFr][jfrom][kfrom] += qs[SizeFr] / dt;
    }

    findMFAC(jfrom,kfrom,qsKR,dqq,qs);

}
```

```
            /******************************/
            /*          ChannRout.c       */
            /******************************/

#include "all.h"

       /*********************************/
       /*         FUNCT: ChannRout      */
       /*********************************/

extern void ChannRout()
{
  int ic,l;

  for(ic=1;ic<=maxlink;ic++)
  {
    for(l=1;l<=nchan_node[ic];l++)
    {
      /*  Note :  When ichn[ic][l+1][1] is less than zero,        */
      /*          then that indicates that the channel            */
      /*          routing for the current link is complete.       */
      /*          There is no routing for the outlet cell         */

      if(ichn[ic][l+1][1] > 0)
      {
       chnchn(ic,l);
      }

    } /* end for loop for the nodes in the link */
  } /* end for loop for the links  */
}

       /*********************************/
       /*         FUNCT: chnchn         */
       /*********************************/

extern void chnchn(int ic,int l)
{

  float a = 1.0;
  float vel = 0.0;
  float wch,dch,sslope,rmanch,sfactor,so, sf, hh, dhdx,
        dq, stordep, hchan,dq_ov;

  int  j,k,jj,kk,jjj,iic, ijun, ill;

  /* row and column of link IC and node l                         */

  j = ichn[ic][l][1];
  k = ichn[ic][l][2];

  /* row and column of following node in link IC                  */

  jj = ichn[ic][l+1][1];
  kk = ichn[ic][l+1][2];

  /* Note : JJJ is a check to see when the end of                 */
  /*        a channel link has been reached.                      */

  jjj = ichn[ic][l+2][1];
```

```
/* Channel characteristics :                                              */

wch = chp[ic][1][2];       /* width                    */
dch = chp[ic][1][3];       /* depth                    */
sslope = chp[ic][1][4];    /* side slope               */
rmanch = chp[ic][1][5];    /* manning's n              */
sfactor = chp[ic][1][6];   /* sinuosity factor         */
stordep = sdep[j][k];      /* Storage depth            */
hchan = hch[j][k];         /* Channel water depth      */
hh = hch[j][k] - stordep;

/*if(fabs(hh) < 1e-4) hh=0.0;  check for negligible water depth */

/* Channel slope                                                          */

so = (e[j][k] - e[jj][kk])/(w*sfactor);

dq_ov = 0.0;

/* If jjj<0, then the end of the channel link has been reached. */
/* Slope is computed with last node of current link and first   */
/*  node of following one                                       */

if(jjj < 0)
{
  for (iic=1;iic<=maxlink;iic++)
  {
    if(jj == ichn[iic][1][1] && kk == ichn[iic][1][2])
    {
      so = (e[j][k] - dch - e[jj][kk] + chp[iic][1][3])/(w*sfactor);
      ijun = iic;
    }
  }
}

dhdx = (hch[jj][kk] - hch[j][k])/(w*sfactor);

sf = (float)(so - dhdx + 1e-30);

/* Nota de Jorge: Sf se deberia quedar con el mismo signo */
if(fabs(sf) < 1e-20) sf = (float)(1e-20);

if (sf < 0.0)
{
  a = (float)(-1.0*a);

  if (jjj < 0)
  {
    /* Take channel chars. of the 1st node of downstream link */

    wch = chp[ijun][1][2];
    dch = chp[ijun][1][3];
    sslope = chp[ijun][1][4];
    rmanch = chp[ijun][1][5];
    sfactor = chp[ijun][1][6];
  }
```

```
    else
    {
      /*Take channel chars. of the next node within current link */

      wch = chp[ic][l+1][2];
      dch = chp[ic][l+1][3];
      sslope = chp[ic][l+1][4];
      rmanch = chp[ic][l+1][5];
      sfactor = chp[ic][l+1][6];
    }

    stordep = sdep[jj][kk];
    hchan = hch[jj][kk];
    hh = hch[jj][kk] - stordep;

    /*if(fabs(hn) < 1e-4) hh = 0.0;check for negligible  depth    */
  }

  /* Determining discharge                                        */

  dq = chnDischarge(hchan,hh,wch,dch,stordep,rmanch,a,sf,sfactor);

  /* Transfer flow from cell (j,k) to (jj,kk)                     */

  dqch[j][k] = dqch[j][k] - dq;
  dqch[jj][kk] = dqch[jj][kk] + dq;

  /* If (j,k) is a printout flow discharge internal location      */

  for (ill=1;ill<=ndis;ill++)
  {
    if(j == iq[ill][1] && k == iq[ill][2]) q[ill] = dq;
  }

  /* Channel sediment routing                                     */

  if(indexeros == 1 && dq != 0.0 && hchan !=0)
  {
    if(a >= 0.0)
    {
      RoutSedChn(dq,sf,j,k,jj,kk,wch,hchan);
    }
    else
    {
      RoutSedChn(-dq,-sf,jj,kk,j,k,wch,hchan);
    }
  }

}   /* End of CHNCHN */
```

```
            /********************************/
            /*        RoutSedChn.c          */
            /********************************/

#include "all.h"

      /**********************************/
      /*        FUNCT: RoutSedChn       */
      /**********************************/

extern void RoutSedChn(float dq,float sf,int jfrom,int kfrom,
                       int jto,int kto,float wch,float hchan)
{
  int SizeFr;
  float TotQs,V,supply,qs[4],adv_factor,
        EHvol[4],SUSvol[4], XSScap[4], BMvol[4];

  TotQs = 0.0;
  V = (float)(dq/(wch*hchan));

  for (SizeFr=1; SizeFr<=3;SizeFr++)
  {
    supply = qovs[SizeFr][jfrom][kfrom] + vols[SizeFr][jfrom][kfrom];
    qs[SizeFr] = 0; /* initialize qs to 0 in case that supply = 0 */

    if (supply != 0)
    {
      /* Potential transport cap. [m3] for size fraction SizeFr   */

      EHvol[SizeFr] = EHcap(SizeFr,wch,hchan,sf,dq);

      /* Suspended sediment transport capacity [m3] by advection  */
      /* Limit this transport by availability                     */

      adv_factor = MIN(V*dt/w,1);

      SUSvol[SizeFr] = qovs[SizeFr][jfrom][kfrom] * adv_factor;

      TransferSed(SizeFr,qovs,SUSvol[SizeFr],jfrom,kfrom,jto,kto);

      /* Excess capacity [m3] to move seds. from the bed-material */

      XSScap[SizeFr] = MAX(0, EHvol[SizeFr] - SUSvol[SizeFr]);

      /* Bed-material transport capacity [m3] by advection        */

      BMvol[SizeFr] = vols[SizeFr][jfrom][kfrom] * adv_factor;

      /* Volume of bed-material transported out of (jfrom,kfrom)  */

      BMvol[SizeFr] = MIN(XSScap[SizeFr],BMvol[SizeFr]);

      TransferSed(SizeFr,vols,BMvol[SizeFr],jfrom,kfrom,jto,kto);


      /* Total volume of transported sediment from outgoing cell  */

      qs[SizeFr] = SUSvol[SizeFr] + BMvol[SizeFr];

      /* Updates total sediment volume leaving cell (jfrom,kfrom) */

      qss[jfrom][kfrom] += qs[SizeFr];
```

```
      /* Keeps track of sediment flux out of (jfrom,kfrom)        */
      SedFluxOut[SizeFr][jfrom][kfrom] += qs[SizeFr] / dt;
      /* Total sediment volume leaving (jfrom,kfrom)              */
      TotQs += qs[SizeFr];
    }
  }
  findMFAC(jfrom,kfrom,TotQs,dq,qs);
}
```

```
              /*******************************/
              /*        RoutOutlet.c         */
              /*******************************/

#include "all.h"

extern void RoutOutlet()
{
  int ill;
  float hout, alfa,qoutch;

  qoutov = 0.0;
  qoutch = 0.0;

  /* FIRST:calculate the flow going out from the overl. portion   */

  alfa = (float)(sqrt(sovout)/pman[iman[jout][kout]]);

  /* Discharge from overland flow.  NOTE: because the water from  */
  /* this part of the outlet overland cell was already "poured"   */
  /* into the channel when updating the channel depth (channDepth)*/
  /* qoutov = 0 when the channel routing is selected              */

  if(h[jout][kout] > sdep[jout][kout])
  {
    qoutov =
      (float)(w*alfa*pow((h[jout][kout]-sdep[jout][kout]),1.667));
  }

  /* Overland water depth at outlet cell is reduced after taking  */
  /* the outflow out of the cell                                  */

  h[jout][kout] = (float)(h[jout][kout] - qoutov*dt/(pow(w,2.0)));

  /* SECOND:calculate the flow going out from the channel portion */

  if(chancheck == 1 && hch[jout][kout] > sdep[jout][kout])
  {
    hout = hch[jout][kout] - sdep[jout][kout];

    qoutch = chnDischarge(hch[jout][kout],hout,wchout,dchout,
                       sdep[jout][kout],rmanout,1,sout,sfactorout);

    dqch[jout][kout] = dqch[jout][kout] - qoutch;
  }

  /* The total outflow at the basin's outlet is given by adding   */
  /* the outflow from the overland & channel portion of the cell  */

  qout = qoutov + qoutch;

  /* Keeping Track of the Total Outflow Volume                    */

  vout = vout + qout*dt;

  /* Checking to see if the Peak Flow has been reached            */

  if(qout > qpeak)
  {
   qpeak = qout;
   tpeak = (float)( iter*dt/60.0);
  }
```

```
  /* Populating the Output Flows at the Watershed Outlet          */

  for(ill = 1;ill <= ndis; ill++)
  {
   if(jout == iq[ill][1] && kout == iq[ill][2])
   {
     q[ill] = qout;
   }
  }

}
```

```
/*******************************/
/*          Settling.c          */
/*******************************/

#include "all.h"

extern void Settling()
{
  int j,k,SizeFr;
  float depth, settlPerc[4], settl[4];

  for(j=1;j<=m;j++)
  {
   for(k=1;k<=n;k++)
   {
     if(ishp[j][k] != nodatavalue)
     {
       if (ishp[j][k] == 1) depth = h[j][k];
       if (ishp[j][k] == 2) depth = hch[j][k];

       if(depth > 0)
       {
          for (SizeFr=1;SizeFr<=3;SizeFr++)
          {
            /* Settling is calculated for                          */
            /* SizeFr=1 (sand); 2 (silt) and 3 (clay)              */

            if(depth > ws[SizeFr]*dt)
            {
              settlPerc[SizeFr] = (float)(ws[SizeFr] * dt / depth);
            }
            else
            {
              settlPerc[SizeFr] = 1;
            }

            settl[SizeFr] = qovs[SizeFr][j][k] * settlPerc[SizeFr];
            vols[SizeFr][j][k] += settl[SizeFr];
            qovs[SizeFr][j][k] -= settl[SizeFr];
          }
       }
     }
   }
  }

}
```

```
/******************************/
/*         RoutSedOut.c       */
/******************************/

#include "all.h"

extern void RoutSedOut()

{
  if(ishp[jout][kout] == 2 )
  {
    if(hch[jout][kout] != 0)
    {
      RoutSedChn(qout,sout,jout,kout,1,1,wchout,hch[jout][kout]);
    }
  }

  else
  {
    if(h[jout][kout] != 0)
    {
      RoutSedOvrl(qoutov,sovout,jout,kout,1,1);
    }
  }
}
```

```
            /******************************/
            /*         SedStats.c         */
            /******************************/

#include "all.h"

extern void SedStats()
{
  int j,k, SizeFr;
  float totdepv,totscourv;

  for(j = 1;j <= m; j++)
  {
    for(k = 1;k <= n; k++)
    {
      if(ishp[j][k] != nodatavalue)
      {
        totdepv = 0.0;
        totscourv = 0.0;
        totsus[j][k] = 0.0;

        for(SizeFr=1;SizeFr<=3;SizeFr++)
        {
          /* Total (sand+silt+clay) deposited sed. volume (m3)      */

          totdepv += vols[SizeFr][j][k];

          /* Total (sand+silt+clay) scoured sed. volume (m3) )      */

          totscourv += ssoil[SizeFr][j][k];

          /* Total (sand+silt+clay) suspended sed. volume (m3)      */

          totsus[j][k] += qovs[SizeFr][j][k];
        }

        /* Net (deposited - eroded) sediment volume (mm)            */

        totnetv[j][k] = (totdepv + totscourv) * 1000 / (w*w);

        /* Min. and Max. net erosion/depostion volume (m3)          */

        aminnetv = MIN(totnetv[j][k] , aminnetv);
        amaxnetv = MAX(totnetv[j][k] , amaxnetv);
```

```
            /* Absolute max. flux conc., susp. volume and suspended    */
            /* conc. at any time step and at any overland cell         */

            if(ishp[j][k] == 1)
            {
              amaxFluxCoutOv = MAX(MaxFluxCout[j][k],amaxFluxCoutOv);
              amaxSusOv = MAX(totsus[j][k],amaxSusOv);

              /* Limiting the mim. water depth to 0.5mm will reduce    */
              /* very high concentration calculations                  */
              if(h[j][k] > 5E-4)
              {
                SusC[j][k] = totsus[j][k] / (h[j][k]*w*w);
                amaxSusCov = MAX(amaxSusCov,SusC[j][k]);
              }
              else
              {
                SusC[j][k] = 0;
              }
            }

            /* .... at any channel cell                                */

            if(ishp[j][k] == 2)
            {

              amaxFluxCoutCh = MAX(amaxFluxCoutCh,MaxFluxCout[j][k]);
              amaxSusCh = MAX(amaxSusCh,totsus[j][k]);

              if(hch[j][k] > 5E-4)
              {
                SusC[j][k] = totsus[j][k] /
                        (hch[j][k]*w*chp[link[j][k]][node[j][k]][2]);

                amaxSusCch = MAX(amaxSusCch,SusC[j][k]);
              }
              else
              {
                SusC[j][k] = 0;
              }
            }
          }
        }
      }
    }
```

```
          /*******************************/
          /*      WriteOutflows.c        */
          /*******************************/


#include "all.h"

extern void WriteOutflow()
{
  int ill,ils,jsed,ksed,i;

  /*****************************/
  /*   write results to screen  */
  /*****************************/
  if(chancheck == 1)
  {
    printf(
    "Time (Min) = %7.2f\tQ (m3/s)= %7.2f\tChan. Depth(m) = %7.3f \n"
                          ,iter*dt/60.0, qout,hch[jout][kout]);
  }
  else
  {
    printf(
    "Time (Min) = %7.2f\tQ (m3/s) = %7.2f\tOver. Depth(m) = %7.3f\n"
                            ,iter*dt/60.0,qout,h[jout][kout]);
  }

  /*****************************************************/
  /*   write outflow and sediment flow to output files  */
  /*****************************************************/

  if(iter==1)
  {
    fprintf(dis_out_file_fptr," Time_(min) ");
    if(indexeros == 1)fprintf(sed_out_file_fptr,"Time_(min) ");
    for (i=1;i<=ndis;i++)
    {
      fprintf(dis_out_file_fptr,"Row%iCol%i ",iq[i][1],iq[i][2]);
      if(indexeros == 1)
          fprintf(sed_out_file_fptr,"Row%iCol%i ",
                                      ised[i][1],ised[i][2]);
    }
    fprintf(dis_out_file_fptr,"\n");
    if(indexeros == 1)fprintf(sed_out_file_fptr,"\n");
  }

  if(icount == nprn)
  {
    /* Write outflow discharge at selected locations              */


    fprintf(dis_out_file_fptr,"%.2f",iter*dt/60.0);

    for(ill=1; ill<=ndis; ill++)
    {
      switch(unitsQ)
      {
        case 1: /* Discharge in m3/s */
          fprintf(dis_out_file_fptr,"%15.3f ",q[ill]);
          break;

        case 2: /* Discharge in cfs */
          fprintf(dis_out_file_fptr,"%15.3f ",
```

```
                              q[ill]* pow(3.28084,3));
          break;

        case 3: /* Discharge in mm/h */
          fprintf(dis_out_file_fptr,"%15.3f ",
                              q[ill] / areaQ[ill] * 360);
          break;
      }
    }

    fprintf(dis_out_file_fptr,"\n");

    /* Writing outflow sediment Data at selected locations      */

    if(indexeros == 1 && sed_out_file[0] != '\0')
    {

      fprintf(sed_out_file_fptr,"%.2f",iter*dt/60.0);

      for(ils=1; ils<=nsed; ils++)
      {
        jsed = ised[ils][1];
        ksed = ised[ils][2];

        qsed[ils] = qss[jsed][ksed]/dt;

        switch(unitsQs)
        {
          case 1: /* Sediment discharge in m3/s */
            fprintf(sed_out_file_fptr,"%15.5f ", qsed[ils]);
            break;
          case 2: /* Sediment discharge in tons/ha/day */
            fprintf(sed_out_file_fptr,"%15.5f ",
                    qsed[ils] / areaQs[ils] * ROs * 86400);
            break;
        }
      }
      fprintf(sed_out_file_fptr,"\n");

      fprintf(debug,"%.2f \t %g \t %g \t %g \n",iter*dt/60,
        SedFluxOut[1][jout][kout],SedFluxOut[2][jout][kout],
        SedFluxOut[3][jout][kout]);
    }

      icount = 1;
  }

  else
  {
    icount = icount + 1;
  }

}
```

```
/*******************************/
/*        WriteGrids.c        */
/*******************************/

#include "all.h"

extern void WriteGrids()
{
  int j1,k1;

  float **WatDep;

  char wr_rname[80],wr_vinfname[80],wr_dname[80],
    wr_cname[80],wr_susname[80],wr_erosdepname[80],
    wr_sandfluxname[80],wr_siltfluxname[80],
    wr_clayfluxname[80],wr_totalfluxname[80],
    wr_sandMFACname[80],wr_siltMFACname[80],
    wr_clayMFACname[80],wr_totalMFACname[80];

  FILE *wr_dname_h = NULL;
  FILE *wr_erosdepname_h = NULL;
  FILE *wr_rname_h = NULL;
  FILE *wr_sandfluxname_h = NULL;
  FILE *wr_siltfluxname_h = NULL;
  FILE *wr_clayfluxname_h = NULL;
  FILE *wr_totalfluxname_h = NULL;
  FILE *wr_sandMFACname_h = NULL;
  FILE *wr_siltMFACname_h = NULL;
  FILE *wr_clayMFACname_h = NULL;
  FILE *wr_totalMFACname_h = NULL;
  FILE *wr_cname_h = NULL;
  FILE *wr_vinfname_h = NULL;
  FILE *wr_susname_h = NULL;

  WatDep = floatMemAlloc2d(m,n);


  if(ipcount == nplt)
  {
    printf("Writing Output Grids, IFCOUNT = %ld  Time = %.2f \n",
                                              ifcount,iter*dt/60.);

    /* SusC: grid holds the values of total suspended sediment    */
    /* WatDep: water depth in basin's overland or channel cells   */

    for(j1=1; j1<=m; j1++)
    {
      for(k1=1; k1<=n; k1++)
      {
        if (ishp[j1][k1] != nodatavalue)
        {
          if(ishp[j1][k1] == 1)
          {
            WatDep[j1][k1] = h[j1][k1];
          }
          else
          {
            WatDep[j1][k1] = hch[j1][k1];
          }
        }
      }
    }
```

```
      /* Writing time series grids                                    */

      /* Water depth in meters                                        */
      write2dTS(dname, wr_dname,wr_dname_h,WatDep,1);

      /* Rainfall intensity in mm/h                                   */
      write2dTS(rname, wr_rname,wr_rname_h,rint,3600000);

      /* Infiltrated volume in mm                                     */
      write2dTS(vinfname,wr_vinfname,wr_vinfname_h,vinf,1000);

      /* Suspended sediment concentration in m3/m3                    */
      write2dTS(cname,wr_cname,wr_cname_h,SusC,1);

      /* Total suspended sediment in mm                               */
      write2dTS(susname,wr_susname,wr_susname_h,totsus,1000/(w*w));

      /* Erosion and deposition in mm                                 */
      write2dTS(erosdepname,wr_erosdepname,wr_erosdepname_h,
                totnetv,1);

      /* Sediment fluxes are in (m3/s)                                */
      write3dTS(sandfluxname,wr_sandfluxname,wr_sandfluxname_h,
                1, SedFluxOut);
      write3dTS(siltfluxname,wr_siltfluxname,wr_siltfluxname_h,
                2, SedFluxOut);
      write3dTS(clayfluxname,wr_clayfluxname,wr_clayfluxname_h,
                3, SedFluxOut);
      write2dTS(totalfluxname,wr_totalfluxname,wr_totalfluxname_h,
                qss,(1/dt));

      /* Sediment concentrations are in m3 sed/ m3 mixture            */
      write3dTS(sandMFACname,wr_sandMFACname,wr_sandMFACname_h,
                1, MFAC);
      write3dTS(siltMFACname,wr_siltMFACname,wr_siltMFACname_h,
                2, MFAC);
      write3dTS(clayMFACname,wr_clayMFACname,wr_clayMFACname_h,
                3, MFAC);
      write2dTS(totalMFACname,wr_totalMFACname,wr_totalMFACname_h,
                MaxFluxCout,1);

      /* Incrementing counters                                        */

      ifcount++;
      if (iter == 1) ipcount++;
      if (iter != 1) ipcount = 1;
    }
    else
    {
      ipcount++;
    }

