WEI ZENG

A Model For Understanding And Managing The Impacts Of Sediment Behavior On River Water Quality

(Under the Direction of M. BRUCE BECK)

A mathematical model (Sediment-Transport-Associated Nutrient Dynamics – STAND) has been developed for the study of sediment-associated water quality issues. The model is intended to simulate changes of water quality constituents associated with sediment behavior. It has a 3-level structure. The first level accounts for the hydraulics of open-channel flow. The second computes sediment transport potential and actual rates based on the information provided by the first level. A non-equilibrium approach is used. In the third level, changes of nutrient concentrations along a studied river are computed with the consideration of nutrient transport, adsorption/desorption, and interstitial water release. In order to calibrate the model, field data were collected from the Oconee River, a major tributary of the Altamaha River in Georgia. Two stations, approximately 17 km distant from each other, were established along the river for the purpose of data collection. Observations of the river's hydraulics, suspended sediment, and water quality (mainly ortho-phosphate, nitrate, temperature, specific conductivity, oxidation-reduction potential, dissolved oxygen, and pH) were collected at the two stations. Another data set collected along a major tributary of the Yellow River in China was also used for calibration and evaluation of the model's hydraulic and sediment transport parts. Calibration and evaluation results are encouraging, which suggests that STAND is a useful tool for the thorough study and understanding of water quality issues associated with sediment behavior. A semi-hypothetical river has been mathematically constructed and the model has been applied to the river for the study of sediment and nutrient behavior under the influence of hydrological conditions, pattern of upstream release, the roles of peak flow and base flow, and lateral inflows of sediment and nutrients.

INDEX WORDS: Mathematical model, Sediment transport, Water quality, Nutrient, Calibration, Evaluation

A MODEL FOR UNDERSTANDING AND MANAGING THE IMPACTS OF SEDIMENT BEHAVIOR ON RIVER WATER QUALITY

by

WEI ZENG

B.E. TSINGHUA UNIVERSITY, CHINA, 1992

A Dissertation Submitted to the Graduate Faculty of The University of Georgia in Partial Fulfillment of the Requirements for the Degree

DOCTOR OF PHILOSOPHY

ATHENS, GEORGIA

2000

A MODEL FOR UNDERSTANDING AND MANAGING THE IMPACTS OF SEDIMENT BEHAVIOR ON RIVER WATER QUALITY

by

WEI ZENG

Approved:

Major Professor: M. Bruce Beck

Committee: E. William Tollner
Ming-yi Sun
C. Rhett Jackson
Todd C. Rasmussen

Electronic Version Approved:

Gordhan L. Patel
Dean of the Graduate School
The University of Georgia
December 2000

DEDICATION

To my wife, Tongrui, who has brought me the brightest days in my life, and has always been supporting, understanding, and encouraging during my study.

ACKNOWLEDGEMENTS

I would like to thank Mr. Demetrius Cox, Mr. Robert Lawler, and Ms. Rong Liu, who work with me in the Wanell School of Forest Resources, for their help with the field work and laboratory analysis during the two data collection campaigns, which provided valuable data sets for the study. During the campaigns, Mr. Larry Robbins, Mr. Fred McCannon, and Mr. Kurt Johnson, who work at the Barnett Shoals Hydropower Station, offered their helping hands in setting up measuring equipment and provided a friendly working atmosphere. It has been pleasant working with them.

I would also like to thank Roger McFarlane, and George Bailey of the United States Geological Survey for providing references and permission for us to use field facilities of the USGS for our data collection.

During the study, I have had interesting and valuable discussions with Dr. Todd Rasmussen, Dr. Rhett Jackson, Dr. Ming-yi Sun, and Dr. William Tollner. Their advice has been inspiring and helpful.

Ms. Jenny Yearwood, administrative coordinator of our research group, has always been helpful whenever I encountered difficulties with practical matters.

My parents have always been supportive in my years of education and study. I thank them for providing me with an excellent education and for encouraging me to aim high and pursue graduate study.

Finally, I would like to pay special thanks to Dr. M. Bruce Beck, my major professor, who has supported, advised, and encouraged me throughout my study. His help is far beyond what I can describe with words.

TABLE OF CONTENTS

Page

CHAPTER 1

INTRODUCTION

Mechanism based mathematical models have been developed to study different behaviors of river systems and to help engineering practice and management decision making based on the studied behaviors. These behaviors include physical transport of water, sediments, and solutes, exchange of sediments between bed and overlying water, exchange of material between dissolved phase and solid phase, production and consumption of dissolved materials etc. Models fall into different categories depending on what components are incorporated in them.

Traditional civil engineering-oriented models such as HEC-6, Fluvial-12, and GSTARS 2.0 are mainly developed to simulate hydraulics, sediment transport and resulting channel morphological changes. Although satisfactory simulation results have been provided by these models, they fall short of the ability to simulate water quality issues because no such component is incorporated in any of them.

On the other hand, traditional water quality models, such as QUAL2E, and WASP5, usually have components for dissolved oxygen, pH, alkalinity, temperature, nutrients, algae, zooplankton, and coliform bacteria etc. Very few of them consider sediments as a water quality component, or as something that affects other components. None of them consider the process of sediment transport and its consequences for water quality.

Given the fact that approximately 95% of phosphorus in streams tends to adhere to sediment particles (Hem 1985, Meybeck 1982), and that suspended sediment transport rate is usually 4 to 20 times as much as that of bed load sediment (Yang 1996), we know

that not only is sediment an important factor that affects water quality, but also it is one of the most important sinks and sources of some nutrients which affect water quality significantly. So incorporating a sediment component in a water quality model, or conversely, adding water quality components to a civil engineering river model, is essential in studying sediment-related water quality issues.

A comprehensive mechanism-based model, Sediment Transport Associated Nutrient Dynamics (STAND) has been developed to study not only sediment as a water quality factor, but also its potential effects on other components as well. STAND has a three-level structure. At the base level, hydrodynamical behavior is solved first, which provides information for subsequent simulations. The second level deals with sediment transport potential and the actual suspended sediment transport rate, as well as the adjustment of channel morphology. The third level deals with the water quality, where the behaviors of nutrients and dissolved oxygen are computed.

As with some of the traditional civil engineering models, STAND has the ability to compute hydrodynamics and sediment transport potential (which is used as the actual sediment transport rate in some models). Since the actual sediment transport rate is usually not the same as the transport potential, it is important to compute the actual rate based on the transport potential. A potential-approaching mechanism is proposed in STAND, which thus distinguishes it from other contemporary models.

Unlike a traditional water quality model, STAND does not cover all the aspects of water quality. Rather, it focuses on the dynamics of nutrients, which are hypothesized to be affected by sediment behavior, and of dissolved oxygen.

To calibrate and validate different levels and components of the model, data sets of high sampling frequency and long periods for the Oconee River were collected in the summer of 1998 and the spring of 1999. To further strengthen the calibration and evaluation of the hydraulics and sediment transport part of the model, data sets of high quality (high frequency, long period, base and extreme hydrologic events) for the Yellow

River of China, which is known for its high sediment concentration, were used. Encouraging results were obtained for most hydraulic, sediment transport, and water quality components.

A partial data set on the Chattahoochee River is available for the author to construct a conceptual Chattahoochee-like river. STAND was then applied to the river to study the effects of different incoming hydraulic, sediment, and nutrient conditions on the downstream receiving waters.

CHAPTER 2

LITERATURE REVIEW

The current status of different aspects in water quality research, as well as the components of these aspects in various mathematical models will be reviewed in this chapter. Sediment related water quality issues will be reviewed in section 2.1. Due to the fact that not many of the previous investigations were focused on the issue of sediment affected stream water quality problems, the author tried hard to review literature as widely as possible. But still, many of the listed literatures are not as closely related as expected. Existing mathematical models accounting for hydraulics, sediment transport, and water quality will be reviewed in section 2.2. The approach used by and the limitations of these models will be discussed.

2.1. Sediment-Related Water Quality Issues

2.1.1. Sediment-Water Interactions In Impounded Water Bodies

Many of the recent studies on sediment-water interactions focused on water bodies such as lakes, estuaries, and reservoirs (Nairn and Mitsch 2000, van Luijn et al. 1999, Mayers et al. 1999, Lehtoranta 1998, Beddig et al. 1997, Brassard et al. 1997, Kleegerg 1997, Ram et al. 1997, Nowicki 1997, Smaal and Zurbrug 1997, Appan and Ting 1996, Reddy et al. 1996, Ferreira et al. 1996, Cowan and Boynton 1996, Rizzo and Christian 1996, Bertuzzi et al. 1995, Alaoui-Mhamdi et al. 1995, Raaphorst and Kloosterhuis 1994, Keizer and Sinke 1992). While direct applications of these studies on river water quality modeling is not that obvious, they do provide the clearness of the

close relationship between sediment and water quality, and the insight of these relationships.

Nitrogen fluxes were measured from sediment samples and incubated under both oxic and anoxic conditions, and at different temperatures. The contribution of nitrogen and carbon oxidation to the sediment oxygen demand was estimated (van Luijn et al. 1999). Mayer et al. (1999) has reported sediments as a source of nutrients to hypereutrophic marshes of Point Pelee in Canada. It was found that high concentrations of nutrients (4 mg/L of phosphorus and over 20 mg/L of nitrogen) existed in the interstitial water, as a result of decomposition of organic matters. The research concluded that the internal regeneration of nutrients from sediments is the cause of the pond's hypereutrophic conditions. Lehtoranta (1998) conducted studies on sedimentation and sediment-water nutrient fluxes in the eastern Gulf of Finland. He studied the variation in nutrient concentrations of sediment between different accumulation areas, magnitude of net sedimentation and nutrient flux from sediment to water column, and the main factors controlling the binding and release of nutrients by sediments. It was found that the sediments retain phosphorus well because of oxic conditions, and that iron plays an important role in sediment phosphorus binding. Beddig et al. (1997) measured and quantified the nitrogen fluxes at the sediment-water interface in the German Bight. Brassard et al. (1997) studied the relationship between suspended sediment and dissolved metal concentrations in Hamilton Harbour. It was found that contaminants could be released from sediments during resuspension events. An equilibrium adsorption model is adopted in predicting the metal concentration in the water column that would be found during resuspension. Kleeberg (1997) found that despite efforts in phosphorus loading reduction in the inflow, Lake Groβer Müggelsee still demonstrates year to year fluctuating and strong internal phosphorus loading with increasing pore water phosphate concentrations during summer, due to formation of anoxic microlayers at the sediment surface. It is found that phosphorus concentrations in the studied lake were controlled by

the internal processes in the summer and by the inflowing River Nieplitz in the winter. Ramm and Scheps (1997) studied the phosphorus balance of a small lake, Lake Blankensee in Germany. During the period from April 1994 to March 1995, the lake's net export of phosphorus is about a quarter of its input from the upstream river, due to strong internal loading during summer (Ramm et al. 1997). This strongly indicates the sediment's potential to influence quality of the overlying water body. Nowicki's (1997) work revealed a potentially important loss pathway for nitrogen in sediment pore waters. Smaal and Zurburg's (1997) research indicated significant regeneration of nitrogen from sediments. Appan and Ting's (1996) studies show that the retention of high nutrient levels in the overlying water column was due to the flux from the sediments. The dissolved oxygen levels (DO), pH and temperature in the overlying water were believed to be the governing factors of the sediment nutrient flux. It was found that at low DO levels, there was a higher net release of phosphorus from the sediments. Also, pH values were found to play an important role in the final concentration of phosphorus and changes in pH were likely to disturb the equilibrium of phosphorus. Reddy et al. (1996) showed the important role of bottom sediments in releasing nutrients to the overlying water column during wind induced sediment resuspension or by diffusion. Ferreira et al. (1996) studied the accumulation of nutrients in sediments near Marco, and quantified nutrient concentrations in solution and in sediment phases. Mazouni et al. (1996) quantified the oxygen consumption rate, ammonium production rate, and ammonium and dissolved inorganic phosphorus concentrations at the sediment-water interface in a French shellfish-farming lagoon. Sediment-water oxygen and nutrient fluxes were measured in three distinct regions of the Chesapeake Bay at monthly intervals during 1 year and portions of several additional years. The data showed strong spatial and temporal patterns (Cowan and Boynton 1996). Sediment-water exchanges of ammonium, filterable reactive phosphorus, and oxygen under aphotic conditions were determined in the Neuse River Estuary. Obvious interrelationships in their spatial and temporal patterns

are found (Rizzo and Christian 1996). Benthic nutrient fluxes were measured at the Gulf of Trieste. It was found that the nutrient fluxes observed were not correlated significantly to the consumption of oxygen, which suggested that anaerobic oxidation processes were important at the sediment-water interface. The nutrient budget indicated that over 90% of nitrogen and over 50% of phosphorus were recycled at the interface, which suggested that sediment acts as an active source and sink for nutrients. It was found that in the sediment layer with a depth of 0-14 cm, the pore water phosphate concentrations range between 2 to 10 μM, and ammonium concentrations between 20 to 120 μM. Benthic fluxes of phosphate were low and a clear efflux was observed only during periods of high bottom-water temperature and low bottom water oxygen concentration in late summer. Most nitrate fluxes were oriented out of sediments, indicating that nitrification prevailed at the sediment-water interface (Bertuzzi et al. 1995). Alaoui-Mhamdi et al (1995) collected data on phosphorus concentration in the water column at various depths and sedimentation flux and phosphorus release at the sediment-water interface. Raaphorst and Kloosterhuis (1994) found that the phosphate concentration in superficial intertidal sediments is controlled by the sorption kinetics together with diffusion, according to sorption experiments and a dynamic model simulating sorption kinetics. Keizer and Sinke (1992) studied the phosphorus cycling in the Loosdrecht lakes in the Netherlands. It was found that the diffusive release of phosphorus from the sediments was small under both aerobic and anaerobic conditions. Inorganic phosphorus was removed from the sediments by downward seepage and uptake in the nutrient cycle during resuspension. The amount of potentially bioavailable phosphorus in the top layer of sediments is less than 15% of the total phosphorus content of the sediments, but is still three times larger than the external phosphorus load.

2.1.2. Sediment-Water Interaction In Streams And Overland Flow

Studies on sediment-water exchange of dissolved and particular material in streams and overland flow also indicates strong ties between the behavior of sediments and of different water quality constituents (Nairn and Mitsch 2000, Croke et al. 2000, Craft et al. 2000, Dezzeo et al. 2000, UusiKamppa et al. 2000, White et al. 2000, Correll et al. 1999, Hondzo 1998, Russell et al. 1998, Walling et al. 1997, Emmerth and Bayne 1996, Heathwaite and Johnes 1996a,b, Garban et al. 1995, House et al. 1995, Santiago et al. 1994, Caraco et al. 1993, Johnes and Burt 1991, Kozerski et al. 1991, Caraco et al. 1989).

Nairn and Mitsch (2000) studied phosphorus removal in two different created wetland ponds in an agricultural and urban watershed. It was found that both wetlands significantly reduced turbidity and enhanced dissolved oxygen, and that the retention of both dissolved reactive phosphorus and total phosphorus was significant. The results strongly implies that the removal of phosphorus is related to that of suspended solids. Nutrient movement was found closely related to the transport of sediments due to overland flow in the native eucalypt forest of southeastern Australia. It was found that the nutrient yields were significantly reduced by delivering the runoff over a cross bank, which had the ability to trap suspended sediments (Croke et al. 2000). Craft et al. (2000) studied the sediment and nutrient accumulation in floodplain and depressional freshwater wetlands of Georgia, USA. They suggested that higher sediment and nutrient accumulations likely reflect the absence of environmentally sound agriculture practices in southeastern Georgia. Dezzeo et al. (2000) studied the deposition of suspended sediments and their associated nutrients in a single flood event of 1995 in the seasonally flooded forests of the Mapire and Caura Rivers, Venezuela. They found that the total amount of deposited nutrients varied over a big range, which is affected by the amount of sediments deposited. UusiKamppa et al. (2000) presented findings regarding phosphorus removal in buffer zones, constructed wetlands, and ponds in Finland, Sweden, and

Denmark. It was found that the retention of total phosphorus was affected by the width of the buffer zones and the surface-area/water shed-area ratio of the constructed wetlands. White et al. (2000) studied the storage of phosphorus by sediment in a northern prairie wetland which received municipal and agro-industrial wastewater. They found that 60% of phosphorus inputs into the marsh were stored in the sediments and that the isotherms showed that the sediments near the inflow had a limited ability for additional phosphorus sorption.

Transport of nutrients during storm events were studied in the Rhode River watersheds (Correll et al. 1999). Concentrations of different nutrients were measured during 76 storm events. The author made an attempt to correlate the concentrations with discharge, though no modeling effort was made to explain the changes of the concentrations. Russell et al. (1998) found that the particulate nitrogen and phosphorus fluxes closely reflect the suspended sediment dynamics of the studied basins, which is controlled by basin size, morphology, and hydraulic conditions. They also speculated that the desorption of nutrients from suspended sediment may be important during storm events. The study of sediment-associated nutrient transport of four contrasting river basins in Great Britain showed that significant differences existed in both specific nitrogen and phosphorus; this reflected variations in the magnitude of the specific suspended sediments yields and in the relative importance of point and nonpoint sources (Walling et al. 1997). A study showed that the upper subbasin area, from Franklin, Georgia, to the headwater of the Chattahoochee River, contributed to 96% of the discharge and 97% of total phosphorus and total suspended solid loads into West Point Lake (Emmerth and Bayne 1996), a fact that strongly suggests the links between the two water quality constituents. It is found that phosphorus transport increases in proportion to the magnitude of the suspended sediment load (Heathwaite and Johnes 1996a). Adsorption of phosphorus onto sediments is considered an important mechanism in the cycling of phosphorus, and phosphorus transformations are considered strongly linked to

the cycling of sediments (Heathwaite and Johnes 1996b). Exchanges at the sediment-water interface in the River Seine, France, were quantified by experiments determining the exchanges accounting for all chemical, biological and physical processes (Garban et al. 1995). Kozerski et al. (1991) found that in the lowland river of River Spree, the transport of most nutrients and pollutants is biologically affected and that sediment play an important role in the process. Caraco et al. (1989, 1993) studied the phosphorus release from sediments affected by the presence of sulphate and found that sulphate concentration of waters was extremely important in controlling phosphorus release from sediments.

House et al. (1995) studied the uptake of inorganic phosphorus to suspended sediments and stream bed sediments. The research reveals the importance of suspended sediments in the fast uptake of solute reactive phosphorus, contrasting the control of influx to the bed sediment in channels with low concentrations of suspended solids. Santiago et al. (1994) studied the various pollutants bound to both suspended and bed sediments in the Rhone River basin. They concluded that suspended sediments are more suitable for synoptic monitoring of the environment quality of the river basin. Johnes and Burt (1991) conducted an analysis on nitrogen loading and possible sediment interactions in the Windrush catchment, United Kingdom. They found that the delivery and cycling of nutrients are significantly affected by suspended sediments. They concluded that sediment transport to surface waters is an important mechanism of the delivery of ammonium, particulate organic nitrogen, and phosphorus. They also concluded that the resuspension of bed sediments and erosion of bank sediments affects the recycling of nutrient fractions adsorbed to sediments.

Hondzo (1998) studied the detailed transfer mechanism of the dissolved oxygen at the sediment water interface. The research found that the shear stress velocity controls the thickness of the diffusive boundary layer and thus the mass transfer coefficient for dissolved oxygen transfer at the interface.

2.1.3. Processes Of Adsorption-Desorption

The adsorption isotherm for bromide was studied (Korom 2000). Column experiments were done to determine the nature of bromide adsorption onto and desorption from a sediment sample from Savannah River Site, South Carolina. It was found that adsorption and desorption were reversible and repeatable, and that adsorption followed an apparent Freundlich isotherm.

Brouwers (1999) proposed a one-dimensional nonequilibrium transport model which describes the unsteady mass transfer between flushing medium and solid phases in soil columns. The Freundlich relationship was used to account for the nonlinear sorption of contaminant on the solids.

It is found that the internal phosphorus loading and resuspension of sediment are strong stabilizing factors supporting the maintenance of shallow eutrophic turbid lakes. The dissolved phosphorus concentration at the interface between the top layer and the deeper sediment is related to sediment phosphorus according to the Langmuir equation. The three parameters of this relation are the maximum adsorption capacity of the sediment, the Langmuir adsorption constant, and the ratio between adsorbed and total phosphorus (van der Molen et al. 1998).

Hartikainen et al. (1996) studied the adsorption-desorption equilibrium between solid and solute phophorus in aerobic and anaerobic sediments. It was found that the sediments have high hydrated aluminum and iron oxides and as a result, had high phosphorus adsorption capacity and a low phosphorus concentration. It was also found that phosphorus and silicon were bound to the same components and competed in the adsorption processes.

Froelich (1988) studied the reversible two-step mechanism for phosphate adsorption onto sediment surfaces and found that the phosphate concentration of unperturbed turbid rivers were controlled near the dynamic equilibrium phosphate

concentration of their particles and quantified the amount of phosphate that can be desorbed from suspended sediment surface.

Meyer (1979) found that the sorption of dissolved phosphorus is an equilibrium process, and that the buffering capacity of the sediments was affected by several factors. Silty sediments were found to have higher buffering capacity than sandy sediments and the phosphorus sorption increased as aluminum and organic matter content increased.

2.1.4. Some Modeling Practice Regarding Sediment-Water Exchange

Mathematical models addressing the interaction between sediment layer and overlying water body were developed (Cerco and Seitzinger 1997, Gent et al. 1996, Engelhardt et al. 1995, James and Bierman 1995, Garsdal et al. 1995, Crabtree et al. 1995, Lin et al. 1995, Rutherford et al. 1995, Crabtree et al. 1994, Mol et al. 1994, Blom and Toet 1993, Cerco and Cole 1993, Coghlan et al. 1993, McGregor et al. 1993, Smits 1993, Janse et al. 1992, van der Molen 1991, Kamp-Nielsen, 1989, Beck 1985, Berner 1980).

Blom and Winkels (1998) proposed the combined use of STRESS-2d (Sediment Transport, Resuspension and Sedimentation in Shallow lakes) and DIASPORA (Diagenesis of Aquatic Sediments and Dispersion of Pollutants due to Resuspension and Sedimentation in Aquatic Ecosystems) in studying sediment and contaminant in a lake in the Netherlands. The combined model was capable of simulating resuspension, erosion, sedimentation, and changes of contaminant concentration.

A combination of models – hydrodynamic, sediment diagenesis, benthic algal, and eutrophication – was fitted to test the seasonal algal influence on the transfer of nutrients to and from sediments (Cerco and Seitzinger 1997).

Younger et al. (1993) studied the role of streambed sediments as a barrier to groundwater pollution using a three-dimensional model. It was found that the

groundwater quality would only be damaged had the river water been heavily polluted for a significantly long period.

Janse et al. (1992) proposed a dynamic eutrophication model called PCLOOS. The model had components of algal groups, zooplankton, fish, detritus, zoobethos, sediment detritus and inorganic phosphorus fractions. The studied lake showed a delayed response to the decreased phosphorus loading until a new equlibrium was reached in five years. The reasons for the lake's resilience in responding to the load reduction were the high phosphorus assimilation efficiency of the cyanobactreia and the high internal recycling of phosphorus.

Van der Molen (1991) proposed a model (SED) describing the release of phosphorus from sediments in shallow eutrophic lakes. SED has only three state variables, dissolved phosphorus, organic particulate phosphorus, and active sediment layer, and simple relationships between the variables and a limited number of parameters. The model considered diffusive transport of dissolved phosphorus to the overlying water, mineralization of organic phosphorus in the sediment, and adsorption/desorption and fixation of inorganic phosphorus. SED is able to replicate phosphorus retention in the studied lake and generate values for the net phosphorus release from the sediment with realistic magnitude and seasonal pattern.

Boers et al. (1991) found that in the shallow Lake Loosdrecht in the Netherlands, the concentrations of phosphate and ammonia were strongly correlated, and the ratio between them was close to that predicted by a mineralization model. It was found that pore water concentrations are valuable indicators for early diagenetic processes of phosphate. The similarity in concentration patterns for phosphate and ammonia indicated that phosphate concentrations were maily governed by the processes of mineralization and diffusion.

Beck (1985) made an attempt to reveal the dynamics of nutrient loading and sediment resuspension for Lake Balaton, Hungary. It was concluded that sewage

discharges are important sources of soluble phosphorus in the Zala catchment, and that further understanding of the in-stream phosphorus-cycle process and the role of marshland drainage as a source of phosphorus are important.

However, many of the studies and modeling practices did not take into account an important factor that affects water composition of studied lakes and impounded water bodies, nutrient transport in the streams that drain into them. Van der Perk (1996) specifically studied the impact of bed sediments on the overlying river water body and revealed that the adsorption-desorption processes and the transformation of nutrients have significant effects on the river water composition.

2.2. The Existing Models

HEC-6 is a one-dimensional movable open channel flow numerical model for the simulation and prediction of river profile changes resulting from scour and deposition over moderate time periods. River geometry is represented by a series of cross sections and the distances between the cross sections. Each cross section is described by a transverse section profile composed of points (with relative transversal distances and elevations as coordinates). This typical approach is also adopted by most river models. This segmentation is the basis for the hydraulics, sediment transport, and water quality. In a one-dimensional approach, hydraulic variables are assumed uniformly distributed in a wetted cross-section, i.e. the value of a variable remains the same at different locations in a cross-section at any time. Any changes happen along the longitudinal direction or against time. In the hydraulic computation, a pseudo-steady-state approach is adopted. A continuous flow record is partitioned into a series of steady flows of variables and durations. In other words, the upstream inflow is considered a series of flow with different constant flow rates. Further, the flow passing the entire simulated reach is considered steady. The equations governing steady flows are used in solving the

hydraulics. Based on the computed steady-state hydraulics, sediment transport is computed. It should be noted that the sediment transport rate computed is in fact the sediment transport potential, which is the actual sediment transport rate for steady flows. In the case where flows remain constant, the simplifications for both hydraulics and sediment transport are acceptable and actually quite computationally efficient. But, if the studied reach has a frequent changing hydraulic condition, these simplifications may lead to significant misrepresentation of the reality by the model. There is no water quality components incorporated in HEC-6 (USACE 1996).

GSTARS 2.0 was developed by Molinas and Yang (1986) for the Bureau of Reclamation, U.S. Department of Interior. The model is also intended to simulate hydraulics, sediment transport, and the resulting channel morphological changes. Unlike HEC-6, which is a pure one-dimensional model, GSTARS 2.0 simulates the flow conditions in a semi-two-dimensional manner. This is achieved by using the concept of stream tubes on top of a one-dimensional backwater model. The base one-dimensional model is similar to the hydraulic model used in HEC-6. Inflows are approximated by a series of step functions, and the flows passing the whole reach are considered steady. The energy equation is used when the flow is subcritical, and the momentum equation is used when there is supercritical flow. Sediment transport potential is computed thereafter. Unlike HEC-6, GSTARS 2.0 does not take the sediment transport potential as the actual transport rate, instead, a non-equilibrium approach is adopted in computing the actual sediment transport rate (Han 1980, Han and He 1990). No water quality component is incorporated in the model (Yang et al. 1998).

Fluvial-12 is formulated for the simulation of hydraulics, sediment transport, and resulting river channel changes. This one-dimensional model was developed by Howard H. Chang at San Diego State University. Unlike HEC-6 and GSTARS 2.0, Fluvial-12's hydraulic computation part has a more dynamic and theoretically sound basis. The full St. Venant equations, which govern open channel hydrodynamics, are solved. No

steady-state approximations are made and the simulated flow in the studied reach can be truly unsteady, which is the case in most rivers with fast changing hydraulic conditions. Sediment transport potential is computed afterwards. A non-equilibrium approach is used in computing the actual sediment transport rate (Zhang et al. 1983). There is no water quality component in the model.

QUAL2E is a comprehensive and versatile river water quality model. It has the ability to simulate up to 15 water quality constituents (dissolved oxygen, biochemical oxygen demand, temperature, algae as chlorophyll a, organic nitrogen, ammonia, nitrite, nitrate, organic phosphorus, dissolved phosphorus, coliforms, an arbitrary non-conservative constituent, and three conservative constituents). Unfortunately, the hydraulic component of the model can only deal with steady state flow, limiting the model's application to rivers whose hydraulic conditions do not change dramatically. There is no sediment component in the model. Neither sediment transport nor its possible impact on the other water quality constituents can be simulated by the model.

WASP5 provides dynamic water quality modeling for aquatic systems, including both the water column and the underlying benthos. The model is capable of simulating the behavior of chemical tracer, sediment, dissolved oxygen, eutrophication processes, simple toxicants, and organic chemicals (Ambrose et al. 1993b). The model uses a hydrodynamic model, DYNHYD5, as its hydraulic base. DYNHYD5 does provide solution of the full St. Venant equations, although using an explicit scheme and a somewhat strange representation (Ambrose et al. 1993a). WASP5 does have a sediment transport component. However, sediment scour and deposition are not considered functions of hydraulic conditions, as in most civil-engineering-models. Rather, the user needs to specify the rate of scour and deposition for the sediments. This approach makes the sediment dynamics a constant process, which is not the case for many rivers.

Johnes (1996) and Johnes et al. (1997) used an export coefficient model to study the impact of land use change on nitrogen and phosphorus load to surface waters. The

model provides catchment scale and long-term (annual) prediction with regard to nutrient loading at a certain point on a river. The model predicted well in the two studied reaches. However, no detailed mechanism, especially the mechanism regarding the effects of sediments on nutrient cycling, was present in the modeling approach, although the author, in one of his earlier publications (Johnes et al. 1991) concluded that the nutrient dynamics were affected by sediment behavior.

Bathurst and Purnama (1991) proposed a physically-based modeling system for water, sediment and contaminant transport at the river basin scale, SHETRAN-UK. The model is based on the SHE hydrological modeling system and is intended to simulate soil erosion, sediment yield, and dissolved and particulate contaminant transport. Sediment and contaminant interactions and sediment transport by size fraction are the core components of the model. However, no attempt was made and no data were presented for the calibration and evaluation of the model.

Van der Perk (1996) studied the role of bed sediments during flood waves by analyzing the observed behavior of dissolved nutrients and suspended solids. Phosphate is considered removed from solution by adsorption to bed sediments or suspended sediments (and biological uptake that can not be identified in the model) down to an equilibrium concentration. Phosphate is released from the interstitial water of the stream bed sediments by erosion, and is produced by decomposition of suspended solids.

Like phosphate, ammonium is considered removed from the water phase by nitrification, possible biological uptake, and adsorption to bed and suspended sediment down to an equilibrium concentration. Ammonium is released from the interstitial water in the stream bed sediment during erosion and resuspension, and is produced by decomposition of suspended solids. Nitrate is considered removed from solution by denitrification and is produced by nitrification.

Suspended solids concentration is a state variable of van der Perk's model, but the model was not intended to simulate its transport, which is a key factor affecting the

quality of overlying river water. Considering the fact that approximately 95% of phosphorus in streams tends to adhere to sediment particles (Hem 1985, Meybeck 1982), and the suspended sediment transport rate is usually 4 to 20 times as much as that of bed load sediment (Yang 1996), it is reasonable to speculate that suspended sediment transport plays an important role in a river's transportation of phosphorus.

Review of the literature indicates the important role sediments play in the change of water quality in a river system, and that more can be revealed by more comprehensive modeling practices supported by substantial data sets. A thorough literature review has shown that such data sets have not been reported for the purpose of calibrating a mechanism based model before. Nor was there a comprehensive water quality (based on hydraulics and sediment transport) modeling study reported.

CHAPTER 3

MODEL DEVELOPMENT

3.1. Overall Structure Of Model STAND

The mathematical model Sediment-Transport-Associated Nutrient Dynamics (STAND), as proposed herein, has a three-level structure, as shown in Figure 3-1. The base level contains the hydraulic component, which computes one-dimensional open-channel hydraulics and provides the information to the higher levels of the model for further computation. The second level is the sediment transport level. Sediment transport potentials and actual sediment transport rates are computed and provided to the top level. The water quality component is on the third level, the top level of the model. Here, the variations of ortho-phosphate, nitrate, ammonium, and dissolved oxygen concentrations are computed.

In a mechanism (or physics) based framework of modeling, a river is usually represented by a set of discrete cross-sections in which different kinds of information (geometric, hydraulic, sediment, and water quality) are included. Figure 3-2 shows this kind of representation. Figure 3-3 shows the definition of some of the hydraulic variables that are essential and important in the computation.

3.2. Hydraulic Component

In the hydraulic component of the model, we seek the solution to a one-dimensional open-channel flow with frequent flood waves passing through, a regime

often considered as unsteady, non-uniform flow. The flow is governed by the St. Venant equations, which are listed below.

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} - q_l = 0 \tag{3-1}$$

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x}\left(\frac{Q^2}{A}\right) + gA\frac{\partial Z}{\partial x} + gAs - q_l U_l = 0 \tag{3-2}$$

where

A = wetted cross-section area, (L^2);

Q = cross-section averaged discharge, (L^3T^{-1});

q_l = rate of lateral inflow, (L^2T^{-1});

Z = surface elevation, (L);

g = gravitational acceleration (LT^{-2});

U_l = longitudinal component of lateral inflow velocity, (LT^{-1});

s = friction slope, (-);

x = spatial scale, longitudinal to the reach studied, (L);

t = time scale, (T).

The friction slope in equation (3-2) is usually evaluated using Manning's formula, as presented below.

$$s = \frac{n^2 Q^2}{A^2 R^{4/3}} \tag{3-3}$$

where

n = Manning's roughness coefficient, ($L^{-1/3}T$);

R = hydraulics radius, (L), see Figure 3-3.

Usually, the St. Venant equations do not have analytical solutions, unless the studied river section has some significant simplifications. Instead, numerical solutions to the equations are often sought. In our model, a Preissmann implicit scheme and an iterative approach are used in solving the equations. Figure 3-4 shows the solution mesh of the Preissmann scheme. Index i and j denote the discretized spatial and temporal scale. Point M is the location in a solution mesh, where partial derivatives are taken. Initially, we know the values of all hydraulic variables along the horizontal base line, which are provided by the initial conditions. As we move from time = 0 to time $\rightarrow \infty$, we seek a solution at time step (j+1) on the basis of known quantities at time step (j), until we reach the time for terminating the simulation.

First, the equations are rewritten and discretized into difference equations. The partial derivative terms with respect to space and time are listed below.

$$\frac{\partial \Phi}{\partial x} \approx \frac{1}{\Delta x_i}\left[q\left(\Phi_{i+1}^{j+1} - \Phi_i^{j+1}\right) + (1-q)\left(\Phi_{i+1}^{j} - \Phi_i^{j}\right)\right] \qquad (3\text{-}4)$$

$$\frac{\partial \Phi}{\partial t} \approx \frac{1}{2\Delta t_j}\left(\Phi_{i+1}^{j+1} + \Phi_i^{j+1} - \Phi_{i+1}^{j} - \Phi_i^{j}\right) \qquad (3\text{-}5)$$

where

F = hydraulic variables of interest (Q, Z, A, R, P etc., as defined in equation (3-2) and Figure 3-3);

Dx = distance between two consecutive cross-sections, (L);

Dt = time interval for computation, (L);

q = a numerical parameter with the value between 0.5 and 1, (-) as shown in Figure 3-4.

The partial derivatives in equation (3-1) and (3-2) are substituted by the difference expression in equation (3-4) and (3-5). Two difference equations are resulted for each point M in the solution mesh at a certain time step (j). They have the form

$$F_i(Q_i^{j+1}, Z_i^{j+1}, Q_{i+1}^{j+1}, Z_{i+1}^{j+1}) = 0 \tag{3-6}$$

$$G_i(Q_i^{j+1}, Z_i^{j+1}, Q_{i+1}^{j+1}, Z_{i+1}^{j+1}) = 0 \tag{3-7}$$

where, F and G denote functions of Q and Z.

Equations (3-6) and (3-7) contain four variables and can not be solved independently. If we denote the number of cross-sections along the studied reach with N, we have (N-1) points M, and thus (2N-2) equations with 2N unknowns at a given time step (j). The two boundary conditions provide the last two equations to complete the system. The upper boundary condition is the discharge-time relationship, which reads

$$G_0 = Q_1^{j+1} - Q_{up}(j+1) = 0 \tag{3-8}$$

where $Q_{up}(j+1)$ is the discharge at the upstream entrance of the studied reach at time step (j+1).

At the downstream end, the stage-time relationship is provided by

$$F_N = Z_N^{j+1} - Z_{down}(j+1) = 0 \tag{3-9}$$

where $Z_{down}(j+1)$ is the stage at the downstream end of the studied reach at time step (j+1). Or, if the rating curve at the downstream end is known, instead of the stage-time relationship, the downstream boundary is given by

$$F_N = Z_N^{j+1} - Z_{ofQ}\left(Q_N^{j+1}\right) = 0 \qquad (3\text{-}10)$$

where Z_{ofQ} is the rating curve function.

Equation (3-8) and (3-9) or (3-10), together with the (2N-2) equations, provide a complete system that is solvable.

$$\begin{cases} G_0\left(Z_1^{j+1}, Q_1^{j+1}\right) = 0 \\ F_1\left(Z_1^{j+1}, Q_1^{j+1}, Z_2^{j+1}, Q_2^{j+1}\right) = 0 \\ G_1\left(Z_1^{j+1}, Q_1^{j+1}, Z_2^{j+1}, Q_2^{j+1}\right) = 0 \\ \cdots \\ F_i\left(Z_i^{j+1}, Q_i^{j+1}, Z_{i+1}^{j+1}, Q_{i+1}^{j+1}\right) = 0 \\ G_i\left(Z_i^{j+1}, Q_i^{j+1}, Z_{i+1}^{j+1}, Q_{i+1}^{j+1}\right) = 0 \\ \cdots \\ F_N\left(Z_N^{j+1}, Q_N^{j+1}\right) = 0 \end{cases} \qquad (3\text{-}11)$$

Equation system (3-11) is a nonlinear system with 2N unknowns. The numerical solution can be obtained by applying a Newton-Raphson iteration algorithm for a nonlinear system of equations. This is an algorithm based on a trial-and-correction procedure. If we denote the true solution as [Z_1^*, Q_1^*, … , Z_i^*, Q_i^*, Z_{i+1}^*, Q_{i+1}^*, … , Z_N^*, Q_N^*]', and the k^{th} trial values as [Z_1^k, Q_1^k, … , Z_i^k, Q_i^k, Z_{i+1}^k, Q_{i+1}^k, … , Z_N^k, Q_N^k]', (the superscripts were omitted, since all the variables here are for time step ($j+1$)) using Taylor's Theorem, (3-11) can be expressed as

$$\begin{bmatrix} G_0\left(Z_1^*, Q_1^*\right) \\ F_1\left(Z_1^*, Q_1^*, Z_2^*, Q_2^*\right) \\ G_1\left(Z_1^*, Q_1^*, Z_2^*, Q_2^*\right) \\ \cdots \\ F_i\left(Z_i^*, Q_i^*, Z_{i+1}^*, Q_{i+1}^*\right) \\ G_i\left(Z_i^*, Q_i^*, Z_{i+1}^*, Q_{i+1}^*\right) \\ \cdots \\ F_N\left(Z_N^*, Q_N^*\right) \end{bmatrix} \approx \begin{bmatrix} G_0\left(Z_1^k, Q_1^k\right) \\ F_1\left(Z_1^k, Q_1^k, Z_2^k, Q_2^k\right) \\ G_1\left(Z_1^k, Q_1^k, Z_2^k, Q_2^k\right) \\ \cdots \\ F_i\left(Z_i^k, Q_i^k, Z_{i+1}^k, Q_{i+1}^k\right) \\ G_i\left(Z_i^k, Q_i^k, Z_{i+1}^k, Q_{i+1}^k\right) \\ \cdots \\ F_N\left(Z_N^k, Q_N^k\right) \end{bmatrix} + J \begin{bmatrix} Z_1^* - Z_1^k \\ Q_1^* - Q_1^k \\ \cdots \\ Z_i^* - Z_i^k \\ Q_i^* - Q_i^k \\ \cdots \\ Z_N^* - Z_N^k \\ Q_N^* - Q_N^k \end{bmatrix} \quad (3\text{-}12)$$

where J is the Jacobian matrix and has the form

$$J = \begin{bmatrix} \frac{\partial G_0}{\partial Z_1} & \frac{\partial G_0}{\partial Q_1} & \cdots & & & & \cdots & 0 \\ \frac{\partial F_1}{\partial Z_1} & \frac{\partial F_1}{\partial Z_1} & \frac{\partial F_1}{\partial Z_2} & \frac{\partial F_1}{\partial Z_2} & & & & \vdots \\ \frac{\partial G_1}{\partial Z_1} & \frac{\partial G_1}{\partial Z_1} & \frac{\partial G_1}{\partial Z_2} & \frac{\partial G_1}{\partial Z_2} & & & & \\ & & \cdots & \cdots & & & & \\ & & \frac{\partial F_i}{\partial Z_i} & \frac{\partial F_i}{\partial Z_i} & \frac{\partial F_i}{\partial Z_{i+1}} & \frac{\partial F_i}{\partial Z_{i+1}} & & \\ & & \frac{\partial G_i}{\partial Z_i} & \frac{\partial G_i}{\partial Z_i} & \frac{\partial G_i}{\partial Z_{i+1}} & \frac{\partial G_i}{\partial Z_{i+1}} & & \\ \vdots & & & & \cdots & \cdots & & \vdots \\ 0 & \cdots & & & & \cdots & \frac{\partial F_N}{\partial Z_N} & \frac{\partial F_N}{\partial Z_N} \end{bmatrix} \quad (3\text{-}13)$$

Generally, the trial values do not make the right hand side of equation (3-11) to be a zero vector. Instead, a residue vector is generated.

$$\begin{bmatrix} G_0\left(Z_1^k, Q_1^k\right) \\ F_1\left(Z_1^k, Q_1^k, Z_2^k, Q_2^k\right) \\ G_1\left(Z_1^k, Q_1^k, Z_2^k, Q_2^k\right) \\ \cdots \\ F_i\left(Z_i^k, Q_i^k, Z_{i+1}^k, Q_{i+1}^k\right) \\ G_i\left(Z_i^k, Q_i^k, Z_{i+1}^k, Q_{i+1}^k\right) \\ \cdots \\ F_N\left(Z_N^k, Q_N^k\right) \end{bmatrix} = \begin{bmatrix} R_1^k \\ R_2^k \\ R_3^k \\ \cdots \\ R_{2i}^k \\ R_{2i+1}^k \\ \cdots \\ R_{2N}^k \end{bmatrix} \tag{3-14}$$

If we denote the $(k+1)^{\text{th}}$ trial values by V^{k+1}, the k^{th} trial by V^k, the difference vector $dZ = V^{k+1} - V^k$, and the residue vector by U, the Newton-Raphson algorithm takes the $(k+1)^{\text{th}}$ trial values to be such that

$$JdZ = -U \tag{3-16}$$

where

$$dZ = \begin{bmatrix} Z_1^{k+1} \\ Q_1^{k+1} \\ \cdots \\ Z_i^{k+1} \\ Q_i^{k+1} \\ \cdots \\ Z_N^{k+1} \\ Q_N^{k+1} \end{bmatrix} - \begin{bmatrix} Z_1^k \\ Q_1^k \\ \cdots \\ Z_i^k \\ Q_i^k \\ \cdots \\ Z_N^k \\ Q_N^k \end{bmatrix} \qquad U = \begin{bmatrix} R_1^k \\ R_2^k \\ R_3^k \\ \cdots \\ R_{2i}^k \\ R_{2i+1}^k \\ \cdots \\ R_{2N}^k \end{bmatrix} \tag{3-17}$$

Equation system (3-16) is a linear system of equations. The coefficient matrix J is a penta-diagonal matrix, thus, the system can be solved for dZ using Gaussian elimination for systems with a banded coefficient matrix. Thus we can obtain the (k+1)th trial values with

$$V^{k+1} = V^k + dZ \tag{3-18}$$

This procedure can be repeated until satisfactory accuracy is achieved.

The solutions at time step (j) provide the status of the modeled reach at time step ($j+1$). We start from time step (0) and obtain solutions at each time step of interest.

3.3. Sediment Transport Component

3.3.1. Computation Of Sediment Transport Potential

Yang's approach (Yang 1973, Yang et al. 1984) is adopted because it provides a straightforward answer to the questions, and thus convenience for incorporation into a mathematical model which is implemented in a high level programming language (C++ in this case).

Yang's total sediment transport potential equation reads

$$\log C_t = 5.435 - 0.286 \cdot \log \frac{wd}{n} - 0.457 \log \frac{U_*}{w} + \left(1.799 - 0.409 \log \frac{wd}{n} - 0.314 \log \frac{U_*}{w}\right) \cdot \log\left(\frac{Vs}{w} - \frac{V_{cr}s}{w}\right) \tag{3-19}$$

where

C_t = total sediment transport potential, (ppm);

w = settling velocity for sediment particles, (LT^{-1}), for simplicity, assumed to be the value when the temperature is 20ºC;

d = representative particle dimension, (L);

n = kinematic viscosity, (L^2T^{-1});

U_* = $(gRs)^{1/2}$, shear velocity, (LT^{-1});

Vs = unit stream power, (LT^{-1});

$V_{cr}s$ = critical unit stream power, (LT^{-1}).

The sediment transport potential can be computed using equation (3-19) at each cross-section at each time step.

3.3.2. Conventional Methods In Computing The Actual Sediment Transport Rate

Chang (1988) used a simplified solution to a two-dimensional convection-diffusion equation to describe the variations of suspended sediment concentrations in his FLUVIAL-12 model.

$$C_s^{i+1} = C_*^{i+1} + \exp\left(-\frac{a\,w}{q}\Delta x\right)\left[C_s^i - C_*^{i+1}\right] \qquad (3\text{-}20)$$

where

C_s = actual suspended concentration, (ML^{-3});

C_* = suspended sediment transport potential (ML^{-3});

a = a coefficient representing the rate at which suspended sediment concentration approaches the potential (-);

q = unit width discharge (L^2T^{-1});

i = spatial location of cross-sections (-).

A similar equation is used in GSTARS 2.0 (Yang et al. 1998).

$$C_s^{i+1} = C_*^{i+1} + \exp\left(-\frac{a\,w}{q}\Delta x\right)\left[C_s^i - C_*^i\right] + \left(\frac{q}{a\,w\Delta x}\right)\left[1-\exp\left(-\frac{a\,w}{q}\Delta x\right)\right]\left[C_*^i - C_*^{i+1}\right] \qquad (3\text{-}21)$$

The suspended sediment settling and entrainment are considered spatially in equation (3-20) and (3-21). However, all terms for actual sediment concentration and transport potential are of the same time step. In other words, no quantity of the prior time

step is considered, which implies that there is no effect of earlier behaviors on the current sediment concentrations. This implication does not seem entirely satisfactory, given the following analysis.

Consider Chezy's formula and a sediment transport potential formula widely used in China, given by equation (3-22) and (3-23).

$$v = C_{ch}\sqrt{RJ} \tag{3-22}$$

$$C_* = a\frac{v^3}{gHw} \tag{3-23}$$

where

v = flow velocity, (LT^{-1});

C_{ch} = Chezy coefficient, ($L^{1/2}T^{-1}$);

J = longitudinal slope, (-);

H = depth, (L).

When the ratio of width to depth of the studied river is large, in other words, the river is a wide shallow one, which is often the case, R is approximately the same as H. We can substitute R with H and rewrite equation (3-22).

$$J = \frac{v^2}{C_{ch}H} \tag{3-24}$$

When there is a prior entrainment, H increases so that then J must decrease. As a result, the sediment transport potential given by equation (3-23) will decrease. The fluvial process is adjusted toward the direction of preventing further entrainment. On the other hand, when there is a prior deposition, H decreases and J increases consequently,

both of which will cause an increase in the transport potential. The fluvial process is adjusted toward the direction of preventing further deposition.

It is reasonable to speculate that the quantities of prior time step(s) have some significant effects on the actual suspended sediment transport rate at the current time step. Solving the advection-dispersion equation for suspended sediment using an implicit scheme provides such inclusion.

3.3.3 Computation Of Actual Sediment Transport Rate

An advection-dispersion equation is used to describe the behavior of suspended sediments.

$$\frac{\partial(AC_s)}{\partial t} + \frac{\partial(QC_s)}{\partial x} = \frac{\partial}{\partial x}\left(AE_s \frac{\partial C_s}{\partial x} \right) + AP_s \qquad (3\text{-}25)$$

where

E_s = dispersion coefficient for suspended sediments (L^2T^{-1});

P_s = source/sink term ($ML^{-3}T^{-1}$).

The source/sink term is determined by the rate of scouring, deposition, and transport. We use a first-order mechanism to describe P_s, as shown in equation (3-26).

$$P_s = -k_{sed}(C_s - C_*) + \frac{C_{sedLat}Q_l}{V_{sec}} \qquad (3\text{-}26)$$

where

k_{sed} is a coefficient that describes the rate of the actual suspended sediment concentration's approaching its potential, (T^{-1});

C_* = suspended sediment transport potential, the value of C_t in equation (3-19) is used here, (ML^{-3});

C_{sedLat} = sediment concentration in lateral inflow, (ML^{-3});

Q_l = bulk discharge of lateral inflow, (L^3T^{-1});

V_{sec} = volume of studied section, (L^3).

We hypothesize that k_{sed} has different expressions for deposition and entrainment, which are

$$k_{sed} = k_{sedDep}\left(\frac{l \cdot w}{q}\right) \tag{3-27}$$

for deposition and

$$k_{sed} = k_{sedEnt}\left(\frac{q}{l \cdot w}\right) \tag{3-28}$$

for entrainment, where

k_{sedDep} = sediment deposition coefficient (T^{-1});

k_{sedEnt} = sediment entrainment coefficient (T^{-1});

l = characteristic length, we use the distance between consecutive cross-sections in our model (L).

The equation is solved numerically using a four-point implicit scheme. The finite difference forms for the terms in equation (3-25) are

$$\frac{\partial C_s}{\partial t} = \frac{C_i^{j+1} - C_i^j}{\Delta t} \tag{3-29}$$

$$\frac{\partial C_s}{\partial x} = \frac{C_{i+1}^{j+1} - C_{i-1}^{j+1}}{2\Delta x}$$

$$\frac{\partial}{\partial x}\left(AE_s \frac{\partial C_s}{\partial x}\right) = A_i^j E_s \frac{C_{i+1}^{j+1} - 2C_i^{j+1} + C_{i-1}^{j+1}}{(\Delta x)^2}$$

Substituting equation (3-29) into (3-25), we have

$$a_i C_{i-1}^{j+1} + b_i C_i^{j+1} + g_i C_{i+1}^{j+1} = d_i \tag{3-30}$$

where

$$a_i = -\frac{Q_i^j}{2\Delta x} - \frac{A_i^j E_s}{(\Delta x)^2}, \quad b_i = \frac{A_i^j}{\Delta t} + \frac{2A_i^j}{(\Delta x)^2}, \tag{3-31}$$
$$g_i = \frac{Q_i^j}{2\Delta x} - \frac{A_i^j E_s}{(\Delta x)^2}, \quad d_i = \frac{A_i^j C_i^j}{\Delta t} + A_i^j P_{Si}^{\ j}$$

for all cross-sections except the very first one. Equation (3-30) and two boundary conditions form a linear system of equations that can be expressed in matrix form.

$$\begin{bmatrix} b_1 & g_1 & & & & & \cdots & 0 \\ a_2 & b_2 & g_2 & & & & & \vdots \\ & a_3 & b_3 & g_3 & & & & \\ & & \cdots & \cdots & \cdots & & & \\ & & & a_i & b_i & g_i & & \\ & & & & \cdots & \cdots & \cdots & \\ \vdots & & & & & a_{n-1} & b_{n-1} & g_{n-1} \\ 0 & \cdots & & & & & a_n' & b_n' \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ \cdots \\ C_i \\ \cdots \\ C_{n-1} \\ C_n \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \\ d_i \\ \\ d_{n-1} \\ d_n \end{bmatrix} \tag{3-32}$$

The upper boundary condition is a given time series of suspended sediment concentration. Thus, $b_1 = 1$, $g_1 = 0$, $d_1 = C_{up}(t)$, the suspended sediment concentration in the incoming flow as a function of time. We assume a linear spatial change at the end of the studied reach, which is

$$C_{n+1}^{j+1} = 2C_n^{j+1} - C_{n-1}^{j+1} \tag{3-33}$$

where C_{n+1}^{j+1} is a linearly extrapolated concentration for the hypothetical cross-section which is just outside the downstream end of the studied reach. Thus we can derive the expressions for a_n' and b_n' as

$$a_n' = a_n - g_n, \quad b_n' = b_n + 2g_n \tag{3-34}$$

Equation system (3-32) has a tri-diagonal coefficient matrix, and can be solved using a Gaussian elimination method for systems with a tri-diagonal coefficient matrix. We start from t = 0, and compute concentration values of time step ($j+1$) using computed values of time step (j), until the desired termination time.

3.3.4. Morphological Adjustment Of Bed

Morphological changes are made based on the magnitude of deposition and entrainment. This process is governed by the sediment continuity equation (Chang 1988), which reads

$$(1-l)\frac{\partial A_b}{\partial t} + \frac{\partial Q_s}{\partial x} - q_s = 0 \tag{3-35}$$

where

l = porosity of bed sediments (-);

A_b = cross-sectional area within some arbitrary frame (L^2);

Q_s = volumetric discharge of bed material sediments (L^3/T);

q_s = rate of lateral sediment inflow (L^2/T).

The two partial derivative terms in equation (3-35) can be written into difference form.

$$\frac{\partial A_b}{\partial t} = \frac{\Delta A_b}{\Delta t}, \qquad (3\text{-}36)$$

$$\frac{\partial Q_s}{\partial x} = \frac{1}{2\Delta x_{i-1}}\left(Q_{s\,i}^{\,j} + Q_{s\,i}^{\,j+1} - Q_{s\,i-1}^{\,j} - Q_{s\,i-1}^{\,j+1}\right)$$

$$q_s = \frac{1}{2}\left(q_{s\,i}^{\,j} + q_{s\,i}^{\,j+1}\right)$$

where

Dx_{i-1} = distance between cross-section (i) and (i-1), (L); and

the superscripts and subscripts are in accordance with the convention of the computational mesh.

Equation (3-35) can be rewritten as

$$\Delta A_b = \frac{\Delta t}{l-1}\left[\frac{1}{2\Delta x_{i-1}}\left(Q_{s\,i}^{\,j} + Q_{s\,i}^{\,j+1} - Q_{s\,i-1}^{\,j} - Q_{s\,i-1}^{\,j+1}\right) - \frac{1}{2}\left(q_{s\,i}^{\,j} + q_{s\,i}^{\,j+1}\right)\right] \qquad (3\text{-}37)$$

Equation (3-37) provides the means of computing the change of bed cross-sectional area, which is the result of deposition or entrainment. The adjustments on river bed elevation are computed by

$$\sum_P \Delta z \cdot \Delta y = \Delta A_b \qquad (3\text{-}38)$$

where

Dz = vertical adjustment of bed elevation along the wetted perimeter, (L);

Dy = length of a section of the wetted perimeter, (L);

P = wetted perimeter, (L), as defined in Figure (3-3).

3.4. Water Quality Component

3.4.1. Phosphorus Dynamics

The transport and exchange of ortho-phosphate is governed by the following advection-dispersion equation. The mechanism is also described conceptually in Figure 3-5.

$$\frac{\partial(AC_P)}{\partial t} + \frac{\partial(QC_P)}{\partial x} = \frac{\partial}{\partial x}\left(AE_P \frac{\partial C_P}{\partial x}\right) + AP_P \qquad (3\text{-}39)$$

where

C_P = phosphate concentration, (ML^{-3});

E_P = phosphate dispersion coefficient, (L^2T^{-1});

P_P = sources/sink term for phosphate, ($ML^{-3}T^{-1}$).

Considerations for the P_P term may include adsorption of ortho-phosphate onto the surface of particles, vertical diffusion caused by concentration gradient, erosion-induced release of interstitial water, and possible biomass uptake. At the present time, adsorption-desorption and erosion-induced interstitial water release are incorporated. A first-order mechanism for the adsorption-desorption processes is described by equation (3-40).

$$P_P = -k_P\left(C_P - C_{Peq}\right) + C_{Pint}\frac{l}{1-l}\frac{S_{ent}}{V_{sect}} \qquad (3\text{-}40)$$

where

k_P = ortho-phosphate adsorption coefficient, (T^{-1});

C_{Peq} = a dynamic equilibrium ortho-phosphate concentration, which is considered a function of the amount of suspended sediments, (ML^{-3});

C_{Pint} = ortho-phosphate concentration in the interstitial water, (ML^{-3});

l = porosity of bed sediments, (-);

S_{ent} = the rate of sediment entrainment from the bed, (L^3T^{-1});

V_{sect} = volume of water in considered section, (L^3).

We hypothesize that the dynamic equilibrium ortho-phosphate concentration is governed by the following equation.

$$C_s - \frac{C_0 - C_{Peq}}{k_f \cdot \left(C_{Peq}\right)^{p_f}} = 0 \qquad (3\text{-}41)$$

where

C_0 = initial concentration of ortho-phosphate before adsorption/desorption processes (replaced by C_P in the computation), (ML^{-3});

k_f and p_f are constant coefficients in the Freundlich adsorption equation (Parfitt et al. 1983).

Derivation of equation (3-41) is shown below. Assuming a container with volume V holds a conservative solvent with initial concentration C_0 and suspended solids of concentration C_s, we have the total mass of solute as C_0V.

Under an equilibrium condition, the Freundlich adsorption equation can be written as

$$w = k_f C_e^{\,p_f} \qquad (3\text{-}42)$$

where

w = amount of adsorption, (MM^{-1});

C_e = equilibrium concentration of solvent, (ML^{-3});

k_f and p_f are coefficients.

The mass of solute that is adsorbed to the solids can be expressed as wC_sV, and the mass of solute that remains in the solvent is C_0V-wC_sV. The concentration of solvent under equilibrium can be expressed as $(C_0V\text{-}wC_sV)/V = C_0\text{-}wC_s$. Thus we have

$$C_e = C_0 - wC_s \tag{3-43}$$

Substituting w with equation (3-42), we have

$$C_e = C_0 - k_f C_e^{\,p_f} C_s \tag{3-44}$$

Substitute C_e with C_{Peq} and rearrange the equation, we obtain equation (3-41). The equation is in an implicit form for C_{Peq}, and can be solved numerically using a Newton-Raphson method.

The above derivation was based on the Freundlich adsorption equation and the principle of mass conservation. Here we attempted to establish a relationship between nutrient concentrations and the amount of suspended sediments present in the water column. Such a relationship allows the equilibrium nutrient concentration to be a variable, a function of the suspended sediment concentration, rather than a constant, thus allowing more nutrient to be adsorbed to solids when the concentration of suspended solids increases, and vice versa.

Numerical solution of equation (3-39) can be obtained using the same algorithm for solving equation (3-25).

3.4.2. Nitrogen Dynamics

Nitrate Nitrogen

The transport and exchange of nitrate nitrogen is governed by the following equation.

$$\frac{\partial(AC_{nitr})}{\partial t} + \frac{\partial(QC_{nitr})}{\partial x} = \frac{\partial}{\partial x}\left(AE_{nitr}\frac{\partial C_{nitr}}{\partial x}\right) + AP_{nitr} \qquad (3\text{-}45)$$

where

C_{nitr} = nitrate concentration, (ML^{-3});

E_{nitr} = dispersion coefficient for nitrate, (L^2T^{-1});

P_{nitr} = source/sink term for nitrate, ($ML^{-3}T^{-1}$).

The source/sink term is considered the sum of production by nitrification and the consumption by denitrification. It is expressed by the following equation (Ambrose et al. 1993a, Brown and Barnwell 1987, van der Perk 1996).

$$P_{nitr} = -k_{DeNi}C_{nitr} + \frac{O_2}{O_2 + M_O}\cdot k_{Nitr}C_{ammo} \qquad (3\text{-}46)$$

where

k_{DeNi} = denitrification rate constant, (T^{-1});

O_2 = dissolved oxygen concentration, (ML^{-3});

M_O = Monod half saturation concentration for dissolved oxygen, (ML^{-3});

k_{Nitr} = nitrification rate constant, (T^{-1});

C_{ammo} = ammonium nitrogen concentration, (ML^{-3}).

The numerical solution to equation (3-45) is obtained using the same algorithm for solving equation (3-25).

Ammonium Nitrogen

The transport and exchange of ammonium nitrogen is governed by the following equation.

$$\frac{\partial(AC_{ammo})}{\partial t} + \frac{\partial(QC_{ammo})}{\partial x} = \frac{\partial}{\partial x}\left(AE_{ammo}\frac{\partial C_{ammo}}{\partial x}\right) + AP_{ammo} \qquad (3\text{-}47)$$

where

E_{ammo} = dispersion coefficient for ammonium nitrogen (L^2T^{-1});

P_{ammo} = source/sink term for ammonium nitrogen ($ML^{-3}T^{-1}$).

The source/sink term is considered the sum of a first order equilibrium term and a term for the consumption of ammonium due to nitrification. It is expressed by the following equation (Ambrose et al. 1993a, Brown and Barnwell 1987, van der Perk 1996).

$$P_{ammo} = -\frac{O_2}{O_2 + M_O} \cdot k_{Nitr} C_{ammo} - k_{AmAd}\left(C_{ammo} - C_{ameq}\right) \qquad (3\text{-}48)$$

where

k_{AmAd} = ammonium adsorption rate constant, (T^{-1});

C_{ameq} = dynamic equilibrium concentration for ammonium in the case of adsorption to solid material, (ML^{-3}).

