

MONTE CARLO BASED FLOOD RISK ANALYSIS USING A GRAPHICS
PROCESSING UNIT-ENHANCED TWO-DIMENSIONAL FLOOD MODEL

by

Alfred J. Kalyanapu

A dissertation submitted to the faculty of
The University of Utah
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Civil and Environmental Engineering

The University of Utah

August 2011

Copyright © Alfred J. Kalyanapu 2011

All Rights Reserved

The University of Utah Graduate School

STATEMENT OF DISSERTATION APPROVAL

The dissertation of Alfred J. Kalyanapu
has been approved by the following supervisory committee members:

| | | |
|-----------------------------|----------|------------------------------------|
| <u>Steven J. Burian</u> | , Chair | <u>05/19/2011</u> Date Approved |
| <u>Christine A. Pomeroy</u> | , Member | <u>05/31/2011</u> Date Approved |
| <u>Brian J. McPherson</u> | , Member | <u>05/19/2011</u> Date Approved |
| <u>Eric R. Pardyjak</u> | , Member | <u>05/19/2011</u> Date Approved |
| <u>Timothy N. McPherson</u> | , Member | <u>05/19/2011</u> Date Approved |

and by Paul J. Tikalsky, Chair of
the Department of Civil and Environmental Engineering

and by Charles A. Wight, Dean of The Graduate School.

ABSTRACT

The goal of this dissertation is to improve flood risk management by enhancing the computational capability of two-dimensional models and incorporating data and parameter uncertainty to more accurately represent flood risk. Improvement of computational performance is accomplished by using the Graphics Processing Unit (GPU) approach, programmed in NVIDIA's Compute Unified Development Architecture (CUDA), to create a new two-dimensional hydrodynamic model, Flood2D-GPU. The model, based on the shallow water equations, is designed to execute simulations faster than the same code programmed using a serial approach (i.e., using a Central Processing Unit (CPU)). Testing the code against an identical CPU-based version demonstrated the improved computational efficiency of the GPU-based version (approximate speedup of more than 80 times).

Given the substantial computational efficiency of Flood2D-GPU, a new Monte Carlo based flood risk modeling framework was created. The framework developed operates by performing many Flood2D-GPU simulations using randomly sampled model parameters and input variables. The Monte Carlo flood risk modeling framework is demonstrated in this dissertation by simulating the flood risk associated with a 1% annual probability flood event occurring in the Swannanoa River in Buncombe County near Asheville, North Carolina. The Monte Carlo approach is able to represent a wide range of possible scenarios, thus leading to the identification of areas outside a single simulation

inundation extent that are susceptible to flood hazards. Further, the single simulation results underestimated the degree of flood hazard for the case study region when compared to the flood hazard map produced by the Monte Carlo approach.

The Monte Carlo flood risk modeling framework is also used to determine the relative benefits of flood management alternatives for flood risk reduction. The objective of the analysis is to investigate the possibility of identifying specific annual exceedance probability flood events that will have greater benefits in terms of annualized flood risk reduction compared to an arbitrarily-selected discrete annual probability event. To test the hypothesis, a study was conducted on the Swannanoa River to determine the distribution of annualized risk as a function of average annual probability. Simulations of samples of flow rate from a continuous flow distribution provided the range of annual probability events necessary. The results showed a variation in annualized risk as a function of annual probability. And as hypothesized, a maximum annualized risk reduction could be identified for a specified annual probability. For the Swannanoa case study, the continuous flow distribution suggested targeting flood proofing to control the 12% exceedance probability event to maximize the reduction of annualized risk. This suggests that the arbitrary use of a specified risk of 1% exceedance may not in some cases be the most efficient allocation of resources to reduce annualized risk.

TABLE OF CONTENTS

| | |
|---|------|
| ABSTRACT..... | iii |
| LIST OF TABLES..... | vii |
| ACKNOWLEDGEMENTS..... | viii |
| Chapter | |
| 1. INTRODUCTION | 1 |
| 1.1 Background..... | 1 |
| 1.1.1 Flood Impacts..... | 1 |
| 1.1.2 Flood Modeling..... | 4 |
| 1.1.3 Uncertainty in Flood Modeling..... | 9 |
| 1.2 Problem Statement..... | 11 |
| 1.3 Research Objectives..... | 13 |
| 1.4 Overview of Dissertation | 13 |
| 2. GRAPHICS PROCESSING UNIT-BASED TWO-DIMENSIONAL FLOOD MODEL..... | 15 |
| 2.1 Introduction..... | 15 |
| 2.2 Methodology..... | 21 |
| 2.2.1 Model Description..... | 21 |
| 2.2.2 Model Validation..... | 26 |
| 2.2.2.1 Laboratory Scale Dam Break..... | 26 |
| 2.2.2.2 Taum Sauk Dam Break..... | 28 |
| 2.2.3 Model Speedup..... | 30 |
| 2.3 Results and Discussion..... | 31 |
| 2.3.1 Model Validation..... | 31 |
| 2.3.1.1 Laboratory Scale Dam Break Simulation..... | 31 |
| 2.3.1.2 Taum Sauk Dam Break Simulation..... | 37 |
| 2.3.2 Model Speedup Calculation..... | 39 |
| 2.3.3 Effect of Domain Minimization..... | 41 |
| 2.3.4 Effect of Spatial Resolution..... | 42 |
| 2.3.5 Speedup Comparison to Other Studies..... | 44 |
| 2.3.6 Limitations of GPUs..... | 45 |

| | | |
|-----------------|--|-----|
| 2.4 | Conclusions..... | 47 |
| 3. | MONTE CARLO BASED FLOOD MODELING FRAMEWORK FOR ESTIMATING PROBABILITY WEIGHTED FLOOD RISK..... | 49 |
| 3.1 | Introduction..... | 49 |
| 3.2 | Methodology..... | 53 |
| 3.2.1 | Monte Carlo Analysis..... | 53 |
| 3.2.2 | Geospatial Output Analysis..... | 56 |
| 3.2.3 | Risk Map Development..... | 57 |
| 3.3 | Case Study..... | 60 |
| 3.4 | Results..... | 63 |
| 3.5 | Summary..... | 69 |
| 4. | ANNUALIZED RISK ANALYSIS APPROACH TO RECOMMEND APPROPRIATE LEVEL OF FLOOD CONTROL..... | 71 |
| 4.1 | Introduction..... | 71 |
| 4.2 | Methodology..... | 77 |
| 4.2.1 | General Flood Damage Analysis Approach..... | 77 |
| 4.2.2 | Flood Damage Approach in This Study..... | 79 |
| 4.2.2.1 | Input Flow Estimation..... | 79 |
| 4.2.2.2 | Hydraulic Modeling..... | 82 |
| 4.2.2.3 | Damage Modeling..... | 84 |
| 4.3 | Case Study..... | 87 |
| 4.4 | Results and Discussion..... | 91 |
| 4.4.1 | No Flood Proofing Alternative..... | 91 |
| 4.4.2 | Flood Proofing Alternatives..... | 94 |
| 4.4.3 | Capital Cost Estimates..... | 97 |
| 4.5 | Summary..... | 100 |
| 5. | CONCLUSIONS..... | 102 |
| APPENDICES | | |
| A. | FIRST-ORDER UPWIND NUMERICAL DISCRETIZATION OF THE ST.VENANT EQUATIONS..... | 107 |
| B. | IMPLEMENTATION STEPS FOR CPU AND GPU FLOOD MODEL FRAMEWORKS..... | 115 |
| REFERENCES..... | | 121 |

LIST OF TABLES

| <u>Table</u> | | <u>Page</u> |
|--------------|--|-------------|
| 2.1 | Comparison of the configuration of the GPU machines..... | 26 |
| 2.2 | Quantified statistics for laboratory scale dam simulation..... | 35 |
| 2.3 | GPU speedup results for Taum Sauk dam break simulations for three sets of model iterations..... | 39 |
| 2.4 | Execution times before and after domain optimization..... | 42 |
| 2.5 | Comparative summary of recent model speeds in the literature for storage cell (SC), dynamic wave (DYN), and diffusive wave (DIF) models..... | 45 |
| 3.1 | Error matrix comparing the flood risk maps using deterministic approach (Map A) and deterministic approach (Map B)..... | 65 |
| 4.1 | Estimated annual damages calculated for the three cases..... | 97 |
| 4.2 | Calculation of cost estimates for implementing flood proofing..... | 98 |

ACKNOWLEDGEMENTS

I would like to express my gratitude to my research advisor, Dr. Steven Burian, for all his invaluable guidance, patience and continued support throughout my graduate study. I also sincerely thank him for giving me many opportunities that help me to grow professionally.

I would also like to thank all my committee members, Dr. Timothy McPherson, Dr. Brian McPherson, Dr. Christine Pomeroy and Dr. Eric Pardyjak, for investing their time, effort and support. I would also like to thank all the fellow students and researchers, past and present, that I have interacted with and befriended here at the University of Utah. Especially, I would like to thank Siddharth Shankar for his valuable collaboration in developing the GPU model, and Dr. Dave Judi for his expert suggestions during the dissertation.

I would also like to thank my parents, my two sisters, my brother-in-laws, my brother and my sister-in-law for their support, thoughts and love throughout my graduate studies. I extend a special thanks to my cute nephews and two nieces for their love and affection.

Finally, I would like to thank Tania, whose influence on me I cannot quantify, who was my inspiration to pursue my doctoral degree, who stood by me during the thick and thin, and believed in me even when I did not believe in myself.

CHAPTER 1

INTRODUCTION

1.1 Background

1.1.1 Flood Impacts

Knowing that more than 2.8 billion of the world's population lives within 15 km of rivers (Small and Cohen, 2004), it is no surprise that floods are one of the more frequently occurring and higher impact natural disasters. Every year, on average, 196 million people in more than 90 countries experience to some degree catastrophic flooding. From 1980 to 2000, floods resulted in more than 170,000 deaths worldwide, an average of nearly 9,000 deaths per year (UNDP, 2004). Although efforts have been directed towards flood management and control, the problem continues to worsen as populations in riparian and coastal areas grow, as more people move to riparian and coastal areas, and as climate patterns change (Burby, 2001; McCarthy et al., 2001; Montz and Grunfest, 2002). Flood risk mitigation is a major challenge facing local, regional and global disaster management agencies (Levy et al., 2005).

Mitigating flood risk can be achieved in today's heavily urbanized world by appropriately using the floodplains through floodplain management practices (Ahmad and Simonovic, 2006; Bedient et al., 2008). In the United States (US), flood management practices have been directed towards flood control, mainly through the

construction of levees by the US Army Corps of Engineers, especially with the passage of the Flood Control Acts of 1917 and 1936 (Wright, 2000). This has evolved significantly over the last 75 years with the inclusion of various structural flood control measures like detention basins, levees, dams and nonstructural measures like flood proofing, permanent evacuation and relocation, land use management, flood hazard maps, hydrologic/hydraulic models, early warning systems, evacuation procedures, building codes and National Flood Insurance Program (NFIP) (Levy et al., 2005; Ahmad and Simonovic, 2006).

However, achieving complete protection from extreme flood events is not technically feasible or economically viable, because one cannot predict the exact flood magnitude and frequency to be able to design flood control with appropriate specifications, because flood events are inherently random and vary in space and time and also with changing climate. For example, during the great USA flood of 1993 in the upper Mississippi River basin which was protected by at least 1572 federal and nonfederal levees along the river, 40 levees under federal operation and 1043 nonfederally operated levees failed or overtopped across the states of Minnesota, Illinois, Missouri and Nebraska, resulting in total estimated damages up to \$20 billion and destroying more than 50,000 homes (Johnson et al., 2004). Thus, in the recent years, there is shift in the perspective of flood policy from the concept of “flood protection” to “flood risk management” (Schanze, 2006).

Flood protection aims at preventing flood hazards up to a certain flood magnitude by providing a certain protection level (e.g., protection against floods with a magnitude equivalent to a 1% exceedance probability). Flood risk management, on the

other hand focuses on avoiding flood risk or minimizing the impacts of floods where flood damages cannot be avoided, by using a combination of floodplain management practices. In the US, with this goal in mind, the US Army Corps of Engineers (USACE) established the National Flood Risk Management Program (NFRMP). The goal of NFRMP is to create a collaborative platform for all the federal, state and local agencies to develop strategies using a combination of floodplain management practices in reducing flood risk. One of the examples of flood risk management implementation in the US is the Swannanoa Flood Risk Management Project (SFRMP), in the Swannanoa watershed in the state of North Carolina. Established in 2005, its objective is to reduce the flood damages and risk of economic losses related to flooding in the Swannanoa valley area. Based on the preliminary engineering studies and community outreach, 50 potential flood risk management projects were identified, including flood warning stations, dam rehabilitation, emergency spillway reconstruction for higher discharge capacity, improvements to river approach and exit of bridges, and flood wall construction. More information about this project can be found at their webpage, <http://www.swannanoafloods.org>.

Continued improvement in floodplain management practices and managing flood risk requires effective public education of risk, careful planning based on estimates of flood extent and reliable fast information exchange with the public during a flood event emergency (FEMA, 2002). The success of these actions is based on accurate and rapid prediction of flood inundation, which is achieved by flood models (Bates et al., 2004; Bates et al., 2005). By improving the prediction of flood depths, velocities and flood inundation extent, one can apply these results with confidence in

the design of flood management alternatives like dams, storage basins, flood walls, levees, enhance the performance of early flood warning systems, understand the impacts of urbanization and climate change on the increase in floods, and much more.

1.1.2 Flood Modeling

The advancement of computer models to simulate floods has had significant impact on the ability to plan, forecast and respond to flood events (Hunter et al., 2007). In the US, hydrologic and hydraulic models were first applied to simulate floods in the 1960s (Crawford and Linsley, 1966), and have since been used to enhance engineering design, planning, floodplain delineation and emergency response. These models range from simple mathematical equations (e.g., rational method) to complex and computationally challenging numerical solutions of partial differential equations, also varying in predictive capacity. The current state of flood modeling involves using computer-based flood models that are freely available like the Hydrologic Engineering Center's River Analysis System (HEC-RAS), the National Center for Computational Hydroscience and Engineering's (NCCHE) CCHE1D, and the US Geological Survey's (USGS) Multi-Dimensional Surface-Water Modeling System (MD_SMS). There are also commercially available flood models like MIKE11, MIKE-FLOOD by DHI Group, FLO-2D by FLO-2D Software, Inc., etc., that are used in flood modeling practice. Most of these models have the capability to integrate with Geographic Information Systems (GIS) and storing and managing spatial data related to floods, using software packages like ArcGIS[®], MapWindow and MapInfo[®].

Flood models typically simulate floods as free surface flows, using one-dimensional (1D) dynamic wave simulations, also known as Saint Venant equations. The 1D representation assumes that flow is parallel to the river channel or floodplain, i.e., flow lateral to the main channel or perpendicular to the main channel flow direction does not occur. It represents the river channel and the floodplain as a series of cross-sections perpendicular to the flow direction. 1D models have been a commonly used class of flood models as they are relatively simple to build and easy to operate (Fread, 1985). They are also computationally efficient and many engineers consider their use acceptable to produce reasonably accurate surface water profiles (Buchele et al., 2006). Examples of 1D flood models include the US Environmental Protection Agency's Storm Water Management Model (SWMM), the US Army Corps of Engineer's HEC-RAS, MIKE11 by DHI Group, FLDWAV model by the US National Weather Service and, WSPRO developed by the USGS. Solving these models for a flood event on a river reach results in flood depths and velocities at all the cross-sections along the river. The flood depths can be interpolated to a water surface elevation, which can be overlaid on a digital surface of ground elevations (e.g., a digital elevation model in an ESRI® grid format, a triangular irregular network). Total flood inundation extent can be derived from all locations with water depths above the ground surface, as shown in Figure 1.1 (Merwade et al., 2008).

Even though 1D models are commonly used, the approach of 1D models is not always appropriate, especially when applied to floodplain flows. They fail to simulate the lateral diffusion of the flood wave. And the discretization of the topography as cross-sections, instead of a continuous surface, is responsible for model uncertainties in

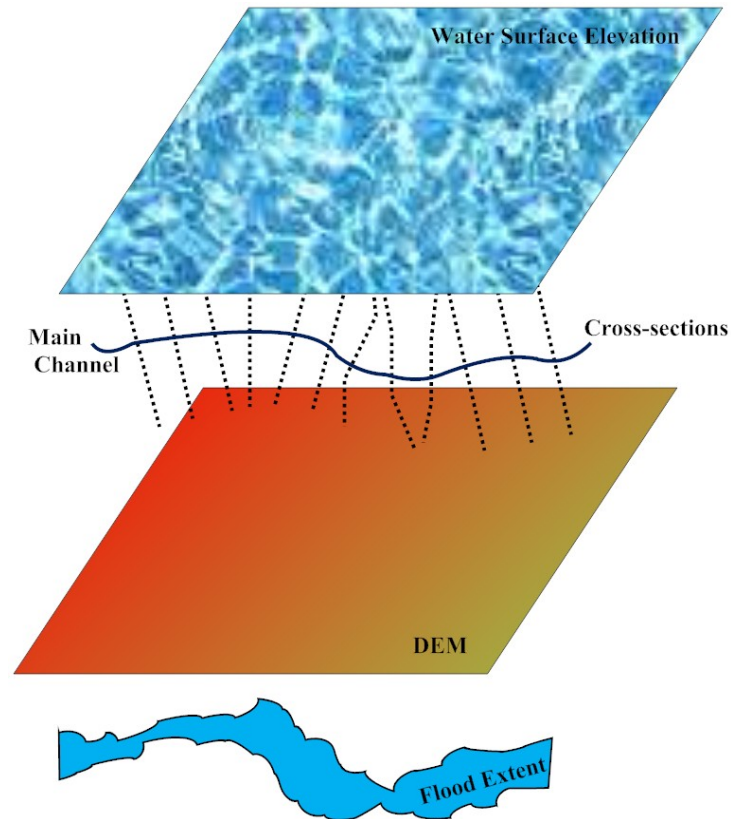


Figure 1.1: Flood inundation mapping process for 1D flood models

sinuous channels. The location of cross-sections is also subjective and may potentially affect the simulated extent of floods (Samuels, 1990). Flood inundation extent extraction through 1D models: 1) is not a seamless process and requires postprocessing and 2) is subjective because water surface elevations are generated through interpolations (Bates and De Roo, 2000).

