

ESTIMATION OF RIVER CHARACTERISTICS FROM REMOTE SENSING DATA

By

Thomas Gabriel Almeida

A DISSERTATION

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

DOCTOR OF PHILOSOPHY

Mechanical Engineering

2012

ABSTRACT

ESTIMATION OF RIVER CHARACTERISTICS FROM REMOTE SENSING DATA

By

Thomas Gabriel Almeida

A framework for estimation of river characteristics from observational data is presented using model inversion methodology. At the center of the method is a cost function defined by the error in observational versus modeled data (e.g., velocity). This cost function is extended through the use of Lagrange multipliers such that an inverse (or ‘adjoint’) model is developed, which is used to obtain a gradient of the cost function with respect to the river characteristic that is to be estimated. Using an appropriate descent algorithm, a model result fitting the data best and constrained by the original model equations is obtained. For this work, the model equations are the shallow water equations describing the flow of water in an open channel, in two-dimensions. The specific characteristic to be estimated is bathymetry. The observational data is in the form of simulated pseudo-steady-state velocity measurements (either surface or depth-averaged), and may be either sparse or full-field. A correlation between surface velocity and depth-averaged velocity is used to allow the use of two-dimensional modeling with three-dimensional data. The hydrostatic assumption inherent in the shallow water equations is shown to have an impact on the ability of the algorithm to accurately estimate the bathymetry; this impact is not more than that of resolution and noise typical of most experimental data. The ability of the algorithm to estimate bathymetry given different types, quantity and quality of velocity data is quantified. The bathymetry estimation is shown to be accurate and robust for multiple observational surface velocity data configurations on two different rivers.

DEDICATION

For my wife, Staci and my sons, Gabriel, Isaac, Elijah and Charles.

ACKNOWLEDGEMENTS

I would like to thank those responsible for the funding of my work at Michigan State University and SRI International. Without financial support and external interest, many interesting questions and problems such as those addressed in this work would never be pursued or solved.

I cannot express how much appreciation I have for the guidance of my advisor, Prof. Farhad Jaber. He has been unnecessarily supportive and helpful to me from the first day I stepped into his undergraduate thermodynamics class. I would not have been able to complete this dissertation if not for my supervisor and colleague, Dr. David Walker; this research started with his ideas and has developed under his guidance. I consider myself very fortunate to be working with him. I would also like to thank my committee members, Prof. Ahmed Naguib, Prof. Phanikumar Mantha, Prof. André Bénard, and Prof. Charles Petty. They have been patient with me and vital to my growth as an engineer and researcher. Additionally, I would like to thank all of the faculty and staff at Michigan State University, as well as my colleagues at SRI International, for surrounding me in excellent environments for learning and research.

Finally, I would like to thank my family and friends for always being there for me and pushing me to always do my best in everything that I encounter in life. I would have nothing in this world without my mother, my father, my sister and my wife. As they all know, if not for the love of my wife and children, I would be lost.

TABLE OF CONTENTS

LIST OF TABLES.....	vii
LIST OF FIGURES.....	viii
NOMENCLATURE.....	xii
INTRODUCTION.....	1
CHAPTER 1	
BACKGROUND AND MOTIVATION.....	6
1.1 Statement of Objectives.....	6
1.2 Summary of Previous Research.....	7
CHAPTER 2	
GOVERNING EQUATIONS AND COMPUTATIONAL METHODOLOGY.....	12
2.1 The Forward Model.....	12
2.2 Variational Framework.....	13
2.3 The Adjoint Solver.....	17
2.4 The Assimilation Algorithm.....	22
CHAPTER 3	
RESULTS AND DISCUSSION.....	25
3.1 Snohomish River	26
3.1.1 Geographical Context.....	26
3.1.2 Depth-Averaged (2D) Velocity Data.....	29
3.1.2.1 Full-Field Data.....	30
3.1.2.2 Sparse Data.....	33
3.1.2.3 Noisy Data.....	35
3.1.3 Surface (3D) Velocity Data.....	37
3.1.3.1 Depth-Averaged to Surface Velocity Correlation.....	38
3.1.3.2 Full-Field Data.....	39
3.1.3.3 Data Resolution.....	41
3.1.3.4 Data Noise.....	45
3.1.4 Hydrostatic Assumption.....	48
3.2 Kootenai River	51
3.2.1 Geographical Context.....	51
3.2.2 Full-Field Depth-Averaged (2D) Velocity Data.....	55
3.2.3 Sparse, Noisy Surface (3D) Velocity Data.....	57
3.3 Summary of Results.....	59
CHAPTER 4	
CONCLUSIONS.....	63

APPENDIX A	
SHALLOW WATER EQUATIONS.....	67
REFERENCES.....	75

LIST OF TABLES

Table 3.1. Bulk estimation statistics for different data resolutions with fixed noise level (20cm/s).....	45
Table 3.2. Bulk estimation statistics for different data noise levels with fixed resolution (20m).....	47
Table 3.3. Complete listing of all cases.....	60

LIST OF FIGURES

Figure 2.1. Vertical coordinate system	12
Figure 2.2. Simplified algorithm flow chart	22
Figure 2.3. Algorithm flow chart	23
Figure 3.1. The Snohomish River: region overview and bathymetry	27
Figure 3.2. ‘True’ simulation results of Snohomish River: (a) bathymetry, (b) velocity magnitude, (c) water surface elevation with velocity vectors, and (d) total water depth	29
Figure 3.3. Comparison of the total water depth for noise-free, full-resolution, depth-averaged velocity data: (a) ‘true’ depth, (b) initial depth, and (c) final estimate of depth	30
Figure 3.4. Comparison of the velocity magnitude for noise-free, full-resolution, depth-averaged velocity data: (a) ‘true’ velocity magnitude, (b) initial velocity magnitude, and (c) final estimate of velocity magnitude.....	30
Figure 3.5. Cost function progression	31
Figure 3.6. Point-by-point comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for noise-free, full-resolution, depth-averaged velocity data.....	32
Figure 3.7. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for noise-free, full-resolution, depth-averaged velocity data	32
Figure 3.8. Velocity magnitude statistics versus total water depth for noise-free, full-resolution, depth-averaged velocity data	33
Figure 3.9 Total water depth statistics versus velocity magnitude for noise-free, full-resolution, depth-averaged velocity data	33
Figure 3.10. Comparison of the total water depth for noise-free, 20m-resolution, depth-averaged velocity data: (a) ‘true’ depth and (b) final estimate of depth	34

Figure 3.11. Comparison of the velocity magnitude for noise-free, 20m-resolution, depth-averaged velocity data: (a) ‘true’ velocity magnitude and (b) final estimate of velocity magnitude	34
Figure 3.12. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for noise-free, 20m-resolution, depth-averaged velocity data	35
Figure 3.13. Comparison of the total water depth for 20cm/s RMS noise, 20m-resolution, depth-averaged velocity data: (a) ‘true’ depth and (b) final estimate of depth	36
Figure 3.14. Comparison of the velocity magnitude for 20cm/s RMS noise, 20m-resolution, depth-averaged velocity data: (a) ‘true’ velocity magnitude and (b) final estimate of velocity magnitude	36
Figure 3.15. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for 20cm/s RMS noise, 20m-resolution, depth-averaged velocity data	37
Figure 3.16. Comparison of the velocity magnitude: (a) 3D surface velocity magnitude and (b) 2D depth-averaged velocity magnitude.....	38
Figure 3.17. Surface velocity versus depth-averaged velocity for (a) Snohomish River and (b) Kootenai River.....	39
Figure 3.18. Comparison of the total water depth for noise-free, full-resolution, surface velocity data: (a) ‘true’ depth and (b) final estimate of depth	40
Figure 3.19. Comparison of the velocity magnitude for noise-free, full-resolution, surface velocity data: (a) ‘true’ velocity magnitude and (b) final estimate of velocity magnitude	40
Figure 3.20. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for noise-free, full-resolution, surface velocity data	41
Figure 3.21. Comparison of the total water depth for noise-free, 20m-resolution, surface velocity data: (a) ‘true’ depth and (b) final estimate of depth	42
Figure 3.22. Comparison of the velocity magnitude for noise-free, 20m-resolution, surface velocity data: (a) ‘true’ velocity magnitude and (b) final estimate of velocity magnitude	42

Figure 3.23. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for noise-free, 20m-resolution, surface velocity data	43
Figure 3.24. Comparison of the total water depth for fixed surface velocity data noise level (20cm/s): (a) ‘true’ depth, (b) estimated depth for 10m resolution data, (c) estimated depth for 20m resolution data, and (d) estimated depth for 40m resolution data.....	44
Figure 3.25. Comparison of the ‘true’ surface velocity magnitude for fixed data noise level (20cm/s): (a) full-resolution, noise-free velocity, (b) velocity for 10m resolution data, (c) velocity for 20m resolution data, and (d) velocity for 40m resolution data.....	44
Figure 3.26. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude with fixed surface velocity error (20cm/s).....	45
Figure 3.27. Comparison of the total water depth for fixed surface velocity data resolution (20m): (a) ‘true’ depth, (b) estimated depth for 0cm/s noise data, (c) estimated depth for 10cm/s noise data, (d) estimated depth for 20cm/s noise data, and (e) estimated depth for 40cm/s noise data.....	46
Figure 3.28. Comparison of the ‘true’ surface velocity magnitude for fixed data resolution (20m): (a) full-field, noise-free velocity, (b) velocity for 0cm/s noise data, (c) velocity for 10cm/s noise data, (d) velocity for 20cm/s noise data, and (e) velocity for 40cm/s noise data.....	47
Figure 3.29. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude with fixed surface velocity resolution (20m).....	48
Figure 3.30. Comparison of the velocity magnitude: (a) 2D depth-averaged velocity, (b) 3D hydrostatic surface velocity, and (c) 3D non-hydrostatic surface velocity	49
Figure 3.31. Surface velocity versus depth-averaged velocity correlation for (a) low-resolution hydrostatic model and (b) high-resolution non-hydrostatic model	50
Figure 3.32. Comparison of the total water depth: (a) 2D ‘true’ depth, (b) 3D hydrostatic estimated depth, and (c) 3D non-hydrostatic estimated depth	50

Figure 3.33. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for different types of forward model	51
Figure 3.34. The Kootenai River: region overview and bathymetry	52
Figure 3.35. ‘True’ simulation results of Kootenai River: (a) bathymetry, (b) velocity magnitude, (c) water surface elevation with velocity vectors, and (d) total water depth	54
Figure 3.36. Kootenai River comparison of the total water depth for noise-free, full-resolution, depth-averaged velocity data: (a) ‘true’ depth, (b) initial depth, and (c) final estimate of depth	55
Figure 3.37. Kootenai River comparison of the velocity magnitude for noise-free, full-resolution, depth-averaged velocity data: (a) ‘true’ velocity magnitude, (b) initial velocity magnitude, and (c) final estimate of velocity magnitude.....	56
Figure 3.38. Kootenai River statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude.....	57
Figure 3.39. Kootenai River comparison of the total water depth for 20cm/s RMS noise, 20m-resolution, surface velocity data: (a) ‘true’ depth and (b) final estimate of depth.....	58
Figure 3.40. Kootenai River comparison of the velocity magnitude for 20cm/s RMS noise, 20m-resolution, surface velocity data: (a) ‘true’ velocity magnitude and (b) final estimate of velocity magnitude	58
Figure 3.41. Kootenai River statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for 20cm/s RMS noise, 20m-resolution, surface velocity data	59
Figure 3.42. Depth estimation error comparison for varying (a) data resolution and (b) data error.....	61
Figure 3.43. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for different rivers and models	62

NOMENCLATURE

$A_{e,w,s,n}$	off-diagonal coefficients
A_p	diagonal coefficient
$B_{e,w,s,n}$	off-variable coefficients
b_i	i^{th} component of body force vector
C_D	Chézy bottom drag coefficient
c_μ	empirical constant in turbulence model
f_{DD}	diagonal dominance factor
g	acceleration due to gravity
H	total water depth
h	bathymetry
J	cost function
k	turbulent kinetic energy
n	Manning roughness coefficient
p	pressure
S_{bf}	bottom friction source term
t	time
t'	'adjoint' time (after time reversal)
U_i	i^{th} component of fluid depth-averaged velocity vector
u_i	i^{th} component of fluid velocity vector
\tilde{u}_i	i^{th} component of estimate velocity data
$\hat{u}_{i,n}$	i^{th} component of n^{th} observational velocity data
$\overline{u'_i u'_j}$	Reynolds stress tensor
V_i	i^{th} component of 'adjoint' velocity vector
x_i	i^{th} component of coordinate system
\hat{x}_n	observational data location
z	vertical coordinate (x_3)

Greek Symbols

α	‘adjoint’ surface elevation
ΔV	differential volume
Δx_i	grid spacing in i^{th} direction
δ	Dirac delta function
δ_{ij}	Kronecker delta
ε	dissipation of kinetic energy
η	water surface elevation
μ	water dynamic viscosity
ν	water kinematic viscosity
ν_t	turbulent eddy viscosity
ω	relaxation coefficient
ϕ_1	regularization weighting term
ϕ_2	regularization weighting term
ϕ_{DD}	diagonal dominance coefficient
τ_b	bottom friction
τ_{ij}	Newtonian stress tensor

INTRODUCTION

‘Can we navigate the river in our boat?’

‘If so, what is the best path?’

‘If not, how far will we have to carry the boat?’

These are simple questions, but depending on the situation and the data available, the path to the answers could be very complicated. This represents the motivation behind the current research: estimating the depth of a river, given perhaps minimal information about the river in question.

Although the work contained in this dissertation is reduced to the specific problem of estimating river depth from surface velocity measurements, it is important to note that the methodology developed can be generalized to estimating other, perhaps multiple characteristics given different types of data. Ongoing work will focus on applying the methodology to other problems, not limited to specific equations, characteristics or data types.

The problem statement could be framed: ‘If I need to know the spatially-varying depth of a certain reach of a river, what kind of data do I need?’ That information could then be passed on to others that can design their observational collection system(s). The data collection and river depth estimation algorithm can then be coupled together. As a first step towards this goal: How well can we estimate the depth, based on the quantity and quality of the available data?

There are many researchers in multiple disciplines working to better understand rivers. In conducting this research, the author has had the privilege of meeting and interacting with some individuals and groups within this ‘river community’. The numerous strands of research can be

grouped into disciplines, such as biologists, ecologists, physical hydrologists, etc. Similarly, a majority of the work being done can generally be separated into two larger groups: experimentalists and numerical modelers. The topic of this dissertation can be seen as an intermediary between these two, generally within the hydrodynamics and physical oceanography discipline. Data assimilation, in general, is intended to be a way of bridging the gap between observations and models. If one is to effectively generate and use tools that bring together experimental data with numerical modeling tools, one must invest in understanding at least some portion of the two. Understanding the state of the art in both is essential to producing a new ‘product’ that reinforces and multiplies their individual capabilities.

The data discussed in this dissertation is surface velocity. The methodology presented has been found to be stable, robust and accurate given data that covers a significant portion of the river to be studied. At present, the two most viable sources of this data are remote-sensing platforms and instrumented drifters. There is very little data of either of these types that has been made available, and an even smaller subset of those have accompanying full-coverage river depth measurements. More often, researchers have point observations of velocity profiles and depths, which can be used to obtain very accurate estimates of river discharge rates. This dissertation differs from their work in that the discharge rate is not as important to the ‘end-user’ as a map of river depth. Recent advances in remote sensing platforms such as along-track interferometry show promise in the future ability to obtain full-coverage velocity vector data along a river, but each system has different parameters that result in data that is of varying resolution and quality. For an airborne platform, reaching a resolution of 20m with RMS error in velocity of 20cm/s is attainable. Instrumented drifters can offer higher resolution, but there is very little information

about the velocity field near the banks of the river, as they tend to preferentially distribute along deeper channels, where the water is moving more swiftly.

The ‘river community’ has been developing and using numerical tools to simulate river flow for many years. The most basic form of numerical modeling, in traditional computational fluid dynamics, is direct numerical simulation (DNS), where fluid flow is resolved at all time and spatial scales by solving the Navier-Stokes equations. The upper limits of DNS are always increasing as numerical techniques, software and hardware are improving continually. That said, the Reynolds numbers of the rivers in this study are completely unfeasible for DNS at this point. In addition, the effects of vegetation and sediment transport only make the problem more difficult and computationally intensive. In river modeling, it has been found that the dominant effects of turbulence can be captured through the use of an appropriate treatment of the bottom friction; the free stream turbulence is secondary (Nelson and Smith (1989)). Most modelers currently use simple and robust turbulence models (e.g., two-equation k - ϵ models with a standard turbulent eddy viscosity approximation), with acceptable results (Nelson, et al. (2005)).

While it is generally accepted in the community of river modelers that precise modeling of the flow in rivers would be captured best by solving the full Navier-Stokes equations, it has also been found that utilizing the hydrostatic assumption to reduce computational cost and complexity can be done without great loss of fidelity. Some in the community rely almost exclusively on two-dimensional, depth-averaged simulations, utilizing shallow-water equation solvers (Steffler and Blackburn (2002)). Some researchers have devoted resources to the development of secondary flow models (so-called 2.5D models) that include some of the more dominant three-

dimensional flow effects within a two-dimensional framework (Kalkwijk and Booij (1986)). More recently, some groups have turned to three-dimensional hydrostatic solvers in their work. These methods and solvers were originally intended for modeling coastal regions, but in certain situations, accurately capture enough of the physics to aid in the understanding of river flows (Landon, Özkan-Haller and Wilson (2012)). To predict the flow in a river, there exists a minimum set of inputs that can vary depending on the modeling approach. In all approaches, one must know, *a priori*, the river bathymetry (a spatially varying map of the river bottom topography measured downwards from a fixed vertical datum), the river flow rate, some measure of the bottom friction, and some information about the downstream boundary. For the equations to be solved, this represents a sufficient set of conditions to make the problem well posed (assuming constant properties). River flow is inherently transient, but for this work, a steady-state approach has been adopted, and all of the simulations have been conducted such that the river flow has reached a stationary state. The details of the numerical simulations included in this work are presented in appropriate chapters.

In this work, a methodology is presented that relies on the use of the shallow water equations. A derivation of the shallow water equations from the Navier-Stokes equations is included in Appendix A. The method and algorithm require observations of the river surface velocities. The observational data used to drive the algorithm herein is entirely simulated. It is notable, however, that any realistic sources of data (or observations of river velocities) will include all of the physics of water flowing in a river, and thus the approximations of the solvers must be understood. The algorithm has been exercised on multiple sets of data, with each set intended to aid in the understanding of the limits of the algorithm and the effects of the modeling

assumptions. For the generation of the observational data, three methods are used (before down-sampling and/or degrading of the data): two-dimensional depth-averaged hydrostatic solver; three-dimensional hydrostatic solver; and three-dimensional non-hydrostatic solver. The effects of the different modeling approaches to generating observational data are investigated, and the algorithm is shown to be robust and accurate.

The remainder of this dissertation is organized as follows: In Chapter 1, the objectives of the research are described, and a summary of relevant research is given as background and motivation. In Chapter 2, the analytical and numerical methodology is described, with special care taken to describe the derivation of the adjoint equations and solver, as well as the overall assimilation algorithm. Chapter 3 is devoted to the exercising of the model, including studies on the impact of different modeling approaches and both quantity and quality of data. Concluding remarks are included in Chapter 4, with some insights as to the future applications of the algorithm and methodology.

CHAPTER 1

BACKGROUND AND MOTIVATION

1.1 Statement of Objectives

The objective of the research program described in this dissertation is to develop a methodology for the estimation of river characteristics given velocity measurement data, to develop an algorithm to implement that methodology, and to exercise the algorithm to determine the usefulness of the approach in a real-world situation. To that end, there are four critical steps that have been accomplished: 1) establish a mathematical framework for the methodology; 2) develop the numerical tools needed to solve said equations; 3) validate the model/algorithm on ‘perfect’ simulated data; and 4) exercise the model/algorithm on different data sets (types of data, quality of data, different rivers/geometries, e.g.). These steps are described in detail in subsequent sections of this work, but a brief, general description of these objectives is given here:

The first step has been accomplished by developing an adjoint model using variational analysis, starting from the equations that describe the flow of water in rivers. For the second step, an adjoint solver has been developed along with an algorithm that calculates the adjoint variables as well as the gradient, while monitoring the value of the cost function. The most significant part of this step was the development of the adjoint solver; the forward model (Delft3D) is used to solve for the velocity field in the river, given a flow-rate, bathymetry (estimate), and a bottom-friction coefficient. The algorithm and model has been tested on simulated data, generated using D3D for a known bathymetry. This observational data was then used to evaluate the model and algorithm. For the final two steps, various simulated data have been down-sampled and/or degraded (by

adding noise) to fully exercise the model. The algorithm has also been tested on two different rivers.

1.2 Summary of Previous Research

The fidelity of any model is dependent on the fidelity of the model inputs. In order to accurately model the flow of water in a river, one needs to know as much as possible about the river in question. To first order, the most significant characteristics are the river discharge rate and the river depth or bathymetry. Many researchers have spent a great deal of time and effort on how best to obtain accurate estimates for these two variables. There are many experimental methods for assessing these parameters, but most of the methods are expensive and time-consuming. More importantly, they typically require the ability to be in the river. The applications of river modeling are vast, covering a broad range of environmental, commercial and military concerns. With many of these applications, it may be very difficult or impossible for one to have access to the river of interest. Thus, it has become evident that more effort should be devoted to assessing the capabilities of alternative methods in parameter estimation.

The estimation of bathymetry using remotely sensed data is not a new concept. There are researchers around the world investigating this problem. These various groups have different overall aims and requirements. Several groups have investigated estimation of ocean bathymetry on very large scale. This data can be useful in predicting wave propagation and tidal flows, including erosion effects. For example, Bell (1999) used X-Band radar images to estimate near-shore ocean bathymetry, with promising results, providing useful data for tidal estimation applications. On an even larger spatial scale, Sandwell and Smith (2001), used geosat altimeter

data to estimate ocean bathymetry. Their spatial resolution was on the order of 1km, which can be very useful for large-scale modeling. Van Dongeren, et al. (2008) used an assimilation approach to estimate bathymetry, using a wave roller model. The data used were wave celerity from video imagery, or roller dissipation from radar. This methodology shows much promise for nearshore estimation.