The dynamic equilibrium concentration for ammonium is expressed by equation (3-49), which has a form similar to equation (3-41).

$$C_s - \frac{C_{am0} - C_{ameq}}{k_{amf} \cdot \left(C_{ameq}\right)^{p_{amf}}} = 0 \qquad (3\text{-}49)$$

where

C_{am0} = initial ammonium concentration before adsorption/desorption processes (replaced by C_{ammo} in computation), (ML^{-3});

C_{ameq} = equilibrium concentration of ammonium in case of solid's adsorption, (ML^{-3});

k_{amf} and p_{amf} are coefficients in the Freundlich equation.

The numerical solution to equation (3-47) is obtained using the same algorithm for solving equation (3-25).

3.4.3. Dissolved Oxygen Component

The dynamics of dissolved oxygen are governed by equation (3-50).

$$\frac{\partial(AO_2)}{\partial t}+\frac{\partial(QO_2)}{\partial x}=\frac{\partial}{\partial x}\left(AE_{O2}\frac{\partial O_2}{\partial x}\right)+AP_{O2} \qquad (3\text{-}50)$$

where

E_{O2} = dispersion coefficient for dissolved oxygen, (L^2T^{-1});

P_{O2} = source/sink term for dissolved oxygen, ($ML^{-3}T^{-1}$).

The source/sink term is determined by reaeration, photosynthesis and respiration, sediment oxygen demand, and nitrogenous biochemical oxygen demand. The whole process is presented in equation (3-51) (Ambrose et al. 1993a, Brown and Barnwell 1987, van der Perk 1996).

$$P_{O2}=k_{reae}(O_{sat}-O_2)+(a_3m-a_4r)\cdot C_{algae}-\frac{r_{sod}}{R}-a_5b_1C_{ammo} \qquad (3\text{-}51)$$

where

k_{reae} = reaeration rate constant, (T^{-1});

O_{sat} = oxygen saturation concentration, (ML^{-3});

a_3 = rate of oxygen production per unit of algal photosynthesis, (M/M);

m = algal growth rate, (T^{-1});

a_4 = rate of oxygen production per unit of algae respired, (M/M);

r = algal respiration rate, (T^{-1});

C_{algae} = algal concentration, (ML^{-3});

r_{sod} = sediment oxygen demand rate, ($ML^{-2}T^{-1}$);

a_5 = rate of oxygen uptake per unit of ammonia nitrogen oxidation, (M/M);

b_1 = ammonia oxidation rate coefficient, (T^{-1}).

The oxygen saturation concentration is determined by an empirical equation (USACE 1982, Bowie et al. 1985), which reads

$$O_{sat} = 14.6 \exp\left[-\left(0.027767 - 0.00027T + 0.000002T^2\right)T\right] \qquad (3\text{-}52)$$

where

T = temperature, (°C).

The reaeration rate constant is determined by the following empirical equation (Long 1984, Bowie et al. 1985).

$$k_{reae} = 1.923 \frac{v^{0.273}}{R^{0.894}} \qquad (3\text{-}53)$$

where v is stream flow velocity and R is the hydraulic radius.

3.5. Summary

STAND is a comprehensive water quality model that incorporates hydrodynamics, sediment transport, and water quality issues. It steps beyond the traditional civil-engineering models by providing the ability to deal with water quality issues. Furthermore, it provides the functionality of computing sediment behavior, using a fully dynamic approach (as opposed to many other models), and its impact on water quality, which clearly distinguishes it from other contemporary water quality models. The unique features STAND has enable us to apply the model in the study of stream water quality problems associated with sediments.

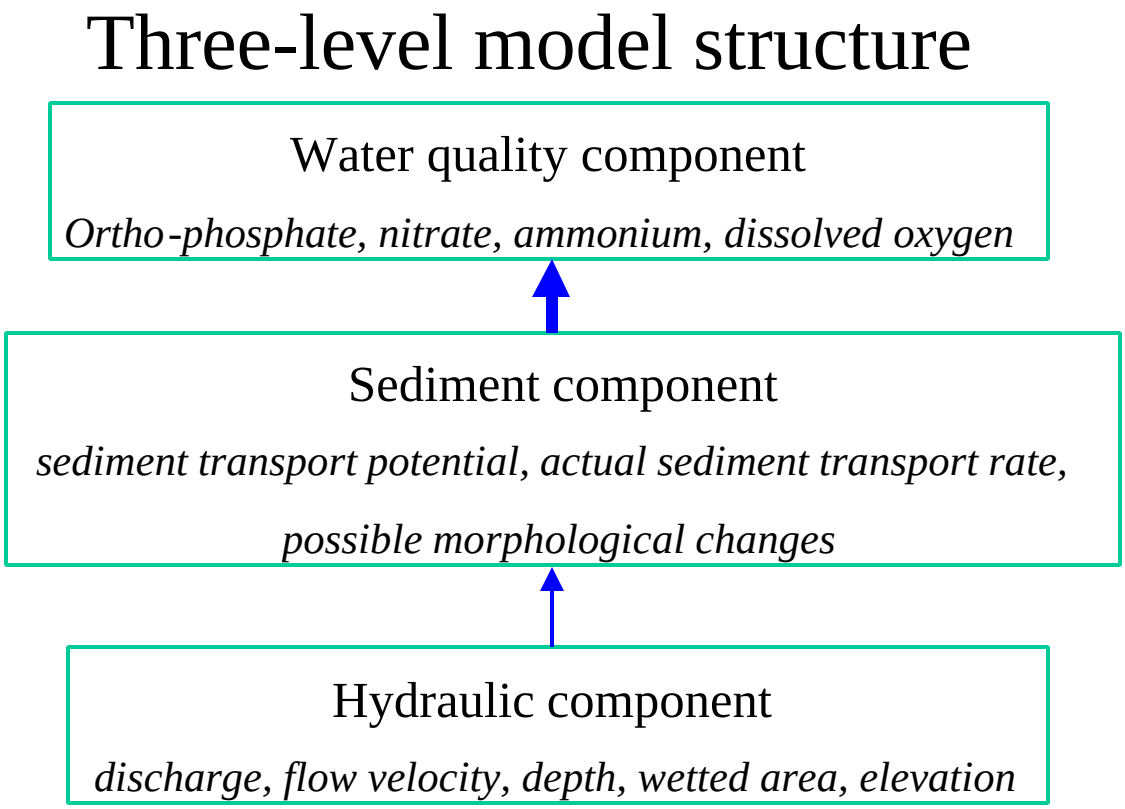


Figure 3-1. Structure of model STAND

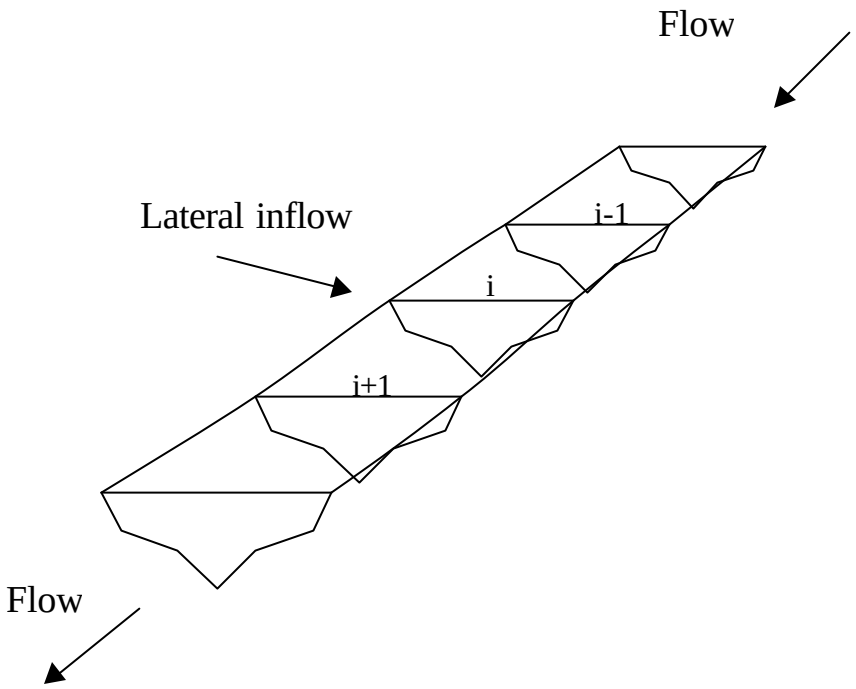
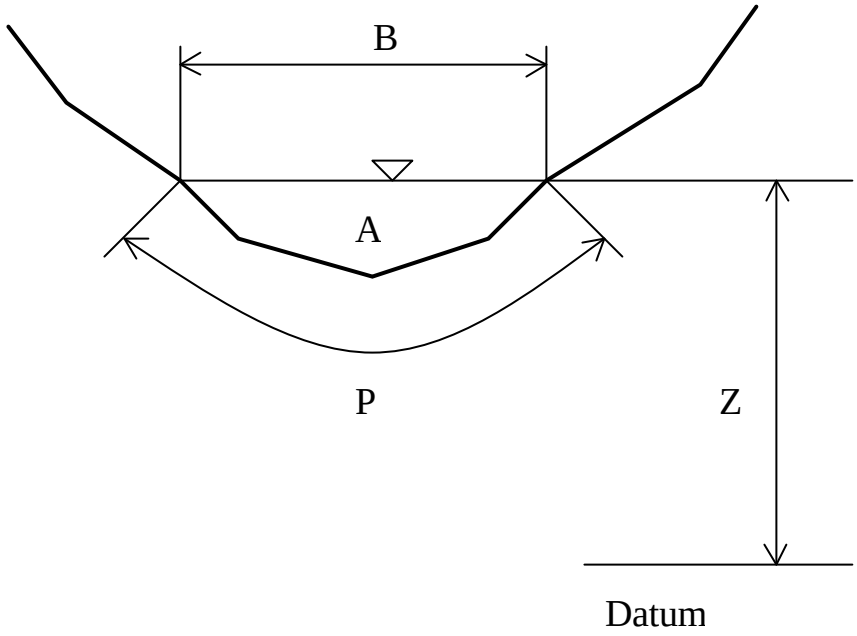


Figure 3-2. Discretized representation of a river section



Definitions:

A = cross-section wetted area;

B = surface width;

D = A/B, hydraulic depth;

P = wetted perimeter;

R = A/P, hydraulic radius;

Z = surface elevation;

Figure 3-3. Hydraulic variables in a cross-section

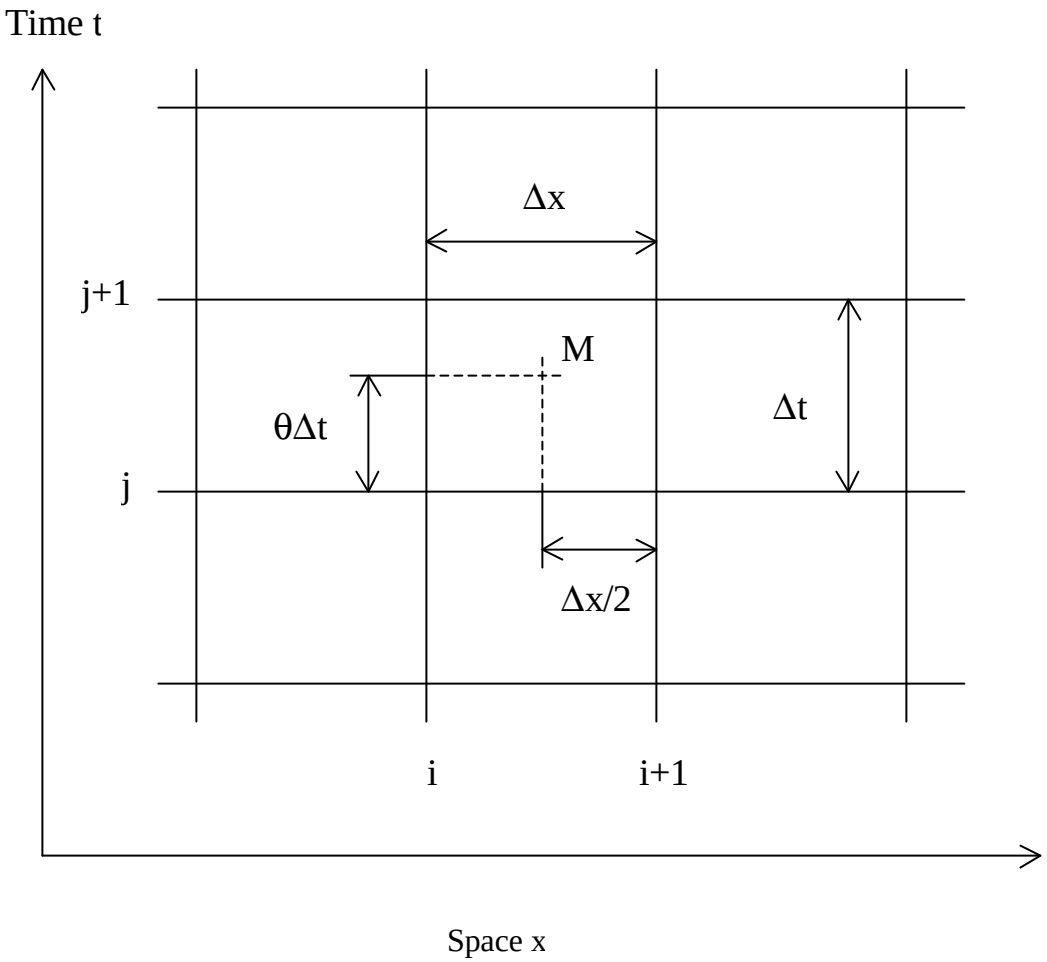


Figure 3-4. Space-time domain for the solution of St. Venant Equations

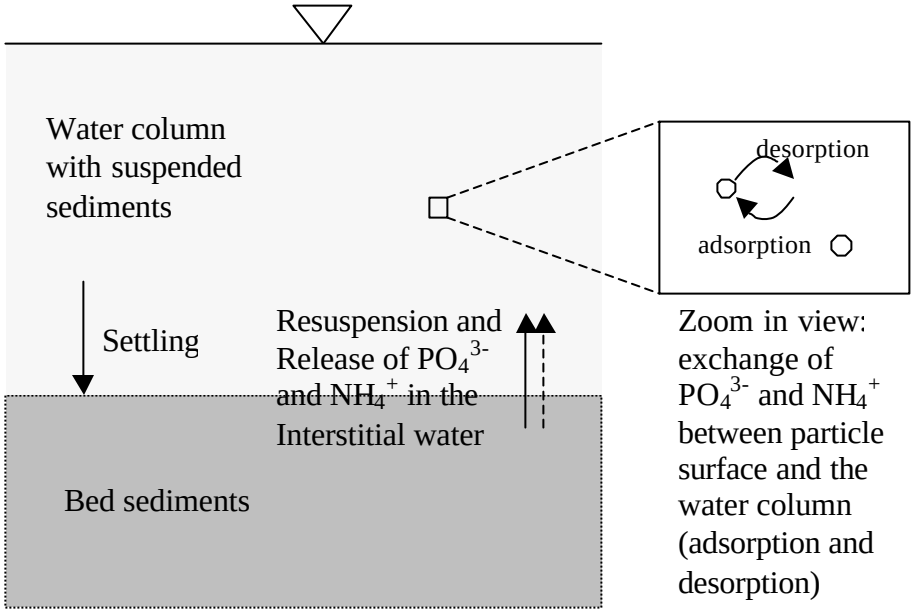


Figure 3-5. Interactions between sediment particles and the water column

CHAPTER 4

FIELD METHOD AND AVAILABLE DATA SETS

In order for the model to be successfully calibrated and evaluated, quality data sets are important. We obtained several data sets of various hydraulic and water quality variables on two rivers. Data sets of hydrodynamics and sediment transport are provided by the Chinese Hydrologic Yearbook (1973, and 1974) for the Weihe River, a major tributary of the Yellow River in China. Data sets of hydrodynamics, suspended sediment concentration, concentrations of ortho-phosphate, nitrate, and dissolved oxygen, among others, are available through two lengthy periods of intensive water sampling and data collection for the Oconee River, in 1998 and 1999.

4.1. The Weihe River

The Weihe River is the largest tributary of the Yellow River, which is known for its extremely high sediment concentrations, with a total length of 818 km and a drainage area of 134,767 km^2. The origin and the upper reach of the river are in a mountainous area. The lower part of the river sits in a vast plain area, which is called the “800 li Chin Area” (a li is a Chinese length unit which is equivalent to 0.5 km; Chin is the name of the first emperor’s dynasty) and historically is famous for its abundance of resources and prosperity. The Weihe valley remains an important area of wealth today. The Weihe River played an important role in maintaining the area’s prosperity. On the other hand, the potential problems caused by flooding pose serious threats to the people and property

in this area. Because of the importance of the area, comprehensive hydrologic data have been collected systematically, persistently, and intensively for the past several decades. Starting from the mouth of the river, where it meets the Yellow River, until 140 km upstream, more than 40 cross-sections were allocated for the purpose of measuring channel morphological changes. Nine gauging stations are scattered along the reach, covering a length of 120 km.

The studied reach is a 77.4 km along section of the Weihe River (Figure 4-1). The upper end of the reach is at Lintong Gauging Station. The lower end is at Huaxian Gauging Station. The average annual amount of water passing Lintong is 7.075×10^9 m^3, and the amount passing Huaxian is 7.362×10^9 m^3. The average annual amount of sediment passing Lintong is 3.49×10^8 ton, and the amount passing Huaxian is 3.54×10^8 ton. Weinan Gauging Station is located in between the upper and lower end of the reach. Hydrologic data were collected at all three stations on a regular basis.

Data sets of discharge, surface elevation, sediment concentration, and flow velocity were measured every 4 or 5 days during regular flow conditions, and were measured several times a day during high flow or high sediment content conditions. Sediment particle size distributions were measured about 10 to 20 times a year. Channel morphological measurements were conducted several times a year. At least two of those measurements cover all the cross-sections in the studied reach.

Data of a 120-day period, taken from the 1973 Chinese Hydrologic Yearbook, were chosen for the calibration of hydrodynamics and suspended sediment transport, because of the extreme hydrologic conditions presented. A similar data set of 140-day period were chosen for the purpose of evaluation. Selected sections of the data are shown in Figures 4-3 through 4-8. Several significant hydrologic events can be seen in the figures. It appears that the studied reach is upstream-driving. The importance of the source/sink will be discussed in Chapter 5.

4.2. The Oconee River

The Oconee River is a major tributary of the Altamaha River in the State of Georgia. It drains an area of about 13,600 km^2 covering 20 central Georgia counties (USACE 1974) including three major cities, Athens, Milledgeville, and Dublin. The main headwater tributaries of the Oconee River are the Middle and North Oconee Rivers which originate in the Piedmont Province about 35 miles (56 km) northwest of Athens. The river flows generally south-southeast for about 209 km through the Piedmont Physiographic Province before crossing the Fall Line at Milledgeville (USACE 1974). The river then flows approximately 89 km through a transition zone know as the Fall Line Hills District before entering the upper Coastal Plain for the remainder of its 129 km passage to a juncture with the Ocmulgee to form the Altamaha River (Hodler and Schretter 1986).

The basin is located in both the Piedmont and Coastal Plain physiographic provinces. In the Piedmont province, the terrain is generally rolling to hilly and contains a few small mountains. In the mountainous areas, the elevation reaches 1,000 feet (305 m) above mean sea level. The province has deeply weathered igneous and metamorphic bedrock, the metamorphic being composed largely of quartzites, schists, and slate with granites and basic and ultra-basic igneous rock interspersed. Red soils predominate with sandy clay and silty clay textures (GADNR 1972).

The streams are relatively narrow, shallow, and with steep gradients, and thus, fast flows. The climate is generally characterized by long, rainy springs, long, warm summers, and short mild winters. The mean annual precipitation is a little more than 1,200 mm (USACE 1977).

Major development centers in the Oconee River Basin occur at Athens, Milledgeville, and Dublin as well as in Hall and Gwinnett counties in the northeastern part of the greater Atlanta metropolitan area (Hodler and Schretter 1986). Smaller areas

of development are found above the Fall Line at Eatonton and at Glenwood/Mt. Vernon near the river terminus (GADNR 1976a). Total population of the Oconee River basin was 327,000 in 1980 and is expected to reach approximately 437,000 by the year 2000, with most growth occurring in the principal urban areas (GADNR 1984).

Due to the absence of major population or industrial centers, nutrient loading and other water quality problems are less pronounced than on the upper Ocmulgee or Chattahoochee rivers (GADNR 1984, Evans 1991, Mauldin and McCollum 1992). Periodic and localized deteriorations in water quality are produced by municipal wastewater treatment facilities, urban non-point source discharges, kaolin processing wastewaters, and by hypolimnetic (deepwater) discharges from Sinclair Dam near Milledgeville (GADNR 1976a). Many of these problems have been reduced in magnitude in recent years, and the entire Oconee River presently supports designated uses established by the Georgia Environmental Protection Division (GADNR 1990).

The reach between Barnett Shoals dam and the United States Geological Survey (USGS) gauging station (No. 02218300) near Penfield was selected as the case study reach (Figure 4-2). Barnett Shoals dam is a hydropower facility operated by the Georgia Power Co. It is located approximately 15 miles (24 km) south of Athens, downstream of the confluence of the North Oconee River and the Middle Oconee River. The USGS station near Penfield is approximately 17 km downstream of Barnett Shoals dam. The two sites were selected as sampling stations for the data collection. There are six small tributaries in between the two sites, with the largest one having a drainage area of 153 km^2.

Two independent sampling campaigns were conducted in the periods from July 15 to August 4, 1998 and from March 14 to May 4, 1999. During the 1998 collection period, electronic pressure sensing equipment was set up immediately above the Barnett Shoals dam to measure the hydraulic head over the spillway. The hydraulic head was converted to discharge using a standard method for a rectangular-crest weir. An attached

data logger recorded readings at 15-minute intervals during the whole data collection period.

Gauge levels at Penfield were measured and posted on the Internet by USGS. These gauge level data can also be converted to discharge using a rating curve for the station given by USGS.

An Isco automatic sampler was set up to collect surface water samples for laboratory analysis of suspended sediment and nutrient contents at each site. Water samples were taken with varied time intervals, depending on the hydrological conditions. The sampling intensity ranged between once a day during low flow conditions, to 12 samples a day during flood events. These samples were taken back from the sites on a daily basis to the Warnell School of Forest Resources, University of Georgia for laboratory analysis. The samples were divided into two groups. The first group was used for suspended sediment analysis. The other group was used for the analysis of dissolved contents.

Suspended sediment content was determined by a gravimetric method. The mass of dry, clean beakers were measured first. Then the beakers were filled with sediment bearing water, and the masses of the beakers were measured. These beakers were than put into an oven with constant temperature of 110°C. The masses of the beakers were measured again 24 hours later, when the moisture content was completely driven away from the samples. Assuming a density of 2.65×10^3 kg/m^3 for the sediments, and a density of pure water for the water content of the samples, we can calculate the sediment concentration of the original samples.

The other group of samples was then analyzed in the Environmental Process Control Laboratory (EPCL) for water quality constituents (Beck and Liu 1998). In the EPCL, concentrations of total organic carbon, ortho-phosphate, ammonium, nitrate, and nitrite were measured. However, some of the water quality data were suspect, because of the unstable performance of the EPCL during the 1998 data collection campaign.

The measured quantities are presented in Figures 4-9 through 4-16. From Figure 4-9 and 4-10, we can see clearly that only a single significant and a couple of moderate hydrologic events were captured in 1998's campaign. This is also reflected in the sediment concentration time series (Figure 4-11), with peak concentrations corresponding to the peak flow. It seems that there were dilution effects with regard to ammonium-N concentrations (Figure 4-12). The spikes in the figure are probably caused by the poor performance of the instrument. Total organic carbon data do show a peak corresponding to the hydrologic event (Figure 4-13), unfortunately, the instrument did not work well for a long period during the first days and thus reliable data of the first 170 hours are not available. Figure 4-14 does not show a lot of variations of ortho-phosphate-P concentration, other than an obvious spike at the downstream end and a sudden decrease around time = 250 hours. Nitrite-N data seem quite good, with a peak corresponding to the flow event (Figure 4-15). Nitrate-N concentration seem to be on a decreasing trend and had a slight increase around the major hydrologic event (Figure 4-16). Unfortunately, the Isco sampler did not work well at Penfield and some data during the major event were not obtained.

In the 1999 sampling campaign, the same equipment was deployed for the measurement of hydraulics and collection of water samples for determining concentrations of suspended sediment and dissolved contents. The EPCL was mainly used in the analysis of ortho-phosphate, because the phosphate sensor proved to be stable and reliable. Two Hydrolab multi-purpose probes were deployed at Barnett Shoals dam and Penfield station, respectively. These probes are capable of measuring nitrate, dissolved oxygen, temperature, pH, specific conductivity, and oxidation reduction potential. There was a period of about two weeks in which both probes took continuous readings at the sites.

The data collected during the 1999 campaign are presented in Figures 4-17 through 4-26. Figure 4-17 shows the discharge time series at the upstream end of the

studied reach. One major and some moderate (all more significant than those presented in the 1998 campaign) hydrologic events were captured in the rain season (spring) of 1999. The downstream elevation data are shown in Figure 4-18. Significant sediment behavior can be seen corresponding to the hydrologic ones (Figure 4-19). It is interesting to see that the suspended sediment concentration at the downstream end during the major event is higher than that of the upstream end, which indicates possible scour and entrainment in between. This contrasts with what was observed in 1998, when the suspended concentration at the upstream end was higher, indicating a possible deposition during the hydrologic event with lower magnitude. Ortho-phosphate-P data are shown in Figure 4-20. Apparent weekly patterns were observed during the first couple of weeks, which was probably caused by the way the Athens Waste Water Treatment Plant was operated. The pattern was somehow broken by the most significant rain event, probably by the dilution effects. After the rain event which happened around time = 500 hours, the pattern seemed to recur, but with much more significant difference between the ortho-phosphate concentrations at the upper end and the lower end. A possible explanation is that, after the rain event the average suspended sediment concentrations were higher than before, and were sustained for the last half of the campaign period; the presence of more suspended sediment provided higher potential for adsorption of ortho-phosphate onto the particle surfaces, and thus reduced the dissolved phase more significantly. Figure 4-21 shows temprature measured in a two-week period (time = 587 ~ 914 hours, just after the major event, but including a moderate event) in the campaign. Rain event caused cooling can be seen. Specific conductivity is shown in Figure 4-22. Diurnal oscillations can be seen clearly. Figure 4-23 shows the variations of pH values at both ends of the reach. It is interesting to see an obvious upstream diurnal oscillation flattened out at the downstream end, and that the oscillations upstream are themselves attenuated at a rain event. The fact that the Waste Water Treatment Plant is upstream of and close to the upper end, and that the rain event had dilution effects may help explain the observations.

Figure 4-24 presents oxidation/reduction potential for the two-week period. Diurnal oscillations can be seen. Figure 4-25 shows dissolved oxygen at both ends of the reach. The obvious diurnal oscillations were probably caused by the temperature-dependent dissolved oxygen saturation concentration, and thus reaeration, and photosynthesis and respiration of algae. A decrease during the rain event (time ≈ 800 hours) can be seen, and this was probably caused by dilution. Nitrate-N concentrations are shown in Figure 4-26. The reason that a peak event at the downstream end happened before it did at the upper end is hard to explain. It might be that the pasture lands in between the two stations discharged animal waste and caused the event at the lower end. The diurnal oscillations are similar to those presented in the pH data.

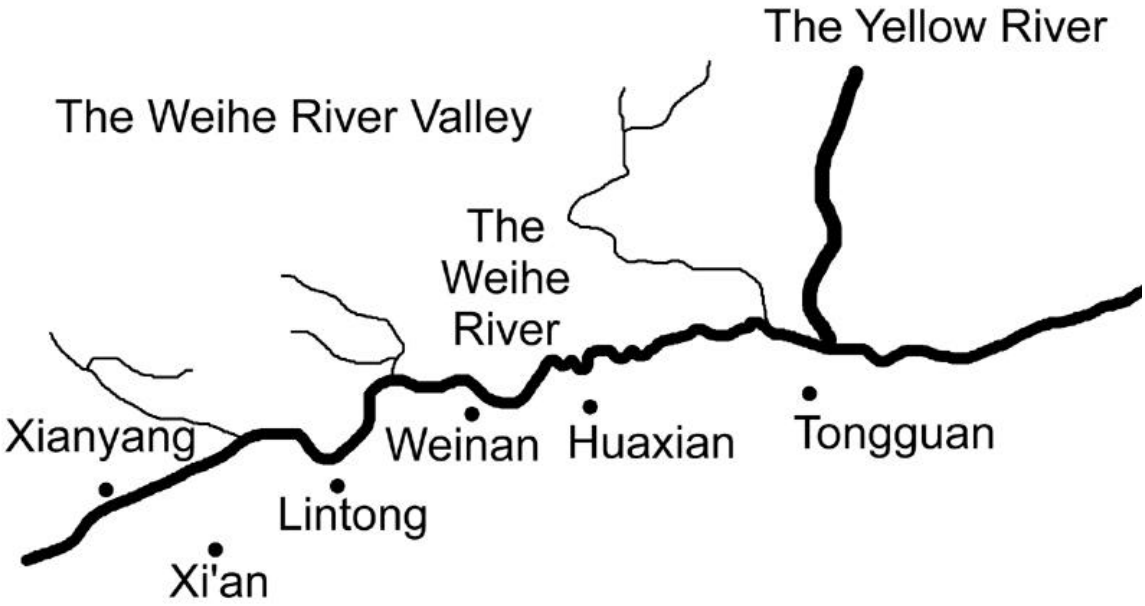


Figure 4-1. The Weihe River

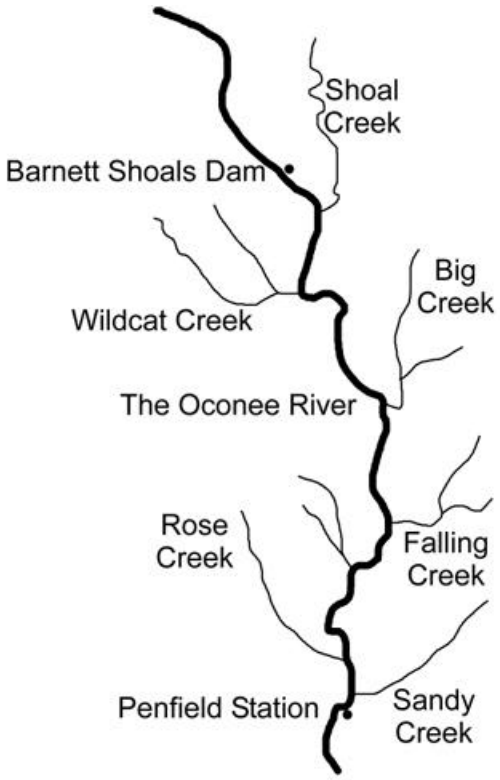
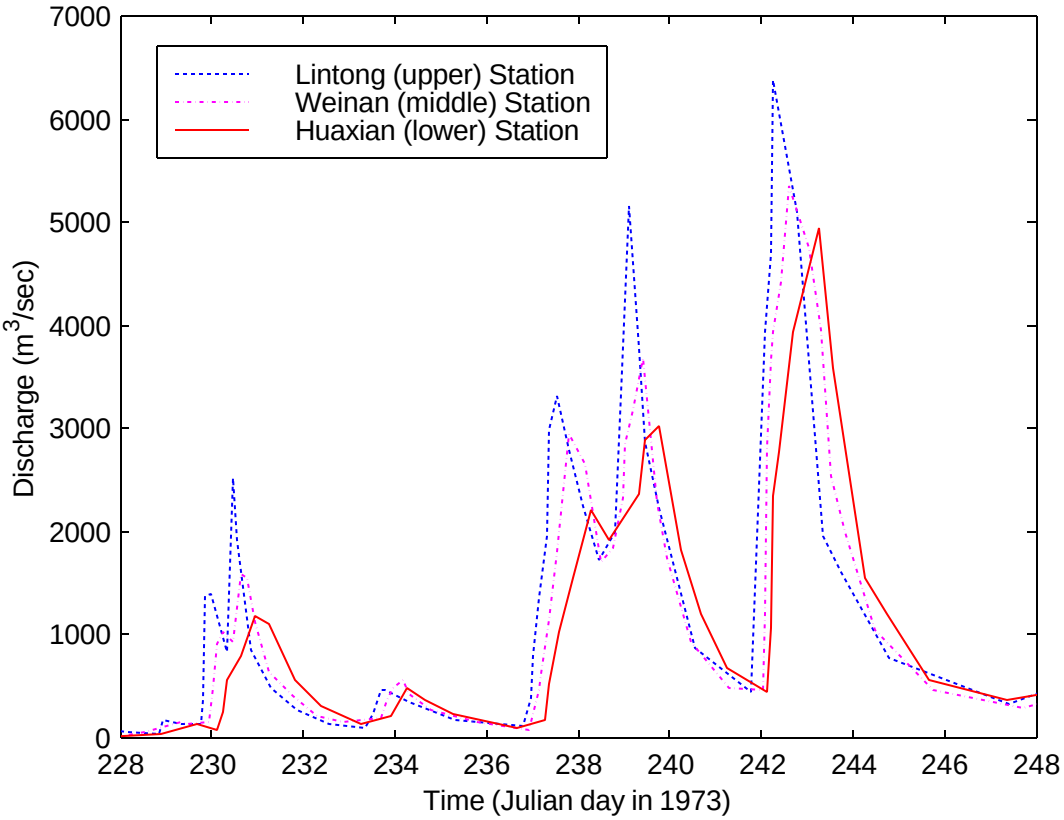
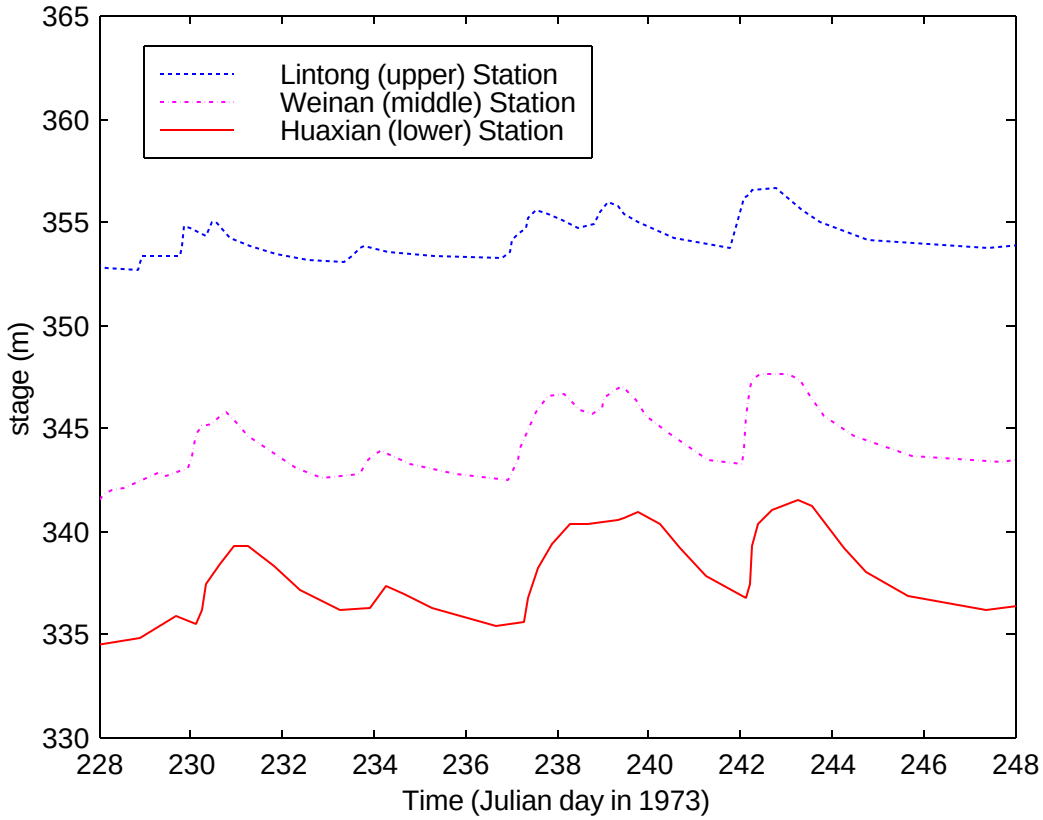


Figure 4-2. The Oconee River



↑ August 16 ↑ September 5

Figure 4-3. Discharge time series on the Weihe River in 1973



↑ August 16 ↑ September 5

Figure 4-4. Elevation time series on the Weihe River in 1973

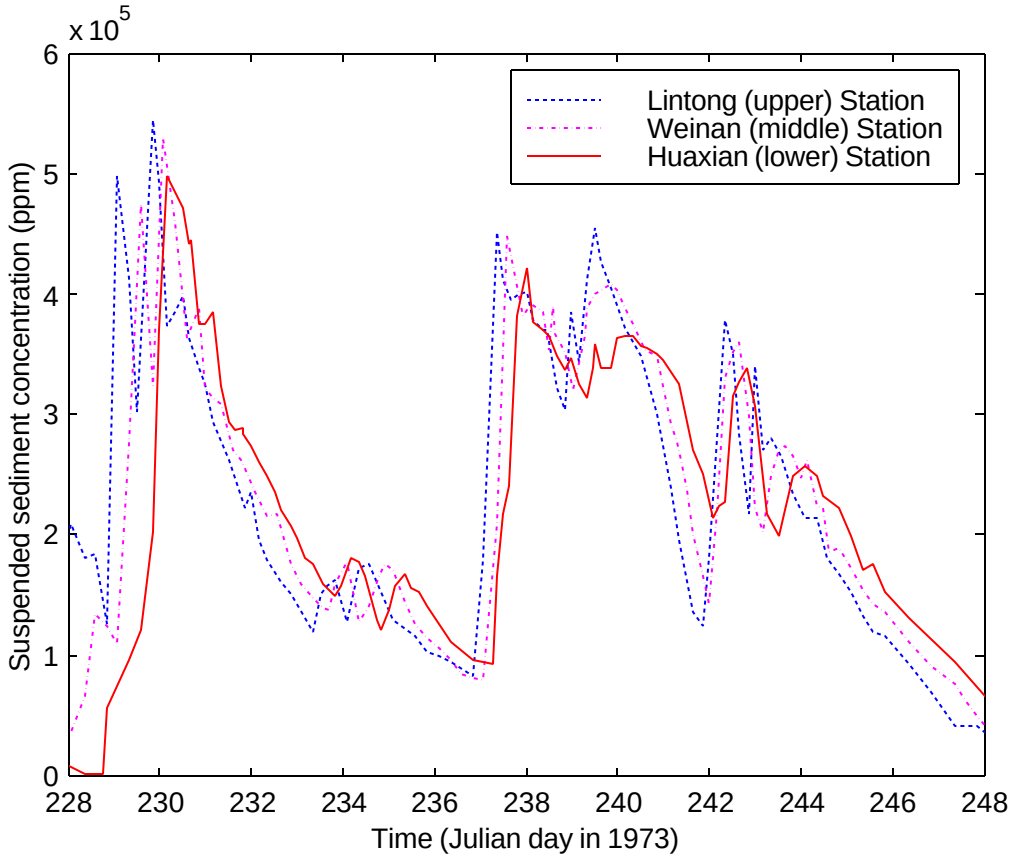


Figure 4-5. Suspended sediment concentrations on the Weihe River in 1973

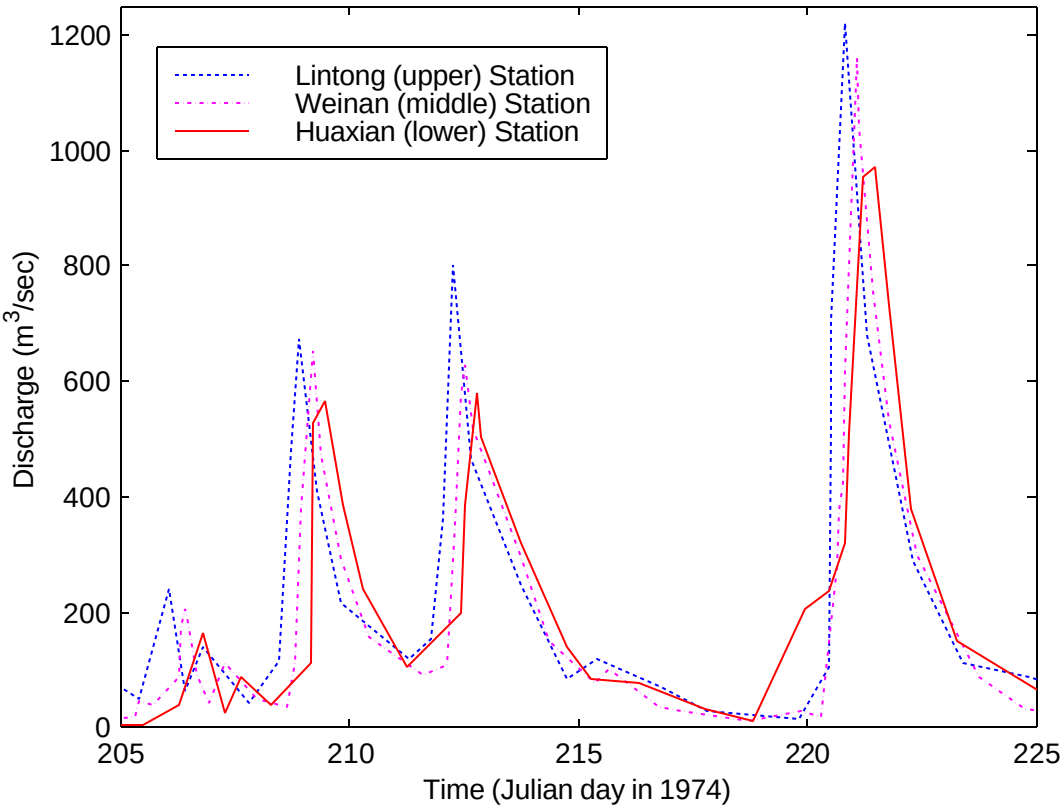
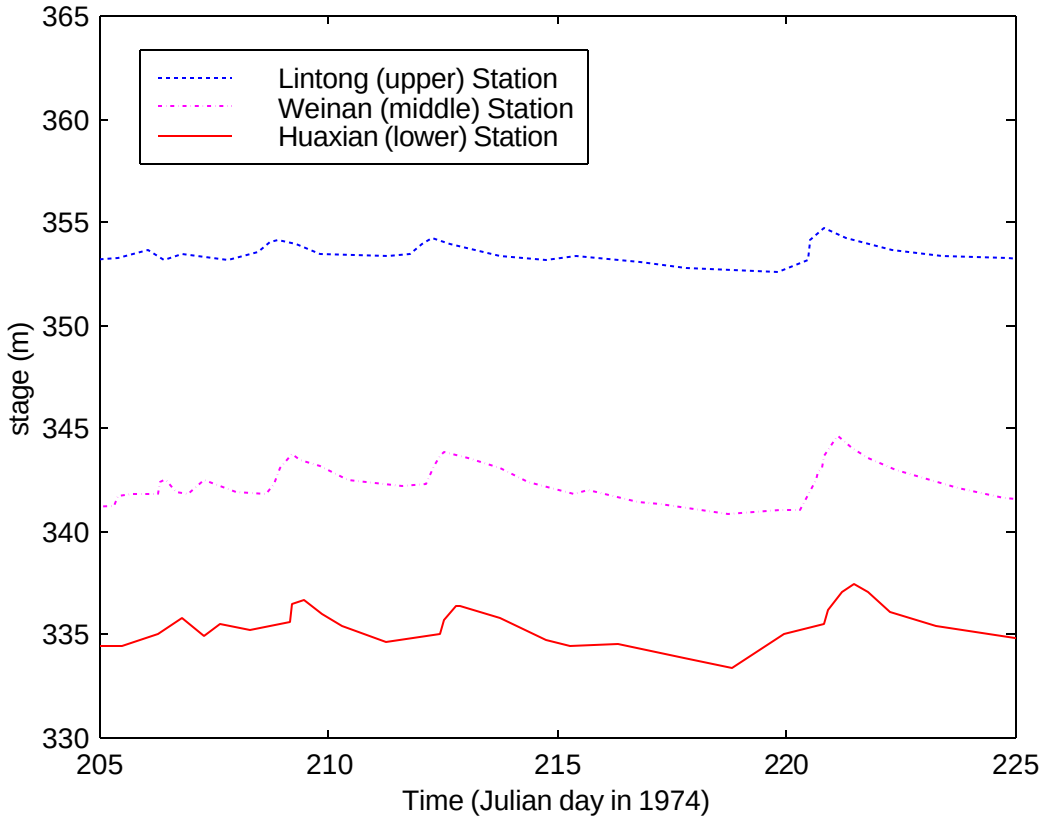


Figure 4-6. Discharge time series on the Weihe River in 1974



↑ July 24

↑ August 13

Figure 4-7. Elevation time series on the Weihe River in 1974

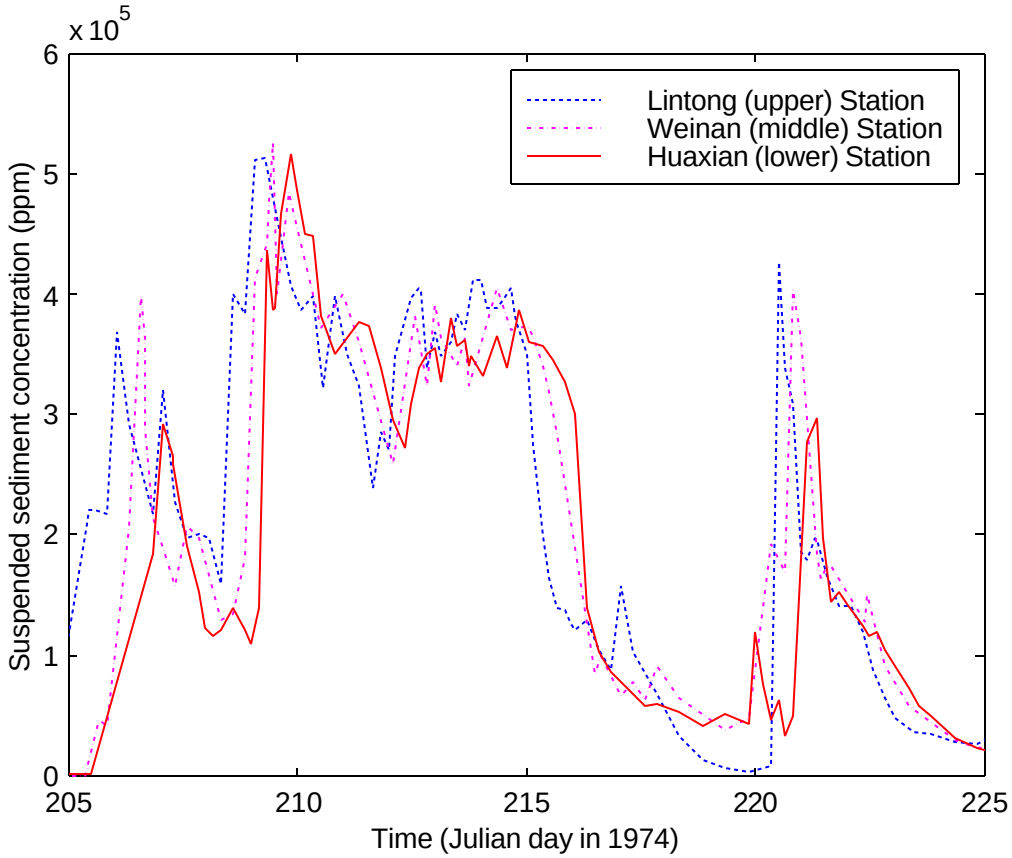


Figure 4-8. Suspended sediment concentrations on the Weihe River in 1974

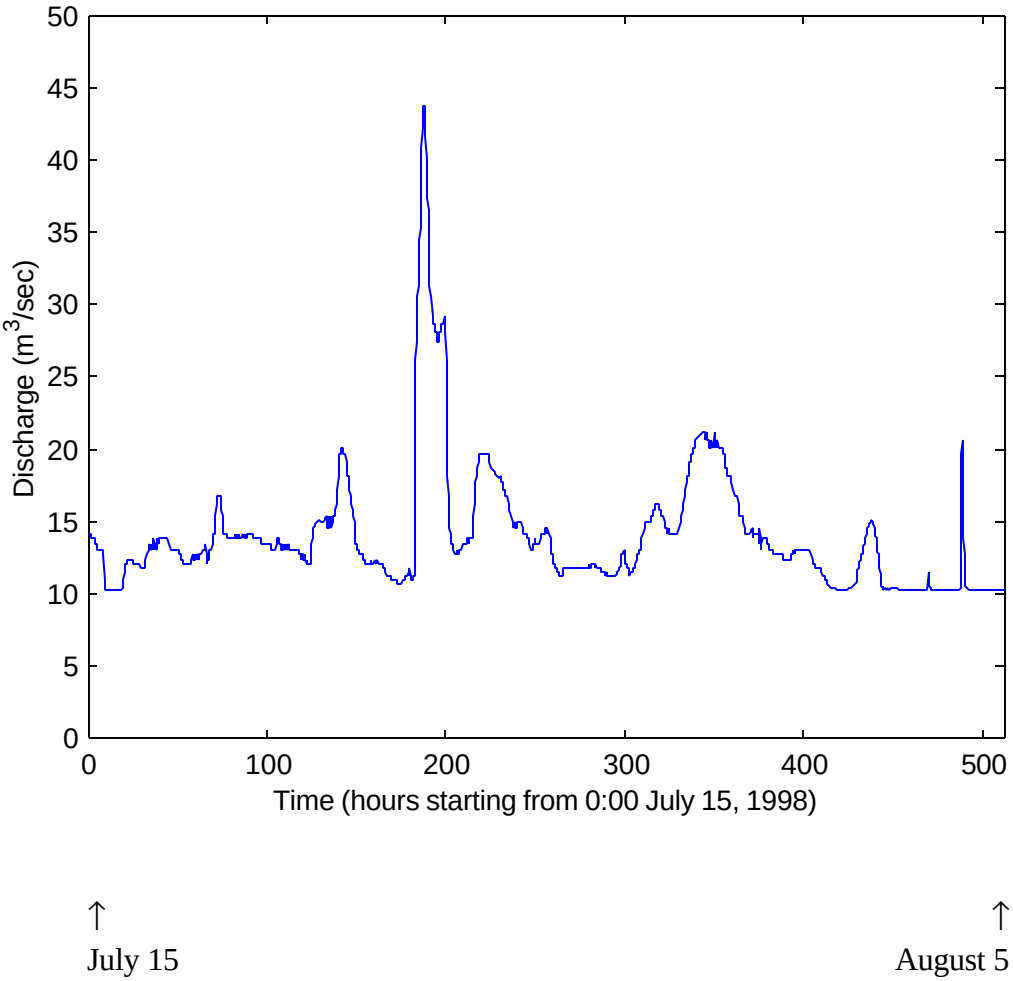


Figure 4-9. Discharge time series at Barnett Shoals dam on the Oconee River, 1998

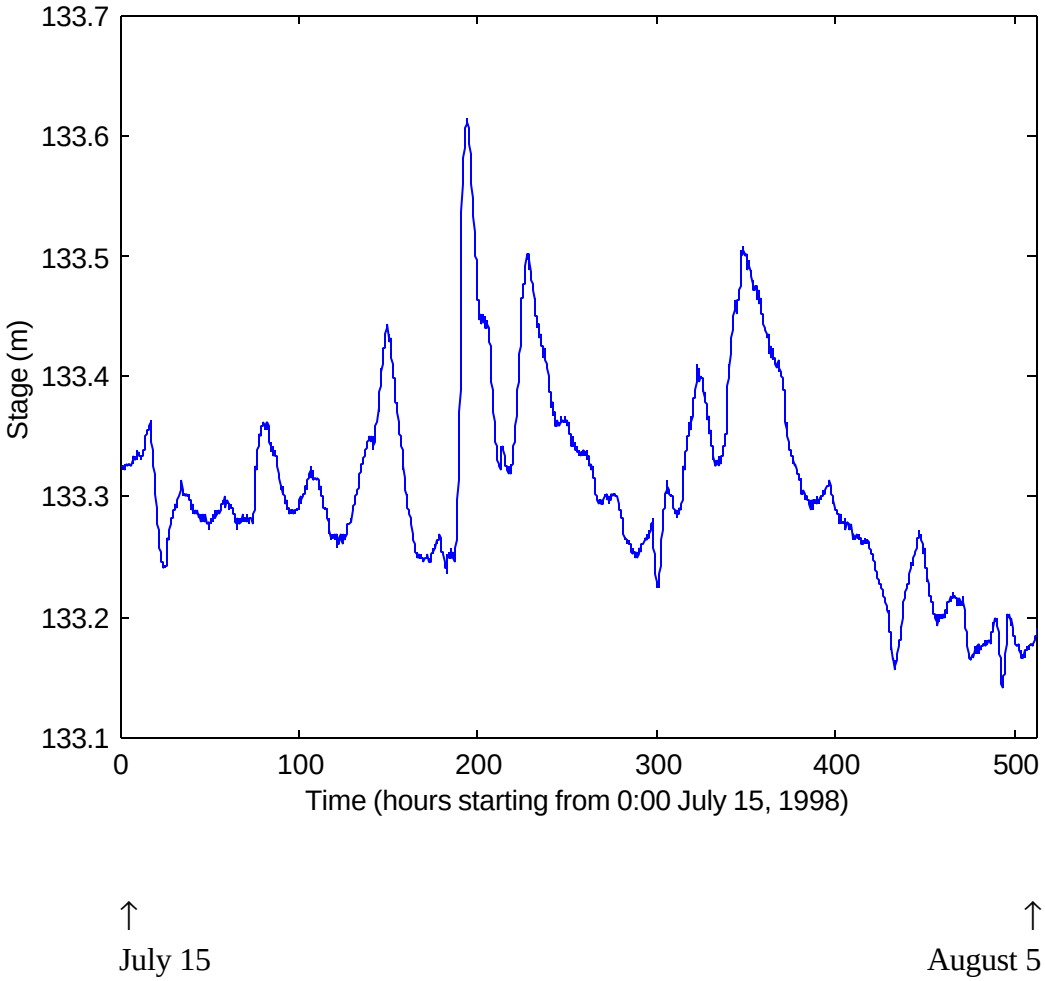
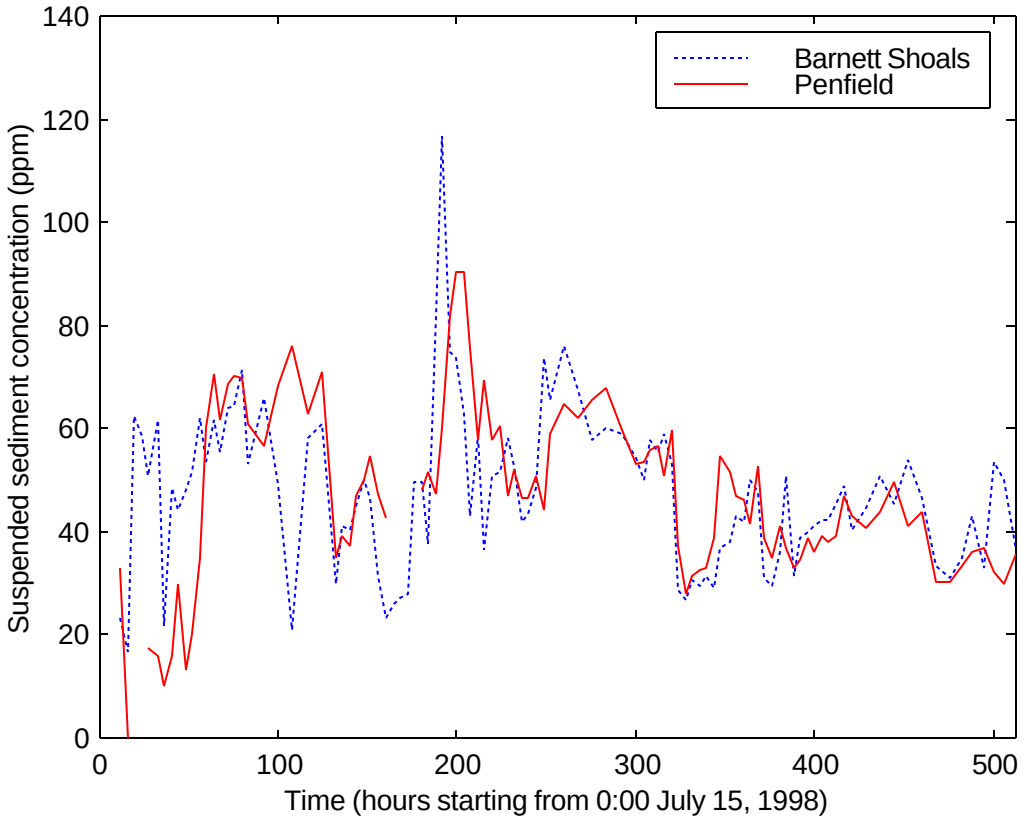


Figure 4-10. Elevation time series at Penfield on the Oconee River, 1998



↑ July 15

↑ August 5

Figure 4-11. Suspended sediment concentration on the Oconee River, 1998

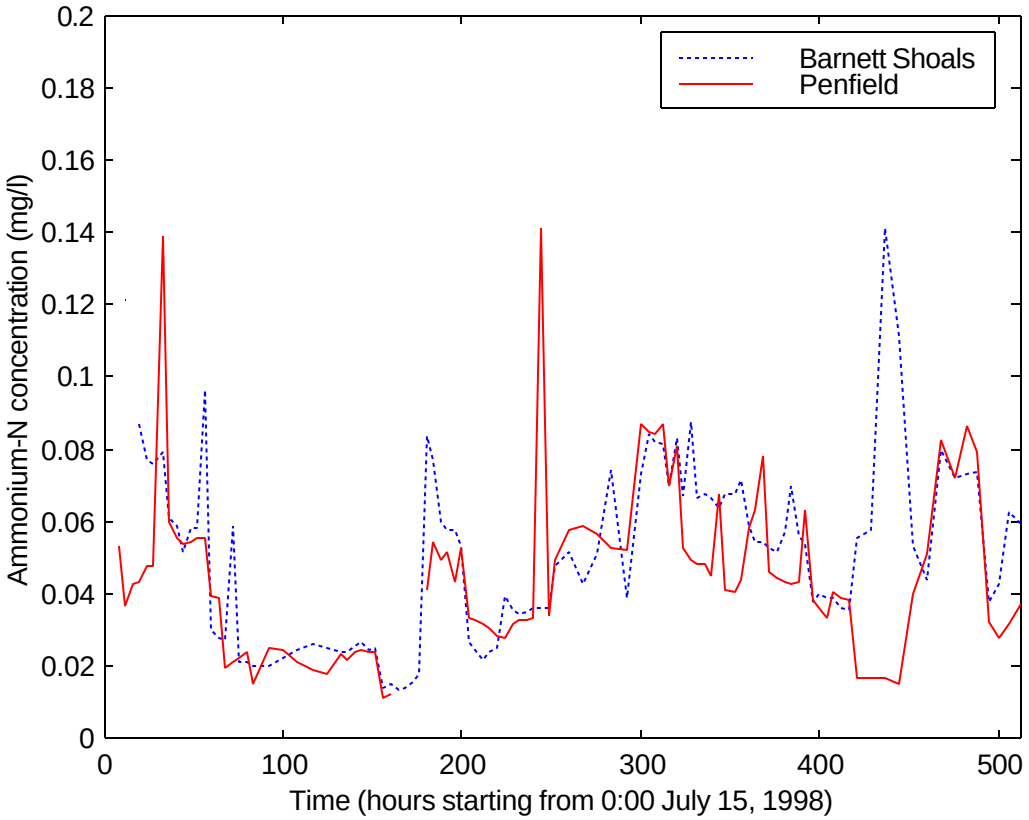
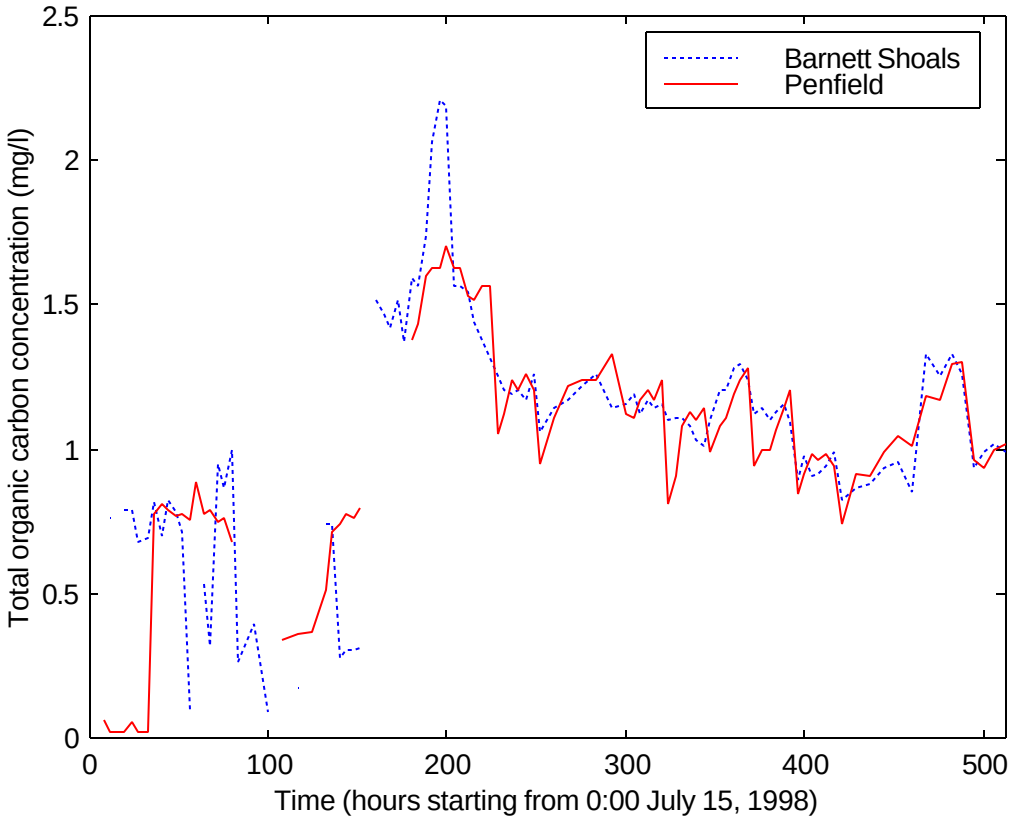