Two-dimensional (2D) flood models eliminate these limitations, and various 2D numerical schemes have been developed in response (Zhang and Cundy, 1989; Lamb et al., 2009; Judi 2009). Applying 2D models enables higher order topographic representation in the simulations. A postprocessing step is not required for calculating flood inundation extent because all the locations with flood depths collectively form the

flood inundation extent (Bates et al., 1995). Moreover, with the increasing availability of high resolution and high accuracy Digital Elevation Models (DEM) for floodplains areas, 2D models can be readily integrated with such data sources (Marks and Bates, 2000). Most researchers agree that, because flows in the floodplain significantly increase in complexity, models based on higher-order equations, such as 2D or 3D equations, should be used (Knight and Shiono, 1996; Bates et al., 1998). The recommendation of NRC (2009) towards Federal Emergency Management Authority (FEMA) was to promote “greater use of 2D hydraulic models”, where needed by the floodplain topography, including preferential flood pathways and existing and planned structures.

A major limiting factor for applying 2D numerical flood models is their computational intensity (Lamb et al., 2009; Judi, 2009). Despite the advances in computer hardware and technology, it can still take a long time to run 2D models. For example, a Central Processing Unit (CPU) based 2D dynamic wave flood model using a first-order upwind finite difference numerical scheme was used to simulate a dam break event for a 62 km² area (624 x 1136 grid cell domain at 9.36 m spatial resolution), on a 2.33 GHz Intel™ Core2Duo® desktop with 2 GB RAM and Windows XP Professional Operating System (basic system characteristics expected for most modelers). The model took 9.1 h to simulate approximately 15 min of flood wave. This limits the number of scenarios, spatial extent and/or level of detail that is expected for a particular flood problem, especially in the development of Monte Carlo based risk assessment methods for flood modeling (Sayers et al., 2000; Buijs et al., 2003; Lamb et al., 2009; NRC, 2009). They require multiple scenario simulations that tremendously increase the

computational intensity in generating flood depths, velocities, extents etc. For discussion's sake, let us consider a flood emergency and evacuation study for the above-mentioned dam break simulation. The lead time for flood warning is very small, maybe around 2 – 3 hours, but it takes 10 hours to run the numerical model. So, model application would not be sufficient for emergency and evacuation study.

Computational intensity is further affected by using higher resolution digital topographic data, and by applying models at regional scales. Bates and de Roo (2000) found that performance of models with a 100 m grid deteriorated significantly when compared to a 25 m grid, and this was much less for a 50 m grid. So, there is a difficult choice for the modeler, whether to use a lower spatial representation and not take advantage of enhanced topographic data or to use a higher resolution but suffer with computational intensity. In addition to computational intensity, flood estimations from models that can be run quickly, while maintaining accuracy, can be used in a more “near or better” than real-time fashion to include dynamic conditions and enhance the emergency management and decision-making capability.

2D flood models have been implemented in a high-performance parallel computing architecture including Flo2DH by Hluchy et al., (2002), RMA by Rao (2005), CalTWiMS by Pau and Sanders (2006), TRENT by Villanueva and Wright (2006), FloodMap-Parallel by Yu (2010) and LISFLOOD-FP by Neal et al. (2009). However, converting existing codes to execute on a parallel programming computer cluster is a complex process requiring significant programming effort (Tran and Hluchy, 2004), and high-performance computing infrastructure is typically limited to academic and

government institutions, limiting their availability to private consulting and engineering firms. There is a need for a simpler and a cost effective way to build 2D flood models.

1.1.3 Uncertainty in Flood Modeling

Numerical models are but conceptualizations of reality and hence reduce physical complexity by simplifications through the systems of equations (Wagener and Gupta, 2005). The model parameters, input data and model structure are calibrated with observed data, and when needed, the parameters are altered or forced (within their acceptable ranges) to fit the model simulations to the observed data, and the validated model is used for prediction, as illustrated in Figure 1.2.

However, prior estimation of feasible ranges of parameters does not guarantee the model prediction within a close range of observations, especially when it is extrapolated to other problem locations, like in the case of the two red dots outside the observation data in Figure 1.2. The lack of correlation between model parameters and physical floodplain characteristics results in significant uncertainty in prediction, especially if the model is extrapolated to predict the system behavior at a different location and/or flood event.

Flood model uncertainty is thus of critical concern when modeling results are used to set policy, decision making and emergency planning. Failure to acknowledge uncertainty could result in wasteful overdesign of flood protection/mitigation systems, or could lead to inadequate preparation for potential situations and even failure of these systems.

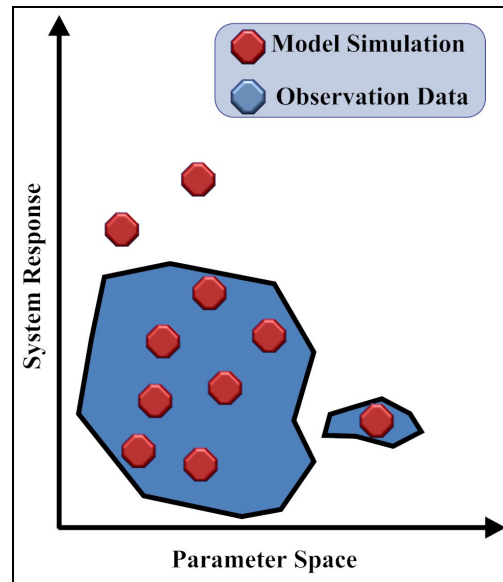


Figure 1.2. Typical approach for model calibration, validation and prediction (*adapted from Smith, 2007*)

Thus, a careful and detailed calibration and uncertainty analysis is required for a successful application of flood inundation models (Duan et al., 1992; Beven and Binley, 1992; Yang et al., 2008). For the past two decades, several studies on uncertainty analysis of models have been conducted, including Aronica et al. (2002), Bates et al. (2004), Werner et al. (2005), Pappenberger et al. (2005), studying the effect of surface roughness coefficient, Aronica et al. (1998), the grid cell size, and Purvis et al. (2008) for flow characteristics etc.

However, the common representation of flood model simulation results remains a deterministic flood inundation map based on a single simulation, or at best, a few scenarios (Apel et al., 2006). These limitations in the analysis have been due to the lack of data, and lack of higher dimensional modeling capabilities. Unfortunately, these deterministic approaches rely on the use of a single or limited parameter sets and does not account for the uncertainties in the modeling process (Bates et al., 2004) and may

lead to an inaccurate hazard assessment (Di Baldassarre et al., 2010). To remedy the shortcomings of a single deterministic simulation of a floodplain, probabilistic modeling approaches are emerging. Probabilistic flood mapping, using Monte Carlo framework, is designed to incorporate uncertainty from input data and model parameters, represent spatial and temporal risk, and present flood maps in terms of probabilities and percentages (Aronica et al., 2002; Romanowicz and Beven, 2003; Bates et al., 2004; Hall et al., 2005; Pappenberger et al., 2006; Di Baldassarre et al., 2010). However, these approaches are mainly based on simple planar hydraulic models and model implementation using the full shallow water wave equations has not been found in the literature. Incorporating a physically-based 2D hydraulic model would improve the model evaluations because of the improved spatial representation and accuracy of flood depths and velocities compared to 1D and simplified 2D planar models (Bates et al., 2004). This would also aid in better understanding the sources of flood risk that is essential for flood risk management.

1.2 Problem Statement

Advancement of computer models to simulate floods has significant impacts on the ability to plan, forecast and respond to flood events. 1D flood models are commonly used in flood modeling applications; some of them, like HEC-RAS, WSPRO etc., are even available for free. However, they are prone to limitations, including simplified topographic representation through cross-sections, interpolation in water surface elevation generation and problems simulating complex lateral floodplain flows, etc. 2D flood models alleviate these limitations with their ability to simulate complex flows and

with higher order topographic representation. However, a major limiting factor for applying 2D numerical flood models is their computational intensity. High-performance computing facilities are available to only a few academic research institutes and a few consulting firms because of tremendous financial costs related to installing and maintaining these computing facilities. There is a need to search for simpler and more cost-effective alternatives.

Another critical challenge facing flood modeling is the uncertainty involved in modeling floods. It is a significant challenge, especially when models are used in determining policy, decision making and emergency planning. Failure to incorporate uncertainty could potentially result in wasteful overdesign of flood protection/mitigation measures, or even worse, lead to inadequate preparation for flood protection/mitigation. Incorporating uncertainty is imperative for successful use of hydrologic and hydraulic models. However, flood models are being operated in a deterministic (or single simulation) fashion relying on the use of a single or limited parameter sets and they in general do not account for the uncertainties in the modeling process. Probabilistic modeling approaches are emerging to reduce this limitation but these approaches are mainly based on simple planar hydraulic models (Di Baldassarre et al., 2010). Flood models using the full shallow water wave equations have not been found in the literature. Thus, there is a need to use physically-based 2D flood models in a probabilistic framework and demonstrate the usefulness in flood risk management.

1.3 Research Objectives

The goal of the proposed research is to improve flood risk management by enhancing flood model computational capability and incorporating flood modeling uncertainty and demonstrating the importance of acknowledging uncertainty in flood risk applications. This is accomplished by (1) developing a computationally efficient Graphics Processing Unit (GPU) based 2D flood model by using an efficient and robust upwind numerical scheme to solve the complete 2D Saint Venant equations, (2) developing a Monte Carlo based probabilistic flood modeling framework to incorporate data and parameter uncertainties and generate probability weighted flood risk, and (3) applying the Monte Carlo based framework to study the benefits of implementing flood risk management alternatives.

1.4 Overview of Dissertation

The proposed goals of the research and the details of the methods and results are presented in the following chapters. The research included the development of a computationally efficient GPU-based 2D flood model. It is explained in the Chapter 2 and information is provided on the GPU framework, the development of a flood model and the optimization of the model for computational performance. The flood model is validated by comparing results from a laboratory exercise to show its ability to accurately estimate flood depths and velocities. The model is then further validated with a case study using high water mark data collected for the Taum Sauk dam break event. The computational improvement of the GPU-based flood model is observed, and the performance of GPU as a function of domain size and spatial resolution is explored.

Chapter 3 focuses on development of a new Monte Carlo based probabilistic flood modeling framework. The framework is used to incorporate data and parameter uncertainty and estimate probability weighted flood risk. The chapter includes a description of the framework and an application to determine the flood risk of the Swannanoa River in North Carolina. The ability of the framework in providing detailed flood risk compared to a deterministic flood modeling approach is demonstrated. Chapter 4 focuses on the application of the probabilistic flood modeling framework to the analysis of flood risk management alternatives. The financial impact in terms of reduction in flood damages by implementing flood proofing for different design flood events is studied. The annualized risk concept is used in the formulation of design alternatives. Chapter 5 summarizes the results of the dissertation and presents the conclusions.

CHAPTER 2

GRAPHICS PROCESSING UNIT-BASED TWO-DIMENSIONAL FLOOD MODEL

2.1 Introduction

With more than 21% of the world's population living within 30 km of the coast (Gommes et al., 1997), catastrophic flood events continue to cause an increasing amount of casualties, economic impact and infrastructure damage. On average, 196 million people in more than 90 countries are exposed to flooding each year (UNDP, 2004), while in the United States (US), by 2005, flood damages increased to \$ 6 billion per year (FEMA, 2002; Levy et al., 2005). While already significant, Pielke et al. (2002) have analyzed US flood damage statistics between 1934 and 2000, found them to be increasing and projected them to continue to increase because of population growth and migration patterns and changing storm event patterns.

Floodplain management actions are implemented to prevent and mitigate flood impacts on humans, ecology and the economy (Ahmad and Simonovic, 2006). Flood management has evolved over the last 50 years with key areas of advance including development of flood hazard maps, hydrologic/hydraulic models, flood warning systems, evacuation procedures, and flood insurance programs (Levy et al., 2005). Continued improvement requires effective public education of flood risk, careful planning based on

estimates of flood inundation extent and reliable and fast information exchange with the public during an emergency (FEMA, 2002). The success of these actions is in part based on accurate and rapid flood modeling. Flood modeling can be used in various flood management activities, including engineering design, planning, floodplain delineation and emergency response. Complete flood inundation analysis requires the estimation of flood flows (e.g., rainfall-runoff generated, dam break, etc.), hydraulic modeling to route the flow and compute water surface elevations and flow velocities, and analysis tools to delineate the flood inundation extent and assess impacts (Knebl et al., 2005; Merwade et al., 2008).

Computer models have been applied to simulate floods for more than four decades (Crawford and Linsley, 1966). Because of the ease of use and efficiency, floods are typically modeled using a one-dimensional (1D) dynamic wave approach (e.g., Hydrologic Engineering Center River Analysis System or HEC-RAS, MIKE 11, etc.). However, 1D models are not always appropriate because of the inability to simulate the lateral diffusion and inaccuracies due to cross-section discretization (Samuels, 1990; Bates and De Roo, 2000). Two-dimensional (2D) models (Zhang and Cundy, 1989; Tayfur et al., 1993; Lamb et al., 2009) eliminate the primary limitations of 1D models (Samuels, 1990; Bates and De Roo, 2000; Marks and Bates, 2000; García – Navarro et al., 2008; Judi et al., 2010) by enabling higher order topographic representation and preferential flood pathways in the simulations (Bates et al., 1992; Bates et al., 1995; NRC, 2009). Therefore, the National Research Council (NRC, 2009) has recommended that the Federal Emergency Management Agency (FEMA) promote “greater use of 2D

hydraulic models” where warranted by the floodplain topography, including preferential flood pathways and existing and planned structures.

Although the advantages of 2D models are well known in the research community, their computational intensity and complexity remain major limiting factors for modeling practitioners (Lamb et al., 2009; Judi et al., 2010). Indeed, despite past advances in computer hardware and technology, 2D simulations still require substantial time to complete, potentially preventing their application to meet modeling time constraints. Different flood management objectives require different lead times (time period between initiation and termination of the flood management project), as shown in Figure 2.1. In general, the necessary computational intensity of a flood model increases with the level of detail and turnaround time needed for a particular flood management activity. It is apparent from Figure 2.1 that for flood risk and warning applications, the average modeling time must be less than a few hours, which might not be possible for 2D models. Recent advances in model speed and computational power may provide the ability to run a single flood simulation using a 2D model for smaller areas within designated time constraints. However, the current trend towards the use of Monte Carlo methods that involve many simulations for generating flood risk information mandates the need for even faster models (Sayers et al., 2000; Buijs et al., 2003; Lamb et al., 2009; NRC, 2009).

One approach to improve the computational performance of 2D flood models is simplifying the numerical code, such as using the horizontal projection of water levels that depart significantly from the physical basis of a hydrodynamic model. Another method eliminates the convective terms in the momentum equation (Lamb et al., 2009).











|  Scale |  Level of Detail |  Project Duration |  Model Time |
|--|---|--|--|
|  <p>National</p> |  <p>National Policy Risk Communication</p> | <p>Approx. 6-12 months Years</p> | <p>Weeks Months</p> |
|  <p>Regional</p> |  <p>Post event analysis Flood Warning Floodplain Maintenance</p> | <p>Detailed 6-12 months 12-24 hours 3-4 weeks</p> | <p>Weeks 2-3 hours Days</p> |
|  <p>Local</p> |  <p>Design Project Risk Communication</p> | <p>Very Detailed Months-years Hours</p> | <p>Months Hours</p> |

Figure 2.1. Activity duration and modeling time scale for different flood modeling tasks (adapted from Hall et al., 2003).

However, it has been found that the dynamic terms in the governing equations are critical to solve the complex flow patterns that arise with irregular topography (Leopardi et al., 2002). Another approach typically adopted is to use a coarser spatial resolution to decrease the number of computations, but Ferziger and Peric (2002) found this generally results in an increase in discretization error and may result in reduced accuracy (Yu and Lane, 2006; Merwade et al., 2008).

A recent approach to improving computation speed of flood models is parallelizing the source code. Two-dimensional flood models have been implemented in a high-performance parallel computing architecture including Flo2DH by Hluchy et al.