Other groups have investigated rivers, in particular. Hildale and Raff (2007) directly measured river bathymetry using LiDAR. Their resolution and estimation errors were excellent, even on small scales, but the LiDAR is mounted on a helicopter, and they take care to note that to get a discernable return from the river bottom, they must have relatively clear water, and can only go to depths of about 10m. Fonstad and Marcus (2005) and Jordan and Fonstad (2005) used remotely sensed imagery in combination with simple 1D channel models with good fidelity, although the bulk parameters such as depth, slope and stream-power are the focus; the cross-stream variation in depth, for example, is not something that would result from their approach. Hyperspectral imagery was shown to be a potentially powerful tool in river bathymetry estimation by Legleiter, et al. (2009). One requirement of this approach is for the signal to be dominated by bottom-reflected radiance.

Other groups have studied large rivers. Durand, et al. (2008) used an assimilation approach to estimate the bathymetry and slope of a 240km reach of the Amazon River. The reported errors are excellent, but the spatial resolution is low (~270m), which is not a problem on such a large river. Similarly, Smith and Pavelsky (2008) investigated another large river, the Lena in Siberia.

They used satellite data covering a 316km heavily braided reach of the river, primarily focused on seasonal variation of discharge rates.

Both data assimilation and model inversion have a wide range of applications. The most notable of these applications is in global (and local) weather prediction. The ability to predict future weather patterns is of great importance to people around the globe, and so a large amount of resources have been devoted to maturing the process. There are huge amounts of data available from the various global entities that use these approaches. Bennett (2002) describes the various techniques and approaches in detail. From a mathematical viewpoint, these methods appear to be suitable to many additional applications. Additionally, Navon (1998) gave a nice summary of the state-of-the-art in adjoint parameter estimation.

Wang, et al. (1992) used a very similar methodology to that proposed for this work. They used variational data assimilation (VDA) to estimate the initial conditions for atmospheric predictions. They presented some interesting findings with regards to the sensitivity of the convergence of the cost function reduction to various perturbations. For example they discussed how the cost function is highly sensitive to errors in data where the most intensive events occur. (I also found it interesting that the shallow-water equations (SWEs) are used for atmospheric modeling.) Li, et al. (1993) applied VDA to global geopotential modeling to estimate large-scale ocean currents using a semi-Lagrangian SWE model. Other groups have contributed to oceanographic predictions (e.g. Bertino (2003) and Egbert and Erofeeva (2002)).

The use of variational methodology is not limited to environmental modeling. These techniques have been successfully applied to other physical phenomena, such as electro-magnetic wave propagation (using the Maxwell's equations). Rekanos and Räisänen (2003) applied inverse modeling to find buried scatterers. They solved Maxwell's equations using a finite-difference time-domain (FDTD) method and formulated the inverse model with Lagrange multipliers, using a similar methodology as presented in this work. They were able to successfully reconstruct an accurate permittivity field using seven transmitters/receivers (5% RMS error in permittivity over the field).

More recently, a different approach to estimating river bathymetry from surface velocities has been pursued by Landon, Özkan-Haller and Wilson (2012) at Oregon State University. They have implemented a Kálmán filtering approach that uses a covariance matrix of velocity and bathymetry generated by multiple ensemble simulations of a river using the Regional Ocean Modeling System (ROMS). Their initial results are promising. However, the approach is dependent on the number of ensembles, and remains computationally expensive.

The subject of the current research is to take these wide-ranging efforts and applications and apply them in a new way, developing a methodology to estimate river characteristics (specifically bathymetry) using remotely sensed data. Specific requirements on the accuracy of the algorithm have not yet been established, though the values reported in this dissertation have been accepted thus far as being sufficiently accurate for the potential end-users' needs; the purpose of this research is to establish a capability. The algorithm development is complete, and the modeling framework is in place. The algorithm has been exercised on multiple data sets,

using two different rivers. The details of the models, algorithms, rivers and results are presented in detail in subsequent chapters.

CHAPTER 2

GOVERNING EQUATIONS AND METHODOLOGY

2.1 The Forward Model

The shallow water equations (SWEs), including the effects of bottom roughness can be used to estimate the flow of water in a river in a two-dimensional, depth-averaged sense. The vertical coordinate system with regards to the water column is shown in Figure 2.1.

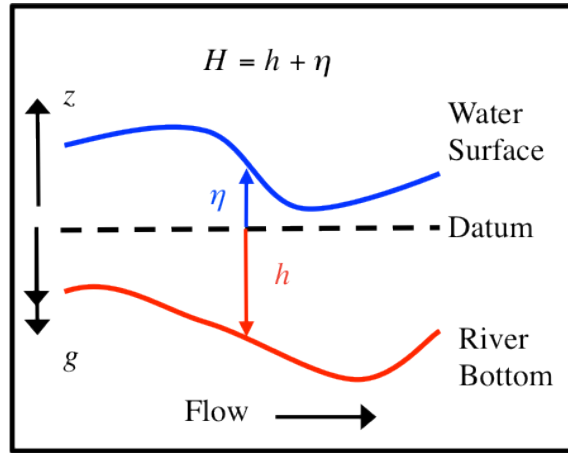


Figure 2.1. Vertical coordinate system. For interpretation of the references to color in this and all other figures, the reader is referred to the electronic version of this dissertation.

A derivation of the SWEs is given in Appendix A. Throughout this dissertation, tensor notation is used; repeated indices represent a summation over the indices. The equations for water surface elevation (continuity) and velocity components (momentum) are

$$\frac{\partial \eta}{\partial t} + \frac{\partial (HU_j)}{\partial x_j} = 0 \quad \text{and} \quad (2.1)$$

$$\frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j} = -g \frac{\partial \eta}{\partial x_i} - gn^2 U_i \frac{(U_j U_j)^{\frac{1}{2}}}{H^{\frac{4}{3}}}, \quad (2.2)$$

where the total water depth H is defined as the sum of the water surface elevation (measured upwards from a vertical datum) and the bathymetry (measured downward from the same datum), $H = \eta + h$. The last term in the momentum equation arise from the momentum loss due to bottom friction, using the Chézy formulation

$$\tau_{b,i} = \frac{\rho g U_i (U_j U_j)^{\frac{1}{2}}}{C_D^2}. \quad (2.3)$$

The Chézy bottom roughness coefficient, C_D , is related to the Manning roughness coefficient, n , through

$$C_D = \frac{H^{\frac{1}{6}}}{n}. \quad (2.4)$$

2.2 Variational Framework

The parameter estimation begins with the cost function, or penalty functional, which is constructed using the error in the observational versus modeled data

$$J = \sum_n \left\{ \left(\tilde{u}_j - \hat{u}_{j,n} \right) \left(\tilde{u}_j - \hat{u}_{j,n} \right) \right\}, \quad (2.5)$$

where $\hat{u}_{j,n}$ are the observed velocities (surface or depth-averaged), and \tilde{u}_j represent the model estimated velocities, interpolated to the observation locations.

We wish to minimize this measure of the error between the modeled and observed data. To do this, the cost function is augmented through the use of Lagrange multipliers to constrain the solution such that the model equations are satisfied, leading to

$$\begin{aligned}
J = \iiint \left[\sum_n \left\{ \left(\tilde{u}_j - \hat{u}_{j,n} \right) \left(\tilde{u}_j - \hat{u}_{j,n} \right) \right\} \delta(x - \hat{x}_n) \right. \\
+ \alpha \left\{ \frac{\partial \eta}{\partial t} + \frac{\partial(HU_j)}{\partial x_j} \right\} \\
\left. + V_i \left\{ \frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j} + g \frac{\partial \eta}{\partial x_i} + gn^2 U_i \frac{(U_j U_j)^{\frac{1}{2}}}{H^{\frac{4}{3}}} \right\} \right] dx_1 dx_2 dt.
\end{aligned} \tag{2.6}$$

To ensure the uniqueness of the bathymetry estimate, regularization should be included in the construction of the cost function:

$$\begin{aligned}
J = \iiint \left[\sum_n \left\{ \left(\tilde{u}_j - \hat{u}_{j,n} \right) \left(\tilde{u}_j - \hat{u}_{j,n} \right) \right\} \delta(x - \hat{x}_n) \right. \\
+ \alpha \left\{ \frac{\partial \eta}{\partial t} + \frac{\partial(HU_j)}{\partial x_j} \right\} \\
+ V_i \left\{ \frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j} + g \frac{\partial \eta}{\partial x_i} + gn^2 U_i \frac{(U_j U_j)^{\frac{1}{2}}}{H^{\frac{4}{3}}} \right\} \\
\left. + \phi_1 h^2 + \phi_2 \frac{1}{2} \left\{ \left(\frac{\partial h}{\partial x_j} \right) \left(\frac{\partial h}{\partial x_j} \right) \right\} \right] dx_1 dx_2 dt.
\end{aligned} \tag{2.7}$$

For this work, the regularization terms have been set to zero, with minimal effect. A necessary condition for the minimum of J is that the first variations with respect to each of the functional variables are zero. To minimize this cost function, we first find the gradient of the cost function with respect to each of the variables/parameters concerned. At the minimum, all components of the gradient must vanish. Taking the first variation of the cost function with respect to η , U_i , α , V_i and h and setting them equal to zero does this. It is clear that the original model equations are

recovered when taking the first variation of J with respect to α and V_i . The first variation of J with respect to η leads to

$$\delta J_\eta = \iiint \left[\alpha \left\{ \frac{\partial \delta \eta}{\partial t} + \frac{\partial (\delta \eta U_j)}{\partial x_j} \right\} + V_i \left\{ g \frac{\partial \delta \eta}{\partial x_i} - \frac{4}{3} g n^2 U_i \frac{(U_j U_j)^{\frac{1}{2}}}{H^{\frac{7}{3}}} \delta \eta \right\} \right] dx_1 dx_2 dt. \quad (2.8)$$

Setting this equal to zero and integrating by parts results in:

$$0 = \iiint \left[\left\{ \frac{\partial (\alpha \delta \eta)}{\partial t} - \frac{\partial \alpha}{\partial t} \delta \eta + \frac{\partial (\alpha U_j \delta \eta)}{\partial x_j} - \frac{\partial \alpha}{\partial x_j} U_j \delta \eta \right\} + \left\{ g \frac{\partial (V_i \delta \eta)}{\partial x_i} - g \frac{\partial V_i}{\partial x_i} \delta \eta - \frac{4}{3} g n^2 V_i U_i \frac{(U_j U_j)^{\frac{1}{2}}}{H^{\frac{7}{3}}} \delta \eta \right\} \right] dx_1 dx_2 dt. \quad (2.9)$$

Since the integral is over an arbitrary region, the terms in the integrand must vanish independently. Since $\delta \eta$ is arbitrary, the collection of terms multiplying it must go to zero. This yields the adjoint form of the continuity equation:

$$-\frac{\partial \alpha}{\partial t} - \frac{\partial \alpha}{\partial x_j} U_j = g \frac{\partial V_i}{\partial x_i} + \frac{4}{3} g n^2 \frac{(U_j U_j)^{\frac{1}{2}}}{H^{\frac{7}{3}}} (V_i U_i). \quad (2.10)$$

The other terms that are not proportional to $\delta \eta$ involve the boundary and initial conditions. These must vanish as well, and are discussed in more detail (as a full set for both α and V_i) below.

Finally, we apply a formal time reversal by substituting in $t' = -t$:

$$\frac{\partial \alpha}{\partial t'} + (-U_j) \frac{\partial \alpha}{\partial x_j} = g \frac{\partial V_i}{\partial x_i} + \frac{4}{3} g n^2 \frac{(U_j U_j)^{\frac{1}{2}}}{H^{\frac{7}{3}}} (V_i U_i). \quad (2.11)$$

Applying the same methodology in taking the first variation of J with respect to U_i results in the ‘adjoint momentum’ equation:

$$\begin{aligned} \frac{\partial V_i}{\partial t'} + (-U_j) \frac{\partial V_i}{\partial x_j} = H \frac{\partial \alpha}{\partial x_i} - \left(V_j \frac{\partial U_j}{\partial x_i} - V_i \frac{\partial U_j}{\partial x_j} \right) \\ - 2 \sum_n \left\{ (\tilde{u}_i - \hat{u}_{i,n}) \right\} \delta(x - \hat{x}_n) - \frac{g n^2}{H^{\frac{7}{3}}} \left[V_i (U_j U_j)^{\frac{1}{2}} + \frac{U_i}{(U_j U_j)^{\frac{1}{2}}} (V_j U_j) \right]. \end{aligned} \quad (2.12)$$

For brevity, the last term on the RHS in each of the equations (2.11) and (2.12) will be denoted $S_{bf,\alpha}$ and $S_{bf,v}$, respectively.

Establishing the correct boundary conditions is critical to obtaining an accurate adjoint solution, both for open boundaries (inflow and outflow), and for closed (land-water) boundaries. Analysis of the variational equations (2.9, e.g.) at the boundaries leads to the following contributions to the cost function:

$$\delta J_\eta = \iiint \left[\frac{\partial(\alpha \delta \eta)}{\partial t} + \frac{\partial(\alpha U_j \delta \eta)}{\partial x_j} + g \frac{\partial(V_j \delta \eta)}{\partial x_j} \right] dx_1 dx_2 dt \quad \text{and} \quad (2.13)$$

$$\delta J_{U_i} = \iiint \left[\frac{\partial(V_i \delta U_i)}{\partial t} + \frac{\partial(\alpha H \delta U_i)}{\partial x_i} + \frac{\partial(V_i U_j \delta U_i)}{\partial x_j} \right] dx_1 dx_2 dt. \quad (2.14)$$

Again, the choice of $\delta\eta$ and δU_i are arbitrary. Using the appropriate forward model boundary conditions (full-slip for cases where the boundary layer cannot be resolved, as in most geophysical simulations),

$$\frac{\partial U_t}{\partial x_n} = 0; \quad U_n = 0; \quad \frac{\partial \eta}{\partial x_n} = 0, \quad (2.15)$$

together with the above derived equations at a land-water boundary, reduces the boundary conditions for the adjoint variables to:

$$\frac{\partial V_t}{\partial x_n} = 0; \quad V_n = 0; \quad \frac{\partial \alpha}{\partial x_n} = 0. \quad (2.16)$$

For the estimation of bathymetry, the first variation of J with respect to h is calculated, leading to

$$\begin{aligned} \delta J_h = \iiint \left[\left\{ \frac{\partial(\alpha U_j \delta h)}{\partial x_j} - \frac{\partial \alpha}{\partial x_j} U_j \delta h - S_{bf,\alpha} \delta h \right. \right. \\ \left. \left. + \phi_1 h \delta h - \phi_2 \left(\frac{\partial^2 h}{\partial x_j \partial x_j} \right) \delta h \right\} dx_1 dx_2 dt. \right] \end{aligned} \quad (2.17)$$

It can be shown that in order to reduce the value of the cost function, the gradient of h , δh , should be proportional to and opposite in sign to the sum of its coefficients:

$$\delta h \propto U_j \frac{\partial \alpha}{\partial x_j} + S_{bf,\alpha} - \phi_1 h + \phi_2 \left(\frac{\partial^2 h}{\partial x_j \partial x_j} \right). \quad (2.18)$$

2.3 The Adjoint Solver

The adjoint equations can be interpreted as advection equations with source terms involving gradients of the variables. In the formulation of the adjoint numerical scheme, it was determined

that using a finite volume formulation would lend itself to better handling these equations/source terms. To do so, continuity is applied to equations (2.11) and (2.12) which results in the final conservative form of the equations:

$$\frac{\partial H\alpha}{\partial t'} + \frac{\partial H(-U_j)\alpha}{\partial x_j} = gH \frac{\partial V_i}{\partial x_i} + HS_{bf,\alpha} \text{ and} \quad (2.19)$$

$$\begin{aligned} \frac{\partial HV_i}{\partial t'} + \frac{\partial H(-U_j)V_i}{\partial x_j} = & H^2 \frac{\partial \alpha}{\partial x_i} \\ & - H \left(V_j \frac{\partial U_j}{\partial x_i} - V_i \frac{\partial U_j}{\partial x_j} \right) - 2H \sum_n \left\{ (\tilde{u}_i - \hat{u}_{i,n}) \right\} \delta(x - \hat{x}_n) - HS_{bf,v}. \end{aligned} \quad (2.20)$$

To solve the adjoint equations, a new finite-volume solver has been developed (ADJ2DI). The solver is fully implicit, utilizing a Gauss-Seidel scheme. It can run on an arbitrary (nearly) orthogonal, curvilinear grid, and can be run in parallel on multiple processors. The forward model, D3D utilizes an alternating direction implicit method, but this method proved to be unstable for the adjoint equations, due to the nature of the source terms. In the new solver, at each point the equations are fully coupled by solving for the three variables successively, as opposed to solving for the entire field for each variable before moving on to the next variable field. This was shown to be necessary, as the first-order coupling between adjoint variables is through gradient and divergence operators.

Some modifications to the ADJ2DI solver were necessary to obtain a solution in low velocity regions. There are two important facets to the solver, one is to ensure diagonal dominance of the matrix equations, and the other is to deal with decoupling of the equations in low velocity

regions. To ensure the diagonal dominance (of the full solver matrix, including α and V_l), a ‘pseudo-variable time stepping scheme’ was implemented. At each point, the matrix coefficients consist of the typical advection diagonal and off-diagonal terms. In addition, there are off-diagonal terms due to the coupling between variables. It turns out that these ‘off-variable’ matrix coefficients drive the diagonal dominance and the stability of the solver. To circumvent this problem, an additional (spatially-varying) term was added to the diagonal term. This term, ϕ_{DD} , is calculated by including all of the coefficients on all of the variables in the model equations for a spatial location. We multiply a ‘diagonal dominance factor’, f_{DD} , (currently a value of 2 is used) by the sum of the absolute values of the off-diagonal (and off-variable) coefficients and subtract the absolute value of the diagonal coefficient. This is done for each of the variables, and at each spatial location, following the general structure:

$$\phi_{DD} \approx \frac{1}{dt} = f_{DD} \left(\sum_l |A_l| + \sum_l |B_l| \right) - |A_p|. \quad (2.21)$$

Here the A_l coefficients represent the advection operator (l is the index over neighboring cells, N, S, E and W for a five point computational molecule) and the B_l coefficients represent the off-diagonal coefficients. When solving for the variable, the ϕ_{DD} term is added to the diagonal term, and an additional term appears on the right-hand side of the equation multiplied by the previous value for the variable at that location:

$$\alpha_p^{n+1} = \frac{Q_{p,\alpha} - \sum_l A_l \alpha_l + \phi_{DD,\alpha} \alpha_p^n}{A_p + \phi_{DD,\alpha}}, \text{ with} \quad (2.22)$$

$$Q_{p,\alpha} = g \frac{\partial V_1}{\partial x_1} + g \frac{\partial V_2}{\partial x_2} + S_{bf,\alpha}. \quad (2.23)$$

Here, $Q_{p,\alpha}$ includes the entire right-hand side of the α equation. It is clear that in the limit of a steady-state solution, the terms involving the ϕ_{DD} drop out.

The other important modification to the solver is in the treatment of low velocity regions. Here, an approach is described that is similar in nature to the SIMPLE algorithm that is used to calculate the pressure correction to ensure continuity in some Navier-Stokes solvers. In the SIMPLE algorithm, the velocity and pressure corrections are related using the discrete equations for momentum and continuity, such that the pressure correction can be written in terms of the velocity field (see Patankar (1981) and Ferziger and Perić (2002)). The shared methodology is that we wish to write the α equation solely in terms of α (and similarly for V_i). Even with the diagonal dominance term included, when the velocity approaches zero, the equations de-couple to the point that the α equation does not contain any terms that have α in them. In these regions, the source terms drive the solution to diverge. For this reason, a numerical approximation was included to make these source terms implicit within the larger solution matrix. This results in the gradient of α being essentially replaced with a Laplacian operator on V_i in the V_i equation, and a similar substitution for the divergence of V_i results in a Laplacian operator on α in the α equation.

For clarity, here the tensor notation is expanded into component form. The numerical approximation for α is given in equations (2.22) and (2.23), and the approximation for V_I is as follows:

$$V_{1,p}^{n+1} = \frac{Q_{p,v1} - \sum_l A_l V_{1,l} + \phi_{DD,v1} V_{1,p}^n}{A_{p,v1} + \phi_{DD,v1}}, \text{ with} \quad (2.24)$$

$$Q_{p,v1} = H \frac{\partial \alpha}{\partial x_1} - V_2 \frac{\partial U_2}{\partial x_1} + V_1 \frac{\partial U_2}{\partial x_2} + 2 \sum_n (\tilde{u}_1 - \hat{u}_{1,n}) \delta(x - \hat{x}_n) + S_{bf,v1}. \quad (2.25)$$

To make the source terms in the α equation implicit, we separate the source terms in the V_I equation (and similarly the V_2 equation) into terms involving α and the rest, denoted with a tilda.

$$\tilde{Q}_{p,v1} = -V_2 \frac{\partial U_2}{\partial x_1} + V_1 \frac{\partial U_2}{\partial x_2} + 2 \sum_n (\tilde{u}_1 - \hat{u}_{1,n}) \delta(x - \hat{x}_n) + S_{bf,v1}. \quad (2.26)$$

Then the RHS of α can be modified as:

$$\begin{aligned} Q_{p,\alpha} &= g \frac{\partial V_1}{\partial x_1} + \dots \\ &= g \frac{\partial}{\partial x_1} \left(\frac{H \frac{\partial \alpha}{\partial x_1}}{A_{p,v1} + \phi_{DD,v1}} \right) + g \frac{\partial}{\partial x_1} \left(\frac{\tilde{Q}_{p,v1} - \sum_l A_l V_{1,l} + \phi_{DD,v1} V_{1,p}^n}{A_{p,v1} + \phi_{DD,v1}} \right) + \dots \end{aligned} \quad (2.27)$$

Finally, the terms with α in them can be converted into off-diagonal coefficients on α ($B_{l,\alpha}$), while the rest of $Q_{p,\alpha}$ can be calculated and used as a ‘typical’ source term. Similar numerical approximations making the coupling source terms implicit and circumventing the low-velocity issue are applied to all variables. As with the SIMPLE algorithm, in the limit of a converged solution, the approximation becomes exact.