    /* Freeing memory                                                 */

    fMemFree2d(WatDep, m,n);
}
```

```
    /*********************************/
    /*          FUNCT: write2dTS      */
    /*********************************/

extern void write2dTS(char *name, char *wr_name,FILE *wr_name_h,
                float **ParamValue, float UnitsConv)
{
  int j1,k1;

  if(name[0] != '\0')
  {
    strncpy(wr_name,name,21);
    sprintf(wr_name,"%s.%1d",name,ifcount);
    if((wr_name_h=fopen(wr_name,"w"))==NULL)
    {
      printf("Can't open time-series file : %s \n",wr_name);
      exit(EXIT_FAILURE);
    }
    fprintf(wr_name_h,"%s",header);

    for(j1=1; j1<=m; j1++)
    {
      for(k1=1; k1<=n; k1++)
      {
        if (ishp[j1][k1] != nodatavalue)
        {
          fprintf(wr_name_h,"%10.5f ",ParamValue[j1][k1]*UnitsConv);
        }
        else
          fprintf(wr_name_h,"%i ",nodatavalue);
      }
      fprintf(wr_name_h,"\n");
    }

    fclose(wr_name_h);
  }
  return;

}
```

```
      /**********************************/
      /*          FUNCT: write3dTS       */
      /**********************************/

extern void write3dTS(char *name, char *wr_name,FILE *wr_name_h,
                int layer, float **ParamValue[])
{
  int j1,k1;

  if(name[0] != '\0')
  {
    strncpy(wr_name,name,21);
    sprintf(wr_name,"%s.%1d",name,ifcount);
    if((wr_name_h=fopen(wr_name,"w"))==NULL)
    {
      printf("Can't open time-series file : %s \n",wr_name);
      exit(EXIT_FAILURE);
    }
    fprintf(wr_name_h,"%s",header);

    for(j1=1; j1<=m; j1++)
    {
      for(k1=1; k1<=n; k1++)
      {
        if (ishp[j1][k1] != nodatavalue)
        {
          fprintf(wr_name_h,"%10.5f ",ParamValue[layer][j1][k1]);
        }
        else
          fprintf(wr_name_h,"%i ",nodatavalue);
      }
      fprintf(wr_name_h,"\n");
    }

    fclose(wr_name_h);
  }
  return;

}
```

```
/******************************/
/*          Reset.c          */
/******************************/

#include "all.h"

extern void Reset()
{
  int j,k,SizeFr;

  /* Resetting MaxFluxCout, qss, SedFluxOut and MFAC to 0.0        */

  for(j=1; j<=m; j++)
  {
    for(k=1; k<=n; k++)
    {
      MaxFluxCout[j][k] = 0.0;
      qss[j][k] = 0.0;

      for (SizeFr=1;SizeFr<=3;SizeFr++)
      {
        SedFluxOut[SizeFr][j][k] = 0.0;
        MFAC[SizeFr][j][k] = 0.0;
      }
    }
  }
}
```

```
/*******************************/
/*       WriteSummFlow.c      */
/*******************************/

#include "all.h"

extern void WriteSummFlow()
{

  double PercentFlow,InitialVol,FinalVol,InVolume,OutVolume;

  InitialVol = init_ch_vol   +  init_ov_vol;
  FinalVol   = final_ch_vol  +  final_ov_vol;
  InVolume   = InitialVol + vin;
  OutVolume = vinftot + vout + FinalVol;

  PercentFlow = 100.0 * (InVolume - OutVolume) / InVolume;

  /* ... summary of Output Flow */

  fprintf(Summ_file_fptr,
                            "\nSUMMARY OF FLOW OUTPUT\n");
  fprintf(Summ_file_fptr,"=====================\n\n");

  fprintf(Summ_file_fptr,
    "Peak Discharge (m3/s).......................................=");
  fprintf(Summ_file_fptr,"%15.2f \n", qpeak);

  fprintf(Summ_file_fptr,
    "Time to Peak (min)..........................................=");
  fprintf(Summ_file_fptr,"%15.2f \n",tpeak);

  fprintf(Summ_file_fptr,
    "Initial Surface Volume (m3).................................=");
  fprintf(Summ_file_fptr,"%15.2f \n", InitialVol);

  fprintf(Summ_file_fptr,
    "Volume of Rainfall - retention (m3)........................=");
  fprintf(Summ_file_fptr,"%15.2f \n", vin);

  fprintf(Summ_file_fptr,
    "Volume leaving the Watershed (m3)..........................=");
  fprintf(Summ_file_fptr,"%15.2f \n",vout);

  fprintf(Summ_file_fptr,
    "Percentage of Vout to Vin...................................=");
  fprintf(Summ_file_fptr,"%15.2f \n",(vout/vin)*100.0);

  fprintf(Summ_file_fptr,
    "Final Surface Volume (m3)...................................=");
  fprintf(Summ_file_fptr,"%15.2f \n", FinalVol);

  fprintf(Summ_file_fptr,
    "Volume Infiltrated (m3).....................................=");
  fprintf(Summ_file_fptr,"%15.2f \n",vinftot);

  fprintf(Summ_file_fptr,
    "Percentage of Vinftot to Vin...............................=");
  fprintf(Summ_file_fptr,"%15.2f \n", (vinftot/vin)*100.0);

  fprintf(Summ_file_fptr,
    "Percent Mass Balance...................................... =");
  fprintf(Summ_file_fptr,"%15.2f\n",PercentFlow);
```

```
  fprintf(Summ_file_fptr,
    "\nHYDROLOGICAL VARIABLES MINIMUM AND MAXIMUM VALUES\n");
  fprintf(Summ_file_fptr,
    "===============================================\n\n");

  fprintf(Summ_file_fptr,
    "Min. Rainfall Intensity (mm/hr)............................=");
  fprintf(Summ_file_fptr,"%15.2f \n", aminrain*3600.*1000);

  fprintf(Summ_file_fptr,
    "Max. Rainfall Intensity (mm/hr)............................=");
  fprintf(Summ_file_fptr,"%15.2f \n", amaxrain*3600.*1000);

  fprintf(Summ_file_fptr,
    "Min. Infiltration Depth (mm)...............................=");
  fprintf(Summ_file_fptr,"%15.2f \n", aminvinf*1000);

  fprintf(Summ_file_fptr,
    "Max. Infiltration Depth (mm)...............................=");
  fprintf(Summ_file_fptr,"%15.2f \n", amaxvinf*1000);

  fprintf(Summ_file_fptr,
    "Min. Overland Depth (m)....................................=");
  fprintf(Summ_file_fptr,"%15.2f \n",amindepth);

  fprintf(Summ_file_fptr,
    "Max. Overland Depth (m)....................................=");
  fprintf(Summ_file_fptr,"%15.2f \n",amaxdepth);

  fprintf(Summ_file_fptr,
    "Min. Channel Depth (m).....................................=");
  fprintf(Summ_file_fptr,"%15.2f \n", amincdepth);

  fprintf(Summ_file_fptr,
    "Max. Channel Depth (m).....................................=");
  fprintf(Summ_file_fptr,"%15.2f \n\n",amaxcdepth);

}
```

```
/*******************************/
/*         SedVolumes.c        */
/*******************************/

#include "all.h"

extern void SedVolumes()
{
  int j,k,SizeFr;

  for(j=1;j<=m;j++)
  {
    for(k=1;k<=n;k++)
    {
      if(ishp[j][k] != nodatavalue)
      {
        for(SizeFr =1;SizeFr<=3;SizeFr++)
         tot_eroded[SizeFr] += ssoil[SizeFr][j][k];

        if(ishp[j][k] == 1)
        {
          for(SizeFr =1;SizeFr<=3;SizeFr++)
          {
            sus_ov[SizeFr] += qovs[SizeFr][j][k];
            dep_ov[SizeFr] += vols[SizeFr][j][k];
            tot_ov[SizeFr] += qovs[SizeFr][j][k] +
                              vols[SizeFr][j][k];
          }
        }

        if(ishp[j][k] == 2)
        {
          for(SizeFr =1;SizeFr<=3;SizeFr++)
          {
            sus_ch[SizeFr] += qovs[SizeFr][j][k];
            dep_ch[SizeFr] += vols[SizeFr][j][k];
            tot_ch[SizeFr] += qovs[SizeFr][j][k] +
                              vols[SizeFr][j][k];
          }
        }
      }
    }
  }
}
```

```
/******************************/
/*        WriteSummSed.c     */
/******************************/

#include "all.h"

extern void WriteSummSed()
{

  double PercentSed,TotRemOvr,TotRemChn,TotEroded,TotLeaving,
        TotSusRemOvr,TotDepRemOvr,TotSusRemChn,TotDepRemChn;

  struct gstats totnetv_stats;

  totnetv_stats = gridstats(totnetv);

  TotEroded    = fabs(tot_eroded[1]+tot_eroded[2]+tot_eroded[3]);
  TotSusRemOvr = sus_ov[1] + sus_ov[2] + sus_ov[3];
  TotDepRemOvr = dep_ov[1]+dep_ov[2]+dep_ov[3];
  TotRemOvr    = tot_ov[1]+tot_ov[2]+tot_ov[3];
  TotSusRemChn = sus_ch[1]+sus_ch[2]+sus_ch[3];
  TotDepRemChn = dep_ch[1]+dep_ch[2]+dep_ch[3];
  TotRemChn    = tot_ch[1]+tot_ch[2]+tot_ch[3];
  TotLeaving   = sed_out[1]+sed_out[2]+sed_out[3];

  PercentSed = 100.0 * (TotLeaving + TotRemOvr + TotRemChn
                     - TotEroded) / TotEroded ;

  /* ... summary of output sediment                                    */

  fprintf(Summ_file_fptr,
    "\nSUMMARY OF SEDIMENT OUTPUT : Volume in Cubic Meters\n");
  fprintf(Summ_file_fptr,
    "(Percentages from the total eroded \n");
  fprintf(Summ_file_fptr,
    "==================================================\n\n");

  fprintf(Summ_file_fptr,
    "Total Volume of Sand Eroded..................................=");
  fprintf(Summ_file_fptr,"%15.2f \n",fabs(tot_eroded[1]));

  fprintf(Summ_file_fptr,
    "Total Volume of Silt Eroded..................................=");
  fprintf(Summ_file_fptr,"%15.2f \n",fabs(tot_eroded[2]));

  fprintf(Summ_file_fptr,
    "Total Volume of Clay Eroded..................................=");
  fprintf(Summ_file_fptr,"%15.2f \n",fabs(tot_eroded[3]));

  fprintf(Summ_file_fptr,
    "Total Volume of Material Eroded..............................=");
  fprintf(Summ_file_fptr,"%15.2f \n\n", TotEroded);


  fprintf(Summ_file_fptr,
        "Volumes of Eroded Sediment Remaining on the Overland:\n");

  fprintf(Summ_file_fptr,
    "------------------------------------------------------\n");

  fprintf(Summ_file_fptr,
    "Total Volume of Suspended Sand...............................=");
  fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
```

```
                    sus_ov[1],fabs(sus_ov[1]/tot_eroded[1]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Suspended Silt.............................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                    sus_ov[2],fabs(sus_ov[2]/tot_eroded[2]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Suspended Clay.............................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                    sus_ov[3],fabs(sus_ov[3]/tot_eroded[3]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Suspended Sediment.........................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n\n",
                  TotSusRemOvr,fabs(TotSusRemOvr/TotEroded*100));

fprintf(Summ_file_fptr,
  "Total Volume of Deposited Sand.............................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                    dep_ov[1],fabs(dep_ov[1]/tot_eroded[1]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Deposited Silt.............................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                    dep_ov[2],fabs(dep_ov[2]/tot_eroded[2]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Deposited Clay.............................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                    dep_ov[3],fabs(dep_ov[3]/tot_eroded[3]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Deposited Sediment.........................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n\n",
                  TotDepRemOvr,fabs(TotDepRemOvr/TotEroded*100));

fprintf(Summ_file_fptr,
  "Total Volume of Sand Remaining on the Overland.............=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                    tot_ov[1],fabs(tot_ov[1]/tot_eroded[1]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Silt Remaining on the Overland.............=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                    tot_ov[2],fabs(tot_ov[2]/tot_eroded[2]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Clay Remaining on the Overland.............=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                    tot_ov[3],fabs(tot_ov[3]/tot_eroded[3]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Eroded Material Remaining on the Overland..=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n\n",
                        TotRemOvr,fabs(TotRemOvr/TotEroded*100));

fprintf(Summ_file_fptr,
       "Volumes of Eroded Sediment Remaining in the Channels:\n");
fprintf(Summ_file_fptr,
        "-----------------------------------------------------\n");

fprintf(Summ_file_fptr,
  "Total Volume of Suspended Sand.............................=");
```

```
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                  sus_ch[1],fabs(sus_ch[1]/tot_eroded[1]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Suspended Silt.............................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                  sus_ch[2],fabs(sus_ch[2]/tot_eroded[2]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Suspended Clay.............................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                  sus_ch[3],fabs(sus_ch[3]/tot_eroded[3]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Suspended Sediment.........................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n\n",
                TotSusRemChn,fabs(TotSusRemChn/TotEroded*100));

fprintf(Summ_file_fptr,
  "Total Volume of Deposited Sand.............................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                 dep_ch[1],fabs(dep_ch[1]/tot_eroded[1]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Deposited Silt.............................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                  dep_ch[2],fabs(dep_ch[2]/tot_eroded[2]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Deposited Clay.............................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                  dep_ch[3],fabs(dep_ch[3]/tot_eroded[3]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Deposited Sediment.........................=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n\n",
              TotDepRemChn,fabs(TotDepRemChn/TotEroded*100));

fprintf(Summ_file_fptr,
  "Total Volume of Sand Remaining in the Channels.............=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                  tot_ch[1],fabs(tot_ch[1]/tot_eroded[1]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Silt Remaining in the Channels.............=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                  tot_ch[2],fabs(tot_ch[2]/tot_eroded[2]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Clay Remaining in the Channels.............=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                  tot_ch[3],fabs(tot_ch[3]/tot_eroded[3]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Eroded Material Remaining in the Channels..=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n\n",
                      TotRemChn,fabs(TotRemChn/TotEroded*100));

fprintf(Summ_file_fptr,
  "Volumes of Eroded Sediment Leaving the Watershed:\n");
fprintf(Summ_file_fptr,
  "-----------------------------------------------\n");

fprintf(Summ_file_fptr,
```

```
  "Total Volume of Eroded Sand Leaving the Watershed..........=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                 sed_out[1],fabs(sed_out[1]/tot_eroded[1]*100));

fprintf(Summ_file_fptr,
  "Total volume of Eroded Silt Leaving the Watershed..........=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                 sed_out[2],fabs(sed_out[2]/tot_eroded[2]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Eroded Clay Leaving the Watershed..........=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n",
                 sed_out[3],fabs(sed_out[3]/tot_eroded[3]*100));

fprintf(Summ_file_fptr,
  "Total Volume of Eroded Material Leaving the Watershed......=");
fprintf(Summ_file_fptr,"%15.2f -> %5.2f %% \n\n",
                     TotLeaving,fabs(TotLeaving/TotEroded*100));

fprintf(Summ_file_fptr,
  "Percent Mass Balance.......................................=");
fprintf(Summ_file_fptr,"%15.2f \n\n", PercentSed);


/* ... Minimum and Maximum Values for Various Variables         */

fprintf(Summ_file_fptr,
          "\nSEDIMENT VARIABLES MINIMUM AND MAXIMUM VALUES\n");
fprintf(Summ_file_fptr,
          "=============================================\n\n");

fprintf(Summ_file_fptr,
  "Max. Flux Conc. Overland (m3/s/m3/s).......................=");
fprintf(Summ_file_fptr,"\t%g \n", amaxFluxCoutOv);

fprintf(Summ_file_fptr,
  "Max. Flux Conc. Channels (m3/s/m3/s).......................=");
fprintf(Summ_file_fptr,"\t%g \n", amaxFluxCoutCh);

fprintf(Summ_file_fptr,
  "Max. Suspended Sediment in Overland (mm)...................=");
fprintf(Summ_file_fptr,"\t%g \n", amaxSusOv*1000/(w*w));

fprintf(Summ_file_fptr,
  "Max. Suspended Sediment in Channels (mm)...................=");
fprintf(Summ_file_fptr,"\t%g \n", amaxSusCh*1000/(w*w));

fprintf(Summ_file_fptr,
  "Max. Concentration on Overland (m3/m3).....................=");
fprintf(Summ_file_fptr,"\t%g \n", amaxSusCov);

fprintf(Summ_file_fptr,
  "Max. Concentration in Channels (m3/m3).....................=");
fprintf(Summ_file_fptr,"\t%g \n", amaxSusCch);

fprintf(Summ_file_fptr,
  "Total Net Volume (mm) at the End of the Simulation: \n");

fprintf(Summ_file_fptr,
  "    Minimum vValue ........................................=");
fprintf(Summ_file_fptr,"%15.3f \n", totnetv_stats.min );

fprintf(Summ_file_fptr,
  "    Maximum Value .........................................=");
```

```
  fprintf(Summ_file_fptr,"%15.3f \n",totnetv_stats.max );

  fprintf(Summ_file_fptr,
    "    Mean ...............................................=");
  fprintf(Summ_file_fptr,"%15.3f \n",totnetv_stats.mean );

  fprintf(Summ_file_fptr,
    "    Standard Deviation ..................................=");
  fprintf(Summ_file_fptr,"%15.3f \n",totnetv_stats.stdev );

}
```

```
/*******************************/
/*          MemFree.c          */
/*******************************/

#include "all.h"

extern void GridsMemFree()
{
  int SizeFr;

  /* Deallocate memory for input matrices                          */
  iMemFree2d(ishp,m,n);
  iMemFree2d(isoil,m,n);
  iMemFree2d(iman,m,n);
  fMemFree2d(e,m,n);
  fMemFree2d(ret,m,n);

  if(chancheck == 1)
  {
    iMemFree2d(link,m,n);
    iMemFree2d(node,m,n);
  }

  fMemFree2d(sdep,m,n);

  /* Deallocate memory for output matrices                         */

  if(indexinf == 1) fMemFree2d(vinf,m,n);

  fMemFree2d(h,m,n);
  fMemFree2d(hch,m,n);

  if(indexeros == 1)
  {
    fMemFree2d(qss,m,n);
    fMemFree2d(MaxFluxCout,m,n);
    fMemFree2d(totsus,m,n);
    fMemFree2d(SusC,m,n);
    fMemFree2d(totnetv,m,n);

    /* Deallocate memory for 3-D matrices                          */
    for (SizeFr=0;SizeFr<4;SizeFr++)
    {
      fMemFree2d(qovs[SizeFr],m,n);
      fMemFree2d(vols[SizeFr],m,n);
      fMemFree2d(ssoil[SizeFr],m,n);
      fMemFree2d(SedFluxOut[SizeFr],m,n);
      fMemFree2d(MFAC[SizeFr],m,n);
    }
  }

  fMemFree2d(dqov,m,n);
  fMemFree2d(dqch,m,n);
  fMemFree2d(rtot,m,n);
  fMemFree2d(rint,m,n);
}
```

```
      /**********************************/
      /*        FCT:  fMemFree2d        */
      /**********************************/

/* Deallocates memory for a floating 2D-grid of NoRows*NoCols     */
void fMemFree2d(float** matrix2d, int NoRows, int NoCols)
{
  int k;

  for(k=0; k<(NoRows+1); k++)
  {
    free(*(matrix2d + k));
  }

  free(matrix2d);
}

/* Deallocates memory for a integer 2D-grid of NoRows*NoCols      */
void iMemFree2d(int** matrix2d, int NoRows, int NoCols)
{
  int k;

  for(k=0; k<(NoRows+1); k++)
  {
    free(*(matrix2d + k));
  }

  free(matrix2d);
}
```

```
/*******************************/
/*          RunTime.c          */
/*******************************/

#include "all.h"

extern void RunTime(clock_t finish)
{
  double elapsedTime;

  fprintf(Summ_file_fptr,
    "\nProgram stops at simulation minute:%10.2f\n",(iter-1)*dt/60);

  elapsedTime = (double)(finish - startTime)/CLOCKS_PER_SEC;

  fprintf(Summ_file_fptr,
    "\nCASC2D RUNNING TIME:%10.2f minutes\n", elapsedTime/60);

  fclose(Summ_file_fptr);
}
```

```
          /*******************************/
          /*         Fcts_chann.c        */
          /*******************************/

#include "all.h"

     /*********************************/
     /*       FUNCT: newChnDepth      */
     /*********************************/

extern float newChnDepth(float wch,float dch,float sfactor,
                         int j, int k,float addedVolume)
{
  float area_ch,vol_ch,area_init,vol_init,
    vol_final,newdepth;

  /* Channel area and volume                                   */

  area_ch = wch * dch;
  vol_ch = area_ch * w * sfactor;

  /* Calculates initial area and volume                        */

  if(hch[j][k] <= dch)
   area_init =  wch * hch[j][k];
  else
   area_init = (hch[j][k] - dch) * w + area_ch;

  vol_init = area_init * w * sfactor;

  /* After adding new volume calculates volume                 */

  vol_final = vol_init + addedVolume;

  /* ... and depth corresponding to the final volume           */

  if(vol_final > vol_ch)
   newdepth = dch + (vol_final - vol_ch) / (w*w*sfactor);
  else
   newdepth = vol_final / (wch*w*sfactor);

  return(newdepth);

}
```

```
/**********************************/
/*        FUNCT: chnDischarge    */
/**********************************/

/* Calculates the outflow in m3/s from a cell */

float chnDischarge(float hchan,float hh,float wch,float dch,
          float stordep,float rmanch,float a,float sf,float sfactor)
{
  float area, wp, dQ, vol_ch_avail;

  /* Calculates flow area and wetted perimeter                    */

  if(hchan <= dch)
   {
     area = wch * hh;
     wp = (float)(wch + 2 * hh);
   }
   else
   {
     area = wch * (dch - stordep) + w * (hchan-dch);
     wp = (float)(wch+2*(dch-stordep) + 2*(w-wch) + 2*(hchan-dch));
   }

  dQ = (float)(a*(sqrt(fabs(sf))/rmanch)*
                        (pow(area,1.6667))/(pow(wp,0.6667)));