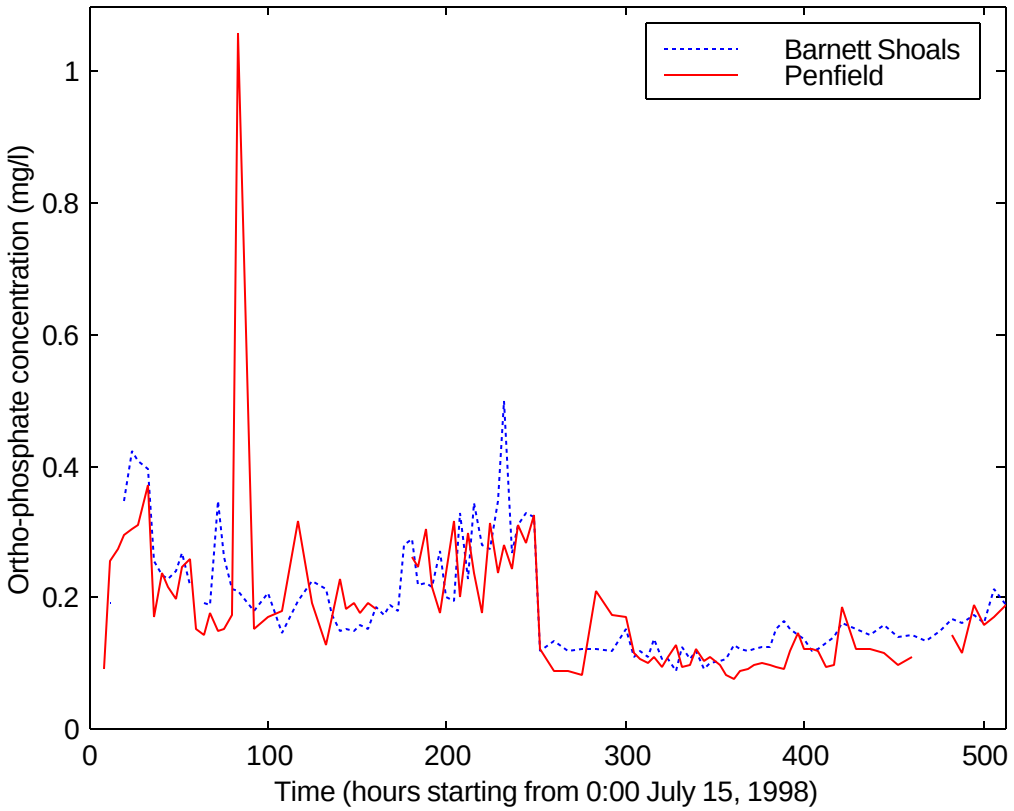
Figure 4-12. Ammonium-N concentration on the Oconee River, 1998



↑ July 15

↑ August 5

Figure 4-13. Total organic carbon concentration on the Oconee River, 1998



↑ July 15

↑ August 5

Figure 4-14. Ortho-phosphate-P concentration on the Oconee River, 1998

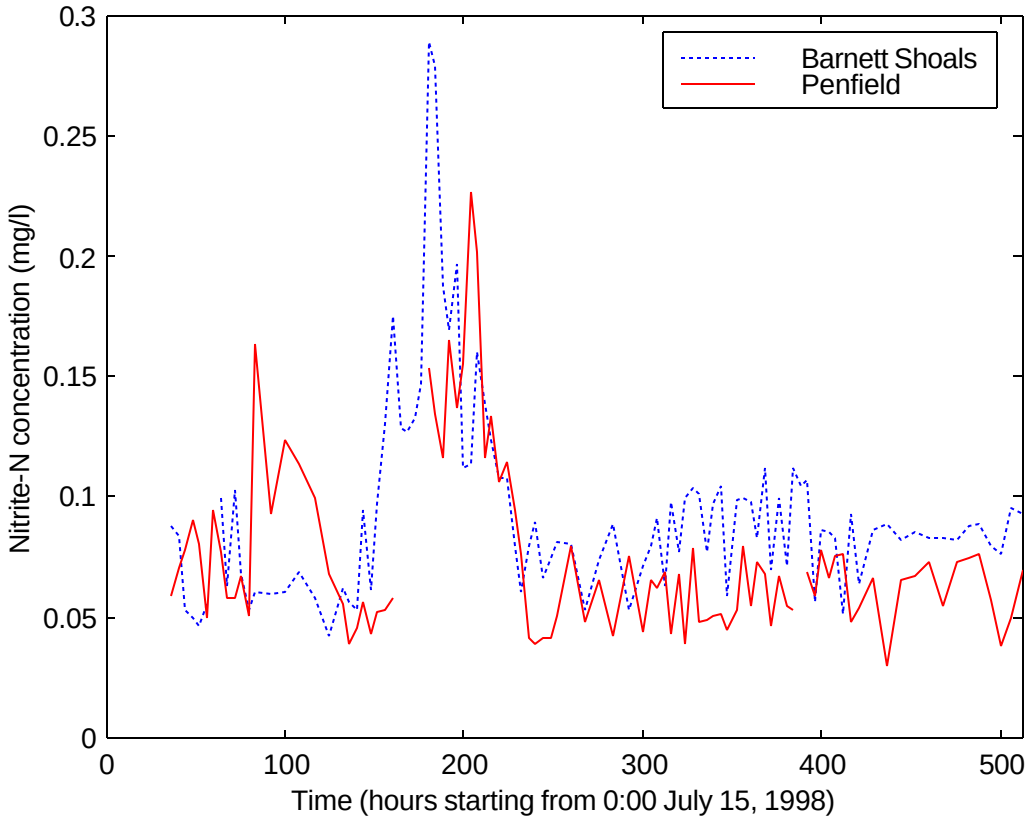
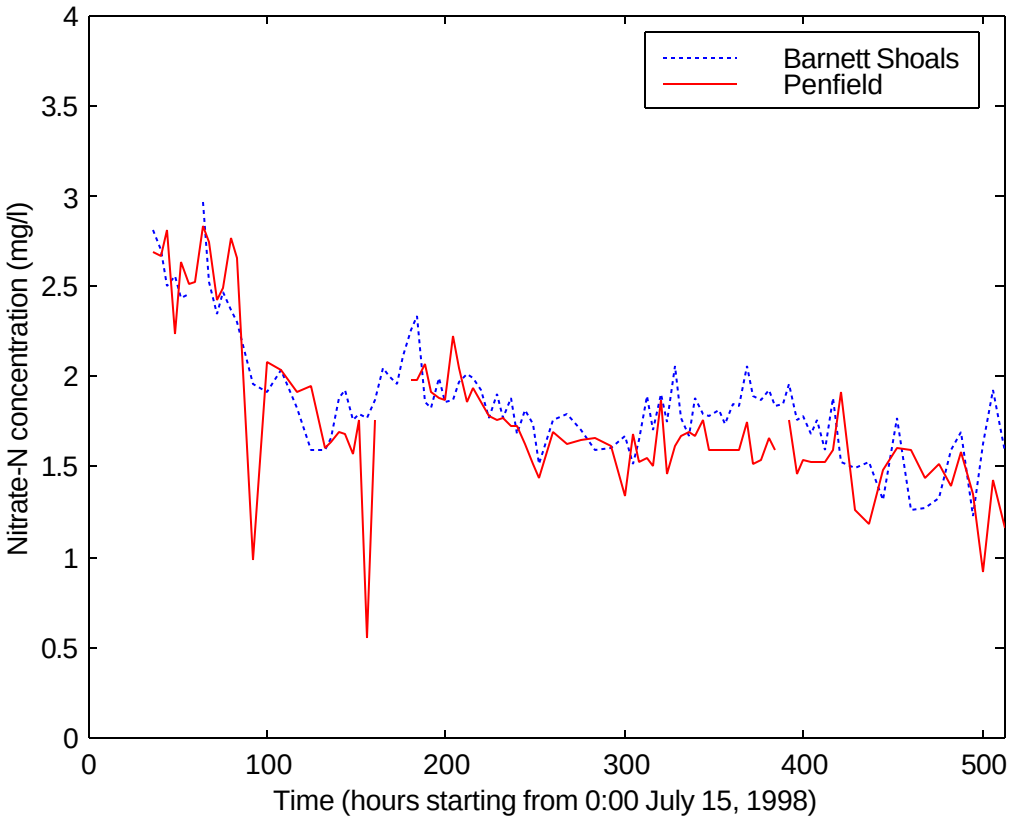
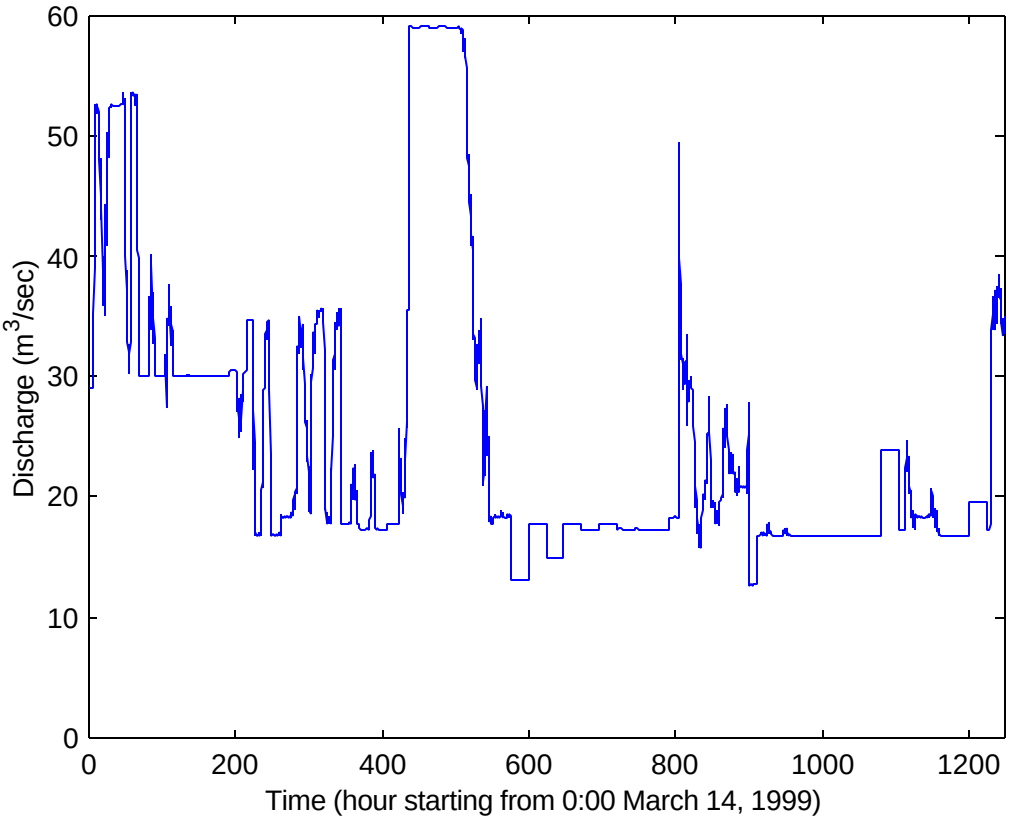


Figure 4-15. Nitrite-N concentration on the Oconee River, 1998



↑ July 15 ↑ August 5

Figure 4-16. Nitrate-N concentration on the Oconee River, 1998



↑ March 14

↑ May 5

Figure 4-17. Discharge time series at Barnett Shoals dam on the Oconee River, 1999

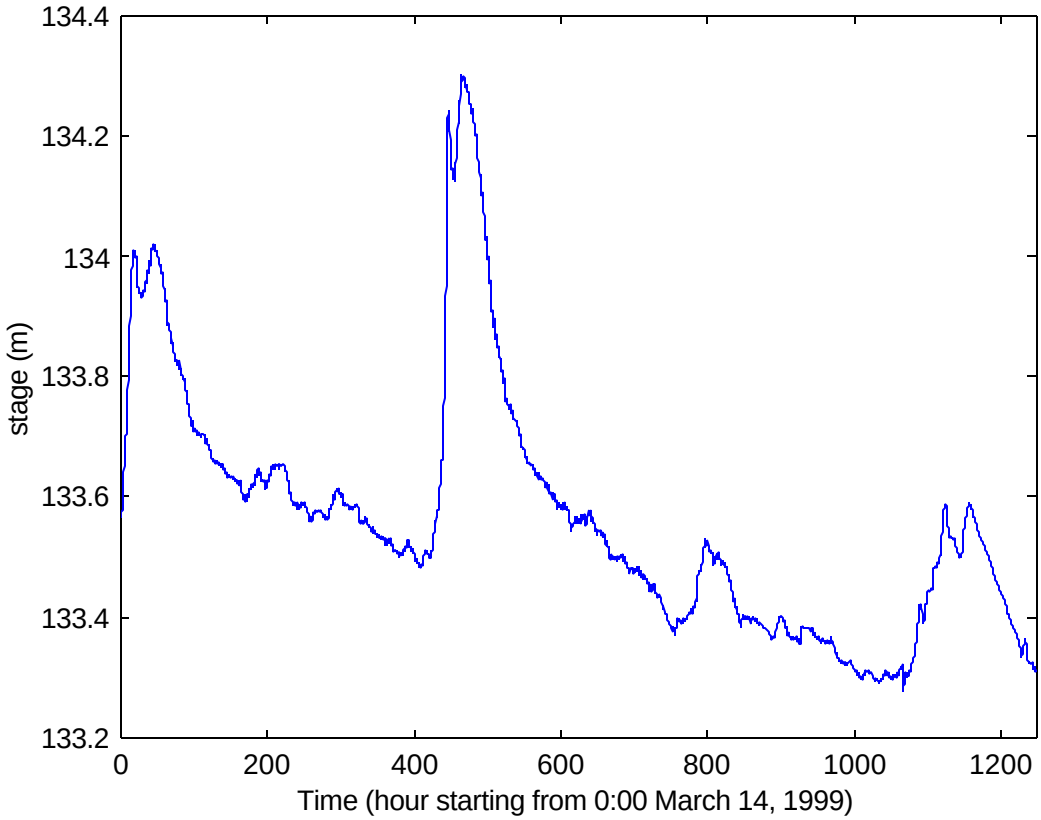
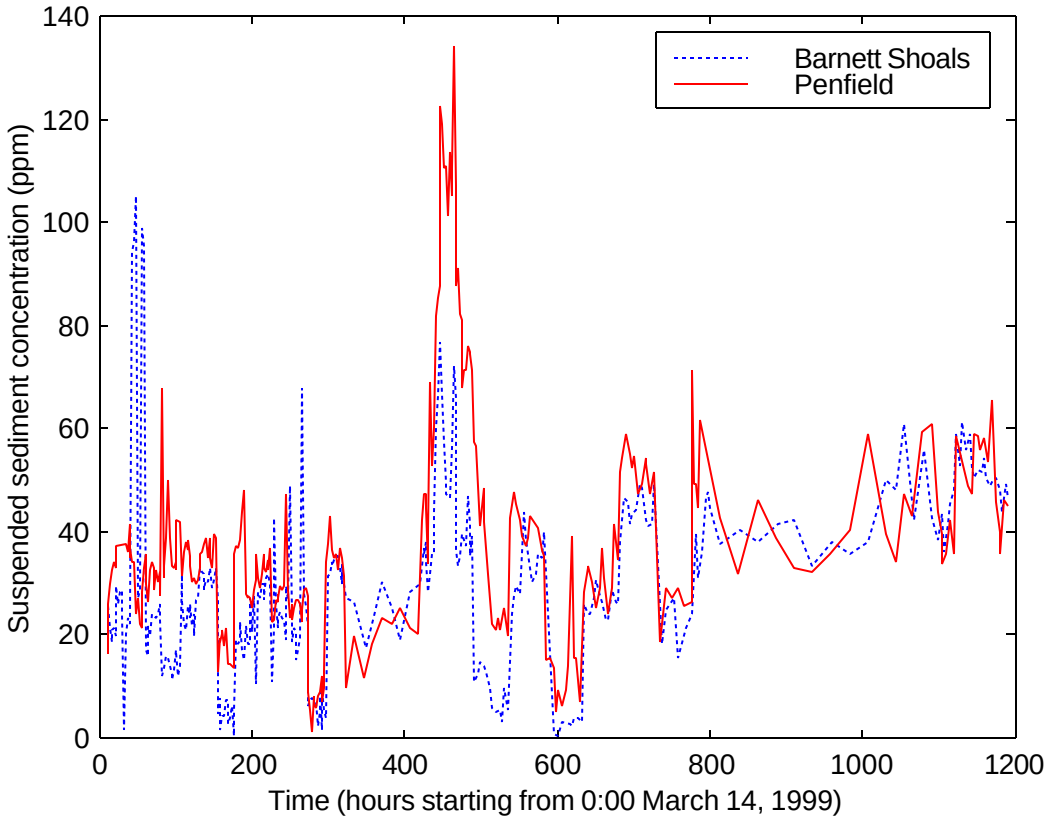
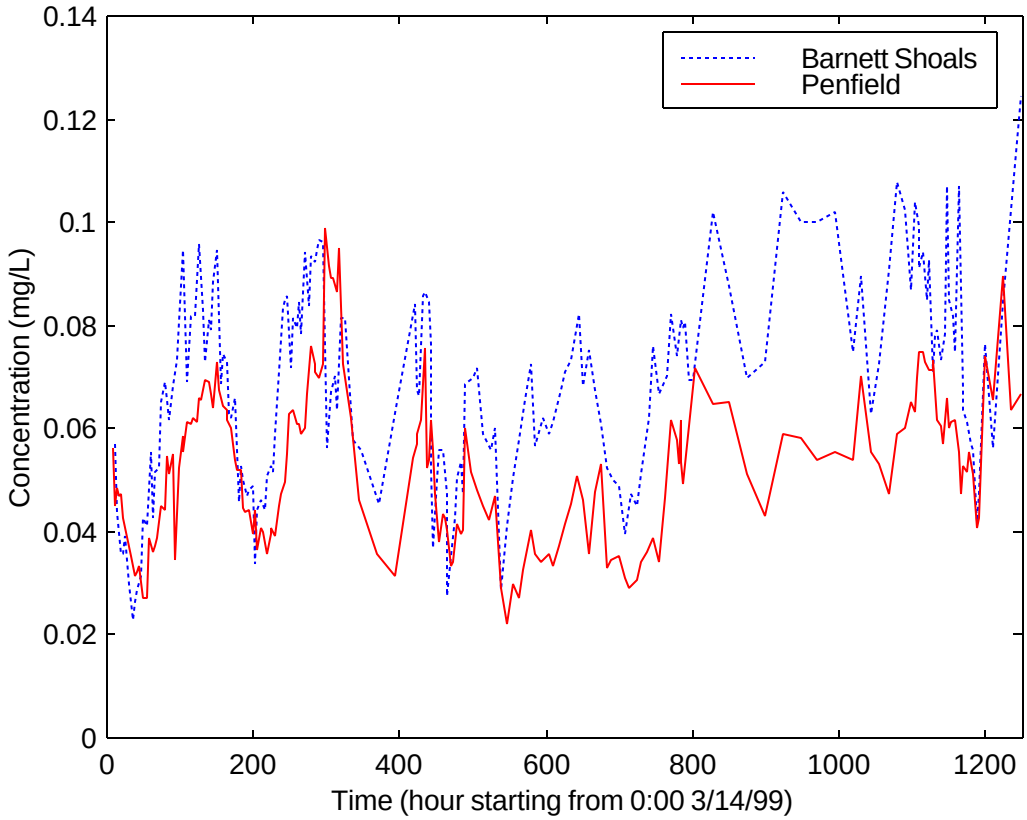


Figure 4-18. Elevation time series at Penfield on the Oconee River, 1999



↑ March 14 ↑ May 5

Figure 4-19. Suspended sediment concentration on the Oconee River, 1999



↑ March 14 ↑ May 5

Figure 4-20. Ortho-phosphate-P concentration time series on the Oconee River, 1999

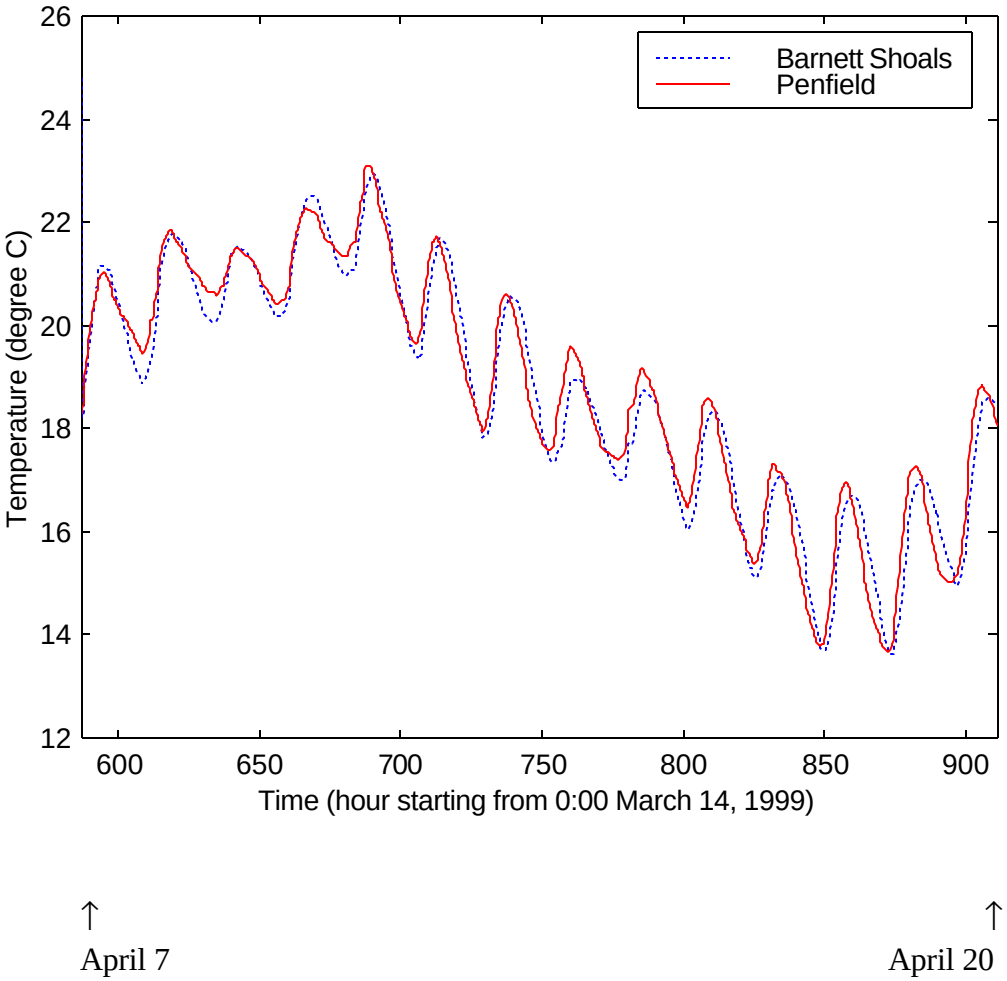


Figure 4-21. Water temperature time series on the Oconee River, 1999

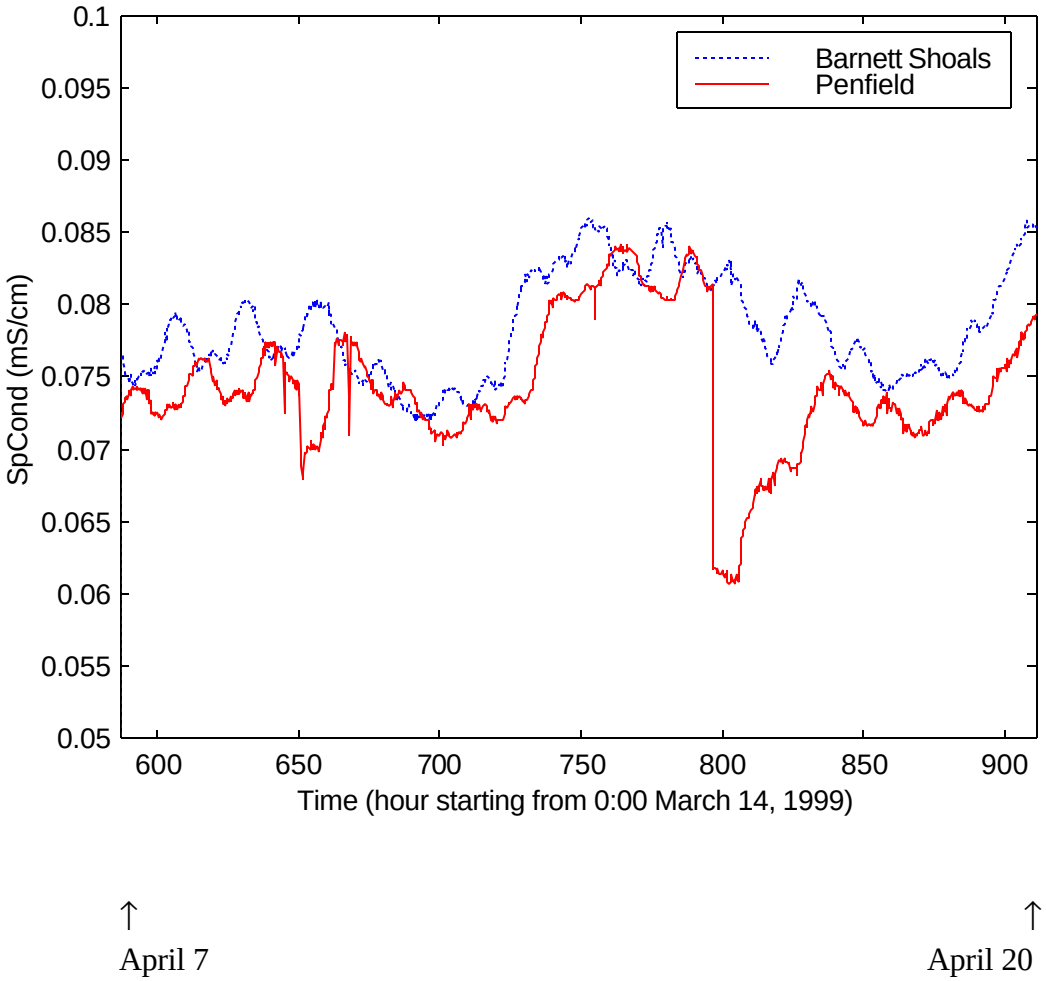


Figure 4-22. Specific conductivity time series on the Oconee River, 1999

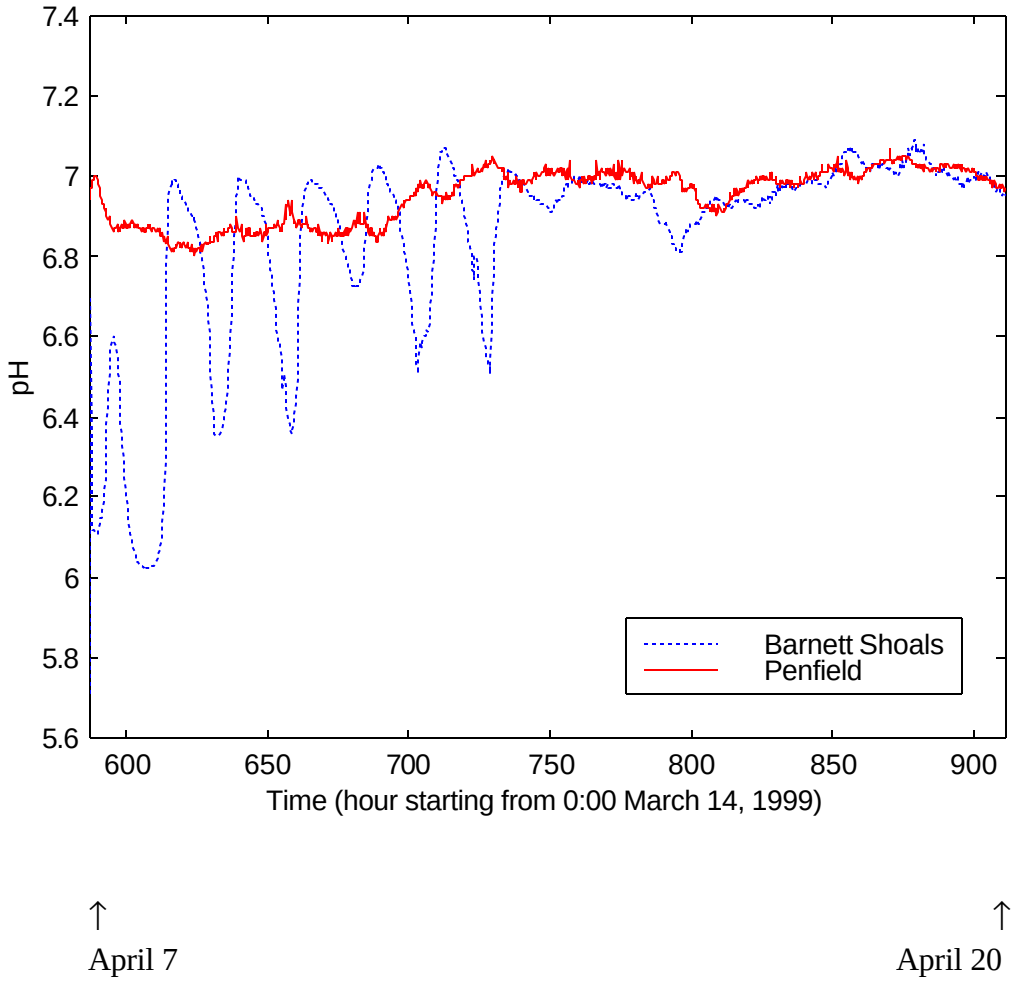


Figure 4-23. pH values on the Oconee River, 1999

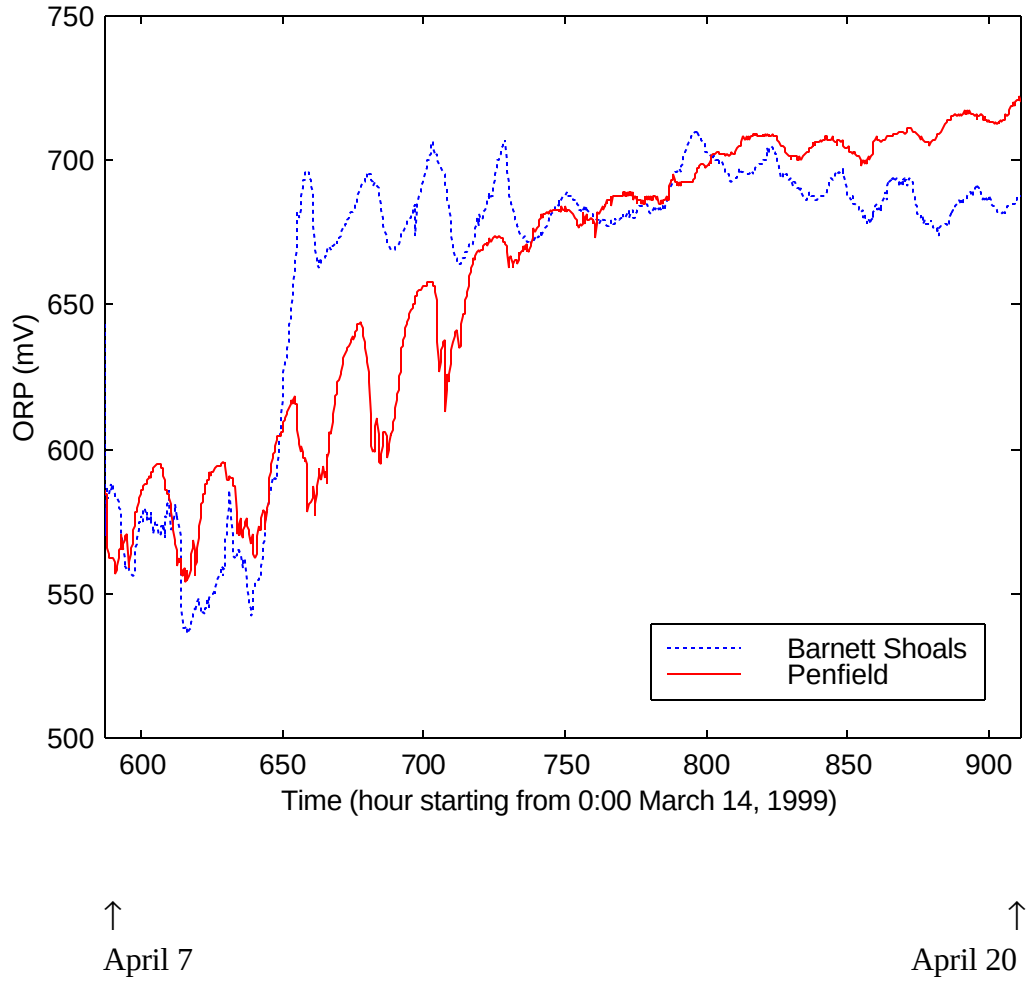


Figure 4-24. Oxidation/reduction potential time series on the Oconee River, 1999

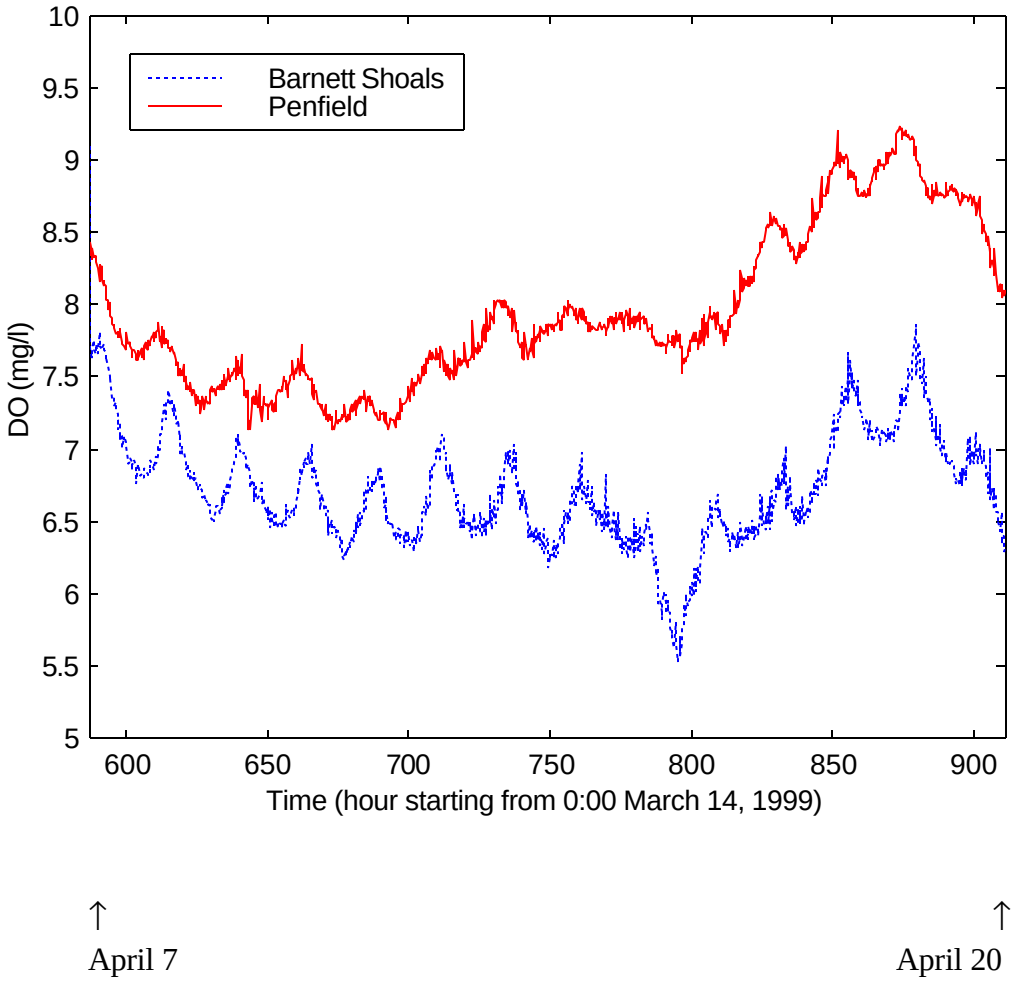


Figure 4-25. Dissolved oxygen time series on the Oconee River, 1999

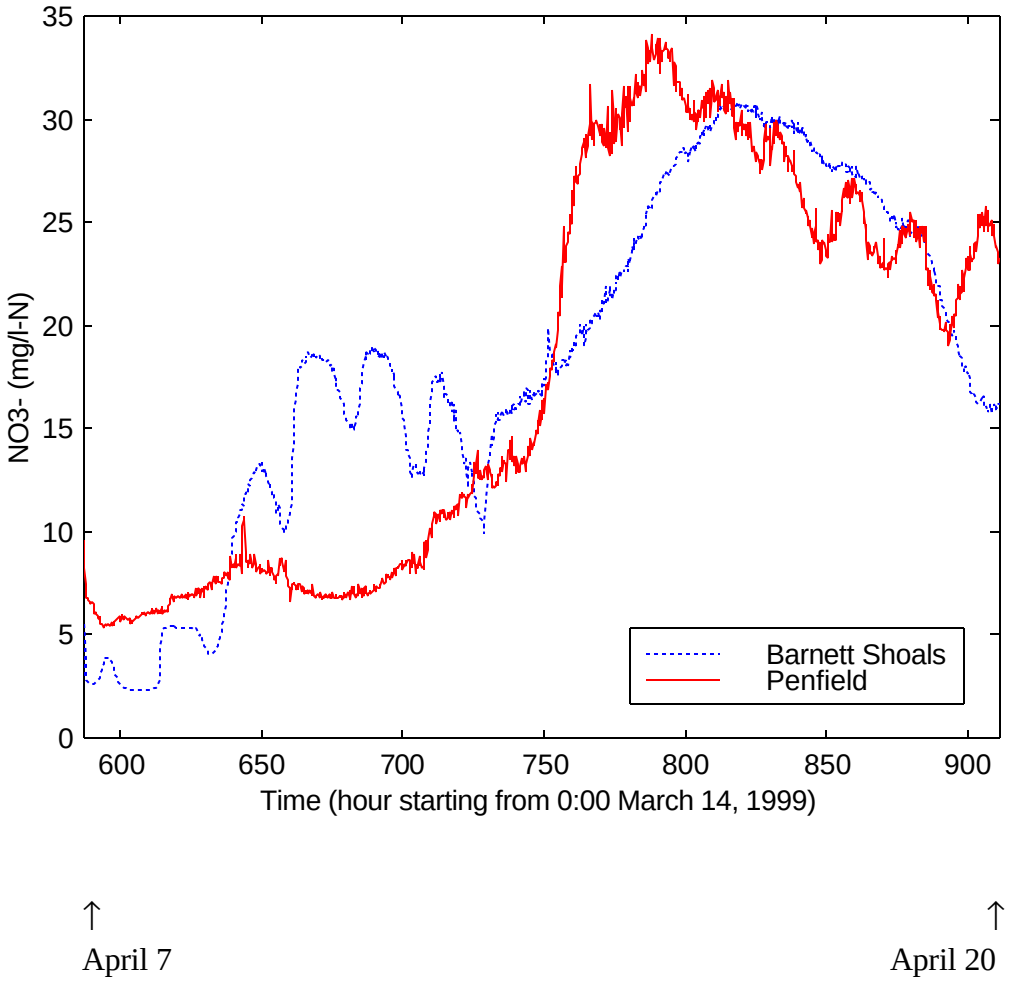


Figure 4-26. Nitrate-N concentration time series on the Oconee River, 1999

CHAPTER 5

MODEL CALIBRATION AND EVALUATION

Calibration and evaluation of the model were conducted separately for hydraulics, sediment transport, and each of the dissolved contents. In Section 5.1. of the chapter, the results of calibration and evaluation of hydraulics based on the Weihe River data in 1973 and 1974 are discussed. In Section 5.2., calibration and evaluation of sediment transport is discussed based on data of the Weihe River in 1973 and 1974, and the Oconee River in 1998 and 1999. Since STAND differs from previous models in respect of the means of computing the non-equilibrium sediment transport, as discussed in Chapter 3, Section 5.3. will present a comparison of the sediment transport results for the present and conventional models. Aspects of calibrating the water quality components of STAND are then reported in Section 5.4. Evaluations were not conducted for some of the dissolved contents, due to the absence of an independent second data set. The files containing data sets used for input to the model, and those containing simulation results are too large to be included in this dissertation. However, they will be included in the archive of this research project.

5.1. Calibration and Evaluation of Hydraulics

In the calibration and evaluation of hydraulics, measured quantities and simulation results at the Weinan Gauging Station (in the middle of the studied reach) are compared. We change the parameter values in seeking the best possible match of simulated and observed quantities.

The hydraulic variables and model parameters are listed in table 5-1. The major parameters to be calibrated are Manning's roughness coefficient, and the numerical coefficient *q*, which represents the location of point M in the schematic representation of the solution mesh.

Table 5-1 Variables and parameters in the hydraulic component

hydraulic variables	
Q = discharge	$[L^3T^{-1}]$
v = flow velocity	$[LT^{-1}]$
Z = surface elevation	[L]
A = wetted cross-section area	$[L^2]$
B = surface width	[L]
P = wetted cross-section perimeter	[L]
R = hydraulic radius of a cross-section	[L]
D = hydraulic depth	[L]
parameters	
n = Manning roughness coefficient	$[L^{-1/3}T]$
q = numerical coefficient	[-]

The value of q ranges from 0.5 to 1.0, and mainly affects the stability of the numerical scheme. We took q to be 0.65 in our entire model evaluation process. The value of Manning's coefficient ranges from 0.010 to 0.050 for most possible documented materials as channel bed or bank. The Chinese Hydrologic Yearbook occasionally provides the value of Manning's coefficient together with other data, which provides guidance for calibrating the model.

Calibration of Manning's coefficient was a trial-and-error process, selecting from the values of 0.040, 0.030, 0.025, 0.020, 0.015, and 0.010. Given the fact that the river is of large scale, with width ranging between about 100 m during low flow to hundreds of meters, or even kilometers in events of floodplain overflow, the smaller values should be given preference.

The simulated discharges and stages at the Weinan (middle) station with Manning's coefficient being 0.040 and 0.010 are shown in Figures 5-1 and 5-2, together with the observed values. It is apparent that stages are very much overestimated when n = 0.040 for all significant hydrologic events, and sometimes even for base flows. On the other hand, stages are underestimated for both hydrologic events and base flows when n = 0.010. Not much difference can be seen from the discharge plot, although a lower value of Manning's coefficient does provide higher values of discharge at the several extreme hydrologic events than the higher value does.

Figures 5-3 and 5-4 show the simulated discharges and stages at Weinan with Manning's coefficient being 0.020 and 0.015, together with the observed values. For discharge simulations, the differences in the results given by the two values can hardly be seen visually. For the simulated stages, the coefficient with a value of 0.020 gives better simulation results for the several minor hydrologic events and for the period when time ≈ 250 – 280 days, while the coefficient with a value of 0.015 gives better results for the several extreme hydrologic events.

It is tempting to make Manning's coefficient a variable rather than a constant, because the overestimates happen mainly during high flow. Setting a higher value of n during base flow and a lower value for peak flow may make the simulation fit the data for all periods. This approach is reasonable, since the Hydrologic Yearbook has recorded extremely small values of n during extreme high flow events.

A variable Manning's coefficient was therefore used in the calibration. Figure 5-5 shows the values of Manning's coefficient as a function of the hydraulic depth. The

value was set to be 0.020 during base flow (depth less than 3.0 meters), 0.015 at extreme high flow (depth greater than 4.5 meters), and changes linearly with depth during mid-magnitude flows. The simulation results are shown in Figure 5-6 and Figure 5-7. It is obvious that a variable Manning's coefficient provided more satisfactory simulation results in both discharge and surface elevation during peak flows, base flows and in between.

There is another problem to consider. If we set the value of *n* too low, numerically supercritical flow during low flow may occur. Supercritical flow is very unlikely for a river in its lower reaches. The numerically supercritical flow will cause a great amount of sediment transport numerically during low flow conditions. The high sediment transport during low flow is also very unlikely under a natural condition.

Moving on to the subject of evaluation of hydraulics, a 1974 data set of the same reach was used in the process. If a fixed Manning's coefficient has to be used, however, we would take $n = 0.020$ as the value of Manning's coefficient for the Weihe River and Figures 5-8 and 5-9 show the discharge and surface elevation at Weinan Gauging Station during a 140-day period. It is apparent that the simulation follows the observed values reasonably well, except that the discharges at the flood peaks are underestimated slightly and the surface elevations at base flows are overestimated a little. This further justifies the notion of using a variable Manning's coefficient, which can be set to a small value at peak flows thus elevating the peak discharges.

When a variable Manning's coefficient is used therefore in the evaluation process, results for discharges and elevations are as shown in Figures 5-10 and 5-11. The peak discharges are simulated better than when using the fixed Manning's coefficient of 0.020, while the surface elevations are similar to those using this fixed value. Making the roughness coefficient a function of hydraulic depth introduces an additional degree of freedom. Thus, more than a single formulation may provide similar simulation results.

However, it is unlikely that a unique formulation would obtain, unless we have enough field data to support it, which is also unlikely.

5.2. Calibration And Evaluation Of Sediment Transport

The variables and parameters in the sediment transport component are listed in Table 5-2. The parameters to be calibrated are the suspended sediment dispersion coefficient, sediment deposition rate constant, and sediment entrainment rate constant. Two sets of parameters were to be obtained for the two case study reaches. A Controlled Random Search (CRS) algorithm was used in finding an appropriate value for the parameters. A complete description of the Controlled Random Search algorithm can be found in Price (1979). Initial runs of the algorithm revealed that the value of the suspended sediment dispersion coefficient merely influences the smoothness of the simulation results with almost no impact on the average simulated suspended sediment concentration. Thus, to expedite the search for the parameter values, several values for the dispersion coefficient were chosen prior to the CRS runs. For each value of the dispersion coefficient, a CRS run has been implemented and the value of the objective function recorded. The parameters in the run with a minimum objective function value are chosen to be the “best” parameter set.

The 1973 Weihe data and the 1998 Oconee data have been used in the calibration process. Results for both the Yellow River and the Oconee River are listed in Table 5-3. However, the values for the sediment entrainment rate constant for the Weihe River are not listed here, because the suspended sediment concentrations were so high that there could have been no process of entrainment during the simulated period, and the CRS runs did not provide values of the entrainment rate constant that converge to a limited range.

Table 5-2 Variables and parameters in the sediment transport component

sediment transport variables	
C_s = suspended sediment concentration	$[ML^{-3}]$
sediment transport parameters	
k_{sed} = rate of sediment concentration approaching transport potential	$[T^{-1}]$
E_s = suspended sediment transport potential	$[L^2T^{-1}]$
k_{sedDep} = sediment deposition rate constant	$[T^{-1}]$
k_{sedEnt} = sediment entrainment rate constant	$[T^{-1}]$

Table 5-3 Parameter values for the sediment transport component

Parameters	Weihe '73	Weihe '74	Oconee '98	Oconee '99	unit
E_s	50.0	50.0	50.0	50.0	m^2/s
k_{sedDep}	1.3×10^{-9}	1.3×10^{-9}	4.81×10^{-9}	4.81×10^{-9}	1/s
k_{sedEnt}	-	-	2.8×10^{-3}	2.8×10^{-3}	1/s

The simulation results using these parameter values are shown in Figure 5-12 and 5-13. It can be seen that the suspended sediment concentrations at Weinan Gauging Station on the Yellow River are simulated fairly well, except two of the major peaks are slightly underestimated. There appear to be some minor fluctuations during the period when observed sediment concentrations were not available. This was possibly caused by the numerically supercritical flows. A simple way to eliminate the effect of numerically

supercritical flows is to deny any sediment transport under a certain low flow. However, an appropriate threshold is not easy to determine.

The Oconee River simulation does not follow the trend of the observed data at the beginning of the record, but improves thereafter. This is probably caused by the laboratory personnel's lack of experience at the beginning of the sediment concentration analysis. It appears that the major peak of the event is underestimated, but this might be caused by the fact that the sampling frequency during the rain event was not high enough to catch the high concentration that lasted for only a short period (according to the simulation).

The 1974 Weihe data and the 1999 Oconee data were used for the purposes of model evaluation, with results shown in Figures 5-14 and 5-15. For the Weihe River, the simulation followed the observed trend fairly well, except that the peak values were slightly underestimated. Considering that the channel might experience significant change after the previous year's high volume of sediment transport and associated characteristic changes of the river, it is reasonable to say that the simulation did well.

For the Oconee River, the simulation followed the trend expressed by the data reasonably well most of the time, although the major peak was underestimated. There was a peak at the beginning of the simulation, arising from a couple of input (upstream suspended sediment concentration) data points that are significantly higher than the other points over this period. These data are believed to be suspect, although we have no reason to exclude them from this analysis. The simulated suspended sediment concentration response slightly underestimated the major high-sediment-concentration event. However, since the model succeeded in replicating observed behavior for the rest of the time in the 1999 period, we have reasons to believe that the model structure is correct and that the underestimation might come from the lack of calibration of hydraulics for the Oconee River.

The influence of strong upstream driving mechanisms on the computation results is of concern, since the data presented in Chapter 4 clearly indicate the similarity between upstream and downstream behaviors. To evaluate the effects of mechanisms other than "pure transport", computations of sediment behavior are conducted with the source/sink mechanism eliminated. Comparisons between the computations with and without the source/sink mechanisms are then made. The 1973 Weihe data and the 1999 Oconee data are used in these computations. The simulations with and without the source/sink mechanisms, along with the observed quantities, are shown in Figure 5-16. Figure 5-17 and Figure 5-18 show the absolute and relative differences between the computed suspended sediment concentrations at Weinan with and without the source/sink mechanism. For the Weihe River, the absolute magnitude of the maximum difference is quite significant (about 2×10^4 ppm), although this is not so compared to the absolute magnitude of the events (one degree of magnitude higher). For the Oconee River, the comparison between the simulations with and without the source/sink mechanisms together with the observations are shown in Figure 5-19. Figures 5-20 and 5-21 show the absolute and relative difference between the computations with and without the source/sink mechanisms. It can be clearly seen that the difference is driven by the hydraulics. It is clear that the mechanism plays an important role in the computation and that a simple "pure transport" mechanism is not enough in describing the model systems.

5.3. Comparison Of Present And Conventional Approach To Computing Non-Equilibrium Sediment Transport

Simulations of suspended sediment transport were done using both the conventional methods and the approach in our model. With the formulation given by equations (3-20) and (3-21), in which the suspended sediment concentrations are considered affected only by the relevant quantities (transport potential, transport rate of

suspended sediment) of upstream locations, satisfactory simulation results could not be obtained by using parameter values suggested by Zhang et al. (1983), Chang (1988), Han et al. (1990), and Yang et al. (1998). This is not surprising, because the derivation to obtain equations (3-20) and (3-21) starts with a convection-diffusion equation without a time derivative term (Chang 1988), which suggests that there is an underlying assumption of steady state suspended sediment behavior. Since the studied system is not in equilibrium condition, this assumption may not be appropriate. By using parameter values that are orders of magnitude away from the suggested values (at the magnitude of 10^{-9}), the observations can be followed by the simulation, but with significant phase lags. The extremely small parameter value almost makes the exponential function unity and thus eliminates the term of suspended sediment transport potential. As a result, the downstream suspended sediment concentration is only a function of upstream suspended sediment concentration and they are almost simultaneous to each other, which is responsible for the apparent phase lags. The proposed approach is described in Section 3.3. Numerical solution of equation (3-25) (a complete equation with time derivative and source/sink mechanisms) is sought. No steady state approximation is made. The comparisons of the simulation results are shown in Figures 5-22 and 5-23. It can be seen that the method our model adopted gave more convincing simulation results than the conventional methods do.

5.4. Calibration Of Water Quality Components

5.4.1. Ortho-Phosphate

The parameters to be calibrated are the ortho-phosphate dispersion coefficient E_p, phosphate rate constant k_P, phosphate concentration in the interstitial water C_{Pint}, and two parameters in the empirical Freundlich equation (3-42) k_f and p_f. The variables and parameters in the phosphate component are listed in Table 5-4.

Table 5-4 Variables and parameters in the ortho-phosphate component

variables in the ortho-phosphate component	
C_P = ortho-phosphate concentration	$[ML^{-3}]$
parameters in the ortho-phosphate component	
E_P = ortho-phosphate dispersion coefficient	$[L^2T^{-1}]$
k_P = ortho-phosphate adsorption coefficient	$[T^{-1}]$
C_{Pint} = phosphate concentration in the interstitial water	$[ML^{-3}]$
k_f = empirical parameter in Freundlich equation	[-]
p_f = empirical parameter in Freundlich equation	[-]

The 1999 data on the Oconee River were used in the calibration of the ortho-phosphate component, using once again the CRS algorithm. We first choose several values for the dispersion coefficient, and for each value then carry out a CRS run in order to estimate values of the other parameters. The value of the objective function was recorded for each run. The parameter set with the smallest objective function value is considered the best. The parameter values obtained are listed in Table 5-5.

Table 5-5 Parameter values of the ortho-phosphate component

parameter	value	unit
E_P	500.0	m^2/s
k_P	8.0×10^{-6}	1/s
C_{Pint}	0.005	g/m^3
k_f	2.5×10^{-3}	-
p_f	0.05	-

The simulation results are shown in Figure 5-24. It can be seen that the simulation slightly underestimated the most significant high-phosphate event and overestimated for a short period around t = 1000 hours. Other than these discrepancies, the observations were followed reasonably well.

In order to evaluate the influence of the adsorption mechanism and the whole source/sink mechanism, computations without them have been conducted. The results are shown in Figures 5-25 to 5-27. It is obvious, that although the downstream behavior is somewhat controlled by the input to the system at the upstream end of the reach, a "pure transport" mechanism can not fully describe the system. The adsorption mechanism (as well as the whole source/sink mechanism) plays an important role in the system and can not be overlooked. It is apparent that the differences are getting larger as the simulations go on; this clearly corresponds to the increasing level of suspended sediment concentrations and clearly demonstrates the influence of sediments on ortho-phosphate dynamics in the water column. The mechanism of interstitial water release has some minor influences on the final results compared to that of the adsorption processes; this is probably caused by the small value of ortho-phosphate concentration in the interstitial water.

5.4.2 Nitrate

The parameters to be calibrated in the nitrate component are nitrification rate constant k_{Nitr}, which sets the rate for the first-order nitrification mechanism, and denitrification rate constant k_{DeNi}, which describes the rate for the first-order denitrification process. The state variable is nitrate concentration C_{nitr}. The variable and the parameters are listed in Table 5-6.

Besides the mechanism in the nitrate component, the mechanism in the ammonium component also contributes to the changes of the nitrate concentration. The ammonium component includes state variable C_{ammo}, ammonium concentration,

parameters M_O, Monod half saturation concentration for dissolved oxygen, k_{Nitr}, nitrification rate constant, k_{AmAd}, ammonium adsorption coefficient, and k_{amf} and p_{amf}, the two parameters in the empirical Freundlich equation (3-42). The variables and parameters in the ammonium component are listed in Table 5-7.

Table 5-6 State variable and parameters in the nitrate component

state variable in nitrate component	
C_{nitr} = nitrate concentration	$[ML^{-3}]$
parameters in nitrate component	
E_{nitr} = nitrate dispersion coefficient	$[L^2T^{-1}]$
k_{Nitr} = nitrification rate constant	$[T^{-1}]$
k_{DeNi} = denitrification rate constant	$[T^{-1}]$

Table 5-7 State variable and parameters in the ammonium component

state variables in the ammonium component	
C_{ammo} = ammonium concentration	$[ML^{-3}]$
parameters in the ammonium component	
E_{ammo} = ammonium dispersion coefficient	$[L^2T^{-1}]$
k_{AmAd} = ammonium adsorption coefficient	$[T^{-1}]$
M_O = Monod half saturation rate for dissolve oxygen	$[ML^{-3}]$
k_{amf} = parameter in the Freundlich equation	[-]
p_{amf} = parameter in the Freundlich equation	[-]

The 1999 Oconee data on nitrate concentrations were used in the calibration. The parameter values for both the nitrate and the ammonium components were obtained by the CRS procedure. The values of the parameters are listed in Table 5-8, and simulation results are shown in Figure 5-28. It is apparent that the simulation did not succeed in replicating the observed behavior of nitrate. There was a high-nitrate-concentration event occurring at Penfield at time ≈ 750 – 800 hours (the downstream station) prior to the occurrence of the corresponding event at the upstream station. This can probably be explained by possible discharge of nitrate (carried by flows in the small tributaries) in between the stations where pasture lands are along the reach. It is extremely hard for a model to simulate such phenomenon without some inputs in between the stations. There is also reasonable doubt on the performance of the nitrate probes, which made the data suspect.

Table 5-8 Parameter values of the nitrate and ammonium components

parameter	*value*	*unit*
E_{nitr}	500.0	m^2/s
k_{Nitr}	1.6×10^{-5}	1/s
k_{DeNi}	2.0×10^{-4}	1/s
E_{ammo}	1000.0	m^2/s
k_{AmAd}	0.05	1/s
M_O	4.0	mg/L
k_{amf}	8.0×10^{-6}	-
p_{amf}	0.02	-

5.4.3 Dissolved Oxygen

The parameters in the dissolved oxygen component are E_O, dissolved oxygen concentration, k_{reae}, reaeration rate constant, a_3, rate of oxygen production per unit of algal photosynthesis, m, algal growth rate, a_4, rate of oxygen production per unit of algae respired, r, algal respiration rate, C_{algae}, algal concentration, r_{sod}, sediment oxygen demand rate, a_5, rate of oxygen uptake per unit of ammonia nitrogen oxidization, and b_1, ammonia oxidization rate coefficient. The state variable and parameters are listed in Table 5-9.

The reaeration rate is determined by flow velocity and hydraulic radius, according to equation (3-53). The parameters in the equation were given by Long (1984). The values of a_3 m a_4 r are given by the QUAL-II model (Brown and Barnwell 1987). Algal concentration was treated as a constant here and the value of it was obtained from calibration. The other parameter values were obtained from a trial-and-error calibration process. The parameter values were listed in Table 5-10.

Table 5-9 State variable and parameters in the dissolved oxygen component

state variable in the dissolved oxygen component	
O_2 = dissolved oxygen concentration	$[ML^{-3}]$
parameters in the dissolved oxygen component	
E_O = dispersion coefficient for dissolved oxygen	$[L^2T^{-1}]$
k_{reae} = reaeration rate constant	$[T^{-1}]$
a_3 = rate of oxygen production per unit of algal photosynthesis	[-]
m = algal growth rate	$[T^{-1}]$
a_4 = rate of oxygen production per unit of algae respired	[-]
r = algal respiration rate	$[T^{-1}]$

C_{algae} = algal concentration	[ML^{-3}]
r_{sod} = sediment oxygen demand rate	[$ML^{-2}T^{-1}$]
a_5 = rate of oxygen uptake per unit of ammonia nitrogen oxidization	[-]
b_1 = ammonia oxidization rate coefficient	[T^{-1}]

Table 5-10 Parameter values in the dissolved oxygen component

parameter	value	unit
E_O	10.0	m^2/s
a_3	1.8	-
m	2.31×10^{-5}	1/s
a_4	1.9	-
r	2.89×10^{-6}	1/s
C_{algae}	0.005	mg/l
r_{sod}	3.0×10^{-5}	g/m^2s
a_5	3.5	-
b_1	6.94×10^{-6}	1/s

The result of the simulation is shown in Figure 5-29. It is apparent that the simulation followed the observed diurnal oscillation fairly well. There is slight underestimation during the last one third of the simulation, but the simulated and observed values approached each other at the end of the simulation. There seems to be some slight phase lag in the simulation. This may come from the lack of calibration of the hydraulics for the Oconee River. However, it seems that the original data (Figure 4-

25) show some phase lag, and the phase lag of the simulation may very much be inherited from the original data.

Again, the impact of source/sink mechanism is computed and shown in Figures 5-30 through 5-32. The significance of the source/sink mechanism is obvious. The diurnal oscillation is probably caused by the absence of algal photosynthesis/respiration, and the reaeration process, which is controlled by the temperature dependent dissolved oxygen saturation concentration.