(2002), RMA by Rao (2005), CalTWiMS by Pau and Sanders (2006), TRENT by Villanueva and Wright (2006), and LISFLOOD-FP by Neal et al. (2009). However, converting existing codes to execute on a parallel programming computer cluster is a complex process requiring significant programming effort (Tran and Hluchy, 2004), and high-performance computing infrastructure is typically limited to academic and government institutions, limiting their availability to private consulting and engineering firms. But recent developments in technology have produced computers with multiple microprocessors capable of being executed in a parallel fashion and flood models have been developed to take advantage of this capability (e.g., FIT2D by Judi et al., 2010).

Another flood model parallelization approach has emerged recently. The advances in integrated circuit technology and graphics hardware, in the past decade, brought about an evolution of graphics hardware, commonly called “graphics cards” in the video game industry (Sony Play Station[®] 3, etc.). The graphics card, also called Graphics Processing Unit (GPU), possesses microcomputer-like programmability similar to a CPU. It is a specialized processor that performs 3D graphics rendering from the microprocessor, typically used in embedded systems, personal computers, workstations and game consoles. GPUs are attractive because they offer extensive computational capabilities including massive parallelism, high memory/data transfer between the motherboard and the GPU, and not just graphics applications but also nongraphics applications. The GPU architecture provides large memory bandwidth and floating point operations per seconds or FLOPS (measure of scientific calculations involving floating point calculations), when compared to conventional CPU. For example, NVIDIA GeForce 6800 Ultra achieves

35.2 GB/s of memory bandwidth, compared to a 3.7 GHz Intel Pentium4 SSE unit whose theoretical peak memory bandwidth is 14.8 GB/s (Owens et al., 2005).

GPU development has enabled parallel computational capability of the GPUs through graphics application programming interfaces (API) like OpenGL and Direct3D (Nguyen, 2007; GPGPU, 2010). This has created a proliferation of General Purpose computation on Graphics Processing Units (GPGPU) approaches for nongraphics applications and scientific computing research (GPGPU, 2010). With the advent of programmable GPUs, legacy scientific computing codes are being reprogrammed following the GPGPU approach using the conventional graphics pipeline using graphics APIs like OpenGL, Cg and Direct3D (Kruger and Westermann, 2003; Wu et al., 2004). Although GPU computing has been incorporated into a wide range of computational and modeling applications with success (Hagen et al., 2005; Anderson et al., 2009), it has found limited application in computational fluid dynamics and flood modeling. Harris et al. (2002) presented a real-time visual simulation of diverse dynamic phenomena using GPU, and presented a speedup of 25x on a NVIDIA® GeForce 4, compared to the same simulation run on a Pentium CPU. Hagen et al., (2005) presented a visual simulation study of shallow water waves using GPU and reported 15x to 30x speedup compared to a CPU simulation. Lamb et al. (2009) presented a speed up of 112x using a diffusive wave flood modeling approach compared to a CPU model.

GPU applications in flood modeling remain an area of emerging research (Lamb et al., 2009; Neal et al., 2009). They have been limited to using the traditional graphics pipeline (using shader programming) or specialized hardware (ClearSpeed™) in which the programmer transforms a computational algorithm into a set of graphics operations.

The drawback of these approaches is the significant complexity of the source code for executing programs, leading to a steep learning curve. A recent advancement that addresses this issue is the introduction of Compute Unified Data Architecture (CUDA) by the NVIDIA® Corporation. CUDA is a parallel programming model and a software environment for parallel computing aimed at engineering and scientific applications. While graphics pipeline-based GPU computing requires prior graphics and computer science background, using the newer and specialized nongraphics pipeline that CUDA offers is straightforward. It is designed to develop application software that transparently scales its parallelism to leverage the increasing number of processor cores, while maintaining a low learning curve (NVIDIA, 2009). CUDA also provides a straightforward means of describing inherently parallel computations and is specifically relevant to data parallel algorithms (Garland et al., 2008). This chapter addresses this emerging research area by describing a new GPU implementation of the full dynamic wave equations in a CUDA framework. The objectives of the chapter are to present the new GPU-based 2D flood model, demonstrate its computational improvements compared to its CPU counterpart and explore the GPU performance as a function of domain size and spatial resolution.

2.2 Methodology

2.2.1 Model Description

The numerical algorithm used in the GPU 2D flood model developed in this study (Flood2D-GPU) is a first-order accurate upwind difference scheme that solves the non-linear hyperbolic shallow water equations. These equations are developed from the

Navier-Stokes equations by integrating the horizontal momentum and continuity equations over depth often referred to as the depth averaged or depth integrated shallow water equations (i.e, Saint Venant equations). The nonconservative form of the partial differential equations is (Tingsanchali and Rattanapitikon, 1999; Judi et al., 2010):

$$\frac{\partial h}{\partial t} + u \frac{\partial h}{\partial x} + v \frac{\partial h}{\partial y} = 0 \quad \text{Continuity Equation (2-1)}$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + g \frac{\partial H}{\partial x} + g S_{fx} = 0 \quad \text{Momentum Equation in x - direction (2-2)}$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + g \frac{\partial H}{\partial y} + g S_{fy} = 0 \quad \text{Momentum Equation in y - direction (2-3)}$$

where, h is the water depth, H is the water surface elevation, u is the velocity in the x -direction, v is the velocity in the y -direction, t is the time, g is the acceleration due to gravity, S_{fx} is the friction slope in the x -direction and S_{fy} is the friction slope in the y -direction. The upwind finite difference numerical scheme is used to discretize governing equations 2-1 – 2-3, as it yields nonoscillatory solutions, through numerical diffusion (Patankar, 1980; Ferziger and Peric, 2002). A staggered grid stencil is used to define the computational domain with the water depth (h) in the center of the cell and u and v velocities on the cell edges. The future model time step is constrained using the Courant condition. Appendix A presents the numerical solution of the equations.

The principal dataset for Flood2D-GPU is topographic data (i.e., digital elevation model (DEM)), which is a uniform grid structure. The numerical solution is calculated on a uniform grid to take advantage of the use of downloadable DEM data. While using an irregular mesh is efficient for model computing, the advantage of using the DEM is that

its extent is the entire computational domain, and no preprocessing step is required to generate a computational mesh, permitting flood model applications to meet time constraints.

Additional data needed for the model include a surface roughness coefficient (Manning's n value) and source boundary information. The roughness of the entire domain is typically represented using a single Manning's n value in 2D modeling applications (Hunter et al., 2007; Judi et al., 2010). The flow hydrograph is an input dataset that can be developed from a hydrologic model, dam break model or observations. The source location of the input hydrograph must be defined. The model does not account for evapotranspiration, infiltration and erosion processes that could affect the surface runoff, especially for long-term flood simulations in the floodplain. Incorporating these physical processes would enable influence on the floodplain simulations in terms of flood velocities, depths and also the duration of flooding.

The Flood2D-GPU model was programmed following the traditional CPU approach in C programming language on an AMD[®] Phenom II X4, 3.04 GHz desktop with 4GB RAM in the Linux 64-bit programming environment. After successful testing and validation, the program was converted in collaboration with the University of Utah Scientific Computing and Imaging (SCI) Institute to the GPU framework using NVIDIA's CUDA. Figure 2.2 illustrates the process flow in the GPU framework with functions color coded according to their implementation. CUDA programming is designed in such a way that the process control algorithm initializes with the CPU inputting the required data and transforming it into appropriate GPU data format. Then, data and process control are transferred to the GPU and the computations are performed.

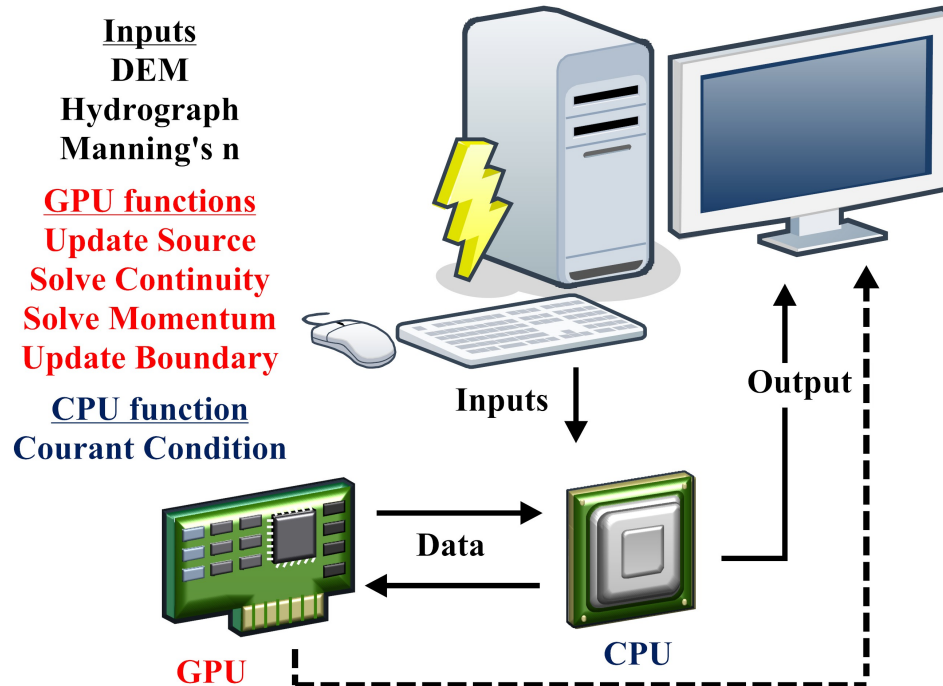


Figure 2.2. Schematic of CUDA GPU computing approach followed by the Flood2D-GPU model (GPU elements shown in red).

The output is transferred back to the CPU for additional computation (if necessary) and storage. GPU programming also has the capability to visualize (dotted line in Figure 2.2) model/program results (e.g., video games, graphic visualizations) using a front-end visualization.

In the CUDA environment, the GPU functions are called kernels. The kernels generate a large number of threads to exploit data parallelism. The model domain is divided into a set of blocks, which are groups of computational elements (Figure 2.3). Threads are assigned to the blocks, and the blocks are scheduled for computation in the GPU Symmetric Multi-Processing (SMP) cores. Each block is allocated a shared memory for computation and storage. The advantage of shared memory allocation is faster

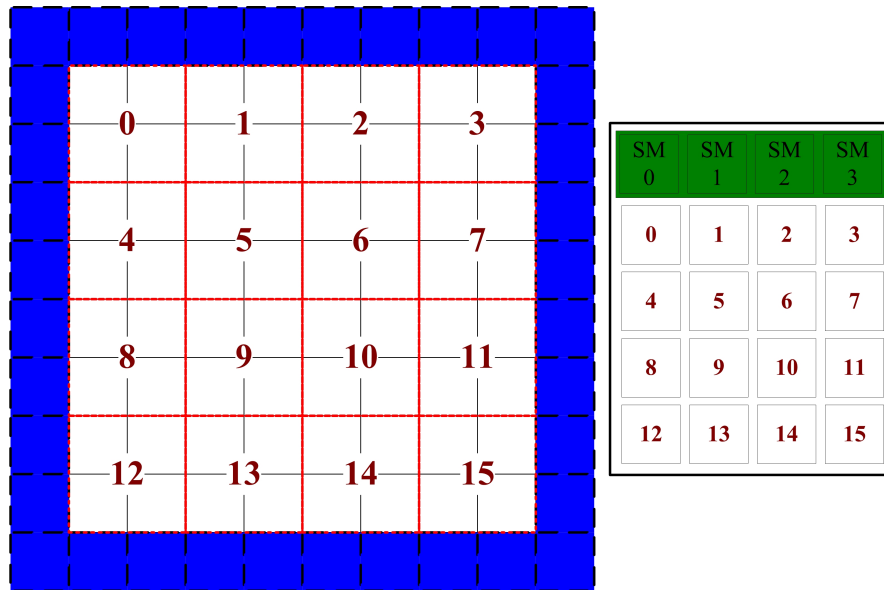


Figure 2.3. The inner cells, bounded by blocks in red (left), are scheduled onto the SMP cores of the GPU (right).

implementation, and memory interaction between the threads within a block, that reduces memory latency, thus increasing computational performance. As Figure 2.3 illustrates, only the inner cells (red boxes) of the computational domain are considered when generating the blocks. Cells on the domain boundary are updated separately by applying appropriate boundary conditions.

The Saint Venant equations (2-1 – 2-3) are used to compute the updated h , u and v values. To find the maximum u , v , and the corresponding h values, a parallel reduction kernel is implemented. For the boundary update, the boundary cells are populated using the ‘stencil’ method from Micikevicius (2009). For efficient thread management, the boundary update function was divided into two different kernels, one for updating the boundary rows, and the other for updating the boundary columns.

The many-core simulation of Flood2D-GPU is benchmarked against the CPU-based version of the model executed on a 3.2 GHz AMD desktop with 4GB of RAM. The

GPU model is tested on two different NVIDIA graphics cards, the GeForce 8400 GS (GPU 1) and the Tesla C1060 (GPU 2). The former, a low-end card, and the latter, a high-end card, are used to investigate the impact of graphics card hardware on model performance. Table 2.1 lists the characteristics of the two GPU applications. All computers use the Linux 64-bit operating system.

2.2.2 Model Validation

Two case studies were selected to verify the ability of the Flood2D-GPU model to reproduce accurate results. The following section presents the case studies that were used in the model validation and verification.

2.2.2.1 Laboratory Scale Dam Break

Tinsanchali and Rattanapitikon (1999) created a physical dam break model and studied their model performance. In their study, they performed several experiments in order to gather data to be compared to their 2D shallow water model as part of the validation process. The data provided in their paper, however, were minimal and it is thought that some of the techniques used to gather data from the model could be improved. A similar step was accomplished in the laboratory and the same case study is

Table 2.1. Comparison of the configuration of the GPU machines

| | GPU 1 | GPU 2 |
|----------------------|-----------------|-------------|
| CPU Frequency | 2.33 GHz | 2.67 GHz |
| RAM | 2 GB | 24 GB |
| NVIDIA Graphics Card | GeForce 8400 GS | Tesla C1060 |
| GPU Frequency | 459 MHz | 1.30 GHz |
| CUDA Cores | 16 | 240 |

used in this research. Here are the details of the experiments taken from Judi (2009). The structure consists of a reservoir and a floodplain, separated by a wall with a 0.1m slot at the centerline that is controlled by a gate. The dam break is simulated by the near instantaneous lifting of the gate and the immediate uncontrolled release of water from the reservoir. The instantaneous reservoir outflow is measured from the change in the storage of the reservoir. A 0.5m outlet was placed at the end of the floodplain. The reservoir and floodplain are constructed using plywood with the exception of one wall on the floodplain that is plexiglass to enhance visibility of the flood wave. A plan and a profile view of the physical model are shown in Figures 2.4 and 2.5. The In-Situ Level TROLL 500 pressure transducers are placed at a distance of 0.7m, 1.4m and 3.1m from the reservoir along the centerline to measure the water depths in the floodplain.

The dam break was simulated for an initial reservoir head of 25 cm several times for consistency. A range of expected values was identified for each sensor. A 2D flow was observed in the floodplain. There was also some flow that was reflected from the walls to the center of the floodplain, forming a hydraulic jump.

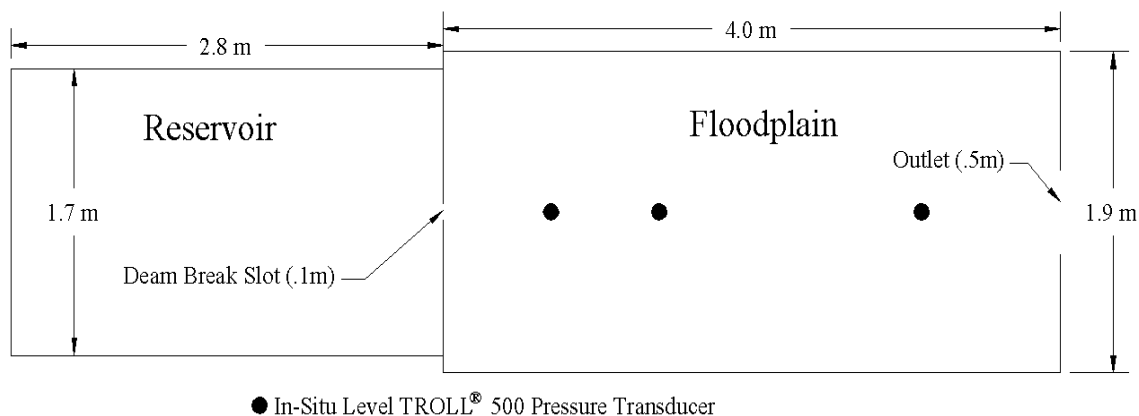


Figure 2.4. Plan view of Laboratory Physical Model (Judi, 2009)

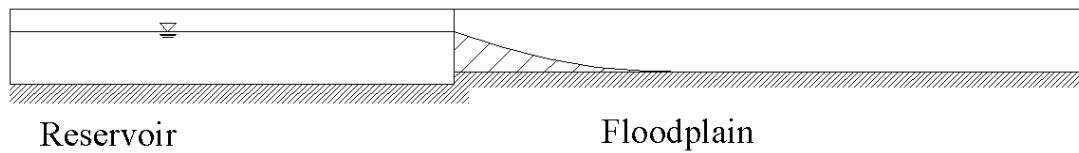


Figure 2.5. Profile view of Laboratory Physical Model (Judi, 2009)

A supercritical regime (Froude Number > 1.0) was observed for a distance from the gate. Some flow drained through the outlet and some flow reflected back from the end walls, and it stalled near the second sensor. It was also found that the speed and depth of the flood wave was dependent on the initial reservoir head.