  /* Limit the outflow by availability                           */

  vol_ch_avail = area * w * sfactor;

  if(dQ*dt > vol_ch_avail) dQ = vol_ch_avail/dt;

  return(dQ);

}
```

```
                /*******************************/
                /*         Fcts_misc.c         */
                /*******************************/

#include "all.h"

      /**********************************/
      /*               FUNCT: max       */
      /**********************************/

/* Finds the maximum between two floating numbers */

extern float MAX(float one,float theOther)

{
  float maximum;

  if (one >= theOther)
    maximum = one;
  else
    maximum = theOther;

  return(maximum);
}

      /**********************************/
      /*               FUNCT: min       */
      /**********************************/

/* Finds the minimum between two floating numbers */

extern float MIN(float one,float theOther)

{
  float minimum;

  if (one <= theOther)
    minimum = one;
  else
    minimum = theOther;

  return(minimum);
}
```

```
        /*******************************/
        /*        Fcts_sed.c          */
        /*******************************/

#include "all.h"

     /**********************************/
     /*        FUNCT: EHcap            */
     /**********************************/

/* Determines the sediment transport capacity based on the          */
/* Engelund and Hansen transport equation. Result is in m3          */

extern float EHcap(int SizeFr,float width,float depth,
                       float sf,float discharge)
{
  float Cw[4], Rh,V, EHcapacity;

  Rh = (width*depth)/(width + 2 * depth);

  V = (float)(discharge/(width*depth));

  Cw[SizeFr] = (float)(0.05*(G/(G-1)) *
            fabs(V*sf)/sqrt((G-1)*g*ds[SizeFr]) *
            sqrt(Rh*fabs(sf)/((G-1)*ds[SizeFr]) ) ) ;

  EHcapacity = (float)(discharge * Cw[SizeFr] / 2.65)*dt ;

  return(EHcapacity);

}
     /**********************************/
     /*        FUNCT: KRcap            */
     /**********************************/

/* Determines the sediment transport capacity based on the          */
/* Kilinc and Richardson transport equation. Result is in m3        */

extern float KRcap(float KRcoeff,float discharge,float width,
                  float sf, int jfrom, int kfrom)
{
  int SoilType;
  float KRcapacity, UnitDischarge,Cusle,Kusle,Pusle;

  UnitDischarge = (float)(fabs(discharge/width));

  SoilType = isoil[jfrom][kfrom];

  Cusle = cfactor[iman[jfrom][kfrom]];
  Kusle = pinf[SoilType][7];
  Pusle = pfactor[iman[jfrom][kfrom]];

  KRcapacity = (float)(KRcoeff * pow(fabs(sf),1.66) *
            pow(UnitDischarge,2.035) * Cusle * Kusle * Pusle
            * width * dt);

  return(KRcapacity);

}
```

```
/***********************************/
/*         FUNCT: TransferSed      */
/***********************************/

/* Transfers a certain volume of sediment, VolToTransfer, from     */
/* a outgoing cell (jfrom,kfrom) to receiving cell (jto,kto). The  */
/* source, can be the suspended,deposited or parent material in    */
/* outgoing cell. The transferred volume of sed. will be added     */
/* in all cases onto the suspended portion of the receiving cell.  */
/* If the outgoing cell is the outlet cell, the VolToTransfer is   */
/* not added into a receiving cell but to the variable that holds  */
/* the total volume of sediment leaving the basin by size fraction*/

extern void TransferSed(int SizeFr,float **Source[],
                        float VolToTransfer,
                        int jfrom,int kfrom,int jto,int kto)
{
  Source[SizeFr][jfrom][kfrom] -= VolToTransfer;

  if(jfrom==jout && kfrom==kout)
  {
    sed_out[SizeFr] += VolToTransfer;
  }
  else
  {
    qovs[SizeFr][jto][kto] += VolToTransfer;
  }
}

/***********************************/
/*         FUNCT: FindMFAC         */
/***********************************/


extern void findMFAC(int jfrom,int kfrom,float totQSout,
                        float discharge,float qs[4])
{
  int SizeFr;

  if(MaxFluxCout[jfrom][kfrom] < totQSout/dt/fabs(discharge)
           &&   fabs(discharge) > 1E-4)
  {
    MaxFluxCout[jfrom][kfrom] =
                             (float)(totQSout/dt/fabs(discharge));

    for (SizeFr = 1; SizeFr <=3; SizeFr++)
    {
      MFAC[SizeFr][jfrom][kfrom] =
                          (float)(qs[SizeFr]/dt/fabs(discharge));
    }
  }
}
```

```
            /*******************************/
            /*         Fcts_stats.c        */
            /*******************************/

#include "all.h"

      /**********************************/
      /*          FUNCT: gridstats      */
      /**********************************/

/* Determines the mean and standard deviation of a grid           */

extern struct gstats gridstats(float **inputgrid)

{
  struct gstats statistics;
  float s;
  int j,k;

  statistics.min = 10E300;
  statistics.max = -10E300;
  statistics.mean = 0;
  s = 0;

  for(j=1;j<=m;j++)
  {
    for(k=1;k<=n;k++)
    {
      if(ishp[j][k] != nodatavalue)
      {
        statistics.min = MIN(statistics.min,inputgrid[j][k]);
        statistics.max = MAX(statistics.max,inputgrid[j][k]);
        statistics.mean += inputgrid[j][k] / ncells;
      }
    }
  }
  for(j=1;j<=m;j++)
  {
    for(k=1;k<=n;k++)
    {
      if(ishp[j][k] != nodatavalue)
      {
        s +=
        (float)(pow(inputgrid[j][k]-statistics.mean,2)/(ncells-1));
      }
    }
  }

  statistics.stdev = (float)(sqrt(s));

  return(statistics);
}
```

I.3 CASC2D-SED INPUT/OUTPUT FILES

Model input and output files are defined in a Project file. The project file name represents the argument of the CASC2D-SED program. The list of input/output files and their definition is found in Table I-1, Table I-2 and Table I-3 below. An example of a project file is found in Appendix III.

I.3.1 Input Files

Required CASC2D inputs and options are:

(1) Simulation control data: computational time step, simulation duration, and number of rows and columns and cell size of the grid.

(2) Precipitation data: precipitation duration and a) if uniform rainfall, the intensity or b) number of raingages, rainfall data time step, and rainfall gages UTM coordinates if distributed rainfall.

(3) Basin outlet characteristics: a) row, column and overland slope of the basin at the outlet and b) link characteristics if channel routing is selected: (link number, total nodes, type, width, depth, side slope, roughness and sinuosity)

(4) Soil parameters: number of different soil types, and a) if infiltration occurs: hydraulic conductivity, suction head, and moisture deficit and b) if erosion option is selected: percentages of sand, silt and clay in soil and USLE soil erodibility factor.

(5) Land use parameters: number of different soil uses, roughness coefficient and interception depth and, if erosion option is selected: USLE cover and management practice factors.

The input files are: the simulation control file (specifying parameters defined in the last 5 paragraphs), channel data file, and precipitation file (if distributed rainfall is used). Input as raster ASCII maps are: DEM, soil index, land use index, watershed mask, and channel links and nodes (if channel routing is selected). The storage depth ASCII grid is an optional input grid.

Examples of input files can be found in Appendix III.

I.3.2 Output Files

CASC2D-SED creates the next output files:

(1) Summary file: contains information on the simulation parameters, channel topology inferred from the link and node maps and the final results of the hydrological and erosion components.

(2) Flow and sediment flow files: containing the water flow [m3/s] and sediment flow [m3/s] at the outlet. Results can optionally be written at other internal locations inside the watershed.

(3) Time-series grids: They are optional and represent simulation values at predetermined times. Output raster maps include: water depth, rainfall rates, accumulated infiltrated depth, total sediment volume, suspended sediment volume, and sediment flux and concentration by size fraction. Time series maps can be later used as input in an animated file.

Examples of CASC2D-SED output files can be found in Appendix III.

Table I-1 CASC2D-SED input and output text files

	Project name	File variable	Description
Input Text Files	CONTROL	control_file	Controls the simulation.
	PRECIP	rain_file	Precipitation data if gages are present [in/hr]
	CHANNEL	chn_file	Channel characteristics data for each link and node.
Output Text Files	SUMMARY	Summ_file	Simulation summary file
	DISCHARGE_OUT	dis_out_file	Hydrograph at outlet and, if selected, at internal locations [m^3/s]
	SEDIMENT_OUT	sed_out_file	Sedigraph at outlet and, if selected, at internal locations [m^3/s]

Table I-2 CASC2D-SED input grids

Project name	File variable	Variable Read / Displayed	Description
ELEVATION	elev_file	e[j][k]	Grid elevation at (j,k) [m]
SOIL	soil_file	isoil[j][k]	Soil type index at (j,k)
LANDUSE	iman_file	iman[j][k]	Land use / land cover index at (j,k)
MASK	shap_file	ishp[j][k]	Watershed and channel network definition at (j,k)
LINK	link_file	link[j][k]	Link number at (j,k)
NODE	node_file	node[j][k]	Node number at (j,k)
STORAGE_DEPTH	sdep_file	sdep[j][k]	Storage deposition at (j,k) [m]

Table I-3 CASC2D-SED time-series output grids

Project name	File variable	Variable Read / Displayed	Description
RAINFALL	rname	rint[j][k]	Rainfall rates at (j,k) [in/hr]
INF_DEPTH	vinfname	vinf[j][k]	Volume of infiltrated water at (j,k) [m]
WATER_DEPTH	dname	h[j][k] or hch[j][k]	Depth of water (channels and overland) at (j,k) [m]
SUS_SED	susname	totsus[j][k]	Total volume of suspended sediment at (j,k) [mm]
SED_CONC	cname	Σqovs[SizeFr][j][k] / (water volume)	Total suspended sediment concentration at (j,k) [m^3/m^3]
EROS_DEP	erosdepname	totnetv[j][k]	Total net erosion at (j,k) [mm]
SAND_FLUX_OUT	sandfluxname	SedFluxOut[1][j][k]	Total sand flux out of (j,k) [m^3/s]
SILT_FLUX_OUT	siltfluxname	SedFluxOut[2][j][k]	Total silt flux out of (j,k) [m^3/s]
CLAY_FLUX_OUT	clayfluxname	SedFluxOut[3][j][k]	Total clay flux out of (j,k) [m^3/s]
TOTAL_FLUX_OUT	totalfluxname	qss[j][k]/dt	Total sediment flux out of (j,k) [m^3/s]
SAND_MFAC_OUT	sandMFACname	MFAC[1][j][k]	Max. flux-averaged concentration of sand out of (j,k) [m^3/s / m^3/s]
SILT_MFAC_OUT	siltMFACname	MFAC[2][j][k]	Max. flux-averaged concentration of silt out of (j,k) [m^3/s / m^3/s]
CLAY_MFAC_OUT	clayMFACname	MFAC[3][j][k]	Max. flux-averaged concentration of clay out of (j,k) [m^3/s / m^3/s]
TOTAL_MFAC_OUT	totalMFACname	MaxFluxCout[j][k]	Max. flux-averaged concentration of sediment out of (j,k) [m^3/s / m^3/s]

I.4 Time-Series Grids Display

I.4.1 AML programs

```
/* ######################################################################
/* #################               DS.AML           #####################
/* ######################################################################

/* ====================================================================
/* PURPOSE:
/* Animates a time series from Arc Info ASCII grids created by
/* the CASC2D-SED hydrological model.  The user has the possibility
/* of creating an animated MPEG file (movie) from the ArcPlot
/* display window by creating a screenshot of each display.
/* In general, this AML can be used to create a movie from any
/* time series of Arc Info ASCII grids if these ASCII grids have
/* the same Prefix (root) name and the Sufix (file extension) in
/* an ordered sequence of arabic numbers.
/*
/* ====================================================================
/* INPUT
/* 1. A time series of Arc Info ASCII grids:
/*    Example 1: one map per frame and n number of frames are shown
/*      depth.1; depth.2; ....; depth.n;
/*    Example 2: four maps per frame and n number of frames are shown
/*      depth.1; depth.2; ....; depth.n;
/*      erosion.1; erosion.2; ....erosion.n;
/*      flow.1; flow.2; ....; flow.n;
/*      rain.1; rain.2; ....; rain.n;
/* 2. A parameter file that will control the display of the grids or
/*  map composition.  Optionally, this parameter file might not be
/*  given but the user needs to enter data manually.
/*
/* ====================================================================
/* OUTPUT:
/*   1. A series of Arc Info grids (if user doesn't opt to kill them)
/*      after making the movie)
/*  Example: User chooses to show every other grid in his input
/*    series, starting in grid 6 and ending in grid 10:
/*    depth6; depth8; depth10
/*   2. An MPEG file (if user decides to create a movie using the
/*      generated grids)
/*
/* ====================================================================
/* USAGE:
/* Arc: &run ds <ParameterFile>
/*
/* ====================================================================
/* ARGUMENTS:
/*  ParamFile: Name of the input data file. If not specified,
/*       the user must enter the input data interactively.
/*
/* ====================================================================
```

```
/* VARIABLES:
/*   Globals:
/*  .CreateMovie: Index: 1: Movie is created; 0: not created
/*  .ElapsedTime: Elapsed time (secs) between CASC2D generated maps
/*  .EndFrame: Frame in which the display or movie will end
/*  .FrameStep: Number of frames to skip when displaying series
/*  .iter: Counter used to name multiple sequential images
/*  .MapTitleXX: Title for each of the displayed maps
/*  .MovieName: Name given to the final MPEG file (movie)
/*  .NameXX: Prefix (root) given to each of the thematic series
/*  .RMTXX: Remap table for each of the thematic series
/*  .SecsPause: Seconds to halt the series display
/*  .ShdSetXX: Shade set for each of the thematic series
/*  .StartFrame: Frame in which the display or movie will end
/*  .wPath: workspace location.  Enter full path if the workspace
/*        is in another location
/*
/* ==================================================================
/* CALLS:
/*  APdisplay.aml;
/*
/* ==================================================================
/* ROUTINES:
/*   GetUserInputData -- Prompts the user to enter input data
/*   GetFileInputData -- Input data is read from a file
/*   CreateGrids      --
/*      1. Creates the grids from the ArcInfo
/*          ASCII files if they don't exists.
/*          They will be named as: PrefixSufix
/*          Example: depth1; depth2; ....depthn;
/*      2. Writes grid stats. in text files
/*    DisplayGrids    -- Displays the series of grids in an Arc Plot
/*          display window according to the specified
/*          remap table and shadesets defined previously
/*          by the user.
/*   Animate          -- Executes the MPEG encoder
/*   KillAllGrids     -- Created grids can be cleaned once the screen
/*          shot has been taken if the user decides so.
/*   GetUserSpecs     -- Get the user's ArcPlot configuration
/*   PutUserSpecs     -- Restores the user's AP configuration
/*   bailout          -- bails of of aml
/* ==================================================================
/* NOTES:
/* Must have the Berkeley MPEG encoder in:
/*                                  $ARCHOME\bin\mpeg_encode.exe
/*
/* ==================================================================
/* HISTORY:
/* 9 Aug 1996 Original coding - Ian DeMerchant
/*   (as mpeg_encode.aml in ArcTools)
/* 15 Nov 2000 Major revision - Stephen Lead (slead@esriau.com.au)
/*   (as movie.aml)
/* 15 Jul 2001 Revision  -- Rosalia Rojas (CSU)
/*      (as DisplaySeries.aml)
/*  This AML has been rewriten for handling different options.
/*  The movie can display 1, 2, 3, 4 or 6 maps per frame
/*  User is able to define the remap table and color scheme
```

```
/*  for each of the maps in the frame.
/*  User is able to define a starting and ending frame and
/*  show them in a predefined interval.
/* ==============================================================


&ARGS DSParamFile

&IF [NULL %DSParamFile%] &THEN
  &CALL GetUserInputData
&ELSE
  &CALL GetFileInputData
&CALL CheckInputErrors
&CALL CreateGrids

/* IN CASE THAT A MOVIE IS NOT TO BE CREATED, THE USER IS GIVEN THE
/* POSIBILITY OF DISPLAYING THE GRIDS AS MANY TIMES AS (S)HE WISHES

&SV ShowSeriesAgain = .FALSE.

&do &until ^ %ShowSeriesAgain%
  &CALL DisplayGrids
  &IF %.CreateMovie% = 0 &THEN
    &DO
      &TYPE
      &TYPE !!! Hello there !!!
      &SV ShowSeriesAgain = ~
      [QUERY 'Do you want to display again the time series' .FALSE.]

      /* Include the next line in case that you changed your mind
      /* after displaying the series (ex. you decide to kill grids)
      &IF  %ShowSeriesAgain% &THEN &CALL GetFileInputData
      &TYPE
    &END
&end

&IF %.CreateMovie% = 1 &THEN ;  &CALL Animate

&CALL KillAllGrids

&RETURN


/* ################################################################
/* ###########   ROUTINE GET_USER_INPUT_DATA     ##################
/* ################################################################

&ROUTINE GetUserInputData

/* GET THE NUMBER OF MAPS PER FRAME */

&TYPE
&SV .MapsPerFrame = ~
               [RESPONSE 'How many maps per frame? 1,2,3,4 or 6?']
&TYPE
```

```
/* DEFINE WHICH FRAMES WILL BE SHOWN

&SV .StartFrame = [RESPONSE 'Start movie in frame number']
&TYPE

&SV .EndFrame =   [RESPONSE 'End movie in frame number']
&TYPE

&SV .ElapsedTime =  ~
                 [RESPONSE 'Elapsed show time between frames (sec.)']
&TYPE

&SV .FrameStep =  [RESPONSE 'Show frame step']
&TYPE

&SV .SecsPause = [RESONSE 'Seconds to pause frames display in AP']

/* Ask for the location of the folder where the ASCII grids are
&SV .wPath =  [RESPONSE 'ASCII grids location (full path)']
&TYPE

/* GET THE NAMES OF THOSE MAPS

 &DO MapNumber = 1 &TO %.MapsPerFrame%
   &sv .Name%MapNumber% = ~
       [RESPONSE 'Name (prefix) of map no. '%MapNumber% :']
 &END
 &TYPE

/* ASK THE USER TO ENTER THE REMAP TABLE NAME FOR EACH OF
/* THE MAPS (S)HE WANTS TO DISPLAY

 &DO MapNum = 1 &TO %.MapsPerFrame%
   &sv .RMT%MapNum% = ~
    [RESPONSE 'Remap table name for the '[value Name%MapNum%]' map']
 &END
 &TYPE

/* ASK THE USER TO ENTER THE SHADESET FILE NAME FOR EACH OF
/* THE MAPS (S)HE WANTS TO DISPLAY

 &DO MapNo = 1 &TO %.MapsPerFrame%
   &sv .ShdSet%MapNo% = ~
    [RESPONSE 'Shadeset file name for the '[value Name%MapNo%]' map']
 &END
 &TYPE

/* ASK THE USER TO ENTER THE TITLE TO DISPLAY FOR EACH OF
/* THE MAPS (S)HE WANTS TO DISPLAY

 &DO MapNo = 1 &TO %.MapsPerFrame%
   &sv .MapTitle%MapNo% = ~
    [RESPONSE 'Title to display for the '[value Name%MapNo%]' map']
 &END
 &TYPE

/* ASK WHETHER THE USER WANTS TO CLEAN THE WORKSPACE
```

```
/* AFTER THE MOVIE IS DONE.