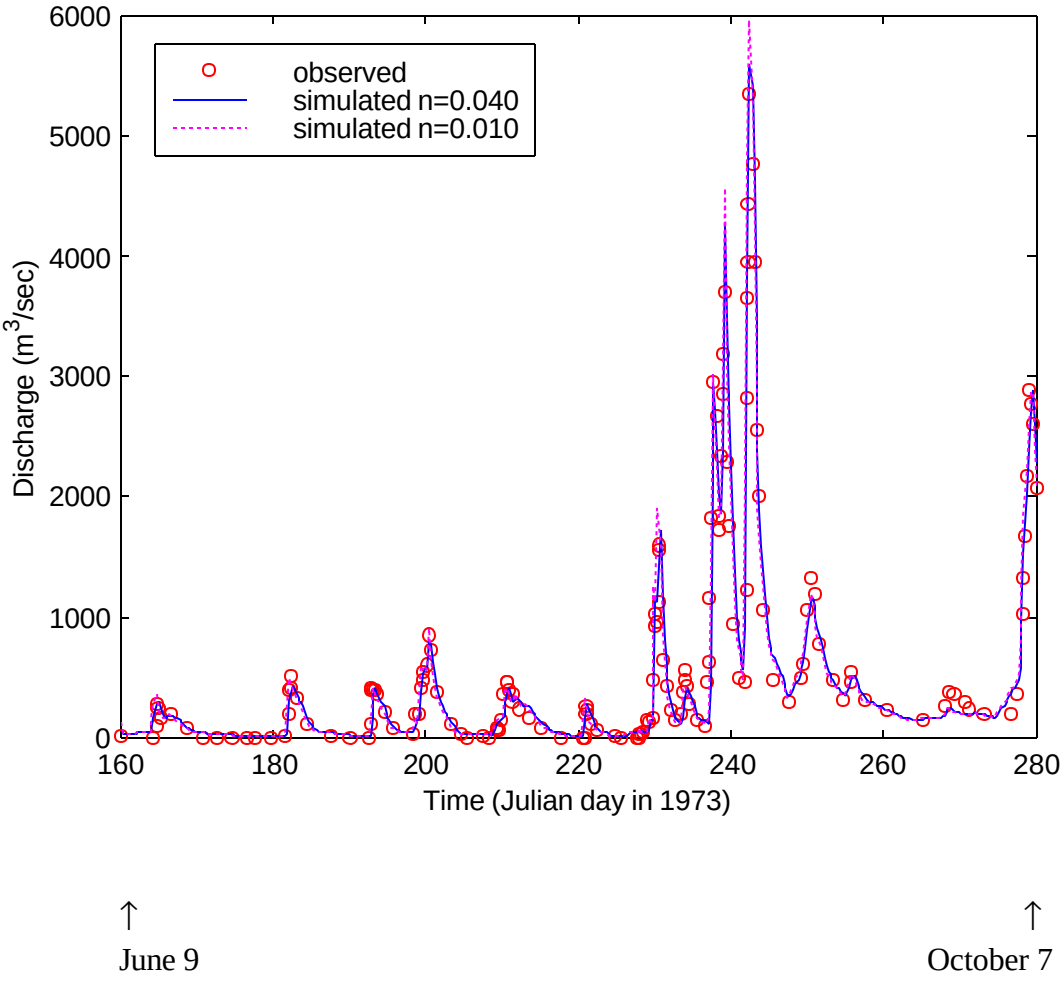


Figure 5-1. Discharge at Weinan Gauging Station, n = 0.040 and 0.010

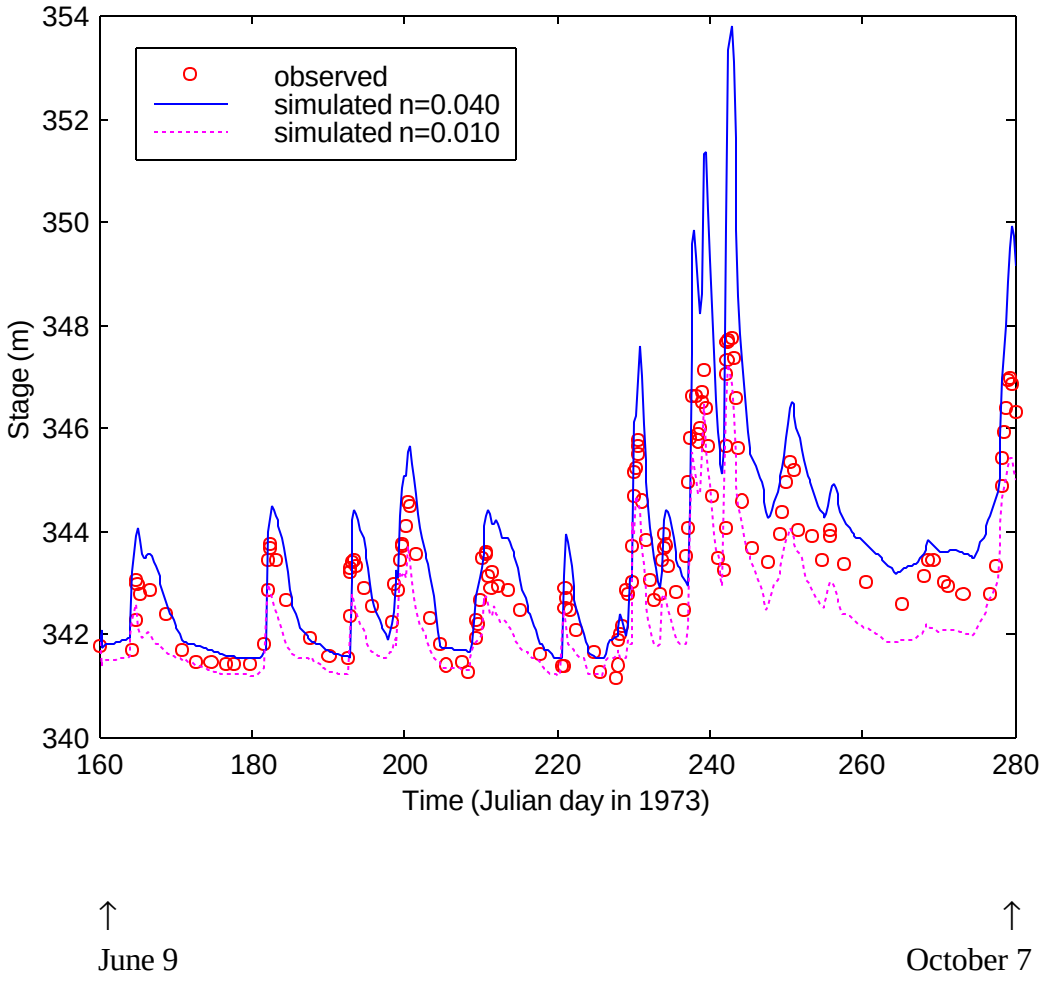


Figure 5-2. Surface elevation at Weinan Gauging Station, n = 0.040 and 0.010

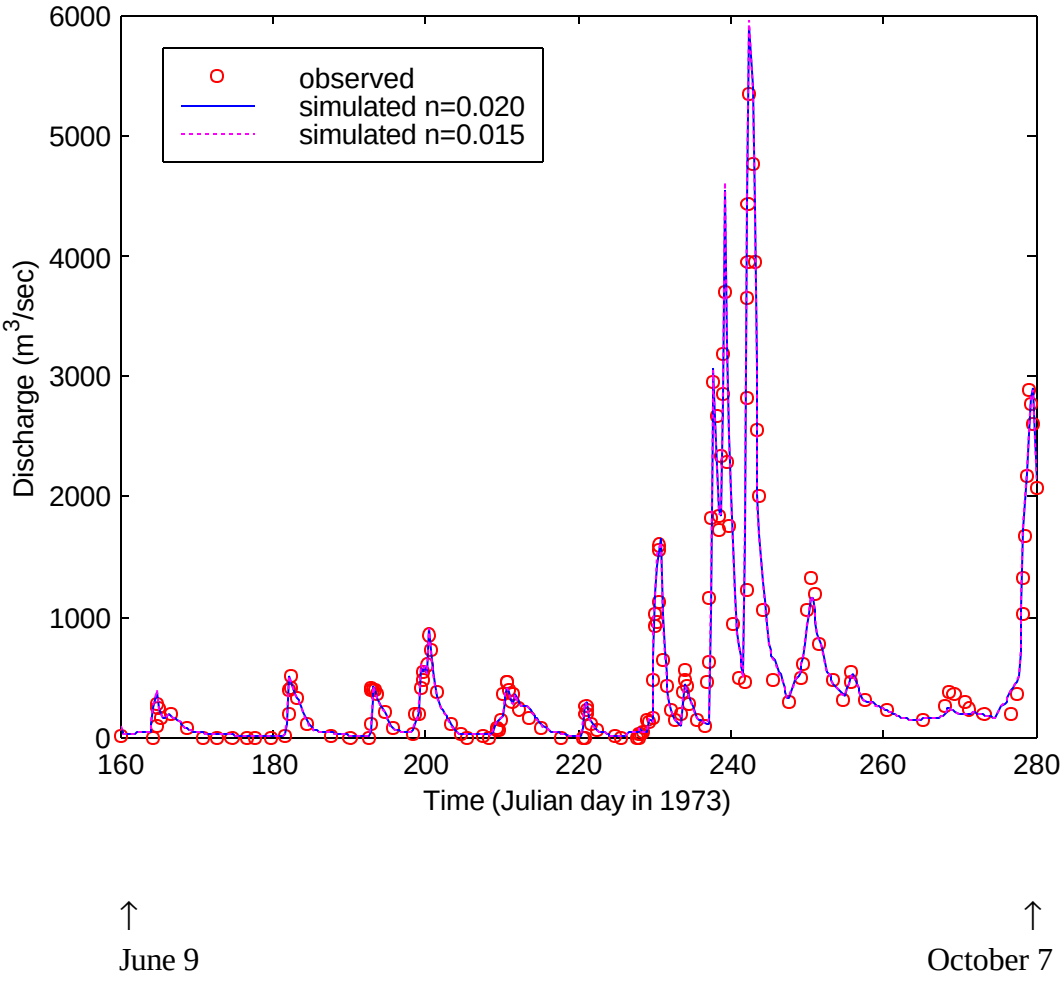


Figure 5-3. Discharge at Weinan Gauging Station, n = 0.020 and 0.015

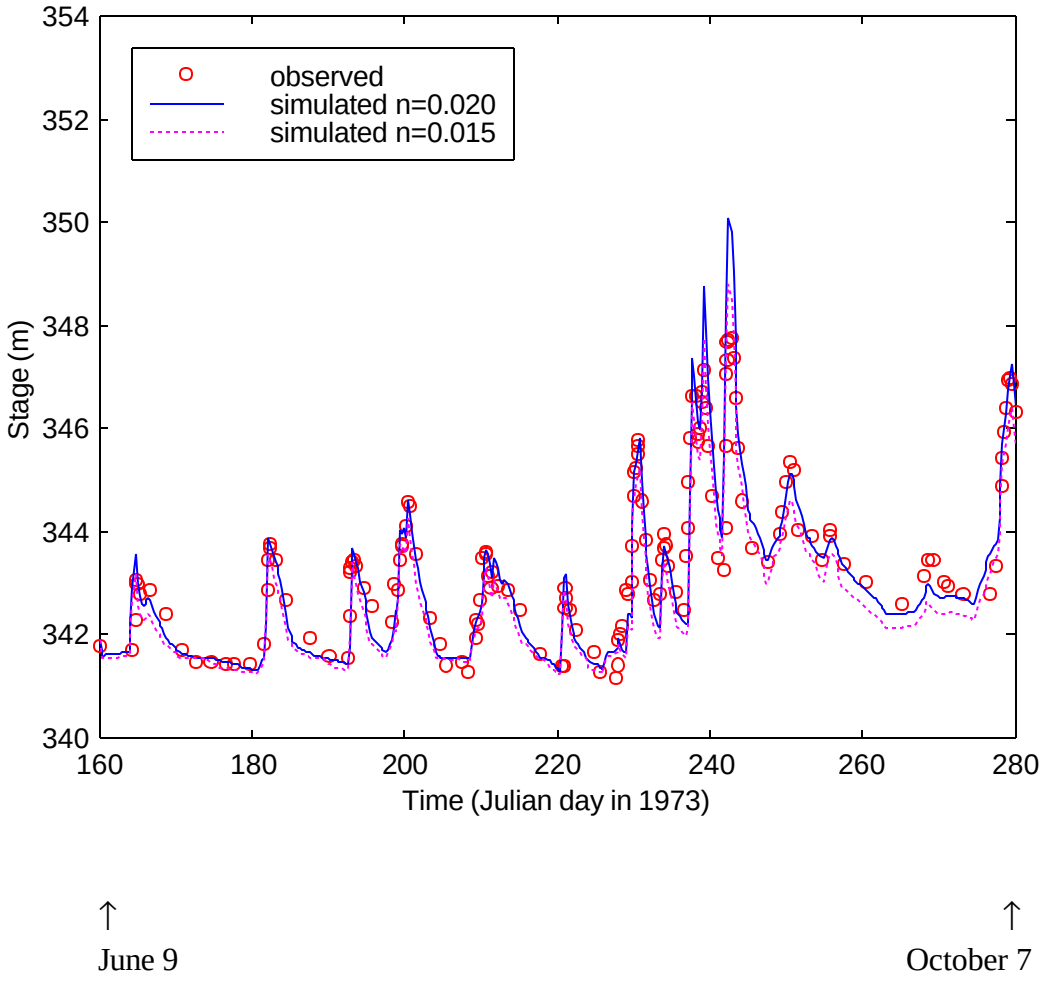


Figure 5-4. Surface elevation at Weinan Gauging Station, n = 0.020 and 0.015

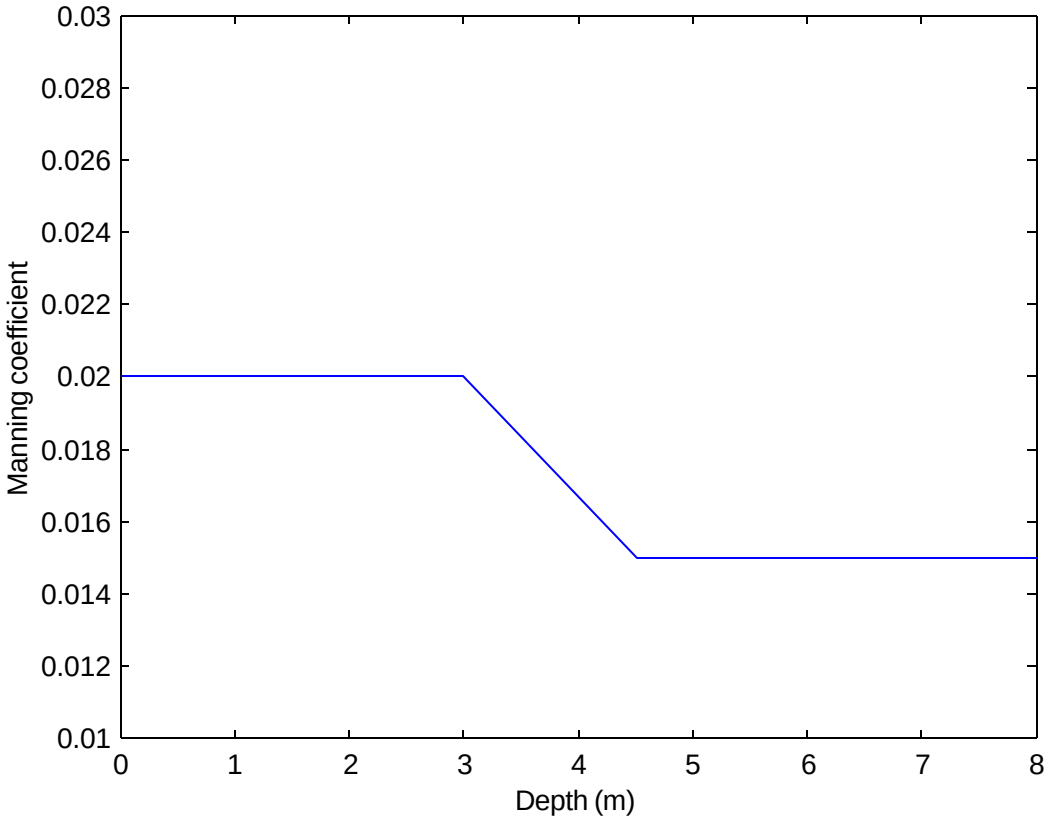


Figure 5-5. Manning's coefficient as a function of hydraulic depth

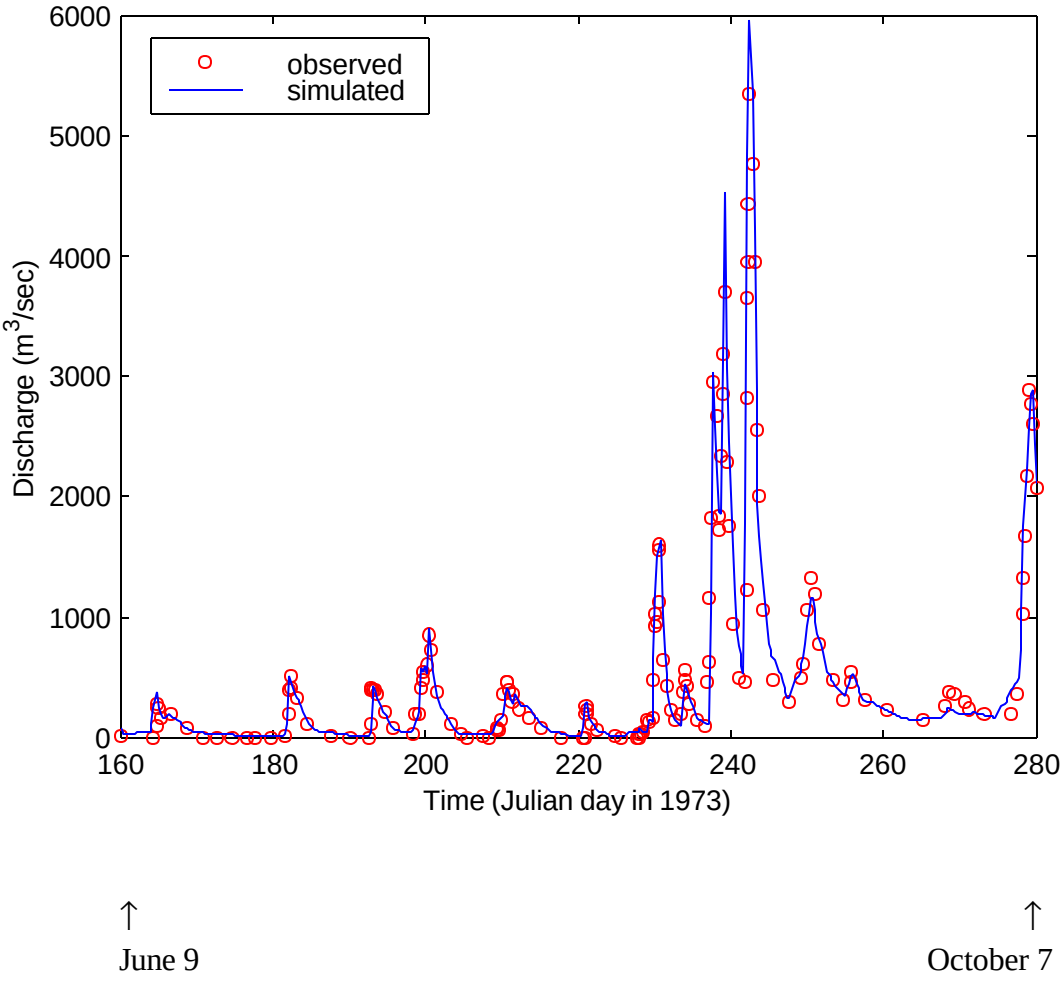


Figure 5-6. Discharge at Weinan Gauging Station, *n* is a function of hydraulic depth

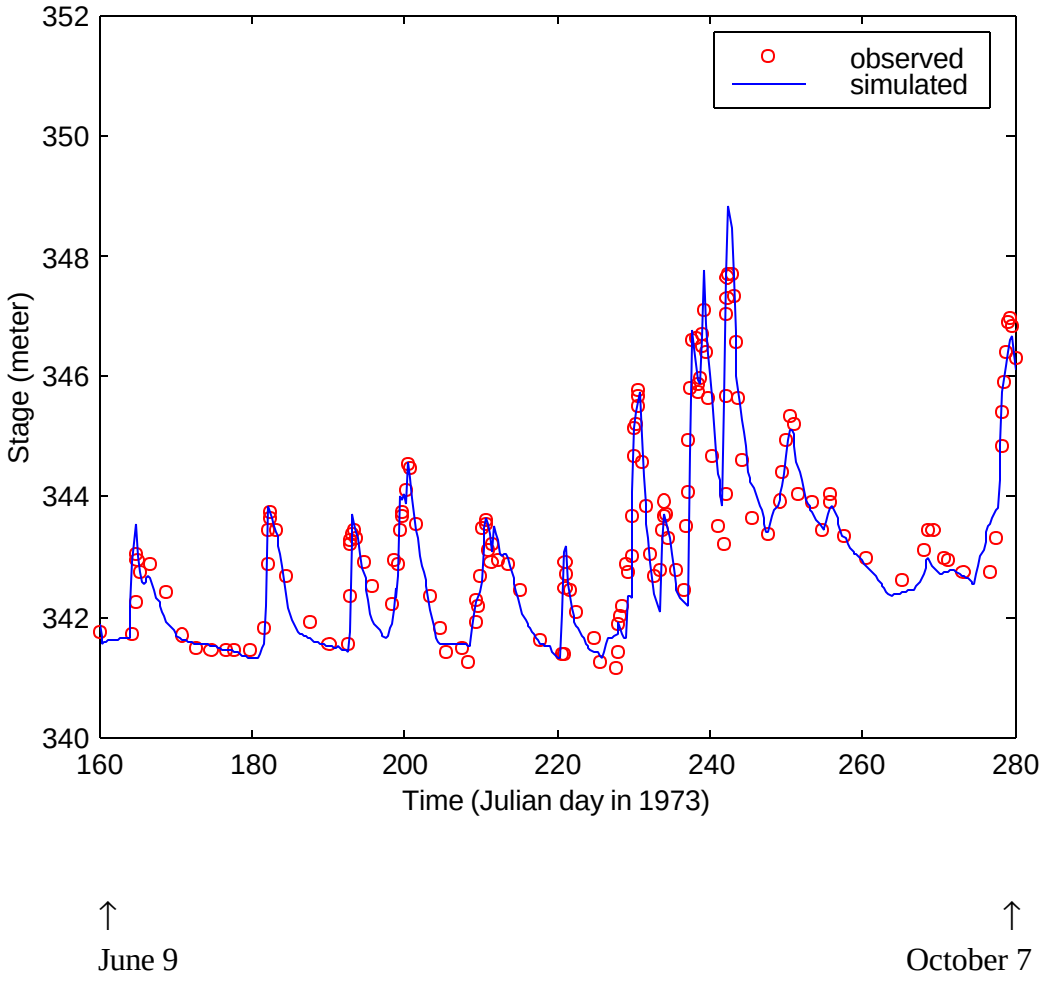


Figure 5-7. Surface elevation at Weinan Gauging Station, *n* is a function of hydraulic depth

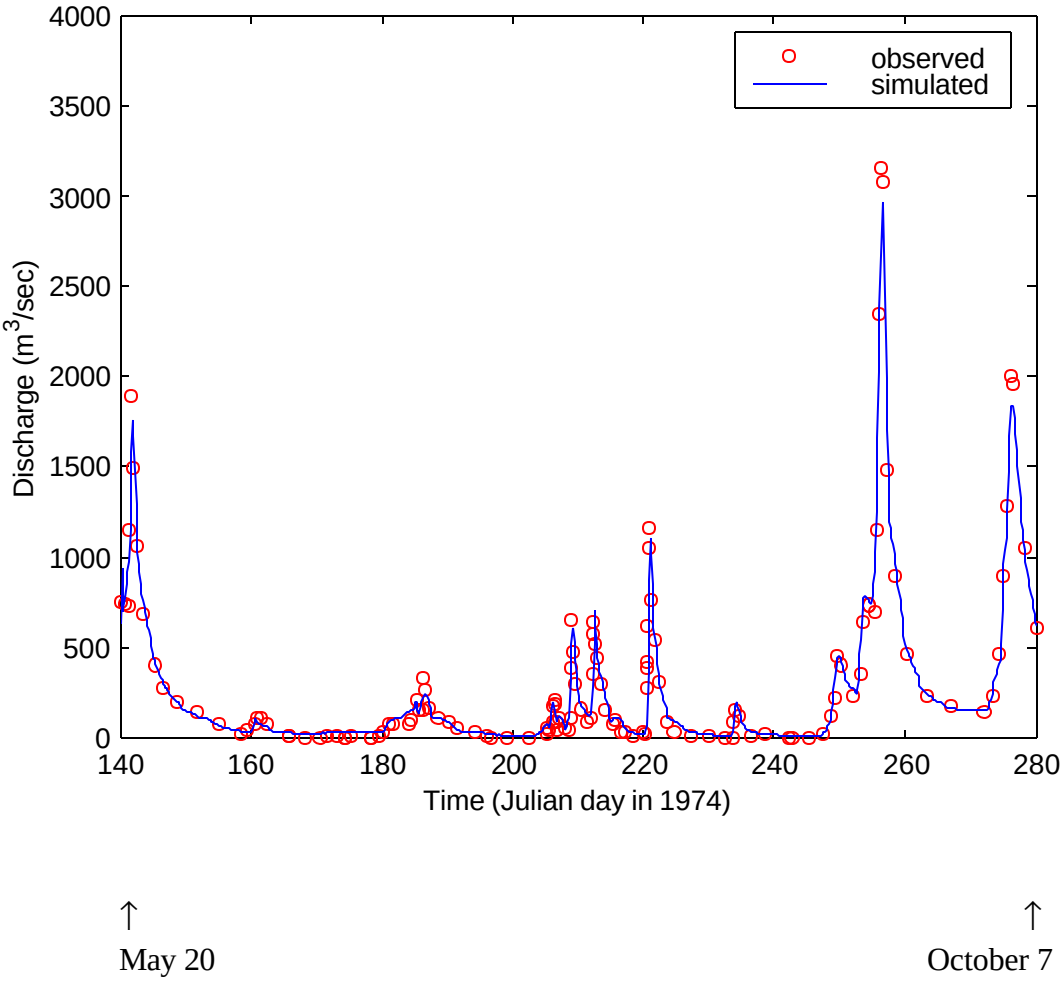


Figure 5-8. Discharge at Weinan Gauging Station, evaluation, n = 0.020

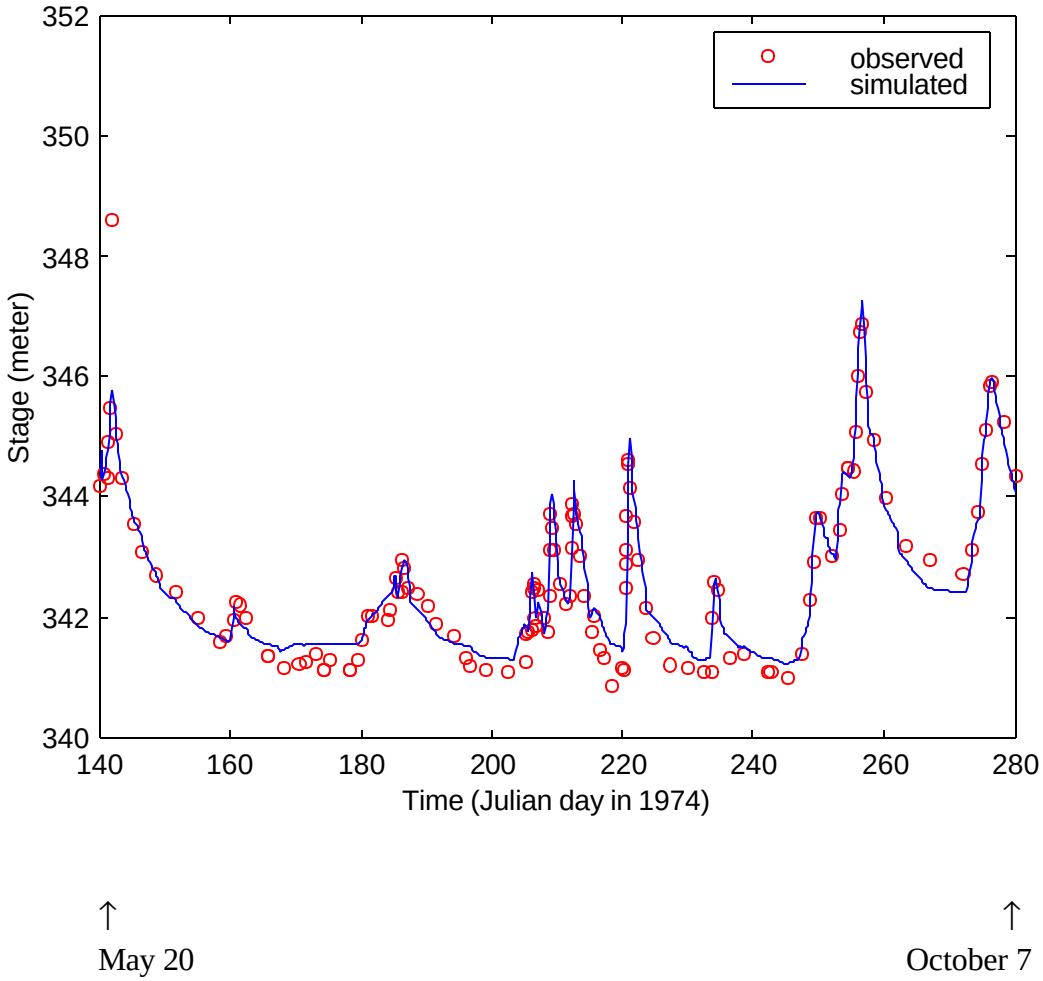


Figure 5-9. Surface elevation at Weinan Gauging Station, evaluation, n = 0.020

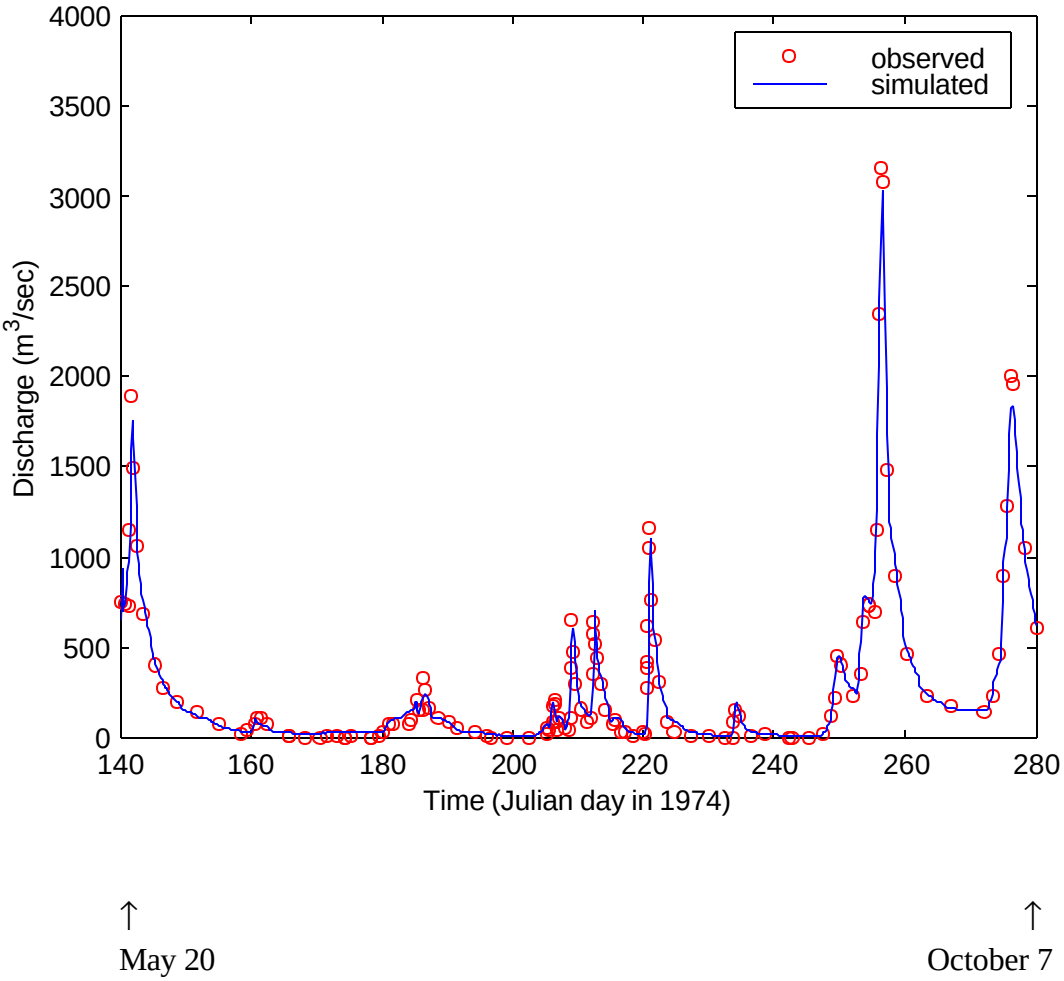


Figure 5-10. Discharge at Weinan Gauging Station, *n* is a function of depth

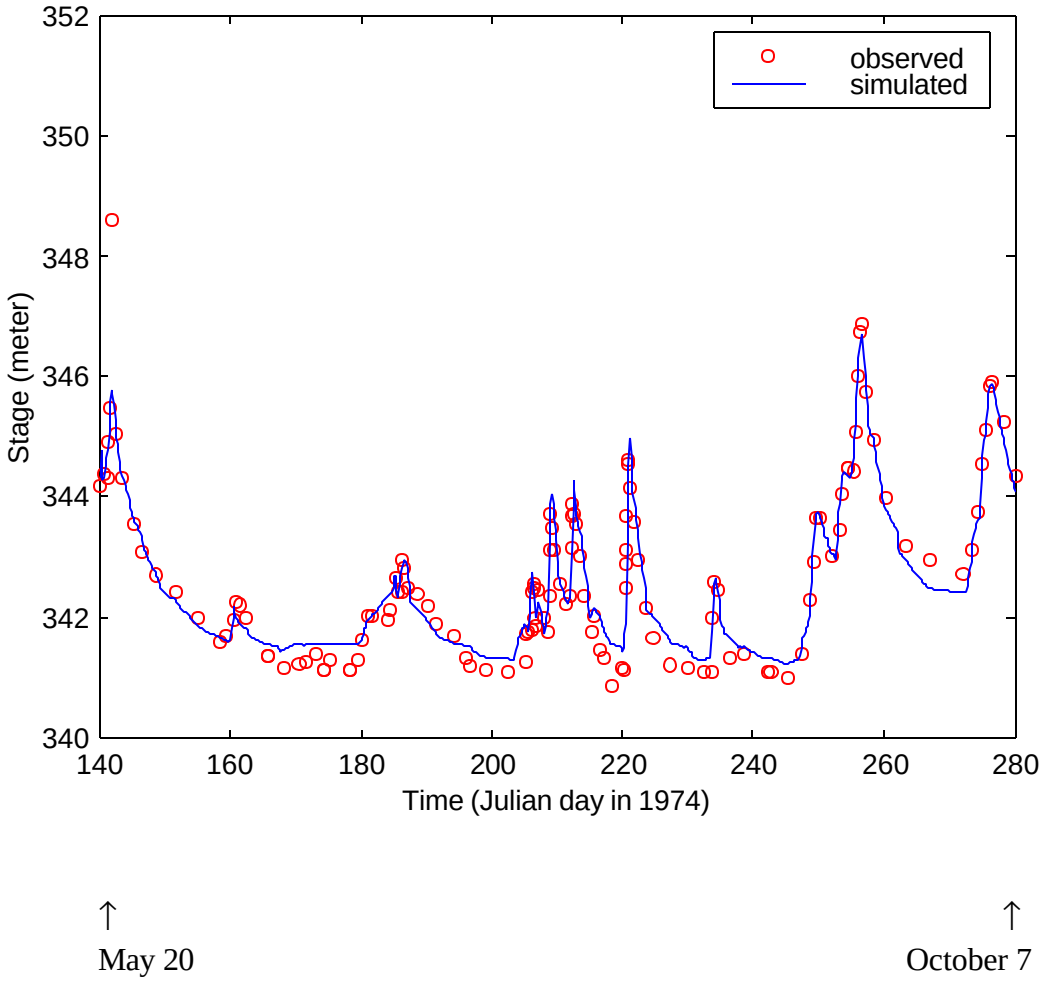


Figure 5-11. Surface elevation at Weinan Gauging Station, *n* is a function of depth

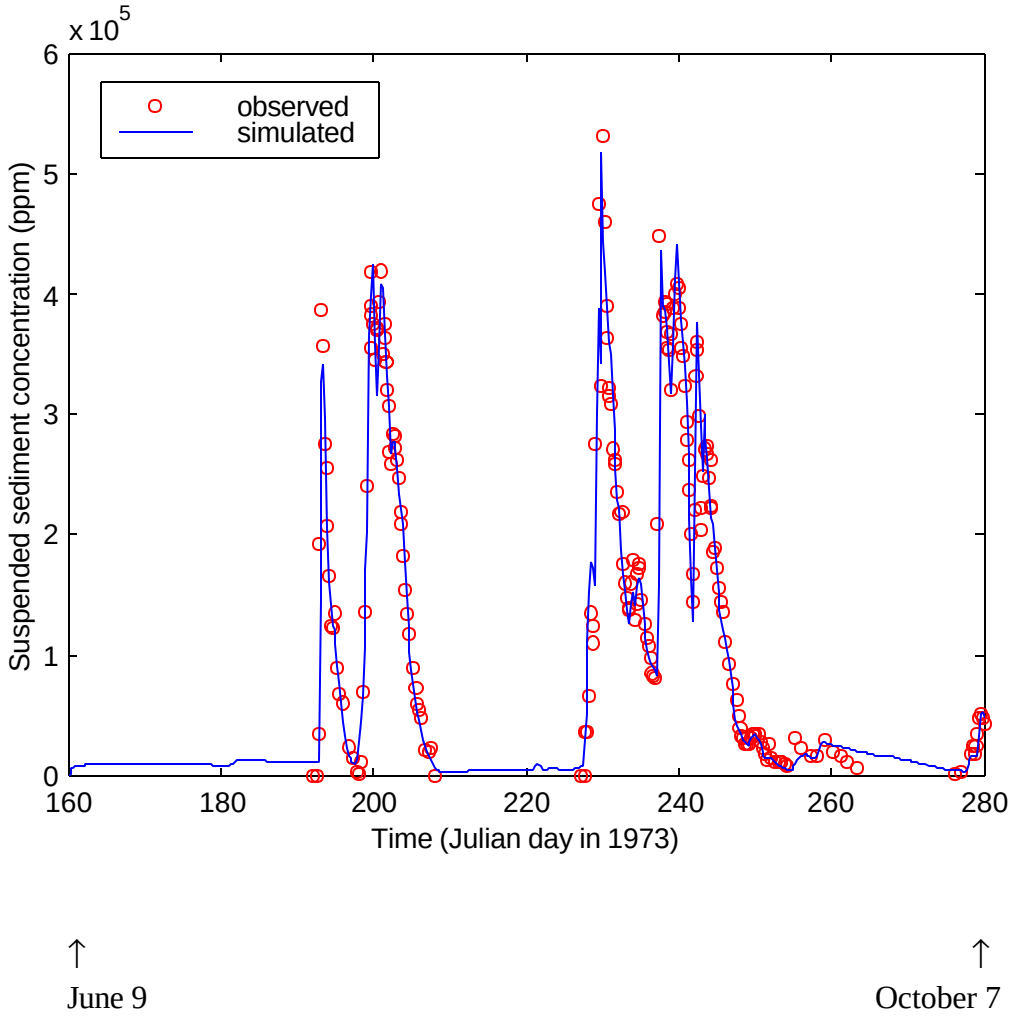


Figure 5-12. Suspended sediment concentration at Weinan Gauging Station, calibration

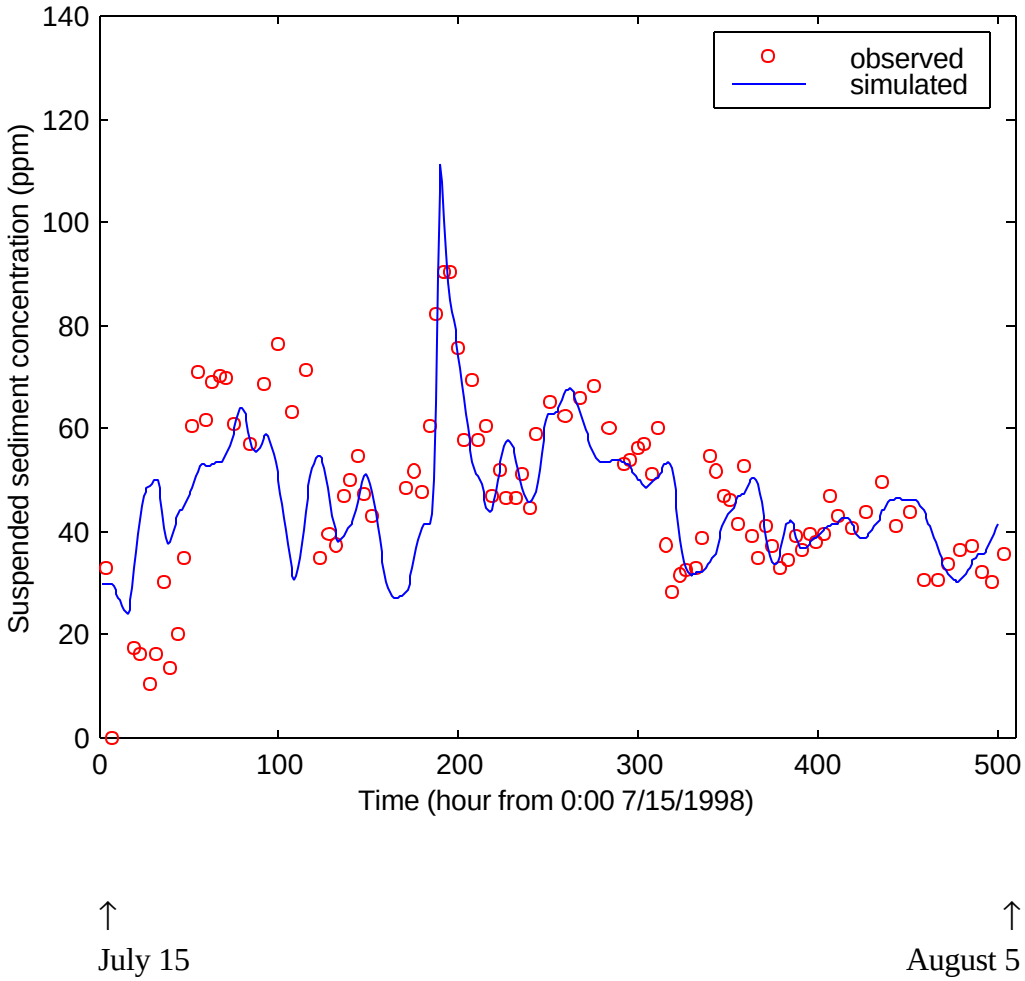


Figure 5-13. Suspended sediment concentration at Penfield, calibration

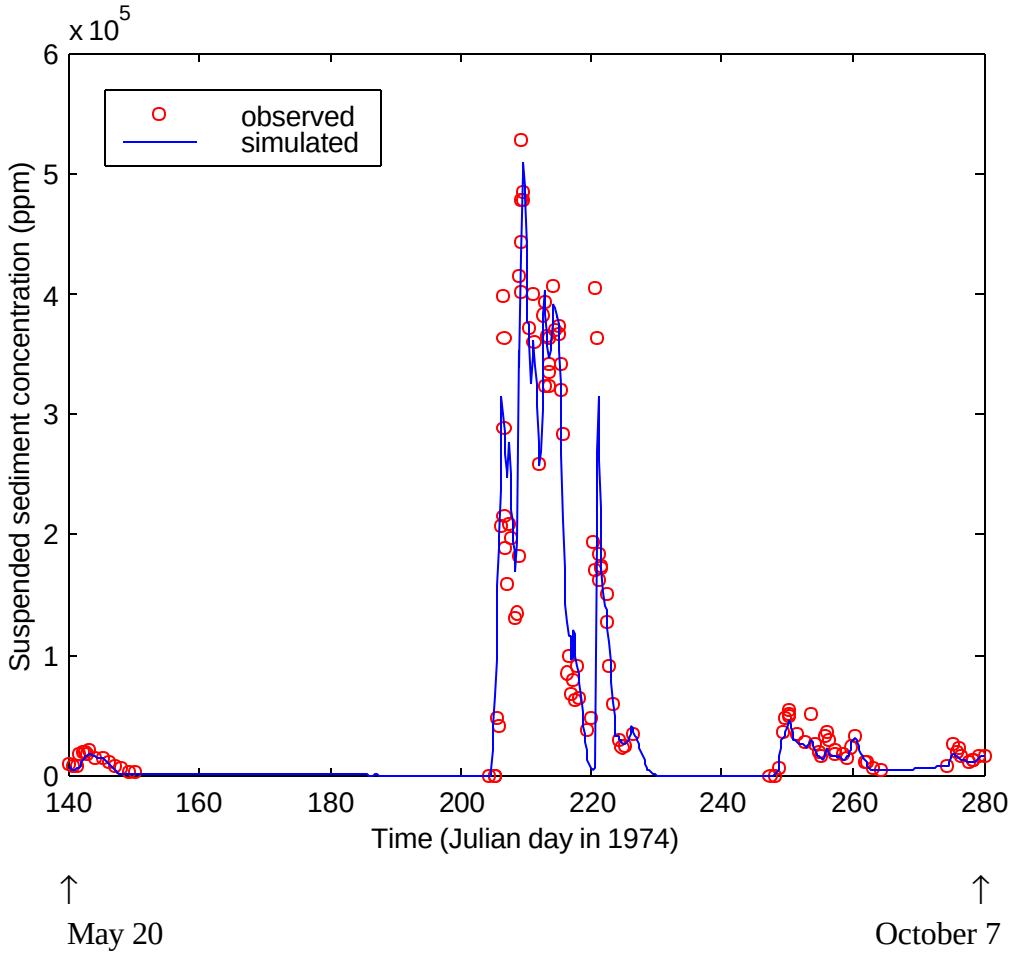
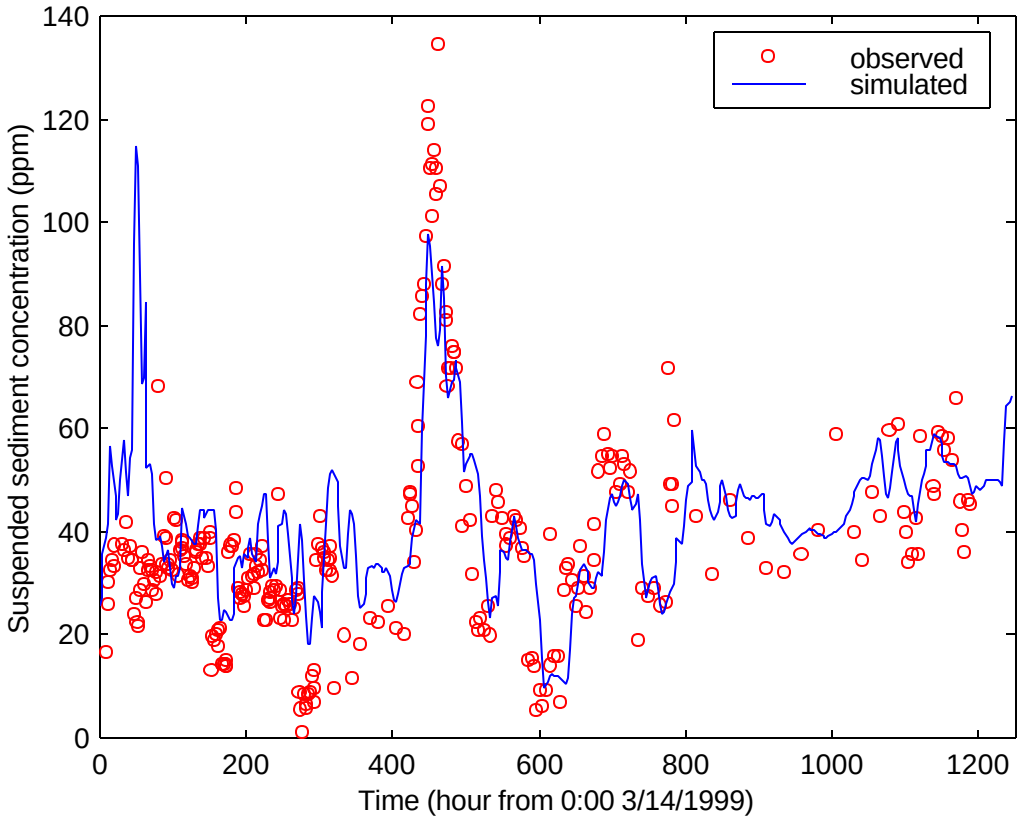


Figure 5-14. Suspended sediment concentrations at Weinan Gauging Station, evaluation



↑ March 14 ↑ May 5

Figure 5-15. Suspended sediment concentration at Penfield, evaluation

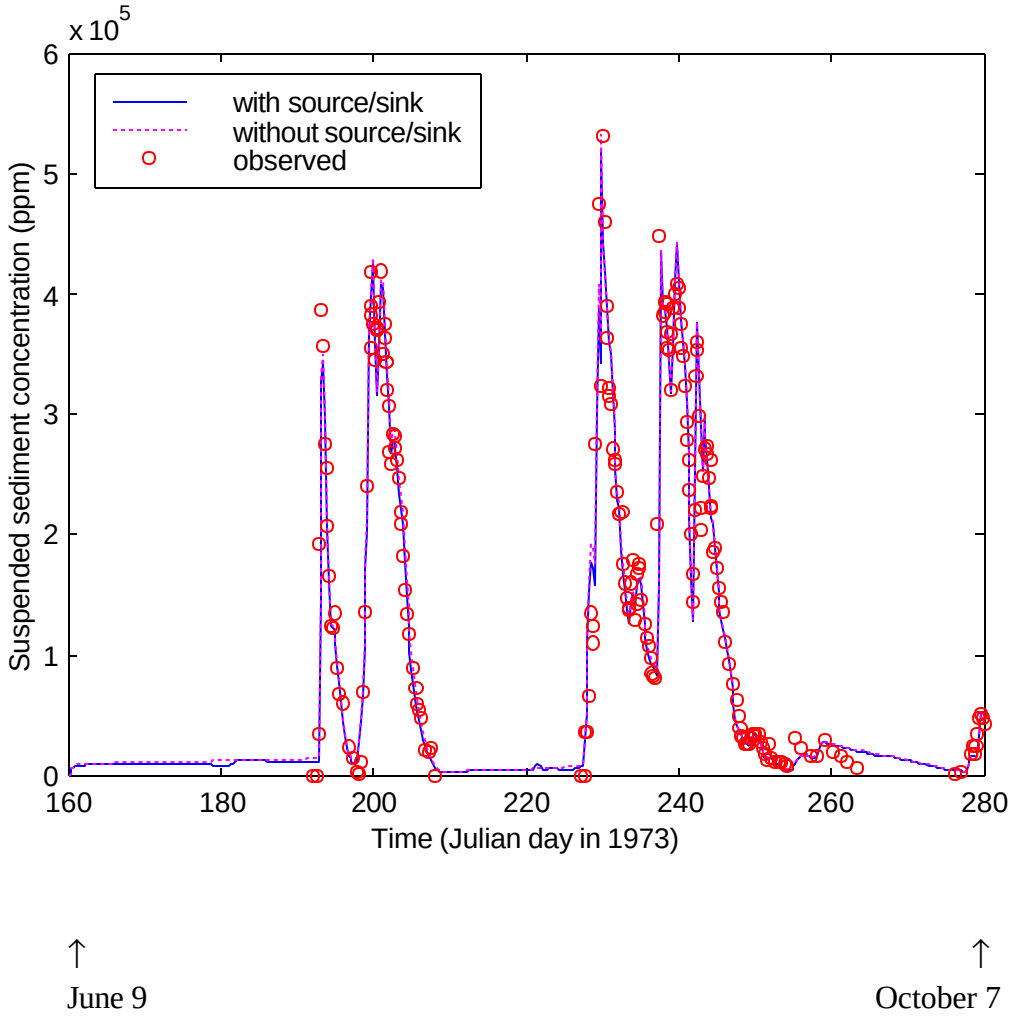


Figure 5-16. Comparison between simulations of suspended sediment concentrations with and without source/sink mechanisms, at Weinan Gauging Station, the Weihe River, 1973

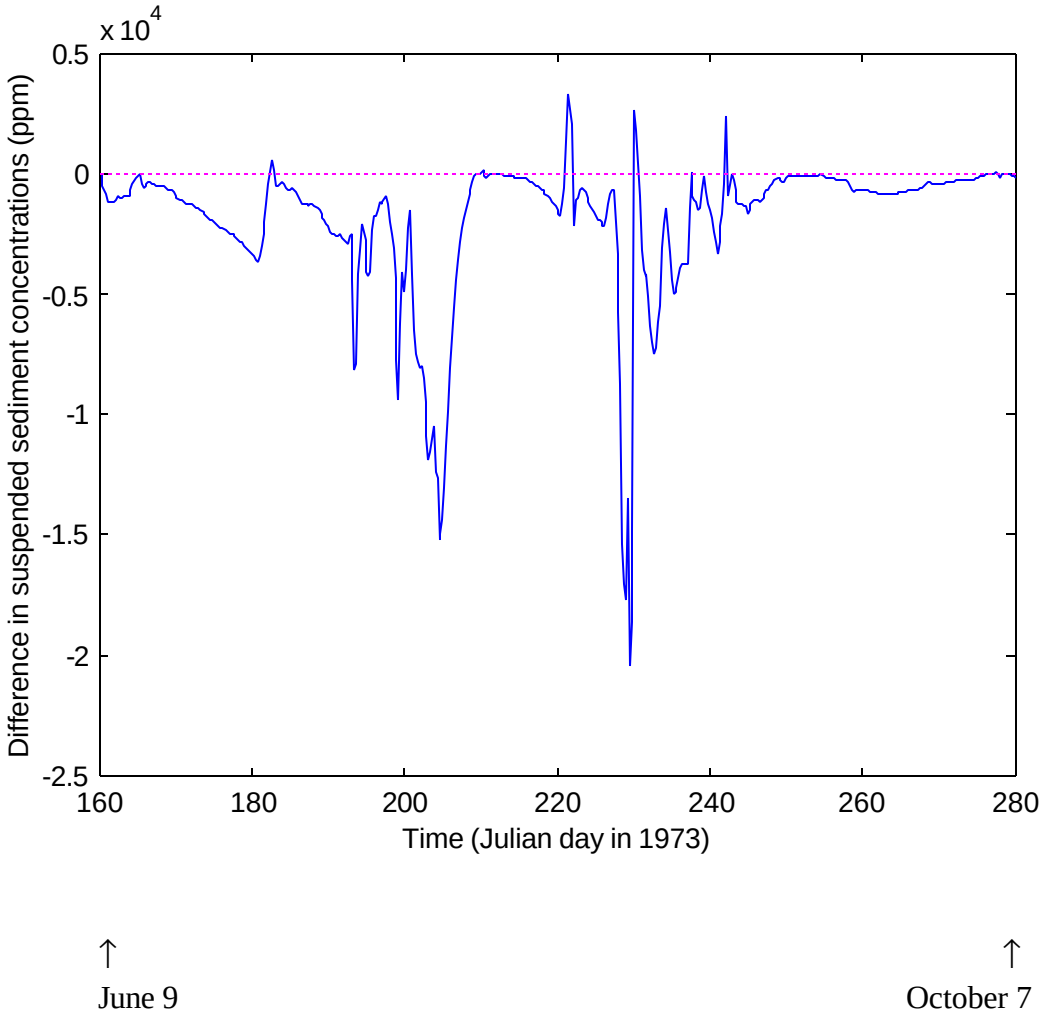


Figure 5-17. Difference between sediment computations with and without source/sink mechanisms, at Weinan Gauging Station, the Weihe River, 1973

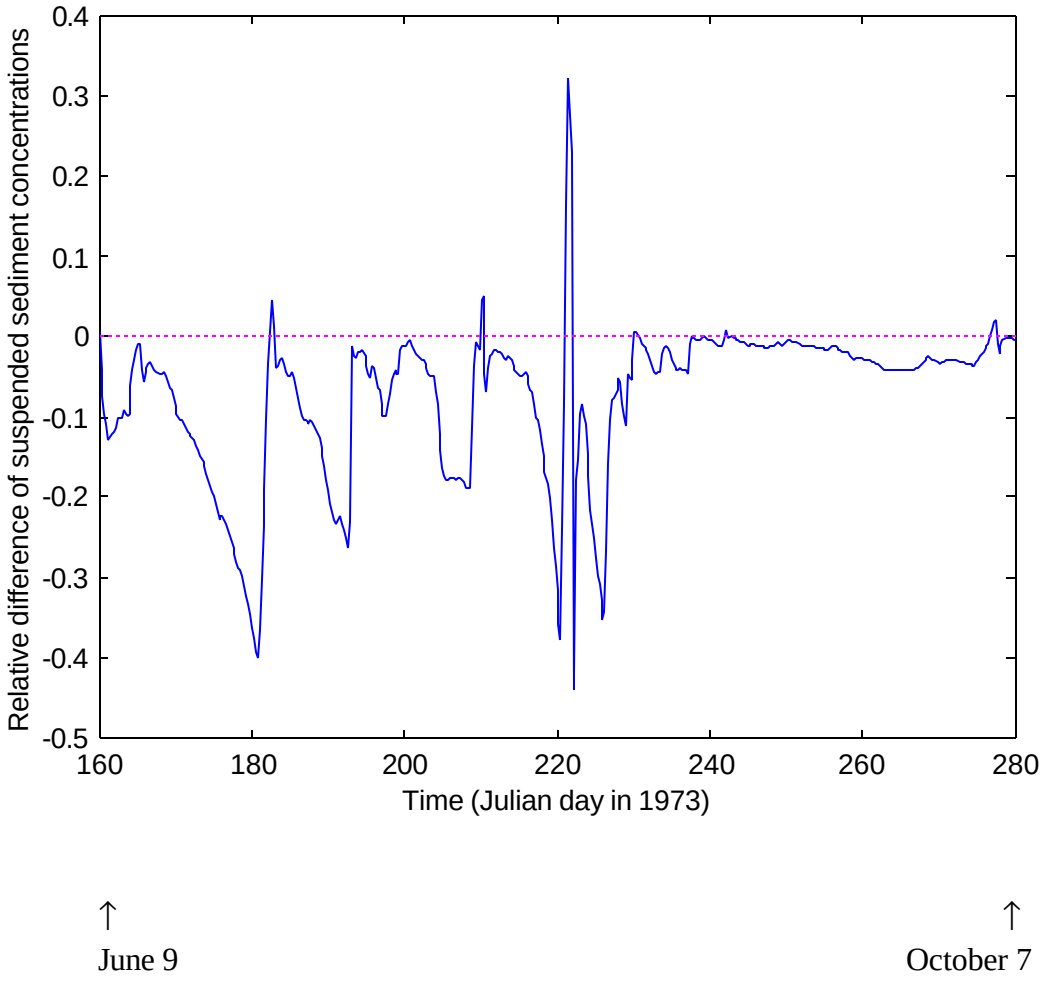


Figure 5-18. Relative difference between simulations with and without source/sink mechanisms, at Weinan Gauging Station, the Weihe River, 1973

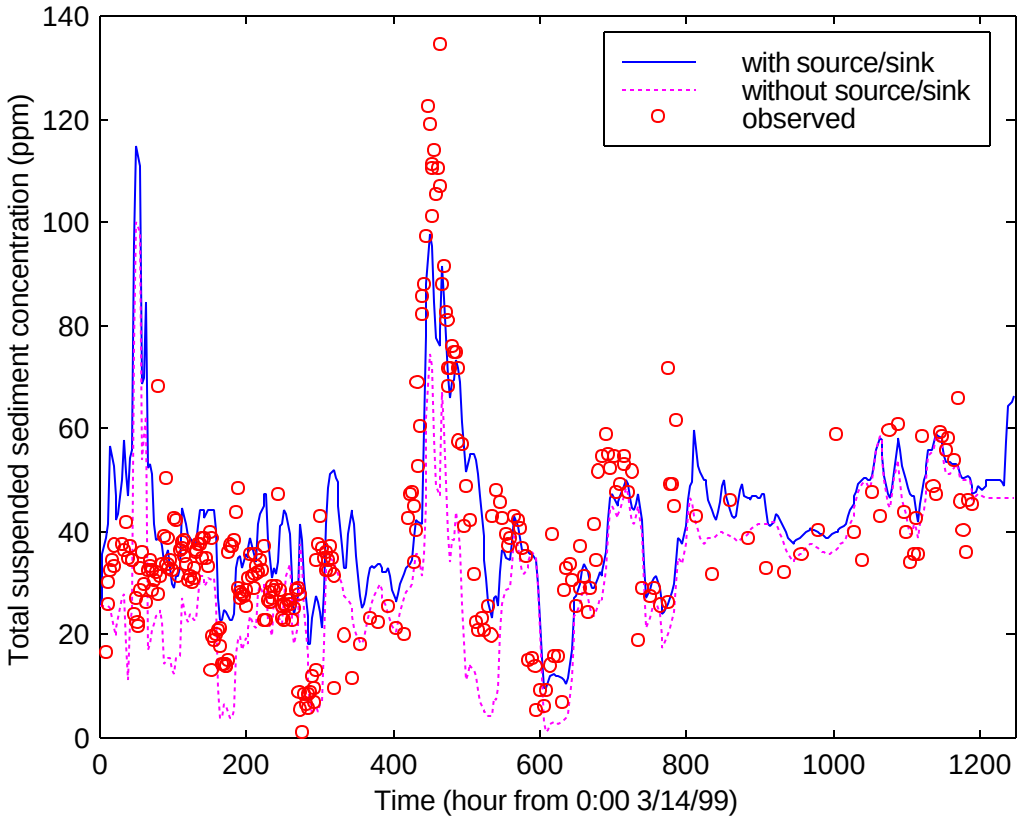


Figure 5-19. Comparison between simulations of total suspended sediment concentrations with and without source/sink mechanisms, at Penfield, the Oconee River, 1999