A digital elevation model (DEM) representation of the physical model was created with a 0.02 m resolution uniform numerical grid of approximately 36,000 grid cells. The boundary conditions used were “walled” boundaries and a free outlet. The walled boundaries are represented by ‘no-slip’ condition where the lateral fluxes (along the width of the floodplain) are set to zero. The free outlet was governed by the minimum of normal and critical depths (Sturm, 2001).

2.2.2.2 Taum Sauk Dam Break

The validation of the CPU and GPU models is also conducted using a dam break event that occurred at the Taum Sauk pump storage hydroelectric plant, located in Reynolds County, Missouri. It is a twin reservoir system, designed to produce electricity during peak periods by discharging water from the upper reservoir to the lower reservoir during peak electricity rate periods and pumping water back to the upper reservoir during off-peak rate periods. The upper reservoir, built on the Proffit Mountain, is approximately 232 m above the floodplain of the East Fork Black River, with a storage capacity of 5.7

M m³. It failed on December 14, 2005, at a 207 m wide section on the northwest side of the Proffit Mountain, releasing the entire storage into East Fork Black River floodplain, through Johnson's Shut-Ins State Park and into a lower storage reservoir within 25 min (Rydlund Jr., 2006).

The topography for this study is a 1/3 arc second (9.36 m) United States Geological Survey (USGS) DEM covering the 62 km² extent of the flood event with 708,864 model grid cells (624 x 1136 grid mesh). The hydrograph in this study is from the USGS analysis of the event (Rydlund Jr., 2006), in which the discharge is developed from a volume analysis of the reservoir and knowledge of the embankment failure. The peak discharge is 8,100 cubic meters per second (cms), which occurs approximately six min after the breach. The simulation uses a Manning's roughness value of 0.035, which Judi et al. (2010) found to be reasonable for 2D models at the Taum Sauk location. The model comparison with observed high water mark data is performed for the channel section between the upstream and downstream reservoirs of the pump-storage system because of the lack of observed high water mark data at and below the lower reservoir.

To quantify the deviation of simulated results from observations, three metrics are used. The first metric is a measure of fit, the $F^{(2)}$ statistic presented by Bates and De Roo (2000):

$$F^{(2)} = \frac{(A^{obs} \cap A^{mod})}{(A^{obs} \cup A^{mod})} \quad (2-4)$$

where, A^{obs} and A^{mod} represent the inundation extent of the observed and modeled data, respectively. The other two metric are statistical comparisons of the two datasets to

measure the over-estimation (Commission) and under-estimation (Omission) of the simulated maximum flood extent (i.e., high-water mark) relative to observations:

$$C = \left(1 - \frac{P_e}{P_t}\right) \times 100 \quad (2-5)$$

$$O = \left(1 - \frac{P_e}{P_u}\right) \times 100 \quad (2-6)$$

where, P_e is the number of common flooded cells, P_t is the total number of model wet cells, and P_u is the total number of observed wet cells, all at high-water mark conditions.

2.2.3 Model Speedup

The model speedup from CPU to GPU is calculated as the ratio of the execution times of CPU to GPU:

$$Speedup = \frac{Time_{CPU}}{Time_{GPU}} \quad (2-7)$$

The computational enhancement of using GPU versus the CPU model is analyzed for three cases. The first case is the speedup of GPU from CPU for the Taum Sauk case study. Second, the GPU to CPU speedup is analyzed for the Taum Sauk case study after the computational domain is reduced to the minimum domain size encompassing the maximum flood extent. The third case investigates the effect of spatial resolution on model speedup. Resampling from a finer to a coarser resolution reduces the size of the

computational domain and reduces the execution time. This method studies the variation in the GPU speedup from finer to coarser resolution. To achieve this, a subwatershed in the Greens Bayou watershed near Houston, Texas is used. The subwatershed is a small urban area (42 km²), about 126 km upstream of Galveston Bay and the Gulf of Mexico. The subwatershed feeds into Greens Bayou, which follows the topography flowing into the Houston Ship Channel before draining into the Gulf of Mexico at Galveston Bay. A flow hydrograph is produced by a distributed hydrologic model (Kalyanapu et al., 2009) for a rainfall event that occurred on February 4, 1991.

2.3 Results and Discussion

2.3.1 Model Validation

2.3.1.1 Laboratory Scale Dam Break Simulation

Based on the tests conducted by Judi (2009), a roughness value of 0.0115 was found to produce closer results to the measurements. This roughness value was used to simulate the dam break. Figures 2.6 – 2.8 present the measured depths for the sensors (error bars representing the range of values, maximum and minimum, obtained from the laboratory scale model) and the simulated depths for an initial head of 25 cm using the above-mentioned roughness value. The flood wave initializes from the dam break source and flows through the floodplain. As it advances in the floodplain, it expands laterally and reaches the side walls at about 2 seconds and reflects back into the center of the floodplain. The flood wave proceeds and reaches the outlet and while some flow is drained at the outlet, a reflected wave returns back into the center of the floodplain. The initial spike in the depths is because the wetting front that is simulated is deeper than that

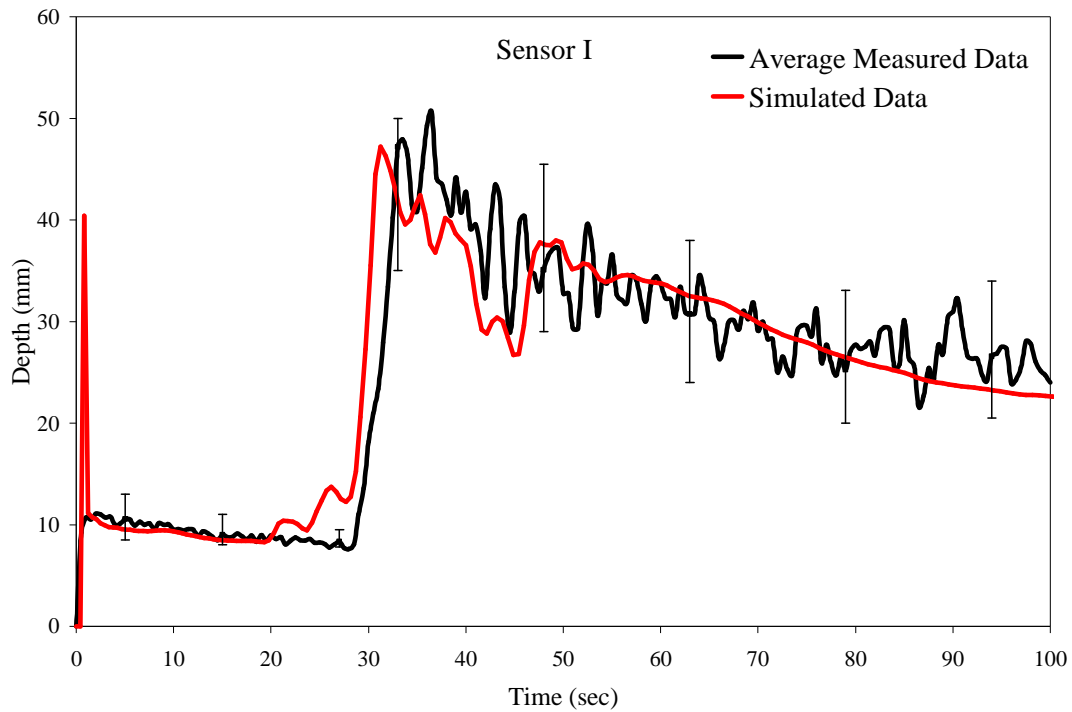


Figure 2.6. Sensor 1 (0.7 m downstream of the dam) comparison of measured and simulated depths.

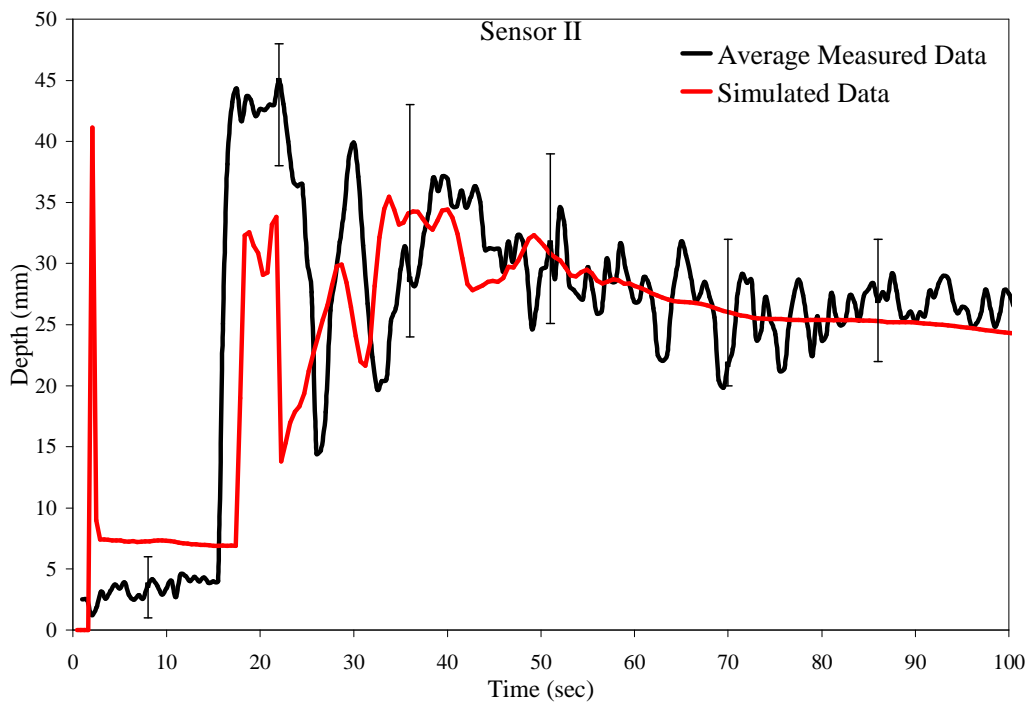


Figure 2.7. Sensor 2 (1.4 m downstream of the dam) comparison of measured and simulated depths.

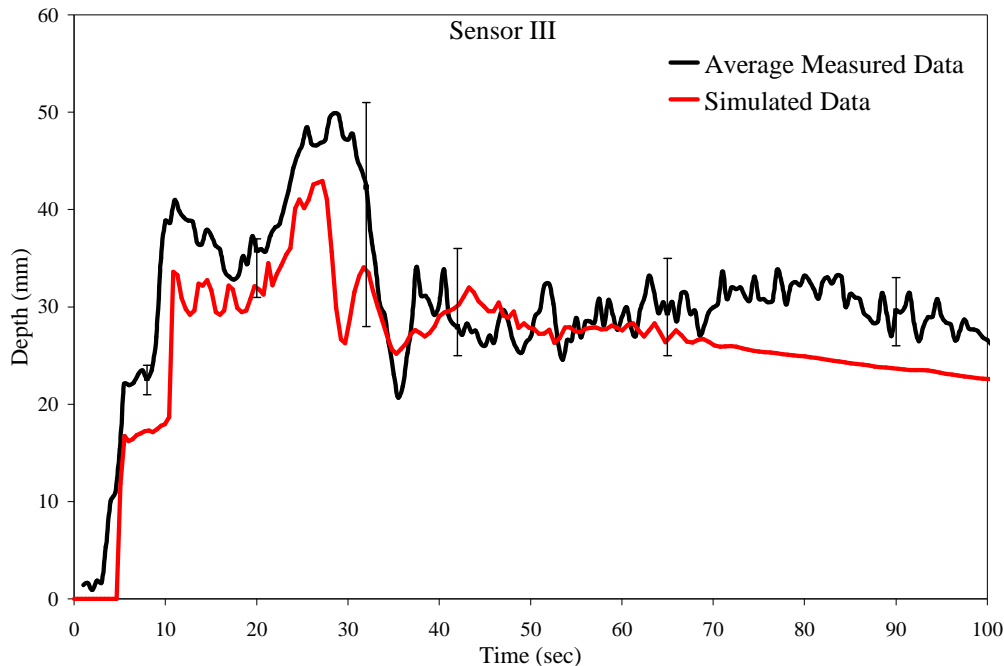


Figure 2.8. Sensor 3 (3.1 m downstream of the dam) comparison of measured and simulated depths.

of the measurements at the first and second sensors. However, it evens out with the measured data near the third sensor.

These plots show that the flood depths simulated agree with the range of measured values, even though the model was not able to represent the wave pattern displayed in the measurements. This can be attributed to the diffusive nature of the first-order upwinding scheme used in the model. While this artificial diffusion results in the loss of accuracy, as noted in Figures 2.6, 2.7 and 2.8, this diffusion is also responsible for the stability of the numerical scheme (Patankar, 1980; Ferziger and Peric, 2002). This is because the artificial diffusion damps any errors that might arise during the course of numerical iterations and prevents these errors from growing. This dampening is observed in this experiment as the initially dry floodplain is flooded with a sudden increase in flood depths by the propagating flood wave. The diffusivity of upwind numerical scheme

stabilized the solution by capturing the shock (defined as the change from dry to wet condition in a cell), which prevents increases in the errors. However, the disadvantage of the first-order upwind scheme is also due to the artificial diffusion. During the “shock” (change of a dry cell to a wet cell), this diffusion causes a smearing effect of the propagating wave front, thus simulating different flood depths along the wave front than the observed flood depths. This diffusive affect would take a few more numerical iterations to dissipate and for the simulated flood depths to be closer to the observed results. To solve the accuracy issue, second-order accurate upwinding schemes are needed (García-Navarro et al., 2008). But overall, the model was able to recreate the temporal variation of the flood depths, which proves its validity in simulating this dam break. A similar performance may be expected when applied for other events. The statistics including the Root Mean Square Error (RMSE), Cumulative Relative Error (CRE), and Bias are calculated at the sensor locations using Equations 2-8 – 2-10 and are presented in Table 2.2.

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (h_i^{sim} - h_i^{obs})^2}{n}} \quad (2-8)$$

$$CRE = \frac{\sum_{i=1}^n |h_i^{sim} - h_i^{obs}|}{\sum_{i=1}^n h_i^{obs}} \quad (2-9)$$

$$Bias = \frac{\sum_{i=1}^n (h_i^{sim} - h_i^{obs})}{n} \quad (2-10)$$

Table 2.2. Quantified statistics for laboratory scale dam simulation

| | RMSE (mm) | CRE (mm) | Bias (mm) |
|------------|-----------|----------|-----------|
| Sensor I | 4.8 | 0.1 | -0.2 |
| Sensor II | 7.6 | 0.2 | -1.3 |
| Sensor III | 6.3 | 0.2 | -4.4 |

At the first sensor, most of this deviation in simulated depths with the observed readings can be attributed to the initial spike in the depths at 0.8 seconds, which is due to the flood wave (Figure 2.9). This depth increase from this wetting front is again observed at the second sensor at 2.1 seconds, as shown in Figure 2.10. The reason behind this depth increase is the initial conditions used in the model. The dam outlet is assigned an initial depth of 0.25 m at the beginning of the simulation. This depth is used in the velocity calculation by solving the momentum and continuity equations. Since the initial velocity is considered to be zero at the reservoir outlet, it would take a few iterations for the model to generate the velocities comparable to the experiment.