2.4 The Assimilation Algorithm

The final solution is of the form of an estimated bathymetry, along with a forward solution using that bathymetry to obtain the flow field. To find the bathymetry that minimizes the cost function, we utilize a gradient descent algorithm. A simplified block diagram of the algorithm is shown in Figure 2.2, and the full algorithm, as implemented, is outlined as a flow chart in Figure 2.3, with a short description following.

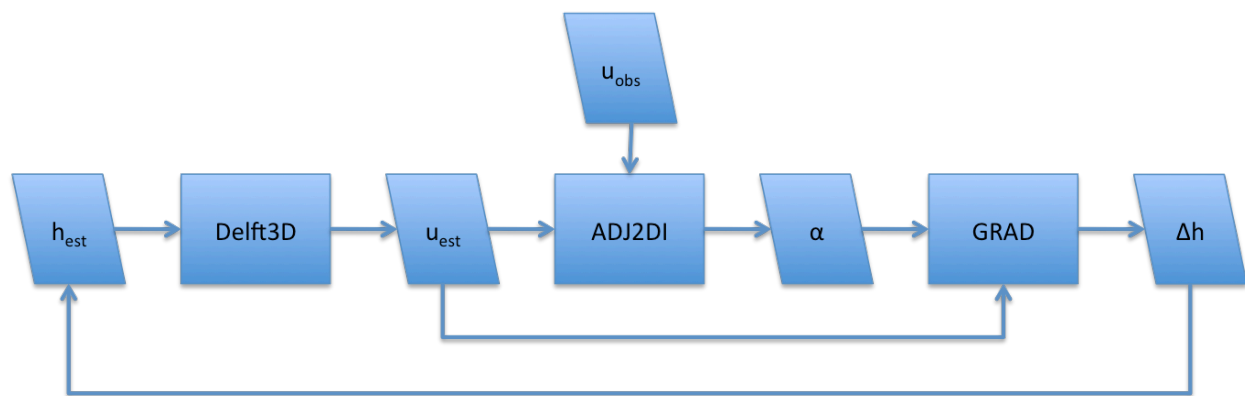


Figure 2.2. Simplified algorithm flow chart.

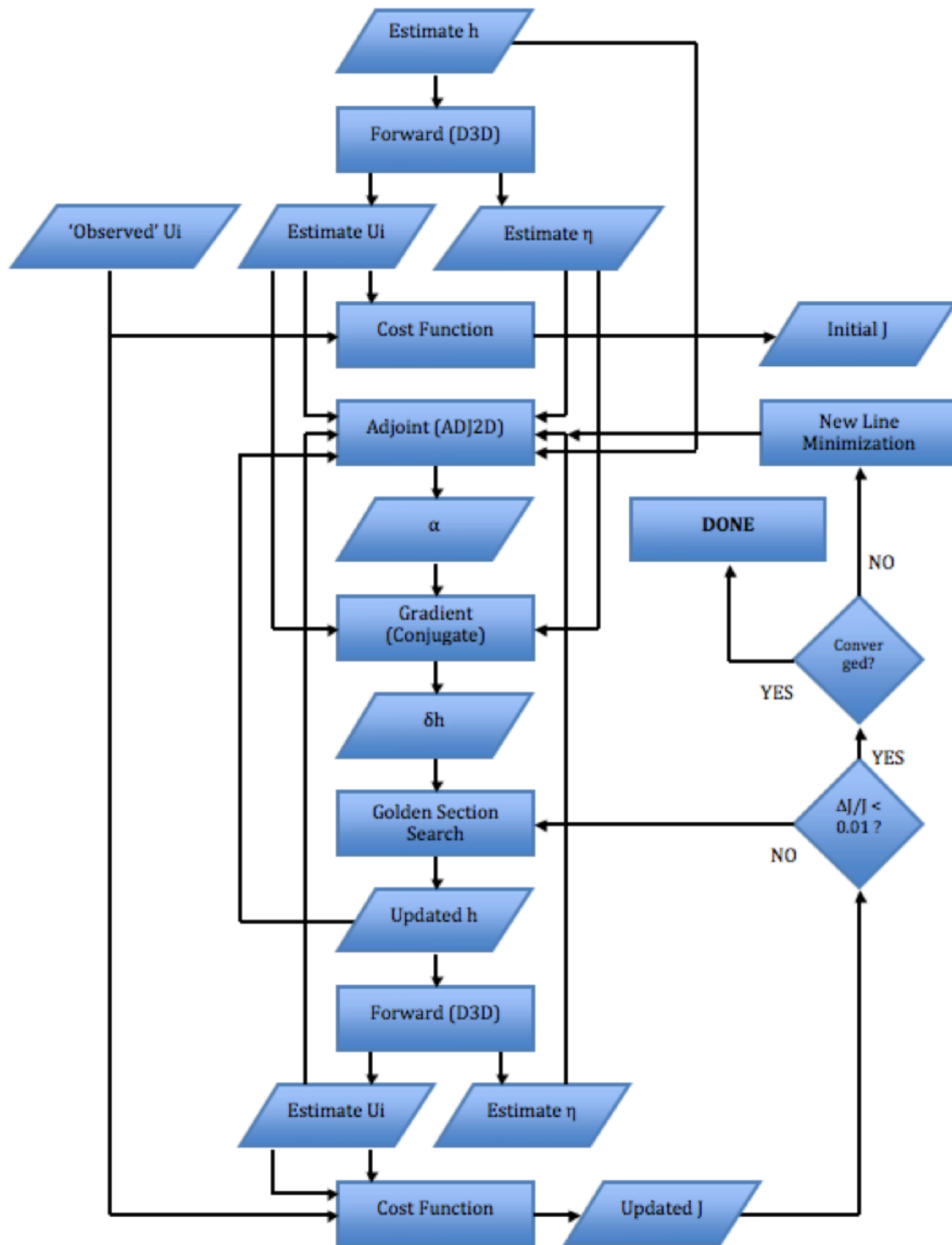


Figure 2.3. Algorithm flow chart.

First, an initial guess bathymetry is generated using knowledge of the land-water boundary of the river. Then an initial forward simulation is run. Using the resultant flow field, the adjoint equations are then solved using the error between the predictions and observations as input. At this point, the forward and adjoint results are used to calculate the gradient of the cost function with respect to the bathymetry. This gradient is used to adjust the bathymetry to reduce the error in the computed flow. A correction proportional to the gradient is added to the present bathymetry estimate. A golden section search is used to determine the scale factor for the correction to the bathymetry, i.e. how far to go in the direction indicated by the gradient. When the minimum of the cost function is obtained using this gradient, a new adjoint solution is obtained using the new bathymetry and most recent forward solution. A new gradient with respect to h is calculated, and the golden section search is repeated. This new gradient could be combined with the previous gradient, if using a conjugate gradient method, which can increase rate of convergence in some cases. The entire process is repeated until the cost function has reached an 'absolute' minimum.

CHAPTER 3

RESULTS AND DISCUSSION

The goal of the project under which this research is funded is to develop methods for river characterization from remote sensing data. The objectives are stated in Chapter 1. The analytical and algorithmic framework was discussed in Chapter 2, and the focus of this chapter is on implementing and exercising the algorithm. While the data is to be surface velocity measurements, it is important to understand the types of data that may be available in real applications. The first type might come from optical imagery on an airborne platform. The data most likely will have some kind of error associated with it. Therefore, it is important to investigate the following questions: what is the best estimate possible, assuming perfect full-field (depth-averaged or surface) velocity data; how does the reduction of quantity and/or quality of data affect the ability to make a prediction; and what are the implications of having surface data versus, for example, depth-averaged data. It is also important to understand any potential effects of modeling assumptions.

The project began with some simple river simulations. Several models were compared and tested. The decision was made to move forward building the assimilation and inverse modeling capability around the forward simulation tool, Delft3D. This tool was developed, delivered and maintained by Deltares in the Netherlands. The source code has recently been released as open-source, and we have downloaded and installed to the latest version of the model. Delft3D is primarily an ocean circulation simulation tool that can be setup to run in two-dimensions (depth-averaged) or three-dimensions using a so-called terrain following boundary-fitted methodology utilizing a sigma coordinate system. The model is run on a staggered Arakawa-C grid using an

alternating direction implicit time stepping scheme. For more details on the numerical aspects of the Delft3D solver, see Deltares (2011).

3.1 Snohomish River

3.1.1 Geographical Context

The first simulations conducted under the project were for both the Skagit River and the Snohomish River in Washington State. The results presented here are from simulations of the Snohomish River. The portion of the Snohomish River that is currently being investigated is a 2.2 km reach characterized by a sharp bend in a meandering section of the river. The average width of the river in this reach is about 120m. Chris Chickadel at the Applied Physics Laboratory at the University of Washington provided the bathymetry. The known bathymetry of the river has many interesting features, including significant bathymetric variation in the form of deep pools and smaller bars. The region of interest is shown in Figure 3.1. Note that in Figure 3.1 and subsequent two-dimensional plots of the Snohomish River, the axes labels are omitted for space.

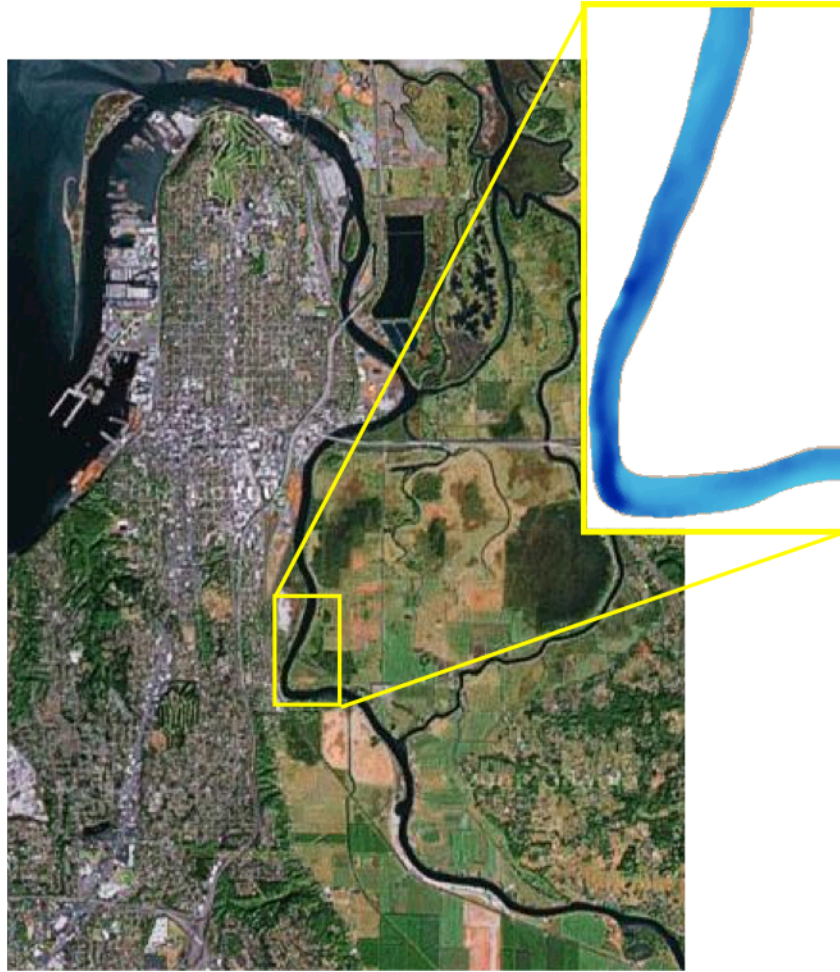


Figure 3.1. The Snohomish River: region overview and bathymetry. This reach is approximately 2.2km long with an average width of 120m. The sharp bend in the river is located at 122.191°W , 47.948°N . In this figure and subsequent two-dimensional figures, the axes labels are omitted for space.

The Snohomish River has been deemed an excellent region for exercising our algorithm development, given the available data. Using simulated data to conduct the first tests has been helpful in previous work. The bathymetry, as received is on a 8101×3336 grid with a spatial resolution of 25cm. For initial testing, this has been down sampled to a boundary fitted, orthogonal curvilinear grid with an average resolution of 3m (722×35 cells in the downstream and cross-stream directions, respectively).

The original simulations aided in the understanding of the benefits and limitations of the models. Several investigations were conducted to test different handling of the boundary conditions for river flows. For the testing of the algorithm, a steady-state, depth-averaged flow field was desired. To that end, the upstream/inflow forward boundary condition was set to a relatively low total volumetric flow rate of $50 \text{ m}^3/\text{s}$, and the effects of bottom friction were included, using a Manning roughness coefficient of $0.05 \text{ s/m}^{1/3}$. The downstream/outflow forward boundary condition that proved to have the least impact on the velocity and water depth was that of a Neumann BC, with $d\eta/dy = -0.0001$. The domain was also extended (4 km) to the North to ensure that the boundary condition had a minimal impact on the region of interest. The results from a Delft3D simulation of the Snohomish River using the ‘true’ bathymetry are shown in Figure 3.2. The inlet is in the southeast corner, and the outlet is at the north. The forward simulation is run for 12 hours, with a time step of 4 seconds. The forward simulations require 2 minutes on 8 CPU.

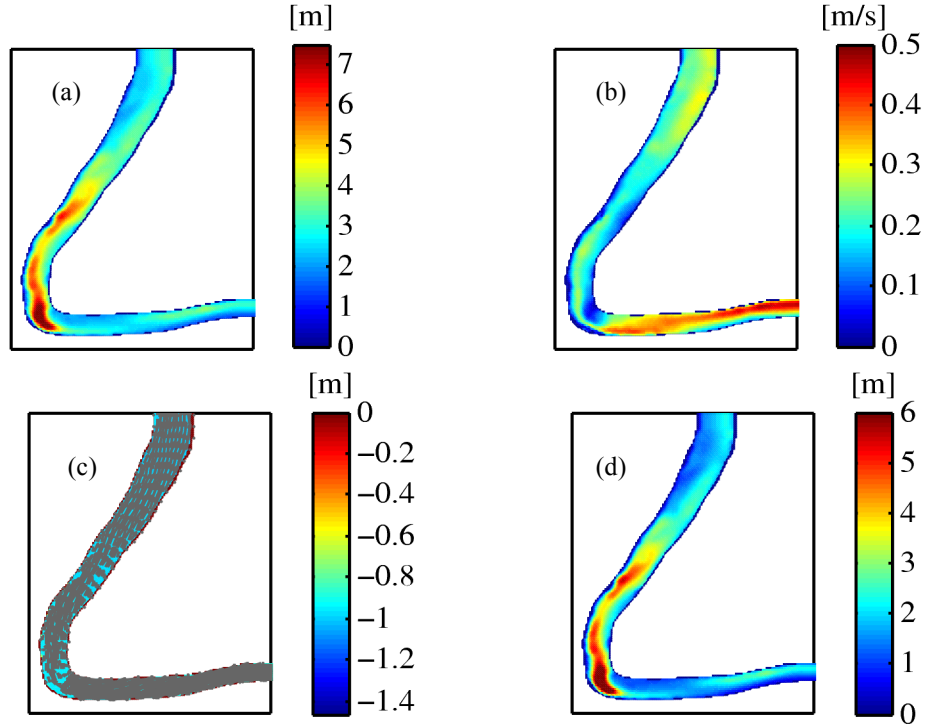


Figure 3.2. ‘True’ simulation results of Snohomish River: (a) bathymetry, (b) velocity magnitude, (c) water surface elevation with velocity vectors, and (d) total water depth.

3.1.2 Depth-Averaged (2D) Velocity Data

After determining the best forward model setup, the initial testing of the adjoint solver and assimilation algorithm given fully consistent, full-field depth-averaged velocity data began. This is an important step in testing the algorithm and provides an upper limit on the performance of the method; while the final data will not be depth-averaged data, the underlying hydrodynamics are being represented as such. This will allow for the separation of errors due to methodology and those due to model assumptions. There are three sub-sections to isolate and compare the effects of data resolution and noise, given data that is consistent with the model.

3.1.2.1 Full-Field Data

The best possible performance of the algorithm is for data that is at the model resolution with no noise and consistent with the hydrodynamics (depth-averaged). It is important to note that the algorithm can predict bathymetry only to within an additive constant; to remove this ambiguity, total water depth is used for all comparisons, unless noted otherwise. This is also the variable of importance for the potential end-users. A comparison of the ‘true’, first guess, and final depth is shown in Figure 3.3. A similar comparison for velocity magnitude is shown in Figure 3.4.

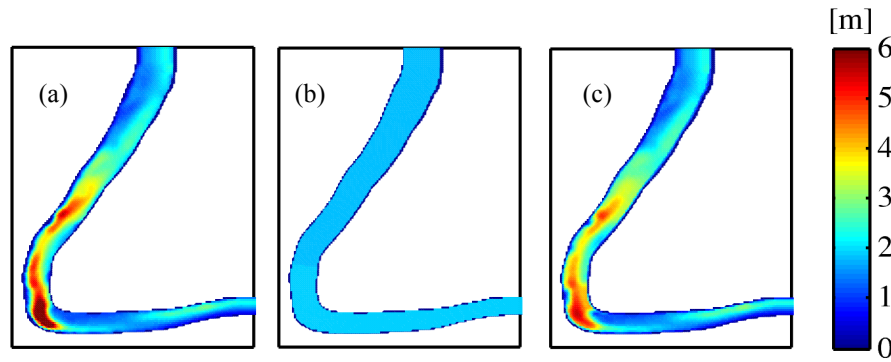


Figure 3.3. Comparison of the total water depth for noise-free, full-resolution, depth-averaged velocity data: (a) ‘true’ depth, (b) initial depth, and (c) final estimate of depth.

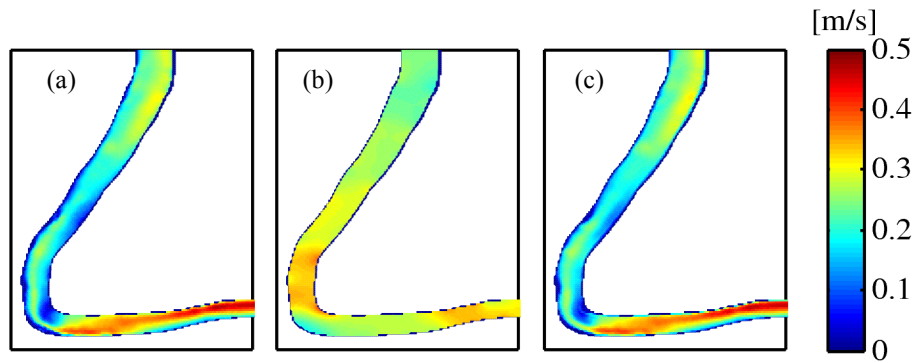


Figure 3.4. Comparison of the velocity magnitude for noise-free, full-resolution, depth-averaged velocity data: (a) ‘true’ velocity magnitude, (b) initial velocity magnitude, and (c) final estimate of velocity magnitude.

After 30 line minimizations, the cost function, a measure of the error variance for the estimated velocity field, shows a decrease of about 99%, as seen in Figure 3.5. The RMS velocity error is reduced to 0.011 m/s, and the depth error is reduced to 0.38 m.

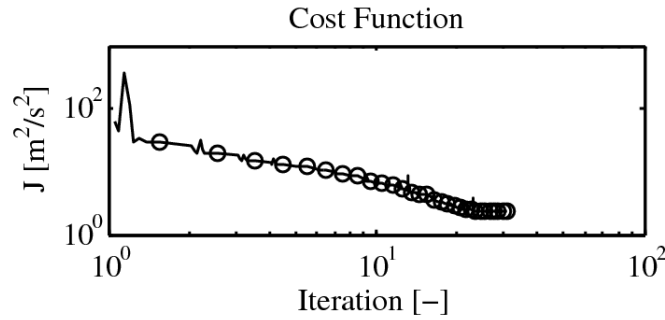


Figure 3.5. Cost function progression.

A point-by-point comparison of the estimated versus simulated ‘true’ results is shown in Figure 3.6. A similar plot is shown in Figure 3.7. Here, the ‘true’ variables are put into 20 equally spaced bins, and both the estimated and ‘true’ are averaged over the ‘true’ bins. Inspection of these results show excellent velocity agreement, while the total water depth show a small bias for shallow water; the algorithm over-predicts the shallow depths. In contrast, the algorithm accuracy decreases with increasing depth, consistently under-estimating depths beyond 3 m. In Figure 3.7, the colors represent the number of data points in each statistical bin; note that there are a relatively small number of data points in the deeper regions of the river.

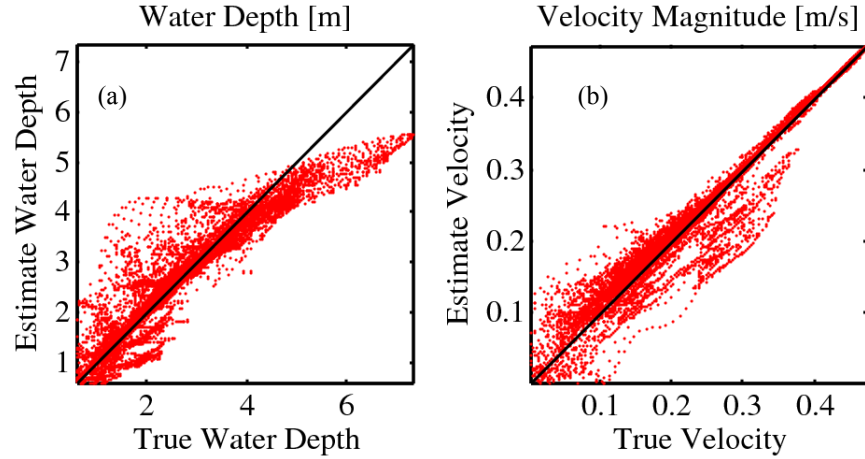


Figure 3.6. Point-by-point comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for noise-free, full-resolution, depth-averaged velocity data.