&SV OrderToKill = ~
   [RESPONSE 'Kill grids after they are displayed (1:yes;0:no) ?']
&TYPE

/* SCARE THE USER IF (S)HE WANTS TO CREATE THE MPEG FILE
  &TYPE
  &TYPE WARNING !!!!
  &TYPE Be aware that creating a movie will take for about
  &TYPE a minute per frame.  Make sure that this time series
  &TYPE is what you want to be displayed.
  &TYPE
  &TYPE
  &TYPE

&SV .CreateMovie = ~
   [RESPONSE 'would you like to create a movie (1:yes;0:no) ?']
&TYPE

&IF %CreateMovie% = 1 &THEN
    &SV .MovieName = [RESPONSE 'Movie name']
&TYPE

/*&SV .3Drendering = ~
/*    [RESPONSE 'would you like a 3-D representation (1:yes;0:no) ?']
/*&TYPE

/*&SV .SurfaceModel = ~
/*    [RESPONSE 'Name of the surface model (include path) ?']
/*&TYPE

/*&SV .Zscale = ~
/*    [RESPONSE 'Vertical scale (>1 exaggeration; <1 reduccion):']
/*&TYPE

&RETURN


/* ######################################################################
/* ############    ROUTINE GET_FILE_INPUT_DATA    ######################
/* ######################################################################

&ROUTINE GetFileInputData

&SV AMLunit = [open %DSParamFile% openstat -read]

/* First four lines are comments
&SV line1 = [read %AMLunit% readstat]  /* comment line
&SV line2 = [read %AMLunit% readstat]  /* comment line
&SV line3 = [read %AMLunit% readstat]  /* comment line
&SV line4 = [read %AMLunit% readstat]  /* comment line
&TYPE

&SV line5 = [unquote [read %AMLunit% readstat]]
&SV .MapsPerFrame = [extract 2 %line5%]
&TYPE Maps Per Frame: %.MapsPerFrame%
```

```
&TYPE

&SV line6 = [unquote [read %AMLunit% readstat]]
&SV .StartFrame = [extract 2 %line6%]
&TYPE Starting Frame = %.StartFrame%
&TYPE

&SV line7 = [unquote [read %AMLunit% readstat]]
&SV  .EndFrame = [extract 2 %line7%]
&TYPE Ending Frame = %.EndFrame%
&TYPE

&SV line8 = [unquote [read %AMLunit% readstat]]
&SV  .ElapsedTime = [extract 2 %line8%]
&TYPE Elapsed time between frames = %.ElapsedTime%
&TYPE

&SV line9 = [unquote [read %AMLunit% readstat]]
&SV  .FrameStep = [extract 2 %line9%]
&TYPE Frame Step = %.FrameStep%
&TYPE

&SV line10 = [unquote [read %AMLunit% readstat]]
&SV  .SecsPause = [extract 2 %line10%]
&TYPE Seconds to pause frame display in AP = %.SecsPause%
&TYPE

&SV line11 = [read %AMLunit% readstat]  /* blank line

&SV line12 = [UNQUOTE [READ %AMLunit% readstat] ]
&SV  .wPath = [extract 2 %line12%]

&TYPE Time series grids location = %.wPath%
&TYPE

&SV line13 = [unquote [read %AMLunit% readstat]]
&DO MapNumber = 1 &TO %.MapsPerFrame%
   &SV .Name%MapNumber% = ~
       [EXTRACT [CALC %MapNumber% + 1] %line13%]
   &TYPE Map No. %MapNumber%: [VALUE .Name%MapNumber%]
&END
&TYPE

&SV line14 = [unquote [read %AMLunit% readstat]]
&DO MapNumber = 1 &TO %.MapsPerFrame%
   &sv .RMT%MapNumber% = ~
       [EXTRACT [CALC %MapNumber% + 1] %line14%]
   &TYPE Map No. %MapNumber% Remap Table: [VALUE .RMT%MapNumber%]
&END
&TYPE

&SV line15 = [unquote [read %AMLunit% readstat]]
&DO MapNumber = 1 &TO %.MapsPerFrame%
   &sv .ShdSet%MapNumber% = ~
       [EXTRACT [CALC %MapNumber% + 1] %line15%]
   &TYPE Map No. %MapNumber% shadeset: [VALUE .ShdSet%MapNumber%]
&END
```

```
&TYPE

&SV line16 = [unquote [read %AMLunit% readstat]]
&DO MapNumber = 1 &TO %.MapsPerFrame%
   &sv .MapTitle%MapNumber% = ~
        [EXTRACT [CALC %MapNumber% + 1] %line16%]
   &TYPE Map No. %MapNumber% title: [VALUE .MapTitle%MapNumber%]
&END
&TYPE

&SV line17 = [unquote [read %AMLunit% readstat]]
&DO MapNumber = 1 &TO %.MapsPerFrame%
   &sv .MapSubTitle%MapNumber% = ~
        [EXTRACT [CALC %MapNumber% + 1] %line17%]
   &TYPE Map No. %MapNumber% subtitle: [VALUE .MapSubTitle%MapNumber%]
&END
&TYPE

&SV line18 = [unquote [read %AMLunit% readstat]]
&SV .OrderToKill = [extract 2 %line18%]
&IF %.OrderToKill% = 1 &THEN; &TYPE Grids deleted after displayed
&ELSE &TYPE Generated grids will not be deleted
&TYPE

&SV line19 = [unquote [read %AMLunit% readstat]]
&SV .CreateMovie = [extract 2 %line19%]
&IF %.CreateMovie% = 1 &THEN; &TYPE An MPEG file will be created
&ELSE &TYPE No MPEG file will be created
&TYPE

&SV line20 = [unquote [read %AMLunit% readstat]]
&SV .MovieName = [extract 2 %line20%]
&TYPE Movie Name = %.MovieName%
&TYPE

&SV closestat = [CLOSE %AMLunit%]

&RETURN

/* #################################################################
/* ############      ROUTINE CREATE_GRIDS         ###################
/* #################################################################

&routine CreateGrids

/* OPEN AND WRITE TITLES IN THE FILES THAT WILL HOLD THE GRIDS STATS.

&DO i = 1 &TO %.MapsPerFrame%
  &SV filexists = [EXIST %.wPath%[value .Name%i%]_desc.txt -FILE]
  &IF %filexists% &THEN
  &SV NULL [DELETE %.wPath%[value .Name%i%]_desc.txt]
  &SV fileunit%i% ~
        [open %.wPath%[value .Name%i%]_desc.txt openstat -WRITE]
  &SV NULL [WRITE [value fileunit%i%] ~
  'GridName        Min.          Max.          Mean          StdDev']
  &SV NULL [WRITE [value fileunit%i%] ~
  '---------------------------------------------------------------']
```

```
&END

&FORMAT 3

/* THE INPUT ASCII GRIDS (FROM CASC2D_SED) ARE CALLED
/*      ex. depth.1, depth.2, AND SO ON
/* GRIDS ARE GOING TO BE CALLED ex. depth1, depth2 AND SO ON

/* IF OLD GRIDS ARE PRESENT, THEY WILL BE REUSED, OTHERWISE, THEY
/* WILL BE CREATED.  MAKE SURE THAT THOSE OLD GRIDS ARE THE ONES YOU
/* WANT TO USE, OTHERWISE, KILL THEM BEFORE STARTING THE CURRENT AML

&DO FrameNumber = %.StartFrame% &TO %.EndFrame% &BY %.FrameStep%

     &TYPE Frame Number: %FrameNumber%

     &DO MapNumber = 1 &TO %.MapsPerFrame%
       &SV String := %.wPath%[value  .Name%MapNumber%]
       &SV gridexist = [EXISTS %String%%FrameNumber% -grid]
       &IF %gridexist% eq .FALSE. &THEN
          &DO
             &TYPE Creating grid
             ASCIIGRID ~
                  %String%.%FrameNumber% %String%%FrameNumber% FLOAT
             &TYPE Calculating statistics
             &DESCRIBE %String%%FrameNumber%
             &SV DataLine := ~
              [FORMAT '%1,10% %2,-15% %3,-15% %4,-15% %5,-15%' ~
              %GRD$GRID% %GRD$ZMIN% %GRD$ZMAX% %GRD$MEAN% %GRD$STDV%]
             &SV NULL [WRITE [value fileunit%MapNumber%] %DataLine%]
             /*&CALL ComputeStats
          &END
     &END
&END

&DO i = 1 &TO %.MapsPerFrame%    /* Closing the stats. files
   &SV NULL [CLOSE [value fileunit%i%]]
&END
&RETURN

/* ####################################################################
/* ###########        ROUTINE DISPLAY_GRIDS        ##################
/* ####################################################################

&ROUTINE DisplayGrids

/* THE VARIABLE iter WAS ORIGINALLY INSIDE THE animate.aml
/* WE INITIALIZE IT OUTSIDE THE LOOP THAT CREATES THE GRIDS

&SV .iter -1

AP

&CALL GetUserSpecs

&SV timesec 0  /* Initialize total time to zero before displaying
```

```
&FORMAT 1

/*&IF %.MapsPerFrame% = 1 &THEN
/*  &RUN DS1.AML
/*&ELSE
  &RUN APdisplay.aml

&CALL PutUserSpecs

QUIT

&RETURN

/* ####################################################################
/* ############          ROUTINE ANIMATE         ###################
/* ####################################################################

/* THIS PART OF THE CODE IS A MODIFIED COPY OF THE animate
/* routine WRITEN BY STEPHEN LEAD IN THE animate.aml

&ROUTINE animate

/* TO CREATE THE ANIMATED MPEG FILE, WE USE THE BERKELEY MPEG ENCODER
/* WITH THE DEFAULT MPEG PARAMETER FILE DEFINED IN THIS AML.

/* THE USER CAN CHANGE THESE PARAMETER VALUES TO HIS/HER LIKING
/* ACCORDING TO THE USERS-GUIDE.PS CREATED AT THE BERKELEY MULTIMEDIA
/* RESEARCH CENTER, WHICH CAN BE FOUND AT:
/* http://bmrc.berkeley.edu/frame/research/mpeg/mpeg_encode.html

/* A GOOD EXPLANATION AND EXAMPLE OF THE MPEG PARAMETER FILE IS FOUND
/* AT: http://www.math.arizona.edu/swig/animation/MPEG/(by Mark Hays)

   /* Calculate the first and last images - the temporary files
   /* contain a prefix with 5 trailing digits. We need to substitute
   /* zeros for any leading blank values.

   &s first 00000
   &s last %.iter%
     &do i = 1 &to [calc 5 - [length %.iter%]]
       &s last 0%last%
     &end

    /* Write out a parameter file for this animation.
    &s param_file [scratchname -file]
    &s fileunit [open %param_file% openstat -write]
    &if %openstat% <> 0 &then
      &do
      &s str Could not create %param_file%...; &call bailout
      &end

    &s null [write %fileunit% 'PATTERN IBBPBBPBBPBB']
    &s null [write %fileunit% 'INPUT']
    &setchar &function '|*' '*|'
    &s null |* write %fileunit% ~
                        |* quote %.MovieName%* [%first%-%last%] *| *|
    &setchar &function [ ]
```

```
    &s null [write %fileunit% 'END_INPUT']
    &s null [write %fileunit% [quote OUTPUT %.wPath%%.MovieName%]]
    &s null [write %fileunit% [quote INPUT_DIR [show work]]]
    &s null [write %fileunit% ~
      [quote INPUT_CONVERT [joinfile [joinfile ~
      [pathname $ARCHOME] bin -SUB] rasttopnm -FILE] *]]
    &s null [write %fileunit% 'BASE_FILE_FORMAT PNM']
    &s null [write %fileunit% 'SLICES_PER_FRAME 1']
    &s null [write %fileunit% 'PIXEL HALF']
    &s null [write %fileunit% 'RANGE 9']
    &s null [write %fileunit% 'PSEARCH_ALG TWOLEVEL']
    &s null [write %fileunit% 'BSEARCH_ALG CROSS2']
    &s null [write %fileunit% 'GOP_SIZE 12']
    &s null [write %fileunit% 'PQSCALE 10']
    &s null [write %fileunit% 'IQSCALE 10']
    &s null [write %fileunit% 'BQSCALE 10']
    &s null [write %fileunit% 'REFERENCE_FRAME ORIGINAL']
    &s null [write %fileunit% 'FORCE_ENCODE_LAST_FRAME']
    &s null [close %fileunit%]

    /* EXECUTE THE ENCODER
    &sys [joinfile [joinfile [pathname $ARCHOME] bin -SUB] ~
          mpeg_encode -FILE] %param_file%

   /* DELETE THE TEMPORARY FILES
   &s deletestat [DELETE %param_file% -file]
      &DO i = 0 &TO %.iter%
      &SV value %i%
         &DO a = 1 &TO [calc 5 - [LENGTH %i%]]
           &S value 0%value%
           &S delstat [DELETE %.MovieName%%value% -file]
         &END
      &END

&RETURN

/* ##################################################################
/* ############       ROUTINE KILL_ALL_GRIDS       ##################
/* ##################################################################
/* KILLS THE INPUT GRIDS IF THE USER DECIDED SO

&ROUTINE KillAllGrids
&IF %.OrderToKill% = 1 &THEN

  &DO FrameNumber = %.StartFrame% &TO %.EndFrame% &BY %.FrameStep%

    &type Killing Grids Number: %FrameNumber%

    &DO MapNumber = 1 &TO %.MapsPerFrame%
      &sv String := %.wPath%[value  .Name%MapNumber%]
      &sv gridexist = [exists %String%%FrameNumber% -grid]
      &if %gridexist% &then
      KILL %String%%FrameNumber% ALL
    &END

  &END
&RETURN
```

```
/* ######################################################################
/* ############          ROUTINE GET_USER_SPECS      ##################
/* ######################################################################

/* DETERMINES THE USERS SPECIFICATIONS BEFORE IMPLEMENTING NEW ONES
/* WITH THIS AML

&ROUTINE GetUserSpecs

   /* GET THE NUMBER OF DECIMALS
   &SV UsersDecimals [SHOW &FORMAT]
   &TYPE UsersDecimals %UsersDecimals%

   /* GET THE DISPLAY WINDOW SPECIFICATION
   &SV UsersAPdisplay [SHOW DISPLAY]
   &TYPE UsersAPdisplay %UsersAPdisplay%

   /* GET THE SHADESET
   &SV UsersShadeSet [SHOW SHADESET]
   &TYPE UsersShadeSet %UsersShadeSet%

   /* GET THE TEXT SPECIFICATIONS
   &SV UserTextFont [SHOW TEXTFONT]
   &SV UserTextQuality [SHOW TEXTQUALITY]
   &SV UserTextSpacing [SHOW TEXTSPACING]
   &SV UserTextSize [SHOW TEXTSIZE]

&RETURN


/* ######################################################################
/* ############          ROUTINE PUT_USER_SPECS      ##################
/* ######################################################################

/* RESTORES THE USER'S SPECIFICATIONS

&ROUTINE PutUserSpecs

   /* RESTORE THE USER'S NUMBER OF DECIMALS
   &FORMAT %UsersDecimals%

   /* RESTORE THE USER'S DISPLAY WINDOW SPECIFICATION
   DISPLAY %UsersAPdisplay%

   /* RESTORE THE USER'S SHADESET
   SHADESET %UsersShadeSet%

   /* RESTORE THE USER'S TEXT SPECIFICATIONS
   TEXTFONT %UserTextFont%
   TEXTQUALITY %UserTextQuality%
   TEXTSPACING %UserTextSpacing%
   TEXTSIZE %UserTextSize%


&RETURN
```

```
/* ####################################################################
/* ##########    ROUTINE CHECK_INPUT_ERRORS         ################
/* ####################################################################

&ROUTINE CheckInputErrors

/* TODO: this is just one of the errors.  I should check for the rest
&IF %.MapsPerFrame% NE 1 AND  ~
    %.MapsPerFrame% NE 2 AND ~
    %.MapsPerFrame% NE 3 AND ~
    %.MapsPerFrame% NE 4 AND ~
    %.MapsPerFrame% NE 6 &THEN
  &DO
    &TYPE WARNING !!!!
    &TYPE You have chosen to display %.MapsPerFrame% grids per frame
    &TYPE Please, Choose to display 1, 2, 3, 4 or 6 grids per frame
    &TYPE
    &CALL bailout
  &END

&RETURN

/* ####################################################################
/* ############         ROUTINE BAIL_OUT          ###################
/* ####################################################################

&ROUTINE bailout
   &IF NOT [variable str] &THEN &S str Bailing out of %aml$file%
   &RETURN; &RETURN &WARNING %str%
   &TYPE
&RETURN
```

```
/* ######################################################################
/* ############          APDISPLAY.AML          ####################
/* ######################################################################

/* ======================================================================
/* PURPOSE:
/* Displays different thematic map series in AP.  Legends are
/* displayed, with the key files being generated automatically if
/* they are not created yet.
/* Titles for each thematic series can be shown.  Time is also
/* displayed.  This time is computed from the elapsed time between
/* CASC2D maps, and the map number in the series.
/*
/* ======================================================================
/* ARGUMENTS:
/*  None
/*
/* ======================================================================
/* VARIABLES:
/*  Passed from the main program (DS.aml) as global
/*  variables.
/*
/* ======================================================================
/* CALLS:
/*  TakeScreenShot.aml
/*
/* ======================================================================
/* ROUTINES:
/*   Legend    -- Creates key files and displays map legends
/*      according to the given remap table and shadeset
/*   TitleText -- Writes map titles (specified in the Parameter file)
/*   TimeText  -- Writes the event simulation time
/*   SetLayoutVariables -- Specifies position of legend, title, sub-
/*      title, simulation time and maps in a given layout
/*
/* ======================================================================
/* HISTORY:
/*   August 18, 2001:  This part of the code is taken from the main
/*  program (DisplaySeries.aml) and becomes a stand alone aml.
/*  November 16, 2001: Display2.aml, Display3.aml....Display6.aml
/*  are re-writen more generically in only one AML (Display.aml).
/*
/* ======================================================================

/* Displays a red line around the AP window
LINECOLOR 2
LINESIZE 0.05

/* I want a white background.  Instead of changing the canvas color

/* I will just put a white patch as background
SHADETYPE COLOR
SHADECOLOR 1
&SV CanvasColor = [SHOW CANVASCOLOR]
&TYPE Canvas Color is %CanvasColor%
```

```
/* SPECIFY ARCPLOT DISPLAY CONFIGURATION
&IF %.MapsPerFrame% EQ 1 OR %.MapsPerFrame% EQ 4 &THEN
  &DO
    DISPLAY 9999 SIZE CANVAS 720 600      /* Landscape
    PAGESIZE 11 8.5                       /* Landscape
    /* &IF %CanvasColor% EQ BLACK &THEN PATCH 0 0 11 8.5
    BOX 0 0 11 8.5                        /* Landscape
  &END
&ELSE
  &DO
    DISPLAY 9999 SIZE CANVAS 530 720      /* Portrait
    PAGESIZE 8.5 11                       /* Portrait
    /*&IF %CanvasColor% EQ BLACK &THEN PATCH 0 0 8.5 11
    BOX 0 0 8.5 11                        /* Portrait
  &END

/* LEGEND AND TITLES WILL NOT CHANGE ALONG THE SIMULATION SO THEY ARE
/* DISPLAYED ONLY ONCE AT THE BEGINING (they won't flicker every time
/* the display is refreshed and the display will be faster)

/* Set variables depending on the number of maps per frame
&CALL SetLayoutVariables

/* Creates Legends and Titles in layout
&CALL Legend
&CALL TitleText

/* DISPLAY THEMATIC MAP SERIES

&DO FrameNumber = %.StartFrame% &TO %.EndFrame% &BY %.FrameStep%

  GNDS TRANSPARENT  /* NODATA cells will be displayed transparently.
                    /* that I can see the map legend

  &SV timesec [calc %FrameNumber% * %.ElapsedTime%]
  &SV timemin [calc %timesec% / 60]
  &CALL TimeText

  &TYPE Displaying frame Number: %FrameNumber%

  &DO n = 1 &TO %.MapsPerFrame%
    MAPLIMITS [UNQUOTE[VALUE MLimits%n%]]
    MAPEXTENT %.wPath%[VALUE .Name%n%]%FrameNumber%
    MAPPOSITION LR LR
    SHADEDELETE ALL
    SHADESET [VALUE .ShdSet%n%]
    GRIDSHADES ~
          %.wPath%[VALUE .Name%n%]%FrameNumber% VALUE [VALUE .RMT%n%]
    LINECOLOR 2
    LINESIZE 0.04
    /*ARCS h:\gismaps\3hydromodel\30\drainagedata\basins_polys
  &END

  &PAUSE &SECONDS %.SecsPause%

  &IF %.CreateMovie% = 1 &THEN &DO
```

```
    &RUN TakeScreenShot
    &TYPE Done: Screen shot of frame Number: %FrameNumber%
  &END

&END

&RETURN

/* ######################################################################
/* ############       ROUTINE LEGEND             ###################
/* ######################################################################

/* CREATES FILES TO STORE THE KEY FILES.
/* KEY FILES ARE DERIVED FROM REMAP TABLES

&ROUTINE Legend

&DO k = 1 &TO %.MapsPerFrame%

  &SV fileunit%k% [OPEN %.wPath%[value .Name%k%].key openstat -write]

  &IF %openstat% <> 0 &THEN
    &DO
       &SV str Could not create key_file ...; &CALL bailout
    &END

  /* DATA WILL BE TAKEN FROM THE REMAP TABLE (DEFINED BY THE USER)
  /* AND WRITEN TO THE TEMPORAL KEY FILE WITH THE CORRESPONDING
  /* KEY FILE FORMAT (see "Drawing Key Legend" help)

  &SV eof 0

  &SV RemapFile%k% [open [value .RMT%k%] openstat -read]
  &SV line [UNQUOTE [READ [value RemapFile%k%] eof]]

  /* I ASSUME THAT THE USER DOESN'T HAVE MORE THAN 100 CLASSES!!!
  /* I WILL ONLY WRITE THE UPPER LIMIT AS THE TEXT CLOSE TO THE KEY
  /* IF YOU WANT TO WRITE THE WHOLE INTERVAL --> CHANGE NEXT LINES
  /* OF CODE TO READ AND WRITE THE 1st AND 2nd ARGUMENT OF %line%

  &DO i = 1 &TO 100 &WHILE %eof% NE 102
     /* 102 is returned value for EOF
     &SV NULL [WRITE [VALUE fileunit%k%] .%i% ]
       &SV key%i% := [QUOTE < [EXTRACT 2 %line%]]
     &SV NULL [WRITE [VALUE fileunit%k%] [value key%i%]]
     &SV line [UNQUOTE [READ [value RemapFile%k%] eof]]
  &END

  &SV NULL [CLOSE [VALUE fileunit%k%]]

  &SV closestat = [CLOSE [VALUE RemapFile%k%]]
&END


/* TODO:CHANGE THE NEXT LINES WITH THE FONTS AND SIZES OF YOUR LIKING
/* NEXT SPECIFICATIONS WILL BE COMMON FOR ALL LEGENDS
```

```
/* Standandard hardware device color index:
/* 1:white; 2:red; 3:green; 4:blue; 5:cyan; 6:magenta; 7:yellow

SHADETYPE COLOR
TEXTCOLOR 1               /* Legend text color: black
TEXTFONT 'Times Bold'     /* Legend Font:smooth panel
TEXTSIZE 0.23 0.23        /* Legend text size: .2 inches
TEXTSPACING 0
TEXTQUALITY PROPORTIONAL
LINESIZE 0.01             /* Line width around key box
LINECOLOR 0               /* Line color (black) around key box
KEYBOX .25 .18            /* Key box width and height (inches)
KEYSEPARATION 0.05 0.03 /* Key dist. to text and between keys

&DO j = 1 &TO %.MapsPerFrame%
  /*BOX [UNQUOTE[VALUE Area%j%]]
  SHADEDELETE ALL
  KEYAREA [UNQUOTE [value Area%j%]]
  SHADESET [value .ShdSet%j%]
  KEYSHADE %.wPath%[value .Name%j%].key
  &SV NULL [DELETE %.wPath%[value .Name%j%].key]  /* Delete key file
&END

&RETURN

/* ######################################################################
/* ############        ROUTINE TITLE_TEXT          ##################
/* ######################################################################

/* CREATES THE TITLES AND SUBTITLES CORRESPONDING TO EACH OF THE
/* THEMATIC SERIES -- TITLES ARE USER DEFINED IN THE PARAMETER FILE

&ROUTINE TitleText

TEXTCOLOR 2               /* Title in red
TEXTSIZE 0.4 0.4          /* Title font height & width: 0.4 inches
TEXTFONT 'Courier bold'   /*
TEXTSPACING 0
TEXTQUALITY PROPORTIONAL

/* MOVE: specifies a coordinate point used to position the cursor and
/* to position text drawn with TEXT and TEXTFILE. Specify only the
/* lower left corner of the title and subtitle block.