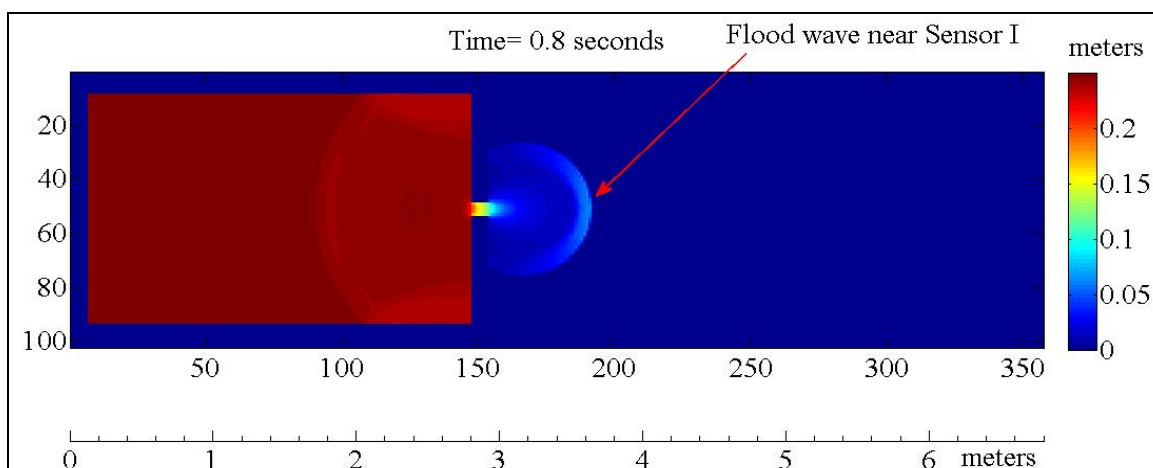


Figure 2.9. Flood wave near the first sensor at 0.8 seconds of simulation

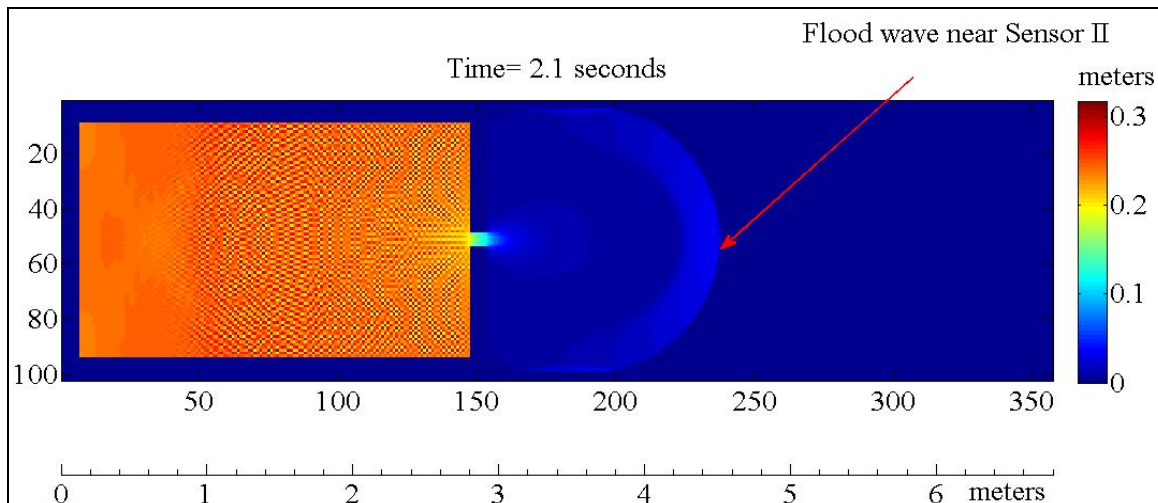


Figure 2.10. Flood wave near the second sensor at 2.1 seconds of simulation

Thus, this slower development of velocity leads to higher water depths for the flood wave than the observed sensor readings, causing the depth increase in the model simulations. This spike at the first sensor lasts for another 0.7 seconds and diminishes after 1.5 seconds. At the second sensor, it lasts for 0.4 seconds and diminishes after 2.5 seconds, as seen in Figures 2.6 and 2.7. However, the RMSE values are within range of width of error bars presented for the three sensors, and also seen in the CRE. The RMSE is also around 10% of the maximum depths that were observed at the three sensors, which represents a better prediction. The bias indicates that the simulated depth values are consistently smaller than the observed average depths in spite of the initial spikes at the first and second sensors. Overall, the model simulated the lab scale dam break event, in spite of model disturbances in the initial 2 seconds of the simulations. In real-world events, it will be difficult to compare the temporal variation of water surface elevations at gage stations, especially for large flood events, where the stations would be ineffective if they are inundated or a threshold water depth is reached. Thus, flood inundation extents

for significant events are derived either using remote sensing techniques or by field survey of high water marks.

2.3.1.2 Taum Sauk Dam Break Simulation

Figure 2.11 presents the flood inundation extent of the event simulated by the GPU model 20 min after the dam breach, the time corresponding to the maximum simulated flood extent. The $F^{<2>}$ statistic and the commission and omission errors are also presented in Figure 2.11. The Flood2D-GPU simulation had excellent agreement with the observed data: 75% overall $F^{<2>}$ statistic value with 15% of the flood extent being overpredicted (commission) and 13% of the flood extent being underpredicted (omission). The model underestimated near the formation of the break and near the western end of the confluence with the East Fork Black River while its overestimation was significant at the location upstream of the confluence. This model validation demonstrated model performance similar to that achieved by Judi et al. (2010) for the FIT2D model for the same Taum Sauk dam break event. As a simple way to check the reasonableness of the simulated water depths, an approximated approach was used. Simulated water depths are compared with the estimations of the observed water depths found by subtracting land surface elevations from observed high water marks. The relative percent difference ranging between -5.8% and 4.2% were found. This provides an approximate check to the water depth calculations although this approach does not account for the physical effects of overland flow.

The timing of the maximum flood inundation is not validated because the time of maximum inundation was not observed. However, Rydlund Jr. (2006) did conduct a post-

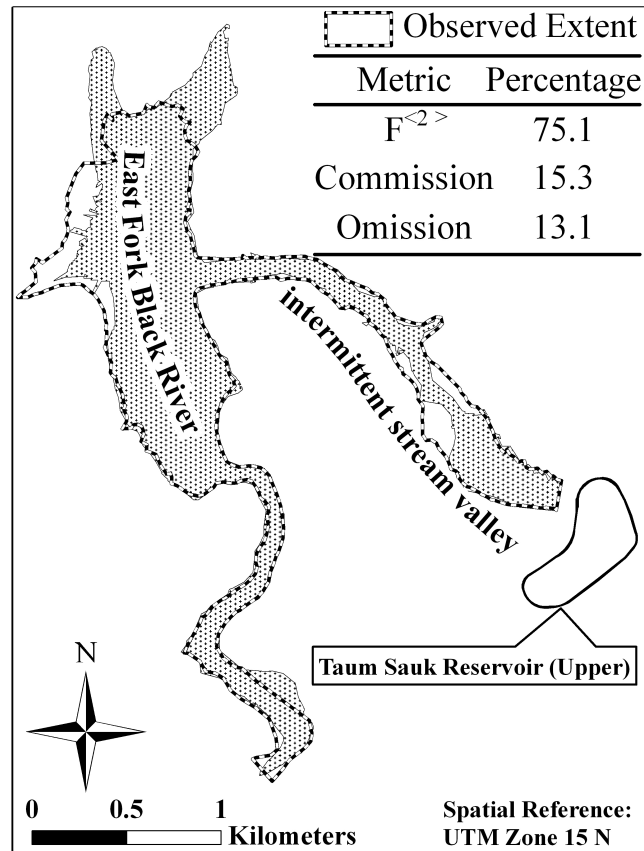


Figure 2.11. Comparison of Flood2D-GPU flood inundation extent with observations for the Taum Sauk Dam Break event at 20 min after the dam breach.

event modeling study of the dam break, which included flood wave routing from the upper reservoir embankment failure to the spillway of the lower reservoir using the dynamic wave unsteady flow models Dam Break (DAMBRK) and Unsteady NETwork (UNET). From these simulations, the flood wave is estimated to have entered the floodplain of East Fork Black River at approximately 5.5 to 6 min after the breach occurred. Additionally, the wave front was simulated by Rydlund Jr. (2006) to have entered the lower reservoir at 29 min after the breach. The Flood2D-GPU model simulated the flood wave entering the East Fork Black River at 6 min after breach and entering the lower reservoir at 31 min after breach. Based on this comparison of

simulated results from two modeling efforts, it can be concluded that Flood2D-GPU reasonably simulates the flood inundation extent and timing.

2.3.2 Model Speedup Calculation

The Taum Sauk dam break flood event was reproduced for 30 min of simulation (the reservoir emptied in 25 min) which required 50k numerical iterations. The speedups for Flood2D-GPU against the CPU benchmark are listed in Table 2.3. The execution time reduced from 173 min using a CPU to 50 min using GPU 1 (GeForce 8400 GS) and 2 min using GPU 2 (Tesla C1060). The significant execution time reductions for the GPU-enabled flood model (especially for the higher end GPU 2) is attributed to their parallel processing capability and computing power provided by the multicore processors. It is observed that implementing the GPU version increased the model performance with speedups of 3.5x on GPU 1 (16 cores) and 84x on GPU 2 (240 cores). It is important to note the speedups obtained by switching from GPU 1 to GPU 2 are in the range of 24x. However, the increase in the number of cores between the two graphics cards is 15x, showing the model speedup is greater than the increase in number of cores. The reason is the difference in the hardware compute capabilities of the two GPUs. GPU 2 supports compute capability 1.3 (compute capability describes the features supported

Table 2.3. GPU speedup results for Taum Sauk dam break simulations for three sets of model iterations

| Taum Sauk (708k cells) | Execution time (min) | | | Speedup | | |
|---------------------------|----------------------|-------|-------|-----------------|-------------------|-----------------|
| | CPU | GPU 1 | GPU 2 | CPU to GPU 1 | GPU 1 to GPU 2 | CPU to GPU 2 |
| 50k | 173.2 | 50.5 | 2.1 | 3.4x | 24.0x | 82.5x |
| 100k | 345.3 | 97.4 | 4.1 | 3.5x | 23.8x | 84.2x |
| 150k | 545.9 | 146.1 | 6.2 | 3.7x | 23.6x | 88.0x |

by CUDA hardware; a higher compute capability is always recommended for enhanced performance) which decreases the number of memory accesses by the threads, compared to GPU 1 with compute capability 1.1 (NVIDIA, 2009). The models were also executed for 100k and 150k numerical iterations to observe model scalability. As expected, as the numerical iterations increased, the execution time also increased proportionally (Table 2.3, Figure 2.12). The increased number of processors available in GPUs is primarily responsible for its better computational performance. The speedups experienced in these three different hardware (one CPU and two different GPUs) are not directly proportional to this increase in processors (i.e., CPU (1 core) to GPU 1 (16 cores) and CPU to GPU 2 (240 cores) do not result in 16x and 240X speedups). This is mostly caused by the cores

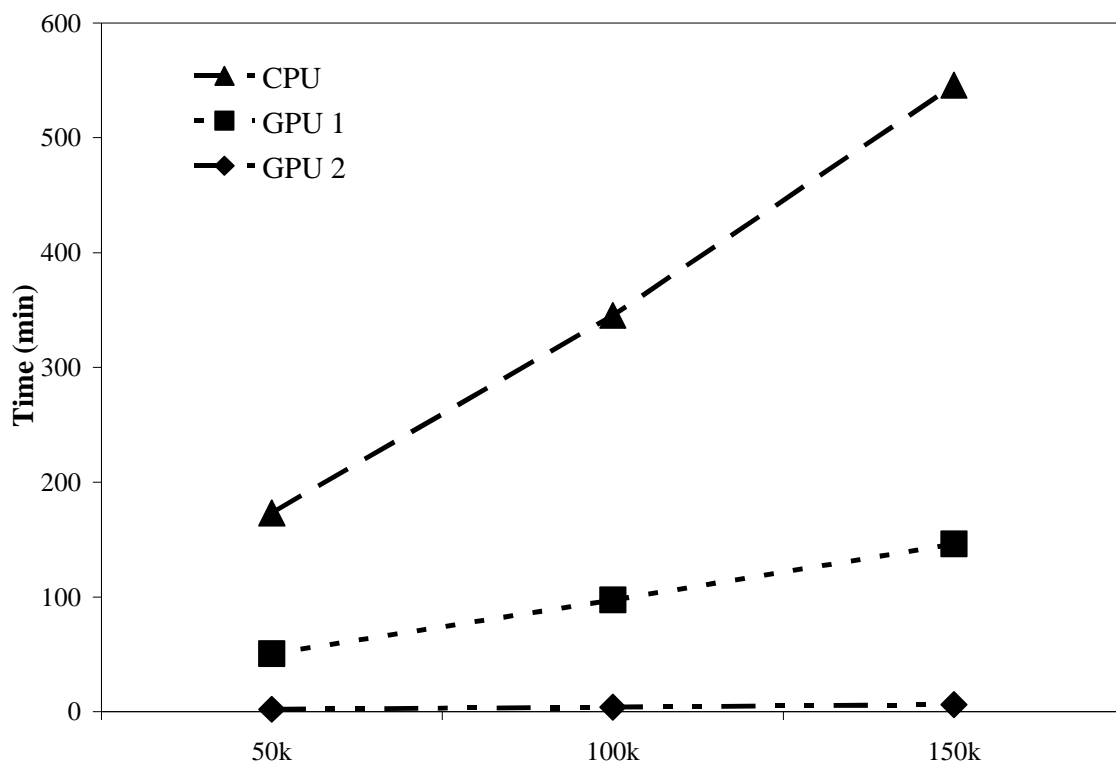


Figure 2.12. Execution times of GPU and CPU models for the Taum Sauk case study simulation.

not being completely parallelized. The major computations have been parallelized in GPU, but a small amount of overhead is not parallelized, as shown in Figure 2.2.

2.3.3 Effect of Domain Minimization

Using a uniform computational grid for flood modeling causes certain areas of the model domain to be unused in flood calculations because the simulated flood extent will not extend to those areas. Figure 2.13 illustrates this problem for the Taum Sauk simulation. The waste is unavoidable because the extent of the flood is not known until a simulation is executed. Iteration permits the modeler to minimize the model domain to

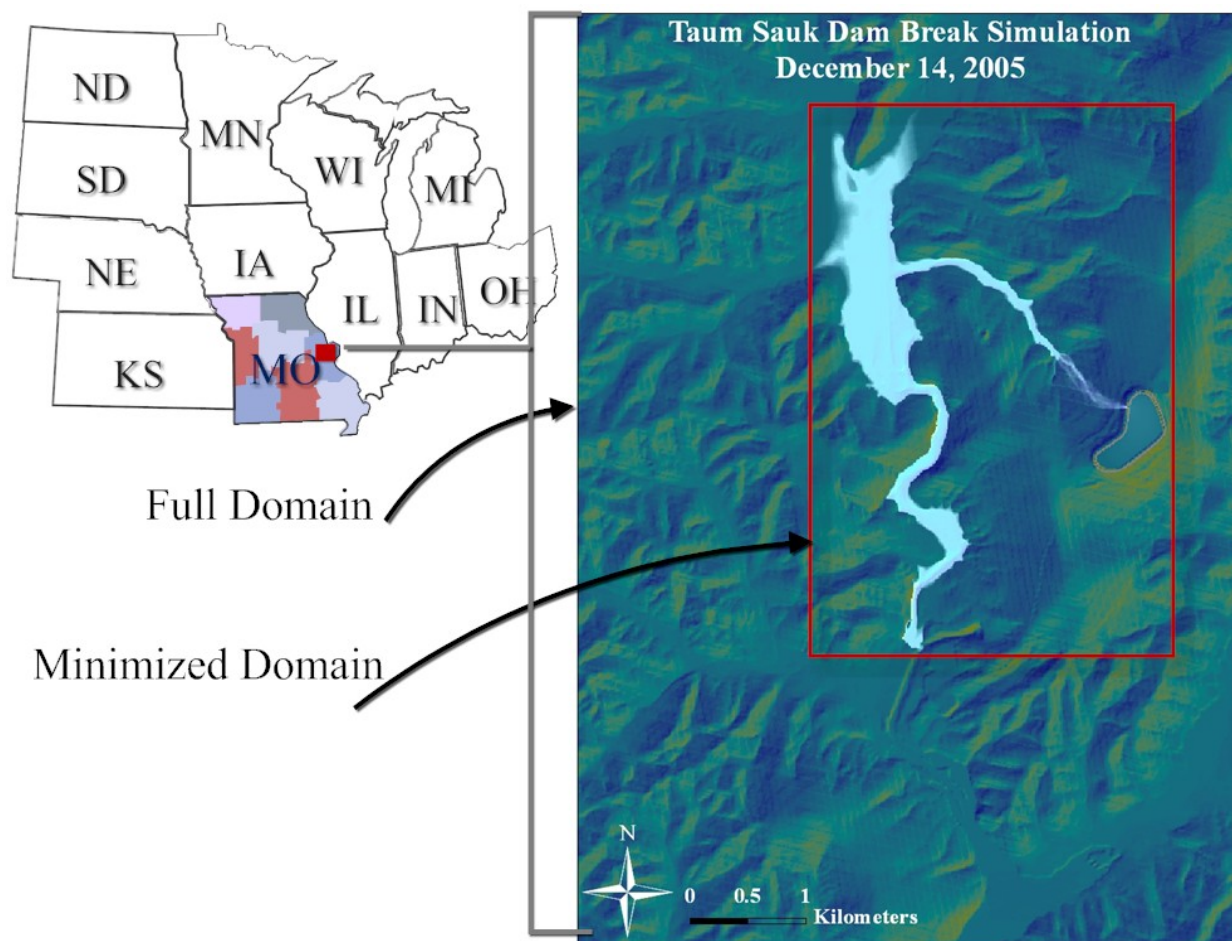


Figure 2.13. Domain minimization for Taum Sauk dam break flood simulation.

the irreducible minimum. However, this practice takes additional time and effort that may be costly in terms of having to run multiple simulations and still meet specific time constraints for flood modeling (see Figure 2.1). For the Taum Sauk simulation, the original computational domain extent was 62 km² and a minimization step following an initial set of simulations reduced the domain to 27 km², as shown in Figure 2.13.

Table 2.4 presents the execution times before and after domain minimization for CPU and GPU 2 only. As expected, the execution times decrease due to the reduced modeling domain. The domain reduction increased the CPU speedup slightly (2.3x), but the 84x speedup achieved by GPU 2 for the original computational domain is much more significant. In effect, GPU without domain minimization has a much greater speedup potential compared to the CPU with the domain minimized (37x for this case study). This is important to note because it suggests even greater value for GPU-based flood models being able to provide a reduction in effort in the modeling because an additional step to minimize the domain may be avoided to save time.