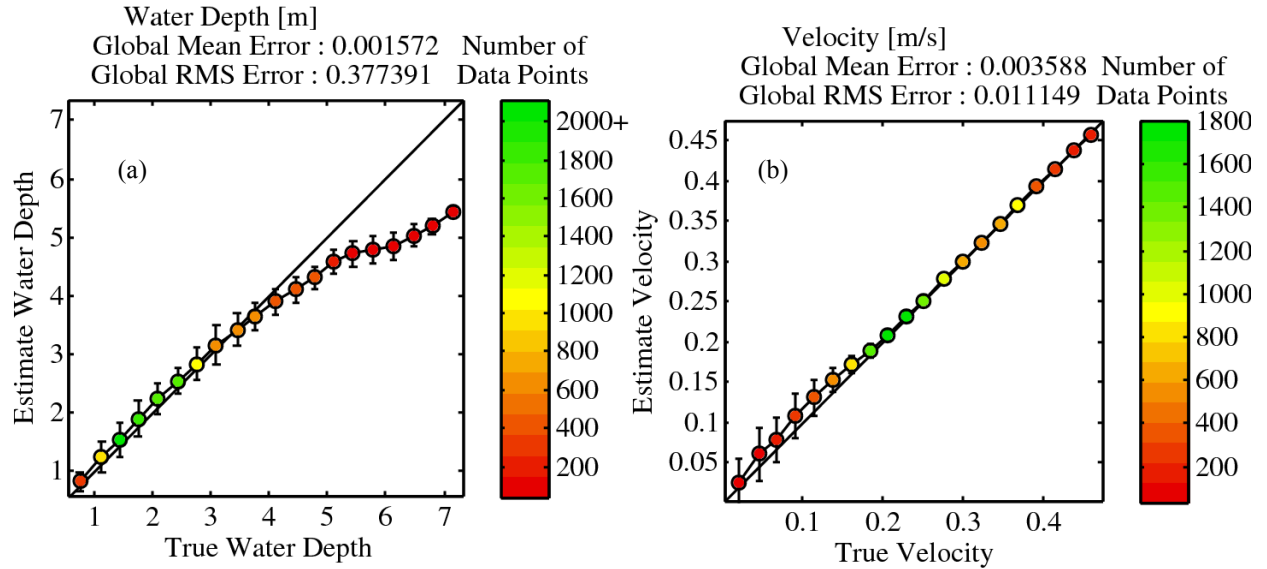


Figure 3.7. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for noise-free, full-resolution, depth-averaged velocity data. The error bars represent one standard deviation of the error between estimated and ‘true’ for each bin of the ‘true’ variable, and the colors represent the number of data points in each statistical bin.

Velocity comparisons versus ‘true’ total water depth are given in Figure 3.8, and depth comparisons versus ‘true’ velocity magnitude are shown in Figure 3.9. There is no strong correlation between water depth and velocity, as shown in Figures 3.8(a) and 3.9(a). This is attributed to the competing effects of continuity and bottom friction. Upon inspection of the

errors in the estimates, Figure 3.8(b) shows that the largest velocity errors are in the deepest water, while in Figure 3.9(b) the largest depth errors are in the low-velocity regions.

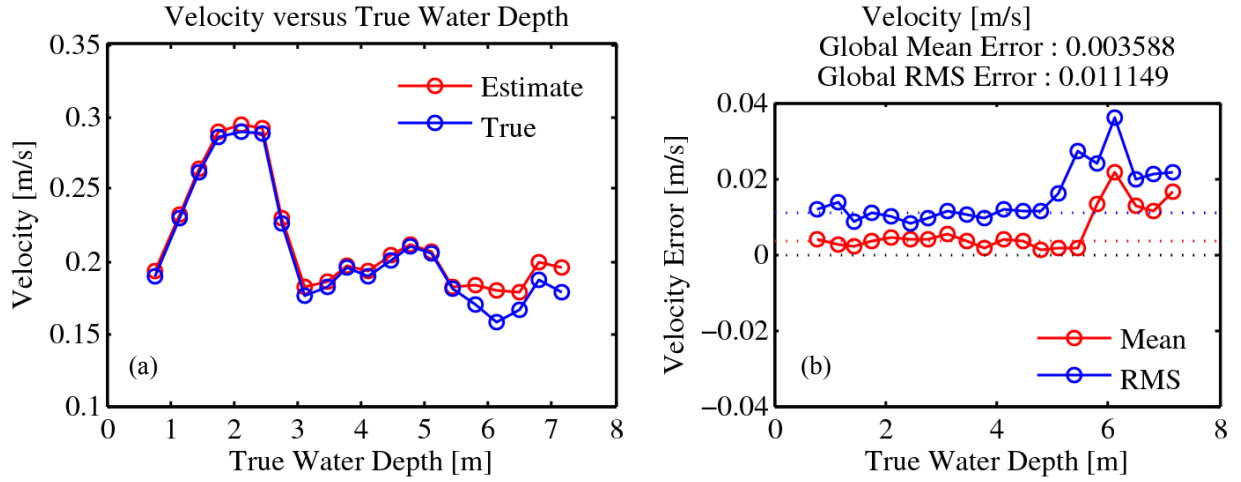


Figure 3.8. Velocity magnitude statistics versus total water depth for noise-free, full-resolution, depth-averaged velocity data.

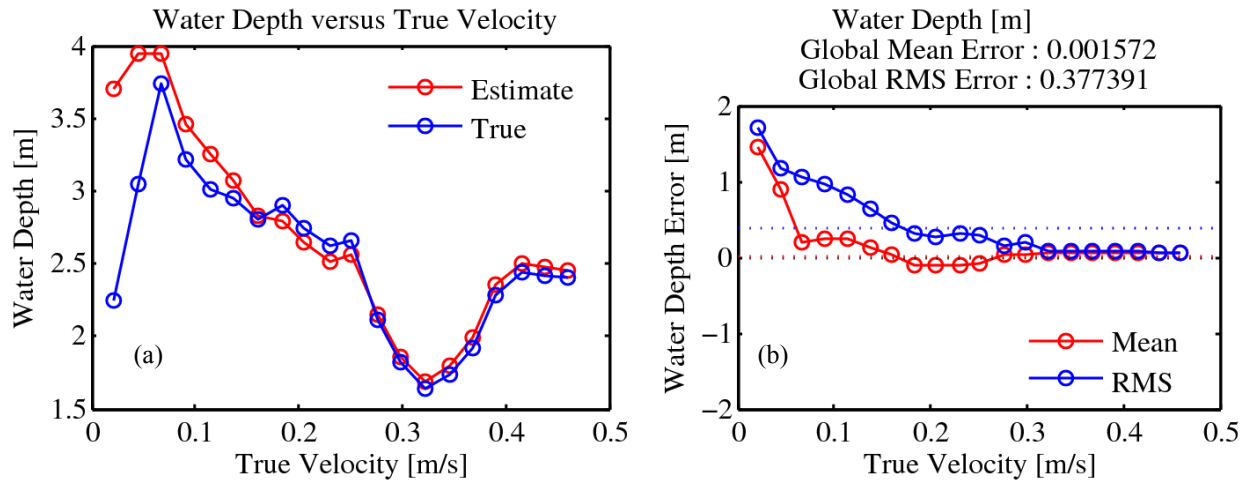


Figure 3.9 Total water depth statistics versus velocity magnitude for noise-free, full-resolution, depth-averaged velocity data.

3.1.2.2 Sparse Data

After the ceiling has been established for the best possible bathymetry estimation, the potential impact of realistic data types can be explored in more detail. The first source of error

investigated is data resolution. This is covered in additional detail in the corresponding section regarding three-dimensional data. However, to separate the impact of three-dimensionality and data resolution, the first set of data is that of depth-averaged velocity data down-sampled to 20 m resolution, with no noise. At this resolution, there are about six data points across the river. The total water depth field comparison is shown in Figure 3.10, followed by the velocity magnitude comparison in Figure 3.11.

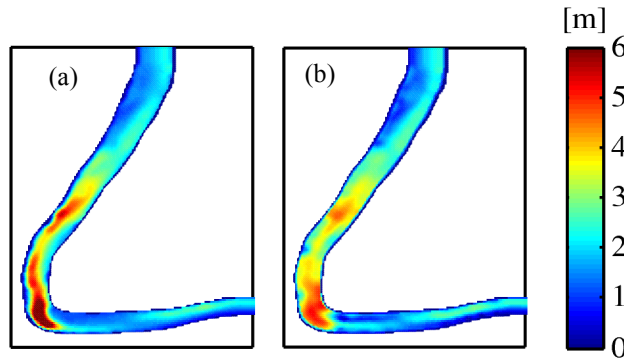


Figure 3.10. Comparison of the total water depth for noise-free, 20m-resolution, depth-averaged velocity data: (a) ‘true’ depth and (b) final estimate of depth.

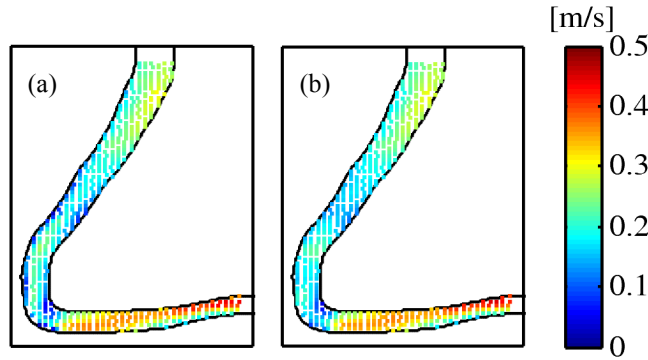


Figure 3.11. Comparison of the velocity magnitude for noise-free, 20m-resolution, depth-averaged velocity data: (a) ‘true’ velocity magnitude and (b) final estimate of velocity magnitude.

The statistics of the bathymetry estimation are similar to those of the full resolution case discussed in Section 3.1.2.1, but the RMS error in velocity has increased to 0.015 m/s, and the

RMS depth error has nearly doubled to 0.63 m. The statistically binned plots are shown in Figure 3.12.

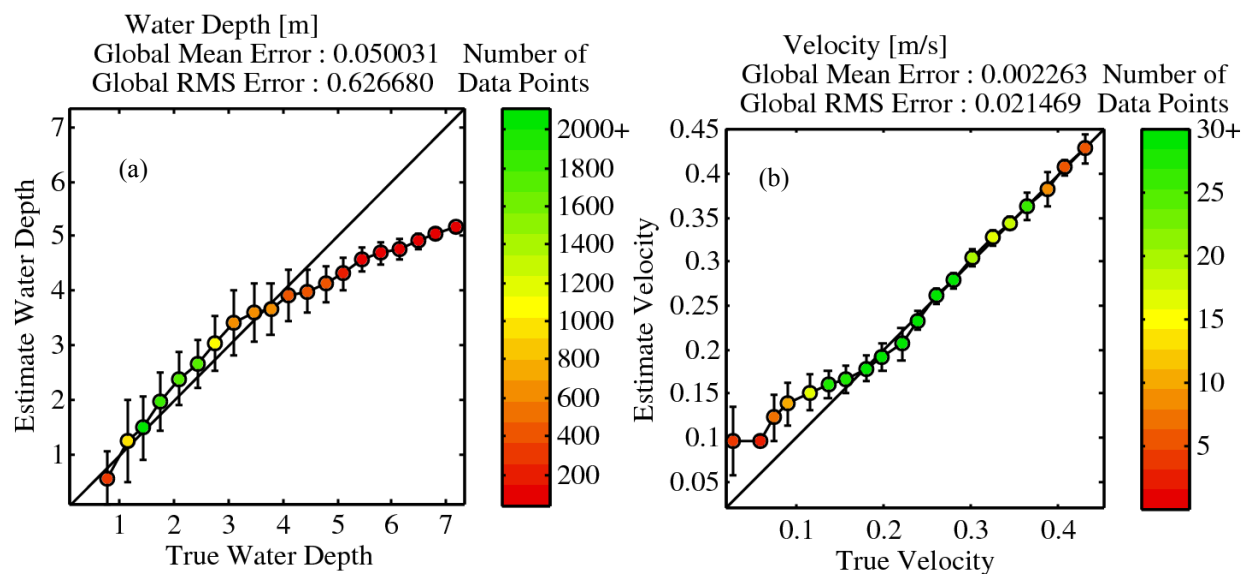


Figure 3.12. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for noise-free, 20m-resolution, depth-averaged velocity data. The error bars represent one standard deviation of the error between estimated and ‘true’ for each bin of the ‘true’ variable, and the colors represent the number of data points in each statistical bin.

3.1.2.3 Noisy Data

To further exercise the algorithm, and in particular to verify robustness, it is necessary to understand the effect of adding noise to the data. Hence, the second source of error investigated is data noise. This is also covered in additional detail in the corresponding section regarding three-dimensional data. However, to separate the impact of three-dimensionality, data resolution and data error, the second set of data is that of depth-averaged velocity data down-sampled to 20 m resolution, with 20 cm/s RMS noise added, sampled from a Gaussian distribution. The total water depth field comparison is shown in Figure 3.13, followed by the velocity magnitude comparison in Figure 3.14.

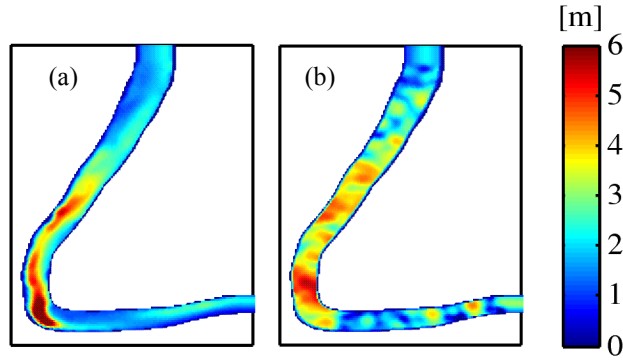


Figure 3.13. Comparison of the total water depth for 20cm/s RMS noise, 20m-resolution, depth-averaged velocity data: (a) ‘true’ depth and (b) final estimate of depth.

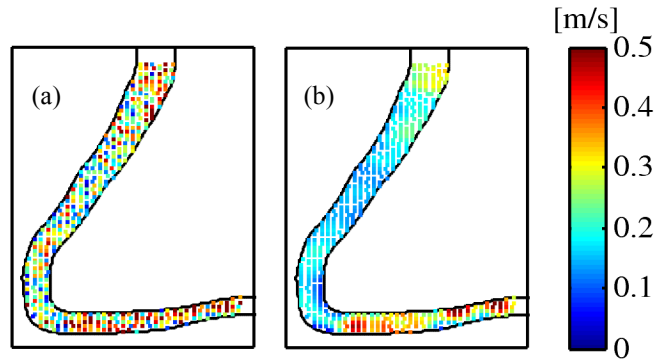


Figure 3.14. Comparison of the velocity magnitude for 20cm/s RMS noise, 20m-resolution, depth-averaged velocity data: (a) ‘true’ velocity magnitude and (b) final estimate of velocity magnitude.

The statistics of the bathymetry estimation are similar to those of the 20 m resolution case discussed in Section 3.1.2.2, but the RMS error in velocity has increased again to 0.091 m/s, and the RMS depth error has nearly doubled to 1.12 m. The statistically binned plots are shown in Figure 3.15.

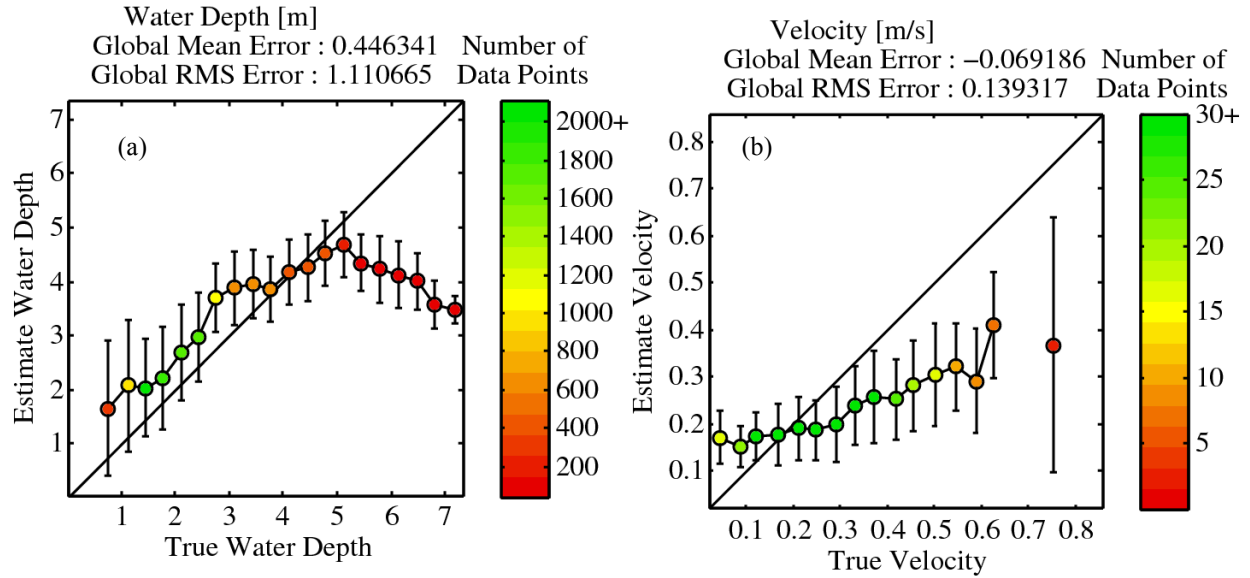


Figure 3.15. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for 20cm/s RMS noise, 20m-resolution, depth-averaged velocity data. The error bars represent one standard deviation of the error between estimated and ‘true’ for each bin of the ‘true’ variable, and the colors represent the number of data points in each statistical bin.

The previous three sub-sections are included to provide evidence of the ability of the models and algorithm to converge to a reasonable solution given data that has been degraded in both quantity and quality. They also give an initial understanding of the impact of quality and quantity of data on the ability of the algorithm to accurately predict river bathymetry. These effects are investigated in further detail after first including three-dimensionality in the model.

3.1.3 Surface (3D) Velocity Data

Solving the depth-averaged flow in a river is significantly less computationally expensive than solving for the entire three-dimensional velocity field. If it would be possible to obtain a good bathymetry estimate given only surface velocity data while using a predominantly two-dimensional flow solver, the algorithm would be deemed useful for the end-users. If the accuracy penalty were quantifiable, then it would be even more so. In the community of researchers

studying river flows, it is generally assumed that the correlation between surface and depth-averaged velocity is strong. A general rule of thumb is the surface velocity is about 30% greater than the depth-averaged velocity, though the ratio varies depending on bottom roughness (Rantz (1982) and Plant, Keller, Hayes and Spicer (2005)). This has been verified using simulated data in the following section.

3.1.3.1 Depth-Averaged to Surface Velocity Correlation

To determine a correlation between depth-averaged velocity and surface velocity, the Delft3D model has been configured with 15 vertical layers (on a terrain-following σ -coordinate system). A comparison of the 3D surface velocity and 2D depth-averaged velocity fields is shown in Figure 3.16.

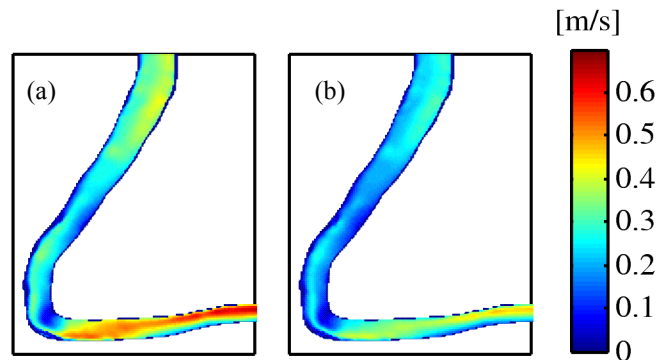


Figure 3.16. Comparison of the velocity magnitude: (a) 3D surface velocity magnitude and (b) 2D depth-averaged velocity magnitude.

A scatter plot of the velocity comparison for two different rivers is shown in Figure 3.17, along with a least-squares linear fit; the Kootenai River will be discussed in more detail in Section 3.2. The correlation is very high for these rivers. Using these two sets of data, a linear depth- to surface- velocity empirical model was developed and implemented in the algorithm, such that

$$\tilde{u}_i = \max \left\{ 1, \frac{1.31(U_j U_j)^{\frac{1}{2}} - 0.006}{(U_j U_j)^{\frac{1}{2}}} \right\} U_i. \quad (3.1)$$

This form is used to ensure that the empirical surface velocity goes to zero when the depth-averaged velocity goes to zero, without changing direction.

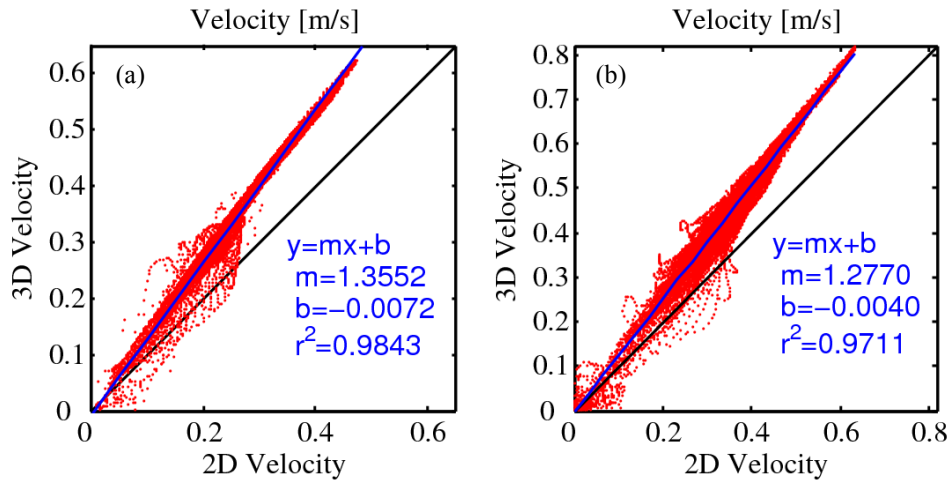


Figure 3.17. Surface velocity versus depth-averaged velocity for (a) Snohomish River and (b) Kootenai River.