&DO m = 1 &TO %.MapsPerFrame%

  MOVE [VALUE xTitlePos%m%] [VALUE yTitlePos%m% ]
  TEXT [value .MapTitle%m%]
  MOVE [VALUE xTitlePos%m%] [CALC [VALUE yTitlePos%m%] - 0.4]
  TEXT [value .MapSubTitle%m%]
&END

&RETURN
```

```
/* ####################################################################
/* ############        ROUTINE TIME_TEXT         ####################
/* ####################################################################

/* CALCULATES SIMULATION TIME AND DISPLAYS IT ON THE LAYOUT

&ROUTINE TimeText

/* Needed to put a black box on top of the time so that the last
/* time is not shown (the display window is not being refreshed )

SHADETYPE COLOR
SHADECOLOR 0

/* (xmin, ymin) has been specified in the SetVariables routine
&SV xmax = [CALC %xmin% + 2]
&SV ymax = [CALC %ymin% + 0.55]

PATCH %xmin% %ymin% %xmax% %ymax%

MOVE [CALC %xmin% + 0.25] [CALC %ymin% + 0.25]

TEXTSIZE 0.40 0.40
TEXTSPACING 0
TEXTQUALITY PROPORTIONAL
TEXTFONT 'omega bold'
TEXT 't =  '
TEXT [QUOTE [format '%1%' %timemin%]]
MOVE [CALC %xmin% + 1.8] [CALC %ymin% + 0.25]
TEXTFONT 'times'            /* ?????????????????????
TEXT ' min.'                /* How to write the whole thing
                            /* with only one TEXT command ?
&RETURN

/* ####################################################################
/* ############        ROUTINE SET_VARIABLES     ####################
/* ####################################################################

&ROUTINE SetLayoutVariables

&IF %.MapsPerFrame% EQ 1 &THEN
  &DO
    &SV MLimits1 = ' 0 0 11 8.5 '
    &SV Area1 = '0.2 5.0 4.3 8.4'
    &SV xTitlePos1 = 3.5
    &SV yTitlePos1 = 7.35
    &SV xmin = 6.75
    &SV ymin = 0.75
  &END
```

```
&IF %.MapsPerFrame% EQ 2 &THEN          /*  ____________
  &DO                                   /*  |            |
    &SV MLimits1 = '0.1 5.4 8.4 10.9'   /*  |     1      |
    &SV MLimits2 = '0.1 0.1 8.4 5.5'    /*  |____________|
    &SV Area1 = '0.1 8.9 2.85 10.65'    /*  |            |
    &SV Area2 = '0.1 3.6 2.85 5.35'     /*  |     2      |
    &SV xTitlePos1 = 2.4                /*  |____________|
    &SV yTitlePos1 = 10.6
    &SV xTitlePos2 = %xTitlePos1%
    &SV yTitlePos2 = [CALC %yTitlePos1% - 5.3 ]
    &SV xmin = 4.20
    &SV ymin = 0.1
  &END

/* this one needs more work. Check values
&IF %.MapsPerFrame% EQ 3 &THEN
  &DO
    &SV MLimits1 = '0.0 7.4 8.5 11.0'
    &SV MLimits2 = '0.0 3.7 8.5 7.3'
    &SV MLimits3 = '0.0 0.0 8.5 3.6'
    &SV Area1 = '0.1 8.9 2.85 10.65'
    &SV Area2 = '0.1 3.6 2.85 5.35'
    &SV Area3 = '0.1 3.6 2.85 5.35'
    &SV xTitlePos1 = 5.75       /*  ____________
    &SV yTitlePos1 = 0.9        /*  |            |
    &SV xTitlePos2 = 5.75       /*  |     1      |
    &SV yTitlePos2 = 6.3        /*  |____________|
    &SV xTitlePos3 = 6.3        /*  |            |
    &SV yTitlePos3 = 5          /*  |     2      |
    &SV xmin = 2.20             /*  |____________|
    &SV ymin = 0.1              /*  |            |
  &END                          /*  |     3      |
                                /*  |____________|
&IF %.MapsPerFrame% EQ 4 &THEN
  &DO
    &SV MLimits1 = '0 4.25 5.5 8.5'
    &SV MLimits2 = '5.5 4.25 11 8.5'
    &SV MLimits3 = '0 0 5.5 4.25'
    &SV MLimits4 = '5.5 0 11 4.25'
    &SV Area1 = '0.05 6.95 2.05 8.45'
    &SV Area2 = '5.40 6.95 7.40 8.45'
    &SV Area3 = '0.05 2.75 2.05 4.25'
    &SV Area4 = '5.40 2.75 7.40 4.25'
    &SV xTitlePos1 = 2.70
    &SV yTitlePos1 = 8.05       /*  _________________________
    &SV xTitlePos2 = 8.10       /*  |            |            |
    &SV yTitlePos2 = 8.05       /*  |     1      |     2      |
    &SV xTitlePos3 = 2.70       /*  |____________|____________|
    &SV yTitlePos3 = 3.85       /*  |            |            |
    &SV xTitlePos4 = 8.10       /*  |     3      |     4      |
    &SV yTitlePos4 = 3.85       /*  |____________|____________|
    &SV xmin = 3.0
    &SV ymin = 0.05
  &END
```

```
&IF %.MapsPerFrame% EQ 6 &THEN
  &DO
    &SV MLimits1 = '0.0 7.2 4.2 10.8'  /*Landscape: 0 4.2 3.6 8.4
    &SV MLimits2 = '4.2 7.2 8.4 10.8'  /*Landscape: 3.6 4.2 7.2 8.4
    &SV MLimits3 = '0.0 3.6 4.2 7.2'   /*Landscape: 7.2 4.2 10.8 8.4
    &SV MLimits4 = '4.2 3.6 8.4 7.2'   /*Landscape: 0 0 3.6 4.2
    &SV MLimits5 = '0.0 0.0 4.2 3.6'   /*Landscape: 3.6 0 7.2 4.2
    &SV MLimits6 = '4.2 0.0 8.4 3.6'   /*Landscape: 7.2 0 10.8 4.2
    &SV Area1 = '0.05 9.15 1.65 10.85'
    &SV Area2 = '4.2 9.15 5.8 10.85'
    &SV Area3 = '0.05 5.55 1.65 7.25'
    &SV Area4 = '4.2 5.55 5.8 7.25'
    &SV Area5 = '0.05 1.95 1.65 3.65'
    &SV Area6 = '4.2 1.95 5.8 3.65'
    &SV xTitlePos1 = 2          /*  _______________________
    &SV yTitlePos1 = 10.2       /* |           |           |
    &SV xTitlePos2 = 6.15       /* |     1     |     2     |
    &SV yTitlePos2 = 10.2       /* |___________|___________|
    &SV xTitlePos3 = 2          /* |           |           |
    &SV yTitlePos3 = 6.6        /* |     3     |     4     |
    &SV xTitlePos4 = 6.15       /* |___________|___________|
    &SV yTitlePos4 = 6.6        /* |           |           |
    &SV xTitlePos5 = 2          /* |     5     |     6     |
    &SV yTitlePos5 = 3.0        /* |___________|___________|
    &SV xTitlePos6 = 6.15
    &SV yTitlePos6 = 3.0
    &SV xmin = 2.20
    &SV ymin = 0.05
  &END

&RETURN
```

```
/* ######################################################################
/* ############       ROUTINE TAKE_SCREEN_SHOT     ###################
/* ######################################################################

/* THIS PART OF THE CODE IS A PARTIAL COPY OF THE createimages
/* routine WRITEN BY STEPHEN LEAD IN THE animate.aml

/* IMPORTANT: I HAD TO CHANGE MY SCREEN DISPLAY TO 'TRUE COLORS'
/* (NOT 256 OR 65536 COLORS)

/* CREATE A TEMPORARY SCREENSHOT IMAGE, CONVERT TO SUNRASTER.
/* OMIT THE CONVERSION STAGE IF YOU ARE NOT USING WINDOWS NT.

&SV tempimage [scratchname -file]

/* screensave command:IN WINDOWS NT THE IMAGE IS SAVED AS A BMP FILE
/* UNLESS OTHERWISE SPECIFIED.  SINCE JPEG IS ALREADY A COMPRESSED
/* IMAGE, AND THIS IMAGE WILL BE FURTHER COMPRESSED IN THE MPEG
/* ENCODER (THUS, LOOSING RESOLUTION) WE SAVE THE FILES AS SUNRASTER

SCREENSAVE %tempimage%
&SV rsimage [scratchname -file]
ARC CONVERTIMAGE %tempimage% %rsimage% sunraster none
&SV delstat [delete %tempimage% -file]
&SV delstat [delete %tempimage%w -file]

/* NOTE: ANOTHER WAY TO CREATE AN ANIMATED FILE IS TO CREATE GIF
/* OR JPEG FILES AND TO USE A PROGRAM TO CREATE AN ANIMATED GIF.
/* ADVANTAGE: EASY TO USE, VERY VERSATIL. DISADVANTAGE: SOFTWARE
/* NOT FOR FREE. A GOOD STARTING POINT TO LEARN ABOUT ANIMATED GIFs
/* AND COMPARE SOFTWARE: http://www.webreference.com/dev/gifanim/
/* IN CASE THAT YOU DECIDE TO CREATE AN ANIMATED GIF,YOU CAN END
/* THE AML HERE.  NO NEED TO EXECUTE THE animate SUBROUTINE.

/* COPY THIS IMAGE A NUMBER OF TIMES TO PROVIDE MULTIPLE FRAMES,
/* ELSE THE IMAGE ONLY APPEARS FOR A SPLIT SECOND IN THE MPEG FILE.
/* SETTING THE APPROXIMATE NUMBER OF FRAMES PER SECOND TO 40 FRAMES

/* SEEMS TO RESULT IN EACH IMAGE LASTING ABOUT 1 SECOND.
/* THE LARGER NUMBER OF repeats --> THE MOVIE DOES NOT GO SO FAST
/* BUT THE TIME IT TAKES TO CREATE THE MOVIE IS BIGGER AND THE
/* MPEG FILE WILL ALSO BE LARGER

&SV repeats 4

&TYPE Copying the image 4 times

&do a = 1 &to %repeats%
  &SV .iter [calc %.iter% + 1]
  &SV image%.iter% [scratchname -prefix %.MovieName% -file]
  &SV copystat [copy %rsimage% [value image%.iter%] -file]
  &if %copystat% > 0 &then
          &do
             &s str Error copying image file...; &call bailout
          &end
&end
```

```
/* DELETE TEMPORARY FILES
&s delstat [delete %rsimage% -file]
&s delstat [delete %rsimage%w -file]


&RETURN

/* ######################################################################
/* ############        ROUTINE BAIL_OUT        ###################
/* ######################################################################


&ROUTINE bailout

   &IF NOT [variable str] &THEN &S str Bailing out of %aml$file%
   &RETURN; &RETURN &WARNING %str%

&RETURN
```

I.4.2 DS.aml parameter file example

```
# DISPLAYSERIES.AML INPUT FILE
# Please, use this file as a template.
# Leave at least one blank space between arguments

No.Maps_Per_Frame_(1;4;6):                4
Start_In_Frame_No.:                       306
End_In_Frame_No:                          307
Elapsed_Time_Between_Frames(sec):         60
Show_By_Frame_Step:                       1
Seconds_To_Pause_Frames_Display_InAP:     5