2.3.4 Effect of Spatial Resolution

The study of the effect of spatial resolution on the performance of Flood2D-GPU versus its CPU counterpart is analyzed using a 41 km² subwatershed of the Greens Bayou watershed in the Houston, Texas, USA metropolitan area. The DEM for the subwatershed

Table 2.4. Execution times before and after domain minimization

| Domain | Iterations (min) | | | | | |
|--|------------------|-------|-------|-------|-------|-------|
| | 50k | | 100k | | 150k | |
| | CPU | GPU 2 | CPU | GPU 2 | CPU | GPU 2 |
| Full Extent (62 km ²) | 173.2 | 2.1 | 345.3 | 4.1 | 545.9 | 6.1 |
| Minimized Domain (27 km ²) | 74.5 | 1.0 | 147.5 | 2.0 | 221.0 | 3.1 |

from the USGS has a spatial resolution of 25.3 m, creating 65.5k grid cells (256 x 256) in the computational domain. For the study, the DEM is resampled to a higher spatial resolution of 6.3 m, producing 1.05 M grid cells (1024 x 1024). Simulations are executed for both resolutions and model speedups determined. Figure 2.14 presents the speedup for 50k iterations for CPU to GPU 1, GPU 1 to GPU 2 and CPU to GPU 2. With higher spatial resolution, more grid cells are required to represent a location compared to a coarser resolution, making the former computationally intensive. It is observed from Figure 2.14 that as the spatial resolution increases (grid cells get smaller), the computational performance of GPU 2 significantly increases, nearly two orders of magnitude compared to CPU. Higher resolution data are generally considered to produce more accurate results in flood modeling due to detailed topographic representation and more accurate depiction of flow direction and characteristics (Sanders, 2007; Bales and Wagner, 2009). The results therefore suggest the importance of parallel processing computations for flood modeling at higher spatial resolutions. Counterintuitively, it is important to note that the rate of increase of speedup is higher for GPU 1 to GPU 2 than that of CPU to GPU 2. This could be due to the fact that the differences in GPU hardware are more significant than the differences between serial CPU and parallel GPU frameworks. The potential for GPU performance to be improved as a function of spatial resolution is critical to not only meet time constraints, but to also more accurately represent complex topography and flow characteristics (Marks and Bates, 2000; Sanders, 2007).

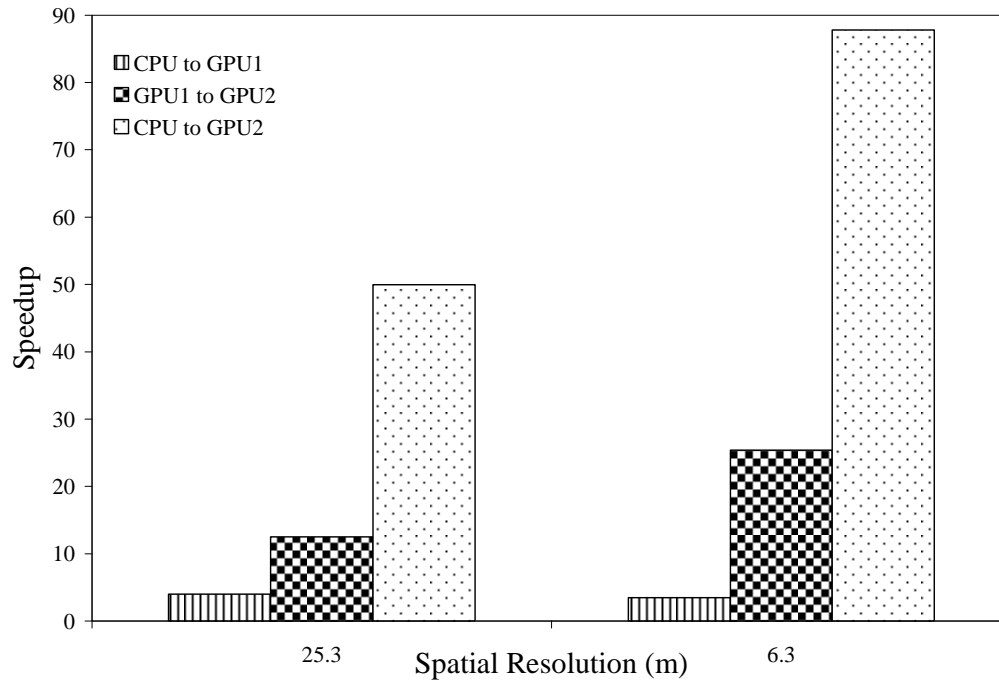


Figure 2.14. GPU 2 speedups of Greens Bayou flood simulation at different spatial resolutions for 50k numerical iterations.

2.3.5 Speedup Comparison to Other Studies

Table 2.5 compares the GPU speedups from CPU with the performance of other flood models documented in the recent literature. It should be noted that these results are only relative as the models and the hardware are different, with different flood events, input parameters, and computational domains. They are presented here to showcase the potential performance enhancement of a variety of parallelization approaches for 2D flood models. It is clear from the recent results presented in Table 2.5 that speedups from parallelization are substantial. The models implementing simplified approaches (e.g., diffusive wave approximation and storage cell) display excellent computational enhancement. The approach described in this dissertation uses the full Saint Venant equations, which are more computationally intensive and accurate. However, it also

Table 2.5. Comparative summary of recent model speeds in the literature for storage cell (SC), dynamic wave (DYN), and diffusive wave (DIF) models

| Description | This paper | | Lamb et al., 2009 | Judi et al., 2009 | Neal et al., 2009 |
|-------------------------|-----------------------------|--------------------------------|--------------------------------|--------------------------|---------------------------|
| Processor Type | GPU | GPU | GPU | CPU | CPU |
| Processor Information | NVIDIA 1.30 GHz Tesla C1060 | NVIDIA 459 MHz GeForce 8400 GS | NVIDIA 575 MHz GeForce 8800GTX | AMD 2.21GHz Opteron 8354 | AMD 2.59 GHz Opteron 2218 |
| Processor cores | 240 | 16 | 112 | 16 | 4-8 |
| Model name | Flood2D-GPU | Flood2D-GPU | JFLOW-GPU | FIT2D | LISFLOOD-FP |
| Additional Optimization | N/A | N/A | N/A | Domain tracking | N/A |
| Dimension | 2D | 2D | 2D | 2D | 1D-2D |
| Approximation | DYN | DYN | DIF | DYN | SC |
| Parallelization method | CUDA | CUDA | DirectX 9 | Java multi-threading | OpenMP |
| Domain size (cells) | 1.05M | 1.05M | 96k | 999k | 3k-3M |
| CPU run time | 13.0 h | 13.0 h | 18h | 0.83h* | 0.01-360h |
| Max speedups | 88x | 3.5x | 114x | 240x* | 5.8x |

* run time and speedup reported by Judi et al. (2010) included domain optimization algorithm which is independent of the computer platform.

provided excellent speedup. This suggests the power of the GPU may provide more complete numerical approaches to be implemented and still achieve the necessary speedup to meet modeling time constraints. With the continued development of new GPU hardware, the emergence of high-level programming languages, and the increasing trend of CPU-GPU interoperability (Owens et al., 2005), the future of GPU for flood modeling as well as other scientific applications seems bright.

2.3.6 Limitations of GPUs

It is to be noted that while GPU applications present significant benefits in terms of speed up and low cost, they also have some disadvantages currently. Firstly, the key

limitation of GPU is that it is applicable only to applications that can be inherently parallelizable, like numerical solutions with explicit temporal discretization and solutions where each grid cell in the domain can be processed independently of the neighboring grid cells (Hagen et al., 2005). The GPU parallelism is exploited in this dissertation because of the highly parallelizable nature of nonlinear hyperbolic Saint Venant equations. Secondly, the numerical precision offered by the graphics cards manufacturers like NVIDIA® and AMD® is restricted to single-precision, which is below par compared to the IEEE-754 standard for reasons of numerical efficiency (Menon, 2008). However, double-precision capabilities in the GPUs have recently started being implemented on newer GPU hardware architecture (e.g., NVIDIA offers double-precision for compute capability 1.3 and above, NVIDIA, 2009). Thirdly, GPUs experience reduction in performance when executing logical conditions within the GPU kernels (Richardson, 2009). For example, “if” conditions are used in Flood2D-GPU to implement upwinding for the convective terms of equations 2-1 – 2.3. This results in requiring a higher number of threads to process the subsequent velocity and depth calculations from the equations than it would without the “if” condition. Finally, GPUs are generally made with a fixed memory capacity and do not allow an easy increment of memory because the processor and memory physically reside on the same hardware. Thus, these limitations need to be taken into consideration by the analyst or flood modeler in the selection of GPU-based flood modeling applications at a wide-area scale.

2.4 Conclusions

In this chapter, the NVIDIA CUDA GPU-based 2D dynamic flood model Flood2D-GPU is introduced. The model is tested and validated using a laboratory scale dam break experiment and Taum Sauk dam break failure event in Missouri. The computational advantage of using GPU versus an equivalent CPU model is presented in three different ways. First, the computational enhancement of using this parallel programming technique is presented with computational speedups ranging between 82x and 88x compared to a CPU model implementing the same numerical algorithms. For this dam break flood scenario, the GPU-enabled 2D model executed on the Tesla C1060 is able to simulate 30 min of the flood event in 2 min, which is better than real time and able to meet the time constraints for most emergency response and flood evacuation situations. The speedups experienced from CPU to 8400GS to Tesla C1060 are not directly proportional to the increase in number of processors present in these hardware. Second, the computational domain is minimized to include only the flood extent, reducing the computational intensity of the models. It is observed that while the domain reduction increased the CPU speedup, it is less compared to the speedup from GPU. The GPU model presents modelers with more flexibility to be less precise with modeling domain extent, thus reducing additional time in preprocessing flood models, which is significant for emergency operations. Third, the effect of spatial resolution on speedups is studied. It is observed that the parallel processing power of GPU is more evident at higher spatial resolution with a larger number of grid cells, which better incorporates complex topography and flow characteristics and is preferred for flood studies. Overall, the Flood2D-GPU flood model provides a useful parallelization approach implementing

the full dynamic wave permitting more accurate and faster flood simulation results to be obtained. The future of GPU implementation in flood modeling has great potential with developments in GPU hardware, software and ever increasing availability expected.

CHAPTER 3

MONTE CARLO BASED FLOOD MODELING FRAMEWORK FOR ESTIMATING PROBABILITY WEIGHTED FLOOD RISK

3.1 Introduction

Floods have disastrous effects in terms of casualties, economic impacts, and infrastructure damage (Morss et al., 2005; European Parliament, 2007; Carter, 2009). Arguably, the advancement of computer models has had one of the more significant impacts on the ability to plan, forecast and respond to flood events. In the US, flood models were first applied to simulate floods in 1960s (Crawford and Linsley, 1966), and have since been used to enhance engineering design, planning, floodplain delineation and emergency response.

However, flood models are but conceptualizations of reality and hence, reduce physical complexity through simplifications of systems of equations (Wagener and Gupta, 2005). The model parameters and other inputs are calibrated with observed data, and when needed, the parameters are altered or forced (within their acceptable ranges) to fit the model predictions to the observed data, and the validated model is used for prediction. However, prior estimation of feasible ranges of parameters does not guarantee the model prediction within a close range of observations, especially when it is extrapolated to other problem locations. The lack of correlation between conceptual

model parameters and physical watershed characteristics results in significant uncertainty in prediction, especially if the model is extrapolated to predict the system behavior at a different location and/or flood event. Advances in computer technology, including geographic information systems (GIS) and remote sensing techniques, have enabled modelers to incorporate the spatial variability of parameters representing hydrologic and hydraulic characteristics at different locations in the system. Nevertheless, comprehensive representation of natural processes using flood models will remain for the foreseeable future macroscopic in comparison to reality.

Flood model uncertainty is of critical concern, especially when modeling results are used to set policy, decision making and emergency planning. Not acknowledging uncertainty could result in wasteful overdesign of mitigation measures, or could lead to inadequate preparation for potential situations, or, even worse, failure of hydraulic systems. Thus, a careful and detailed calibration and prediction uncertainty analysis is required for successful application of hydrologic and hydraulic models in water resources studies (Duan et al., 1992; Beven and Binley, 1992; Vrugt et al., 2003; Yang et al., 2008; Van Griensven et al., 2008).

The common approach in flood modeling still remains applying one-dimensional (1D) deterministic hydraulic models that generate floodplain boundaries depicted using a single boundary of inundation. Recently, the advantages of two-dimensional (2D) hydraulic models have been documented (Zhang and Cundy, 1989; Samuels, 1990; Tayfur et al., 1993; Knight and Shiono, 1996; Bates et al., 1998; Lamb et al., 2009; Judi et al., 2010) and recommended for application by emergency management agencies (NRC, 2009). This has led to the development of advanced 2D hydraulic models. Despite

these developments in modeling floods, the most common representation of simulation results remains a deterministic flood inundation map based on a single simulation. Unfortunately, this relies on the use of a single parameter set and does not account for the uncertainties in the modeling process (Bates et al., 2004) and may lead to an inaccurate hazard assessment (Di Baldassarre et al., 2010).

To remedy the shortcomings of a single deterministic simulation of a floodplain, probabilistic modeling approaches are emerging. Probabilistic flood mapping is designed to incorporate uncertainty from input data and model parameters, represent spatial and temporal risk, and present flood maps in terms of probabilities and percentages (Romanowicz and Beven, 2003; Aronica et al., 2002; Bates et al., 2004; Hall et al., 2005; Pappenberger et al., 2006; Di Baldassarre et al., 2010).

Smemoe et al. (2007) present a probabilistic flood mapping approach using Monte Carlo analysis and the 1D HEC-RAS model from the US Army Corps of Engineers. The methodology generates a spatially continuous flood probability map using random samples of flood flows, rainfall, the Curve Number (CN) and Manning's roughness coefficient (n). Aronica et al. (2002) present an approach applying a 2D hydraulic model to generate a flood probability map. Their approach uses LISFLOOD-FP, a simplified 2D raster-based flood inundation model, and randomly samples from probability distributions of model parameters and represents the flooded and nonflooded areas in a binary pattern for each of these stochastic samples. These areas are weighted using the difference in simulation and observed flood data. While this approach takes into account the incorporated uncertainty in model parameters, it requires vast amounts of observed flood data to compare with all the simulations. In another study, Di Baldassarri et al.

(2010) present a comparative analysis of a physically-based 2D flood model (TELEMAC-2D) in a deterministic single simulation approach and LISFLOOD-FP in a probabilistic framework. They show that using a simple planar model (LISFLOOD-FP) in a probabilistic framework results in a more descriptive flood hazard map. They conclude that visualizing flood hazards as a probability is superior to a delineated map based on a single simulation. Apel et al. (2006) describe a flood risk analysis framework that includes hydrological input, flood routing and consequent failure of flood protection structures, inundation and property damage. They applied the framework to study the flood risk of the Rhine River in Germany using simple representations of complex deterministic models.

Although simplified 2D planar hydraulic models applied in a probabilistic framework have been shown to be superior to a deterministic approach, needed improvements remain. A key advancement is to incorporate the use of a physically-based 2D hydraulic model because of the improved spatial representation and accuracy of flood depths and velocities compared to 1D and simplified 2D planar models (Bates et al., 2004). The constraint to using physically-based 2D hydraulic models in probabilistic frameworks has been the simulation time requirement for each simulation, let alone numerous simulations in a Monte Carlo analysis. Thus, probabilistic analyses with 2D hydraulic models have been limited to a smaller number of scenarios or smaller spatial domains (Sayers et al., 2000; Buijs et al., 2003; Lamb et al., 2009; NRC, 2009;). Recently, one solution to the simulation time constraint of 2D hydraulic models has been programming using Graphics Processing Unit (GPU) approaches (Lamb et al., 2009; Kalyanapu et al., 2011). Extending this computational advance, the objective of this

chapter is to present a Monte Carlo based flood inundation simulation framework that generates probability weighted flood risk by applying a GPU-enhanced 2D hydraulic model. The analysis framework produces a flood probability map, intersects it with the spatially-distributed depths and velocities, and determines the flood risk using a depth-velocity-risk relationship. The chapter includes a description of the framework, methods and a demonstration to determine the flood risk for a 1% flood event (meaning a flood event with a 1% chance of occurring every year) in the Swannanoa River in North Carolina.

3.2 Methodology

The Monte Carlo flood risk modeling framework presented here has three modules, as shown in Figure 3.1: Monte Carlo Analysis, Geospatial Output Analysis and Risk Map Development. The details of these three modules follow.