For the remainder of this chapter, the empirical relationship is used in the bathymetry estimation algorithm, and it is shown to work very well. The empirically modeled velocity is used in the calculation of the cost function and adjoint data source terms. The advection velocities are not modified for the adjoint solver.

3.1.3.2 Full-Field Data

It is necessary to understand the effect of using the empirical surface-to-depth-velocity correlation. The first step in assessing the correlation is to use full-resolution, noise-free data. The simulated surface velocity is generated by running the Delft3D model with 15 vertical σ -

layers. Then, the river bathymetry is estimated using the correlation to ‘reconstruct’ the surface velocity from 2D, depth-averaged simulations. The total water depth field comparison is shown in Figure 3.18, followed by the velocity magnitude comparison in Figure 3.19.

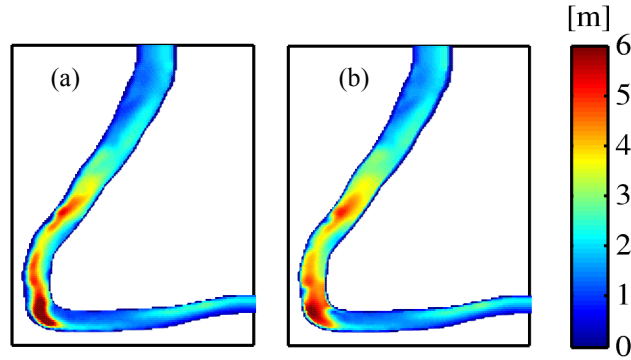


Figure 3.18. Comparison of the total water depth for noise-free, full-resolution, surface velocity data: (a) ‘true’ depth and (b) final estimate of depth.

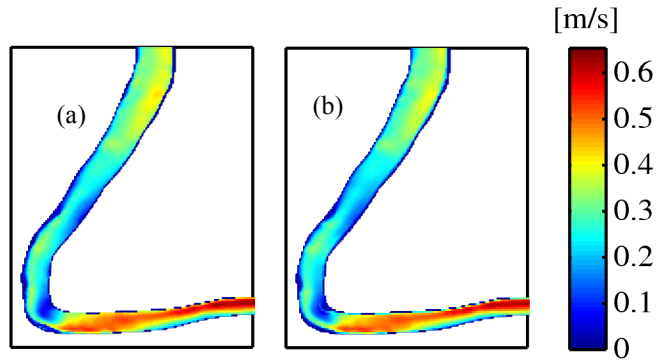


Figure 3.19. Comparison of the velocity magnitude for noise-free, full-resolution, surface velocity data: (a) ‘true’ velocity magnitude and (b) final estimate of velocity magnitude.

The statistics of the bathymetry estimation are similar to those of the 2D assimilation case discussed in Section 3.1.2.1; the RMS error in velocity is now 0.018 m/s, and the RMS depth error is 0.50 m. The statistically binned plots are shown in Figure 3.20.

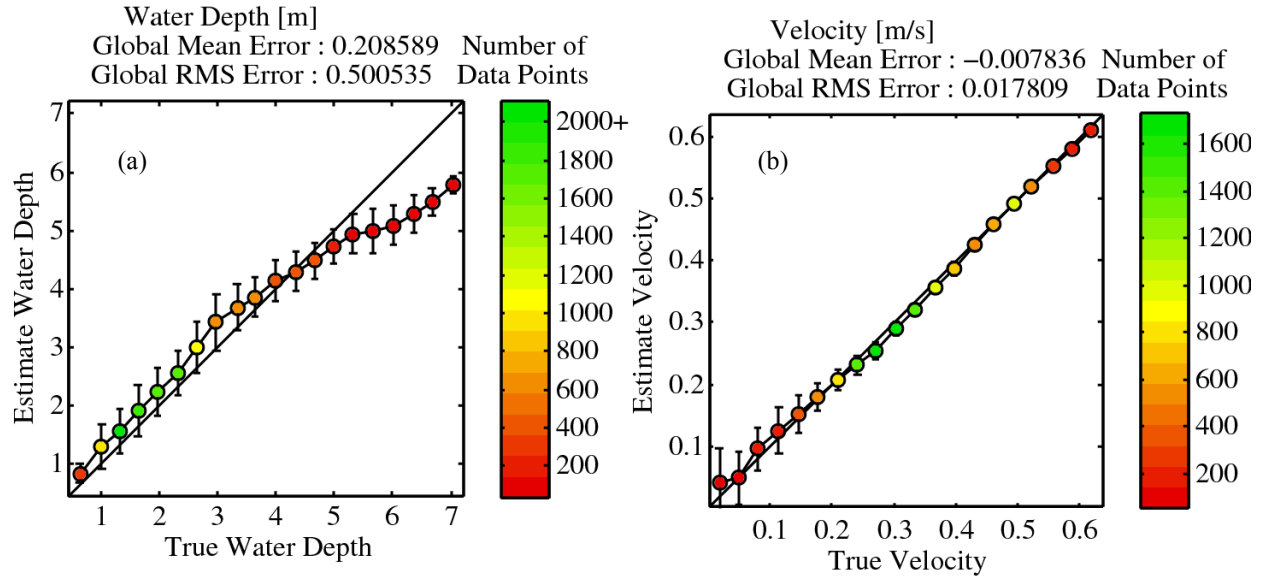


Figure 3.20. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for noise-free, full-resolution, surface velocity data. The error bars represent one standard deviation of the error between estimated and ‘true’ for each bin of the ‘true’ variable, and the colors represent the number of data points in each statistical bin.

This case provides an estimate of the ability of the empirical correlation to effectively estimate the surface velocity from a two-dimensional simulation. The following sub-sections are intended to investigate the impact of reduced resolution surface velocity data. The effects are separated into resolution effects and data noise-level effects.

3.1.3.3 Data Resolution

The effect of data resolution can be investigated by degrading the ‘true’ surface velocity data. First, the effect of resolution with out any noise is shown to be similar to that seen in the two-dimensional bathymetry estimation, as in section 3.1.2.2, down-sampling the surface velocity to 20m resolution. The total water depth field comparison is shown in Figure 3.21, followed by the velocity magnitude comparison in Figure 3.22.

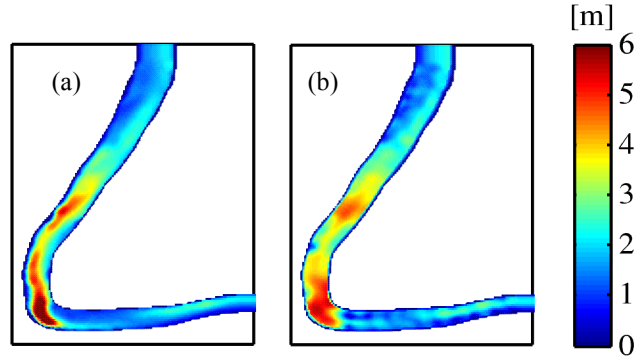


Figure 3.21. Comparison of the total water depth for noise-free, 20m-resolution, surface velocity data: (a) ‘true’ depth and (b) final estimate of depth.

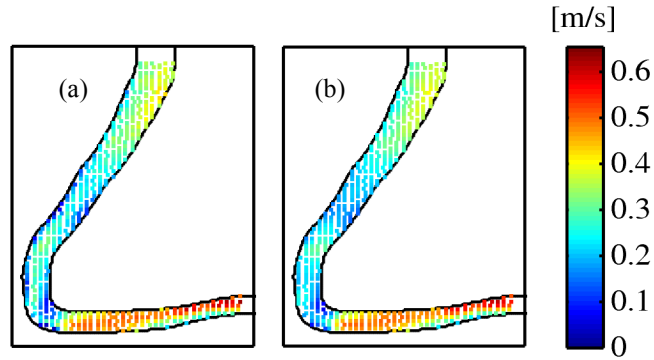


Figure 3.22. Comparison of the velocity magnitude for noise-free, 20m-resolution, surface velocity data: (a) ‘true’ velocity magnitude and (b) final estimate of velocity magnitude.

With respect to statistics, those of the bathymetry estimation using low resolution surface velocity data are similar to those of the 2D assimilation case discussed in Section 3.1.2.2, although the RMS error in velocity has increased to 0.030 m/s, and the RMS depth error has increased to 0.69 m. The statistically binned plots are shown in Figure 3.23.

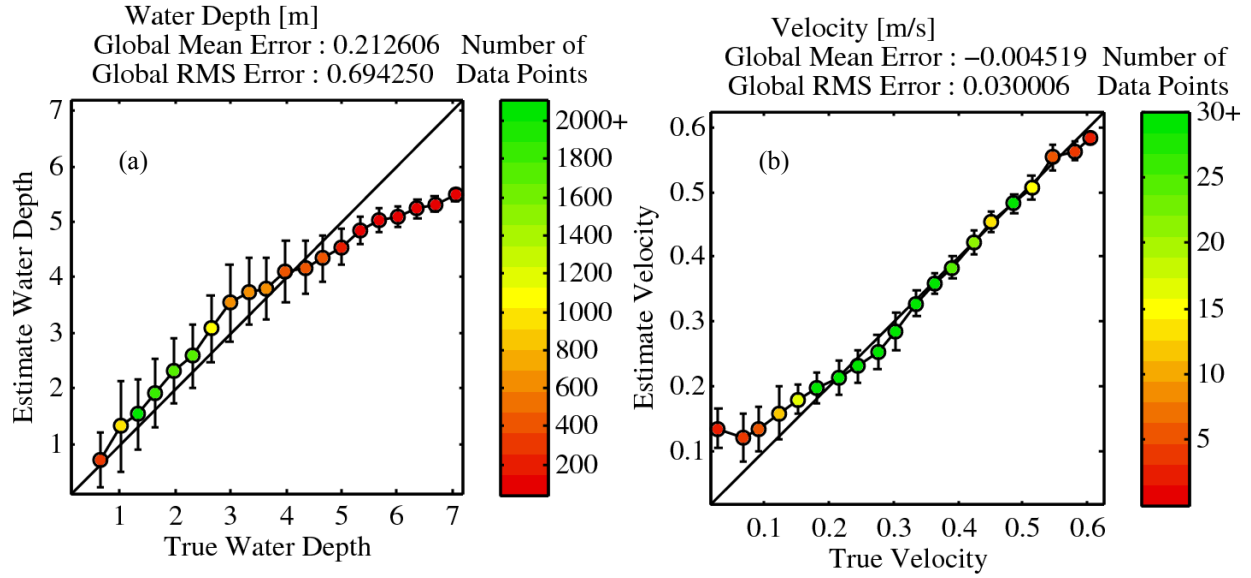


Figure 3.23. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for noise-free, 20m-resolution, surface velocity data. The error bars represent one standard deviation of the error between estimated and ‘true’ for each bin of the ‘true’ variable, and the colors represent the number of data points in each statistical bin.

The correlation has been shown to be robust with respect to the empirical correlation. Further, data from a real sensor system will typically exhibit noise, and there is usually a trade-off between resolution and noise level. Here, the effect of resolution is estimated by holding the error in the data fixed while varying the resolution of the data. Three cases, all with 20cm/s noise with resolutions varying from 10m to 40m are examined. At 10m resolution, there are 10-12 data points across the river, and at 40m resolution, there are only 2-4. The estimated depths are shown in Figure 3.24, and the ‘true’ velocity comparisons are in Figure 3.25.

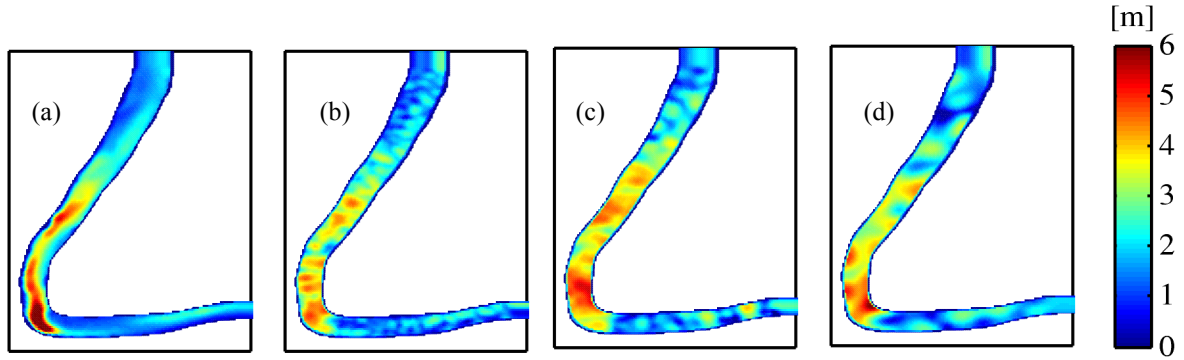


Figure 3.24. Comparison of the total water depth for fixed surface velocity data noise level (20cm/s): (a) ‘true’ depth, (b) estimated depth for 10m resolution data, (c) estimated depth for 20m resolution data, and (d) estimated depth for 40m resolution data.

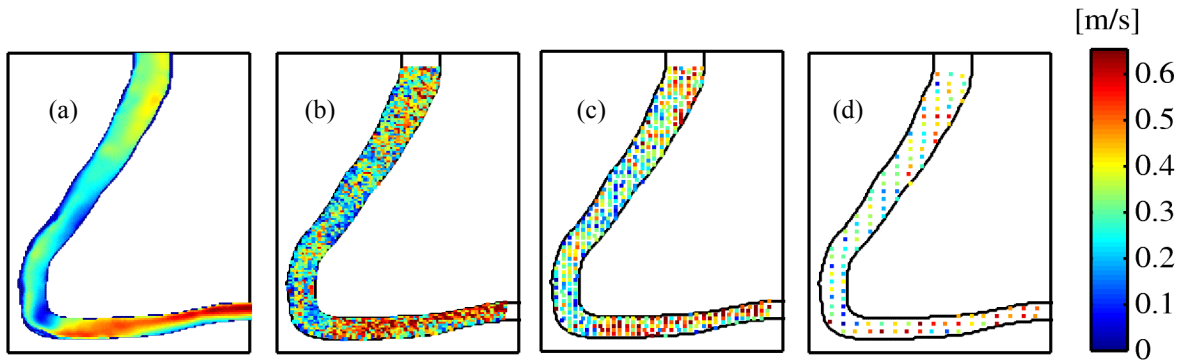


Figure 3.25. Comparison of the ‘true’ surface velocity magnitude for fixed data noise level (20cm/s): (a) full-resolution, noise-free velocity, (b) velocity for 10m resolution data, (c) velocity for 20m resolution data, and (d) velocity for 40m resolution data.

The statistics for the data resolution comparison are shown in Table 3.1, and a comprehensive comparison is shown in Figure 3.26, through the use of the binned statistics. It is notable that the algorithm performs equally well on data that is at 20m resolution and 40m resolution. However, if the bias (mean) error is removed, the variance in the depth error does decrease as the resolution increases.

Table 3.1. Bulk estimation statistics for different data resolutions with fixed noise level (20cm/s).

Resolution [m]	Mean Depth Error [m]	RMS Depth Error [m]	Mean Velocity Error [m/s]	RMS Velocity Error [m/s]
10	0.0602	0.7139	-0.0243	0.1328
20	0.4811	1.0555	-0.0641	0.1423
40	0.2153	1.0430	-0.0480	0.1226

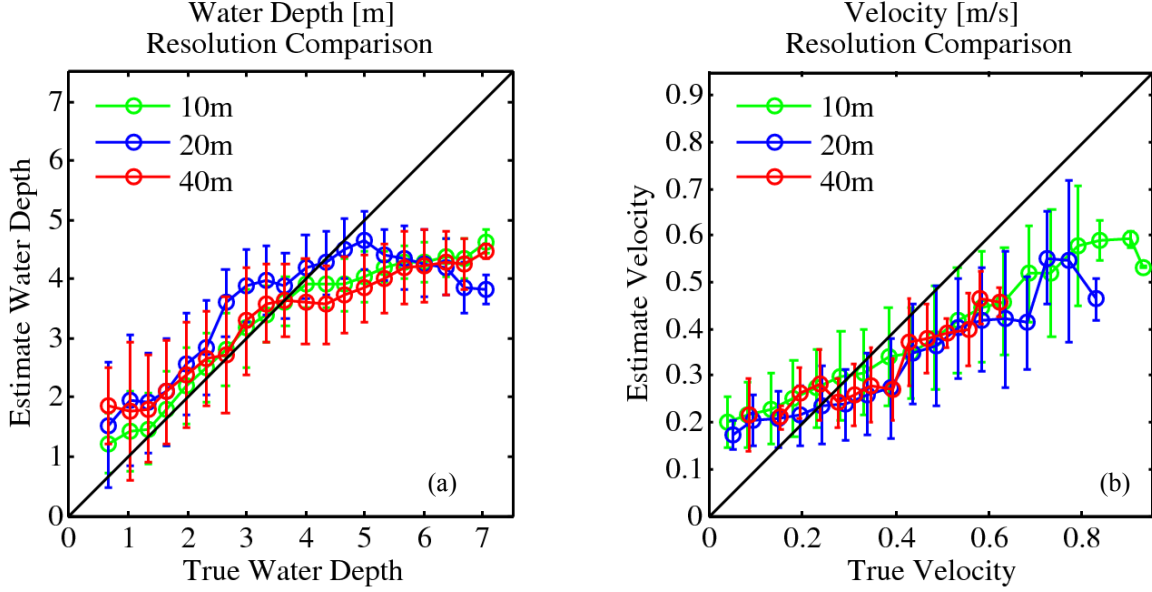


Figure 3.26. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude with fixed surface velocity error (20cm/s). The error bars represent one standard deviation of the error between estimated and ‘true’ for each bin of the ‘true’ variable.

3.1.3.4 Data Noise

In this section, the effect of data noise level is estimated by holding the resolution of the data fixed while varying the noise in the data. Three cases, all with 20m resolution data with noise levels varying from 10cm/s to 40cm/s are examined. The estimated depths are shown in Figure 3.27, and the ‘true’ velocity comparisons are in Figure 3.28.

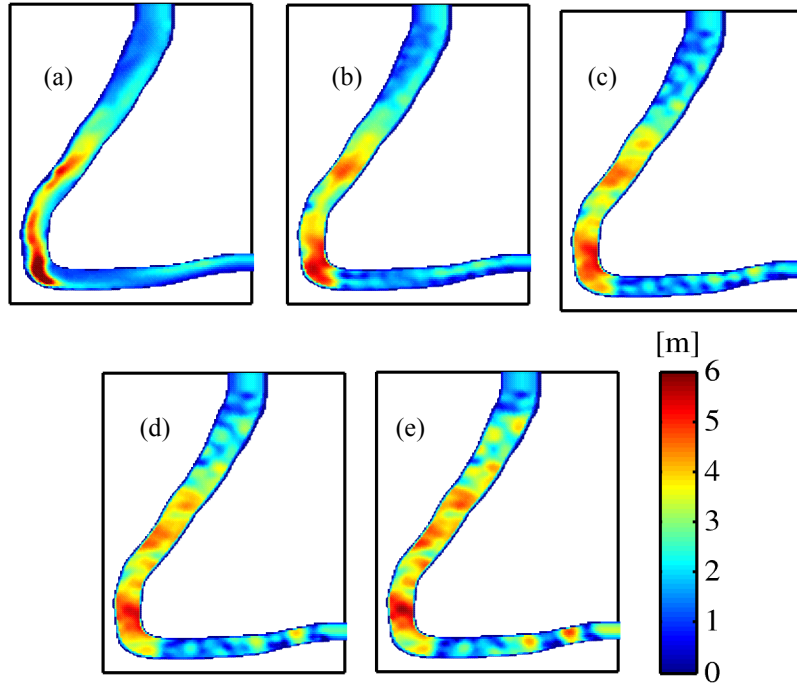


Figure 3.27. Comparison of the total water depth for fixed surface velocity data resolution (20m): (a) ‘true’ depth, (b) estimated depth for 0cm/s noise data, (c) estimated depth for 10cm/s noise data, (d) estimated depth for 20cm/s noise data, and (e) estimated depth for 40cm/s noise data.

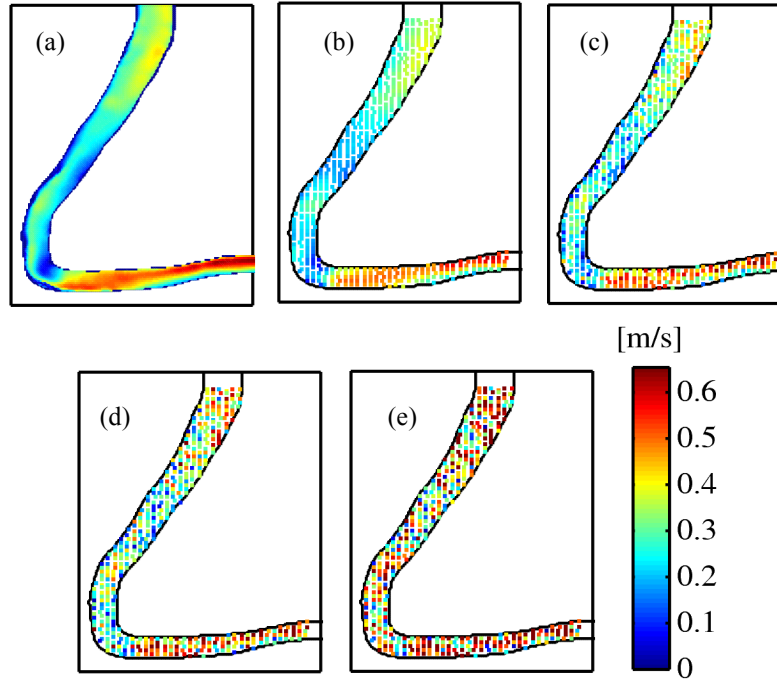


Figure 3.28. Comparison of the ‘true’ surface velocity magnitude for fixed data resolution (20m): (a) full-field, noise-free velocity, (b) velocity for 0cm/s noise data, (c) velocity for 10cm/s noise data, (d) velocity for 20cm/s noise data, and (e) velocity for 40cm/s noise data.