Map(s)_Location:  h:\simulations\distribprecip\17oct81\30m\hydromaps\
Map(s)_Name(s):   rain infvol depth  conc
Rema(s)_Table(s): rain2.rmp infvol.rmp depth.rmp concl4.rmp
Shade(s)_Set(s):  rain.shd infvol.shd depth.shd concl4.shd
Map_Title(s):     'Rainfall' 'Infiltr.' 'Depth' '      Cv'
Map_Subtitle(s):  '   [mm/h]' '     [mm]'    '    [m]' '    [m3/m3]'
Kill_Grids_after_Displayed(1:yes;0:no):   0
Create_Movie(1:yes;0:no):                 0
Movie_Name:                               HydroSumm30.mpeg
```

APPENDIX II

II.1 Goodwin Creek DEM, Land Use, Soil Type, and Watershed Definition at Different Spatial Resolution ---------- II-2

Figure II.1.1 DEM, watershed definition, soil type, and land use at 30-m ---------- II-2
Figure II.1.2 DEM, watershed definition, soil type, and land use at 90-m ---------- II-3
Figure II.1.3 DEM, watershed definition, soil type, and land use at 150-m ---------- II-4
Figure II.1.4 DEM, watershed definition, soil type, and land use at 210-m ---------- II-5
Figure II.1.5 DEM, watershed definition, soil type, and land use at 270-m ---------- II-6
Figure II.1.6 DEM, watershed definition, soil type, and land use at 330-m ---------- II-7

II.2 Percentage of the Area Covered by each Soil Type for each Grid Size and Sub-basin ---------- II-8

Figure II.2.1 Percentage of the area covered by each soil type in sub-basin 4 ---------- II-8
Figure II.2.2 Percentage of the area covered by each soil type in sub-basin 6 ---------- II-8
Figure II.2.3 Percentage of the area covered by each soil type in sub-basin 7 ---------- II-9
Figure II.2.4 Percentage of the area covered by each soil type in sub-basin 8 ---------- II-9
Figure II.2.5 Percentage of the area covered by each soil type in sub-basin 14 --------- II-10

II.3 Percentage of the Area Covered by each Soil Type for each Grid Size and Sub-basin ---------- II-11

Figure II.3.1 Percentage of the area covered by each land use type in sub-basin 4 ----- II-11
Figure II.3.2 Percentage of the area covered by each land use type in sub-basin 6 ----- II-11
Figure II.3.3 Percentage of the area covered by each land use type in sub-basin 7 ----- II-12
Figure II.3.4 Percentage of the area covered by each land use type in sub-basin 7 ----- II-12
Figure II.3.5 Percentage of the area covered by each land use type in sub-basin 14---- II-13

II.1 Goodwin Creek DEM, Land Use, Soil Type, and Watershed Definition at Different Spatial Resolution

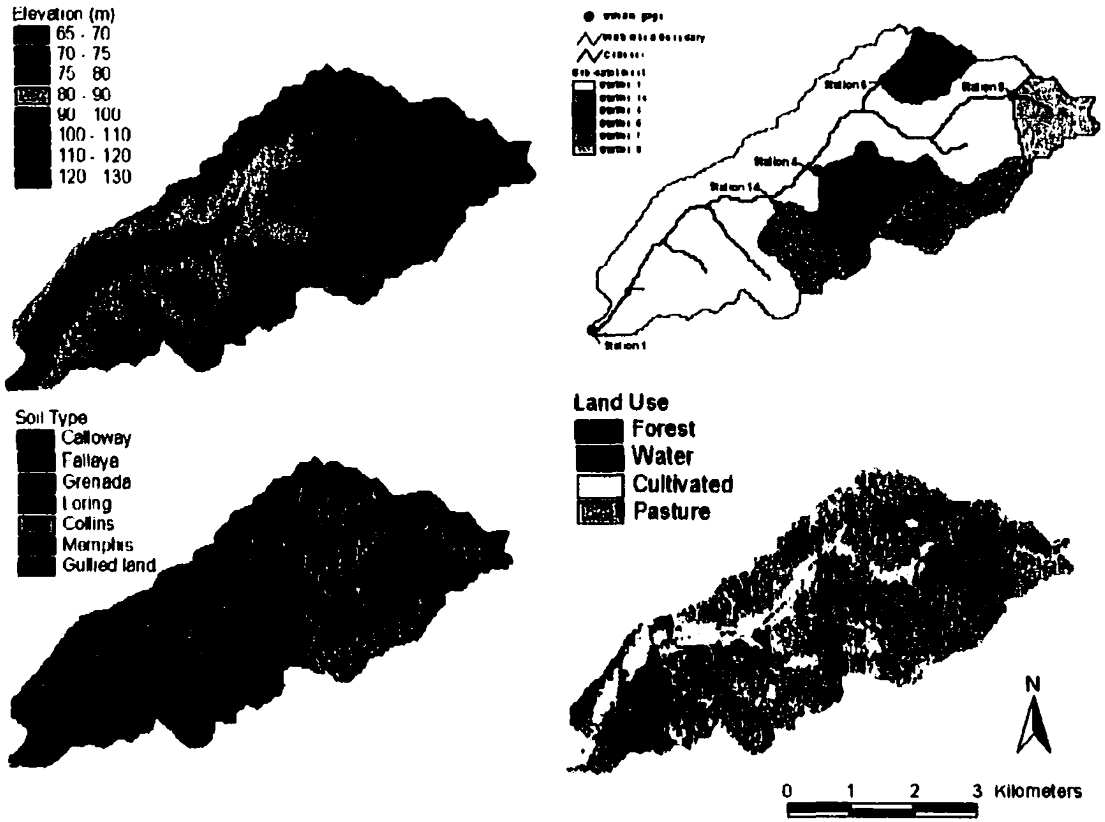


Figure II.1.1 DEM, watershed definition, soil type, and land use at 30-m

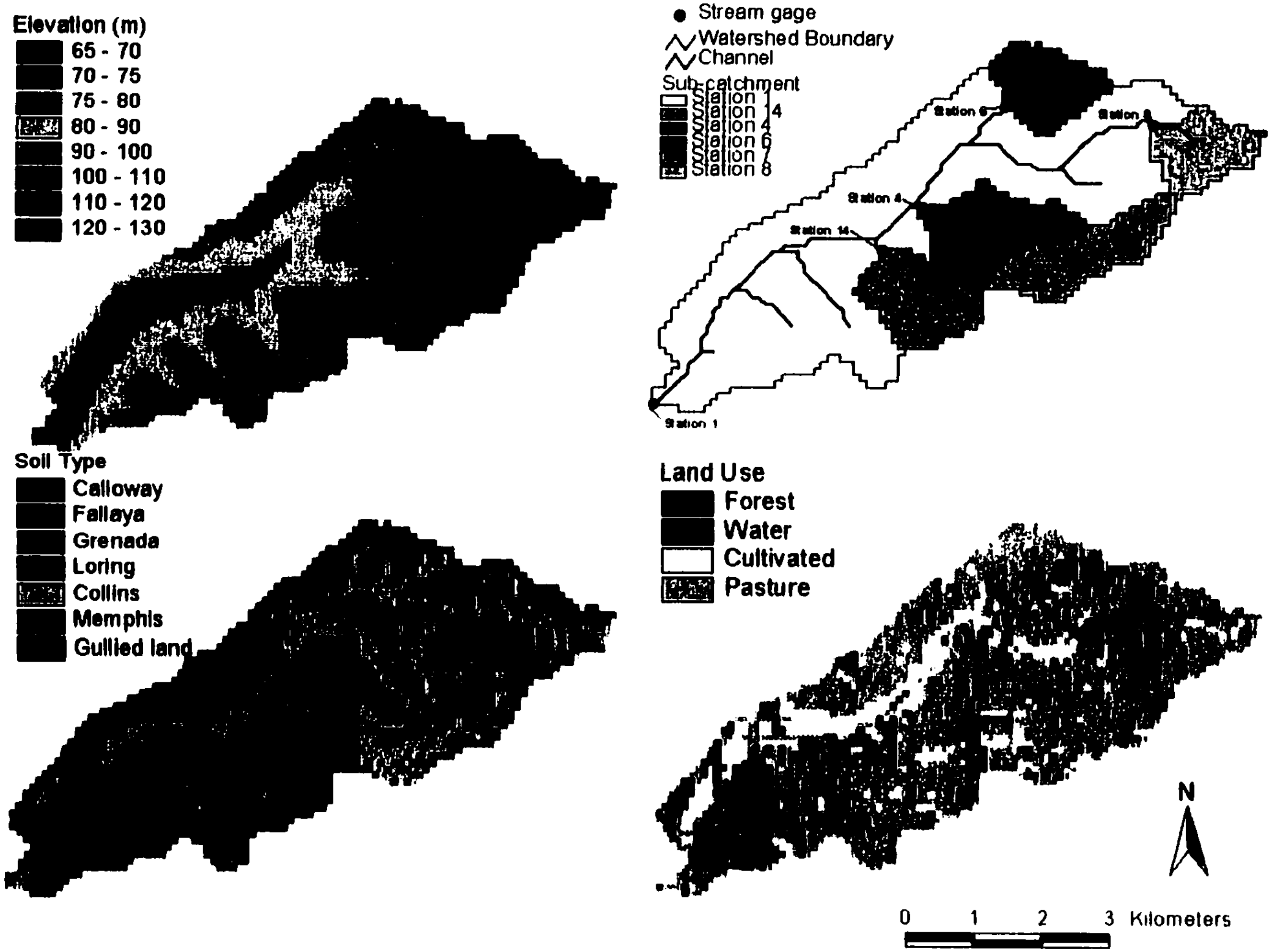


Figure II.1.2 DEM, watershed definition, soil type, and land use at 90-m

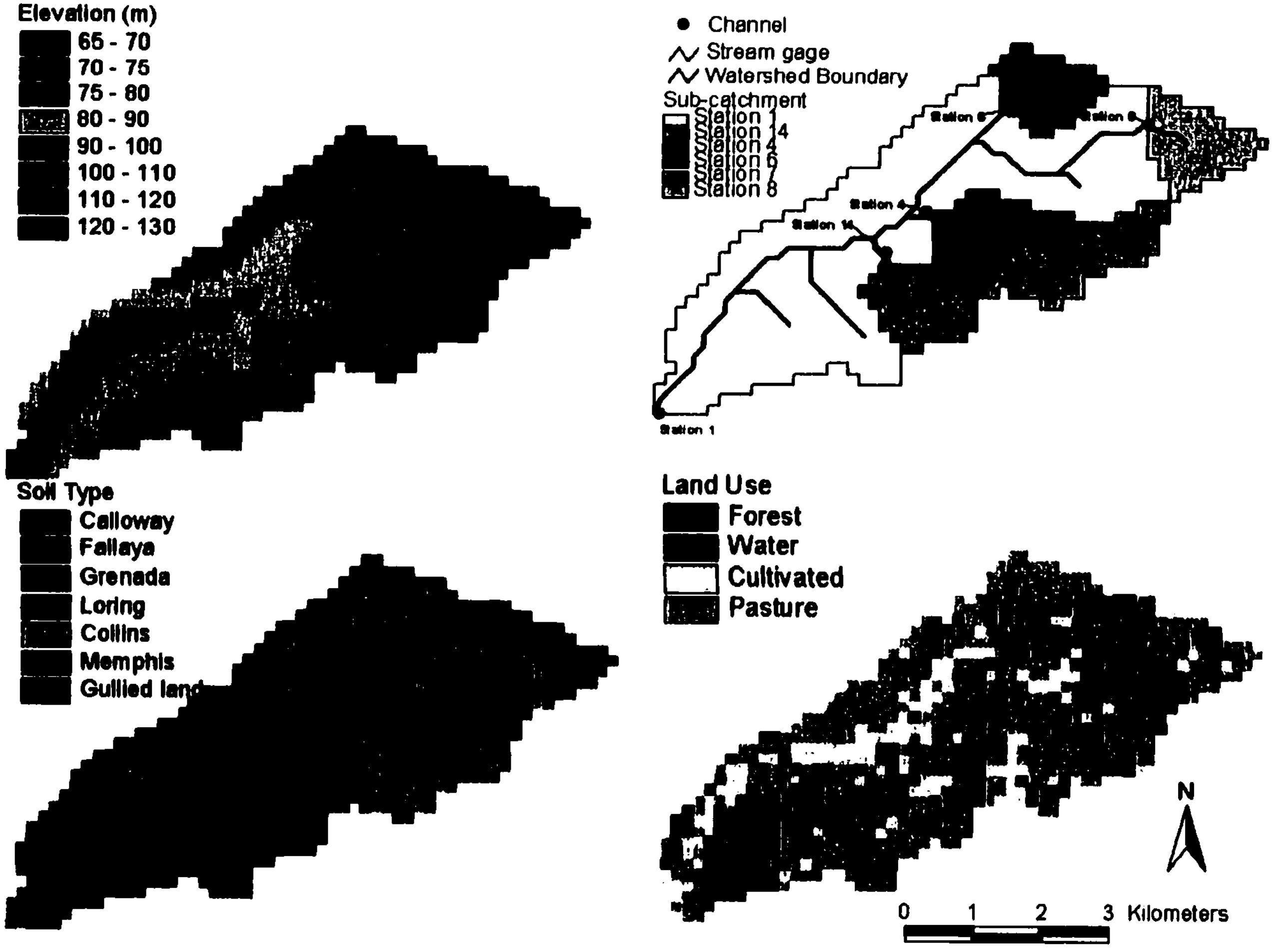


Figure II.1.3 DEM, watershed definition, soil type, and land use at 150-m

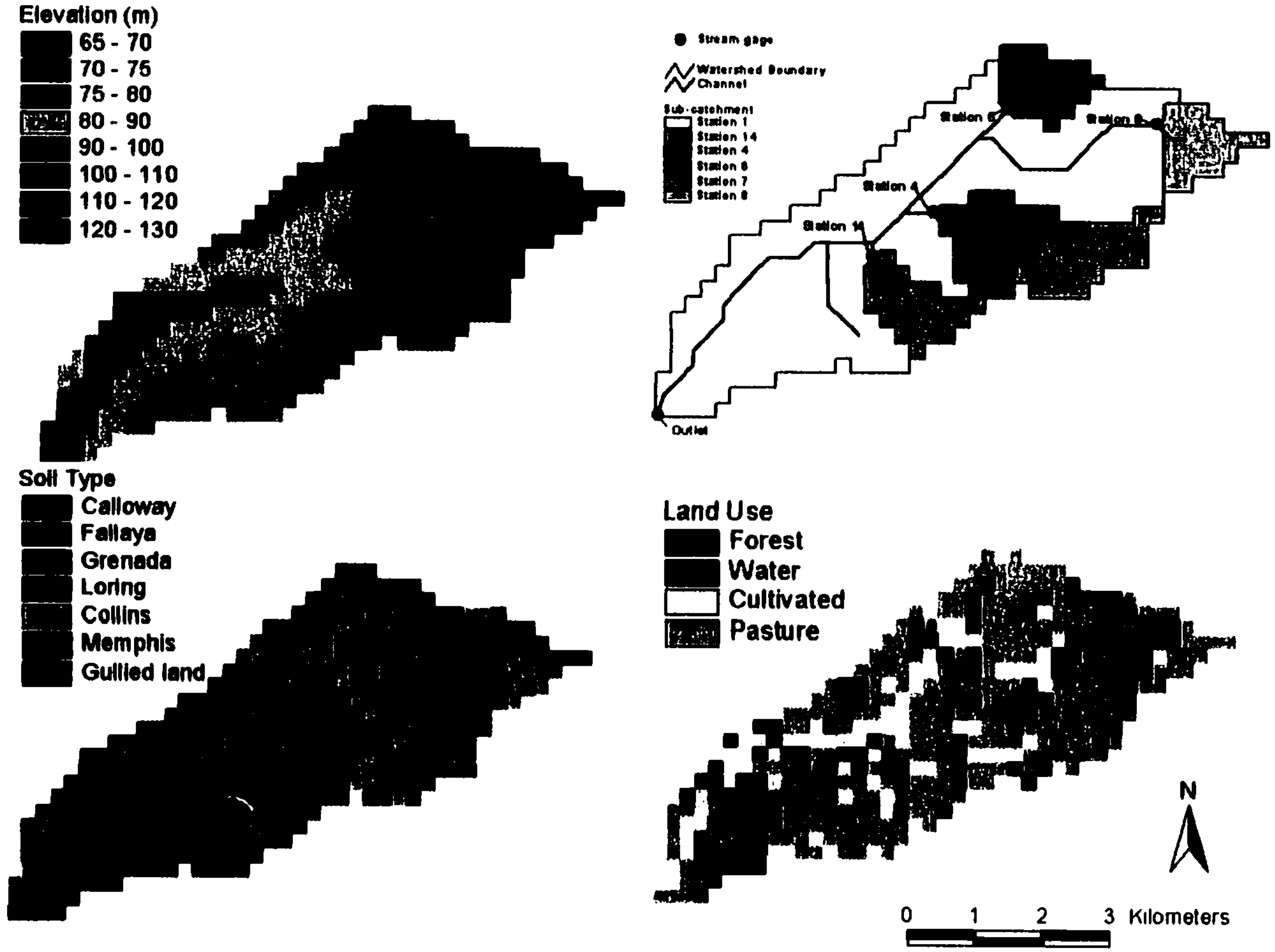


Figure II.1.4 DEM, watershed definition, soil type, and land use at 210-m

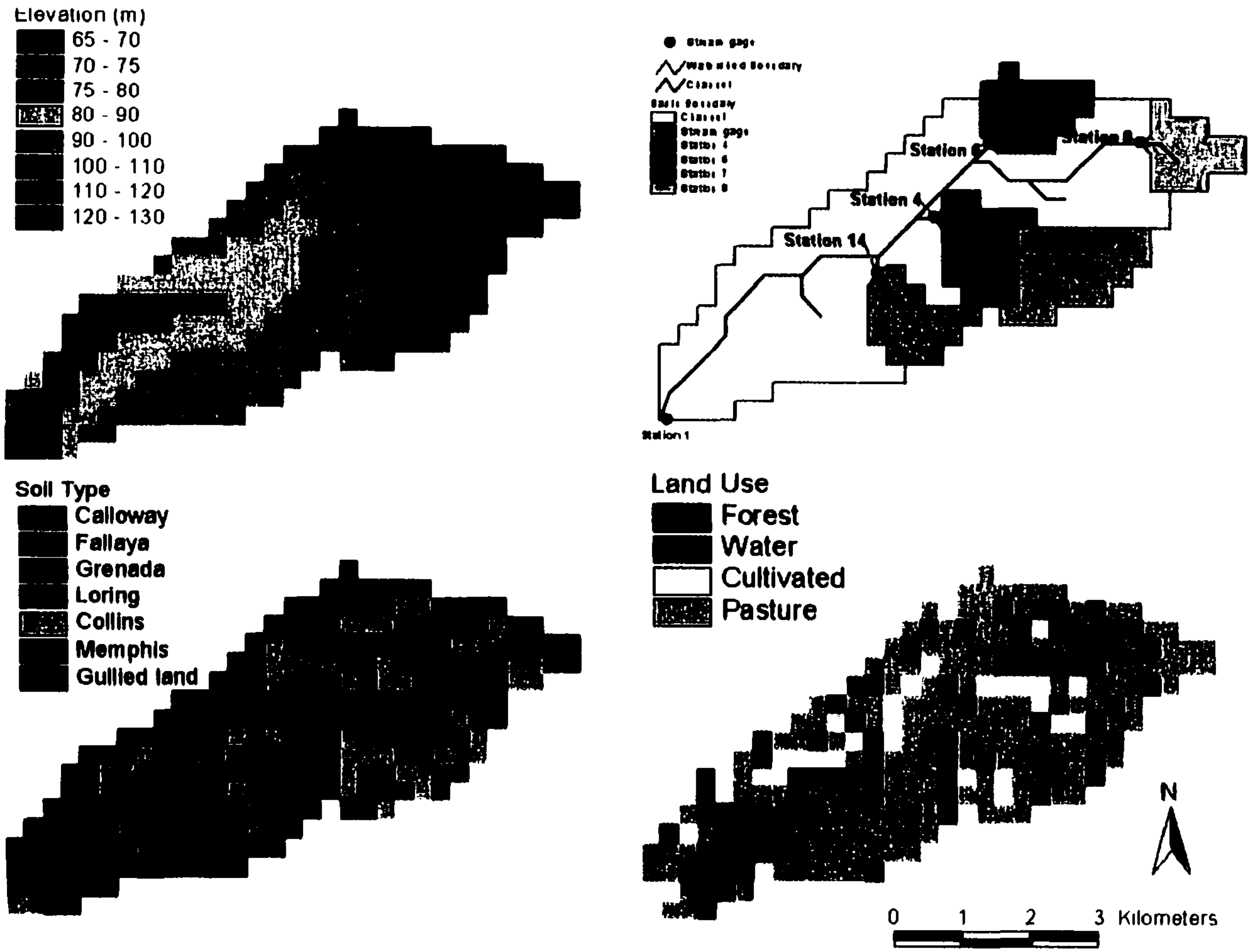


Figure II.1.5 DEM, watershed definition, soil type, and land use at 270-m

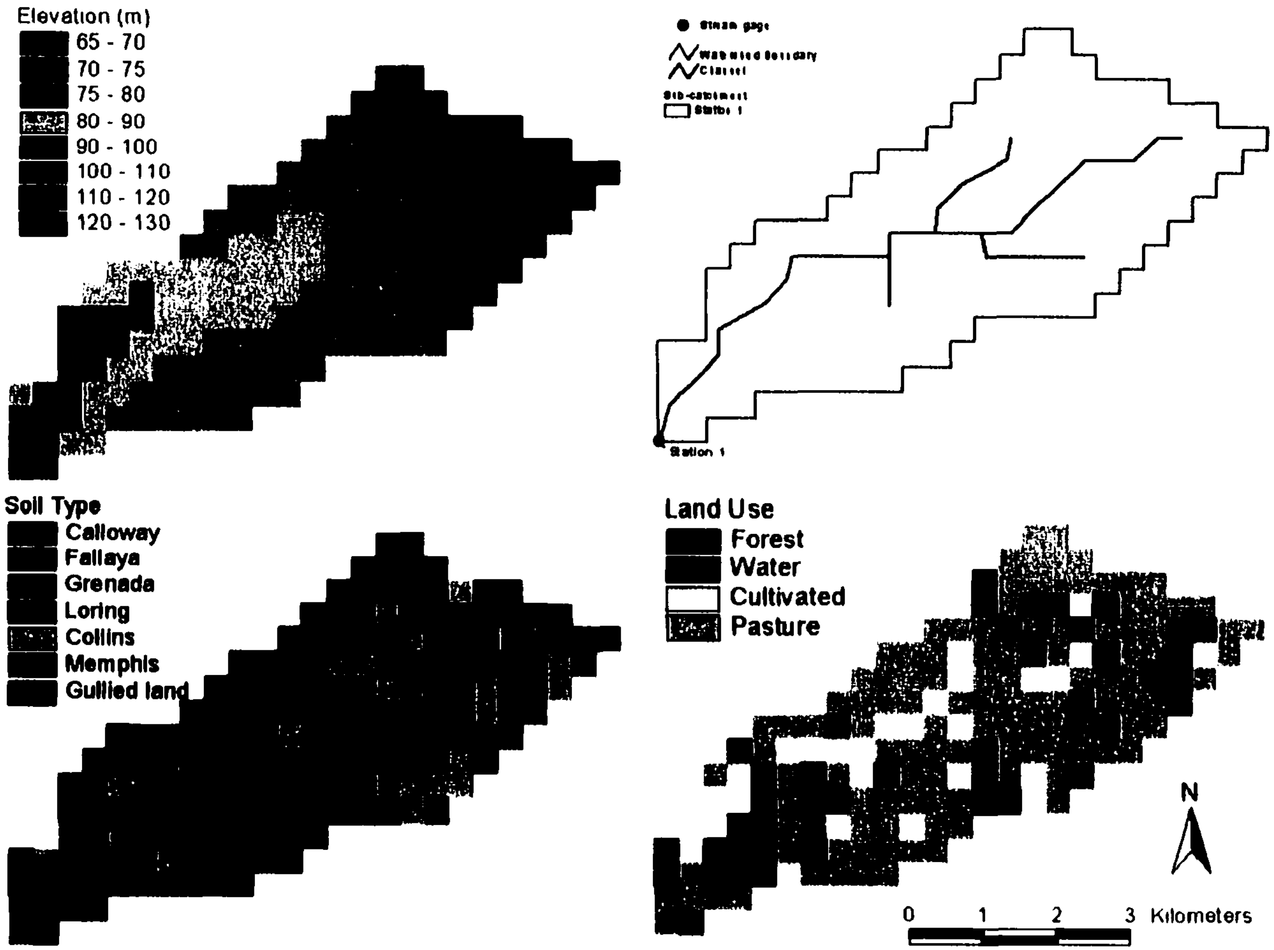
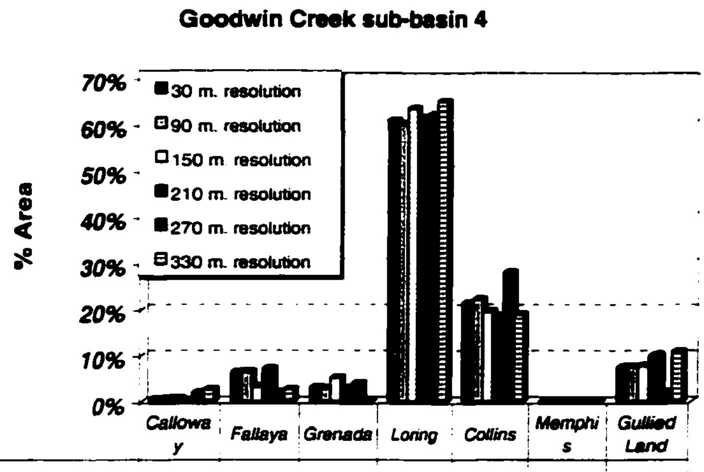


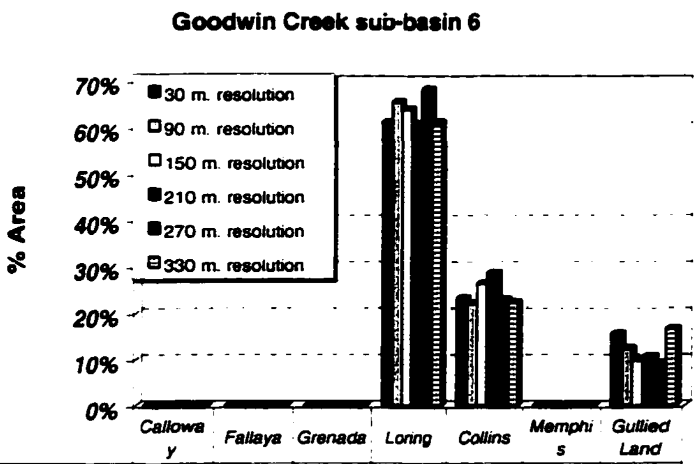
Figure II.1.6 DEM, watershed definition, soil type, and land use at 330-m

II.2 Percentage of the Area Covered by each Soil Type for each Grid Size and Sub-basin



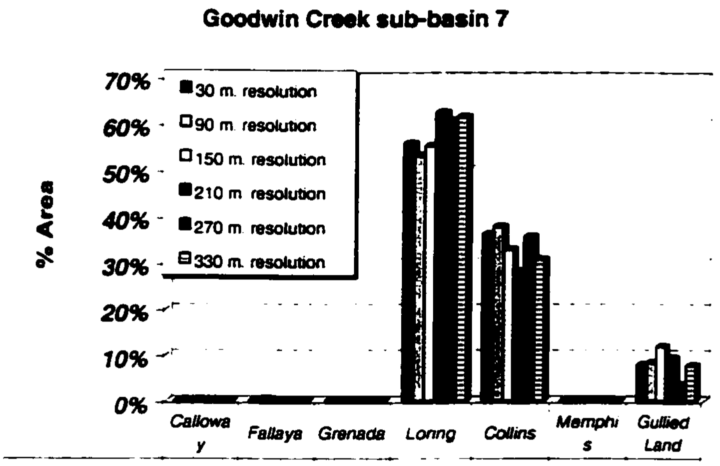
	Calloway	Fallaya	Grenada	Loring	Collins	Memphis	Gullied Land
30 m. resolution	1%	6%	3%	61%	21%	0%	7%
90 m. resolution	1%	7%	3%	60%	22%	0%	8%
150 m. resolution	1%	3%	5%	63%	20%	0%	8%
210 m. resolution	0%	7%	3%	61%	19%	0%	10%
270 m. resolution	2%	2%	4%	62%	28%	0%	2%
330 m. resolution	3%	3%	0%	65%	19%	0%	11%

Figure II.2.1 Percentage of the area covered by each soil type in sub-basin 4



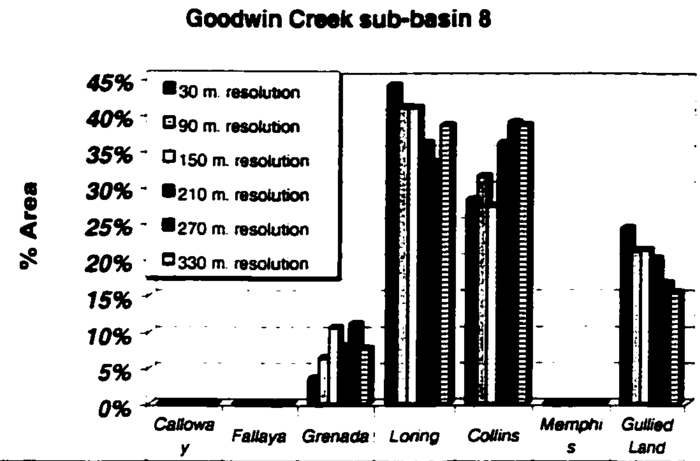
	Calloway	Fallaya	Grenada	Loring	Collins	Memphis	Gullied Land
30 m. resolution	0%	0%	0%	61%	23%	0%	16%
90 m. resolution	0%	0%	0%	66%	22%	0%	12%
150 m. resolution	0%	0%	0%	64%	26%	0%	10%
210 m. resolution	0%	0%	0%	61%	29%	0%	11%
270 m. resolution	0%	0%	0%	68%	23%	0%	9%
330 m. resolution	0%	0%	0%	61%	22%	0%	17%

Figure II.2.2 Percentage of the area covered by each soil type in sub-basin 6



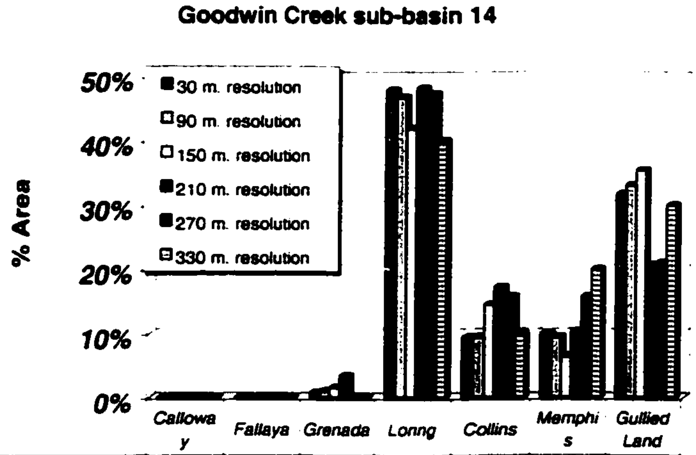
	Calloway	Fallaya	Grenada	Loring	Collins	Memphis	Gullied Land
30 m. resolution	0%	0%	0%	56%	36%	0%	8%
90 m. resolution	0%	0%	0%	53%	38%	0%	8%
150 m. resolution	0%	0%	0%	55%	33%	0%	12%
210 m. resolution	0%	0%	0%	63%	28%	0%	9%
270 m. resolution	0%	0%	0%	61%	36%	0%	4%
330 m. resolution	0%	0%	0%	62%	31%	0%	8%

Figure II.2.3 Percentage of the area covered by each soil type in sub-basin 7



	Calloway	Fallaya	Grenada	Loring	Collins	Memphis	Gullied Land
30 m. resolution	0%	0%	4%	44%	28%	0%	24%
90 m. resolution	0%	0%	6%	41%	31%	0%	21%
150 m. resolution	0%	0%	11%	41%	27%	0%	21%
210 m. resolution	0%	0%	8%	36%	36%	0%	20%
270 m. resolution	0%	0%	11%	33%	39%	0%	17%
330 m. resolution	0%	0%	8%	38%	38%	0%	15%

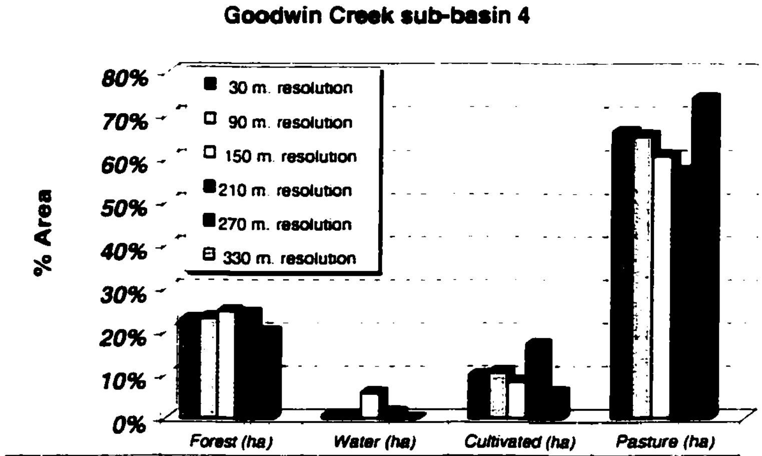
Figure II.2.4 Percentage of the area covered by each soil type in sub-basin 8



	Calloway	Fallaya	Grenada	Lonng	Collins	Memphis	Gullied Land
30 m. resolution	0%	0%	1%	48%	9%	10%	32%
90 m. resolution	0%	0%	1%	47%	9%	9%	33%
150 m. resolution	0%	0%	2%	42%	15%	6%	35%
210 m. resolution	0%	0%	3%	48%	17%	10%	21%
270 m. resolution	0%	0%	0%	47%	16%	16%	21%
330 m. resolution	0%	0%	0%	40%	10%	20%	30%

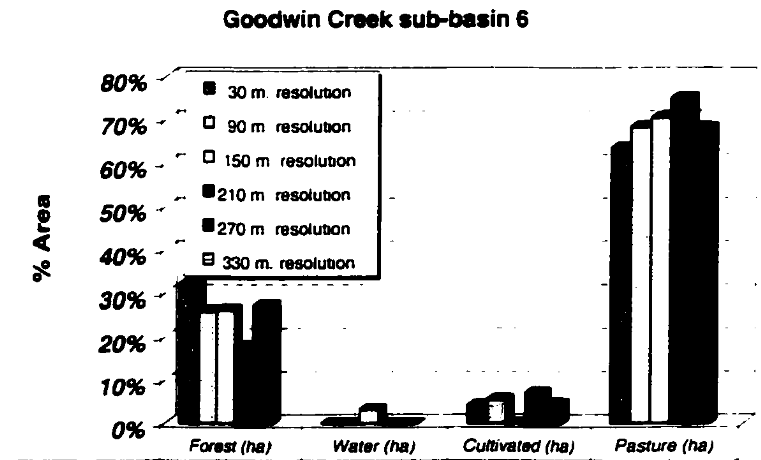
Figure II.2.5 Percentage of the area covered by each soil type in sub-basin 14

II.3 Percentage of the Area Covered by each Soil Type for each Grid Size and Sub-basin



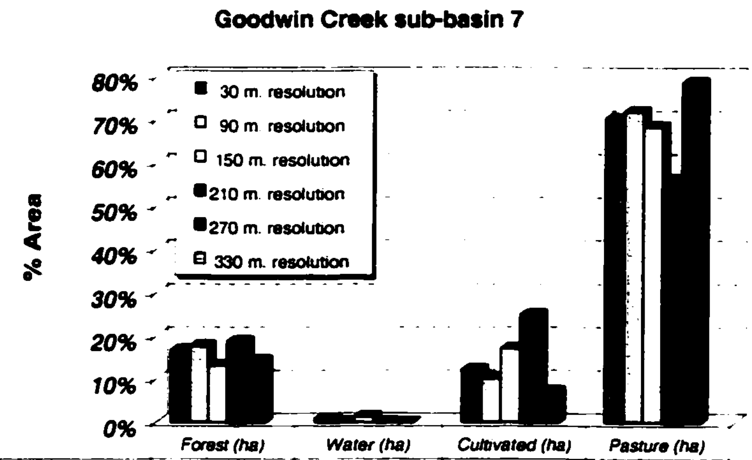
	Forest (ha)	Water (ha)	Cultivated (ha)	Pasture (ha)
30 m. resolution	23%	1%	10%	66%
90 m. resolution	23%	0%	11%	65%
150 m. resolution	25%	6%	8%	61%
210 m. resolution	24%	1%	17%	57%
270 m. resolution	20%	0%	6%	74%
330 m. resolution				

Figure II.3.1 Percentage of the area covered by each land use type in sub-basin 4



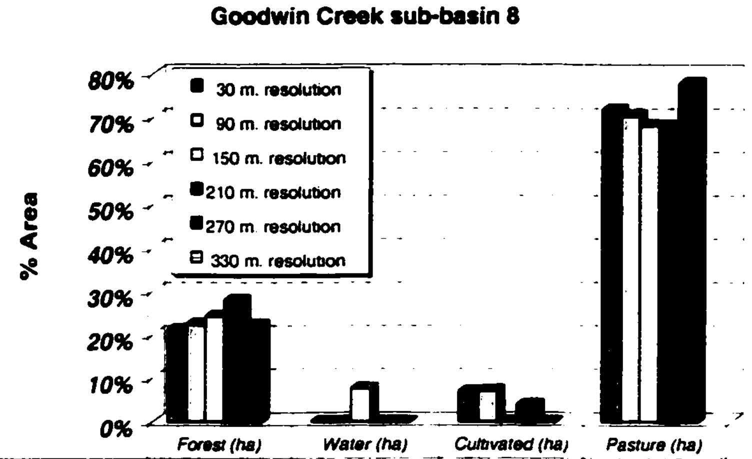
	Forest (ha)	Water (ha)	Cultivated (ha)	Pasture (ha)
30 m. resolution	32%	0%	4%	64%
90 m. resolution	26%	0%	6%	68%
150 m. resolution	26%	3%	0%	70%
210 m. resolution	18%	0%	7%	75%
270 m. resolution	27%	0%	5%	68%
330 m. resolution				

Figure II.3.2 Percentage of the area covered by each land use type in sub-basin 6



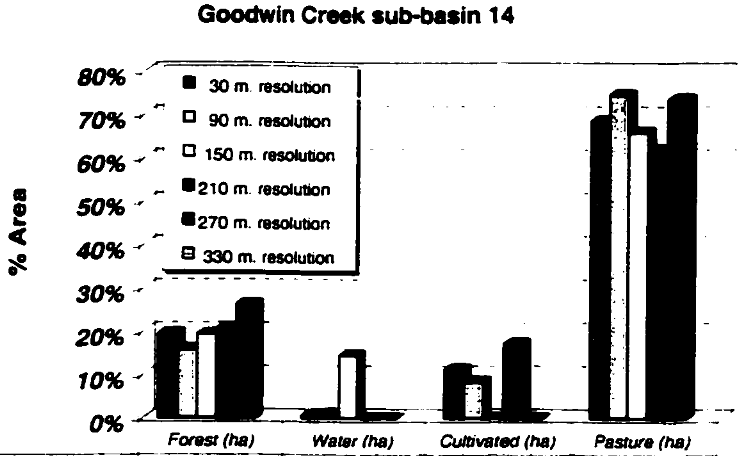
	Forest (ha)	Water (ha)	Cultivated (ha)	Pasture (ha)
30 m. resolution	17%	1%	12%	70%
90 m. resolution	18%	0%	10%	72%
150 m. resolution	13%	1%	17%	68%
210 m. resolution	19%	0%	25%	56%
270 m. resolution	14%	0%	7%	79%
330 m. resolution				

Figure II.3.3 Percentage of the area covered by each land use type in sub-basin 7



	Forest (ha)	Water (ha)	Cultivated (ha)	Pasture (ha)
30 m. resolution	21%	0%	7%	72%
90 m. resolution	22%	0%	7%	71%
150 m. resolution	24%	8%	0%	68%
210 m. resolution	28%	0%	4%	68%
270 m. resolution	22%	0%	0%	78%
330 m. resolution				

Figure II.3.4 Percentage of the area covered by each land use type in sub-basin 7



	Forest (ha)	Water (ha)	Cultivated (ha)	Pasture (ha)
30 m. resolution	19%	1%	11%	69%
90 m. resolution	16%	1%	8%	75%
150 m. resolution	19%	15%	0%	66%
210 m. resolution	21%	0%	17%	62%
270 m. resolution	26%	0%	0%	74%
330 m. resolution				

Figure II.3.5 Percentage of the area covered by each land use type in sub-basin 14

APPENDIX III

III.1 Precipitation Daily Totals Preceding Precipitation Events ---------- III-2

Figure III.1.1 Precipitation daily totals preceding the a) calibration and b) and c) validation events ---------- III-2

III.2 Calibration Event Input Files ---------- III-3

III.2.1 Project file: run4.prj ---------- III-3

III.2.2 Control file: run4.dat ---------- III-4

III.2.3 Channel file: chann0.045.dat ---------- III-5

III.3 Calibration Event Output Files ---------- III-6

III.3.1 Summary file: run4.sum ---------- III-6

III.3.2 Discharge file: run4.q ---------- III-10

III.3.3 Sediment discharge file: run4.q ---------- III-10

III.4 Event 2 Hydrographs and Sediment Graphs ---------- III-11

Figure III.4.1 Observed and simulated hydrographs for validation event 2 ---------- III-11
Figure III.4.2 Observed and simulated sediment graphs for validation event 2 ---------- III-12

III.5 Event 3 Hydrographs and Sediment Graphs ---------- III-13

Figure III.5.1 Observed and simulated hydrographs for validation event 3 ---------- III-13
Figure III.5.2 Observed and simulated sediment graphs for validation event 3 ---------- III-14

III.1 PRECIPITATION DAILY TOTALS PRECEDING PRECIPITATION EVENTS

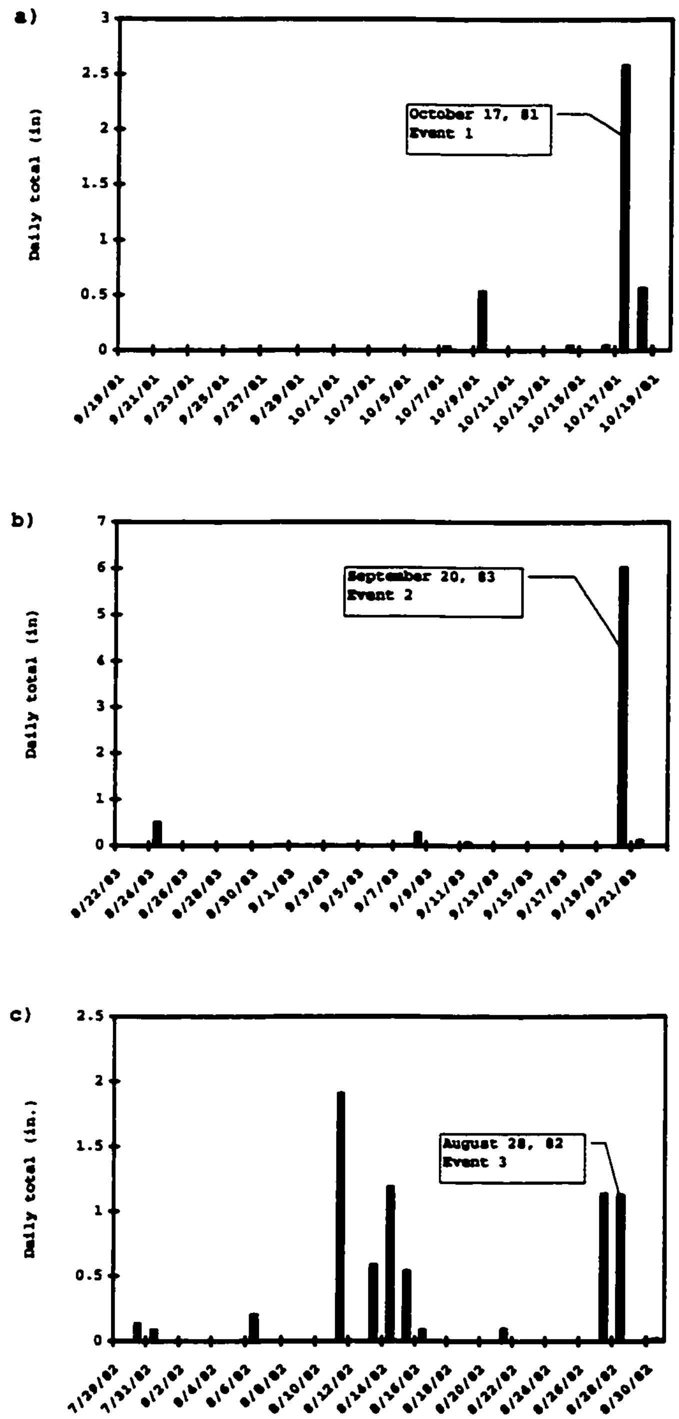


Figure III.1.1 Precipitation daily totals preceding the a) calibration and b) and c) validation events

III.2 Calibration Event Input Files

III.2.1 Project file: run4.prj