3.2.1 Monte Carlo Analysis

The Monte Carlo Analysis module executes a process to randomly sample flood model parameter or input variables (e.g., surface roughness, peak flow discharge, etc.) to incorporate uncertainty into the flood inundation analysis. In this chapter, the framework is demonstrated using the peak flow as the random variable. A random number generator is used to select values from a user-specified probability distribution function or PDF (e.g., Uniform, Normal and Log-Normal). The Uniform PDF is the default distribution because it does not assume prior knowledge of parameter distribution, it is appropriate in the absence of verifiable data, and it provides equal probability throughout the specific

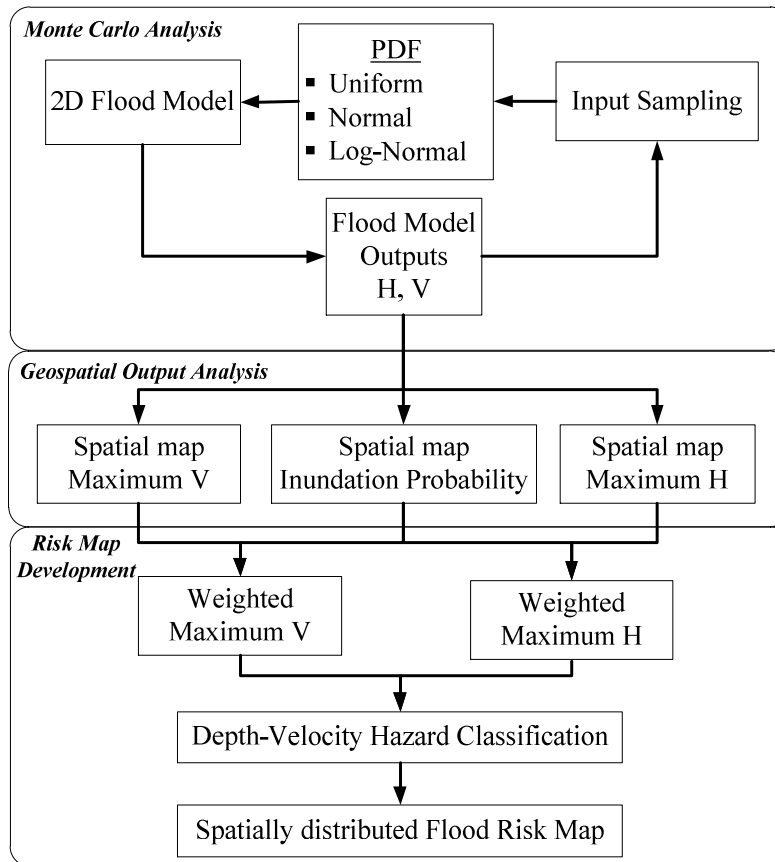


Figure 3.1. Monte Carlo Flood Risk Modeling Framework

parameter range (Beven and Binley, 1992; Freer et al., 1996).

The randomly generated input parameter sets are then used to drive a 2D hydraulic model. In this study, a new GPU flood model (Flood2D-GPU) developed in NVIDIA's CUDA programming environment is used (Kalyanapu et al., 2011). The modeling framework uses a 2D unsteady numerical flood model (Flood2D-GPU) that solves the nonlinear hyperbolic shallow water equations using a first-order accurate upwind difference scheme. These equations are developed from the Navier-Stokes equations by integrating the horizontal momentum and continuity equations over depth often referred to as the depth averaged or depth integrated shallow water equations (i.e.,

Saint Venant equations). The nonconservative form of the partial differential equations is (Tingsanchali and Rattanapitikon, 1999; Judi et al., 2010; Kalyanapu et al., 2011):

$$\frac{\partial h}{\partial t} + \frac{\partial uh}{\partial x} + \frac{\partial vh}{\partial y} = 0 \quad \text{Continuity Equation (3-1)}$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + g \frac{\partial H}{\partial x} + g S_{fx} = 0 \quad \text{Momentum Equation in x - direction (3-2)}$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + g \frac{\partial H}{\partial y} + g S_{fy} = 0 \quad \text{Momentum Equation in y - direction (3-3)}$$

where, h is the water depth, H is the water surface elevation, u is the velocity in the x -direction, v is the velocity in the y -direction, t is the time, g is the acceleration due to gravity, S_{fx} is the friction slope in the x -direction and S_{fy} is the friction slope in the y -direction. The upwind finite difference numerical scheme is used to discretize governing equations (3-1 – 3-3), as it yields nonoscillatory solutions, through numerical diffusion (Patankar, 1980; Ferziger and Peric, 2002). A staggered grid computational stencil is used to define the computational domain with the water depth (h) in the centre of the cell and u and v velocities on the cell edges. The model requires a digital elevation model to represent topography, Manning's n for surface roughness representation and a flow hydrograph. A 2D model was chosen because of its better representation of flood flow (especially in floodplains), simultaneous flood extent delineation and instantaneous flood velocities at all nodes in the computational domain. Flood2D-GPU was validated for accuracy and found to provide significantly reduced computational time (up to two orders of magnitude) compared to the same flood model implemented serially in a CPU-based

environment (Kalyanapu et al., 2011). More details about Flood2D-GPU can be found in Chapter 2 of this dissertation.

3.2.2 Geospatial Output Analysis

A flood model realization is generated for every randomly sampled parameter value and the model output is stored in GIS-ready ASCII format. Using the framework's GIS-based postprocessing environment, these ASCII files are converted into ESRI® GRID raster format, and the flood inundation delineation process is performed by computing the water depth at each model grid cell and identifying cells with a flood depth more than zero (or a user-specified threshold value) as inundated. The flood probability is calculated for each grid cell in the computational domain, as shown in the equation 3-4 below. It is calculated as the number of times a grid cell is flooded divided by the total number of flood simulations (Smemoe et al., 2007).

$$P_{cell} = \frac{\sum_{i=1}^N X_i}{N}, \quad \begin{array}{l} X_i = 1, \text{ for flooded cell} \\ X_i = 0, \text{ for non - flooded cell} \end{array} \quad (3-4)$$

where, P_{cell} = calculated flood probability at each grid cell, X_i = assigned weight based on whether the cell is flooded or nonflooded and N = total number of flood simulations.

These probabilities at all the locations in the computational domain can be represented as continuous maps (Aronica et al., 2002; Pappenberger et al., 2005; Werner et al., 2005) or as contour maps of probabilities (Smemoe et al., 2007). The probability map or contours, calculated using equation 3-4 represents the chance that a flood extent

will be bigger than the one shown in a flood inundation map. These data can be transformed into GIS raster datasets for further analysis.

3.2.3 Risk Map Development

The term ‘Flood Risk’ is commonly considered as the combination of the probability of a flood event and its impacts (European Parliament, 2007). It can be quantified using Helm’s equation (Helm, 1996), represented by the product of probability of an event occurring and its consequences. In this study, the flood risk is calculated by a modified version of Helm’s equation by replacing consequences with flood magnitude (depths and velocities), as shown in equation 3-5. The flood depths and flood velocities simulated from Flood2D-GPU are used along with the with the flood probability. A flood risk map is created using a continuous flood inundation probability map, where the probability weight included in flood risk is based on the number of simulations (equation 3-4) and not on the probability of flood event (100 years = 1% exceedence probability):

$$Flood\ Risk = Event\ probability \times Flood\ Magnitude \quad (3-5)$$

where, probability is the probability of flood inundation (equation 3-4), and flood magnitude is represented by flood depths, flood velocities, flood duration, rate of inundation extent increase, etc.

While 1D flood models can also be implemented for flood risk, this would be limited to an average representation of the flood velocities and depths as they are only simulated at limited locations. It is suggested that using a complex representation of the

flow phenomena results in a detailed flood simulation (Hunter et al., 2007). Using 2D flood models would result in a detailed spatial variation of flood risk, which can be used in flood damage assessment, flood mitigation and planning, floodplain management, etc.

From equation 3-5, probability weighted flood risks in terms of maximum velocity and maximum flood depths are computed from the model results. Based on the depth-velocity curves (shown in Figure 3.2) from ACER Technical Memorandum No. 11 (ACER, 1988), the spatial flood risk map is created. This technical document was published by the US Bureau of Reclamation to provide guidelines for dam safety hazard classification. They compiled various curves of depth versus velocity that are related to potential lives-in-jeopardy. From this collection of curves, a depth-velocity danger level relationship corresponding to permanent residences, commercial and public buildings is employed. The flood danger level is classified as low-danger zone, judgment zone and high-danger zones. For each grid cell in the low-danger zone, the possible lives-in-jeopardy is assumed to be zero. In the high-danger zone, lives-in-jeopardy is assumed to be 100% of the total population in the grid cell. The judgment zone represents a zone where the lives-in-jeopardy is considered to be variable between zero to 100% and it is up to the analyst to use engineering judgment (ACER, 1988). For this study, the loss of life in a judgment zone is assumed to be 50% of the population in a grid cell. Even though these guidelines are developed for estimating downstream hazards for dam break events, it is applied here to demonstrate the use of probabilistic flood approaches for flood risk/hazard assessment.

To quantify the impact of using the probabilistic approach compared to the deterministic approach, the loss of life resulting from flooding is calculated. To quantify

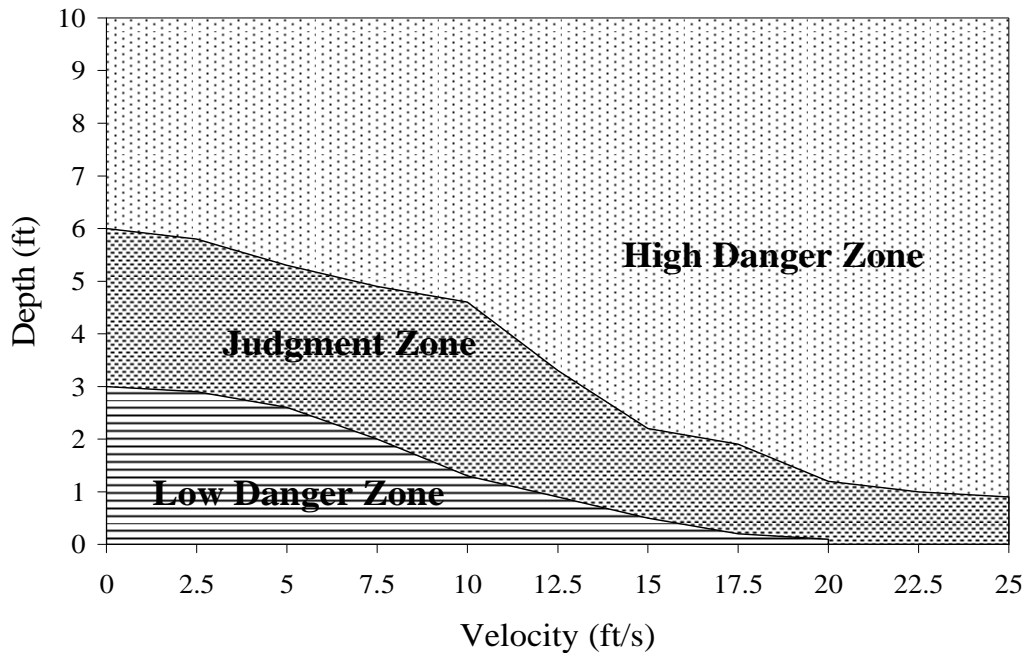


Figure 3.2. Water depth - flow velocity hazard classification diagram for flood risk classification (Source: ACER Technical Memorandum No. 11, 1988)

the loss of life, two factors are identified to be important (Graham, 1999): the population at risk (PAR), which is the number of people occupying the floodplain and, the danger level of flooding. This study takes the spatial variability of these two factors into account for each location in the area of interest. The PAR is determined by using the census block data, in a vector polygon shapefile format, from the North Carolina OneMap program which is directed by the North Carolina Geographic Information Coordinating Council (GICC) (NCONEMAP, 2011). The shapefile layer contains census blocks along with the population in each block. After importing this layer into GIS, the population density is calculated by using the total population in each census block and its area. This shapefile is then converted to a raster layer with the same spatial resolution as the rasters developed from Flood2D-GPU simulations. This population density is converted to PAR by

multiplying the population density with the grid cell area, resulting in distributed PAR layer. This approach has also been applied in studies involved in calculating the lives-in-jeopardy due to flooding and the population affected by flooding (Frey et al., 2010; Qi and Altinakar, 2011). The lives-in-jeopardy for each grid cell is calculated by multiplying the PAR with the percent loss-of-life values associated with the low, judgment and high danger zones of the spatially varied flood risk map. Aggregating the lives-in-jeopardy values for all the grid cells results in the total lives-in-jeopardy for the floodplain.

3.3 Case Study

To demonstrate the framework, a spatially distributed flood risk map is generated for a 1% river flood event in the Upper Swannanoa River. The Swannanoa River watershed is located in the mountains of western North Carolina in Buncombe County (Figure 3.3) from Asheville to Montreat in the state of North Carolina. It is part of the larger French Broad River Basin. This area is selected because of its proximity to the southeastern coast of the US, exposing it to the potential path of flood-causing hurricanes and tropical storms. Communities in the Swannanoa River watershed have been severely affected through flooding by Hurricanes Francis and Ivan in 2004, including Montreat, Black Mountain, Swannanoa and Asheville. For this study, the 32 km Swannanoa River reach is selected with a drainage area of 133.1 sq. km, upstream of the confluence of the Swannanoa River and French Broad River, including the cities Black Mountain, Swannanoa, Asheville and part of the town of Woodfin.

As stated earlier, Flood2D-GPU in the Monte Carlo framework operates with an input hydrograph. The input flow hydrograph is selected as the random variable for this

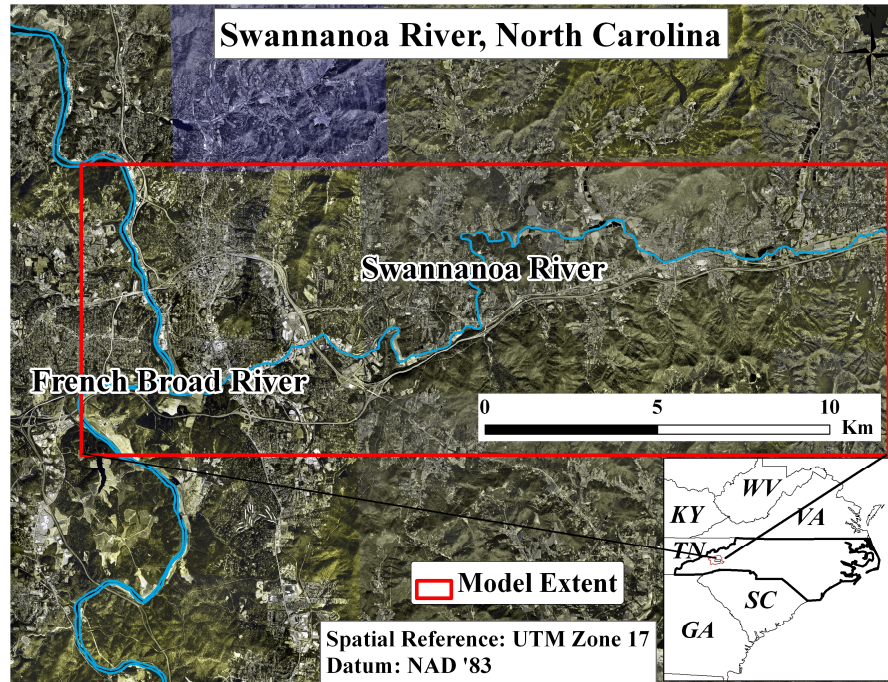


Figure 3.3. Swannanoa River flood study area

study. The 1% peak flow for the river reach is determined using US Geological Survey (USGS) flood frequency regression estimates. To create the hydrograph corresponding to this flow, we chose to use a generic hydrograph shape created for the watershed using HEC-HMS and the 100-year, 24-hr SCS type II design rainfall event with a depth of 155.7 mm selected from the National Oceanographic and Atmospheric Administration, Atlas 14 Precipitation–Frequency Atlas of the United States (Bonnin et al., 2004). The general hydrograph shape is scaled proportionally to a new hydrograph using the randomly sampled 1% peak discharge value drawn from a uniform distribution having a central value of 270 cubic meters per second (cms). The 270 cms value is the 1% annual peak discharge determined using the regression equations from the USGS Scientific Investigations Report (Weaver et al., 2009). The range of the uniform distribution is set to be within the standard error of prediction of the regression estimates found in the

USGS study (peak discharge range is from 180.7 cms to 404.6 cms), as shown in Figure 3.4.

It should be noted that a different range of peak flows could be used with a different probability distribution, but the objective of this study is to incorporate the uncertainty introduced by varying the flood peak discharges. To demonstrate the probabilistic approach, a total of 50 samples of these flows are extracted and flow hydrograph is proportionally adjusted for the new flow value. The sample size of 50 is a lower sample size for Monte Carlo simulations, which typically range about 10,000 samples or more. However, the model outputs from Flood2D-GPU were very data-intensive (50 random samples resulted in 10.8 Gigabytes of data). Thus, this arbitrary lower sample size of 50 was chosen in this study. Flood simulation is performed on each

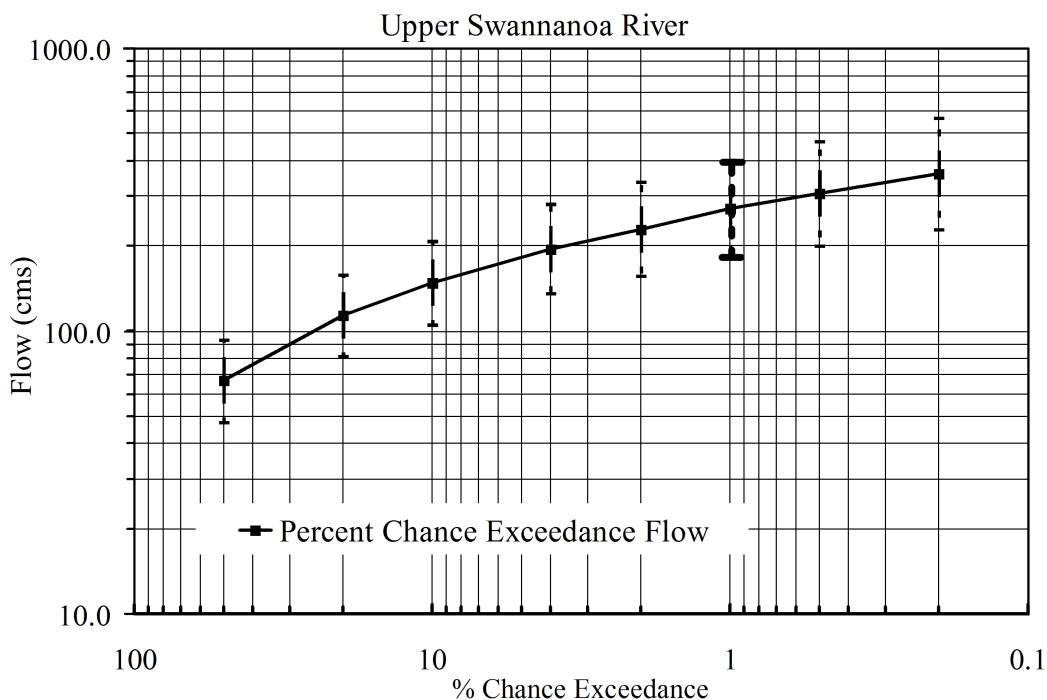


Figure 3.4. Flood Frequency characteristics curve for upper Swannanoa River. (1% event is highlighted in black)

of these samples using the Flood2D-GPU model. The flood model outputs including flood depths and flood velocities are compiled at regular intervals to capture the flow development and to use for probability weighted flood risk calculations. A flood depth threshold of 10 cm was used to identify whether a grid cell is flooded or not.