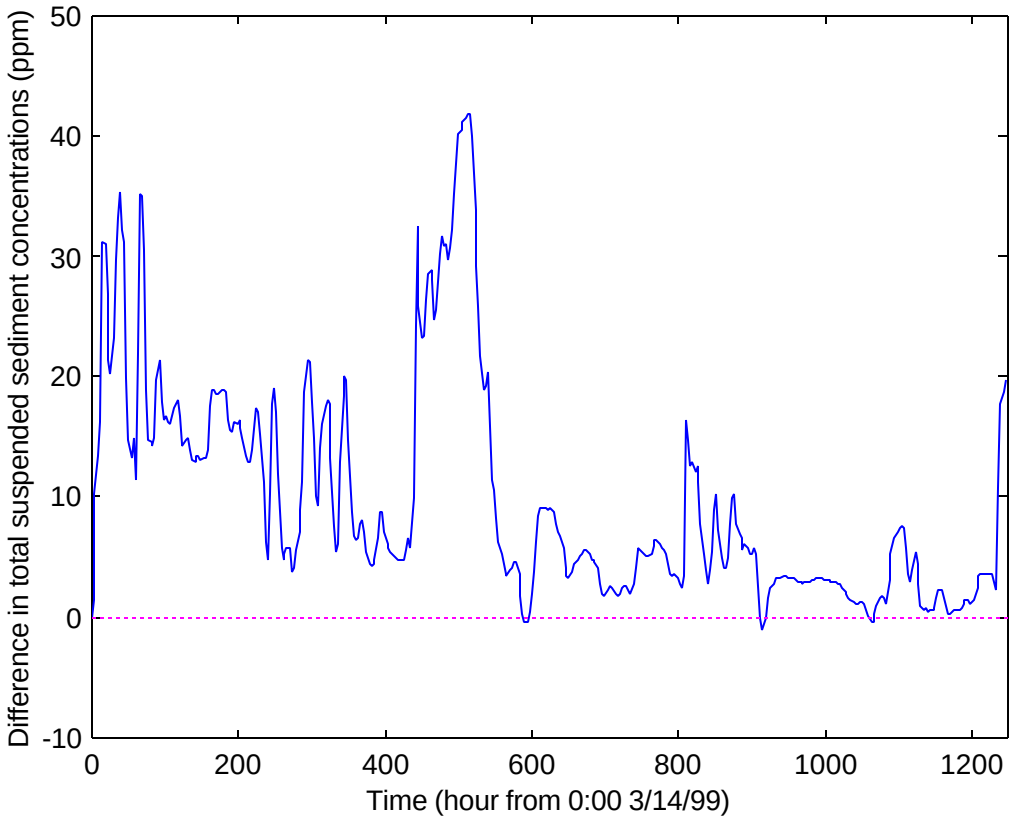


Figure 5-20. Difference between sediment computations with and without source/sink mechanisms, at Penfield, the Oconee River, 1999

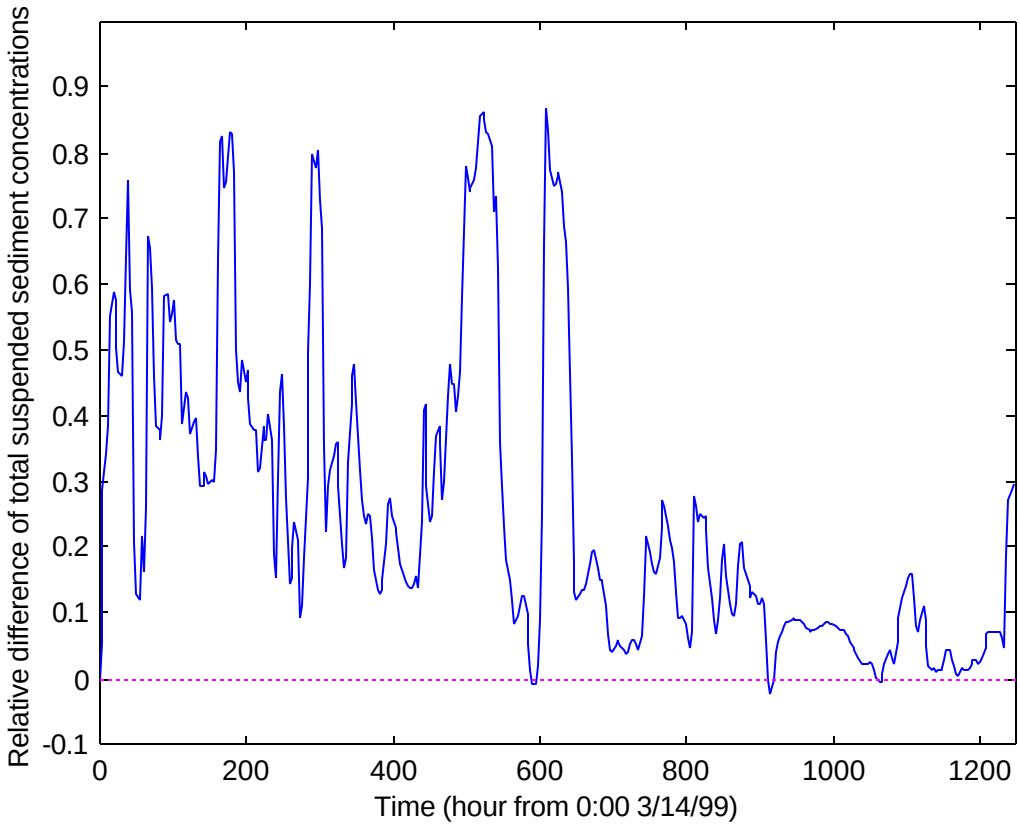
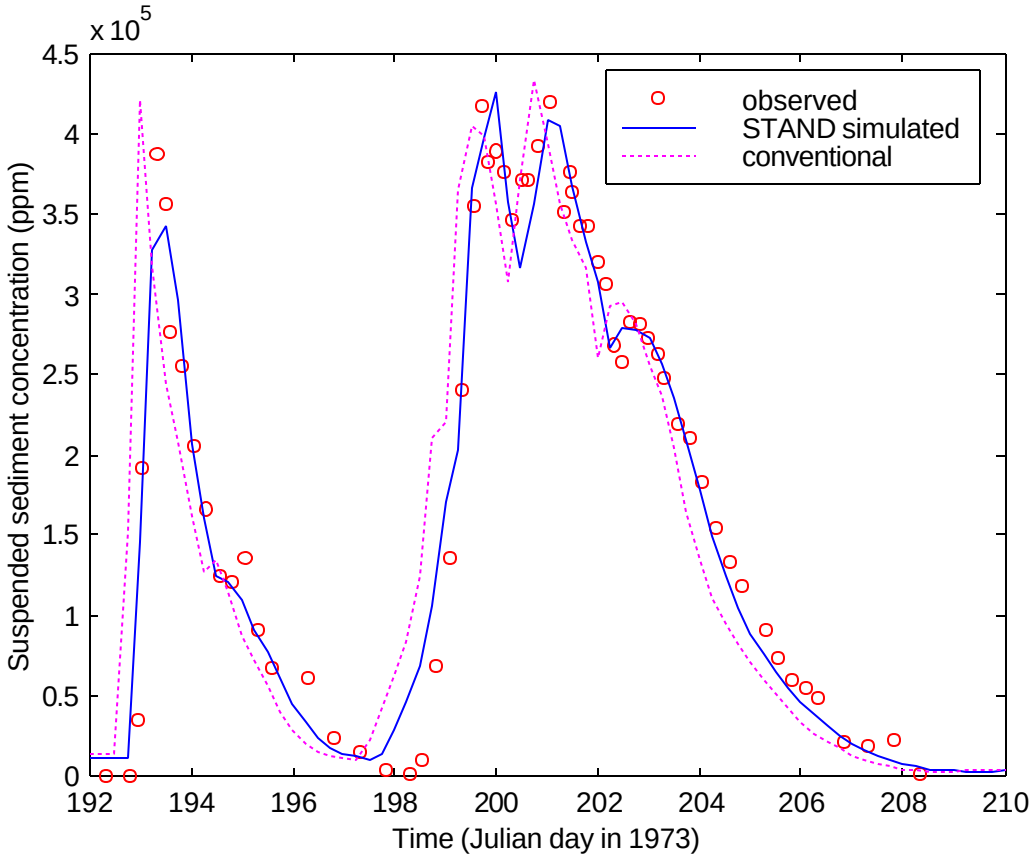
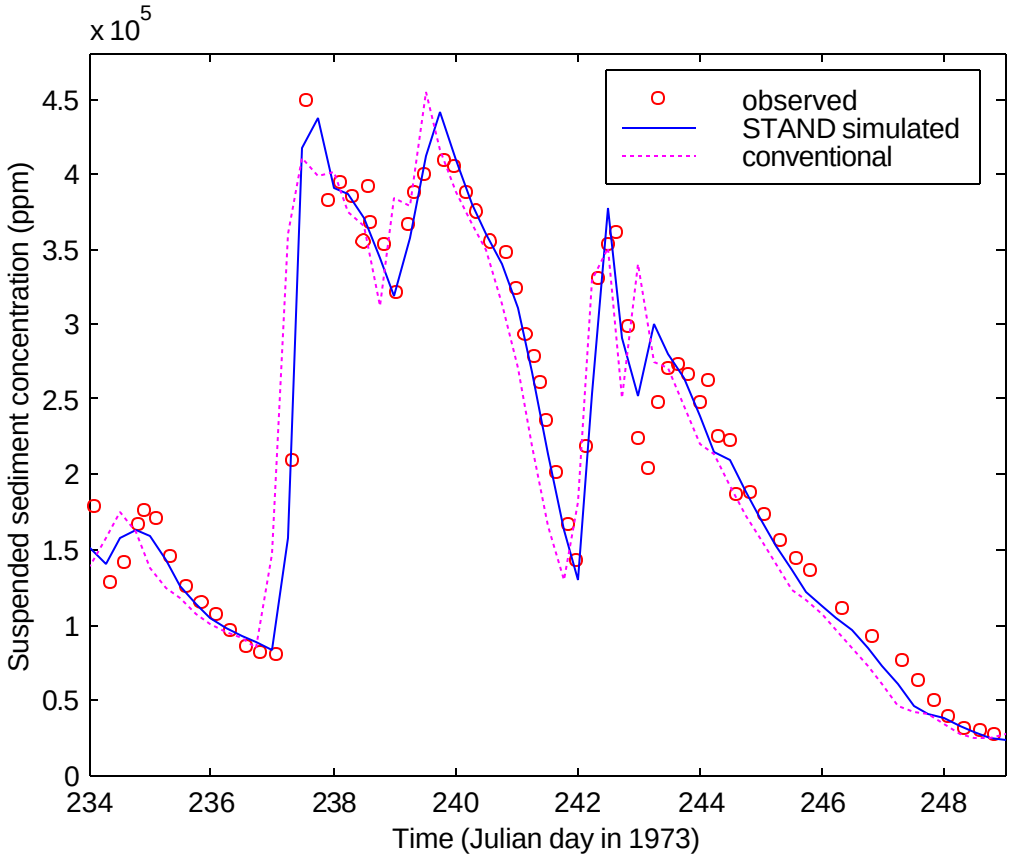


Figure 5-21. Relative difference between sediment computations with and without source/sink mechanisms, at Penfield, the Oconee River, 1999



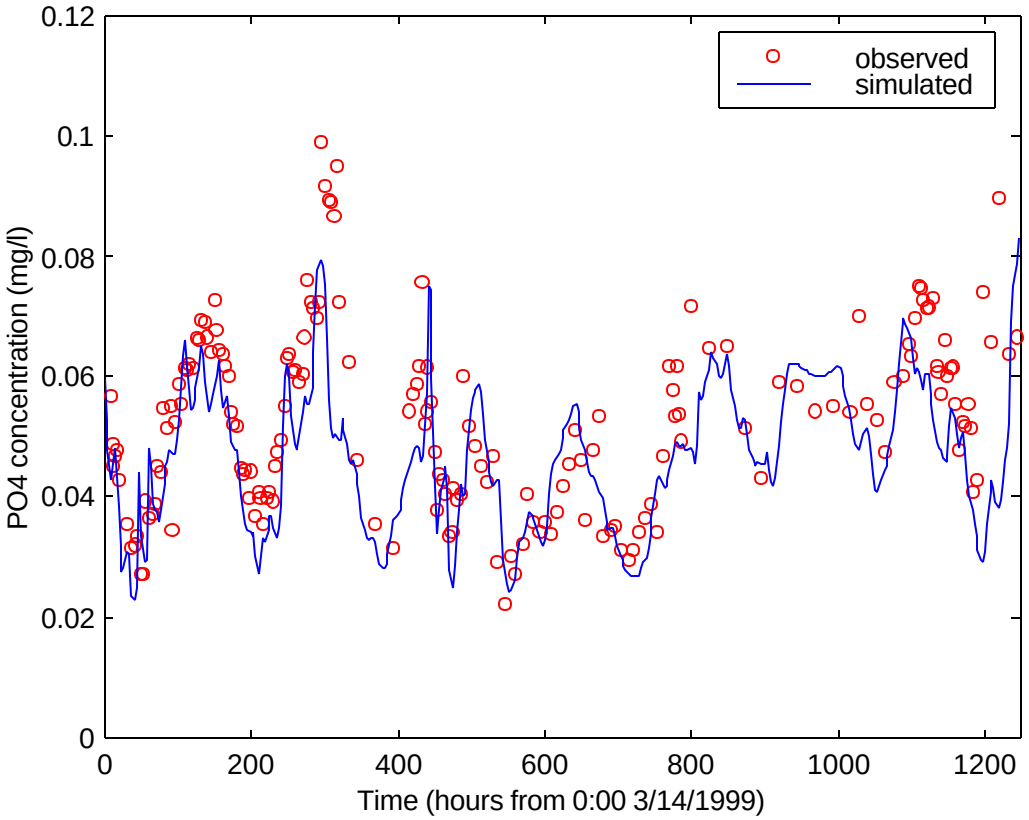
↑ July 11 ↑ July 29

Figure 5-22. Comparison between conventional methods and the adopted one, suspended sediment concentration at Weinan Gauging Station



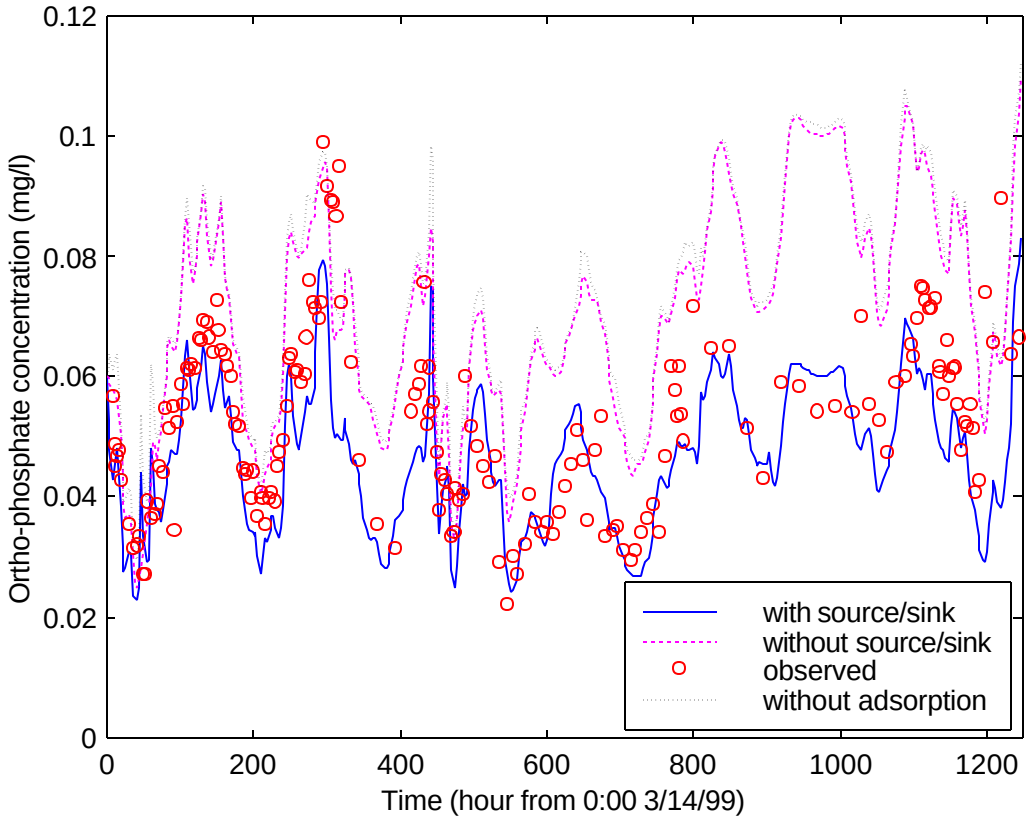
↑ August 22 ↑ September 6

Figure 5-23. Comparison between conventional methods and the adopted one, suspended sediment concentration at Weinan Gauging Station



↑ March 14 ↑ May 5

Figure 5-24. Ortho-phosphate concentration at Penfield (1999)



↑ March 14 ↑ May 5

Figure 5-25. Comparison between simulations of Ortho-phosphate concentrations with and without source/sink mechanism, at Penfield, the Oconee River, 1999

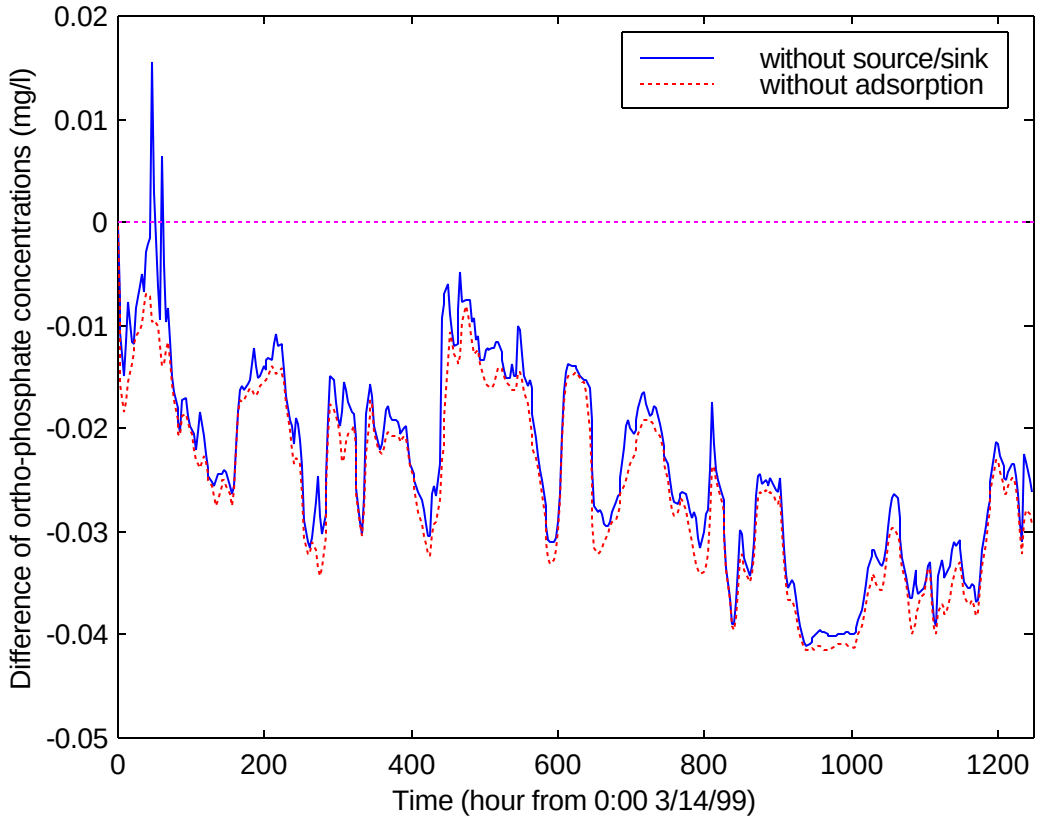
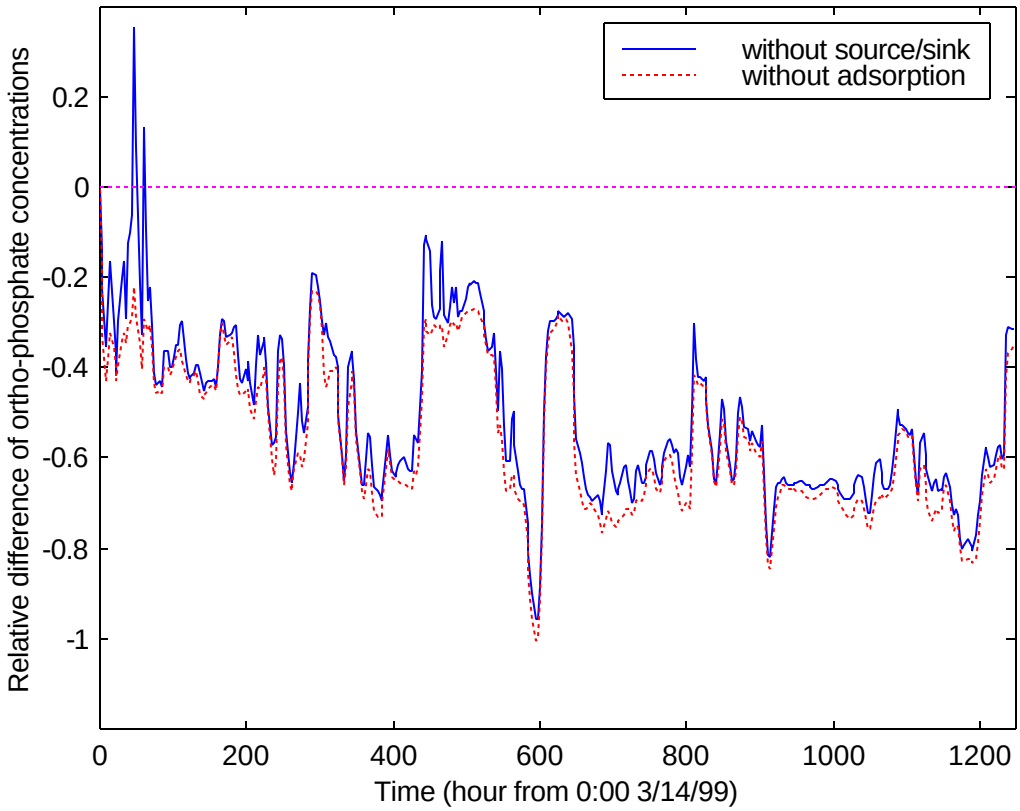


Figure 5-26. The difference between computations of ortho-phosphate-P concentrations with and without source/sink mechanism (blue solid curve); and difference between ortho-phosphate computations with and without the adsorption mechanism (red dashed curve)



↑ March 14 ↑ May 5

Figure 5-27. The relative difference between computations of ortho-phosphate-P concentrations with and without source/sink mechanism (blue solid curve); and difference between ortho-phosphate computations with and without the adsorption mechanism (red dashed curve)

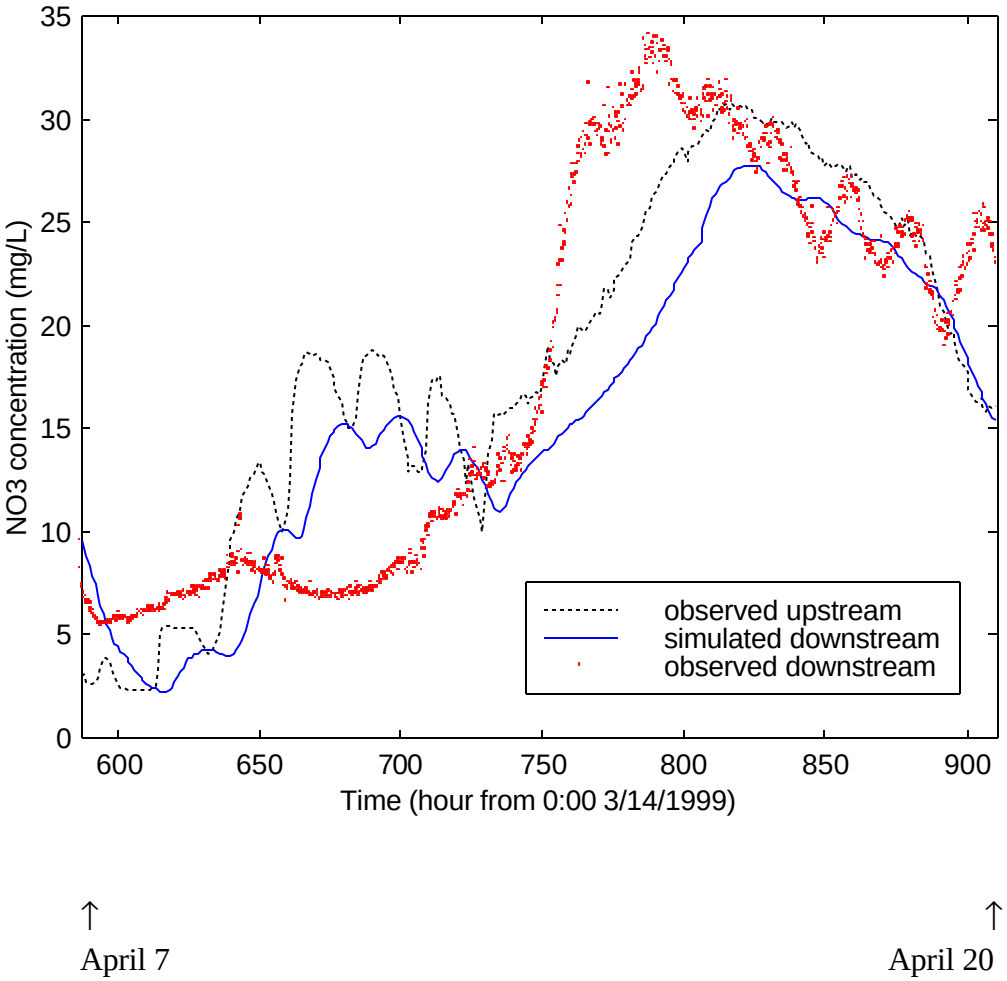


Figure 5-28. Nitrate concentration at Penfield (1999)

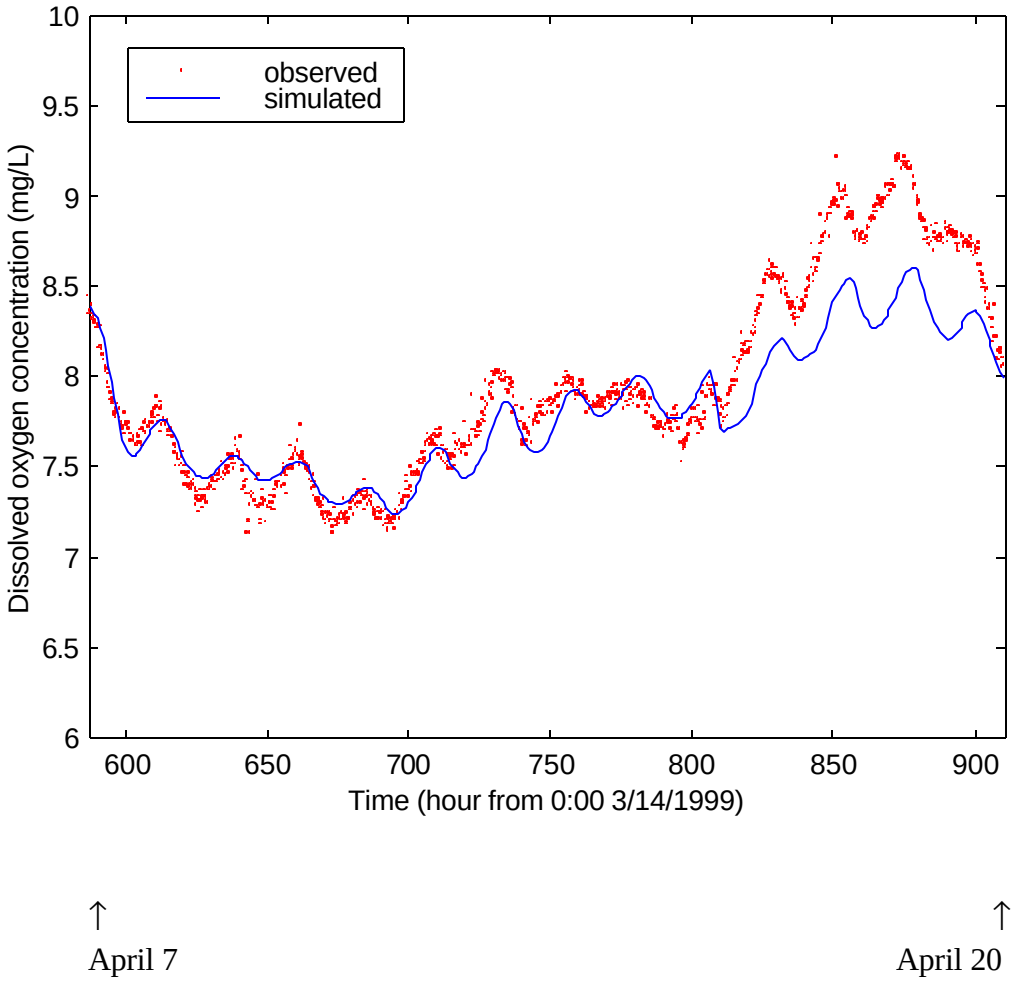


Figure 5-29. Dissolved oxygen concentration at Penfield (1999)

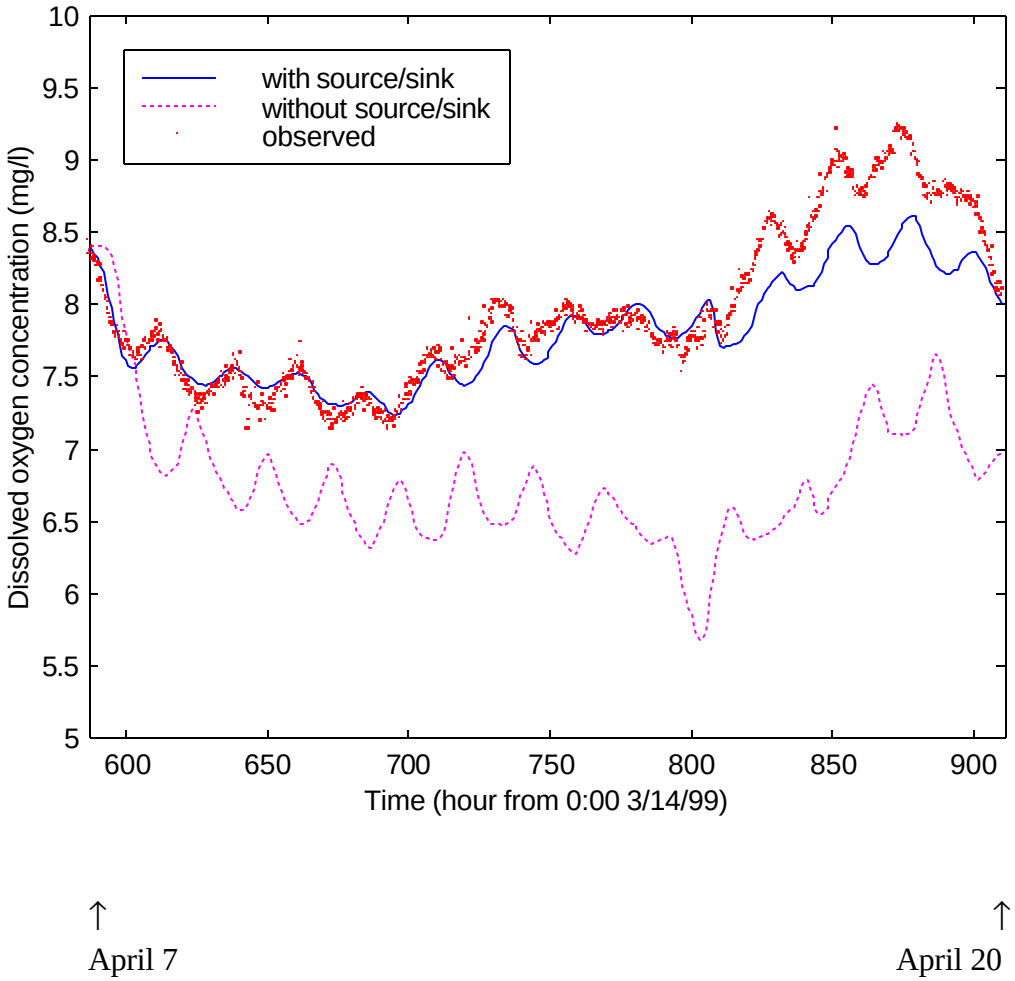


Figure 5-30. Comparison between simulations of dissolved oxygen concentrations with and without source/sink mechanisms, at Penfield, the Oconee River, 1999

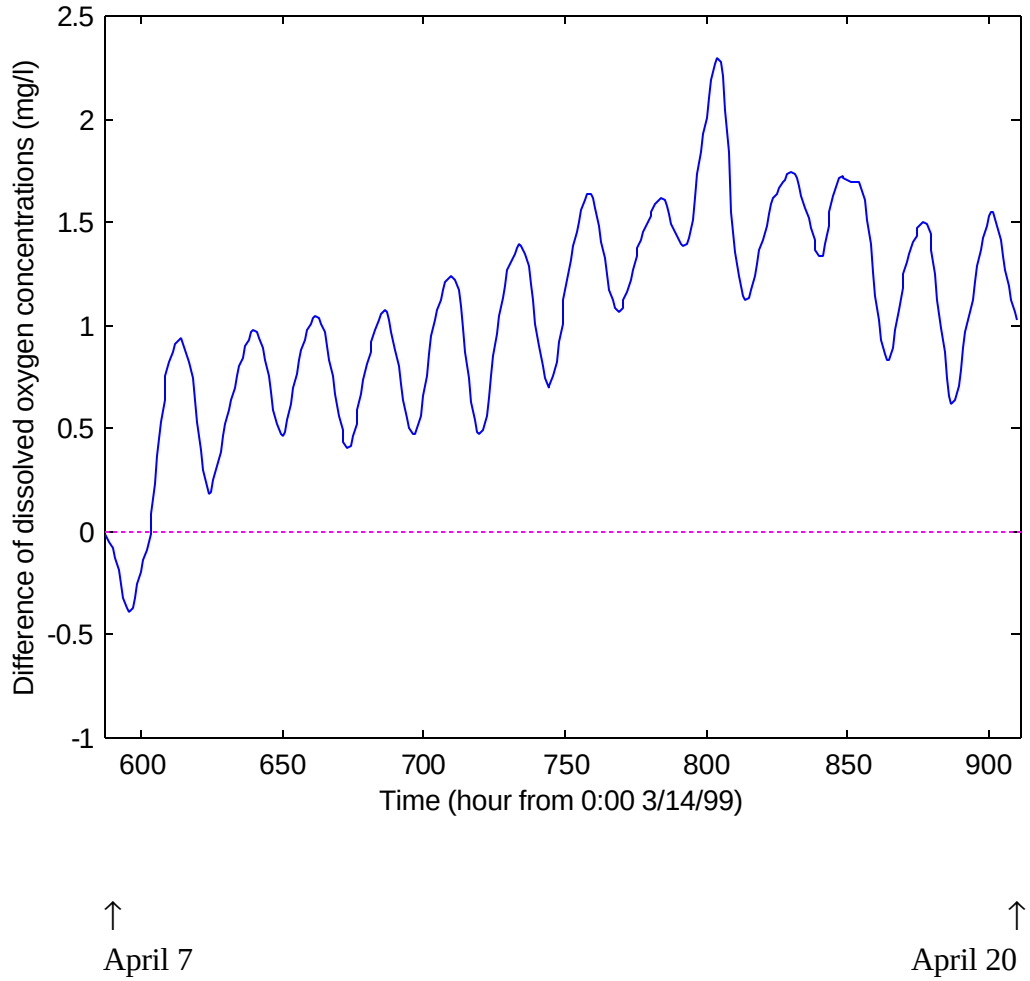


Figure 5-31. Difference between dissolved oxygen computations with and without source/sink mechanism, Penfield, the Oconee River, 1999

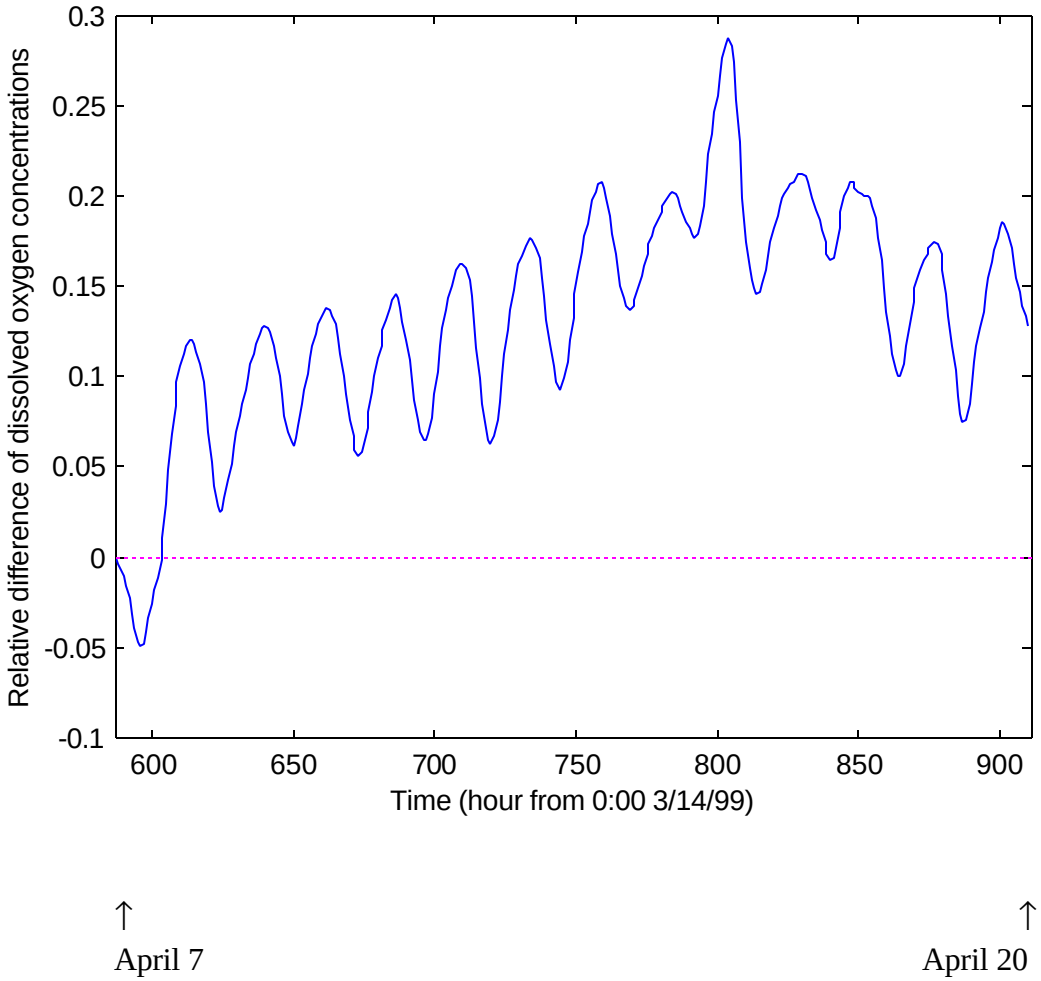


Figure 5-32. Relative difference between dissolved oxygen computations with and without source/sink mechanism, Penfield, the Oconee River, 1999

CHAPTER 6

UNCERTAINTY ANALYSIS

Having calibrated (and evaluated parts of) the model, it is natural to question the uncertainty involved in the model and its parameters. In this chapter, the issue of uncertainty is analyzed using Monte Carlo Simulation. We try to obtain a clear view of how much uncertainty there is in each component of the model and identify which parameter affects the simulation results most significantly. The studied reach is the Oconee River. A range of values is assigned to each parameter in the model. The ranges are determined by adding and subtracting a certain value to and from the calibrated parameter values. Five hundred values for each parameter are generated within the range assuming that the probability of the parameter is of uniform distribution. The five hundred combinations of randomly selected parameter values were drawn and fed to the model. Boundary conditions of the model are the data obtained in the 1999 campaign. The ranges of the parameters are presented in Table 6-1.

Table 6-1 Parameter values for the Monte Carlo Simulation

Parameter	*range of value*	*units*
n	0.030 – 0.035	-
E_s	500.0 – 1000.0	m^2/s
k_{SedDep}	1.0×10^{-9} – 1.0×10^{-8}	1/s
k_{SedEnt}	1.0×10^{-4} – 1.0×10^{-3}	1/s
l	0.30 – 0.80	-

Table 6-1 Parameter values for the Monte Carlo Simulation (continued)

E_P	300.0 – 700.0	m^2/s
k_P	4×10^{-6} – 1.2×10^{-5}	1/s
C_{Pint}	0 – 0.2	mg/l
k_f	1.25×10^{-3} – 3.75×10^{-3}	-
p_f	0.025 – 0.075	-
E_{nitr}	300.0 – 700.0	m^2/s
k_{DeNi}	0 – 1.0×10^{-4}	1/s
k_{Nitr}	0 – 1.0×10^{-4}	1/s
E_{ammo}	1.0- 1100.0	m^2/s
k_{AmAd}	0 – 0.5	1/s
k_{amf}	0 – 1.0×10^{-4}	-
p_{amf}	0 – 0.1	-
M_n	1.5 – 6.5	mg/l
M_{am}	0 – 0.1	mg/l
E_{oxy}	1.0 – 100.0	m^2/s
a_3	1.4 – 1.8	-
a_4	1.6 – 2.3	-
m	1.16×10^{-5} – 3.47×10^{-5}	1/s
r	5.79×10^{-7} – 5.79×10^{-6}	1/s
r_{SOD}	1.5×10^{-5} – 4.5×10^{-5}	g/m^2s
a_5	3.0 – 4.0	-
b_1	0.1 – 1.0	1/s

The location of concern is the downstream end of the studied reach, Penfield. For the entire simulation period, all state variables at this location have been recorded for the 500 parameter combinations. The results are shown in Figure 6-1 through 6-14.

The simulation results of discharge at Penfield are shown in Figure 6-1. The maximum and minimum envelop values are so close to the mean values that they can hardly be distinguished visually. Because the uncertainty in hydraulic computation mainly comes from the determination of the Manning's coefficient, the range of the coefficient value is small, and the magnitude of the discharge is not too high to have significant variations out of the small variations of the coefficient, so little uncertainty associated with the hydraulics is reasonable and understandable. However, since no detailed morphological data are available, and given that cross-section profiles were determined arbitrarily, there can be significant uncertainty associated with river morphology and thus hydraulics. This issue is interesting, but it falls out of the scope of this dissertation and thus is not addressed here. The statistics for the discharges at time ≈ 500 and 800 hours are shown in Table 6-2. It can be seen that the range and standard deviation are fairly small.

Figure 6-2 shows the simulation results of suspended sediment concentration at Penfield. The maximum and minimum envelop can be seen clearly. The uncertainties do not seem to be too significant, seen from the range of variations in the simulation results, partly because of the narrow range chosen for the parameters. The frequency distribution at time = 420 hours and 1000 hours are shown in Figures 6-3 and 6-4. A Bell shaped curve can be seen in both figures, which implies that the distribution of the results might be normal. The statistics of the simulation results are shown in Table 6-3. The uncertainty associated with the computation of suspended sediment concentration appears to be more significant than with that of hydraulics. This is reasonable, because the understanding of sediment transport is not as mature as that of hydraulics, and because

the computation of sediment transport is based on the results of hydraulics. Nonetheless, the standard deviations and the coefficient of variations are still relatively small.

Table 6-2 Statistics of simulated discharge at Penfield

Statistics	Discharge (m^3/sec)	
	time = 500 hours	time = 800 hours
Mean	59.0228	18.2760
Minimum	59.0161	18.2717
Maximum	59.0289	18.2813
Range	0.0128	0.0096
Standard Deviation	0.0036	0.0027
Coef. Variation	0.006%	0.015%

Table 6-3 Statistics of simulated suspended sediment concentration at Penfield

Statistics	Suspended sediment concentration (ppm)	
	time = 420 hours	time = 1000 hours
Mean	30.0642	37.3150
Minimum	28.5891	35.9174
Maximum	31.5444	38.6011
Range	2.9553	2.6837
Standard Deviation	0.6786	0.5736
Coef. Variation	2.26%	1.54%

Figure 6-5 shows the simulation results of ortho-phosphate concentrations. The lower bound of the simulation is well limited, but the concentrations can occasionally reach very high levels compared to the average values. It was found that none of the changes of a *single* relevant parameter in the given ranges could generate results like that. It was also found that the combination of high values of certain parameters might have caused this phenomenon. Parameters that give simulation results with extreme behaviors were identified from the original parameter seed group. Figure 6-6 reveals that the combination of a high value of porosity and a high value of ortho-phosphate concentration in the interstitial water clearly contributes to the generation of extreme behaviors. This is reasonable and understandable, because once the bed sediment is scoured (or resuspended), the vast amount (caused by the high porosity) of release of (high ortho-phosphate containing) interstitial water does cause significant increase of ortho-phosphate in the overlying water column. But given the nature of a river, a very loose bed sediment layer and interstitial water with high ortho-phosphate concentration does not seem to last long. This conflicts with our assumption in the model of constant porosity and constant interstitial ortho-phosphate concentration. Thus the results with extreme behaviors were excluded for the following analysis. Figures 6-7 and 6-8 show the frequency distribution of the simulated ortho-phosphate concentration at time = 420 hours and 1000 hours. Skewed bell shaped curves can be seen. The statistics of the results are listed in Table 6-4. It can be seen that the distributions are skewed but the standard deviations are fairly small, considering that the computation of ortho-phosphate concentration is based on the hydraulics and sediment behavior computation and has inherited uncertainty from them.

Figure 6-9 shows the simulation results of dissolved oxygen concentration at Penfield. The average values are narrowly bounded by the maximum and minimum envelops. The frequency distribution at time = 600 hours and 800 hours are shown in Figure 6-10 and 6-11. The values seem to be fairly concentrated in narrow ranges and

uniformly distributed. The statistics are shown in Table 6-5. Again, given the fact that the dissolved oxygen component is on top of the first two levels of the model, the uncertainty represented by the statistics does seem to be very significant.

Table 6-4 Statistics of simulated ortho-phosphate concentration at Penfield

Statistics	Ortho-phosphate concentration (mg/l)	
	time = 420 hours	time = 1000 hours
Mean	0.0750	0.0900
Minimum	0.0650	0.0796
Maximum	0.1065	0.1264
Range	0.0415	0.0467
Standard Deviation	0.0062	0.0060
Coef. Variation	8.27%	6.66%

Table 6-5 Simulation results of dissolved oxygen concentration at Penfield

Statistics	Dissolved oxygen concentration (mg/l)	
	time = 600 hours	time = 800 hours
Mean	8.0683	8.2825
Minimum	7.5381	7.7342
Maximum	8.7408	8.7805
Range	1.2027	1.0463
Standard Deviation	0.3453	0.2975
Coef. Variation	4.23%	3.59%

There is possibilities of further exploring uncertainty issues using model STAND in future studies in engineering and management practice. It would be interesting to see how the issue of uncertainty affects decision making. But this will be left for future studies.

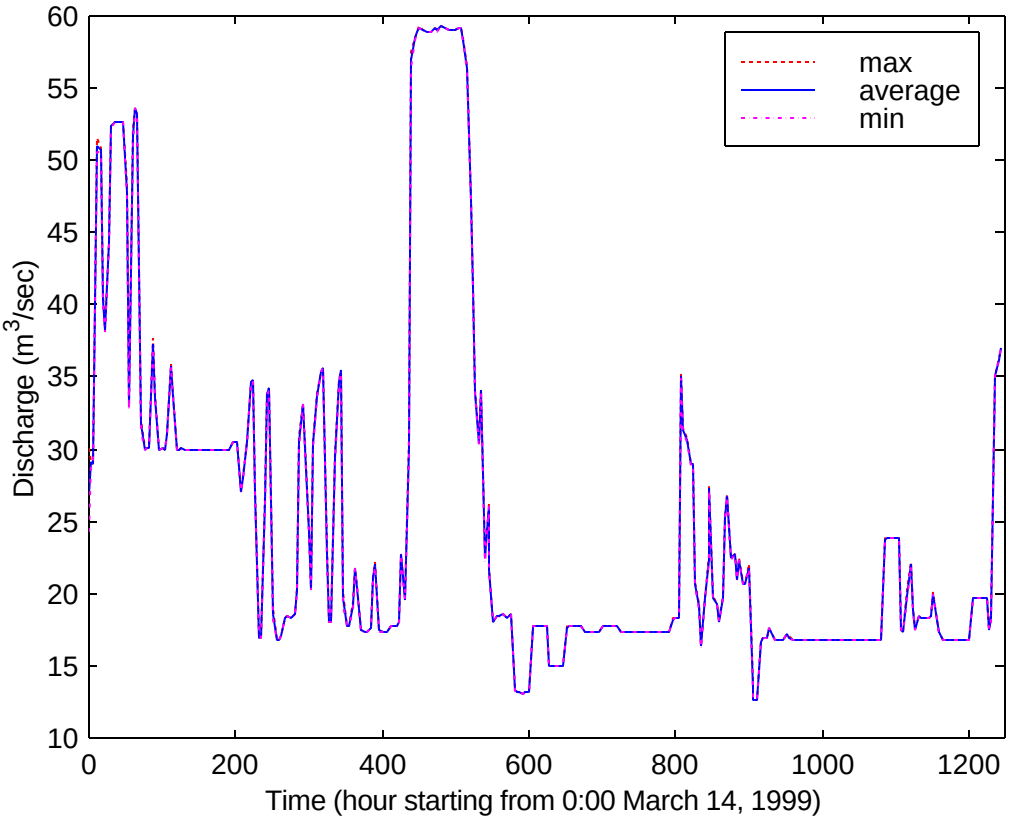


Figure 6-1. Simulation results of discharge at Penfield

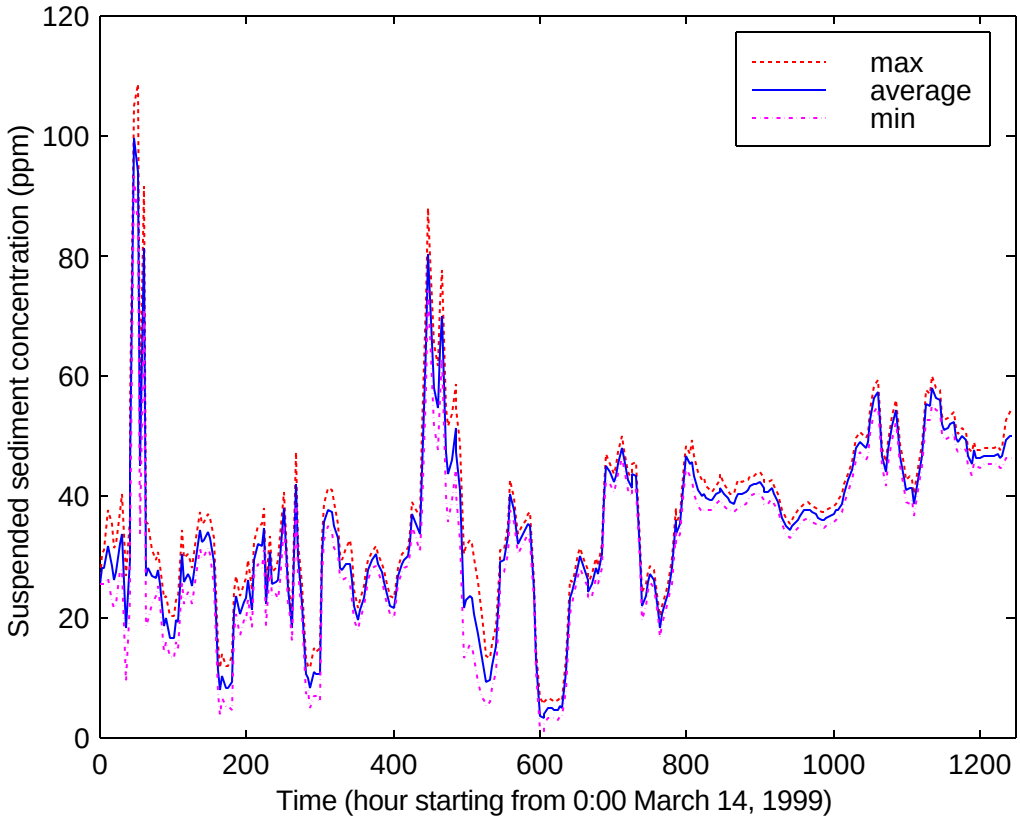


Figure 6-2. Simulation results of suspended sediment concentrations at Penfield

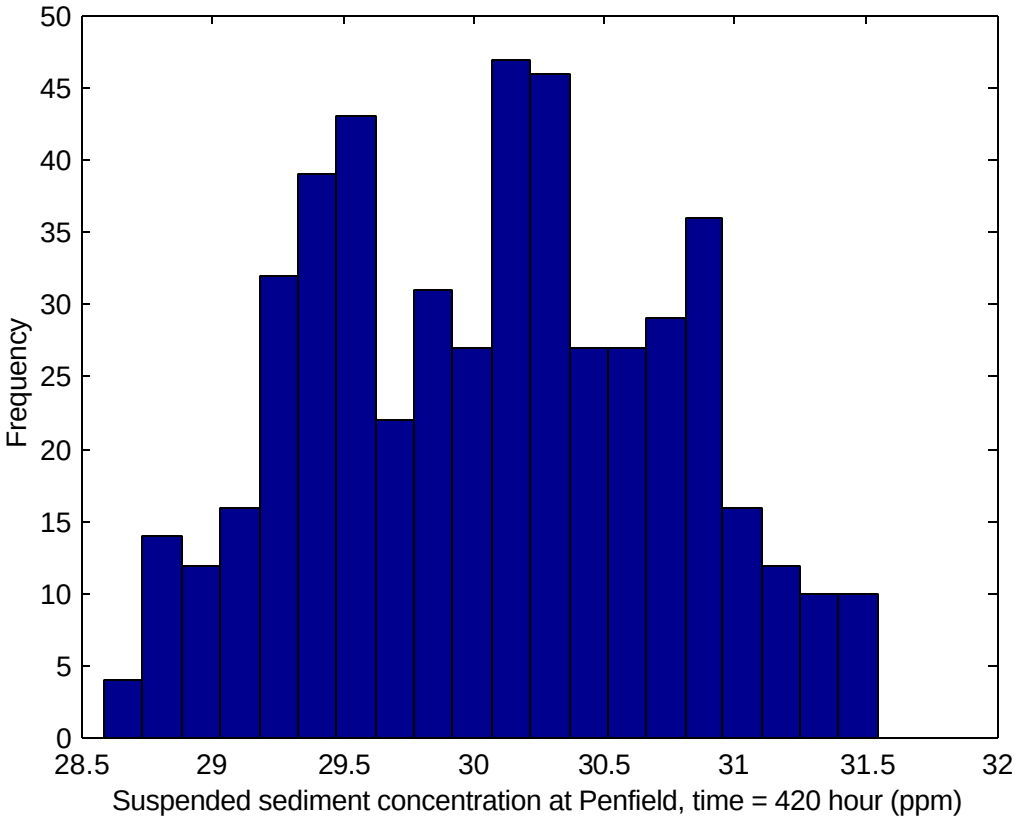


Figure 6-3. Histogram of suspended sediment concentration at Penfield when time = 420 hour

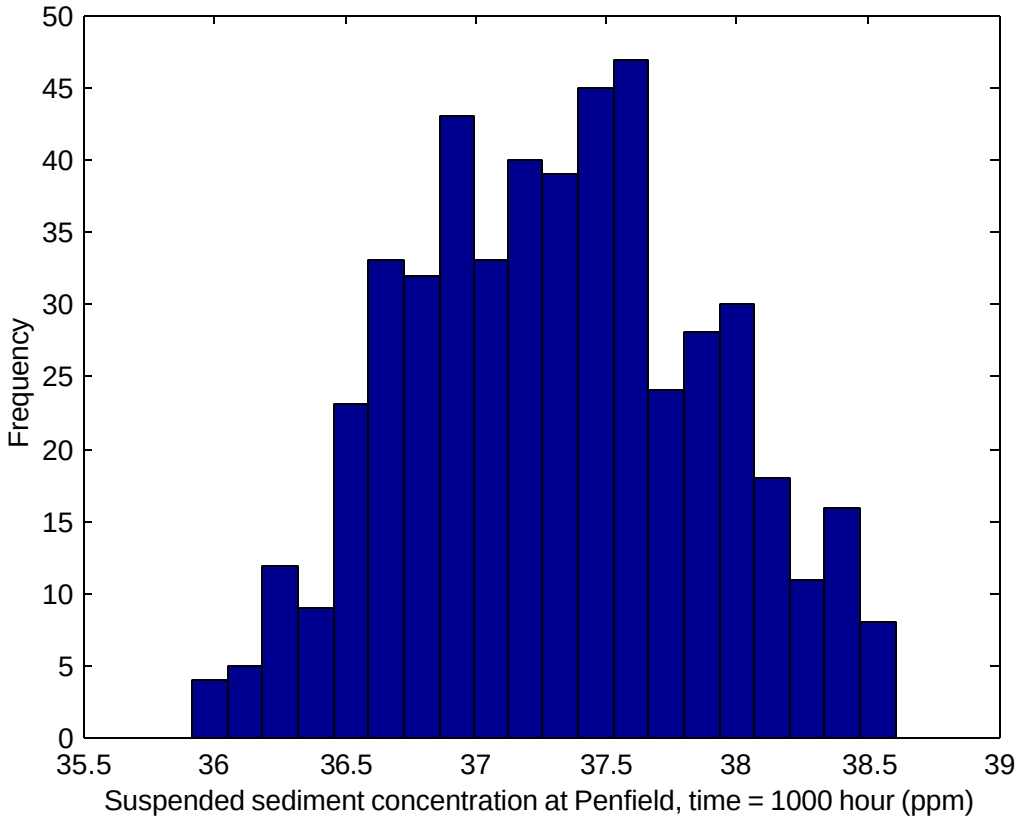


Figure 6-4. Histogram of suspended sediment concentration at Penfield when time = 1000 hour