The statistics for the data noise level comparison are shown in Table 3.2, and a comprehensive comparison is shown in Figure 3.29, through the use of the binned statistics.

Table 3.2. Bulk estimation statistics for different data noise levels with fixed resolution (20m).

Data Error	RMS Error [cm/s]	Mean Error [m]	Depth Error [m]	RMS Error [m]	Mean Velocity Error [m/s]	RMS Velocity Error [m/s]
0		0.2126		0.6943	-0.0045	0.0300
10		0.3203		0.8758	-0.0181	0.0713
20		0.4811		1.0555	-0.0641	0.1423
40		0.7096		1.3158	-0.1954	0.2993

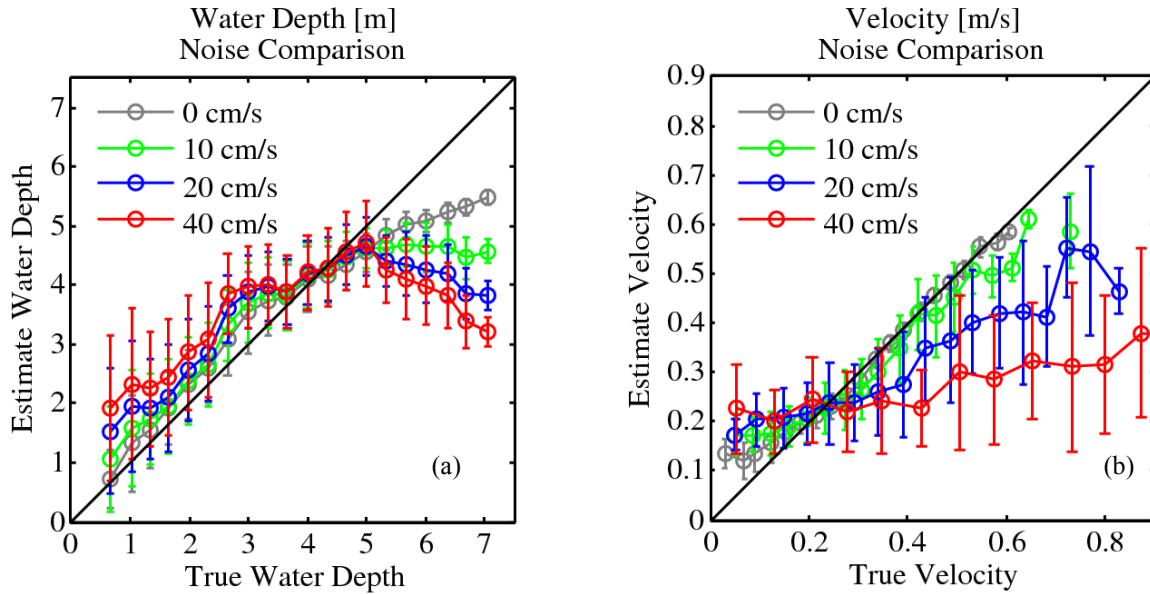


Figure 3.29. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude with fixed surface velocity resolution (20m). The error bars represent one standard deviation of the error between estimated and ‘true’ for each bin of the ‘true’ variable.

3.1.4 Hydrostatic Assumption

To this point, all of the simulations have been conducted using a hydrostatic solver. In this section, the modeling error that is associated with this assumption is shown to be no more significant than the typical errors associated with using real data. That is to say that actual surface velocities collected in an experiment would be similar to those resulting from a simulation using a hydrostatic solver. The most significant advantages of utilizing the hydrostatic assumption are computational efficiency and robustness. The Delft3D model can be run in non-hydrostatic mode, but there are some limitations. The use of a σ -grid is not allowed and the model cannot be run in parallel. To obtain similar vertical resolution as the previous simulations, the vertical grid was uniformly spaced at a resolution of 20cm. To truly infer the difference between the two model results, both were run using the same vertical grid configuration. These

simulations were also conducted at twice the horizontal resolution of the previous bathymetry estimation for more accuracy.

A comparison of velocity fields is shown in Figure 3.30. It is worth noting that the surface velocities obtained using the hydrostatic model with fixed vertical resolution and higher horizontal resolution result are slightly lower than those obtained at lower resolution on a σ -grid. The most striking difference between the two surface velocities is the relative uniformity of the non-hydrostatic case. It does not reach the same peak magnitude upstream, and is slightly higher downstream. The lower velocity regions are very similar.

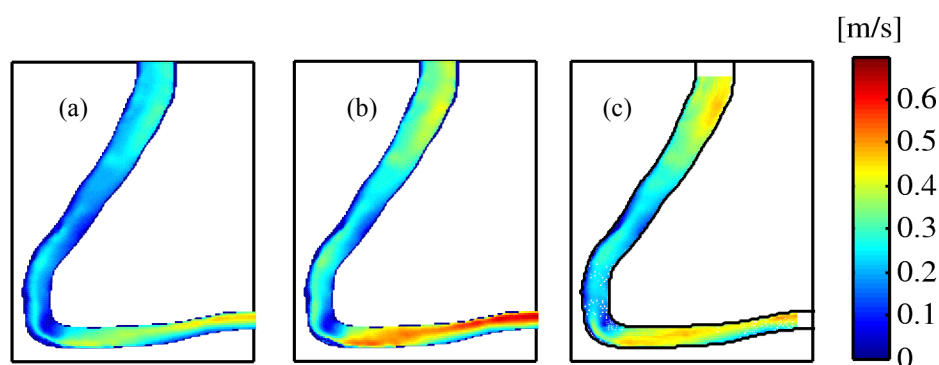


Figure 3.30. Comparison of the velocity magnitude: (a) 2D depth-averaged velocity, (b) 3D hydrostatic surface velocity, and (c) 3D non-hydrostatic surface velocity.

The hydrostatic assumption does potentially impact the empirical correlation that has been used thus far. The correlation is shown and compared to the previous hydrostatic data in Figure 3.31. Because the original correlation still fits the data and for consistency, no adjustments were made with respect to the empirical depth-averaged to surface velocity correlation in the bathymetry estimation algorithm.

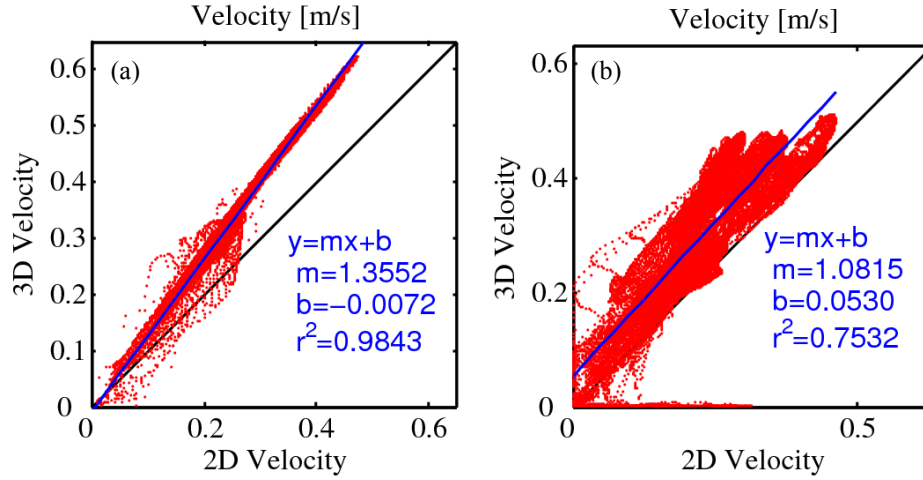


Figure 3.31. Surface velocity versus depth-averaged velocity correlation for (a) low-resolution hydrostatic model and (b) high-resolution non-hydrostatic model.

To assess the impact of the hydrostatic assumption, bathymetry estimation was conducted using the full-field non-hydrostatic surface velocity as the observational data. The results are compared to the results using hydrostatic surface velocity data in Figure 3.32.

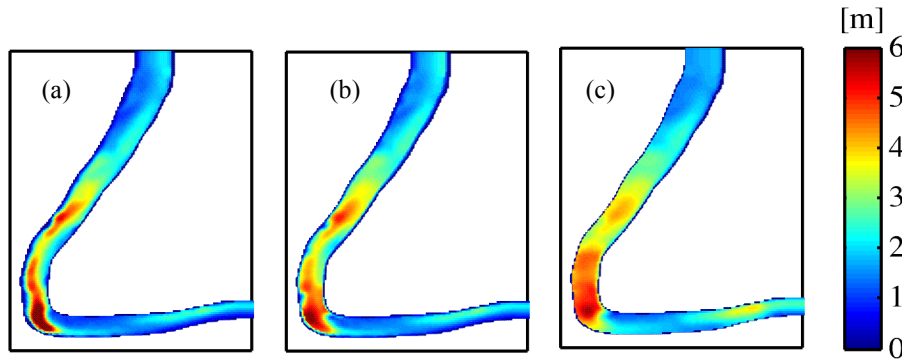


Figure 3.32. Comparison of the total water depth: (a) 2D ‘true’ depth, (b) 3D hydrostatic estimated depth, and (c) 3D non-hydrostatic estimated depth.

With respect to statistics of the bathymetry estimation, the RMS error in velocity has increased to 0.057 m/s, and the RMS depth error has increased to 0.80 m. The statistically binned plots are shown in Figure 3.33. The largest errors in the estimation are in the low velocity and low depth

regions, but overall, the hydrostatic assumption appears to have no more effect than the effects of ‘real’ data resolution and noise.

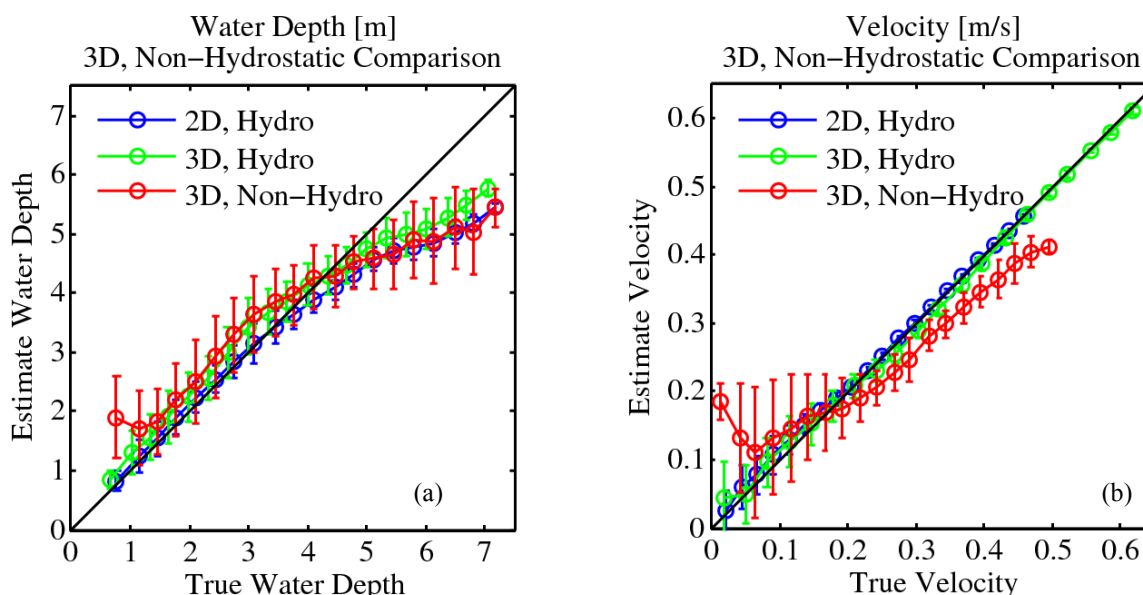


Figure 3.33. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for different types of forward model. The error bars represent one standard deviation of the error between estimated and ‘true’ for each bin of the ‘true’ variable.

3.2 Kootenai River

3.2.1 Geographical Context

To further exercise the algorithm and models, bathymetry estimation was conducted for the Kootenai River in Idaho. The reach that is currently being investigated is a 20 km reach characterized by several bends in a meandering section of the river. The average width of the river in this reach is about 200m. Rich McDonald at the USGS provided the bathymetry. The bathymetry represents a collection of various different data sets over multiple years, using different experimental techniques (e.g., LiDAR and surveys). The region of interest is shown in Figure 3.34. Note that in Figure 3.34 and subsequent two-dimensional plots of the Kootenai River, the axes labels are omitted for space.

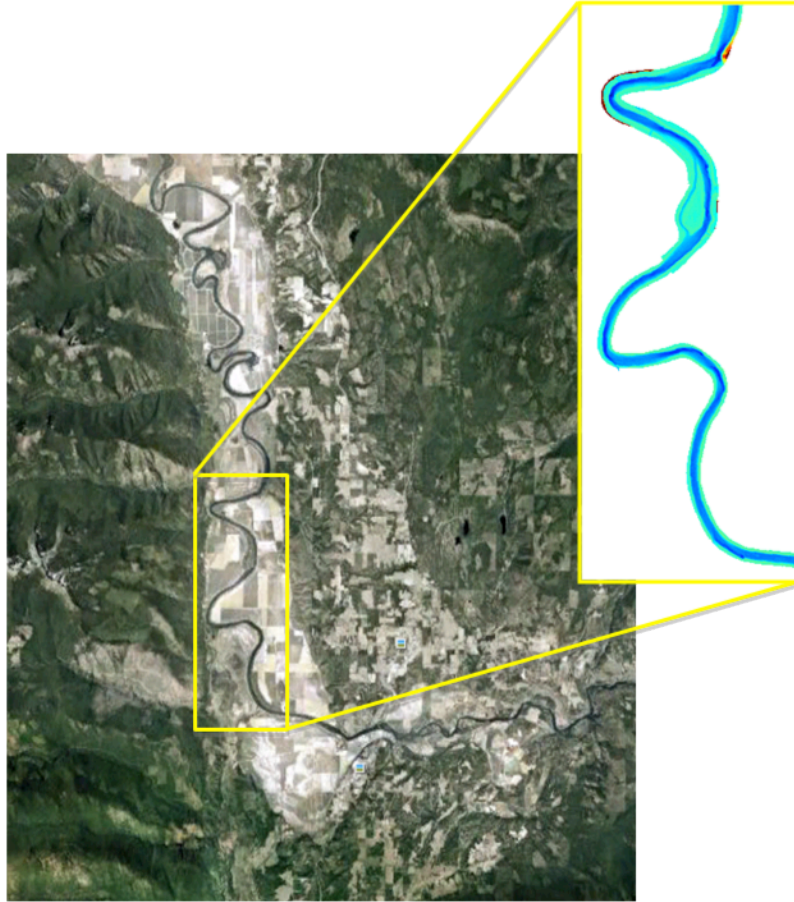


Figure 3.34. The Kootenai River: region overview and bathymetry. This reach is approximately 20km long with an average width of 200m. The sharp bend in the river in the northwest is located at 116.418°W, 48.786°N. In this figure and subsequent two-dimensional figures, the axes labels are omitted for space.

This reach differs from the Snohomish River in length (20km versus 2km) and flow-rate ($220\text{m}^3/\text{s}$ versus $50\text{m}^3/\text{s}$). The bathymetry has been interpolated onto a boundary fitted, orthogonal curvilinear grid with an average resolution of 7m (2718 x 26 cells in the downstream and cross-stream directions, respectively). Recall that the Snohomish River simulations had a nominal resolution of about 3m.

The upstream/inflow boundary condition was set to a total volumetric flow rate of $220\text{ m}^3/\text{s}$, and the effects of bottom friction were included, using a Manning roughness coefficient of 0.05

$\text{s/m}^{1/3}$. The downstream/outflow forward boundary condition was the same as the Snohomish River outflow treatment. The domain was extended (8 km) to the North to ensure that the boundary condition had a minimal impact on the region of interest. The results from a Delft3D simulation of the Kootenai River using the ‘true’ bathymetry are shown in Figure 3.35. Similar to the Snohomish, the inlet is in the southeast corner, and the outlet is at the north. The forward simulation is run for 24 hours, with a time step of 4 seconds. The forward simulations require 15 minutes on 8 CPUS.

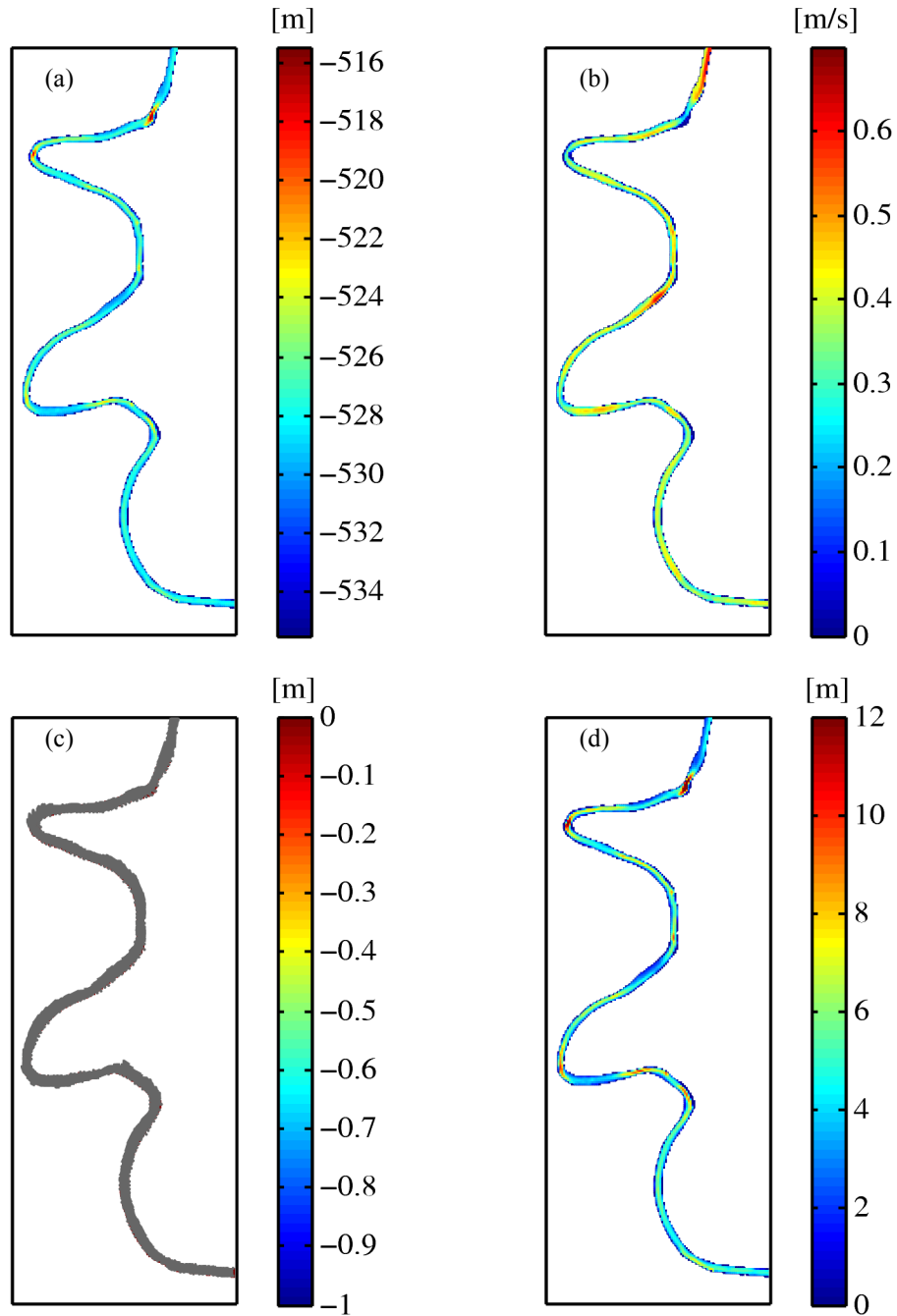


Figure 3.35. ‘True’ simulation results of Kootenai River: (a) bathymetry, (b) velocity magnitude, (c) water surface elevation with velocity vectors, and (d) total water depth.

Only two cases were conducted for the Kootenai, with the intent of assessing the robustness of the algorithm/models: 2D, full-field representing a best-case scenario; and a 3D, sparse noisy case. Results are presented below.

3.2.2 Full-Field Depth-Averaged (2D) Velocity Data

The best possible performance of the algorithm is for data that is at the model resolution with no noise and consistent with the hydrodynamics (depth-averaged). A comparison of the ‘true’, first guess, and final depth is shown in Figure 3.36. A similar comparison for velocity magnitude is shown in Figure 3.37. Because this is a different river, the initial guess is once again included to show the significant changes in both fields as a result of the assimilation.