```
#############  INPUT DATA FILES  #############
CONTROL                     Input/run4.dat
PRECIP                      Input/17oct81.pre
CHANNEL                     input/chann0.045.dat

#############  INPUT DATA GRIDS  #############
ELEVATION                   Input/DEM90.asc
SOIL                        Input/soils90.asc
LANDUSE                     Input/luse90.asc
MASK                        Input/mask90.asc
LINK                        Input/links90.asc
NODE                        Input/nodes90.asc

##############  OUTPUT FILES  ################
SUMMARY                     Output/run4.sum
DISCHARGE_OUT               Output/run4.q
SEDIMENT_OUT                Output/run4.qs

######  OUTPUT TIME SERIES GRID NAMES  #######
RAINFALL                    outmaps/rain
INF_DEPTH                   outmaps/infvol
WATER_DEPTH                 outmaps/depth
SED_CONC                    outmaps/conc
EROS_DEP                    OutMaps/netvol
SAND_MFAC_OUT               outmaps/mfacSAND
SILT_MFAC_OUT               outmaps/mfacSILT
CLAY_MFAC_OUT               outmaps/mafacCLAY
TOTAL_MFAC_OUT              outmaps/mfacTOT
```

III.2.2 Control file: run4.dat

```
2  42000  9000  30  150
57  1  0.003539
1
1

1
16              900
231568.7117     3791553.7426
233072.7721     3793603.9817
236856.3335     3794839.5058
236459.33       3795732.9309
236662.9499     3793699.8824
238396.5605     3794734.2989
239756.4195     3795151.8743
234644.3966     3793655.021
237777.6768     3796476.9739
238787.2374     3796056.2956
233758.5726     3794401.0867
235620.4058     3795906.558
234840.6096     3792778.0814
233122.6924     3791785.0733
236523.6428     3792701.0902
238663.3566     3793326.2882

4  0
0.25           3            0.0036          1
0.01           0            0               1
0.15           1            0.18            1
0.2            1.5          0.072           1

1  1  7
0.336        22     0.29  0.25  0.55  0.2   0.4
0.3072       14     0.29  0.25  0.55  0.2   0.1
0.3552       17     0.29  0.3   0.6   0.1   0.2
0.3648       22     0.29  0.25  0.55  0.2   0.4
0.3456       18     0.29  0.3   0.6   0.1   0.2
0.432        22     0.29  0.3   0.6   0.1   0.1
0.384        15     0.29  0.25  0.55  0.2   0.1

1  6  3
 57            1           2071.98
 27            42            365.88
 11            55            145.86
 34            57            166.07
 14            78            135.85
 34            36            151.46

 1  6  2
 57            1           2071.98
 27            42            365.88
 11            55            145.86
 34            57            166.07
 14            78            135.85
 34            36            151.46
```

III.2.3 Channel file: chann0.045.dat

```
1  12
 2         20         3          0         0.045        1
 2         20         3          0         0.045        1
 2         20         3          0         0.045        1
 2         20         3          0         0.045        1
 2         20         3          0         0.045        1
 2         20         3          0         0.045        1
 2         20         3          0         0.045        1
 2         20         3          0         0.045        1
 2         20         3          0         0.045        1
 2         20         3          0         0.045        1
 2         20         3          0         0.045        1
 2         20         3          0         0.045        1
 2  11
 2         22         3.5         0         0.045        1
 2         22         3.5         0         0.045        1
 2         22         3.5         0         0.045        1
 2         22         3.5         0         0.045        1
 2         22         3.5         0         0.045        1
 2         22         3.5         0         0.045        1
 2         22         3.5         0         0.045        1
 2         22         3.5         0         0.045        1
 2         22         3.5         0         0.045        1
 2         22         3.5         0         0.045        1
 2         22         3.5         0         0.045        1
 3  7
 2         27         4.3         0         0.045        1
 2         27         4.3         0         0.045        1
 2         27         4.3         0         0.045        1
 2         27         4.3         0         0.045        1
 2         27         4.3         0         0.045        1
 2         27         4.3         0         0.045        1
 2         27         4.3         0         0.045        1
 4  9
```

............[*NOTE: MIDDLE SECTION OF THE CHANNEL FILE HAS BEEN DELETED]*

```
18  16
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
 2         34         6.05        0         0.045        1
```

III.3 CALIBRATION EVENT OUTPUT FILES

III.3.1 Summary file: run4.sum

```
CONTROL Input Data
=================
Control file : Input/run4.dat
DT = 2.00 sec NITER= 42000 NITRN = 9000 NPRN = 30 NPLT = 150

Overland outlet cell:
JOUT = 57 KOUT = 1 SOVOUT = 0.003539

CHANCHECK= 1
ELCONV = 1.00

Rainfall Data
=============
IRAIN = 1
Rainfall file: Input/17oct81.pre
NRG = 16 NREAD = 900
XRG[1] = 231568.718750  YRG[1] = 3791553.750000
XRG[2] = 233072.765625  YRG[2] = 3793604.000000
XRG[3] = 236856.328125  YRG[3] = 3794839.500000
XRG[4] = 236459.328125  YRG[4] = 3795733.000000
XRG[5] = 236662.953125  YRG[5] = 3793700.000000
XRG[6] = 238396.562500  YRG[6] = 3794734.250000
XRG[7] = 239756.421875  YRG[7] = 3795151.750000
XRG[8] = 234644.390625  YRG[8] = 3793655.000000
XRG[9] = 237777.671875  YRG[9] = 3796477.000000
XRG[10] = 238787.234375 YRG[10] = 3796056.250000
XRG[11] = 233758.578125 YRG[11] = 3794401.000000
XRG[12] = 235620.406250 YRG[12] = 3795906.500000
XRG[13] = 234840.609375 YRG[13] = 3792778.000000
XRG[14] = 233122.687500 YRG[14] = 3791785.000000
XRG[15] = 236523.640625 YRG[15] = 3792701.000000
XRG[16] = 238663.359375 YRG[16] = 3793326.250000

Land Use Parameters
===================
NMAN = 4 INDEXSDEP = 0

      Index    Manning   Intercept.   Cusle      Pusle
      -----    -- n --   -- [mm] --   -----      -----
        1      0.250     3.000        0.004      1.000
        2      0.010     0.000        0.000      1.000
        3      0.150     1.000        0.180      1.000
        4      0.200     1.500        0.072      1.000
```

Soil type Parameters
===================
INDEXINF= 1 INDEXEROS= 1 NSOIL= 7

Index	Ks	G	Md	%Sand	%Silt	%Clay	Kusle
----	[cm/h]	[cm]	-----	-----	-----	-----	----
1	0.336	22.000	0.290	0.250	0.550	0.200	0.400
2	0.307	14.000	0.290	0.250	0.550	0.200	0.100
3	0.355	17.000	0.290	0.300	0.600	0.100	0.200
4	0.365	22.000	0.290	0.250	0.550	0.200	0.400
5	0.346	18.000	0.290	0.300	0.600	0.100	0.200
6	0.432	22.000	0.290	0.300	0.600	0.100	0.100
7	0.384	15.000	0.290	0.250	0.550	0.200	0.100

Internal Gages Data

INDEXDIS = 1 NDIS = 6 Q_units = 3

Gage	Row	Column	Area
----	---	------	[has]
1	57	1	2071.980
2	27	42	365.880
3	11	55	145.860
4	34	57	166.070
5	14	78	135.850
6	34	36	151.460

Internal Sediment Gages Data

INDEXSED = 1 NSED = 6 Qs_Units = 2

Gage	Row	Column	Area
----	---	------	[has]
1	57	1	2071.980
2	27	42	365.880
3	11	55	145.860
4	34	57	166.070
5	14	78	135.850
6	34	36	151.460

ASCII grid files header

```
ncols        95
nrows        58
xllcorner    231584.00
yllcorner    3791449.00
cellsize     90.00
NODATA_value       -9999
```

The following Grids have been read:

```
Elevation Grid: Input/DEM90mod_c.asc
Shape Grid:         Input/mask.asc
Soils Grid:         Input/soils90.asc
Landuse Grid:       Input/luse90.asc
Node Grid:  Input/nodes.asc
Link Grid:  Input/links.asc
```

Number of cells in basin: 2558
Basin area: 2071.980 [ha]

Channel Linkage Information

Maximum Number of Links = 18
Maximum Number of Nodes = 24

———— *[NOTE: CHANNEL LINKAGE INPUT DATA HAS BEEN DELETED HERE]*

SUMMARY OF FLOW OUTPUT

Peak Discharge (m3/s)....................................=	40.37
Time to Peak (min)..=	277.07
Initial Surface Volume (m3)...............................=	0.00
Volume of Rainfall - retention (m3).......................=	1497104.00
Volume leaving the Watershed (m3).........................=	465479.41
Percentage of Vout to Vin.................................=	31.09
Final Surface Volume (m3).................................=	10984.83
Volume Infiltrated (m3)...................................=	1020636.44
Percentage of Vinftot to Vin..............................=	68.17
Percent Mass Balance...................................... =	0.00

HYDROLOGICAL VARIABLES MINIMUM AND MAXIMUM VALUES

Min. Rainfall Intensity (mm/hr)...........................=	0.00
Max. Rainfall Intensity (mm/hr)...........................=	51.60
Min. Infiltration Depth (mm)..............................=	0.00
Max. Infiltration Depth (mm)..............................=	84.94
Min. Overland Depth (m)...................................=	0.00
Max. Overland Depth (m)...................................=	0.23
Min. Channel Depth (m)....................................=	0.00
Max. Channel Depth (m)....................................=	1.47

SUMMARY OF SEDIMENT OUTPUT : Volume in Cubic Meters
(Percentages from the total eroded

Total Volume of Sand Eroded...............................=	462.34
Total Volume of Silt Eroded...............................=	983.90
Total Volume of Clay Eroded...............................=	292.30
Total Volume of Material Eroded...........................=	1738.54

Volumes of Eroded Sediment Remaining on the Overland:

Total Volume of Suspended Sand............................=	0.00
Total Volume of Suspended Silt............................=	0.00
Total Volume of Suspended Clay............................=	0.12
Total Volume of Suspended Sediment........................=	0.12
Total Volume of Deposited Sand............................=	314.39

```
Total Volume of Deposited Silt...........................=      535.89
Total Volume of Deposited Clay...........................=       14.20
Total Volume of Deposited Sediment.......................=      864.47
Total Volume of Sand Remaining on the Overland...........=      314.39
Total Volume of Silt Remaining on the Overland...........=      535.89
Total Volume of Clay Remaining on the Overland...........=       14.32
Total Volume of Eroded Material Remaining on the Overland..=    864.59

Volumes of Eroded Sediment Remaining in the Channels:
----------------------------------------------------
Total Volume of Suspended Sand...........................=        0.00
Total Volume of Suspended Silt...........................=        0.03
Total Volume of Suspended Clay...........................=        3.67
Total Volume of Suspended Sediment.......................=        3.71
Total Volume of Deposited Sand...........................=      122.80
Total Volume of Deposited Silt...........................=      128.84
Total Volume of Deposited Clay...........................=        1.00
Total Volume of Deposited Sediment.......................=      252.64
Total Volume of Sand Remaining in the Channels...........=      122.80
Total Volume of Silt Remaining in the Channels...........=      128.87
Total Volume of Clay Remaining in the Channels...........=        4.67
Total Volume of Eroded Material Remaining in the Channels..=    256.35

Volumes of Eroded Sediment Leaving the Watershed:
----------------------------------------------------
Total Volume of Eroded Sand Leaving the Watershed.........=      25.15
Total Volume of Eroded Silt Leaving the Watershed.........=     319.05
Total Volume of Eroded Clay Leaving the Watershed.........=     273.21
Total Volume of Eroded Material Leaving the Watershed.....=     617.41

Percent Mass Balance......................................=      -0.01

SEDIMENT VARIABLES MINIMUM AND MAXIMUM VALUES
=============================================

Max. Flux Conc. Overland (m3/s/m3/s)....................= 0.0468933
Max. Flux Conc. Channels (m3/s/m3/s)....................= 0.0138061
Max. Suspended Sediment in Overland (mm)................= 0.569869
Max. Suspended Sediment in Channels (mm)................= 1.28504
Max. Concentration on Overland (m3/m3)..................= 0.00890181
Max. Concentration in Channels (m3/m3)..................= 0.0091609
Total Net Volume (mm) at the End of the Simulation:
    Minimum Value ......................................=     -11.055
    Maximum Value ......................................=       7.345
    Mean ...............................................=      -0.030
    Standard Deviation .................................=       0.531

Program stops at simulation minute:   1400.00
```

III.3.2 Discharge file: run4.q

Time	Row57Col1	Row27Col42	Row11Col55	Row34Col57	Row14Col78	Row34Col36
1.00	0.000	0.000	0.000	0.000	0.000	0.000
2.00	0.000	0.000	0.000	0.000	0.000	0.000
3.00	0.000	0.000	0.000	0.000	0.000	0.000

.........[*NOTE: THIS SECTION OF THE DISCHARGE FILE HAS BEEN DELETED]*

294.00	6.718	3.537	3.396	3.001	3.840	
295.00	6.686	3.483	3.337	2.957	3.780	
296.00	6.654	3.430	3.279	2.913	3.721	

.........[*NOTE: THIS SECTION OF THE DISCHARGE FILE HAS BEEN DELETED]*

1397.00	0.039	0.005	0.000	0.002	0.001
1398.00	0.039	0.005	0.000	0.002	0.001
1399.00	0.039	0.005	0.000	0.002	0.001
1400.00	0.039	0.005	0.000	0.002	0.001

III.3.3 Sediment discharge file: run4.q

Time	Row57Col1	Row27Col42	Row11Col55	Row34Col57	Row14Col78	Row34Col36
1.00	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
2.00	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
3.00	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

.....[*NOTE: THIS SECTION OF THE SEDIMENT DISCHARGE FILE HAS BEEN DELETED]*

294.00	6.87274	3.62914	5.18821	2.15760	5.53370	8.28419
295.00	6.87603	3.56300	5.04122	2.13972	5.40966	8.11878
296.00	6.87683	3.49812	4.89810	2.12220	5.28760	7.95496
297.00	6.87516	3.43447	4.75876	2.10501	5.16751	7.79274

.....[*NOTE: THIS SECTION OF THE SEDIMENT DISCHARGE FILE HAS BEEN DELETED]*

1398.00	0.01110	0.00115	0.00003	0.00048	0.00003	0.00017
1399.00	0.01107	0.00115	0.00003	0.00048	0.00003	0.00016
1400.00	0.01104	0.00114	0.00003	0.00047	0.00003	0.00016

III.4 Event 2 Hydrographs and Sediment Graphs

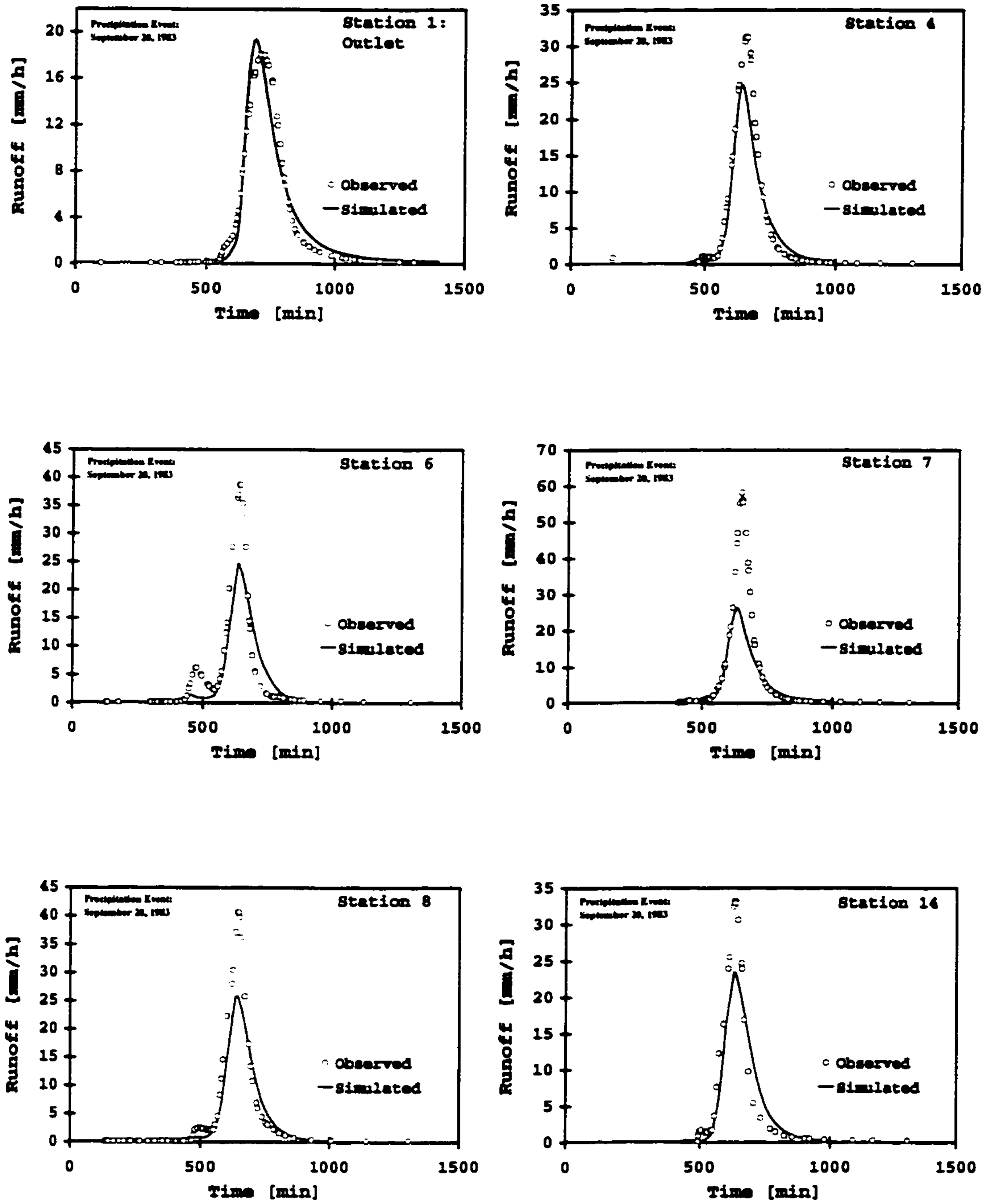


Figure III.4.1 Observed and simulated hydrographs for validation event 2

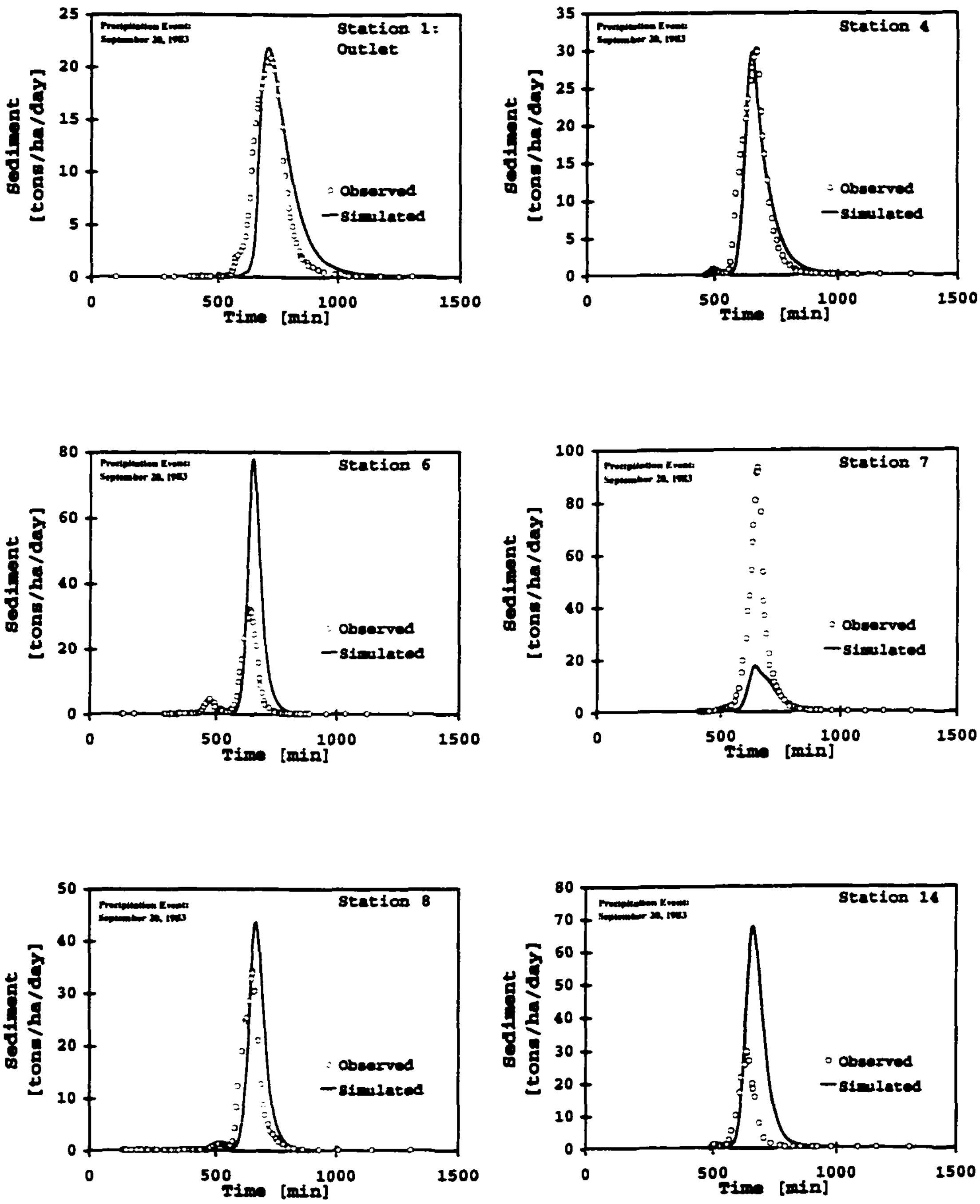


Figure III.4.2 Observed and simulated sediment graphs for validation event 2

III.5 Event 3 Hydrographs and Sediment Graphs

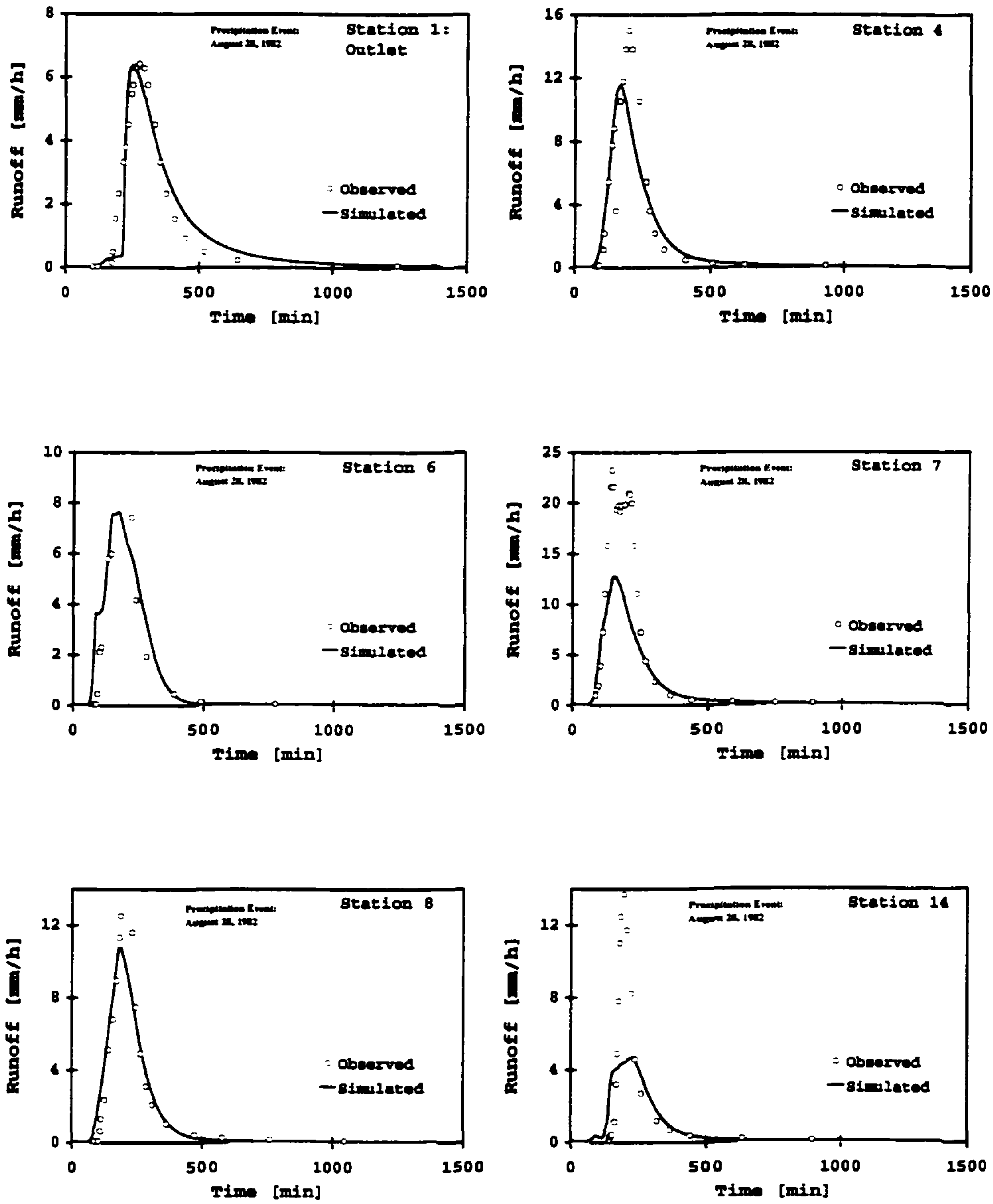


Figure III.5.1 Observed and simulated hydrographs for validation event 3

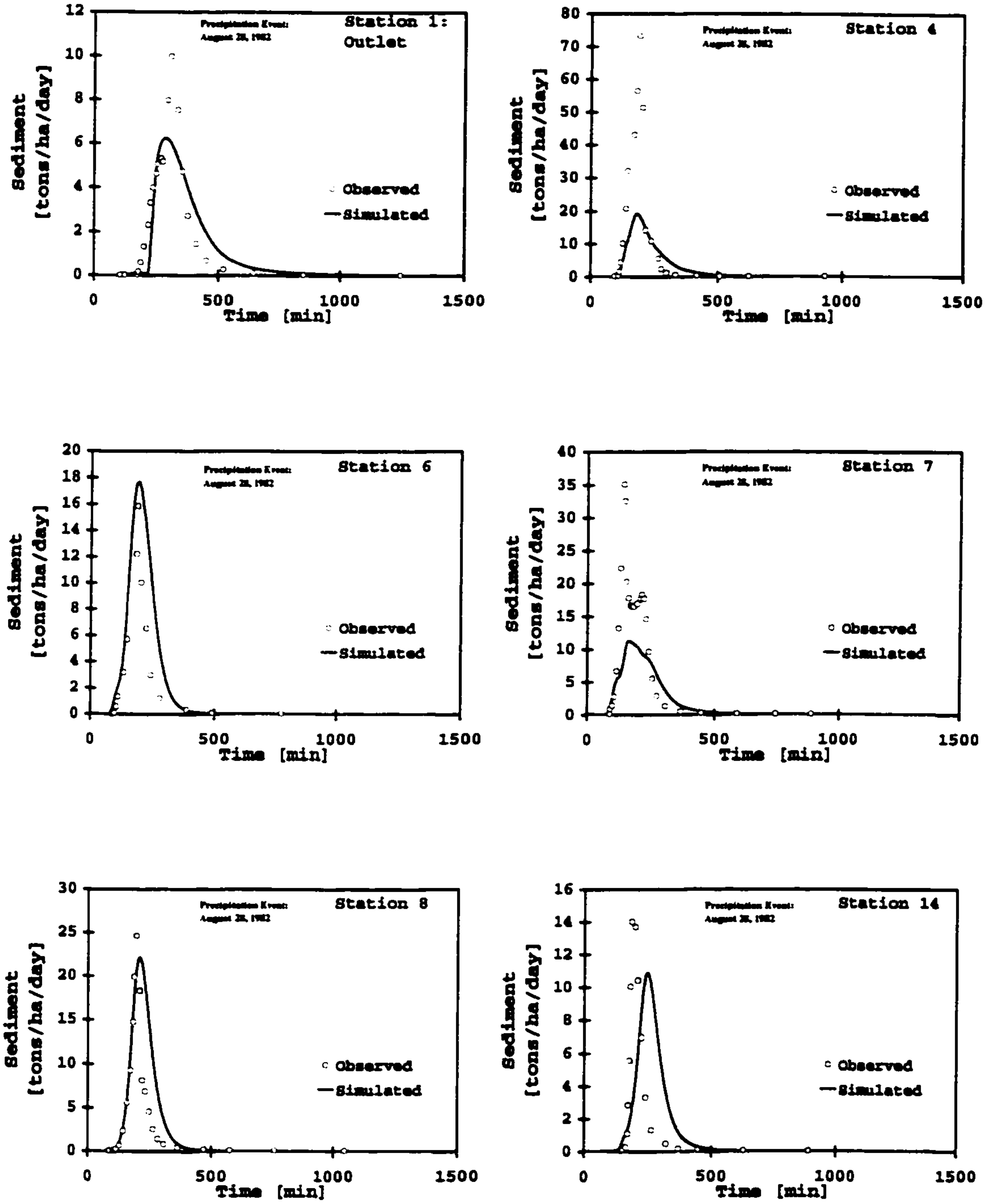


Figure III.5.2 Observed and simulated sediment graphs for validation event 3