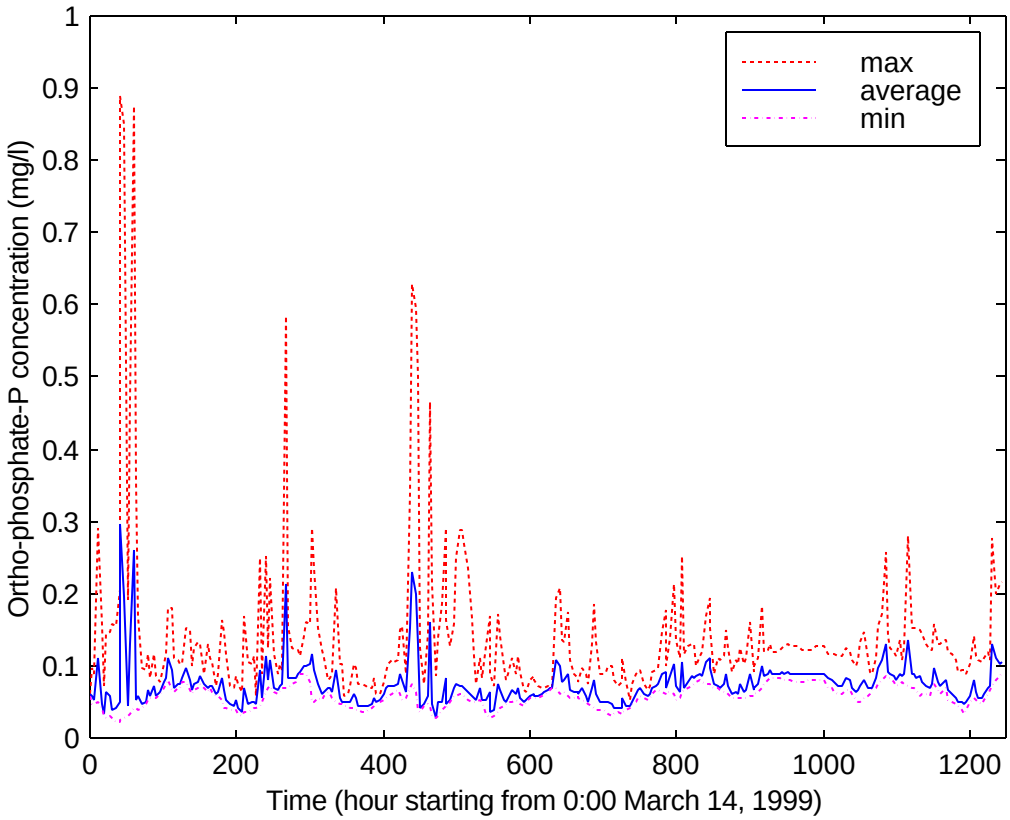


Figure 6-5. Simulation results of ortho-phosphate concentration at Penfield

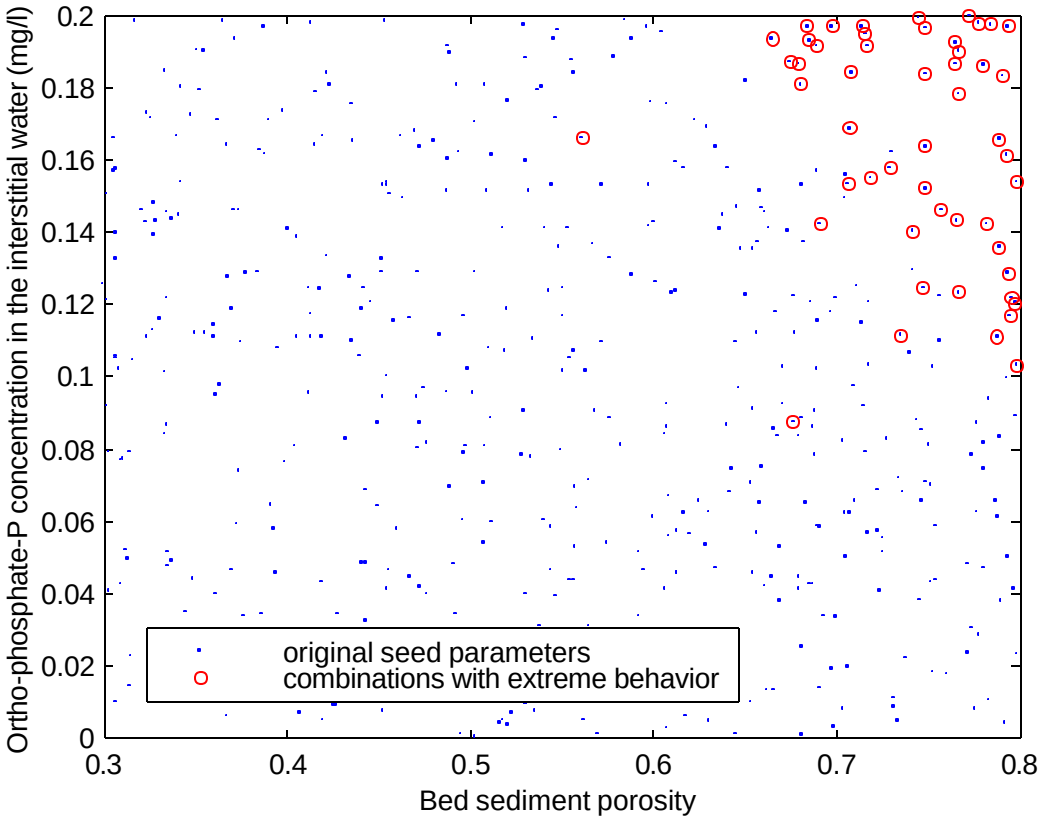


Figure 6-6. Parameter combinations that generate extreme simulation results

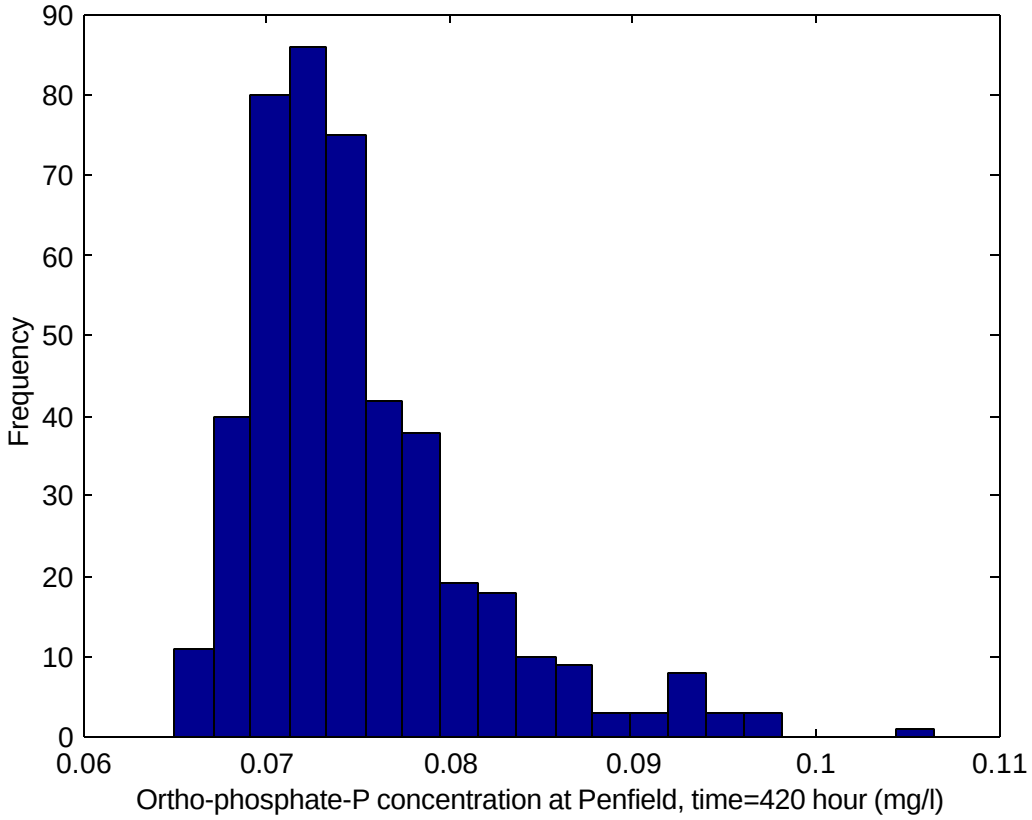


Figure 6-7. Histogram of ortho-phosphate concentration at Penfield when time = 420 hour

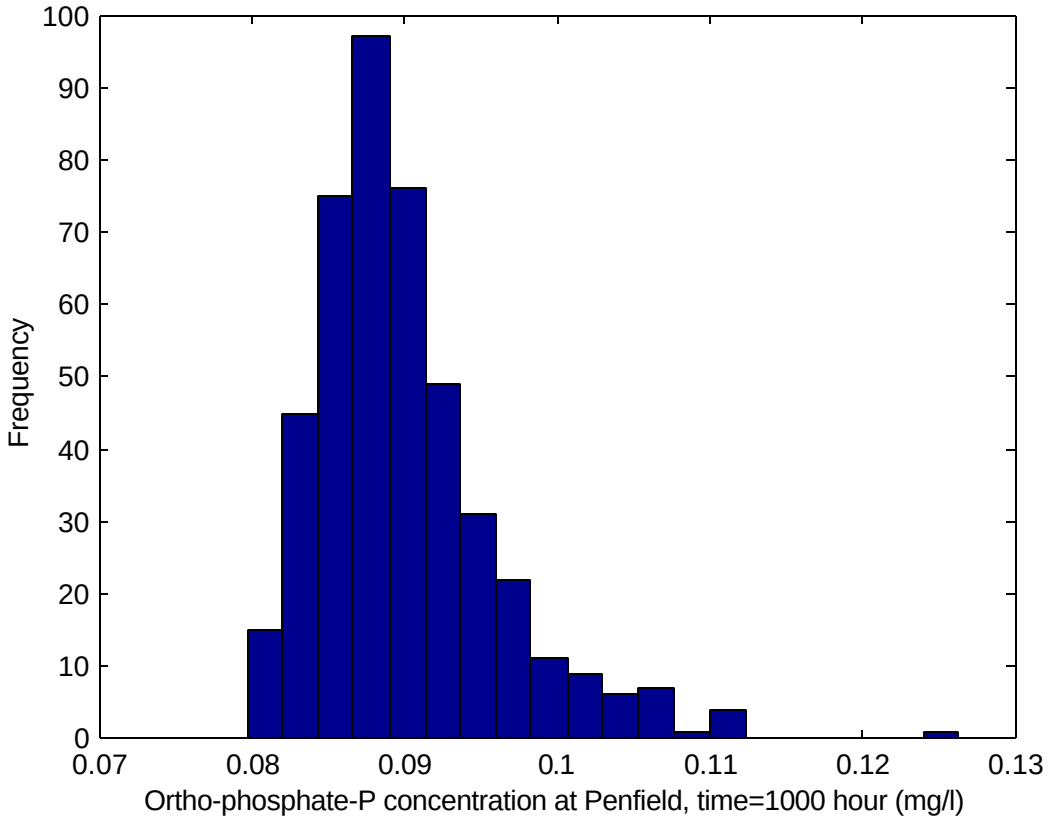


Figure 6-8. Histogram of ortho-phosphate concentration at Penfield when time = 1000 hour

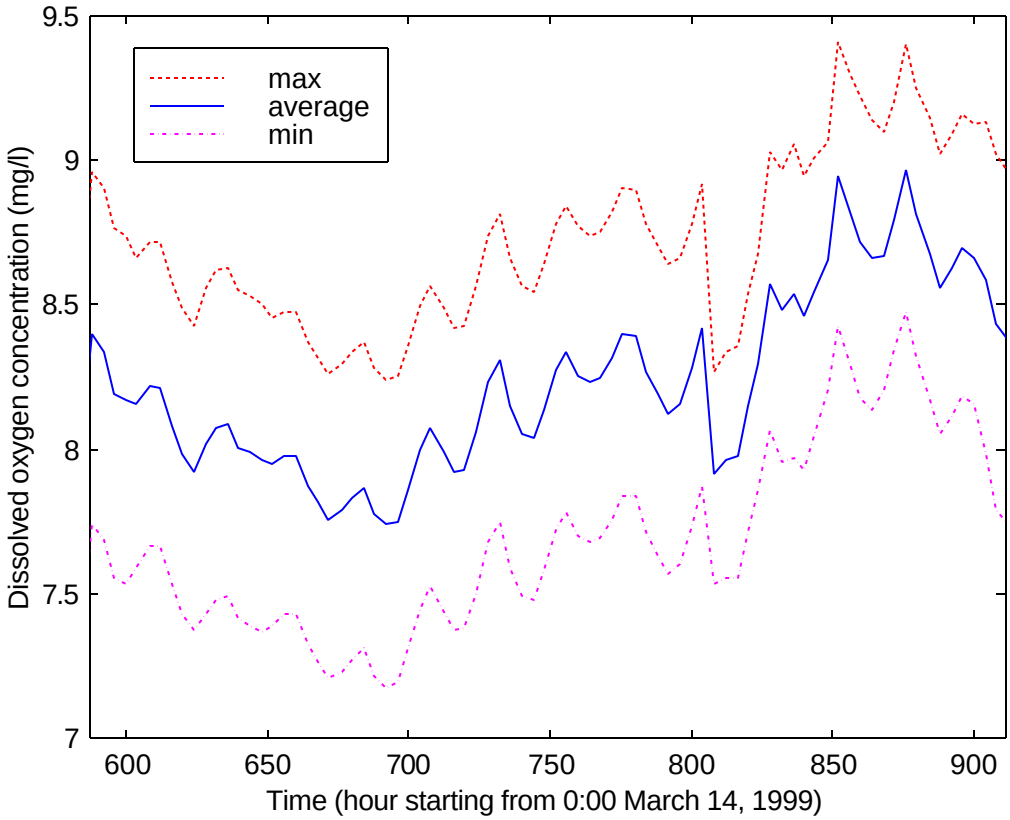


Figure 6-9. Simulation results of dissolved oxygen concentration at Penfield

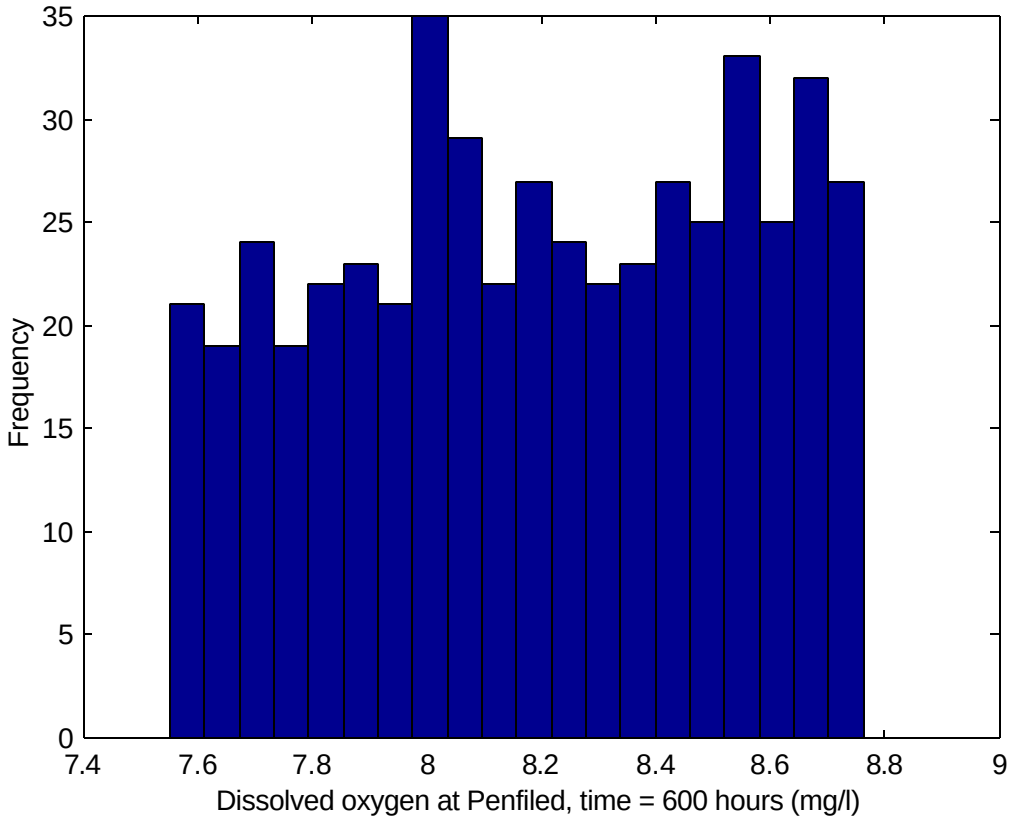


Figure 6-10. Histogram of dissolved oxygen concentration at Penfield when time = 600 hours

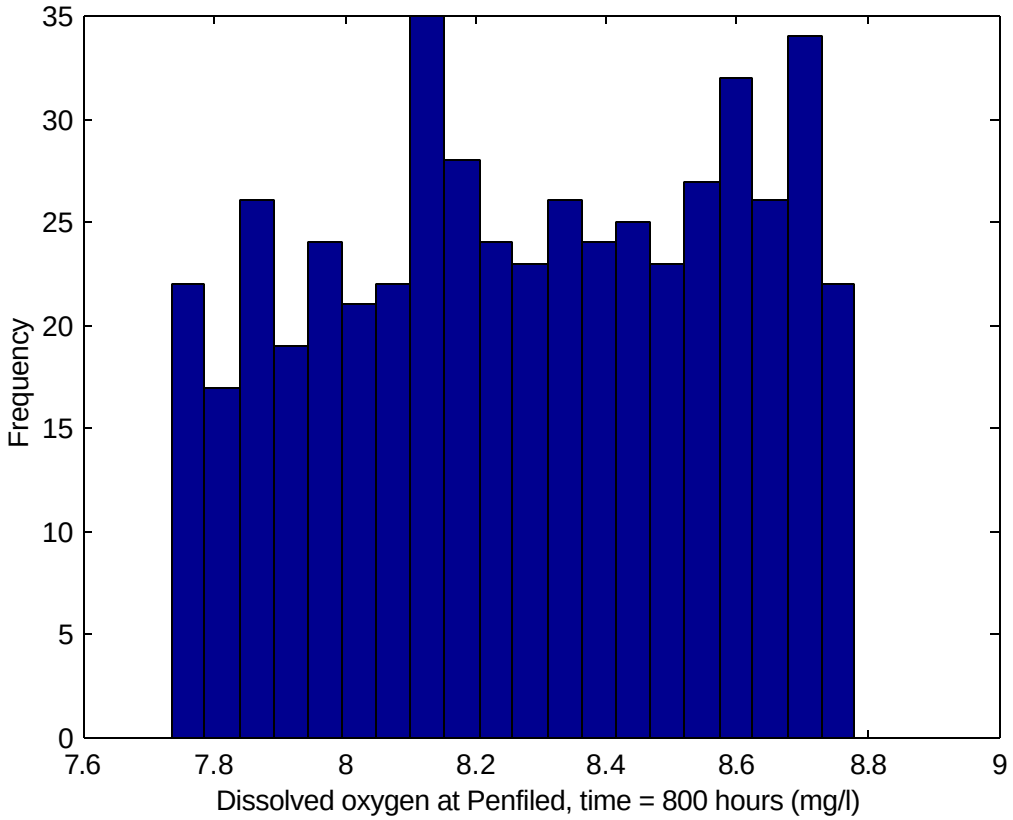


Figure 6-11. Histogram of dissolved oxygen concentration at Penfield when time = 800 hours

CHAPTER 7

APPLICATIONS OF STAND

In this chapter, having calibrated and evaluated STAND, it will be applied to a hypothetical river (referred to as "the River" hereafter) for the study of sediment transport and related water quality problems. The River resembles some characteristics of the reach between Buford Dam and Lake West Point on the Chattahoochee River, a major river in Georgia and the main source of water supply for metro Atlanta. The longitudinal profile of the River is chosen as that of the Chattahoochee River, although cross-section profiles were drawn arbitrarily. Input hydrological data were those posted by USGS for the station Chattahoochee River at Buford Dam near Buford (02334430). Interests and concerns include (1) sediment and associated nutrient transport under the influence of different hydrological conditions, (2) the proportion of sediment transported by peak flow and base flow at a certain station, (3) the differences in sediment transport and water quality between uniform dam release and intermittent hydropower station release, which is the current operational mode, and (4) the influence on downstream receiving water bodies of lateral inflow of sediments and nutrients.

7.1. The Influence Of Hydrological Conditions

The recorded dam releases over 11 years (1988 – 1998) were used as input to the model and the amount of water and sediment passing a particular station (Atlanta, 02334400) were computed. The Buford dam release data are provided by the USGS in the form of daily averaged values. The actual release does not follow this pattern.

Rather, there is an intensive release period which lasts several hours a day, and a minimum discharge of 600 cfs (about 17 m^3/s) is maintained for the rest of the day. All upstream inflow data were reconstructed according to this release pattern. The parameters obtained in the calibration and evaluation of the model using the Oconee River data were used in the simulations.

Figure 7-1 shows the results. A clear correlation can be seen between the amount of sediments and the amount of water passing the station. The relationship between the two quantities can be expressed by a simple linear equation, $\boldsymbol{S} = a_1\boldsymbol{W} + b_1$, where $\boldsymbol{S}$ is the annual amount of sediments passing the station, $\boldsymbol{W}$ is the annual amount of water passing the station, and a_1 and b_1 are the linear regression coefficients. For comparison, the linear trend was also plotted in the figure. The slope coefficient, a_1 is 8.255×10^{-4}, and the intercept coefficient, b_1 is -6.427×10^{5}.

There is also a clear correlation between the amount of sediment and of phosphate passing the station (Figure 7-2), and this too can be expressed by a simple linear equation, $\boldsymbol{P} = a_2\boldsymbol{S} + b_2$, where $\boldsymbol{P}$ is the annual amount of phosphate passing the station, $\boldsymbol{S}$ is the annual amount of sediments passing the station, a_2 and b_2 are the linear regression coefficients. The slope coefficient a_2 is computed to be 0.0010 and the intercept b_2 is -154.24.

Comparisons of the annual amount of water, sediments, and phosphate among the eleven years are shown in Figures 7-6 through 7-8, which also show the proportions of these quantities transported by peak and base flows.

These relationships provide rough estimates of the amounts of sediments and phosphate passing a given station along the studied reach, given the upstream discharge, a quantity that can usually be obtained from climatic conditions and the reservoir operations.

7.2. Peak Flow Vs. Base Flow

The proportion of water transported through the cross-section at Atlanta by peak flow is plotted in Figure 7-3. It is obvious that there is a clear monotonic relationship between the two quantities. This is understandable, because we assume that the Buford dam releases base flow at the rate of 600 cfs for most of the time except a five-hour period in which much more water is released. Thus, the more water released, the larger the proportion the amount of water carried by peak flow. It seems that the relationship can be quantified using an expression with exponential-terms (such as $y = y_0 - p_1 exp(-p_2 x)$, where x is the amount of water passing a station, y is the proportion of water carried by the peak flow, and y_0, p_1, and p_2 are the parameters to be determined). But the derivation is left for further studies.

Similarly, such relationships also exist in the case of sediments and phosphate. The proportion of sediments transported by peak flow vs. the annual amount of water passing Atlanta is shown in Figure 7-4. Again, the monotonic relationship between the two quantities is quite clear. Figure 7-5 shows a similar relationship between the proportion of phosphate transported by peak flow vs. the annual amount of water passing the station of Atlanta.

The author is confident that these relationships can be quantified using the computational results presented here and those that would be obtained by computations using data before the year 1988. Since the computations are very time consuming and beyond the scope of this dissertation, the quantification of these relationships would have to be left for further studies.

7.3. Comparison Between Uniform Dam Release And Intermittent Hydropower Station Release

It is interesting to see the difference between the effects of different upstream release patterns. The daily average discharge values at Buford dam of water year 1998 were used directly as input to the model. The results were compared with those of intermittent upstream release.

Figure 7-9 shows that the amount of water passing the station of Atlanta remains the same regardless of the upstream release pattern, which indicates that there is no retention effect caused by the change of release pattern. Figure 7-10 shows a significant decrease of sediment transported by the same amount of water, which implies that the pattern of release does affect the transport of sediment significantly. The water under the uniform release pattern only transported 63% of the sediment transported under the intermittent release pattern passing the station of Atlanta. The transport of phosphate is also affected by the release pattern of Buform dam. Figure 7-11 shows the difference between the amount of phosphate passing Atlanta under the two patterns. Under the uniform pattern, the amount of phosphate transported through the station of Atlanta is only 75% of the amount under the intermittent release pattern.

7.4. The Influences Of Lateral Inflow Of Sediments And Nutrients On Water Quality Of The Receiving Water Bodies

To study the influences of lateral inflow of sediments and nutrient on receiving water bodies, cases with different magnitudes of sediments and nutrient (ortho-phosphate) inflows are considered. The information regarding lateral inflow of water, sediments, and ortho-phosphate is listed in Table 7-1. Uniform lateral inflow of all the material is used in the study here, for the cause of simplicity. However, the model is

capable of having lateral inflows of material with both temporal and spatial variations. For the comparison of the effect of different magnitudes of ortho-phosphate in the lateral inflow on downstream water quality, cases 1 through 3 have different ortho-phosphate concentrations. For the comparison of the effects of lateral inflow of sediments, cases 3 through 5 have different lateral inflow sediment concentrations. The upstream input data of the water year 1998 (the first 1000 hours) are used in the simulations.

Table 7-1 Cases of lateral inflows of water, sediments, and ortho-phosphate

Cases of lateral inflows	Discharge of lateral inflow ($m^3/s/m$)	Sediment concentration in lateral inflow (g/m^3)	Ortho-phosphate concentration in lateral inflow (g/m^3)
case 1	5×10^{-5}	200.0	0.2
case 2	5×10^{-5}	200.0	0.5
case 3	5×10^{-5}	200.0	1.0
case 4	5×10^{-5}	1000.0	1.0
case 5	5×10^{-5}	2000.0	1.0

The simulation results are shown in Figures 7-12 through 7-14. For cases 1 through 3, the ortho-phosphate budget for the reach between Buford dam and Atlanta is shown in Figure 7-12. Three different magnitudes of ortho-phosphate concentrations in the lateral inflow are applied in the simulations. Two points can be seen from the computations. First, in each case, there appears to be a positive retention (from the dissolved phase) of ortho-phosphate in the reach. The ortho-phosphate that appears to be retained is the part that is adsorbed to the surfaces of the sediment particles and

transported with them towards downstream. Second, the outputs and net storage of ortho-phosphate in the reach are almost proportional to the amount of inputs into it. In each case, the amount of ortho-phosphate passing Atlanta is around 80% of the total inputs, while the amount retained (adsorbed to sediment particles) is about 20% of the total inputs, which is the sum of Buford dam release and the lateral inflows.

In cases 3 through 5, the ortho-phosphate concentrations in the lateral inflows remain the same, while the sediment concentrations in the lateral inflows range from 200 to 2000 mg/l. The sediment budgets of the simulations are shown in Figure 7-13. It is obvious that in each case, net entrainment happens in the studied reach. This is quite reasonable, since the sediment concentration in the released water from the Buford dam is only 20 mg/l, which is very much below the possible sediment transport potential, and as a result, entrainment happens. It can also be seen that the higher the amount of sediment that is put into the reach (either from the upstream end of the reach, or from the sides of the river as lateral inflow), the lower the amount of entrainment. This is also understandable, since the sediments getting into the reach help reduce the difference between the actual sediment concentration and the sediment transport potential, and thus reduce the amount of bed sediments that is needed in the process of entrainment. Figure 7-14 shows the ortho-phosphate budget in these cases. It appears that the differences of sediment concentration in the lateral inflow only have slight influences on the ortho-phosphate budget, and mainly when the sediment concentration in the lateral inflow is low. An explanation for this can probably be better presented by the analysis of Figures 7-15 through 7-17.

Figure 7-15 shows the discharge time series of a certain period in the simulation. The intermittent pattern of the discharge can be seen clearly. The corresponding suspended sediment concentration time series is presented in Figure 7-16, and the corresponding ortho-phosphate concentration in Figure 7-17. It can be seen that the changes (with cases, where the suspended sediment concentration in the lateral inflow

varies) of ortho-phosphate concentration only happens when the flow and the corresponding sediment concentration are low, except case 5, which will be discussed later. This is understandable, because what equation (3-41) says is that the more suspended sediment present in the water column, the lower the equilibrium ortho-phosphate concentration and vice versa. But when the suspended sediment concentration is so high and as a result the equilibrium ortho-phosphate concentration is so low, further increase of suspended sediment concentration can no longer have significant impact on the equilibrium concentration. This is why the curve of ortho-phosphate concentration in case 5 overlaps the curve in case 4. Since the change in ortho-phosphate concentration corresponding to the cases only happens with low flow, it is easy to understand that the impact on the final budget is not that significant.

Coming back to the exception mentioned above, the interesting shape of the curve representing suspended sediment concentration in case 5 will be discussed. When the sediment concentration in the lateral inflow is lower than a certain "threshold", where the addition of the lateral sediments does not cause the suspended sediment concentration in the stream to exceed the transport potential, the actual suspended sediment concentration corresponds well to the flow, in other words, it is flow-dominated. This is what occurs in cases 3 and 4. However, when the lateral inflow of sediments is high enough that the addition causes the suspended sediment concentration in the stream to exceed the transport potential, the increase of discharge no longer causes the increase of suspended sediments and the only effect it has is dilution. That is why we see a decrease of suspended sediment concentration corresponding to the moderately high discharges. Nevertheless, when the discharge is high enough to provide a higher sediment transport potential, an increase of suspended sediment concentration can be seen corresponding to the peak discharge (the peaks shown around time = 310, 340, and 360 hours, Figure 7-16).

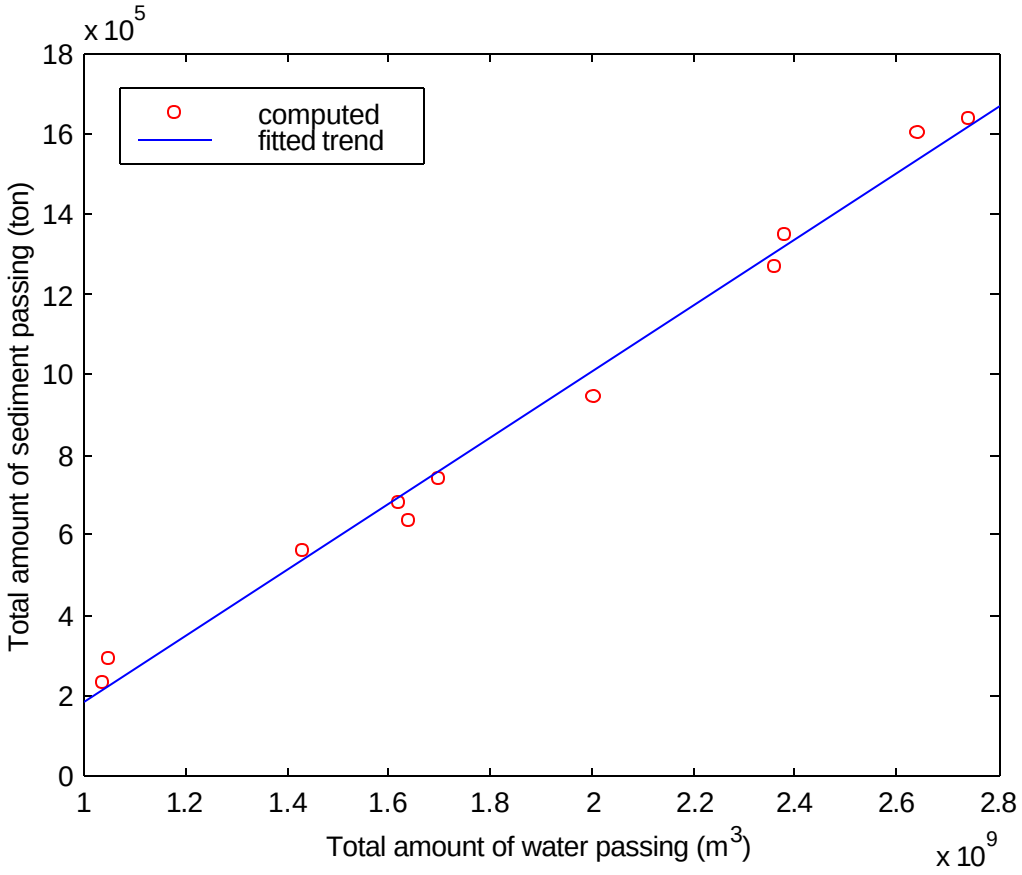


Figure 7-1. Annual amount of sediments vs. amount of water passing Atlanta

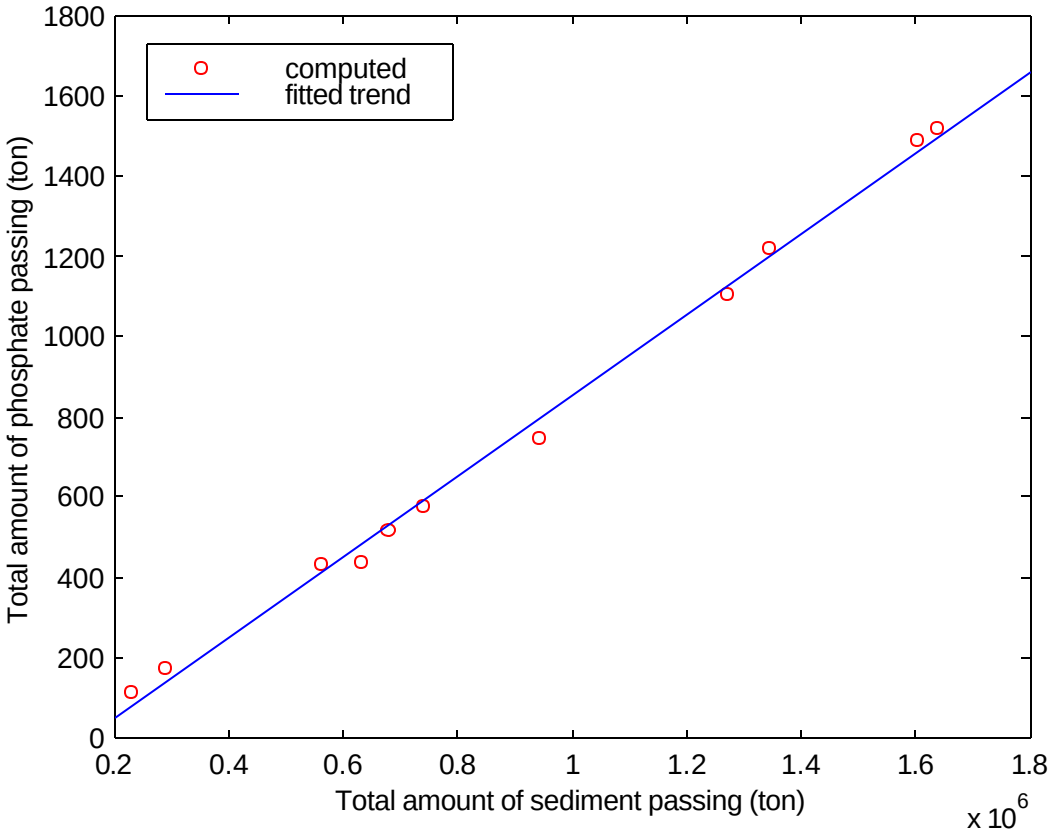


Figure 7-2. Annual amount of phosphate vs. amount of sediments passing Atlanta

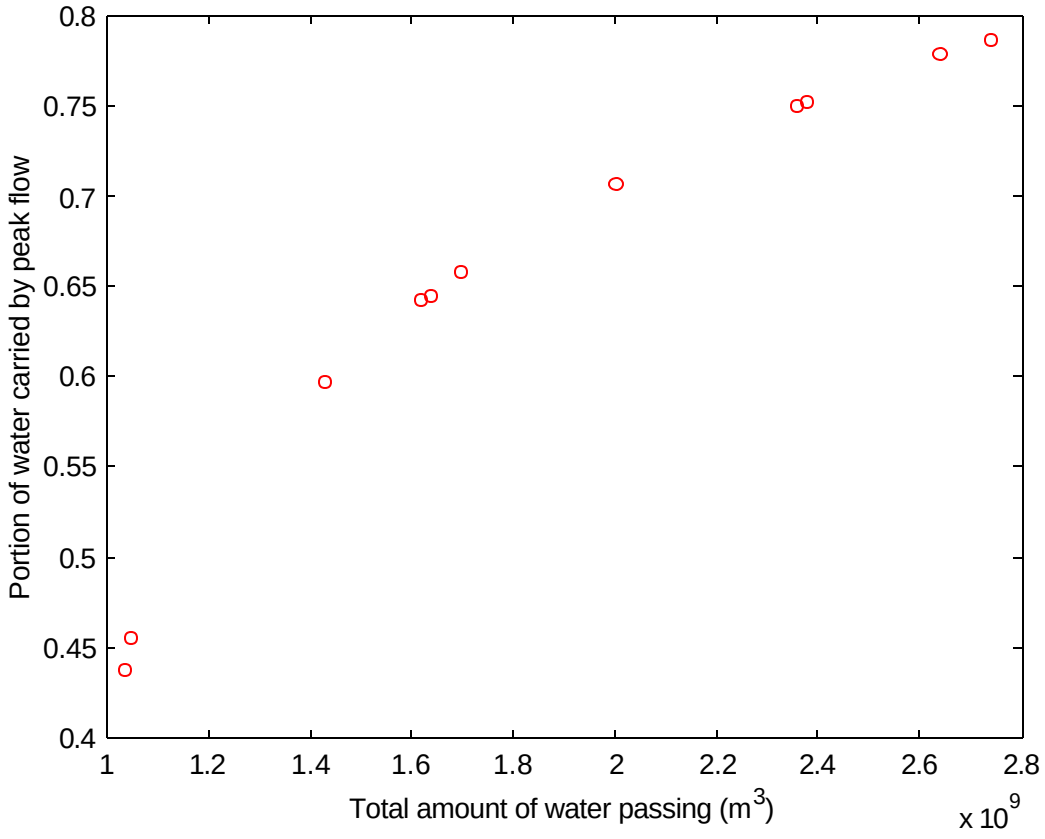


Figure 7-3. Proportion of water carried by peak flow

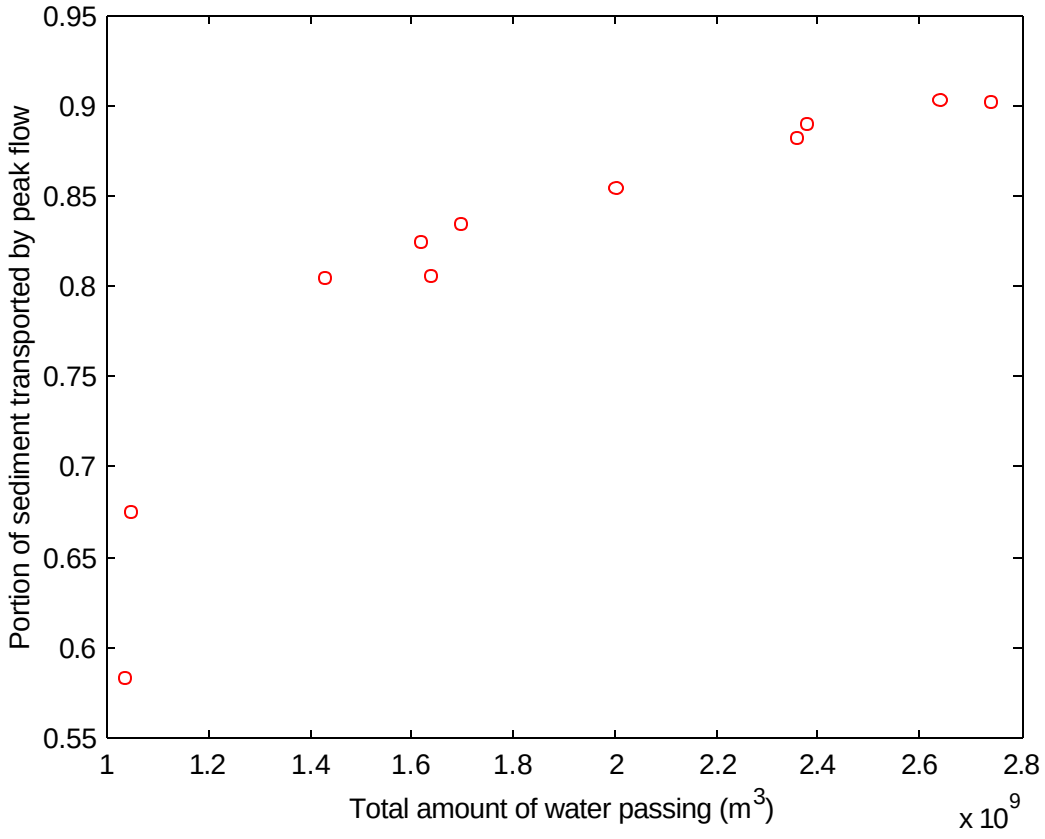


Figure 7-4. Proportion of sediments transported by peak flow

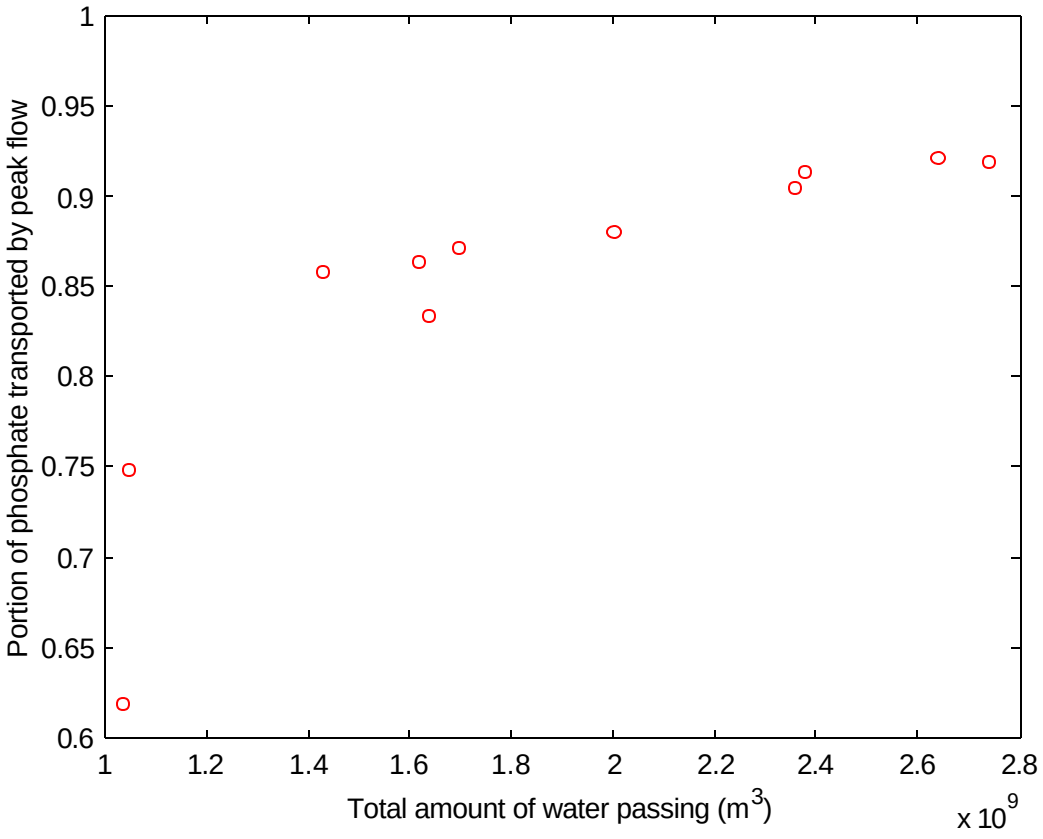


Figure 7-5. Proportion of phosphate transported by peak flow

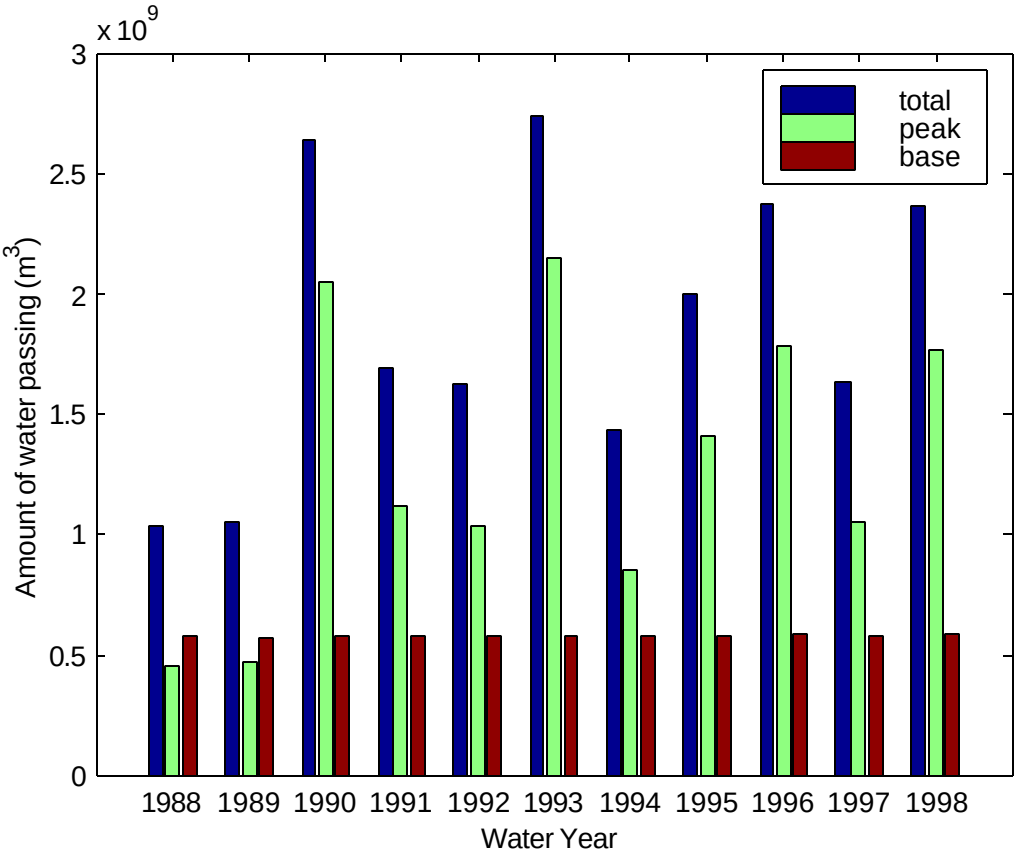


Figure 7-6. Amount of water passing Atlanta (02336000)

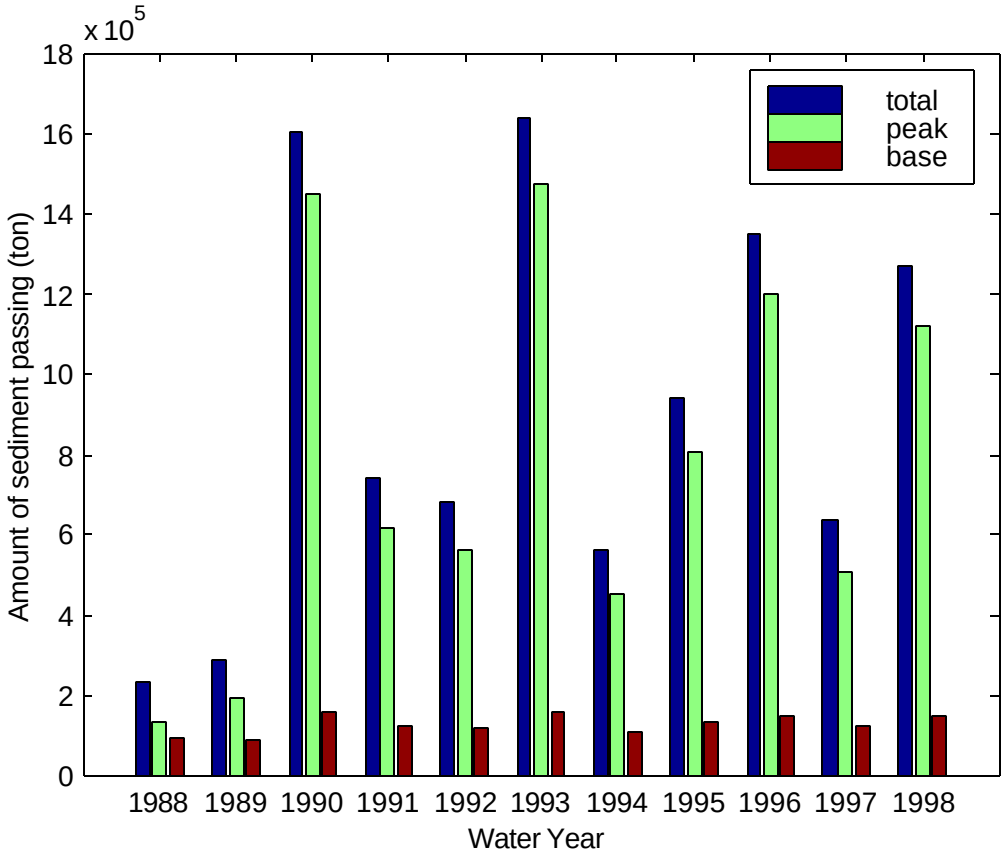


Figure 7-7. Amount of sediment passing Atlanta (02336000)

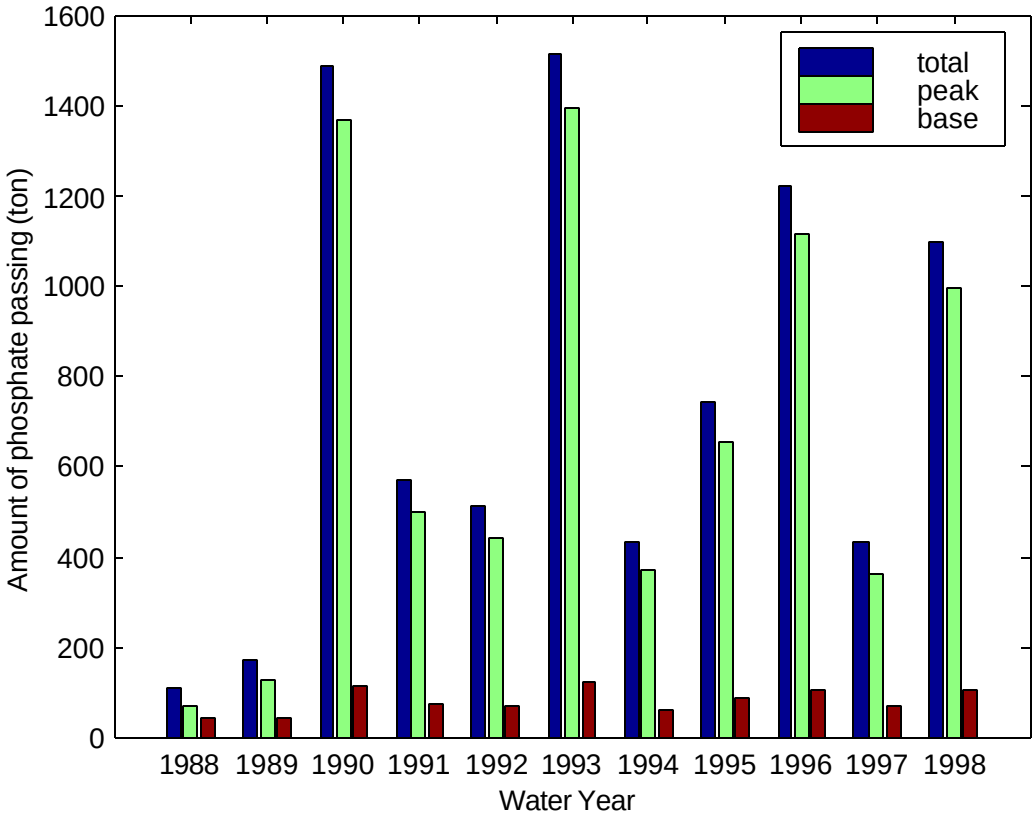


Figure 7-8. Amount of phosphate passing Atlanta

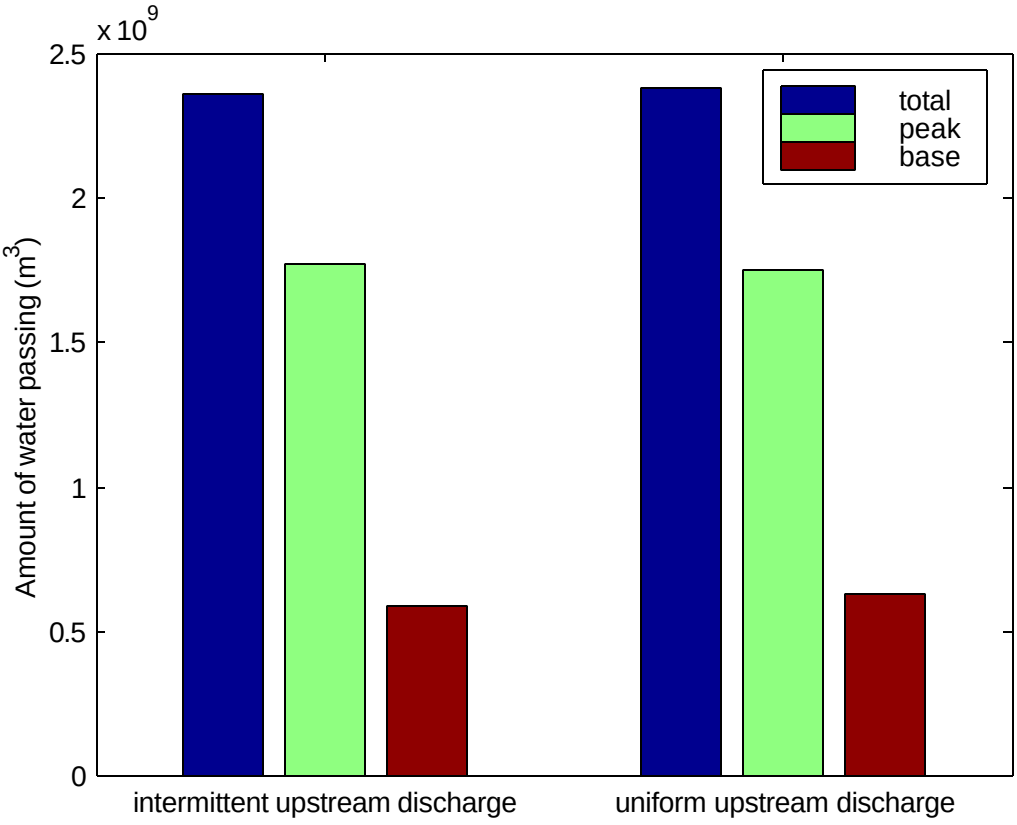


Figure 7-9. Amount of water passing Atlanta upon different upstream release patterns

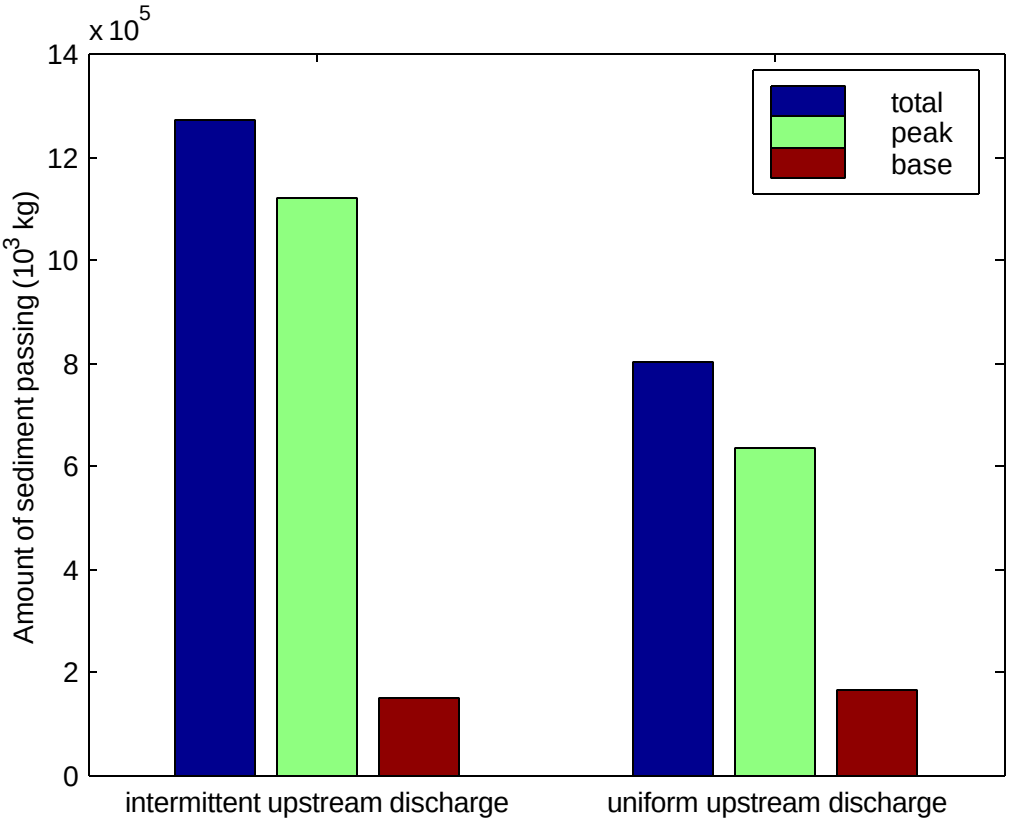


Figure 7-10. Amount of sediment passing Atlanta upon different upstream release patterns

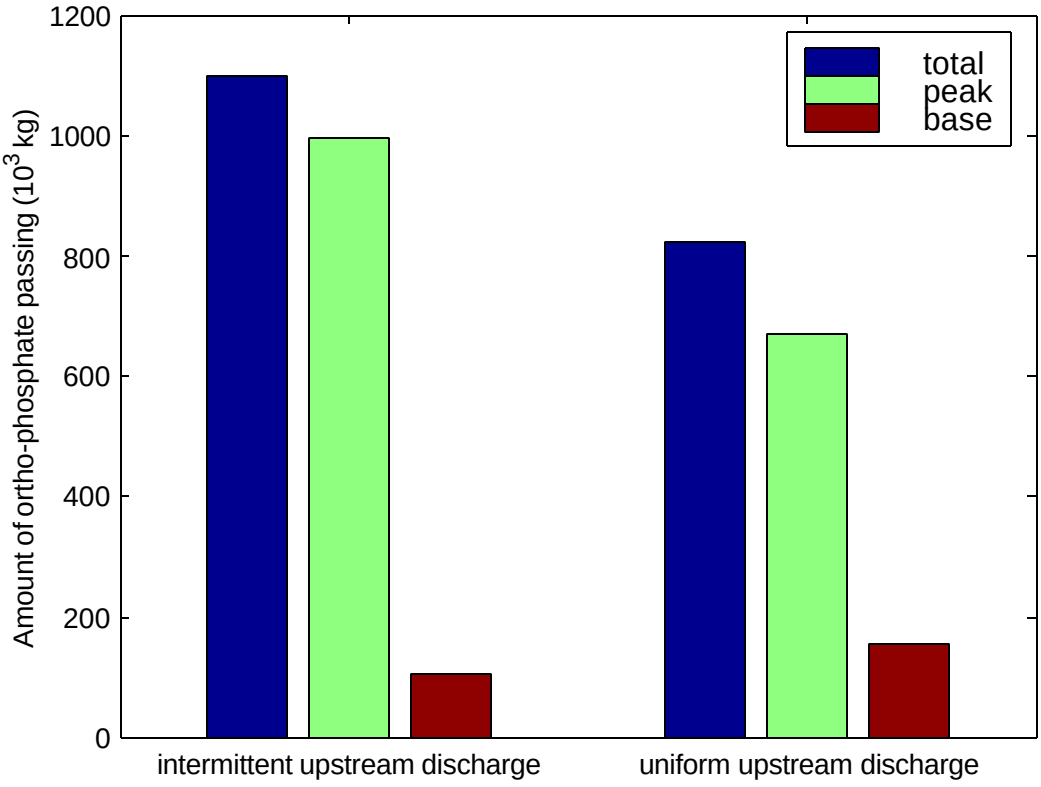


Figure 7-11. Amount of phosphate passing Atlanta upon different upstream release patterns

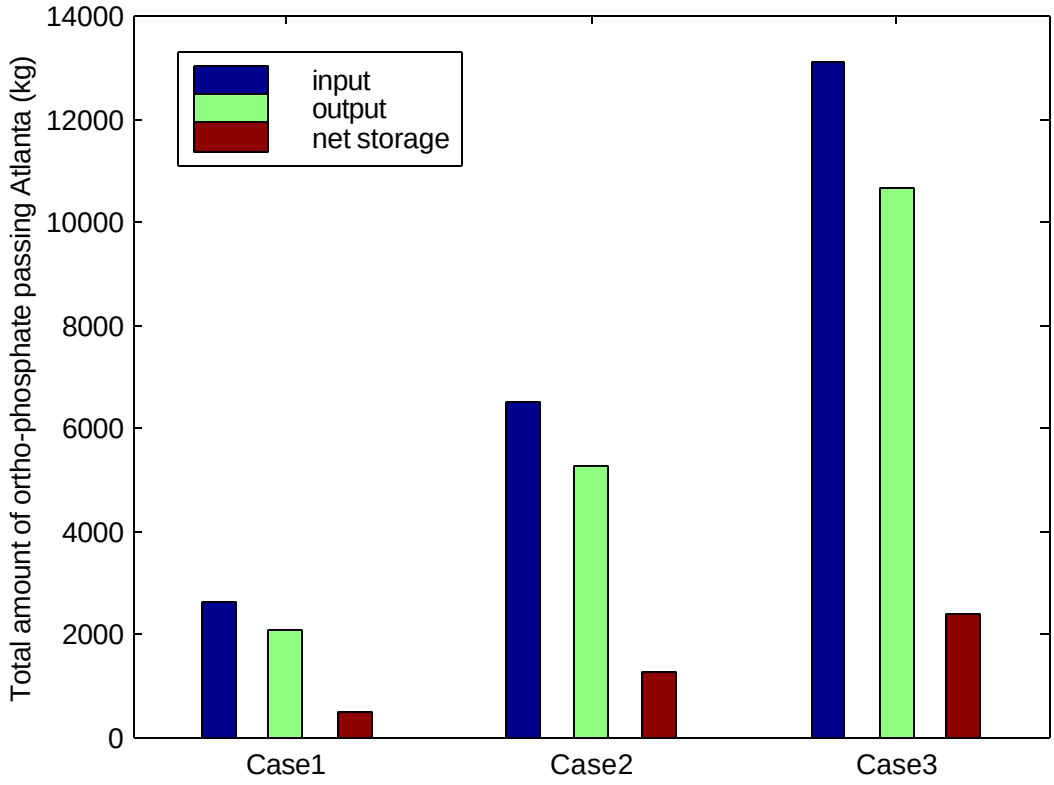


Figure 7-12. Ortho-phosphate budget for the studied reach (between Buford dam and Atlanta) under the influence of lateral inflow of ortho-phosphate