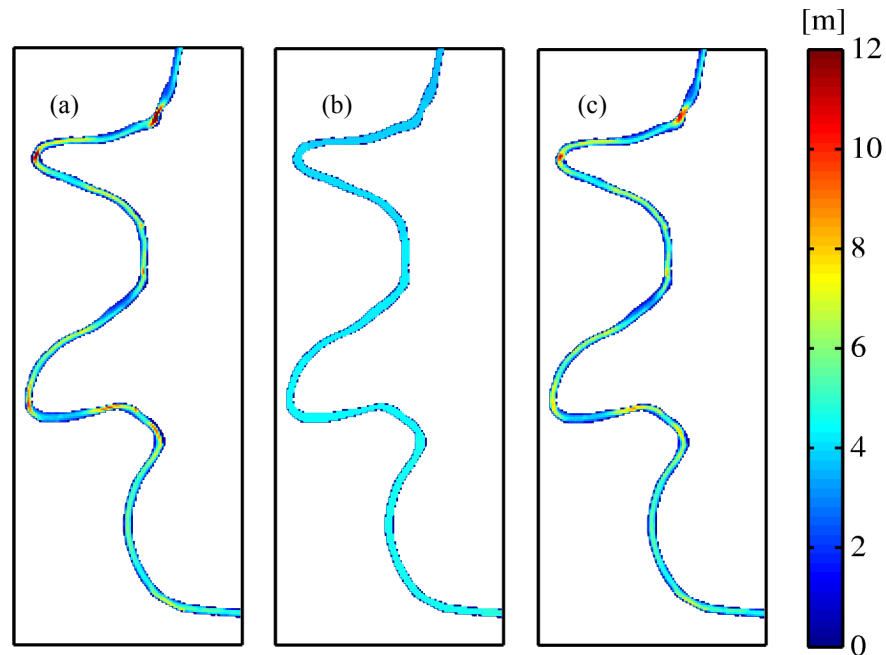


Figure 3.36. Kootenai River comparison of the total water depth for noise-free, full-resolution, depth-averaged velocity data: (a) ‘true’ depth, (b) initial depth, and (c) final estimate of depth.

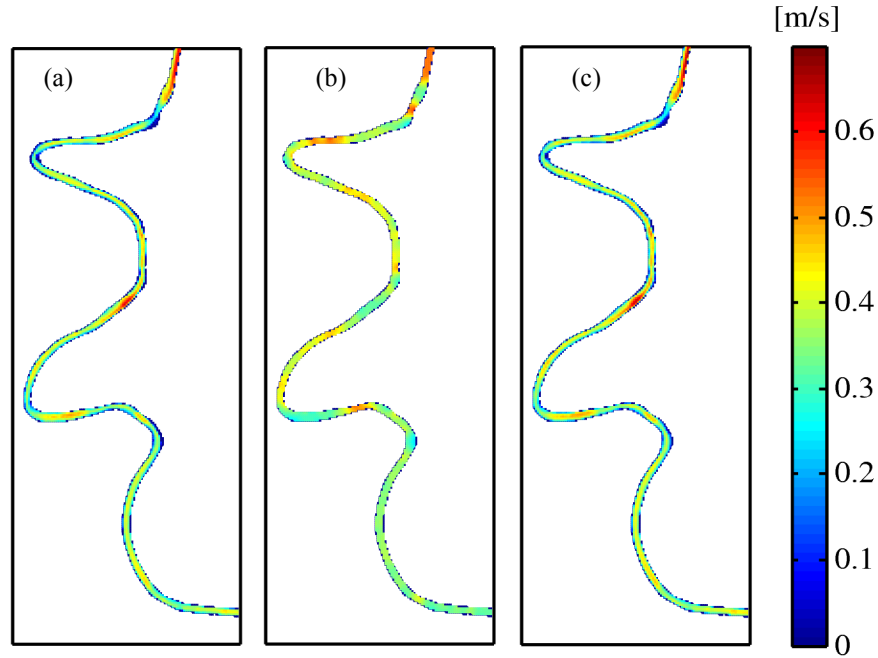


Figure 3.37. Kootenai River comparison of the velocity magnitude for noise-free, full-resolution, depth-averaged velocity data: (a) ‘true’ velocity magnitude, (b) initial velocity magnitude, and (c) final estimate of velocity magnitude.

After 30 line minimizations, the cost function shows an overall increase in match between observed and estimated velocity fields of about 95%. The RMS velocity error is 0.020 m/s, and the depth error is 0.75 m. The statistically binned plots are shown in Figure 3.38. The maximum depth for which the algorithm accurately estimates depth has increased (relative to the corresponding Snohomish case) to about 7m. While the RMS depth error is higher than seen in the Snohomish results, it is important to note differences in the model setup and the rivers themselves. The velocities are very similar, but the Kootenai River is much wider and deeper; also, the model resolution has increased from 3m to 7m. Still, the largest errors are again observed in regions with fewer data points.

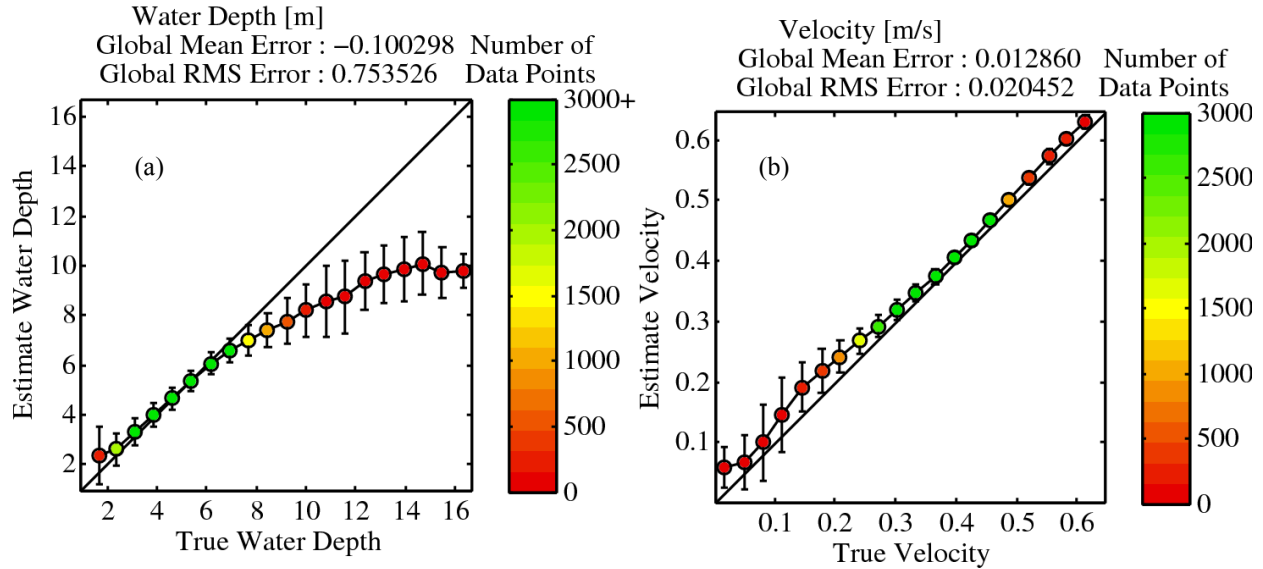


Figure 3.38. Kootenai River statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude. The error bars represent one standard deviation of the error between estimated and ‘true’ for each bin of the ‘true’ variable, and the colors represent the number of data points in each statistical bin.

These results indicate that the methodology should be robust to different rivers, though more exercising and characterization of the influence of various model inputs on bathymetry estimation is necessary (e.g., average bed slope and bottom friction coefficient impacts).

3.2.3 Sparse, Noisy Surface (3D) Velocity Data

As a final test of the algorithm, synthetic data was generated from a 3D model of the Kootenai River, down-sampled to 20m resolution, with 20cm/s RMS noise added. This corresponds with the similar Snohomish River case described in Sections 3.1.3.3 and 3.1.3.4, but the Kootenai is also wider, and 20m resolution data results in 8-10 data points across the river. A comparison of the ‘true’, first guess, and final depth is shown in Figure 3.39. A similar comparison for velocity magnitude is shown in Figure 3.40. The empirical correlation between depth-averaged and surface velocity has not been modified (for consistency).

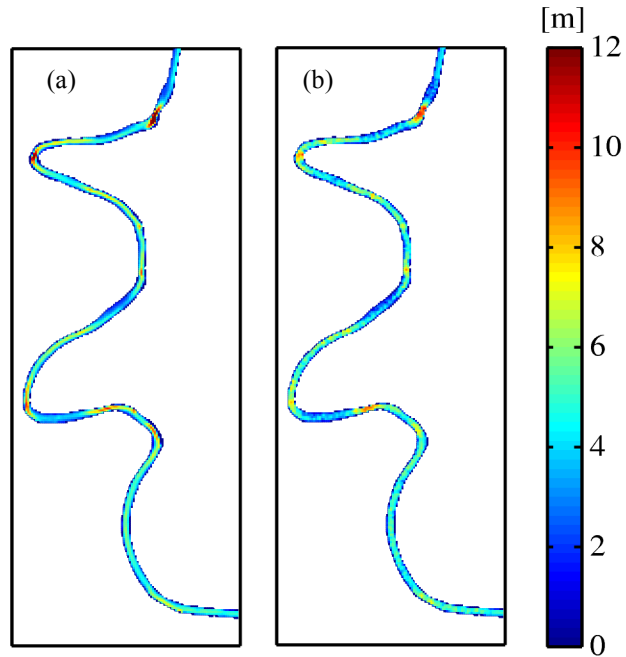


Figure 3.39. Kootenai River comparison of the total water depth for 20cm/s RMS noise, 20m-resolution, surface velocity data: (a) ‘true’ depth and (b) final estimate of depth.

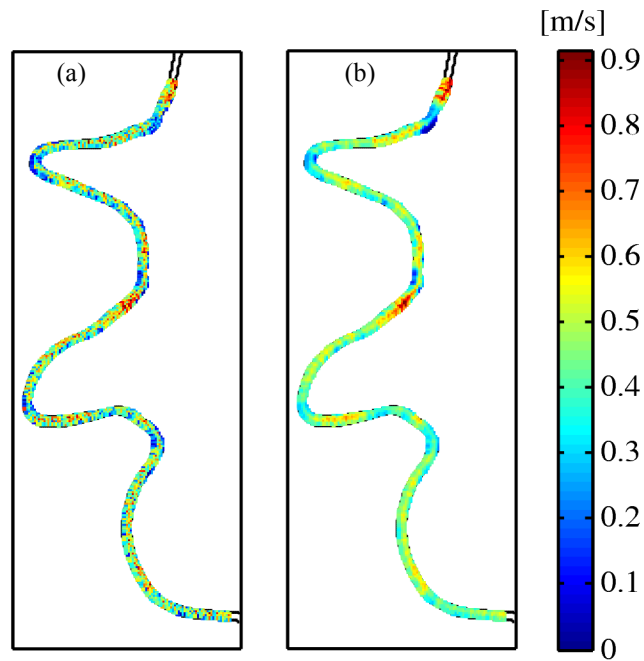


Figure 3.40. Kootenai River comparison of the velocity magnitude for 20cm/s RMS noise, 20m-resolution, surface velocity data: (a) ‘true’ velocity magnitude and (b) final estimate of velocity magnitude.

With the data degraded as such, the RMS velocity error has increased to 0.136 m/s, and the depth error is 1.21 m. The statistically binned plots are shown in Figure 3.41.

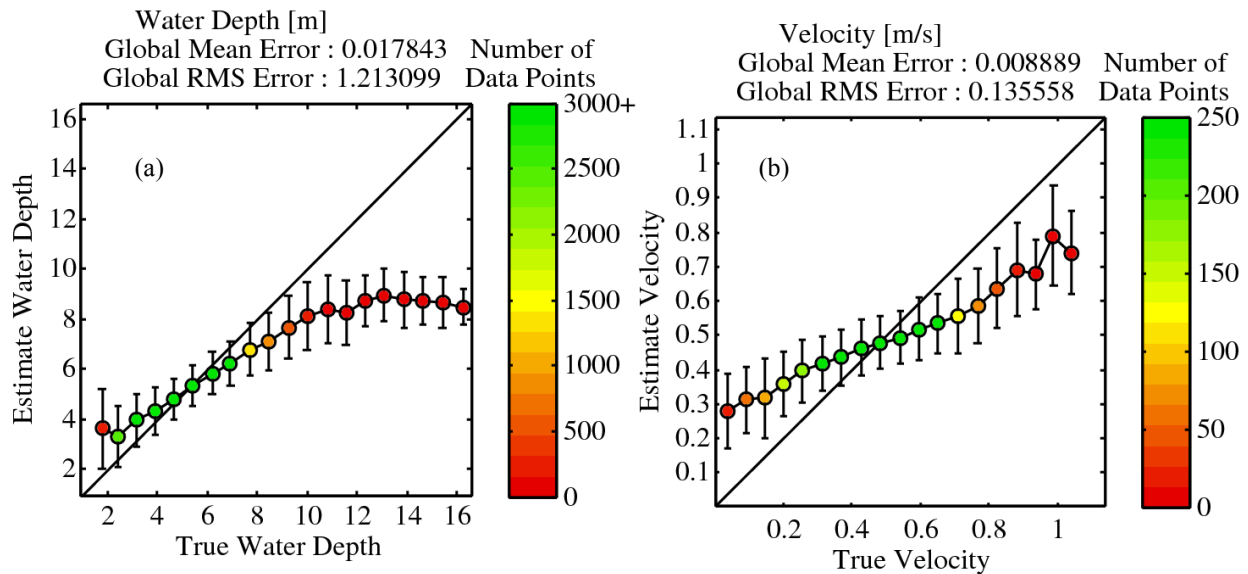


Figure 3.41. Kootenai River statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for 20cm/s RMS noise, 20m-resolution, surface velocity data. The error bars represent one standard deviation of the error between estimated and ‘true’ for each bin of the ‘true’ variable, and the colors represent the number of data points in each statistical bin.

This is the final case investigated as a part of this work. Further discussion of all results follow in the next section.

3.3 Summary of Results

A complete listing of all thirteen cases used to exercise the bathymetry estimation algorithm is provided in Table 3.3. The bathymetry estimation has been shown to be a function of the river, the forward model setup, the data resolution, and the data noise level. The algorithm and all accompanying models have been shown to be robust to model (river) geometries, as well as to various amounts of data degradation. The resultant RMS depth errors fall between 0.3m and

1.3m, showing very good overall performance. The algorithm results follow expected trends in depth errors versus data noise level (Section 3.1.3.4).

Table 3.3. Complete listing of all cases.

River	Forward Model [2D/3D; Non-Hydro]	Data Resolution [m]	Data RMS Error [cm/s]	Mean Depth Error [m]	RMS Depth Error [m]	Mean Velocity Error [m/s]	RMS Velocity Error [m/s]
Snohomish	2D;Hydro	3	0	0.0016	0.3774	0.0036	0.0111
Snohomish	3D;Hydro	3	0	0.2086	0.5005	-0.0078	0.0178
Snohomish	3D;Non-hydro	3	0	0.3489	0.7998	-0.0376	0.0568
Snohomish	2D;Hydro	20	0	0.0500	0.6267	0.0023	0.0215
Snohomish	2D;Hydro	20	20	0.4463	1.1107	-0.0692	0.1393
Snohomish	3D;Hydro	20	0	0.2126	0.6943	-0.0045	0.0300
Snohomish	3D;Hydro	20	10	0.3203	0.8758	-0.0181	0.0713
Snohomish	3D;Hydro	20	20	0.4811	1.0555	-0.0641	0.1423
Snohomish	3D;Hydro	20	40	0.7096	1.3158	-0.1954	0.2993
Snohomish	3D;Hydro	10	20	0.0602	0.7139	-0.0243	0.1328
Snohomish	3D;Hydro	40	20	0.2153	1.0430	-0.0480	0.1226
Kootenai	2D;Hydro	7	0	-0.1003	0.7535	0.0129	0.0205
Kootenai	3D;Hydro	20	20	0.0178	1.2131	0.0089	0.1356

The algorithm performance for different input data resolution and noise level is summarized in Figure 3.42. The plots include all 3D, hydrostatic cases for both rivers. The most curious aspect of Figure 3.42(a) is that the model performance does not decline with an increase in data resolution from 20m to 40m, as indicated by the red circles. It is important to note that most remote sensing platforms, in post-processing of the images used to construct a velocity field, must trade off between higher resolution and lower noise level. Considering that and the results of the bathymetry estimation algorithm, it seems that for a given river (in particular a given river width), there is an optimal trade-off. To put it another way, there is a minimum number of data

points required across the river to estimate the cross-stream profile of river depth. If that threshold is met, reducing the noise level would be better than increasing the data resolution. The 2.2km reach of the Snohomish River averages 120m in river width, such that 40m resolution data provides between three and four data points across the river.

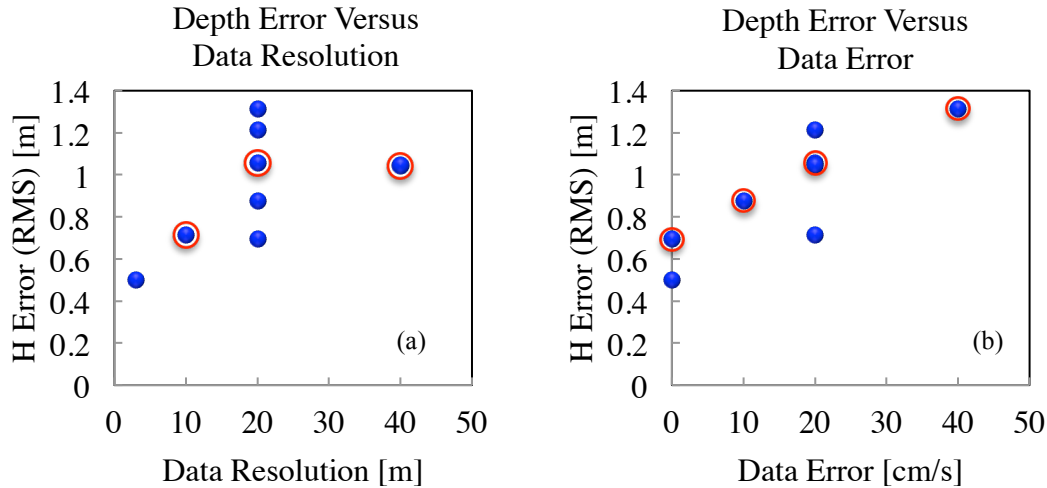


Figure 3.42. Depth estimation error comparison for varying (a) data resolution and (b) data error. The blue symbols represent all 3D, hydrostatic cases; the red circles indicate Snohomish River cases with fixed data error in (a) and fixed data resolution in (b).

An important test of the algorithm is to show that it performs well on different types of rivers. While there are a multitude of different types of rivers and river classes, this study is based on meandering reaches, and the algorithm has been shown to be robust for two substantially different river reaches. This is shown in the model comparisons in Figure 3.43. The difference in river scales is obvious, and the overall algorithm performance is very good for both. The Kootenai River is wider, deeper, and longer, and has higher maximum velocities. For computational efficiency, the model resolution was lower in the Kootenai setup. That is the most

likely reason for the slightly better performance on the Snohomish River, rather than the algorithm itself.

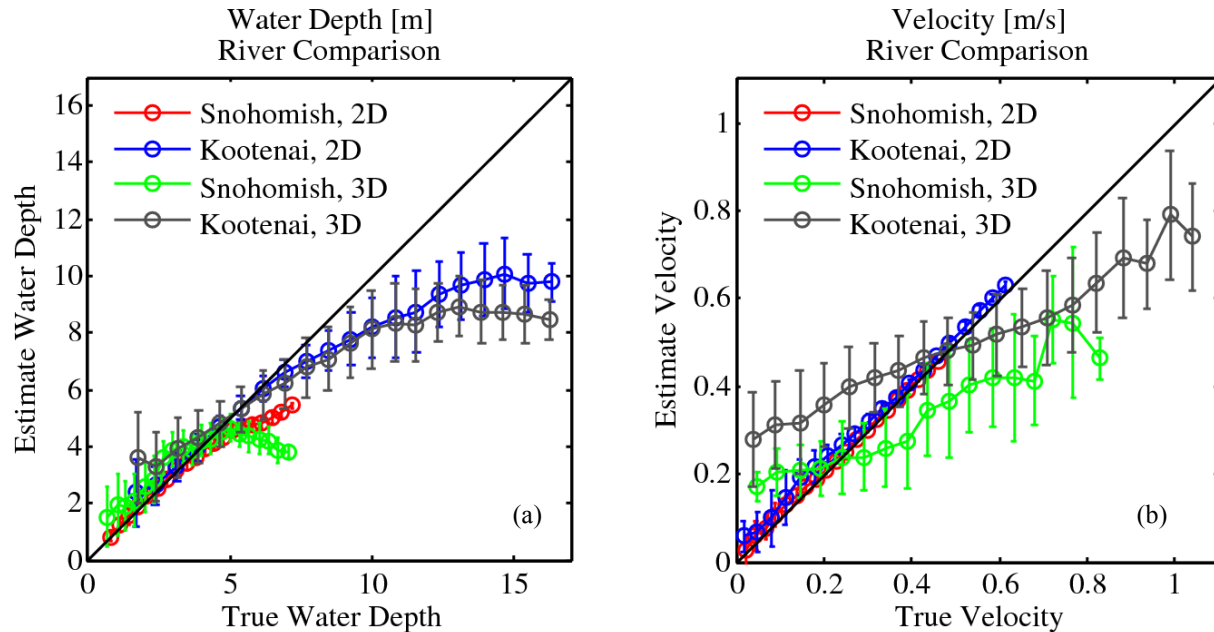


Figure 3.43. Statistical comparison of estimated versus ‘true’ (a) total water depth and (b) velocity magnitude for different rivers and models. The error bars represent one standard deviation of the error between estimated and ‘true’ for each bin of the ‘true’ variable.

Finally, the overall depth estimation errors attributable to the hydrostatic assumption have been shown to be on the order of those due to realistic data resolution and noise levels (Section 3.1.4), though there is a notable difference in the errors as a function of downstream distance. When the surface velocity data is from a full-resolution non-hydrostatic model, the algorithm over predicts upstream depths and under predicts downstream depths for the Snohomish River under the specific conditions used here. This is most likely due to an inconsistency between the bottom slope and bottom friction for the hydrostatic assumption, and warrants further study. For practical use, the hydrostatic assumption coupled with the depth-averaged to surface velocity empirical correlation should be sufficient.

CHAPTER 4

CONCLUSIONS

The mathematical framework for the estimation of river bathymetry using velocity data has been completed. A numerical solver for the adjoint variables has been developed and tested; this has been inserted into an algorithm that can be used to estimate the bathymetry in rivers. The algorithm has been exercised on multiple sets of data for two different reaches of two rivers. The results show that the proposed methodology can provide an accurate estimate of river bathymetry given surface velocity data. The effects of different types of data, three-dimensionality, and experimental errors have been assessed.

The mathematical framework, outlined in Section 2.2, is defined by first defining a cost function representing the error between modeled and observed surface velocities. The cost function is then constrained through the introduction of Lagrange multipliers representing the adjoint variables, multiplied by the shallow-water equations. Variational calculus is applied to the augmented cost function, resulting in the adjoint equations and the gradient with respect to the bathymetry.

Numerical tools have been developed to solve the adjoint equations and calculate the gradient of the bathymetry. Three important aspects of the adjoint solver, described in Section 2.3, were shown to be necessary to achieve stability requirements. Even though the adjoint equations appear similar in nature to the shallow-water equations, there are source terms involving the gradients of the adjoint variables that make existing numerical techniques unstable. The

algorithm incorporates an iterative gradient descent technique that drives the cost function to a minimum by adjusting the bathymetry.

In Chapter 3, algorithm results were presented for multiple sets of velocity data on two different rivers. The algorithm was proven to be robust for 2D and 3D data, with data resolutions varying from 3m to 40m and at data noise levels varying from 0cm/s to 40cm/s. For fully consistent depth-averaged, noise-free, full-resolution velocity data, the algorithm produces bathymetry estimates that are well within acceptable limits for the potential end-users. As expected, the algorithm produces bathymetry fields that are accurate up to a maximum depth, after which the water depth is under-estimated. This effect is amplified when using surface velocity data.

To allow the use of depth-averaged methodology given surface velocity observations, an empirical correlation was defined (in Section 3.1.3). The correlation was proven to be robust for the two rivers studied, resulting in very good bathymetric estimation. The empirical relationship was derived using a hydrostatic model, but the non-hydrostatic effects that are present in realistic rivers were shown to have no more impact on the algorithm than the effects of noise and resolution in typical observational data. The correlation that is implemented in the algorithm is consistent with previous observations of river velocity fields. The correlation has not been fully tested for robustness with respect to additional river types, such as steeper bottom slopes and varying bottom frictions. The two rivers used for model and algorithm testing represent typical, low-bottom slope meandering reaches. Further work would be necessary to expand the algorithm to braided reaches with significantly different conditions.

Overall, the bathymetry fields predicted by the algorithm result in RMS depth errors that lie between 0.3m and 1.3m RMS error, with the best model performance in low velocity and low depth regions. For the rivers investigated here, the low velocity and low depth regions of the rivers are notably small; in part, this explains the degraded performance in those regimes. Due to computational expense, the larger river model setup was at a lower resolution than the small reach. This model resolution affects the algorithm performance. Though not fully investigated, it appears as though a metric based on data resolution versus model resolution (or even river width) may aid in understanding algorithm performance.

This work represents the beginning of the development of a tool or component of a set of tools that will aid in practical situations such as directing dredging in large rivers after a storm even in which sediment transport has made traveling routes depths uncertain and most measurement techniques difficult (due to suspended sediment). Future planned development of this algorithm will revolve around exercising on actual experimental data of different types. This will greatly aid in the further understanding of the impact of modeling assumptions on the data and in the model's ability to estimate depth given that data. Additionally, the methodology can be applied to other aspects of river characterization, such as bottom friction estimation.

APPENDIX A

APPENDIX A SHALLOW WATER EQUATIONS

The shallow water equations (SWEs) can be derived from first principles or by integrating the Navier-Stokes equations. In this section the latter method will be used. The vertical coordinate system is defined in Figure 2.1.

Conservation of mass for fluid flow is defined by the continuity equation (in tensor notation, with Einstein summation convention)

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_j}{\partial x_j} = 0. \quad (\text{A.1})$$

Likewise, conservation of momentum, in its most general form is defined by the Navier-Stokes equations

$$\rho \left(\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} \right) = - \frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_i} + b_i, \quad (\text{A.2})$$

where τ_{ij} is the stress tensor for Newtonian fluids, given by

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \mu \frac{\partial u_i}{\partial x_i} \delta_{ij}. \quad (\text{A.3})$$

Here δ_{ij} is the Kronecker delta. It should be noted that b_i represents the summation of all body forces. In river flows, the effects of the Coriolis force are negligible such that the only significant body force is due to gravity. This is verified upon examination of the Rossby number, which is the ratio of inertial to Coriolis forces and is defined by

$$Ro = \frac{U}{fL}, \quad (\text{A.4})$$

where f is the Coriolis parameter, approximately 10^{-4} for intermediate latitudes. For large Rossby numbers, the effects of Coriolis can be considered negligible; for most rivers, the Rossby numbers are typically between 10 and 1000 (using the mean velocity magnitude and river width as the velocity and length scales).

River flow is incompressible but turbulent. The Reynolds numbers are very high due to the properties of water and the large length scales. For this reason, it is necessary to include the effects of turbulence on the mean flow. This is typically handled through the use of the Reynolds-averaged Navier-Stokes (RANS) equations. The equations are exact, though the quality of a RANS model solution is dependent on the turbulence closure used. In deriving the shallow-water equations, an eddy viscosity concept is presented here.

For incompressible fluid flows, assuming constant properties (density and viscosity), the continuity and RANS equations reduce to (here the overbar represents averaging)

$$\frac{\partial \bar{u}_j}{\partial x_j} = 0 \text{ and} \quad (\text{A.5})$$

$$\frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} + \nu \frac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j} - \overline{u'_i u'_j} + \bar{g}_i. \quad (\text{A.6})$$

The third term on the RHS is due to the Reynolds decomposition $(u_i = \bar{u}_i + u'_i)$, and can be modeled with an eddy viscosity such that

$$-\overline{u'_i u'_j} = \nu_t \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \frac{2}{3} k \delta_{ij}. \quad (\text{A.7})$$

The eddy viscosity (ν_t) and the turbulent kinetic energy (k) are calculated separately using an appropriate turbulence model. For this work, a two-equation k - ϵ model is used with

$$\nu_t = c_\mu \frac{k^2}{\varepsilon} . \quad (\text{A.8})$$

For more details and general information on turbulence modeling in hydraulic applications, see Rodi (2000). This results in the closed form of the RANS equations (here the overbar has been removed for simplicity)

$$\frac{\partial u_j}{\partial x_j} = 0 \text{ and} \quad (\text{A.9})$$

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j} \left(\nu_t \frac{\partial u_i}{\partial x_j} \right) + g_i . \quad (\text{A.10})$$

Utilizing the hydrostatic assumption (the vertical velocities are small relative to the horizontal velocities, and the acceleration of the vertical velocity is small relative to the other terms in the vertical momentum equation) and assuming gravity is in the x_3 direction ($g_3 = g$), reduces the u_3 momentum equation to

$$\frac{\partial p}{\partial x_3} = \rho g . \quad (\text{A.11})$$

Integrating over depth from η to an arbitrary depth z results in the hydrostatic pressure distribution

$$p = \rho g (\eta - z) . \quad (\text{A.12})$$

This leads to the pressure gradient in the horizontal momentum equations at depth z

$$\frac{\partial p}{\partial x_i} = \rho g \frac{\partial \eta}{\partial x_i} . \quad (\text{A.13})$$

Note that the horizontal gradient of z is zero, as this is simply a vertical coordinate. To obtain the depth-integrated SWEs, we integrate over depth. First, the continuity equation (applying the Leibnitz rule)

$$\begin{aligned}
0 &= \int_{-h}^{\eta} \frac{\partial u_j}{\partial x_j} dz \\
&= \int_{-h}^{\eta} \left\{ \frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \right\} dz + \int_{-h}^{\eta} \frac{\partial u_3}{\partial x_3} dz \\
&= \frac{\partial}{\partial x_1} \int_{-h}^{\eta} u_1 dz - u_1 \Big|_{z=\eta} \frac{\partial \eta}{\partial x_1} - u_1 \Big|_{z=-h} \frac{\partial h}{\partial x_1} \\
&\quad + \frac{\partial}{\partial x_2} \int_{-h}^{\eta} u_2 dz - u_2 \Big|_{z=\eta} \frac{\partial \eta}{\partial x_2} - u_2 \Big|_{z=-h} \frac{\partial h}{\partial x_2} \\
&\quad + u_3 \Big|_{z=\eta} - u_3 \Big|_{z=-h}.
\end{aligned} \tag{A.14}$$

At $z = -h$, we have a no-slip boundary condition, as well as no normal flow

$$u_i \Big|_{z=-h} = 0, \tag{A.15}$$

and at the surface ($z = \eta$), we also have no relative normal flow

$$\frac{\partial \eta}{\partial t} + u_1 \frac{\partial \eta}{\partial x_1} + u_2 \frac{\partial \eta}{\partial x_2} + u_3 = 0, \tag{A.16}$$

which reduces the depth-integrated continuity equation to

$$0 = \frac{\partial}{\partial x_1} \int_{-h}^{\eta} u_1 dz + \frac{\partial}{\partial x_2} \int_{-h}^{\eta} u_2 dz + \frac{\partial \eta}{\partial t}. \tag{A.17}$$

Defining the depth-averaged velocity as

$$U_i = \frac{1}{H} \int_{-h}^{\eta} u_i dz \tag{A.18}$$

further reduces the depth-integrated form of continuity to

$$\frac{\partial \eta}{\partial t} + \frac{\partial (HU_j)}{\partial x_j} = 0. \quad (\text{A.19})$$

For the depth-averaged momentum equations, we first augment the left-hand side with the product of the velocity and the continuity equation such that the left-hand side becomes (in conservative form)

$$\frac{\partial u_i}{\partial t} + \frac{\partial u_j u_i}{\partial x_j}. \quad (\text{A.20})$$

Then we integrate over depth (again, applying the Leibnitz rule):

$$\begin{aligned} & \int_{-h}^{\eta} \left\{ \frac{\partial u_i}{\partial t} + \frac{\partial u_j u_i}{\partial x_j} \right\} dz \\ &= \frac{\partial}{\partial t} \int_{-h}^{\eta} u_i dz + u_i \Big|_{z=\eta} \frac{\partial \eta}{\partial t} + u_i \Big|_{z=-h} \frac{\partial h}{\partial t} \\ &+ \frac{\partial}{\partial x_j} \int_{-h}^{\eta} u_j u_i dz + u_i u_j \Big|_{z=\eta} \frac{\partial \eta}{\partial x_j} + u_i u_j \Big|_{z=-h} \frac{\partial h}{\partial x_j}. \end{aligned} \quad (\text{A.21})$$

Next, we assume that the horizontal velocity profiles are uniform such that

$$\begin{aligned} & \frac{\partial}{\partial x_j} \int_{-h}^{\eta} u_j u_i dz + u_i u_j \Big|_{z=\eta} \frac{\partial \eta}{\partial x_j} + u_i u_j \Big|_{z=-h} \frac{\partial h}{\partial x_j} \\ & \approx H \frac{\partial U_j U_i}{\partial x_j} + U_j U_i \frac{\partial H}{\partial x_j} = \frac{\partial H U_j U_i}{\partial x_j}, \end{aligned} \quad (\text{A.22})$$

and utilize the surface and bottom boundary conditions for the velocity to reduce the left-hand side to

$$\frac{\partial H U_i}{\partial t} + \frac{\partial H U_j U_i}{\partial x_j}. \quad (\text{A.23})$$

For the right-hand side of the horizontal momentum equations, the integral form is

$$\begin{aligned}
& \int_{-h}^{\eta} \left\{ -g \frac{\partial \eta}{\partial x_i} + \frac{\partial}{\partial x_j} \left(\nu_t \frac{\partial u_i}{\partial x_j} \right) \right\} dz \\
& = -gH \frac{\partial \eta}{\partial x_i} + \int_{-h}^{\eta} \left\{ \frac{\partial}{\partial x_j} \left(\nu_t \frac{\partial u_i}{\partial x_j} \right) \right\} dz.
\end{aligned} \tag{A.24}$$

The first term results from the surface elevation being independent of depth, and the second term can be separated into two components: horizontal diffusion (dominated by turbulent diffusion) and vertical diffusion (dominated by surface and bottom stresses), such that

$$\int_{-h}^{\eta} \left\{ \frac{\partial}{\partial x_j} \left(\nu_t \frac{\partial u_i}{\partial x_j} \right) \right\} dz = \frac{\partial}{\partial x_j} \left(\overline{\nu_t} \frac{\partial U_i}{\partial x_j} \right) + \tau_{s,i} - \tau_{b,i}. \tag{A.25}$$

where the overbar on the eddy viscosity indicated a depth-averaged horizontal eddy viscosity. In many river flows, the effects of the wind stresses can be neglected at the surface, but the bottom stress is important. The bottom stress can be modeled in many ways, but we chose to use a Chézy formulation in junction with a Manning coefficient correlation. The bottom stress is then

$$\tau_{b,i} = \frac{\rho g U_i \left(U_j U_j \right)^{\frac{1}{2}}}{C_D^2}, \tag{A.26}$$

where n is the Manning coefficient, with units of $s/m^{1/3}$. Putting together the two sides of the momentum equations results in

$$\frac{\partial H U_i}{\partial t} + \frac{\partial H U_j U_i}{\partial x_j} = -gH \frac{\partial \eta}{\partial x_i} + \frac{\partial}{\partial x_j} \left(\overline{\nu_t} \frac{\partial U_i}{\partial x_j} \right) - g n^2 U_i \frac{\left(U_j U_j \right)^{\frac{1}{2}}}{H^{\frac{1}{3}}}. \tag{A.27}$$

The effects of the viscosity (both turbulent and molecular) are generally dominated by the bottom friction; when explicitly including the bottom friction, the additional turbulent effects are

small relative to the other terms on the right-hand side. Rearranging and recasting the depth-averaged momentum equations in non-conservative form, the SWEs are defined by

$$\frac{\partial \eta}{\partial t} + \frac{\partial (HU_j)}{\partial x_j} = 0 \quad \text{and} \quad (A.28)$$

$$\frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j} = -g \frac{\partial \eta}{\partial x_i} - gn^2 U_i \frac{(U_j U_j)^{\frac{1}{2}}}{H^{\frac{4}{3}}} . \quad (A.29)$$

REFERENCES

REFERENCES

- Bell, P. (1999). Shallow water bathymetry derived from an analysis of X-band marine radar images of waves. Coastal Engineering. vol. 37, pp. 513-527.
- Bennett, A. (2002). Inverse Modeling of the Ocean and Atmosphere. New York, New York: Cambridge University Press.
- Bernard, P. and Wallace, J. (2002). Turbulent Flow. Hoboken, New Jersey: John Wiley & Sons, Inc.
- Bertino, L, Evensen, G. and Wackernagel, H. (2003). Sequential data assimilation techniques in oceanography. International Statistical Review. vol. 71, no. 2, pp. 223-241.
- Chaudry, H. (2008). Open-Channel Flow. New York, New York: Springer Science+Business Media, LLC.
- Deltares (2011). Deflt3D-FLOW User Manual. Delft, The Netherlands. Deltares.
- Duan, J. and Nanda, S. (2006). Two-dimensional depth-averaged model simulation of suspended sediment concentration distribution in a groyne field. Journal of Hydrology. vol 327, pp. 426-437.
- Durand, M, Andreadis, K, Alsdorf, D, Lettenmaier, D, Moller, D. and Wilson, M. (2008). Estimation of bathymetric depth and slope from data assimilation of swath altimetry into a hydrodynamic model. Geophysical Research Letters. vol. 35, L20401, pp. 1-5.
- Egbert, G. and Erofeeva, S. (2002). Efficient inverse modeling of barotropic ocean tides. Journal of Atmospheric and Oceanic Technology. Vol. 19, pp. 183-204.
- Ferziger, J. and Perić, M. (2002). Computational Methods for Fluid Dynamics. Heidelberg, Germany: Springer-Verlag.
- Fonstad, M. and Marcus, W. (2005). Remote sensing of stream depths with hydraulically assisted bathymetry (HAB) models. Geomorphology. vol. 72, pp. 320-339.
- Fu, L. and Cazenave, A. (2001). Satellite Altimetry and Earth Sciences. San Diego, California: Academic Press.
- Garcia-Navarro, P. and Vasquez-Cendon, M. (2000). On numerical treatment of the source terms in the shallow water equations. Computers & Fluids. vol. 29, pp. 951-979.

- Goutal, N. and Maurel, F. (2002). A finite volume solver for 1D shallow-water equations applied to an actual river. International Journal for Numerical Methods in Fluids. vol. 38, pp. 1-19.
- Guerts, B. (1997). Inverse modeling for large-eddy simulation. Physics of Fluids. vol. 9, no. 12, pp. 3585-3587.
- Hilldale, R. and Raff, D. (2008). Assessing the ability of airborne LiDAR to map river bathymetry. Earth Surface Processes and Landforms. vol. 33, pp. 773-783.
- Hinze, J. (1975). Turbulence. New York, New York: McGraw-Hill.
- Jordan, D. and Fonstad, M. (2005). Two dimensional mapping of river bathymetry and power using aerial photography and GIS on the Brazos River, Texas. Geocarto International. vol. 20, no. 3, pp. 13-20.
- Kalkwijk, J. and Booij, R. (1986). Adaptation of secondary flow in nearly horizontal flow. Journal of Hydraulic Research. vol. 24, no. 1, pp. 19-37.
- Landon, K, Özkan-Haller, T. and Wilson, G. (2012). Depth inversion using drifter observations on the Kootenai River. 2012 Ocean Sciences Meeting. PosterID B0874.
- Legleiter, C, Roberts, D. and Lawrence, R. (2009). Spectrally based remote sensing of river bathymetry. Earth Surface Processes and Landforms. vol. 34, pp. 1039-1059.
- Li, Y, Navon, I, Courtier, P and Gauthier, P. (1993). Variational data assimilation with a semi-Lagrangian semi-implicit global shallow-water equation model and its adjoint. Monthly Weather Review. vol. 121, pp. 1759-1769.
- Muzaferija, S, Perić, M, Sames, P. and Schellin, T. (1999). A Two-fluid Navier-Stokes solver to simulate water entry. Twenty-Second Symposium on Naval Hydrodynamics. pp. 638-651.
- Navon, I. (1997). Practical and theoretical aspects of adjoint parameter estimation and identifiability in meteorology and oceanography. Dynamics of Atmospheres and Oceans. vol. 27, pp. 55-79.
- Nelson, J, Bennett, J. and Wiele, S. (2005). Flow and sediment transport modeling, in Tools in Fluvial Geomorphology (eds G. Kondolf and H. Piégay). John Wiley & Sons, Ltd. Chichester, UK. pp. 539-576.
- Nelson, J. and Smith, J. (1989). Flow in meandering channels with natural topography, in River Meandering. (eds S. Ikeda and G. Parker). Washington, D.C. American Geophysical Union. pp. 69-102.

- Patankar, S. (1981). A calculation procedure for two-dimensional elliptic situations. Numerical Heat Transfer. vol. 4, no. 4, pp. 409-425.
- Plant, W, Keller, W, Hayes, K. and Spicer, K. (2005). Streamflow properties from time series of surface velocity and stage. Journal of Hydraulic Engineering. vol. 131, no. 8, pp. 657-664.
- Pope, S. (2000). Turbulent Flows. New York, New York: Cambridge University Press.
- Rantz, S. (1982). Measurement and computation of streamflow. U.S. Geological Survey Water Supply Paper 2175. Washington, DC. U.S. Government Printing Office.
- Rekanos, I. and Räisänen, A. (2003). Microwave imaging in the time domain of buried multiple scatterers by using an FDTD-based optimization technique. IEEE Transactions on Magnetics. vol. 39, no. 3, pp. 1381-1384.
- Rodi, W. (2000). Turbulence Models and Their Applications in Hydraulics. International Association for Hydraulic Research (IAHR) Monograph Series. Third Edition. A. A. Balkema, Rotterdam, Netherlands.
- Sanders, B. and Katopodes, N. (2000). Adjoint sensitivity analysis for shallow-water wave control. Journal of Engineering Mechanics. vol. 126, no. 9, pp. 909-919.
- Sandwell, D. and Smith, W. (2001). Bathymetric estimation. Satellite Altimetry and Earth Sciences. (eds L. Fu and A. Cazenave). Academic Press. pp. 441-457.
- Smith, L. and Pavelsky, T. (2008). Estimation of river discharge, propagation speed, and hydraulic geometry from space: Lena River, Siberia. Water Resources Research. vol. 44, W03427, pp. 1-11.
- Steffler, P. and Blackburn, J. (2002). River2D: Two-dimensional depth averaged model of river hydrodynamics and fish habitat. Introduction to depth averaged modeling and user's manual. Edmonton, University of Alberta.
- Stelling, G. and Duinmeijer, S. (2003). A staggered conservative scheme for every Froude number in rapidly varied shallow water flows. International Journal for Numerical Methods in Fluids. vol. 43, pp. 1329-1354.
- Tannehill, J, Anderson, D. and Pletcher, R. (1997). Computational Fluid Mechanics and Heat Transfer. Washington, DC. Taylor & Francis.
- Valle-Levinson, A, Reyes, C. and Sanay, R. (2003). Effects of bathymetry, friction, and rotation on estuary-ocean exchange. Journal of Physical Oceanography. vol. 33, pp. 2375-2393.

- van Donegren, A, Plant, N, Cohen, A, Roelvink, D, Haller, M. and Catalan, P. (2008). Beach Wizard: Nearshore bathymetry estimation through assimilation of model computations and remote observations. Coastal Engineering. vol. 55, pp. 1016-1027.
- Wang, Z, Navon, I, Le Dimet, F. and Zou, W. (1992). The second order adjoint analysis: Theory and applications. Meteorological and Atmospheric Physics. vol. 50, pp. 3-20.
- Yang, Z. and Khangaonkar, T. (2009). Modeling tidal circulation and stratification in Skagit River estuary using an unstructured grid ocean model. Ocean Modelling. vol. 28, pp. 34-49.