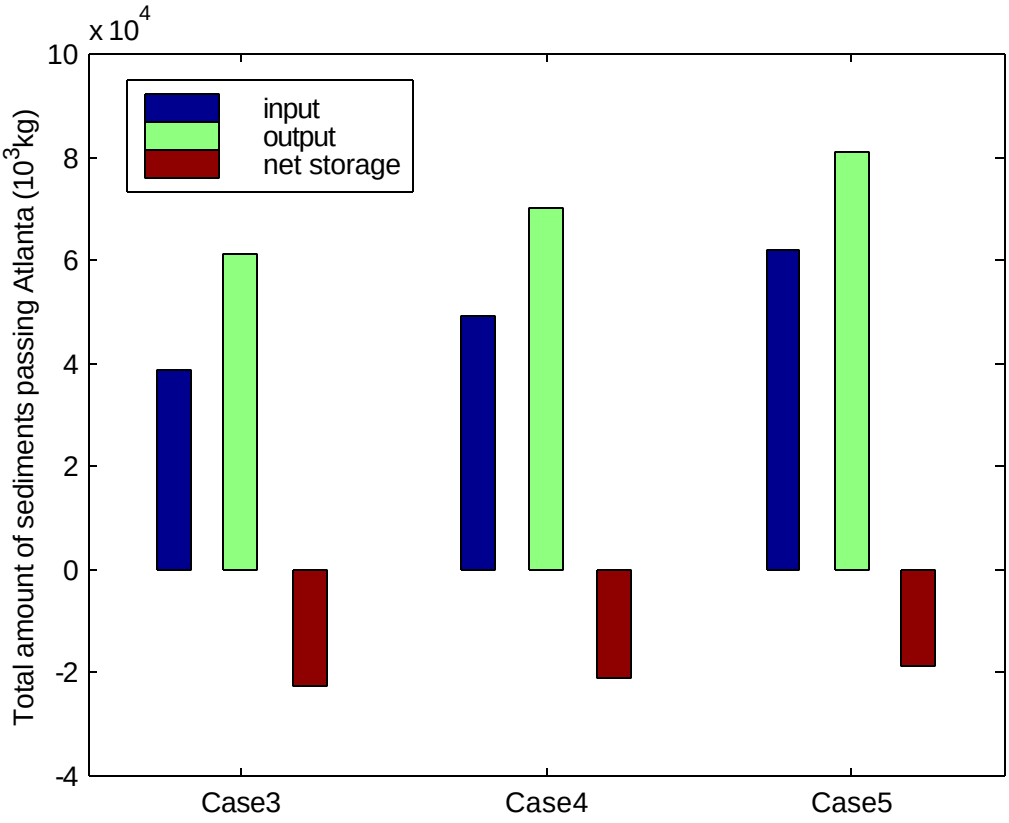


Figure 7-13. Sediment budget for the studied reach (between Buford dam and Atlanta) under the influence of lateral inflow of sediments

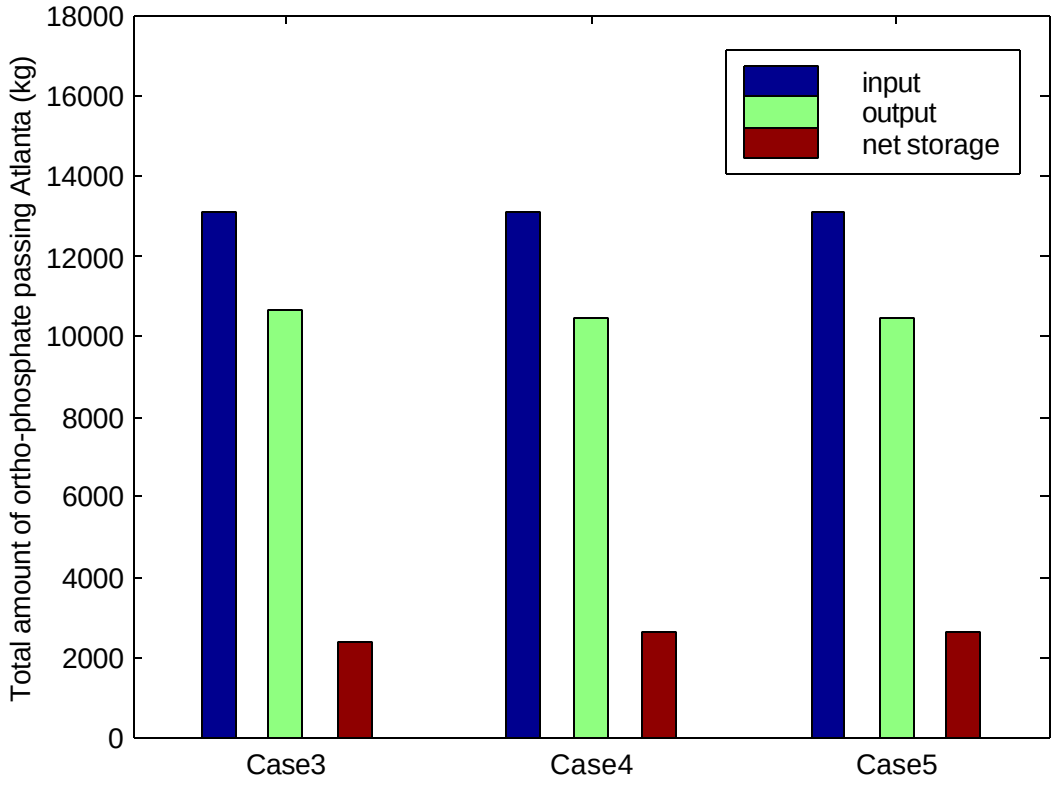


Figure 7-14. Ortho-phosphate budget for the studied reach (between Buford dam and Atlanta) under the influence of lateral inflow of sediments

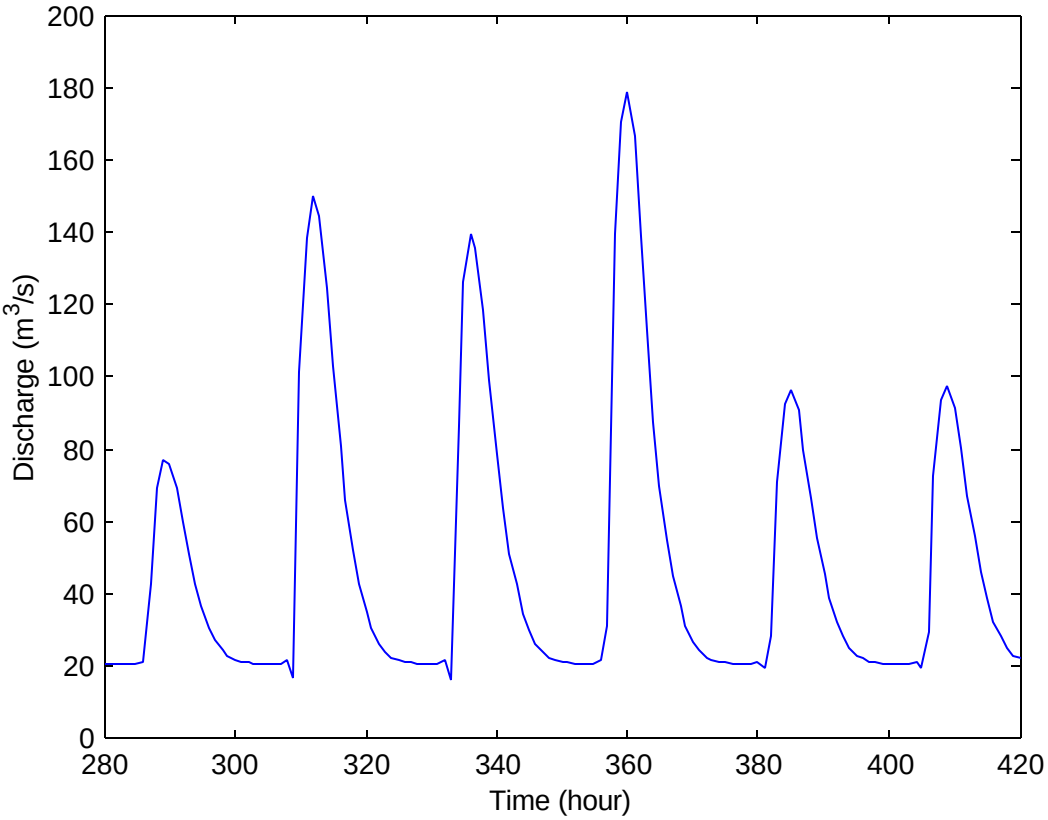


Figure 7-15. Discharge time series at Atlanta in a certain time period

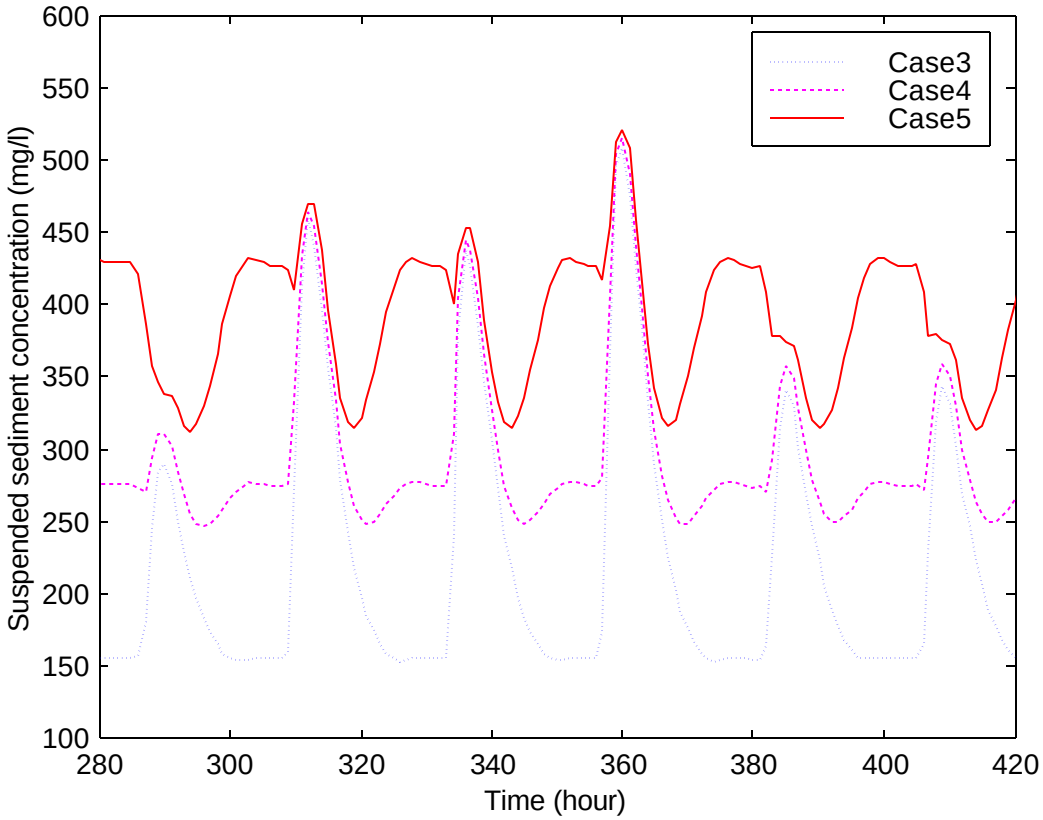


Figure 7-16. Suspended sediment concentration at Atlanta in the corresponding time period (Figure 7-15)

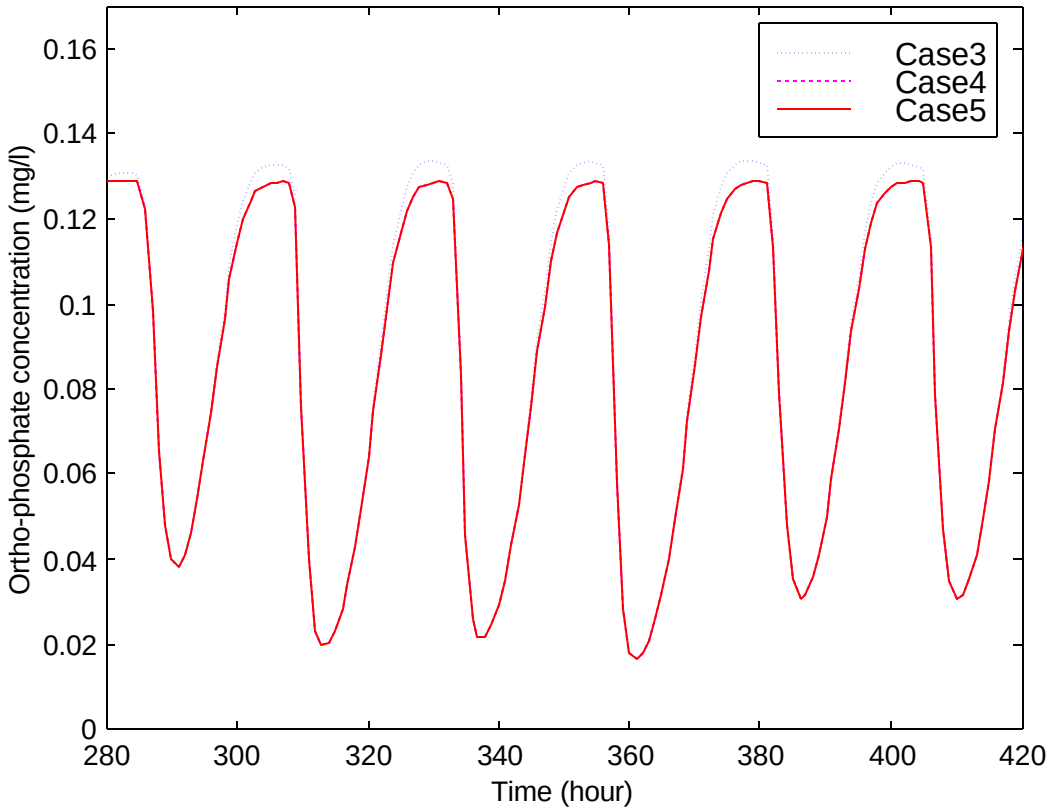


Figure 7-17. Ortho-phosphate concentration at Atlanta in the corresponding time period (Figure 7-15)

CHAPTER 8

CONCLUSIONS

The previous chapters show why and how STAND is developed, how data were collected to test the model, how it is calibrated, and how it can be used in the study of sediment-related water quality problems. The results in Chapter 5 clearly indicate successful calibrations and evaluations with regard to components of hydraulics, sediment transport, and most of the water quality constituents. STAND steps beyond traditional civil-engineering models by providing the ability to deal with water quality issues. It also provides a unique capacity that most existing water quality models do not have by taking into consideration the effects of sediment behavior on water quality constituents. Such a comprehensive tool has not been reported in the literature and is certainly useful in the study of water quality issues associated with sediment behaviors. However, there is still room for further improvement.

Currently, STAND has difficulties simulating the channel morphological changes on the Yellow River of China. This is probably caused by the high sediment concentration and resulting high rate of scouring and deposition. It could also be the result of numerical supercritical flow – a situation caused by an improper value of the Manning coefficient under shallow water conditions. When the model is applied to the Oconee River and the Chattahoochee River, the morphology component works well. Arbitrarily shutting down the morphology component during shallow water periods can possibly solve the problem, but the definition of a shallow water condition is somewhat subjective and hard to make.

In computing the suspended sediment concentration, Yang's total load equation is used in the model. This is because the equation provides straightforwardness and is easy to code. Other suspended sediment transport functions (Lane and Kalinske 1941, Einstein 1950, Brooks 1963, Chang et al. 1965) can also be added into the model. It is strongly suggested that simple equations be used, because some equations with complexity require many iterations and a significant amount of time for computation. Bed load equations (Chang et al. 1965, Einstein 1942, Einstein 1950, Kalinske 1947, Meyer-Peter, and Muller 1948, Parker 1990) can also be added to the model, although the computation will not affect the subsequent computation of water quality constituents.

Currently, the model takes d_{50} as the representative dimension for the entire sediment. The settling velocity is computed using this representative dimension. In many other civil-engineering-models, sediment behavior can be computed by size fractions. This feature could be added to the model, but since there were no available data to calibrate this particular feature, it was not incorporated in STAND.

The sediment-related water quality model can also be used to address water quality problems caused by heavy metal or other pollutants, if components are added. Adding other components is not much more difficult than electronically copying a block of file and pasting it somewhere else, for someone who knows the mechanism of the codes. The already made codes can mostly be reused to describe other water quality constituents. The only thing that needs to be changed is the source/sink term, because different water quality constituents have different generation/consumption mechanisms.

An ideal description of the water quality mechanisms of a river section would include the exchanges of water quality constituents between surface water and suspended sediment, the exchange of sediments between the bed and the overlying water column, the exchange of water quality constituents between bed sediments and the interstitial water, and the effects of possible seepage. In STAND, the interstitial water component is considered external to the closed system composed of open flow, suspended sediments,

and bed sediments. As a result, the concentrations of ammonia and phosphate in the interstitial water are considered constant, and should be given by the user as a parameter. This simplification was made because of the fact that we lack the information of the quality constituents of the interstitial water and bed sediments. To incorporate the dynamic interstitial water into the model, we need to have data regarding subsurface flow and water constituents of the subsurface flow, along the whole studied reach.

Finally, the adsorption-desorption process was based on the Freundlich equation, which is an empirical equation. An ideal approach to improve and strengthen this part of the model is to conduct laboratory experiments regarding the adsorption isotherm of phosphate (and possibly ammonium) on sediments. Special attentions should be paid to the relationship between suspended sediment concentrations and the amount of dissolved matters adsorbed to the particles. The relationship used in the model was based on mass balance of the dissolved matters and has sound theoretical basis. But it would be further strengthened if it can be backed by the suggested experiments.

REFERENCES

Alaoui-Mhamdi, M., and L. Aleya (1995), "Assessment of the Eutrophication of Al Massira Reservoir (Morocco) by Means of a Survey of the Biogeochemical Balance of Phosphate", Hydrobiologia, Vol 297, 75-82

Appan, A., and Ting, D.S. (1996), "A laboratory study of sediment phosphorus flux in two tropical reservoirs", Water Science and Technology, Vol. 34, No.7-8, pp. 45-52.

Ambrose, R.B., Wool, T.A., and Martin, J.L. (1993), The Water Quality Analysis Simulation Program, WASP5, Part A: Model Documentation, Environmental Research Laboratory, United States Environmental Protection Agency.

Ambrose, R.B., Wool, T.A., and Martin, J.L. (1993), The Dynamic Estuary Model Hydrodynamics Program, DYNHYD5, Model Documentation and User Manual, Environmental Research Laboratory, United States Environmental Protection Agency.

Arakel, A.V. (1995), "Towards developing sediment quality assessment guidelines for aquatic systems – an Australian perspective", Australian Journal of Earth Sciences, Vol.42, No.4.

Bathurst, J.C. and Purnama, A. (1991), "Design and application of a sediment and contaminant transport modelling system", Sediment and stream water quality in a

changing environment: trends and explanation (Proceedings of the Vienna Symposium, August 1991) IAHS Publication no.203, pp.305-313.

Beck, M.B. (1985), “Lake eutrophication: identification of tributary nutrient loading and sediment resuspension dynamics”, Applied mathematics and computation, Vol.17, pp.433-458.

Beck, M.B. (1987), “Water quality modeling: a review of the analysis of uncertainty”, Water Resources Research, Vol.23, No.8, pp.1393-1442.

Beck, M.B. and R. Liu (1998), “On-line analyzers and modeling in wastewater treatment plant control”, Preprint, Pre-Conference Workshop on Online Analyzers Based Control of Wastewater Treatment Plants, Water Environment Federation, 71st Annual Conference and Exposition, Orlando, Florida, (1998), 11p.

Beddig, S., Brockmann, U., Dannecker, W., Korner, D., Pohlmann, T., Puls, W., Radach, G., Rebers, A., Rick, H.J., Schatzmann, M., Schlunzen, H., Schulz, M. (1997), “Nitrogen fluxes in the German Bight”, Marine Pollution Bulletin, Vol. 34, No. 6, pp.382-394.

Bertuzzi, A., Faganeli, J., and Brambati, A. (1995), “Annual variation of benthic nutrient fluxes in shallow coastal waters (Gulf of Trieste, Northern Adriatic Sea)”, Marine Ecology, Vol.17, pp.261-278.

Blom, G. and Toet, C. (1993), “Modeling sediment transport and sediment quality in a shallow Dutch lake (Lake Ketek)”, Water Science and Technology, Vol.28, No.8-9, pp.79-90.

Blom, G. and Winkels, H.J. (1998), "Modelling sediment accumulation and dispersion of contaminants in lake Ijsselmeer (the Netherlands)", Water Science and Technology, Vol.37, No.6-7, pp.17-24.

Boers, P. and de Bles, F. (1991), "Ion concentrations in interstitial water as indicators for phosphorus release processes and reactions", Water Research, Vol.25, No.5, pp.591-598.

Bowie, George L., William B. Mills, Donald B. Porcella, Carrie L. Campbell, James R. Pagenkopf, Gretchen L. Rupp, Kay M. Johnson, Peter W.H. Chan, Steven A. Gherini and Charles E. Chamberlin (1985), Rates, Constants, and Kinetics Formulations in Surface Water Quality Modeling, Environmental Research Laboratory, Office of Research and Development, USEPA

Brassard, P., Kramer, J.R., and Collins, P.V. (1997), "Dissolved metal concentrations and suspended sediment in Hamilton Harbour", Journal of Great Lakes Research, Vol. 23, No. 1, pp.86-96

Brooks, N.H. (1963), "Calculation of suspended load discharge from velocity concentration parameters", Proceedings of Federal Interagency Sedimentation Conference, U.S. Department of Agriculture, Miscellaneous Publication no.970.

Brouwers, H.J.H. (1999), "Transport models for desorption from natural soils packed in flushed columns", Water Resources Research, Vol.35, No.6, pp.1771-1780.

Brown, L.C. (1987), "Uncertainty analysis in water quality modeling using QUAL2E", in Beck, M.B. (ed.) System Analysis in Water Quality Management, pp.309-319.

Brown, L.C. and Barnwell, T.O. (1987), The Enhanced Stream Water Quality Models QUAL2E and QUAL2E-UNCAS: Documentation and user manual, Environmental Research Laboratory, Office of Research and Development, USEPA

Caraco, N.F., Cole, J.J., and Likens, G.E. (1989), "Evidence for sulphate-controlled phosphorus release fro sediments of aquatic systems", Nature, Vol.341, No.6240, pp.316-318.

Caraco, N.F., Cole, J.J., and Likens, G.E. (1991), Chapter 12. A Cross-System study of phosphorus resease from lake sediments. pp.241-258. In J.J.Cole, G. Lovett, and S. Findlay (eds.). Comparative Analysis of Ecosystems: patterns, mechanisms, and theories. Springer-Verlag, NY.

Caraco, N.F., Cole, J.J., and Likens, G.E. (1993), "Sulfate control of phosphorus availability in lakes", Hydrobiologia, Vol.253, pp.275-280.

Carman, R., Aigars, J., and Larsen, B. (1996), "Carbon and nutrient geochemistry of the surface sediments of the Gulf of Riga, Baltic Sea", Marine Geology, Vol. 134, pp.57-76.

Carter, R.F., Hopkins, E.H., and Perlman, H.A. (1988), Low-Flow Profiles of the Upper Oconee River and Tributaries in Georgia, Department of Interior, U.S. Geological Survey.

Cerco, C.F. and Cole, T. (1993), "3-Dimensional eutrophication model of Chesapeake bay", Journal of Environmental Engineering-ASCE, Vol.119, No.6, pp.1006-1025.

Cerco, C.F. and Seitzinger, S.P. (1997), “Measured and Modeled Effects of Benthic Algea on Eutrophication I Indian River-Rehoboth Bay, Delaware”, Estuaries, Vol. 20, No.1, 231

Chambers, P.A. and Prepas, E.E. (1994), “Nutrient dynamics in riverbeds – the impact of sewage effluent and aquatic macrophytes”, Water Research, Vol.28, No.2, pp.453-464.

Chang, F.M., Simons, D.B., and Richardson, E.V. (1965), “Total bed-material discharge in alluvial channels”, U.S. Geological Survey Water-Supply Paper 1498-I.

Chang, H.H. (1982), “Mathematical Model for Erodible Channels”, Journal of the Hydraulics Division, ASCE, 108(HY5), 678-689

Chang, H.H. (1984), “Modeling of River Channel Changes”, Journal of the Hydraulics Division, ASCE, 110(2), 157-172

Chang, H.H. (1988), Fluvial Processes in River Engineering, John Wiley & Sons, Inc.

Chang, H.H. and Stow, D. (1989), “Mathematical Modeling of Fluvial Sediment Delivery”, Journal of Waterway, Port, Coastal, and Ocean Engineering, ASCE, 115(3), 311-326

Chang, H.H. (1993), Generalized Computer Program FLUVIAL-12 Mathematical Model for Erodible Channels, Users Manual

Cheney, W. and Kincaid, D. (1994), Numerical Mathematics and Computing, Brooks/Cole Publishing Company

Coghlan, B.P., Ashley, R.M., and Jefferies, C. (1993), "An appraisal of suspended sediment transport modeling methods for an interceptor sewer", Water Science and Technology, Vol.27, No.5-6, pp.81-91.

Correll, D.L., Jordan, T.E., and Weller, D.E. (1999), "Transport of nitrogen and phosphorus from Rhode River watersheds during storm events", Water Resources Research, Vol.35, No.8, pp.2513-2521.

Cowan, J.L.W., and Boynton, W.R. (1996), "Sediment-Water Oxygen and Nutrient Exchanges Along the Longitudinal Axis of Chesapeake Bay: Seasonal Patterns, Controlling Factors and Ecological Significance", Estuaries, Vol.19, No.3, 562

Crabtree, R.W., Garsdal, H., Geng, R., Mark, O., and Dorge, J. (1994), "Mousetrap – a deterministic sewer flow quality model", Water Science and Technology, Vol.30, No.1, pp.107-115.

Crabtree, R.W., Ashley, P., and Gent, R. (1995), "Mousetrap – modeling of real sewer sediment characteristics and attached pollutants", Water Science and Technology, Vol.31, No.7, pp.43-50.

Craft, C.B. and Casey, W.P. (2000), "Sediment and nutrient accumulation in floodplain and depressional freshwater wetlands of Georgia, USA", Wetlands, Vol.20, No.2, pp.323-332.

Croke, J., Hairsine, P., and Fogarty, P. (2000), "Nutrient movement due to overland flow in managed native Eucalyptus forests, southeastern Australia", Water Air And Soil Pollution, Vol.122, No.1-2, pp.17-35.

Dezzeo, N., Herrera, R., Escalante, G., and Chacon, N. (2000), "Deposition of sediments during a flood event on seasonally flooded forests of the lower Orinoco River and two of its black-water tributaries, Venezuela", Biogeochemistry, Vol.49, No.3, pp.241-257.

Duchesne, S., Beck, M.B., and Reda, A.L.L. (2000), "Ranking stormwater control strategies under uncertainty: the River Cam case study", Symposium Preprints of WATERMATEX 2000, Proceedings of the 5th International Symposium on Systems Analysis and Computing in Water Quality Management, pp.6.17-6.24.

Eintein, H.A. (1942), "Formula for the transportation of bed-load", Transactions of the ASCE, Vol.107.

Einstein, H.A. (1950), "The Bed-Load Function for Sediment Transportation in Open Channel Flows", Technical Bulletin No. 1026, United States Department of Agriculture

Emmerth, P.P. and Bayne, D.R. (1996), "Urban Influence on Phosphorus and Sediment Loading of West Point Lake, Georgia", Water Resources Bulletin, Vol. 32, No. 1, 145-154.

Engelhardt, C., Prochnow, D., and Bungartz, H. (1995), "Modeling and simulation of sedimentation processes in a lowland river", Mathematics and Computers In Simulation, Vol.39, No.5-6, pp.627-633.

Evans, J.W. (1991), A Fisheries and Recreational Use Survey of the Upper Ocmulgee River, Georgia Department of Natural Resources, Game and Fish Division, Final Report, Federal Aid Project F-33.

Evans, J.W. (1994), A Fisheries Survey of the Oconee River between Sinclair Dam and Dublin, Georgia, Georgia Department of Natural Resources, Wildlife Resources Division.

Ferreira, M.F., Chiu, W.S., Cheok, H.K., Cheang, F., and Sun W. (1996), "Accumulation of nutrients and heavy metals in surface sediment near Macao", Marine Pollution Bulletin, Vol. 32, No. 5, pp.420-425

French, R.H. (1985), Open-Channel Hydraulics, McGraw-Hill

Froelich, P.N. (1988), "Kinetic control of dissolved phosphate in natural rivers and estuaries: A primer on the phosphate buffer mechanism", Limnology and Oceanography, Vol.33, No.4, pp.649-668.

Galindo, E.A., Celaya, J.A., Munoz, G.F., and Sericano, J.L. (1996), "Organic contaminants in sediments from San Qaintin Bay, Baja California, Mexico", Marine Pollution Bulletin, Vol. 32, No. 4, pp.378-381

Garban, B., Ollivon, D., Poulin, M., Gaultier, V., Chesterikoff, A. (1995), "Exchange at the Sediment-Water Interface in the River Seine, Downstream from Paris", Water Research, Vol.29, No.2, 473-481.

Garsdal, H., Mark, O., Dorge, J., and Jepsen, S.E. (1995), "Mousetrap – modeling of water-quality processes and the interaction of sediments and pollutants in sewers", Water Science and Technology, Vol.31, No.7, pp.33-41.

Gent, R., Crabtree, B., and Ashley, R. (1996), “A review of model development based on sewer sediments research in the UK”, Water Science and Technology, Vol.33, No.9, pp.1-7.

Georgia Department of Natural Resources, Environmental Protection Division (1972), Water Qaulity Investigation of the Oconee River Basin, Georgia Environmental Protection Division, Atlanta, Georgia.

Georgia Department of Natural Resources, Environmental Protection Division (1976), Environmental Corridor Study, Georgia Environmental Protection Division, Atlanta, Georgia.

Georgia Department of Natural Resources, Environmental Protection Division (1984), Oconee River Basin Water Quality Management Plan, Second Edition, Georgia Environmental Protection Division, Atlanta, Georgia.

Georgia Department of Natural Resources, Environmental Protection Division (1990), Water Quality in Georgia, 1988-1989, Georgia Environmental Protection Division, Atlanta, Georgia.

Han, Q., (1980), “A study on the nonequilibrium transport of suspended load”, Proceedings of the International Symposium on River Sedimentation, Beijing, China, 793-802 (in Chinese)

Han, Q., and He, M. (1990), “A mathematical model for reservoir sedimentation and fluvial processes”, International Journal of Sediment Research, Vol.5, No.2, pp.43-84.

Hartikainen, H. (1996), "Cooccurrence and potential chemical competition of phosphorus and silicon in lake sediment", Water Research, Vol.30, No.10, pp.2472-2478.

Heathwaite, A.L. and Johnes, P.J. (1996), "Trends in nutrients", Hydrological Processes, Vol.10, pp.263-293.

Heathwaite, A.L. and Johnes, P.J. (1996), "Contribution of nitrogen species and phosphorus fractions to stream water quality in agricultural catchments", Hydrological Processes, Vol.10, pp.971-983.

Hellsten, S.K., Virtanen, M.O., Nenonen, O.S., Kinnunen, K.A., and Riihimaki, J.M. (1993), "Relative importance of internal sources of phosphorus and organic-matter in northern Finnish reservoirs", Water Science and Technology, Vol.28, No.6, pp.85-94.

Hem, J.D. (1985), Study and Interpretation of the Chemical Characteristics of Natural Water, Third Edition, U.S. Geological Survey Water-Supply Paper 2254, pp. 128

Holder, T.W., and Schretter, H.A. (1986), The Atlas of Georgia, Institute of Community and Area Development, University of Georgia, Athens, Georgia.

Hondzo, M. (1998), "Dissolved oxygen transfer at the sediment-water interface in a turbulent flow", Water Resources Research, Vol. 34, No. 12, pp.3525-3533.

House, W.A., Denison, F.H., and Armitage, P.D. (1995), "Comparison of the uptake of inorganic phosphorus to a suspended and stream bed sediment", Water Research, Vol.29, No.3, pp.767-779.

Jain, C.K., Ram, D. (1997), “Adsorption of lead and zinc on bed sediments of the River KALI”, Water Research, Vol. 31, No. 1, pp.154-162.

James, R.T. and Bierman, V.J. (1995), “A preliminary modeling analysis of water-quality in Lake Okeechobee, Florida – calibration results”, Water Research, Vol.29, No.12, pp.2755-2766.

Janse, J.H., Aldenberg, T., and Kramer, P.R.G. (1992), “A mathematical model of the phosphorus cycle in Lake Loosdrecht and simulation of additional measures”, Hydrobiologia, No. 233, pp.119-136

Johnes, P.J. and Burt, T.P. (1991), “Water quality trends in the Windrush catchment: nitrogen speciation and sediment interactions”, Proceedings of the Vienna Symposium, IAHS Publication no. 203.

Johnes, P.J. (1996), “Evaluation and management of the impact of land use change on the nitrogen and phosphorus load delivered to surface waters: the export coefficient modelling approach”, Journal of Hydrology, Vol. 183, pp.323-349.

Johnes, P.J. and Heathwaite, A.L. (1997), “Modelling the impact of land use change on water quality in agricutural catchments”, Hydrological Processes, Vol.11, pp.269-286.

Kalinske, A.A. (1947), “Movement of sediment as bed-load in rivers”, Transactions of the American Geophysical Union, Vol.28, No.4.

Kamp-Nielsen, L. (1989), “Sediment-Water Exchange Models”, Chapter 17 in: S.E. Jorgensen & M.J. Gromiec(EDS.) Mathematical Submodels in Water Quality Systems. Developments in Environmental Modelling14. Amsterdam: Elsevier, pp. 371-399

Keizer, P. and Sinke, A.J.C. (1992), "Phosphorus in the sediment of the Loosdrecht lakes and its implications for lake restoration perspectives", Hydrobiologia, Vol.233, pp.39-50.

Kleeberg, A. and Kozerski, H.-P. (1997), "Phosphorus release in Lake Groβer Müggelsee and its implications for lake restoration", Hydrobiologia, No 342/343, pp.9-26

Korom, S.F. (2000), "An adsorption isotherm for bromide", Water Resources Research, Vol.36, No.7, pp.1969-1974.

Kozerski, H.-P., Kohler, J. and Schellenberger, G. (1991), "Transport of particulate nutrients and pollutants in the low land river spree", Sediment and Stream Water Quality in a Changing Environment: Trends and Explanation (Proceedings of the Vienna Symposium, August 1991) IAHS Publication no. 203, pp.359-366.

Lane, E.W. and Kalinske, A.A. (1941), "Engineering calculations of suspended sediment", Transactions of the American Geophysical Union, Vol.20, pt.3, pp.603-607.

Lehtoranta, J. (1998), "Net sedimentation and sediment-water nutrient fluxes in the eastern Gulf of Finland (Baltic Sea)", Vie Et Milieu-Life and Environment, Vol.48, No.4, pp.341-352.

Lijklema, L., Koelmans, A.A., and Portielje, R. (1993), "Water quality impacts of sediment pollution and the role of early diagenesis", Water Science and Technology, Vol.28, No.8-9, pp.1-12.

Lin, C.F., Lee, Y.C., and Hao, O.J. (1995), "Modeling of nitrogen release in lake-sediments", Journal of the Chinese Institute of Engineers, Vol.18, No.1, pp.81-87.

Long, E.G. (1984), Letter to Ray Whittemore of Tufts University from Texas Department of Water Resources.

Mauldin, A.C. II, and McCollum, J.C. (1992), Status of the Chattahoochee River Fish Population Downstream of Atlanta, Georgia, Georgia Department of Natural Resources, Game and Fish Division, Final Report, Federal Aid Project F-26.

Maurer, W.R., Claflin, T.O., Rada, R.G., and Rogala, J.T. (1995), "Volume loss and mass-balance for selected physicochemical constituents in Lake Pepin, upper Mississippi River, USA", Regulated Rivers - Research and Management, Vol.11, No.2, pp175-184.

Mayer, T., Ptacek, C., and Zanini, L. (1999), "Sediments as a source of nutrients to hypereutrophic marches of Point Pelee, Ontario, Canada", Water Research, Vol. 33, No. 6, pp.1460-1470

Mazouni, N., Gaertner, J.C., Deslouspaoli, J.M., Landrein, S., and Doedenberg M.G. (1996), "Nutrient and oxygen exchange at the water-sediment interface in a shellfish farming lagoon (Thau, France)", Journal of Experimental Marine Biology and Ecology, Vol. 205, No. 1-2, pp.91-113.

McGregor, I., Ashley, R.M., and Oduyemi, K.O. (1993), "Pollutant release from sediments in sewer systems and their potential for release into receiving waters", Water Science and Technology, Vol.28, No.8-9, pp.161-169.

Meybeck, M. (1982), "Carbon, nitrogen and phosphorus transport by world rivers", American Journal of Science, Vol. 282, pp.401-450.

Meyer, J.L. (1979), "The role of sediments and bryophytes in phosphorus dynamics in a headwater stream ecosystem", Limnology and Oceanography, Vol.24, No.2, pp.365-375.

Meyer-Peter, E. and Muller, R. (1948), "Formula for bed-load transport", Proceedings of International Association for Hydraulic Research, 2nd meeting, Stockholm.

Mol, G.A. and Benoist, A.P. (1994), "Prediction of sediment quality in catchment areas of the River Rhine – operation and use of the HORIZON model", Water Science and Technology, Vol.29, No.3, pp.115-120.

Molinas, A., and Yang, C.T. (1986), Computer Program User's Manual for GSTARS (Generalized Stream Tube Model for Alluvial River Simulation), United States Bureau of Reclamation, Denver, CO.

Nairn, R.W. and Mitsch, W.J. (2000), "Phosphorus removal in created wetland ponds receiving river overflow", Ecological Engineering, Vol.14, No.1-2, pp.107-126.

Nowicki, B.L. (1997), "Nitrogen losses through sediment denitrification in Boston Harbor and Massachusetts Bay", Estuaries, Vol. 20, No. 3, pp.626-639

Parker, G. (1990), "Surface-based bedload transport relationship for gravel rivers", Journal of Hydraulic Research, Vol.28, No.4, pp.417-436.

Pfenning, K.S. and McMahon P.B. (1996), "Effect of nitrate, organic carbon, and temperature on potential denitrification rates in nitrate-rich river bed sediments", Journal of Hydrology, Vol.187, pp.283-295

Price, W.L. (1979), “A controlled random search procedure for global optimisation”, The Computer Journal, Vol.20, No.4, 367-370.

Qiu, S. and McComb, A.J. (1994), “Effects of oxygen concentration on phosphorus release from reflooded air-dried wetland sediments”, Australian Journal of Marine and Freshwater Research, Vol.45, No.7, pp.1319-1328.

Raaphorst, W.V., and H.T. Kloosterhuis (1994), “Phosphate Sorption in Superficial Intertidal Sediments”, Marine Chemistry, Vol.48, 1-16

Ramm, K. and Scheps, V. (1997), “Phosphorus balance of a polytrophic shallow lake with the consideration of phosphorus release”, Hydrobiologia, Vol. 342, pp.43-53

Reda, A.L.L., and Beck, M.B. (1997), “Ranking strategies for stormwater management under uncertainty: sensitivity analysis”, Water Science and Technology, Vol.36, No.5, pp.357-371.

Reddy, K.R., Fisher, M.M., and Ivanoff, D. (1996), “Resuspension and diffusive flux of nitrogen and phosphorus in a hypereutrophic lake”, Journal of Environmental Quality, Vol.25, No.2, pp.363-371.

Reddy, K.R., Flaig, E., Scinto, L.J., Diaz, O. and Debusk, T.A. (1996), “Phosphorus Assimilation in a Stream System of the Lake Okeechobee Basin”, Water Resources Bulletin, Vol. 32, No. 5, 901-915

Rizzo, W.M., and Christian, R.R. (1996), “Significance of Subtidal Sediments to Heterotrophically-Mediated Oxygen and Nutrient Dynamics in a Temperate Estuary”, Estuaries, Vol.19, No.2B, 475

Russell, M.A., Walling, D.E., Webb, B.W., and Bearne, R. (1998), “The composition of nutrient fluxes from contrasting UK river basins”, Hydrological Processes, Vol.12, No.9, pp.1461-1482.

Rutherford, J.C., Boyle, J.D., Elliott, A.H., Hatherell, T.V., and Chiu, T.W. (1995), “Modeling benthic oxygen-uptake by pumping”, Journal of Environmental Engineering-ASCE, Vol.121, No.1, pp.84-95.

Santiago, S., Thomas, R.L., Larbaigt, G., Corvi, C., Rossel, D. Tarradellas, J., Gregor, D.J., McCarthy, L., and Vernet, J.P. (1994), “Nutrient, heavy-metal and organic pollutant composition of suspended and bed sediments in the Rhone River”, Aquatic Sciences, Vol.56, No.3, pp.220-242.

Smaal, A.C. and Zurburg, W.(1997), “The Uptake and Release of Suspended and Dissolved Material by Oysters and Mussels in Marennes-Oleron Bay”, Aquatic Living Resources, Vol. 10, No.1, 23

Smits, J.G. and Van der Molen, D.T. (1993), “Application of SWITCH, a model for sediment water exchange of nutrients, to Lake Veluwe in the Netherlands”, Hydrobiologia, Vol. 253, No. 1-3, pp. 281-300

Simons, D.B. and Sentürk, F. (1992), Sediment Transport Technology: Water and Sediment Dynamics, Water Resources Publication

Tim, U.S. and Jolly, R. (1994), “Evaluating agricultural nonpoint-source pollution using integrated geographic information-systems and hydrologic/water quality model”, Journal of Environmental Quality, Vol.23, No.1, pp.25-35.

United States Army Corps of Engineers (1971), Expended Flood Plain Information Study for the Upper Oconee River Basin Georgia

United States Army Corps of Engineers (1974), Environmental Invertory, Altamaha River Basin, Georgia, U.S. Army Engineer District, Savanah, Georgia.

United States Army Corps of Engineers and U.S. Waterways Experiment Station (1996), HEC-6: Version 4.1 Scour and Deposition in Rivers and Reservoirs, Downloaded from: http://www.wrc-hec.usace.army.mil.

United States Army Engineer Waterways Experiment Station (1982), CE-QUAL-R1: A Numerical One-Dimensional Model of Reservoir Water Quality. U.S. Army Corps of Engineers Instruction Report E-821.

UusiKamppa, J., Braskerud, B., Jansson, H., Syversen, N., and Uusitalo, R. (2000), "Buffer zones and constructed wetlands as filters for agricultural phosphorus", Journal of Environmental Quality, Vol.29, No.1, pp.151-158.

Valett, H.M., Morrice, J.A., Dahm, C.N., and Campana, M.E. (1996), "Parent lithology, surface-groundwater exchange and nitrate retention in headwater streams", Limnology and Oceanography, Vol.41, No.2, pp.333-345.

Van der Molen, D.T. (1991), "A simple, dynamic model for the simulation of the release of phosphorus from sediments in shallow, eutrophic systems", Water Research, No.25, pp. 737-744

Van der Molen, D.T., Portielje, R., Boers, P.C.M. and Lijklema, L. (1998), “Changes in sediment phosphorus as a result of eutrophication and oligotrophication in Lake Veluwe, The Netherlands”, Water Research, Vol. 32, No. 11, pp.3281-3288

Van der Perk, Marcel (1996), Muddy Waters: Uncertainty issues in modelling the influence of bed sediments on water composition, Faculteit Ruimtelijke Wetenschappen Universiteit Utrecht.

Van Luijn, F., Boers, P.C.M., Lijklema, L., and Sweerts, J.-P.R.A. (1999), “Nitrogen fluxes and processes in sandy and muddy sediments from a shallow eutrophic lake”, Water Research, Vol.33, No.1, pp.33-42.

Van Rijn, L.C. (1984), “Sediment Transport, Part I: Bed Load Transport”, Journal of Hydraulic Engineering, ASCE, Vol.110, No.10, 1431-1456

Van Rijn, L.C. (1984), “Sediment Transport, Part II: Suspended Load Transport”, Journal of Hydraulic Engineering, ASCE, Vol.110, No.11, 1613-1641.

Wall, G.J., Bos, A.W., and Marshall, A.H. (1996), “The relationship between phosphorus and suspended sediment loads in Ontario watersheds”, Journal of Soil and Water Conservation, Vol.51, No.6, pp.504-507.

Walling, D.E., B.W. Webb, and M.A. Russell (1997), “Sediment-Associated Nutrient Transport in UK Rivers”, Symp. 4: Freshwater Contamination, IAHS Publication, No. 243, 69

White, J.S., Bayley, S.E., and Curtis, P.J. (2000), “Sediment storage of phosphorus in a northern prairie wetland receiving municipal and agro-industrial wastewater”, Ecological Engineering, Vol.14, No.1-2, pp.127-138.

Xie, J. (1990), River Simulation, Water Resources and Hydropower Publication (in Chinese)

Yang, C.T. (1972), “Unit Stream Power and Sediment Transport”, Journal of the Hydraulic Division, ASCE, Vol. 98, No. HY10, 1805-1826

Yang, C.T. (1974), “Incipient Motion and Sediment Transport”, Journal of the Hydraulic Division, ASCE, Vol. 99, No. HY10, 1679-1704

Yang, C.T. and Molinas, A. (1982), “Sediment Transport and Unit Stream Power Function”, Journal of the Hydraulic Division, ASCE, Vol. 108, No. HY6, 774-793

Yang, C.T. (1988), “Sediment Transport and Unit Stream Power”, Chapter 8 in: Civil Engineering Practice, 5/Water Resources/Environmental, edited by Cheremisinoff, Paul N., Nicholas P. Cheremisinoff and Su Ling Cheng, Technomic Publishing Company, Inc., pp. 265-289

Yang, C.T. (1996), “Sediment Transport in the Yellow River”, Journal of Hydraulic Engineering, ASCE, Vol. 122, No.5, 237-244

Yang, C.T. (1996), Sediment Transport: Theory and Practice, McGraw-Hill Companies, Inc.

Yang, C.T., Treviño, M.A., Simões, F.J.M. (1998), User's Manual for GSTARS 2.0 (Geeralized Stream Tube model for Alluvial River Simulation version 2.0), U.S. Department of the Interior, Bureau of Reclamation, Technical Service Center, Denver, Colorado

Zhang, Q., Zhang, Z., Yue, J., Duan, Z., and Dai, M. (1983), "A Mathematical Model for the Prediction of the Sedimentation Process in Rivers", Proceedings of the 2nd Interational Symposium on River Sedimentation, Nanjing, China

Zhang, R., Xie J., Wang, M., and Huang, J. (1989), Dynamics of River Sediments, Water Resources and Hydropower Publication (in Chinese)

Zhou, X. (1995), Computational Hydraulics, Tsinghua University Publication (in Chinese)

APPENDIX

C++ SOURCE CODE OF MODEL STAND

A.1. Header File

```
// header file hyd.h
// Abstract
// Declaration of class x_section

# include <iostream.h>
# include <fstream.h>
# include <string.h>
# include <math.h>
# include <stdlib.h>

const int maxNumSect=80; // maximum number of x-sections
const int maxNumObservations=1500; // maximum number of observations
// served as up- or down-stream conditions
const double nu=1.003e-6; // kinematic viscocity
const double g=9.81; // gravitational acceleration



class x_section
{
private:
      double width; // surface width
      double depth; // depth
      double wetArea; // wetted area of x-section
      double wetPeri; // wetted perimeter
      double hydR; // hydraulic radius
      double level; // surface level
      int numStn; // number of stations along a x-section
      double prf[150][2]; // cross-section profile
public:
      x_section(void);
      void setGauge(double gauge);
      void setProfile(int n, double profile[150][2]);
      void setfeature(void);
      double getWidth(void);
      double getDepth(void);
      double getArea(void);
      double getP(void);
      double getR(void);
};
```

```
//Declaration of class river
class river: public x_section
{
private:
      // -----> input info
      int numSections; // number of x-sections in the studied reach
      int numStnsInEachSection[maxNumSect]; // number of stations in
each sect.
      int numPrevStations[maxNumSect]; // number of stations above cur.
sect.
      int sumStns; // number of all the stations
      double csprofile0[4000][2]; // cross-section profiles original
      double csprofile1[4000][2]; // cross-section profiles to be
updated
      double sectionLocations[maxNumSect]; // location of each x-
section
      double sectionDistance[maxNumSect]; // distance in between

      //int numTimeSteps; // number of computational time steps
      double time; // time in prototype
      double maxTimeInterval; // maximum length of time step (sec)
      double timeInterval; // length of each time step (sec)
      double totalTimeSimulated; // total length of time simulated
(hour)

      int numObsFlow; // number of upstream flow rate observations
      double upstreamFlow[10000][2]; // upstream flow conditions
      int indicator1; // index showing time's corresponding position in
      // upstreamFlow[indicator1][0]
      int ratingCurve; // indicator of whether the following array
                      // is a rating curve or observed down'stage
      int numObsStage; // number of downstream stage observations
                       // or number of points in the rating curve
      double downstreamStage[5000][2]; // downstream stage
      // if ratingCurve == true
      //    1st column is discharge in cms
      //    2nd column is stage corresponding to discharge
      // if ratingCurve == false
      //    1st column is time of observation
      //    2nd column is observation of downstream stage
      int indicator2; // similar to indicator1, but with regard to
      // downstreamStage[indicator2][0]
      int numObsSusp; // number of upstream TSS observations
      double upSuspCon[maxNumObservations][2];
      // incoming suspended sediment concentration
      int indicator3; // similar to indicator1 and indicator2

      //double upStrCdn[maxTimeSteps][3]; // upstream conditions
      // This is a 3-column matrix, with the 1st one being upstream
      // discharge time series, 2nd one downstream stage time series,
      // and 3rd one upstream suspended sediment concentrations

      //double initialFlow[maxNumSect];
      //double initialStage[maxNumSect];
      //double initialSusp[maxNumSect]; // initial suspended sediment
concentration
      double initCdn[maxNumSect][7]; // initial conditions
```

```
      // This is a 7-column matrix, with the 1st one being initial
      // discharge condition, 2nd one initial stage condition,
      // 3rd one initial suspended sediment concentrations, and
      // 4th one initial phosphate concentrations
      // 5th one initial nitrate concentrations
      // 6th one initial ammonium concentrations
      // 7th one initial oxygen concentrations
      //double numCalibPoints; // number of observed points
      //double calibData[maxNumObservations][2]; // observed values for
calibration
      // first column time vector, second column suspended sediment
concentration

      int numObsPhos; // number of upstream phosphate observations
      double upPhos[maxNumObservations][2];
      // incoming PO4 concentration
      int indicator4; // simular to indicator1 and 2 and 3

      int numObsNitr; // number of upstream nitrate observations
      double upNitr[maxNumObservations][2];
      // incoming NO3 concentration
      int indicator6; // similar to indicator4

      int numObsOxy;
      // number of upstream and downstream oxygen conc'ns
      double upOxy[maxNumObservations][3];
      // upstream and downstream oxygen concentrations
      int indicator7; // similar to indicator6

      int numObsTemp; // number of temperature points
      double Temp[maxNumObservations][2];
      int indicator8;

      int numObsQlateral;
      // number of observations for lateral discharge
      double Qlateral[50][maxNumSect]; // later inflow rate
      int indicator9;

      int numObsCslateral;
      // number of observations for lateral sed inflow
      double Cslateral[50][maxNumSect]; //
      int indicator10;

      int numObsPO4lateral;
      // number of observations for lateral po4 inflow
      double PO4lateral[50][maxNumSect];
      // phosphate concentration in lateral inflow
      int indicator11;

      int numOfOutput; // total numbers of outputs
      double timeOfOutput[10000];
      // time in prototype when output is processed
      double po4Calib[1000]; // downstream po4 data
      double Jpo4;
      int indicator5; // position of time in timeOfOutput[indicator5]
```

```
	// <----- end input info

	// -----> variables to be computed
	double discharge[maxNumSect][2];
	// discharge at each section at each time step
	double stage[maxNumSect][2];
	// stage at each section at each time step
	double B[maxNumSect][2]; // width
	double P[maxNumSect][2]; // wetted perimeter
	double A[maxNumSect][2]; // wetted area
	double R[maxNumSect][2]; // hydraulic radius
	double D[maxNumSect][2]; // hydraulic depth
	//double velocity[maxNumSect][2]; // average velocity

	//double bedLoad[maxNumSect][2]; // bedload transport rate
	double suspLoadPot[maxNumSect][2];
	// suspended load transport potential
	double suspLoadAct[maxNumSect][2]; // suspended load actual rate
	double suspLoadActCon1[maxNumSect]; // conventional method 1
	double suspLoadActCon2[maxNumSect]; // conventional method 2
	double phos[maxNumSect][2]; // phosphate concentration
	double nitr[maxNumSect][2]; // nitrate concentration
	double ammo[maxNumSect][2]; // ammonium concentration
	double oxygen[maxNumSect][2]; // oxygen concentration
	//double algae[maxNumSect][2]; // algae concentration
	//double phosPart[maxNumSect]; // phosphorus in particulate form

	//solution of tridiagonal system of linear equations
	double solution[maxNumSect];

	// -----> end variables to be computed

	double getUpFlow(void);
	double getDownStage(void);
	double getUpSusp(void);
	double getUpPhos(void);
	double getUpNitr(void);
	double getUpAmmo(void);
	double getUpOxygen(void);
	double getTemp(void);
	double getCpeq(double Css, double C0, double kf, double pf);
	double getCameq(double Css, double C0, double kf, double pf);
	double getQlateral(int iOfSection);
	double getCslateral(int iOfSection);
	double getPO4lateral(int iOfSection);

public:
	//river(void);
	void setInfo(char datafile[20]);
	double computeModule(// parameters for hydraulics
		double roughmax, double roughmin, double Hmax, double Hmin,
		double theta,
		// parameters for sediment transport
		double Es, double d50, double kSedDep,
		double kSedEnt, double lamda,
		// parameters for PO4
		double Ep, double kPhos, double Cpint,
```

```
            double kf, double pf,
            // parameters for NO3
            double En, double kDeNi, double kNitr,
            double Mn,
            // parameters for NH4
            double Eammo, double kAmAd,   double kamf,
            double pamf, double Mam,
            // parameters for O2
            double Eoxy, double rPhoto, double rResp,
            double rAlgalGrowth, double rAlgalResp, double rSOD,
            double alfa5, double beta1);
      int computeHydraulics(double roughmax, double roughmin,
            double Hmax, double Hmin, double theta);
      void computeSedPot(double d50, double roughmax,
            double roughmin, double Hmax, double Hmin);
      void computeSedAct(double roughmax, double roughmin,
            double Hmax, double Hmin, double d50, double Es,
            double kSedDep, double kSedEnt);
      void solve(double alfa[maxNumSect], double beta[maxNumSect],
            double gama[maxNumSect], double delta[maxNumSect]);
      int computeProfile(double lamda);
      void computePhos(double Ep, double kPhos, double lamda,
            double Cpint, double kf, double pf);
      void computeNitr(double En, double kDeNi, double kNitr,
            double Mn, double Mam);
      void computeAmmo(double Eammo, double kNitr, double kAmAd,
            double kamf, double pamf, double Mn, double Mam);
      void computeOxy(double Eoxy, double rPhoto,
            double rResp, double rAlgalGrowth, double rAlgalResp,
            double rSOD, double alfa5, double beta1);

};
```

A.2. Inputting Data

```
// file river1.cpp
// implementation of class river

# include "hyd.h"

//river::river(void)
//{
//}

void river::setInfo(char datafile[20])
{
      // input information needed
      int i, j, stns=0;

      // open file "Info.txt" for information input
      ifstream infile;
      infile.open(datafile);
```

```
// -----> input total number of cross-sections along the river
infile >> numSections;
// <----- end input total num...


// -----> input number of stations along each cross-section
for (i=0; i<=numSections-1; i++)
{
      infile >> numStnsInEachSection[i];
      stns += numStnsInEachSection[i];
      numPrevStations[i]=stns-numStnsInEachSection[i];
}
sumStns = stns; // total number of stations in all x-sections
// <----- end input number of...


// -----> input cross-section profiles
for (i=0; i<=sumStns-1; i++)
{
      infile >> csprofile0[i][0];
      csprofile1[i][0]=csprofile0[i][0];
      infile >> csprofile0[i][1];
      csprofile1[i][1]=csprofile0[i][1];
}
// <----- end input cross...


infile >> totalTimeSimulated;
infile >> maxTimeInterval;

timeInterval = maxTimeInterval;


// -----> input cross-section locations along the reach
for (i=0; i<=numSections-1; i++)
      infile >> sectionLocations[i];
// <----- end input cross-...


// -----> compute distances between cross-sections
for (i=1; i<=numSections-1; i++)
      sectionDistance[i-1] = 1000.0 * (sectionLocations[i-1]
      - sectionLocations[i]);
// <----- end compute...


infile >> numObsFlow;
// -----> input upstream flow rate time series
for (i=0; i<=numObsFlow-1; i++)
{
      infile >> upstreamFlow[i][0];
      infile >> upstreamFlow[i][1];
}
// <----- end input upstream...

infile >> ratingCurve;
```

```
infile >> numObsStage;
// -----> input downstream stage time series
for (i=0; i<=numObsStage-1; i++)
{
      infile >> downstreamStage[i][0];
      infile >> downstreamStage[i][1];
}
// <----- end input downstream...


infile >> numObsSusp;
// -----> input upstream suspended sediment concentration
for (i=0; i<=numObsSusp-1; i++)
{
      infile >> upSuspCon[i][0];
      infile >> upSuspCon[i][1];
}
// <----- end input upstream sus...

infile >> numObsPhos;
// -----> input upstream phosphate concentration
for (i=0; i<=numObsPhos-1; i++)
{
      infile >> upPhos[i][0];
      infile >> upPhos[i][1];
}

infile >> numObsNitr;
// ----->input upstream nitrate concentration
for (i=0; i<=numObsNitr-1; i++)
{
      infile >> upNitr[i][0];
      infile >> upNitr[i][1];
}

infile >> numObsOxy;
// ----> input upstream oxygen conditions
for (i=0; i<=numObsOxy-1; i++)
{
      infile >> upOxy[i][0] >> upOxy[i][1];
}
// <---- end input up...

infile >> numObsTemp;
// ----> input upstream oxygen conditions
for (i=0; i<=numObsTemp-1; i++)
{
      infile >> Temp[i][0] >> Temp[i][1];
}
// <---- end input up...

infile >> numObsQlateral;
// ----> input lateral Q
for (i=0; i<=numObsQlateral-1; i++)
{
      for (j=0; j<=numSections-1; j++)
      {
```

```
            infile >> Qlateral[i][j];
      }
}
// <---- end input later...


infile >> numObsCslateral;
// ----> input lateral Cs
for (i=0; i<=numObsCslateral-1; i++)
{
      for (j=0; j<=numSections-1; j++)
      {
            infile >> Cslateral[i][j];
      }
}
// <---- end input lateral Cs

infile >> numObsPO4lateral;
// ----> input lateral po4
for (i=0; i<=numObsPO4lateral-1; i++)
{
      for (j=0; j<=numSections-1; j++)
      {
            infile >> PO4lateral[i][j];
      }
}
// <---- end input lateral po4


// -----> input initial flow rate along the reach
for (i=0; i<=numSections-1; i++)
{
      infile >> initCdn[i][0];
      discharge[i][0]=initCdn[i][0];
      infile >> initCdn[i][1];
      stage[i][0]=initCdn[i][1];
      infile >> initCdn[i][2];
      suspLoadAct[i][0]=initCdn[i][2];
      infile >> initCdn[i][3];
      phos[i][0]=initCdn[i][3];
      infile >> initCdn[i][4];
      nitr[i][0]=initCdn[i][4];
      infile >> initCdn[i][5];
      ammo[i][0]=initCdn[i][5];
      infile >> initCdn[i][6];
      oxygen[i][0]=initCdn[i][6];
}
// <----- end input initial flow...

// -----> input time to output results
infile >> numOfOutput;
for (i=0; i<=numOfOutput-1; i++)
{
      infile >> timeOfOutput[i];
}
// <----- end input time...
```

```
      infile.close();

}
```

A.3. Computation And Output Control

```
// file river2.cpp
// implementation of class river (continued)

# include "hyd.h"


double river::computeModule(double roughmax, double roughmin,
                        double Hmax, double Hmin,
                        double theta, double Es,
                        double d50, double kSedDep, double kSedEnt,
                        double lamda, double Ep, double kPhos,
                        double Cpint, double kf, double pf,
                        double En, double kDeNi, double kNitr,
                        double Mn, double Eammo, double kAmAd,
                        double kamf, double pamf, double Mam,
                        double Eoxy, double rPhoto, double rResp,
                        double rAlgalGrowth, double rAlgalResp,
                        double rSOD, double alfa5, double beta1)
{
      // river::computeModule
      //   calls different modules that compute hydraulics, sediment
      //   transport potential, actual sediment transport rate,
      //   deposition and entrainment, nutrient transport

      int i, n, std1, std2;
      int nInCsprofile = 0;
      int cs1Start, cs1End, pointsInCs1;
            //cs2Start, cs2End, pointsInCs2;

      // two consecutive cross-section profiles used in the computation
      double cs1[150][2];


      ofstream outfileQ;
      ofstream outfileZ;
      //ofstream ////outfileQsP;
      ofstream outfileQsA;
      //ofstream outfileQsACon1;
      //ofstream outfileQsACon2;
      ofstream outfileP;
      ofstream outfileN;
      ofstream outfileAmmo;
      ofstream outfileO2;
      //ofstream outfilePP;
      ////ofstream outfileA;
      ////ofstream outfileB;
      outfileQ.open("Q.txt");
      outfileZ.open("Z.txt");
```

```
////outfileQsP.open("QsP.txt");
outfileQsA.open("QsA.txt");
//outfileQsACon1.open("QsA1.txt");
//outfileQsACon2.open("QsA2.txt");
outfileP.open("phos.txt");
outfileN.open("nitr.txt");
outfileAmmo.open("ammo.txt");
outfileO2.open("o2.txt");
//outfilePP.open("phosPart.txt");
//outfileA.open("A.txt");
//outfileB.open("B.txt");

// write the initial values of discharge and stage into file
// Q.txt and file Z.txt; at the same time, set the values of
// discharge and stage at time step 1 equal to those at time step
// 0; Compute the sediment transport potential at time step 0,
// write the results into file QsP.txt
for (i=0; i<=numSections-1; i++)
{
      discharge[i][1] = discharge[i][0];

      stage[i][1] = stage [i][0];

      pointsInCs1 = numStnsInEachSection[i];
      cs1Start = nInCsprofile + 0;
      cs1End = nInCsprofile + pointsInCs1 - 1;
      for (n=0; n<=pointsInCs1-1; n++)
      {
            cs1[n][0] = csprofile0[n+cs1Start][0];
            cs1[n][1] = csprofile0[n+cs1Start][1];
      }
      setGauge(stage[i][0]);
      setProfile(pointsInCs1, cs1);
      setfeature();
      B[i][0] = getWidth();
      P[i][0] = getP();
      A[i][0] = getArea();
      R[i][0] = getR();
      D[i][0] = A[i][0]/B[i][0];
      // move the pointer to the beginning of next x-section
      // in the matrix csprofile
      nInCsprofile = nInCsprofile + pointsInCs1;


      suspLoadAct[i][1]=suspLoadAct[i][0];

}
// end for

computeSedPot(d50, roughmax, roughmin, Hmax, Hmin);
for (i=0; i<numSections; i++)
{
      suspLoadPot[i][0]=solution[i];
      suspLoadPot[i][1]=suspLoadPot[i][0];
}

// put a new line at the end of each row of discharg and stage
```

```
// values of the same time step
outfileQ << endl;
outfileZ << endl;
////outfileQsP << endl;
//outfileQsA << endl;
//outfileA << endl;
//outfileB << endl;


time=0.0;
indicator1=0;
indicator2=0;
indicator3=0;
indicator4=0;
indicator5=0;
indicator6=0;
indicator7=0;
indicator8=0;
indicator9=0;
indicator10=0;
indicator11=0;

// -----> start computation from time = 0.0
while (time<totalTimeSimulated) // time
{

      for (timeInterval=maxTimeInterval, std1=std2=-1;
      (std1==-1 || std2==-1) && (timeInterval>=120.0);
      timeInterval/=2.0)
      {

            // -----> compute hydraulics
            std1=computeHydraulics(roughmax, roughmin, Hmax,
                              Hmin, theta);
            // <----- end compute hydraulics

            // -----> compute total load sediment transport
            // potential by Yang's
            computeSedPot(d50, roughmax, roughmin, Hmax, Hmin);
            for (i=0; i<numSections; i++)
            {
                  suspLoadPot[i][1]=solution[i];
            }
            // <----- end compute total load...

            // -----> compute actual sediment transport rate
            computeSedAct(roughmax, roughmin, Hmax, Hmin,
                              d50, Es, kSedDep, kSedEnt);
            // <----- end compute actual...

            // -----> compute changes of x-section profiles
            std2=computeProfile(lamda);
            // <----- end compute changes of x-section...

            // -----> compute phosphate concentrations
            computePhos(Ep, kPhos, lamda, Cpint, kf, pf);
            // <----- end compute phosphate...
```

```
      // -----> compute nitrate concentrations
      computeNitr(En, kDeNi, kNitr, Mn, Mam);
      // <----- end compute nitrate...

      // -----> compute ammonium concentrations
      computeAmmo(Eammo, kNitr, kAmAd, kamf, pamf, Mn,
                  Mam);

      computeOxy(Eoxy, rPhoto, rResp, rAlgalGrowth,
            rAlgalResp, rSOD, alfa5, beta1);

}
// end for

if (std1==-1)
      cout << "hydraulics computation may not converge" <<
                  endl;

if (std2==-1)
      cout << "sediment computation may not converge" <<
                  endl;


// output at preset time
if (time<=timeOfOutput[indicator5] &&
      time+timeInterval/3600>=timeOfOutput[indicator5])
{

      outfileQ << time << " ";

      for (i=0; i<= numSections-1; i++)
      {
            outfileQ << discharge[i][1] << " ";
            outfileZ << stage[i][1] << " ";
            ////outfileQsP << suspLoadPot[i][1] << " ";
            outfileQsA << suspLoadAct[i][1] << " ";
            //outfileQsACon1 << suspLoadActCon1[i] << " ";
            //outfileQsACon2 << suspLoadActCon2[i] << " ";
            outfileP << phos[i][1] << " ";
            outfileN << nitr[i][1] << " ";
            outfileAmmo << ammo[i][1] << " ";
            outfileO2 << oxygen[i][1] << " ";
            //outfilePP << phosPart[i] << " ";
      }


      outfileQ << endl;
      outfileZ << endl;
      ////outfileQsP << endl;
      outfileQsA << endl;
      //outfileQsACon1 << endl;
      //outfileQsACon2 << endl;
      outfileP << endl;
      outfileN << endl;
      outfileAmmo << endl;
      outfileO2 << endl;
```

```
            //outfilePP << endl;
            //outfileA << endl;
            //outfileB << endl;

            indicator5++;
      }
      // end if ...

      if ((time/100.0-floor(time/100.0)) <=0.0025)
            cout << "Time = " << time << endl;

      time+=timeInterval/3600;

      // -----> update the upper condition and output results
      for (i=0; i<= numSections-1; i++)
      {
            discharge[i][0]=discharge[i][1];
            // set discharge[l][0] to the newly computed
            // discharge[l][1] and use the first one as the
            // upper condition to compute a newer latter one

            stage[i][0]=stage[i][1];
            // set stage[l][0] to the newly computed stage[l][1]
            // and use the first one as the upper condition to
            // compute a newer latter one

            suspLoadPot[i][0]=suspLoadPot[i][1];

            suspLoadAct[i][0]=suspLoadAct[i][1];

            phos[i][0]=phos[i][1];

            nitr[i][0]=nitr[i][1];

            ammo[i][0]=ammo[i][1];

            oxygen[i][0]=oxygen[i][1];

      }
      // end for


} // end while

outfileQ.close();
outfileZ.close();
////outfileQsP.close();
outfileQsA.close();
//outfileQsACon1.close();
//outfileQsACon2.close();
outfileP.close();
outfileN.close();
outfileAmmo.close();
outfileO2.close();
//outfilePP.close();
////outfileA.close();
////outfileB.close();
```

```
      return Jpo4;

}
// end of computeModule
```

A.4. Computation Of Open-Channel Hydraulics

```
// file river3.cpp
// implementation of class river

# include "hyd.h"

int river::computeHydraulics(double roughmax, double roughmin,
                        double Hmax, double Hmin, double theta)
{
// ---------------------------------------------------------------------
// river::computeHydraulics
//   used to compute one-dimensional open-channel hydraulics using
//   Preissmann implicit method
//   compute bedload sediment transport rate
//   compute suspended sediment transport potential
//   compute suspended sediment concentration based on advection-
//     diffusion equation
//
// ---------------------------------------------------------------------
      int i, k, n, kmax;
      double AM, // value of A at point M in the computational grid
            BM, RM, QM, ZM, // see explanation of AM
            pZpt, // value of partial derivative Z over partial t
            pApx, // value of partial derivative A over partial x
            pQpt, // value of partial derivative Q over partial t
            pQpx, // value of partial derivative Q over partial x
            pZpx, // value of partial derivative Z over partial x
            sf, // energy slope
            dRdZ1, dRdZ2, // intermediate variables in determining
                        // partial derivative R over partial Z
            G0, // upper boundary condition determined left-hand-side
                // value of the mass conservation equation
            FN, // lower boundary condition determined left-hand-side
              // value of the momentum conservation equation
            normdZ, normdQ, // judging standard for convergence
            dZdQ[2 * maxNumSect],
            // correction terms for the linearized eq. system
            dQ[maxNumSect], dZ[maxNumSect],
            // correction terms for Q and Z respectively
            // x term in the eq. Ax=b
            F,
            G,
            roughness, ql,
            Residue [2 * maxNumSect], // b term in the eq. Ax=b

            // elements in the banded coefficient matrix, A term in the
```

```
      // eq. Ax=b
      // the matrix has the structure
      //
      // d0 c0 f0
      // a0 d1 c1 f1
      // e0 a1 d2 c2 f2
      //    e1 a2 d3 c3 f3
      //       e2 a3 d4 c4 f4
      //           .  .  .  .  .
      //              .  .  .  .  .
      //                 e  a  d  c  f
      //                    e  a  d  c
      //                       e  a  d
      //
      // This is a pentadiagonal system and can be solved using
      // Gaussian elimination
      e[2 * maxNumSect],
      a[2 * maxNumSect],
      d[2 * maxNumSect],
      c[2 * maxNumSect],
      f[2 * maxNumSect],

      xmult, // Gaussian eliminatin correction factor

      // two consecutive cross-section profiles used in the
      // computation
      cs1[150][2], cs2[150][2];



      // variables used in determining cross-section profiles
      int nInCsprofile = 0;
      int cs1Start, cs1End, pointsInCs1,
            cs2Start, cs2End, pointsInCs2;



for (i=0; i<= 2*maxNumSect-1; i++)
{
      e[i]=a[i]=d[i]=c[i]=f[i]=dZdQ[i]=Residue[i]=0.0;
}
// set the initial value in the banded matrix


// prediction-correction iteration index k
k = 0;
kmax = 30;

// initialization of judging factor
normdZ = 1.0;
normdQ = 10.0;


// -----> compute hydraulics along the reach, recompute
// hydraulics if convergence standard is not met
while ((k < kmax) & ((normdZ > 0.02) | (normdQ > 2.0)))
{
```

```
k++;

nInCsprofile = 0;

// -----> compute hydraulic info from the 1st cross-section
for (i = 0; i <= numSections-2; i++)
// sptial scale
{

      // individual x-section profile
      // number of data points in the profile of x-section
      // (i) and (i+1)
      pointsInCs1 = numStnsInEachSection[i];
      pointsInCs2 = numStnsInEachSection[i+1];

      // starting and ending index of the data points of
      // x-section (i) and (i+1) in the matrix csprofile
      cs1Start = nInCsprofile + 0;
      cs1End = nInCsprofile + pointsInCs1 - 1;
      cs2Start = cs1End + 1;
      cs2End = cs1End + pointsInCs2 - 1;

      // profile of x-section (i)
      for (n=0; n<=pointsInCs1-1; n++)
      {
                  cs1[n][0] = csprofile0[n+cs1Start][0];
                  cs1[n][1] = csprofile0[n+cs1Start][1];
      }

      for (n=0; n<=pointsInCs2-1; n++)
      {
                  cs2[n][0] = csprofile0[n+cs2Start][0];
                  cs2[n][1] = csprofile0[n+cs2Start][1];
      }


      // move the pointer to the beginning of next x-
      // section in the matrix csprofile
      nInCsprofile = nInCsprofile + pointsInCs1;


      // computation of the down-left corner of the square
      // surrounding point M
      setGauge(stage[i][0]);
      setProfile(pointsInCs1, cs1);
      setfeature();
      B[i][0] = getWidth();
      P[i][0] = getP();
      A[i][0] = getArea();
      R[i][0] = getR();


      // computation of the down-right corner of the square
      // surrounding point M
      setGauge(stage[i+1][0]);
      setProfile(pointsInCs2, cs2);
      setfeature();
```

```
B[i+1][0] = getWidth();
P[i+1][0] = getP();
A[i+1][0] = getArea();
R[i+1][0] = getR();

// computation of the up-left corner of the square
// surrounding point M
setGauge(stage[i][1]);
setProfile(pointsInCs1, cs1);
setfeature();
B[i][1] = getWidth();
P[i][1] = getP();
A[i][1] = getArea();
R[i][1] = getR();

// determination of roughness
D[i][1] = A[i][1]/B[i][1];
if (D[i][1] > Hmin)
      roughness = roughmin;
else if (D[i][1] < Hmax)
      roughness = roughmax;
else
{
      double a, b;
      a = (roughmax-roughmin)/(Hmax-Hmin);
      b = roughmax - a*Hmax;
      roughness = a*D[i][1]+b;
}


// computation of the up-right corner of the square
// surrounding point M
setGauge(stage[i+1][1]);
setProfile(pointsInCs2, cs2);
setfeature();
B[i+1][1] = getWidth();
P[i+1][1] = getP();
A[i+1][1] = getArea();
R[i+1][1] = getR();

// the determination of dR/dZ at (i, j+1)
setGauge(stage[i][1]+0.02);
setProfile(pointsInCs1, cs1);
setfeature();
dRdZ1 = (getR() - R[i][1])/0.02;

// the determination of dR/dZ at (i+1, j+1)
setGauge(stage[i+1][1]+0.02);
setProfile(pointsInCs2, cs2);
setfeature();
dRdZ2 = (getR() - R[i+1][1])/0.02;


// hydraulic conditions for point M
// wetter area at point M
AM = theta * (A[i+1][1] + A[i][1])/2 +
      (1-theta) * (A[i+1][0] + A[i][0])/2;
```

```
// surface witimeIntervalh at point M
BM = theta * (B[i+1][1] + B[i][1])/2 +
      (1-theta) * (B[i+1][0] + B[i][0])/2;

// hydraulic radius at point M
RM = theta * (R[i+1][1] + R[i][1])/2 +
      (1-theta) * (R[i+1][0] + R[i][0])/2;

// discharge at point M
QM = theta * (discharge[i+1][1] + discharge[i][1])/2
      + (1-theta) * (discharge[i+1][0] +
      discharge[i][0])/2;

// stage at point M
ZM = theta * (stage[i+1][1] + stage[i][1])/2 +
      (1-theta) * (stage[i+1][0] + stage[i][0])/2;

// Expression of partial Z over partial t at point M
pZpt = (stage[i+1][1] + stage[i][1] –
      stage[i+1][0] - stage[i][0]) /
      (2*timeInterval);

// Expression of partial A over partial x at point M
pApx = (theta * (A[i+1][1] - A[i][1]) +
      (1-theta) * (A[i+1][0] - A[i][0])) /
      sectionDistance[i];

// Expression of partial Q over partial t at point M
pQpt = (discharge[i+1][1] + discharge[i][1] -
      discharge[i+1][0] - discharge[i][0]) /
      (2*timeInterval);

// Expression of partial Q over partial x at point M
pQpx = (theta * (discharge[i+1][1] - discharge[i][1])
       + (1-theta) * (discharge[i+1][0] –
      discharge[i][0])) / sectionDistance[i];

// Expression of partial Z over partial x at point M
pZpx = (theta * (stage[i+1][1] - stage[i][1]) +
      (1-theta) * (stage[i+1][0] - stage[i][0])) /
      sectionDistance[i];

// Expression of energy slope
sf = roughness * roughness * QM * QM *
      pow(RM, -4/3)/(AM * AM);

// left hand side of eq. of mass conservation and
// left hand side of eq. of momentum conservation
ql = getQlateral(i+1);
F = BM * pZpt + pQpx - ql; // ql is lateral inflow
G = pQpt + 2 * QM / AM * pQpx –
      QM * QM / (AM * AM) * pApx +
      g * AM * pZpx + g * AM * sf;

// right hand side of the two equations should be 0,
// but there might be some residuas
```

```
        Residue[2*i+1] = - F;
        Residue[2*i+2] = - G;


        // Coefficients in the correction matrix

        // partial F(i) over partial Q(i)
        a[2*i] = - theta / sectionDistance[i];

        // partial F(i) over partial Z(i)
        d[2*i+1] = BM / (2*timeInterval);

        // partial F(i) over partial Q(i+1)
        c[2*i+1] = theta / sectionDistance[i];

        // partial F(i) over partial Z(i+1)
        f[2*i+1] = BM / (2*timeInterval);

        // partial G(i) over partial Q(i)
        e[2*i] = 1/(2*timeInterval) +
              theta/AM*(pQpx-2*QM/sectionDistance[i]) +
              (-pApx*QM*theta/pow(AM, 2)) +
              (g * roughness * roughness *
              QM*theta/AM/pow(RM, 4/3));

        // partial G(i) over partial Z(i)
        a[2*i+1] = (- QM * theta * pQpx * B[i][1] /(AM * AM))
              + QM * QM * theta * B[i][1] / (AM * AM) *
              (1/sectionDistance[i]+pApx/AM) +
              g * theta * (pZpx * B[i][1] / 2 –
              AM/sectionDistance[i]) +
              - g * roughness * roughness * QM * QM *
              theta / AM / (pow(RM, 4/3)) * (2/3/RM*dRdZ1 +
              B[i][1]/2/AM);

        // partial G(i) over partial Q(i+1)
        d[2*i+2] = 1/(2*timeInterval)+
              theta/AM*(pQpx+2*QM/sectionDistance[i]) +
              (-pApx*QM*theta/pow(AM, 2)) +
              (g*roughness * roughness*QM*theta/AM/
              pow(RM, 4/3));

        // partial G(i) over partial Z(i+1)
        c[2*i+2] = (-QM*theta*pQpx*B[i+1][1]/pow(AM, 2)) +
              QM*QM*theta*B[i+1][1]/pow(AM, 2)*(-
              1/sectionDistance[i]+pApx/AM) +
              g*theta*(pZpx*B[i+1][1]/2+AM/sectionDistance[i]
              ) + -g*roughness *
              roughness*QM*QM*theta/AM/pow(RM, 4/3)*
              (2/3/RM*dRdZ2+B[i+1][1]/2/AM);

    }
    // end for (i = 0; i <= numSections-2; i++) loop,
    // spatial scale

    // indicate if convergence standard is not met
```

```
// -----> set boundary conditions
// the upper boundary condition which specifies the
// inflow to the reach ( or the corresponding stages ), G0
G0 = discharge[0][1] - getUpFlow();
Residue[0] = - G0;
d[0] = 1; // partial G0 over partial Q(1) is 1
c[0] = 0; // partial G0 over partial Z(1) is 1

// the lower boundary condition which is the rating
// curve at the lower boundary or the stage time series at
// the lower boundary
FN = stage[numSections-1][1] - getDownStage();
// downstream stage
Residue[2 * numSections-1] = - FN;
a[2*numSections-2] = 0; // partial FN/partial Q(N)
d[2*numSections-1] = 1; // partial FN/partial Z(N)
// <----- end set boundary...

// -----> solve the linear system of equations
// solution of the correction terms
// using numerical solution for a banded linear system
for (i=1; i<=2*numSections-2; i++)
{
      xmult = a[i-1]/d[i-1];
      d[i] = d[i] - xmult * c[i-1];
    c[i]=c[i]-xmult*f[i-1];
      Residue[i]=Residue[i]-xmult*Residue[i-1];

      xmult=e[i-1]/d[i-1];
      a[i]=a[i]-xmult*c[i-1];
      d[i+1]=d[i+1]-xmult*f[i-1];
      Residue[i+1]=Residue[i+1]-xmult*Residue[i-1];
} // end for

xmult=a[2*numSections-2]/d[2*numSections-2];
d[2*numSections-1]=d[2*numSections-1]-
      xmult*c[2*numSections-2];
dZdQ[2*numSections-1]=(Residue[2*numSections-1]
      -xmult*Residue[2*numSections-2])/d[2*numSections-1];
dZdQ[2*numSections-2]=(Residue[2*numSections-2]-
      c[2*numSections-2]
      *dZdQ[2*numSections-1])/d[2*numSections-2];

for (i=2*numSections-3; i>=0; i--)
      dZdQ[i]=(Residue[i]-f[i]*dZdQ[i+2]-
            c[i]*dZdQ[i+1])/d[i];
// see pp.253  "Numerical Mathematics and Computing"
// by Cheney & Kincaid
// <----- end solve...


// -----> correction procedure
normdZ=0.0;
normdQ=0.0;

for (i = 0; i <= numSections-1; i++)
{
```

```
                  dQ[i] = dZdQ[2*i];
                  discharge[i][1] += dQ[i];
                  normdQ += dQ[i] * dQ[i];

                  dZ[i] = dZdQ[2*i+1];
                  stage[i][1] += dZ[i];
                  normdZ += dZ[i] * dZ[i];
            } // for loop
            // <----- end correction procedure

            // set convergence index
            normdZ = sqrt(normdZ);
            normdQ = sqrt(normdQ);

      }
      // <----- end of while loop, the computation of hydraulics for
      // one time step is finished

      if (k >= kmax)
            return -1;
      else
            return 0;
}
// <----- end of function computeHydraulics
```

A.5. Computation Of Sediment Transport Potential

```
// file river4.cpp
// implementation of class river

# include "hyd.h"

void river::computeSedPot(double d50, double roughmax, double roughmin,
                          double Hmax, double Hmin)
{
// compute the total sediment transport potential, given the hydraulic
// conditions: velocity, energy slope, particle size, and sheer
// velocity

      int i;
      double ws, VcrOverWs, subtraction, Clog, v, sf, ustar, roughness;

      if (d50<0.1e-3)
      {
            ws=(2.65-1.0)*g*d50*d50/(18*nu);
      }
      if (d50>=1e-3)
      {
            ws=1.1*sqrt((2.65-1.0)*g*d50);
      }
      else
      {
            ws=10*nu/d50*(sqrt(1.0+0.01*(2.65-
                  1.0)*g*d50*d50*d50/(nu*nu))-1.0);
```

```
	}

	// fall velocity according to Van Rijn's Sediment Transport,
	// Part II: Suspended Load Transport, Journal of Hydraulic
	// Engineering, Vol.110, No. 11, pp.1613-1641

	for (i=0; i<=numSections-1; i++)
	{
		v=fabs(discharge[i][1]/A[i][1]);

		// to make sure no supercritical flow occurs
		if (v/(sqrt(g*A[i][1]/B[i][1]))>=1.0 &&
			discharge[i][1]<100.0)
		{
			v=1.0;
		}

		if (D[i][0] > Hmin)
			roughness = roughmin;
		else if (D[i][0] < Hmax)
			roughness = roughmax;
		else
		{
			double a, b, roughness;
			a = (roughmax-roughmin)/(Hmax-Hmin);
			b = roughmax - a*Hmax;
			roughness = a*D[i][0]+b;
		}

		sf = roughness * roughness * v * v * pow(R[i][1], -4/3);

		ustar=sqrt(g*R[i][1]*sf);


		if (ustar*d50/nu>=70)
			VcrOverWs=2.05;
		else
			VcrOverWs=2.5/(log10(ustar*d50/nu)-0.06)+0.66;

		if ((v*sf/ws-VcrOverWs*sf)>0.0)
			subtraction=v*sf/ws-VcrOverWs*sf;
		else
			subtraction=v*sf/ws;

		Clog=5.435-0.286*log10(ws*d50/nu)-0.457*log10(ustar/ws)+
			(1.799-0.409*log10(ws*d50/nu)-0.314*log10(ustar/ws))*
			log10(subtraction);

		solution[i]=pow(10, Clog);
		// equilibrium suspended sediment concentration in ppm
		solution[i]=1/(1/2650+(1e6/solution[i]-1)*1e-3);
		// convert back to international unit kg/m^3
	}
	// end for

}
```

A.6. Computation Of Actual Sediment Transport

```
// file river5.cpp
// implementation of class river

# include "hyd.h"

void river::computeSedAct(double roughmax, double roughmin,
                        double Hmax, double Hmin, double d50,
                        double Es, double kSedDep, double kSedEnt)
{
      // -----> assign values for the coef matrix elements
      // in the expression Ax=b, A is the coef matrix (tridiagonal),
      // x is the concentration vector to be solved, and b is the
      // RHS of the eq.

      // Description of the tridiagonal matrix A
      // beta0  gama0
      // alfa0  beta1  gama1
      //        alfa1  beta2  gama2
      //               alfa2  beta3  gama3
      //                      ..     ..     ..
      //                             ..     ..    ..
      //                                    alfa  beta  gama
      //                                          alfa  beta

      int i;
      double // shear velocity, kinematic viscosity,
            // velocity(Q/A), dispersion coef, particle size,
            // sediment deposition/entrainment constant
            ksed, ws, ql, sl,

            // tridiagonal elements in solving the advection-diffusion
            //eq
            alfa[maxNumSect], beta[maxNumSect], gama[maxNumSect],

            // RHS of the ad-df difference eq, source/sink term,
            // delta x,
            delta[maxNumSect], product, dx, q, K1, ustar, sf, a1, a2,
            roughness;
      double const pi=3.14159265;

      beta[0]=1.0; gama[0]=0.0; delta[0]=getUpSusp();

      if (d50<0.1e-3)
      {
            ws=(2.65-1.0)*g*d50*d50/(18*nu);
      }
      if (d50>=1e-3)
      {
            ws=1.1*sqrt((2.65-1.0)*g*d50);
      }
      else
      {
```

```
      ws=10*nu/d50*(sqrt(1.0+0.01*(2.65-
            1.0)*g*d50*d50*d50/(nu*nu))-1.0);
}

for (i=1; i<=numSections-2; i++)
{
      dx=(sectionDistance[i-1]+sectionDistance[i])/2;
      alfa[i-1]=-discharge[i][1]/(2*dx)-A[i][1]*Es/(dx*dx);
      beta[i]=A[i][1]/timeInterval+2*A[i][1]*Es/(dx*dx);
      gama[i]=discharge[i][1]/(2*dx)-A[i][1]*Es/(dx*dx);
      // kSed=kSedDep*(ws*dx/q) and kSed=kSedEnt*(q/(dx*ws))
      if (suspLoadAct[i][1]>suspLoadPot[i][1])
            ksed=kSedDep*(B[i][1]*dx*ws)/discharge[i][1];
      else
            ksed=kSedEnt*discharge[i][1]/(B[i][1]*dx*ws);


      ql=getQlateral(i);
      sl=getCslateral(i);
      product=-ksed*(suspLoadAct[i][0]-suspLoadPot[i][1])+
            2*ql*sl/(A[i-1][1]+A[i][1]);
      //product=0.0; // in case of no source/sink effects
      delta[i]=A[i][1]*suspLoadAct[i][0]/timeInterval+
            A[i][1]*product;
}
alfa[numSections-2]=-discharge[numSections-1][1]/
      (2*sectionDistance[numSections-2])-
      A[numSections-1][1]*Es/
      (sectionDistance[numSections-2]*
      sectionDistance[numSections-2])-
      (discharge[numSections-1][1]/
      (2*sectionDistance[numSections-2])-
      A[numSections-1][1]*Es/
      (sectionDistance[numSections-2]*
      sectionDistance[numSections-2]));
beta[numSections-1]=A[numSections-1][1]/timeInterval+
      2*A[numSections-1][1]*Es/
      (sectionDistance[numSections-2]*
      sectionDistance[numSections-2])+
      2*(discharge[numSections-1][1]/
      (2*sectionDistance[numSections-2])-
      A[numSections-1][1]*Es/
      (sectionDistance[numSections-2]*
      sectionDistance[numSections-2]));
// kSed=kSedDep*(ws*dx/q) and kSed=kSedEnt*(q/(dx*ws))
if (suspLoadAct[numSections-1][0]>suspLoadPot[numSections-1][1])
      ksed=kSedDep*(B[numSections-1][1]*dx*ws)/
      discharge[numSections-1][1];
else
      ksed=kSedEnt*discharge[numSections-1][1]/
      (B[numSections-1][1]*dx*ws);


ql=getQlateral(numSections-1);
sl=getCslateral(numSections-1);
product=-ksed*
      (suspLoadAct[numSections-1][0]-suspLoadPot[numSections-
```

```
      1][1])+2*ql*sl/(A[numSections-2][1]+A[numSections-1][1]);
//product=0.0; // in case of no source/sink effects
delta[numSections-1]=A[numSections-1][1]*
      suspLoadAct[numSections-1][0]/timeInterval+
      A[numSections-1][1]*product;

// <----- end assign...

solve(alfa, beta, gama, delta);

suspLoadActCon1[0]=getUpSusp();
suspLoadActCon2[0]=getUpSusp();

for (i=0; i<numSections; i++)
{
      if (i>0)
      {
            if (D[i][0] > Hmin)
                  roughness = roughmin;
            else if (D[i][0] < Hmax)
                  roughness = roughmax;
            else
            {
                  double a, b, roughness;
                  a = (roughmax-roughmin)/(Hmax-Hmin);
                  b = roughmax - a*Hmax;
                  roughness = a*D[i][0]+b;
            }

            dx=sectionDistance[i-1];
            q=discharge[i][1]/B[i][1];
            ustar=sqrt(g*R[i][1]*sf);
            sf = roughness * roughness * discharge[i][1] *
                  discharge[i][1] * pow(R[i][1], -4/3)/
                  (A[i][1]*A[i][1]);
            K1=6*ws/(0.4*ustar);

            if (suspLoadPot[i][1]<suspLoadAct[i][1])
            {
                  a1=1e-9; //1+K1/2.0;
                  a2=1e-9; //0.25;
            }
            else
            {
                  a1=1e-9; //pi*pi/K1+K1/4.0;
                  a2=1e-9; //1.0;
            }

            suspLoadActCon1[i]=suspLoadPot[i][1]+
                  exp(-a1*ws*dx/q)*
                  (suspLoadActCon1[i-1]-suspLoadPot[i][1]);

            suspLoadActCon2[i]=suspLoadPot[i][1]+
                  exp(-a2*ws*dx/q)*
                  (suspLoadActCon2[i-1]-suspLoadPot[i-1][1])+
                  q/(a2*ws*dx)*(1-exp(-a2*ws*dx/q))*
                  (suspLoadPot[i-1][1]-suspLoadPot[i][1]);
```

```
            }// if

            suspLoadAct[i][1]=(solution[i]>0? solution[i]:0);
      }
      // for loop

}
```

A.7. Solution To Linear System Of Equations With Tri-diagonal Coefficient Matrix

```
// file river6.cpp
// implementation of class river

# include "hyd.h"

void river::solve(double alfa[maxNumSect], double beta[maxNumSect],
                  double gama[maxNumSect], double delta[maxNumSect])
{
      int i;
      double xmult;

      // -----> solve the tridiagonal eq. system
      for (i=1; i<=numSections-1; i++)
      {
            xmult=alfa[i-1]/beta[i-1];
            beta[i]=beta[i]-xmult*gama[i-1];
            delta[i]=delta[i]-xmult*delta[i-1];
      }

      solution[numSections-1]=delta[numSections-1]/
            beta[numSections-1];

      for (i=numSections-2; i>=0; i--)
      {
            solution[i]=(delta[i]-
                  gama[i]*solution[i+1])/beta[i];
      }

      // <----- end solve...
}
// end function solve
```

A.8. Computation Of Morphological Changes

```
// file river7.cpp
// implementation of class river (continued)

# include "hyd.h"

int river::computeProfile(double lamda)
{
```

```
int i, j, n, numOfForks, head[40], tail[40];
int cs1Start, cs1End;

double dAb, widthBed, dh, Qs11, Qs12, Qs21, Qs22, std=0.0;

// two consecutive cross-section profiles used in the computation
double left[40], right[40], left0[40], right0[40]; //, sigmaH,
      //h[150], dh[150], dy[150];


for (i=1; i<numSections; i++)
{

      // get stations and elevations of each x-section
      // pointsInCs1 = numStnsInEachSection[i];
      cs1Start = numPrevStations[i];
      cs1End =  numPrevStations[i]+numStnsInEachSection[i]-1;

      //for (n=0; n<=numStnsInEachSection[i]-1; n++)
      //{
      //    cs1[n][0] = csprofile0[n+cs1Start][0];
      //    cs1[n][1] = csprofile0[n+cs1Start][1];
      //}

      // initialize the water surface indicators
      for (j=0; j<40; j++)
      {
            left[j]=0.0;
            right[j]=0.0;
            left0[j]=0.0;
            right0[j]=0.0;
            head[j]=-1;
            tail[j]=-1;
      }
      numOfForks=0;

      // compute area of deposition or entrainment
      Qs11=suspLoadAct[i][0]*discharge[i][0]/2650;
      Qs12=suspLoadAct[i][1]*discharge[i][1]/2650;
      Qs21=suspLoadAct[i-1][0]*discharge[i-1][0]/2650;
      Qs22=suspLoadAct[i-1][1]*discharge[i-1][1]/2650;

      dAb=timeInterval*(Qs11+Qs12-Qs21-Qs22)/
            (sectionDistance[i-1]+sectionDistance[i])/
            (lamda-1.0);

      // adjustment of x-section profile
      // find the locations of water surfaces given stage
      for (n=cs1Start+1; n<=cs1End-1; n++)
      {
            //h[n-cs1Start-1]=stage[i][1]-csprofile0[n][1];
            if (csprofile0[n][1]>=stage[i][1] &&
                  csprofile0[n+1][1]<stage[i][1])
            {
                  numOfForks+=1;
                  left[numOfForks-1]=csprofile0[n+2][0];
                  left0[numOfForks-1]=csprofile0[n+1][0];
```

```
                        head[numOfForks-1]=n+1;
                  }
                  if (csprofile0[n][1]<stage[i][1] &&
                        csprofile0[n+1][1]>=stage[i][1])
                  {
                        right[numOfForks-1]=csprofile0[n-1][0];
                        right0[numOfForks-1]=csprofile0[n][0];
                        tail[numOfForks-1]=n;
                  }
            }

            // determine width on which deposition or entrainment is
            // applied
            widthBed=0.0;
            j=0;

            while (j<numOfForks)
            {
                  if (right[j]>left[j])
                        widthBed+=(right[j]-left[j])+
                        (left[j]-left0[j])/2.0+
                        (right0[j]-right[j])/2.0;
                  j++;
            }


            // determine height of deposition or entrainment
            if (fabs(widthBed)<=1e-4)
                  dh=0.0;
            else
                  dh=dAb/widthBed;


            // adjustment of elevation of points under water
            for (j=0; j<numOfForks; j++)
            {
                  for (n=cs1Start;n<=cs1End; n++)
                  {
                        if (csprofile0[n][0]>=left[j] &&
                              csprofile0[n][0]<=right[j] )
                              csprofile1[n][1]=csprofile0[n][1]+dh;
                  }
            }

            if (fabs(std)<fabs(dh))
                  std=dh;

      }
      // end for


      if (fabs(std)>0.20)
            return -1;
      else
            return 0;

}
```

```
// end of function
```

A.9. Functions That Provide Intermediate Results

```
// file river8.cpp
// implementation of class river (continued)

#include "hyd.h"

double river::getUpFlow(void)
{
      double upFlow;

      while (upstreamFlow[indicator1][0]<time)
            indicator1++;

      upFlow=(time-upstreamFlow[indicator1-1][0])/
            (upstreamFlow[indicator1][0]-
            upstreamFlow[indicator1-1][0])*
            (upstreamFlow[indicator1][1]-
            upstreamFlow[indicator1-1][1])+
            upstreamFlow[indicator1-1][1];

      return upFlow;
}


double river::getDownStage(void)
{
      double downStage;
      int i;

      if (ratingCurve==1)
      {
            for (i=0; downstreamStage[i][0]<discharge[numSections-1][0]
                  && i<= numObsStage-1;i++);

            downStage=downstreamStage[i-1][1]+
                  (discharge[numSections-1][1]-
                  downstreamStage[i-1][0])*
                  (downstreamStage[i][1]-downstreamStage[i-1][1])/
                  (downstreamStage[i][0]-downstreamStage[i-1][0]);

      }
      else
      {
            while (downstreamStage[indicator2][0]<time)
                  indicator2++;

            downStage=(time-downstreamStage[indicator2-1][0])/
                  (downstreamStage[indicator2][0]-
                  downstreamStage[indicator2-1][0])*
                  (downstreamStage[indicator2][1]-
                  downstreamStage[indicator2-1][1])+
```

```
                  downstreamStage[indicator2-1][1];
      }

      return downStage;
}


double river::getUpSusp(void)
{
      double upSusp;

      while (upSuspCon[indicator3][0]<time)
            indicator3++;

      upSusp=(time-upSuspCon[indicator3-1][0])/
            (upSuspCon[indicator3][0]-upSuspCon[indicator3-1][0])*
            (upSuspCon[indicator3][1]-upSuspCon[indicator3-1][1])+
            upSuspCon[indicator3-1][1];

      return upSusp;
}


double river::getUpPhos(void)
{
      double upPhosCon;

      while (upPhos[indicator4][0]<time)
            indicator4++;

      upPhosCon=(time-upPhos[indicator4-1][0])/
            (upPhos[indicator4][0]-upPhos[indicator4-1][0])*
            (upPhos[indicator4][1]-upPhos[indicator4-1][1])+
            upPhos[indicator4-1][1];

      return upPhosCon;
}


double river::getUpNitr(void)
{
      double upNitrCon;

      while (upNitr[indicator6][0]<time)
            indicator6++;

      upNitrCon=(time-upNitr[indicator6-1][0])/
            (upNitr[indicator6][0]-upNitr[indicator6-1][0])*
            (upNitr[indicator6][1]-upNitr[indicator6-1][1])+
            upNitr[indicator6-1][1];

      return upNitrCon;
}


double river::getUpAmmo(void)
{
```

```
	return 0.3;
}


double river::getUpOxygen(void)
{
	double upOxygen;

	while (upOxy[indicator7][0]<time)
		indicator7++;

	upOxygen=(time-upOxy[indicator7-1][0])/
		(upOxy[indicator7][0]-upOxy[indicator7-1][0])*
		(upOxy[indicator7][1]-upOxy[indicator7-1][1])+
		upOxy[indicator7-1][1];

	return upOxygen;
}


double river::getTemp(void)
{
	double Temperature;

	while (Temp[indicator8][0]<time)
		indicator8++;

	Temperature=(time-Temp[indicator8-1][0])/
		(Temp[indicator8][0]-Temp[indicator8-1][0])*
		(Temp[indicator8][1]-Temp[indicator8-1][1])+
		Temp[indicator8-1][1];

	return Temperature;
}


double river::getCpeq(double Css, double C0, double kf, double pf)
{
	double x0=1e-30, x1=10.0, f0, f1, x, f=1.0;
	f0=Css*kf*pow(x0, pf)-(C0-x0);
	f1=Css*kf*pow(x1, pf)-(C0-x1);
	int n=0;

	while ((fabs(f)>=1e-5) && (n<100))
	{
		x=(x0+x1)/2;
		f=Css*kf*pow(x, pf)-(C0-x);
		n++;
		if (f*f1<0)
		{
			x0=x;
			f0=f;
		}
		else
		{
			x1=x;
			f1=f;
```

```
            }
      }


      return x;

}


double river::getCameq(double Css, double C0, double kf, double pf)
{
      double x0=1e-30, x1=2.0, f0, f1, x, f=1.0;
      f0=Css*kf*pow(x0, pf)-(C0-x0);
      f1=Css*kf*pow(x1, pf)-(C0-x1);
      int n=0;

      while ((fabs(f)>=1e-5) && (n<100))
      {
            x=(x0+x1)/2;
            f=Css*kf*pow(x, pf)-(C0-x);
            n++;
            if (f*f1<0)
            {
                  x0=x;
                  f0=f;
            }
            else
            {
                  x1=x;
                  f1=f;
            }
      }

      return x;
}



double river::getQlateral(int iOfSection)
{
      double ql;

      while (Qlateral[indicator9][0]<time)
            indicator9++;

      ql=(time-Qlateral[indicator9-1][0])/
            (Qlateral[indicator9][0]-Qlateral[indicator9-1][0])*
            (Qlateral[indicator9][iOfSection]-
            Qlateral[indicator9-1][iOfSection])+
            Qlateral[indicator9-1][iOfSection];

      return ql;
}


double river::getCslateral(int iOfSection)
{
```

```
      double sl;

      while (Cslateral[indicator10][0]<time)
            indicator10++;

      sl=(time-Cslateral[indicator10-1][0])/
            (Cslateral[indicator10][0]-Cslateral[indicator10-1][0])*
            (Cslateral[indicator10][iOfSection]-
            Cslateral[indicator10-1][iOfSection])+
            Cslateral[indicator10-1][iOfSection];

      return sl;
}



double river::getPO4lateral(int iOfSection)
{
      double pl;

      while (PO4lateral[indicator11][0]<time)
            indicator11++;

      pl=(time-PO4lateral[indicator11-1][0])/
            (PO4lateral[indicator11][0]-PO4lateral[indicator11-1][0])*
            (PO4lateral[indicator11][iOfSection]-
            PO4lateral[indicator11-1][iOfSection])+
            PO4lateral[indicator11-1][iOfSection];

      return pl;
}
```

A.10. Functions That Compute Coefficient Matrices Of Different Water Constituents

```
// file river9.cpp
// implementation of class river

# include "hyd.h"

void river::computePhos(double Ep, double kPhos, double lamda,
                        double Cpint, double kf, double pf)
{
      // -----> assign values for the coef matrix elements
      // in the expression Ax=b, A is the coef matrix (tridiagonal),
      // x is the concentration vector to be solved, and b is the
      // RHS of the eq.

      // Description of the tridiagonal matrix A
      // beta0  gama0
      // alfa0  beta1  gama1
      //        alfa1  beta2  gama2
      //               alfa2  beta3  gama3
      //                      ..     ..     ..
```

```
//                                ..     ..    ..
//                                       alfa  beta  gama
//                                             alfa  beta

int i;
double
// tridiagonal elements in solving the advection-diffusion eq
alfa[maxNumSect], beta[maxNumSect], gama[maxNumSect],

// RHS of the ad-df difference eq, source/sink term, delta x,
// concentration of phosphate adsorbed on particle surfaces (this
// is a function of the concentration of SS
delta[maxNumSect], product, dx, Cpeq[maxNumSect],
Qs11, Qs12, Qs21, Qs22, ql, pl; //conPhosInSS;

beta[0]=1.0; gama[0]=0.0; delta[0]=getUpPhos();
Cpeq[0]=getCpeq(suspLoadAct[0][1], phos[0][0], kf, pf);

for (i=1; i<=numSections-2; i++)
{
      dx=(sectionDistance[i-1]+sectionDistance[i])/2;
      alfa[i-1]=-discharge[i][1]/(2*dx)-A[i][1]*Ep/(dx*dx);
      beta[i]=A[i][1]/timeInterval+2*A[i][1]*Ep/(dx*dx);
      gama[i]=discharge[i][1]/(2*dx)-A[i][1]*Ep/(dx*dx);


      Qs11=suspLoadAct[i][0]*discharge[i][0]/2650;
      Qs12=suspLoadAct[i][1]*discharge[i][1]/2650;
      Qs21=suspLoadAct[i-1][0]*discharge[i-1][0]/2650;
      Qs22=suspLoadAct[i-1][1]*discharge[i-1][1]/2650;

      Cpeq[i]=getCpeq(suspLoadAct[i][1], phos[i][0], kf, pf);

      ql=getQlateral(i);
      pl=getPO4lateral(i);
      if (Qs11+Qs12-Qs21-Qs22 <= 0)
            product= -kPhos*(phos[i][0]-Cpeq[i])+
            (-(Qs11+Qs12-Qs21-Qs22))*Cpint*2*lamda/
            ((1-lamda)*(A[i-1][1]+A[i][1])*sectionDistance[i-1])+
            2*ql*pl/(A[i-1][1]+A[i][1]);
      else
            product=-kPhos*(phos[i][0]-Cpeq[i])+
            2*ql*pl/(A[i-1][1]+A[i][1]);


      //product=0.0; // in case of no source/sink effect
      delta[i]=A[i][1]*phos[i][0]/timeInterval+
            A[i][1]*product;
}
alfa[numSections-2]=-discharge[numSections-1][1]/
      (2*sectionDistance[numSections-2])-
      A[numSections-1][1]*Ep/
      (sectionDistance[numSections-2]*
      sectionDistance[numSections-2])-
      (discharge[numSections-1][1]/
      (2*sectionDistance[numSections-2])-
      A[numSections-1][1]*Ep/
```

```
            (sectionDistance[numSections-2]*
            sectionDistance[numSections-2]));
      beta[numSections-1]=A[numSections-1][1]/timeInterval+
            2*A[numSections-1][1]*Ep/
            (sectionDistance[numSections-2]*
            sectionDistance[numSections-2])+
            2*(discharge[numSections-1][1]/
            (2*sectionDistance[numSections-2])-
            A[numSections-1][1]*Ep/
            (sectionDistance[numSections-2]*
            sectionDistance[numSections-2]));


      Qs11=suspLoadAct[numSections-1][0]*
            discharge[numSections-1][0]/2650;
      Qs12=suspLoadAct[numSections-1][1]*
            discharge[numSections-1][1]/2650;
      Qs21=suspLoadAct[numSections-1-1][0]*
            discharge[numSections-1-1][0]/2650;
      Qs22=suspLoadAct[numSections-1-1][1]*
            discharge[numSections-1-1][1]/2650;

      Cpeq[numSections-1]=getCpeq(suspLoadAct[numSections-1][1],
            phos[numSections-1][0], kf, pf);

      ql=getQlateral(numSections-1);
      pl=getPO4lateral(numSections-1);
      if (Qs11+Qs12-Qs21-Qs22 <= 0)
            product= -kPhos*(phos[numSections-1][0]-
                  Cpeq[numSections-1])+
                  (-(Qs11+Qs12-Qs21-Qs22))*Cpint*2*lamda/
                  ((1-lamda)*(A[numSections-2][1]+A[numSections-1][1])*
                  sectionDistance[numSections-2]);
      else
            product=-kPhos*(phos[numSections-1][0]-
                  Cpeq[numSections-1])+2*ql*pl/(A[numSections-
                  2][1]+A[numSections-1][1]);

      //product=0.0; // in case of no source/sink effect
      delta[numSections-1]=A[numSections-1][1]*
            phos[numSections-1][0]/timeInterval+
            A[numSections-1][1]*product;

      // <----- end assign...


      solve(alfa, beta, gama, delta);

      for (i=0; i<numSections; i++)
            phos[i][1]=(solution[i]>=0.0? solution[i]:0.0);

}


void river::computeNitr(double En, double kDeNi, double kNitr,
                                    double Mn, double Mam)
{
```

```
int i;
double
// tridiagonal elements in solving the advection-diffusion eq
alfa[maxNumSect], beta[maxNumSect], gama[maxNumSect],

// RHS of the ad-df difference eq, source/sink term, delta x,
// concentration of phosphate adsorbed on particle surfaces (this
// is a function of the concentration of SS
delta[maxNumSect], product, dx, o2;

beta[0]=1.0; gama[0]=0.0; delta[0]=getUpNitr();


for (i=1; i<=numSections-2; i++)
{
      dx=(sectionDistance[i-1]+sectionDistance[i])/2;
      alfa[i-1]=-discharge[i][1]/(2*dx)-A[i][1]*En/(dx*dx);
      beta[i]=A[i][1]/timeInterval+2*A[i][1]*En/(dx*dx);
      gama[i]=discharge[i][1]/(2*dx)-A[i][1]*En/(dx*dx);

      o2=oxygen[i][0];
      product=-kDeNi*nitr[i][0]+o2/(Mn+o2)*kNitr*
            ammo[i][0]/(ammo[i][0]+Mam);

      product=0.0; // in case of no source/sink effects
      delta[i]=A[i][1]*nitr[i][0]/timeInterval+
            A[i][1]*product;
}
alfa[numSections-2]=-discharge[numSections-1][1]/
      (2*sectionDistance[numSections-2])-
      A[numSections-1][1]*En/
      (sectionDistance[numSections-2]*
      sectionDistance[numSections-2])-
      (discharge[numSections-1][1]/
      (2*sectionDistance[numSections-2])-
      A[numSections-1][1]*En/
      (sectionDistance[numSections-2]*
      sectionDistance[numSections-2]));
beta[numSections-1]=A[numSections-1][1]/timeInterval+
      2*A[numSections-1][1]*En/
      (sectionDistance[numSections-2]*
      sectionDistance[numSections-2])+
      2*(discharge[numSections-1][1]/
      (2*sectionDistance[numSections-2])-
      A[numSections-1][1]*En/
      (sectionDistance[numSections-2]*
      sectionDistance[numSections-2]));

o2=oxygen[numSections-1][0];
product=-kDeNi*nitr[numSections-1][0]+
      o2/(Mn+o2)*kNitr*ammo[numSections-1][0]/
      (ammo[numSections-1][0]+Mam);

product=0.0; // in case of no source/sink effects
delta[numSections-1]=A[numSections-1][1]*
      nitr[numSections-1][0]/timeInterval+
```

```
            A[numSections-1][1]*product;

      // <----- end assign...


      solve(alfa, beta, gama, delta);

      for (i=0; i<numSections; i++)
            nitr[i][1]=(solution[i]>=0.0? solution[i]:0.0);

}



void river::computeAmmo(double Eammo, double kNitr, double kAmAd,
                                    double kamf, double pamf, double
Mn,
                                    double Mam)
{

      int i;
      double
      // tridiagonal elements in solving the advection-diffusion eq
      alfa[maxNumSect], beta[maxNumSect], gama[maxNumSect],

      // RHS of the ad-df difference eq, source/sink term, delta x,
      // concentration of phosphate adsorbed on particle surfaces (this
      // is a function of the concentration of SS
      delta[maxNumSect], product, dx, Cameq, o2;

      beta[0]=1.0; gama[0]=0.0; delta[0]=getUpAmmo();


      for (i=1; i<=numSections-2; i++)
      {
            dx=(sectionDistance[i-1]+sectionDistance[i])/2;
            alfa[i-1]=-discharge[i][1]/(2*dx)-A[i][1]*Eammo/(dx*dx);
            beta[i]=A[i][1]/timeInterval+2*A[i][1]*Eammo/(dx*dx);
            gama[i]=discharge[i][1]/(2*dx)-A[i][1]*Eammo/(dx*dx);

            Cameq=getCameq(suspLoadAct[i][1], ammo[i][0], kamf, pamf);

            o2=oxygen[i][0];
            product=-o2*(Mn+o2)*kNitr*ammo[i][0]//  /(ammo[i][0]+Mam)
                  -kAmAd*(ammo[i][0]-Cameq);

            product=0.0; // in case of no source/sink effects
            delta[i]=A[i][1]*ammo[i][0]/timeInterval+
                  A[i][1]*product;
      }
      alfa[numSections-2]=-discharge[numSections-1][1]/
            (2*sectionDistance[numSections-2])-
            A[numSections-1][1]*Eammo/
            (sectionDistance[numSections-2]*
            sectionDistance[numSections-2])-
            (discharge[numSections-1][1]/
            (2*sectionDistance[numSections-2])-
```

```
            A[numSections-1][1]*Eammo/
            (sectionDistance[numSections-2]*
            sectionDistance[numSections-2]));
      beta[numSections-1]=A[numSections-1][1]/timeInterval+
            2*A[numSections-1][1]*Eammo/
            (sectionDistance[numSections-2]*
            sectionDistance[numSections-2])+
            2*(discharge[numSections-1][1]/
            (2*sectionDistance[numSections-2])-
            A[numSections-1][1]*Eammo/
            (sectionDistance[numSections-2]*
            sectionDistance[numSections-2]));

      Cameq=getCameq(suspLoadAct[numSections-1][1], ammo[i][0],
            kamf, pamf);
      o2=oxygen[numSections-1][0];
      product=-o2/(Mn+o2)*kNitr*ammo[numSections-1][0]// /
            //(ammo[numSections-1][0]+Mam)
            -kAmAd*(ammo[numSections-1][0]-Cameq);

      product=0.0; // in case of no source/sink effects
      delta[numSections-1]=A[numSections-1][1]*
            ammo[numSections-1][0]/timeInterval+
            A[numSections-1][1]*product;

      // <----- end assign...


      solve(alfa, beta, gama, delta);

      for (i=0; i<numSections; i++)
            ammo[i][1]=(solution[i]>=0.0? solution[i]:0.0);

}


void river::computeOxy(double Eoxy, double rPhoto,
            double rResp, double rAlgalGrowth, double rAlgalResp,
            double rSOD, double alfa5, double beta1)
{

      int i;
      double
      // tridiagonal elements in solving the advection-diffusion eq
      alfa[maxNumSect], beta[maxNumSect], gama[maxNumSect],

      // RHS of the ad-df difference eq, source/sink term, delta x,
      // concentration of phosphate adsorbed on particle surfaces (this
      // is a function of the concentration of SS
      delta[maxNumSect], product, dx, oxySat, T, u, kreO;
      const double algae = 0.005;

      beta[0]=1.0; gama[0]=0.0; delta[0]=getUpOxygen();

      T=getTemp();
      oxySat=14.6*exp(-T*(0.027767-0.00027*T+0.000002*T*T));
```

```
for (i=1; i<=numSections-2; i++)
{
      dx=(sectionDistance[i-1]+sectionDistance[i])/2;
      alfa[i-1]=-discharge[i][1]/(2*dx)-A[i][1]*Eoxy/(dx*dx);
      beta[i]=A[i][1]/timeInterval+2*A[i][1]*Eoxy/(dx*dx);
      gama[i]=discharge[i][1]/(2*dx)-A[i][1]*Eoxy/(dx*dx);

      u=discharge[i][0]/A[i][0];
      kreO=2.6*pow(u, 0.273)/pow(R[i][0], 0.894)/86400;
      //in original form: 1.923*pow(...
      //in original form: pow(u, 0.273), pow(R[i][0], 0.894)

      product=kreO*(oxySat-oxygen[i][0])+(rPhoto*rAlgalGrowth-
            rResp*rAlgalResp)*algae-
            rSOD/R[i][0]-
            alfa5*beta1*ammo[i][0];

      //product=0.0; // in case of no source/sink effects
      delta[i]=A[i][1]*oxygen[i][0]/timeInterval+
            A[i][1]*product;
}
alfa[numSections-2]=-discharge[numSections-1][1]/
      (2*sectionDistance[numSections-2])-
      A[numSections-1][1]*Eoxy/
      (sectionDistance[numSections-2]*
      sectionDistance[numSections-2])-
      (discharge[numSections-1][1]/
      (2*sectionDistance[numSections-2])-
      A[numSections-1][1]*Eoxy/
      (sectionDistance[numSections-2]*
      sectionDistance[numSections-2]));
beta[numSections-1]=A[numSections-1][1]/timeInterval+
      2*A[numSections-1][1]*Eoxy/
      (sectionDistance[numSections-2]*
      sectionDistance[numSections-2])+
      2*(discharge[numSections-1][1]/
      (2*sectionDistance[numSections-2])-
      A[numSections-1][1]*Eoxy/
      (sectionDistance[numSections-2]*
      sectionDistance[numSections-2]));

product=kreO*(oxySat-oxygen[numSections-1][0])+
      (rPhoto*rAlgalGrowth-rResp*rAlgalResp)*algae-
      rSOD/R[numSections-1][0]-
      alfa5*beta1*ammo[numSections-1][0];

//product=0.0; // in case of no source/sink effects
delta[numSections-1]=A[numSections-1][1]*
      oxygen[numSections-1][0]/timeInterval+
      A[numSections-1][1]*product;

// <----- end assign...


solve(alfa, beta, gama, delta);

for (i=0; i<numSections; i++)
```

```
            oxygen[i][1]=(solution[i]>=0.0? solution[i]:0.0);

}
```

A.11. Functions That Compute Hydraulic Variables Of A Cross-Section

```
// file x_sectn.cpp
// implementation of class x_section

# include "hyd.h"

x_section::x_section()
{
}


void x_section::setGauge(double gauge)
{
      level = gauge;
}


void x_section::setProfile(int n, double profile[150][2])
{
      int i, j;
      for(i=0; i<=1; i++)
            for (j=0; j<=n-1; j++)
                  prf[j][i]=profile[j][i];
      numStn=n;

}


void x_section::setfeature(void)
{
      int i;
      double stnStart, stnEnd, P, A, B, R, max, min;

      max=prf[0][1];
      min=prf[0][1];
      for (i=1; i<numStn; i++) // -1; i++)
      {
            if (prf[i][1]>max)
                  max=prf[i][1];
            if (prf[i][1]<min)
                  min=prf[i][1];
      }


      if (level>max)
      {
            //cout << "Stage level higher than levee." << endl;
            level = max - 0.001;
      }
```

```
      if (level<min)
      {
            //cout << "Stage level lower than thelwag." << endl;
            level = min + 0.05;
      }


      P=A=B=R=0.1e-10;

      for ( i = 0; i < numStn-1; i++ ) // for i=1:n-1
      {
            if ( ( prf[i][1] >= level ) &
                  ( prf[i+1][1] <= level ) )
                  // surface level lower than point (i) but higher than
(i+1)
            {

                        // point where water surface begins
                  stnStart = prf[i+1][0] + ( level - prf[i+1][1] )
                        * ( prf[i][0] - prf[i+1][0] )
                        / ( prf[i][1] - prf[i+1][1] );

                  P = P + sqrt ( ( stnStart - prf[i+1][0] ) *
                        ( stnStart - prf[i+1][0] ) +
                        ( level - prf[i+1][1] ) *
                        ( level - prf[i+1][1] ) );

                  A = A + fabs ( ( level - prf[i+1][1] ) *
                        ( prf[i+1][0] - stnStart ) / 2 );
            }

            if ( ( prf[i][1] < level ) &
                  ( prf[i+1][1] < level ) )
                  // surface level above both point (i) and (i+1)
            {

                  P = P + sqrt ( ( prf[i][0] - prf[i+1][0] ) *
                        ( prf[i][0] - prf[i+1][0] ) +
                        ( prf[i][1] - prf[i+1][1] ) *
                        ( prf[i][1] - prf[i+1][1] ) );

                  A = A + fabs ( ( level - ( prf[i][1] +
                        prf[i+1][1] ) / 2 ) *
                        ( prf[i+1][0] - prf[i][0] ) );
            } // end if

            if ( ( prf[i][1] <= level ) &
                  ( prf[i+1][1] >= level ) )
                  // surface level higher than point (i) but lower than
(i+1)
            {

                  stnEnd = prf[i+1][0] +
                        ( level - prf[i+1][1] ) *
                        ( prf[i][0] - prf[i+1][0] ) /
                        ( prf[i][1] - prf[i+1][1] );
```

```
                  P = P + sqrt ( ( stnEnd - prf[i][0] ) *
                        ( stnEnd - prf[i][0] ) +
                        ( level - prf[i][1] ) *
                        ( level - prf[i][1] ) );

                  A = A + fabs ( ( level - prf[i][1] ) *
                        ( stnEnd - prf[i][0] ) / 2 );
                  B = B + stnEnd - stnStart;
            } // end if

      } // end of for loop

      R = A / P;

      width = B;
      depth = A / B;
      wetArea = A;
      wetPeri = P;
      hydR = R;

} // end of function


double x_section::getWidth(void)
{
      return width;
}


double x_section::getDepth(void)
{
      return depth;
}


double x_section::getArea(void)
{
      return wetArea;
}


double x_section::getP(void)
{
      return wetPeri;
}


double x_section::getR(void)
{
      return hydR;
}
```

A.12. Application File

```
// main.cpp
// application

# include "hyd.h"

int main()
{
      double roughmax, roughmin, Hmax, Hmin, //roughness,
            theta, d50, Es, kSedDep, kSedEnt, lamda, Ep,
            kPhos, Cpint, kf, pf, En, kDeNi, kNitr, Eammo,
            kAmAd, kamf, pamf, Mn, Mam, Eoxy, rPhoto,
            rResp, rAlgalGrowth, rAlgalResp, rSOD, alfa5,
            beta1, J;
      char datafile[40];

      river testRiver;

      cout << "Please type in name of datafile: ";
      cin >> datafile;
      cout << endl;
      testRiver.setInfo(datafile);

      Es=50.0; //50.0; // 500.0 for a trial on June 30.
      roughmax=0.030; // 0.020 for Weihe...;  0.030 for Oconee
      roughmin=0.030; // 0.015 for Weihe...;  0.030 for Oconee
      Hmax=3.0; //depth when roughness=roughmax
      Hmin=4.5; //depth when roughness=roughmin
      //roughness =0.020; //0.020 for Weihe; //0.030 for Oconee;
      kSedDep = 4.8128e-009;
      //1.3e-9 for Weihe //4.8128e-009 for Oconee;
      kSedEnt = 0.0028;
      //1.8194e-006; //1.875e-6; //0.0028 for Oconee, Weihe;

      theta=0.65;
      d50=6.2745e-004;
      //6.2745e-004 for the Oconee ... //4.3585e-004 for Weihe
      lamda=0.4; //0.5; //0.4 for Oconee;

      Ep=500.0; //865.191; //1500.0; //
      kPhos=8e-6; //8e-6 for Oconee; //5e-6;
      Cpint=0.005; //0.005 for Oconee;
      kf=2.5e-3; //2.5e-3 for Oconee; //1.3e-3;
      pf=0.05; //0.05 for Oconee; 0.02;

      En=500.0; //1000.0;
      kDeNi=0.2e-5;
      kNitr=1.6e-5;

      Eammo=1000.0;
      kAmAd=5e-2;
      kamf=8e-6;
      pamf=0.02;
```

```
        Mn=4.0;
        Mam=0.05;

        Eoxy=10.0;
        rPhoto=1.8;
        rResp=1.9;
        rAlgalGrowth=2.0/86400.0;
        rAlgalResp=0.25/86400.0;
        rSOD=3e-5; //0.8e-6; // variable according to Qual-II mannual
        // = 3e-5 with out consideration of sediment concentration
        alfa5=3.5;
        beta1=0.6/86400.0;


        J = testRiver.computeModule(roughmax, roughmin, Hmax, Hmin,
              theta, Es, d50, kSedDep,
              kSedEnt, lamda,   Ep, kPhos, Cpint, kf, pf, En, kDeNi,
              kNitr, Mn, Eammo, kAmAd, kamf, pamf, Mam, Eoxy,
              rPhoto, rResp, rAlgalGrowth, rAlgalResp, rSOD, alfa5,
              beta1);

        cout << " J = " << J << endl;

        return 0;
}
```