3.4 Results

The time required to simulate a single flood hydrograph realization is 12.3 min, which is much faster than the 15.5 hours required for the same model to be executed in a CPU framework. For the 50 simulations, the Flood2D-GPU framework requires 10.2 hours of execution time, while the same framework implementing a CPU-based code requires 32.3 days.

Figure 3.5 shows the 1% varying flood risk near Biltmore Village. It shows a spatially varying flood risk map near the confluence of Sweeten Creek and Swannanoa River. Along the Swannanoa River, the risk extending into the floodplain for at least 150 m is classified as high danger, which means significant risk for damage to buildings. Flood risk is clearly conveyed in this probabilistic map. Although flood risk maps produced with a deterministic approach can communicate risk, it will not cover a range of possible scenarios. For floodplain management purposes, including various possible scenarios in consideration of flood risk provides more detail and a greater confidence in risk.

To demonstrate the improvement of using a probabilistic approach to estimate flood risk over a deterministic approach, flood risk maps are generated using a single flood simulation (deterministic map) with the 1% flood hydrograph and using 50 flood

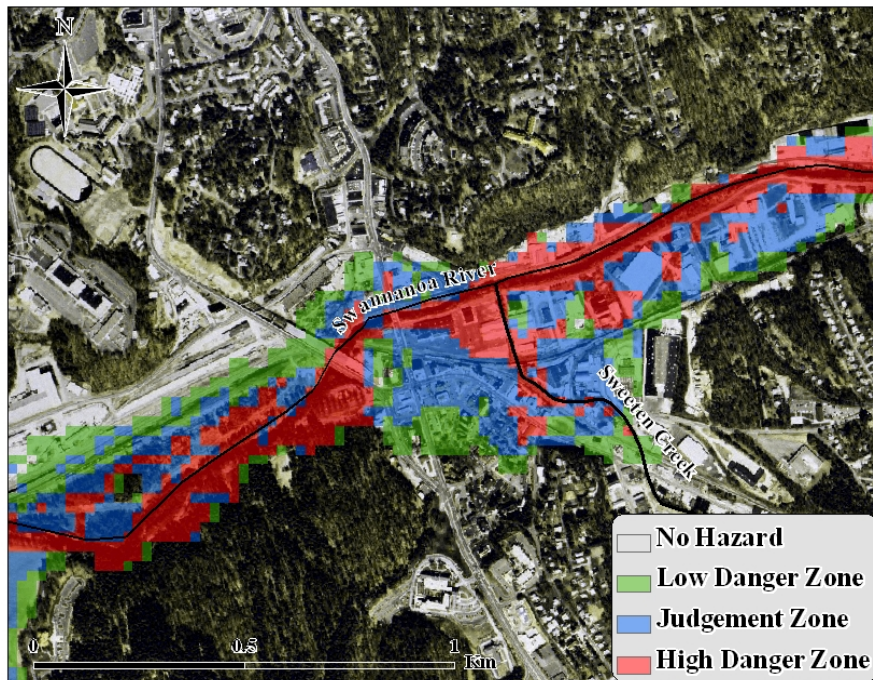


Figure 3.5. Spatially varied 1% Flood Risk map for Swannanoa River, North Carolina

simulations with randomly sampled flood hydrographs within the standard error of prediction of the peak flood discharge.

Figure 3.6 presents the spatially varied flood risk of 1% flood event at a residential area near Asheville, with potential high flood risk, along the Swannanoa River and its urban floodplain. The top half of the figure displays the flood risk map based on a single simulation (Map A). The bottom half of the figure, presents the probability weighted flood risk map (Map B) at the same location of the river reach estimated using 50 different random samples. To quantify the variability in the flood risk maps, an error matrix is presented in Table 3.1. This table simultaneously presents the distribution of cells that are classified at a certain hazard level in both the probabilistic and deterministic maps. For example, 1892 cells are classified as “No Hazard” in Map A while these cells are classified as “Low Hazard” in Map B. Of all cells that are classified with a certain

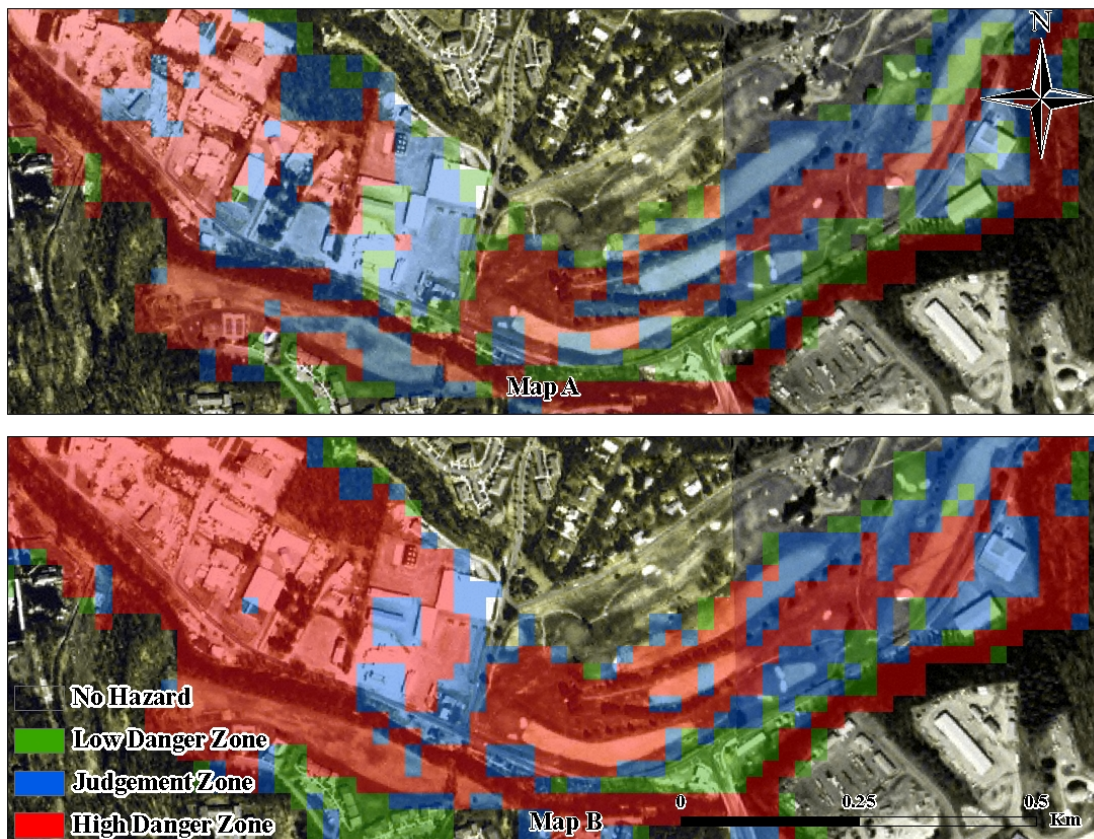


Figure 3.6. 1% event Flood Risk map near Asheville for Swannanoa River, North Carolina; Map A – single simulation; Map B – 50 simulations.

Table 3.1. Error matrix comparing the flood risk maps using deterministic approach (Map A) and probability weighted approach (Map B)

| Map A (no. of grid cells) | Map B (no. of grid cells) | | | |
|------------------------------|---------------------------|---------------|------------------|----------------|
| | No Hazard | Low Hazard | Judgment Zone | High Hazard |
| No Hazard | 360,079 | 1892 | 88 | 37 |
| Low Hazard | 0 | 2207 | 1639 | 97 |
| Judgment Zone | 0 | 0 | 2348 | 1685 |
| High Hazard | 0 | 0 | 0 | 6760 |

hazard (i.e., 16753 cells as Low to High hazard in either maps), 67.5% of these cells are identified with the same hazard in Map A and Map B (diagonal values in Table 3.1). Map B has 32.5% of the cells that are classified at higher risk than those in Map A (i.e., when flood risk in Map A is Low, flood risk in Map B is in Judgment zone or in High hazard level). In other words, Map A underestimated the flood risk in Map B.

One may observe that areas in the urban floodplain, especially located in residential and commercial regions, that are determined as low or medium (judgment zone) in the deterministic map, have been modified to medium or high hazard zones in the probabilistic map. Map B is built taking various possible flows into consideration. Simulating many different flooding scenarios with different flow hydrographs enables accounting for many equally possible flooding scenarios with varying flood depths and velocities that could cause significantly different flood risk compared to a single flood simulation. This has significant implications in the estimation of flood damage and potentially on the floodplain management.

To further investigate the reason behind the underestimation observed in Map A, the topographic slope of the area is derived. Figure 3.7 presents the spatial distribution of the relative underestimation by Map A overlaid with the topographic slope of the region. The Swannanoa River runs through the areas with steep slopes (Fox et al., 2008). Most of this underestimation is observed upstream of the steep and narrow section (slope > 10%) of the river. The main channel of the river narrows in this section, making it a channel dominant flow. This narrowing of the channel constricts the flows upstream causing back water effect resulting in significant flooding and flood risk. Map B is based on 50 different flood hydrographs, of which some simulations have hydrographs with more than

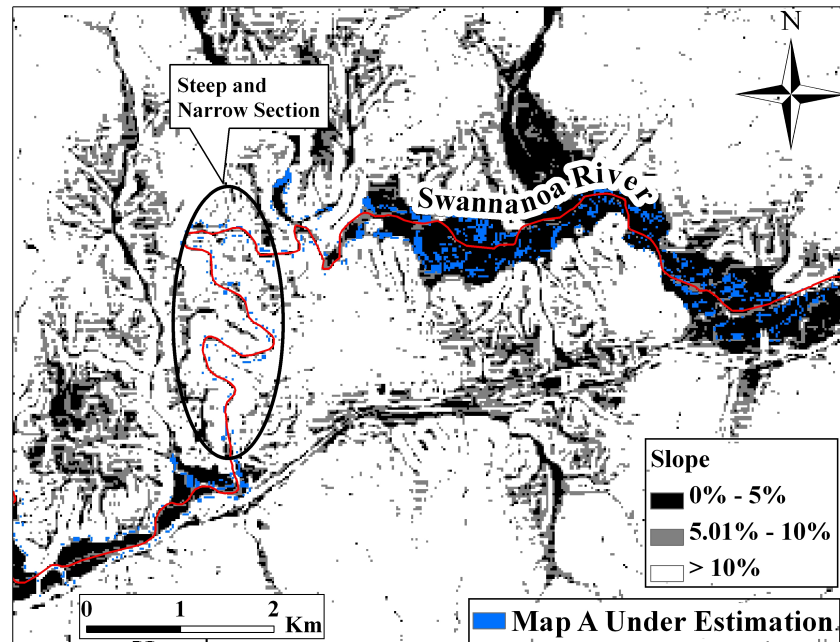


Figure 3.7. Spatial distribution of relative underestimation of map A

270 cms peak discharge. Thus, during those high flow conditions, the area upstream of the steep section is flooded, which is observed in Map B. But, Map A is based on only one flow condition (peak discharge = 270 cms) and thus represents only one of many possible flooding situations. Thus it is evident that a probabilistic flood risk map incorporates flow conditions of a statistical design event with its potential variability due to its inherent uncertainties.

The improvement in the flood risk estimation using a probabilistic framework is also quantified in the histogram plot of the number of cells in each flood risk category shown in Figure 3.8. This histogram was created applying the framework using 1, 5, 10, 25, 40 and 50 samples. It is important to note that as the number of random samples increase, more locations are classified to a higher risk level. With increasing random samples, the cells with low hazard designation increased by 3.8%, cells within the

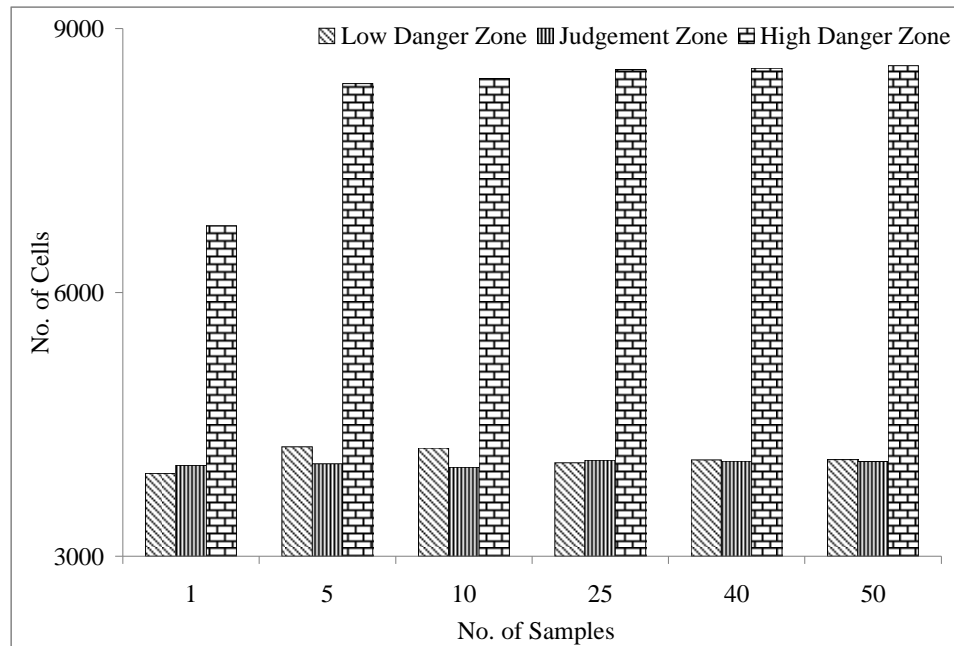


Figure 3.8. Flood Zone Variation with sample size

judgment zone almost remained constant and the number of high danger zone cells increased by 21.2% between using 1 sample (deterministic approach) and 50 samples (probability weighted approach).

To quantify the improvement of using the probabilistic approach versus the deterministic approach, the lives-in-jeopardy values from the two approaches are estimated as explained in the methodology. For the deterministic approach, the lives-in-jeopardy is estimated to be 925 persons while the probabilistic approach resulted in 1105 persons. The deterministic approach underestimated the loss by 15.98%. The ‘lives-in-jeopardy’ estimates are not realistic, because this analysis does not reflect the human response to disasters. It is natural for people in jeopardy to evade danger. So, when a building is being flooded by an overflowing river, there usually is enough time for the

people within the building to evade because flood water spreads gradually over into floodplains. This is not accounted in the calculations presented here.

3.5 Summary

This study presents a computationally efficient Monte Carlo based 2D flood inundation framework for determining probability weighted flood risk. The framework consists of three modules. The first module implements a Monte Carlo Analysis based on user-defined random sampling of flood model parameters and input variables. Each sample of parameters and variables is used to execute a GPU-enhanced 2D hydraulic model to compute spatially distributed and time-variant output of flood depths and velocities and inundation extent, which are passed to the next step of the process. In the Geospatial Output Analysis module, the outputs from the simulations are compiled. Each grid cell in the model domain is analyzed to quantify the probability of being inundated. In the Risk Map Development module, a flood risk map is produced by weighting the spatial maps of depths and velocities by the flood inundation probability and determining risk from a Water Depth-Velocity Hazard Classification Diagram.

A 1% design flood event for Swannanoa River in North Carolina is used and flood risk maps were developed using single simulation (deterministic) and multiple simulation (probabilistic) approaches. The deterministic approach underestimated the flood risk by 32.5% relative to the probabilistic approach. As the number of samples increased, compared to the deterministic approach, probabilistic approach estimated areas with low hazard and high hazard increased by 3.8% and 21.2%, respectively. This difference in flood risk translates into significant underestimation of lives in jeopardy in

populated areas. This is quantified by calculating the lives-in-jeopardy, where the probabilistic approach simulated 1105 lives are lost while deterministic simulated 925. While these numbers are not realistic, they do reflect an underestimation of 16%, which can be very significant if the floodplain is highly populated. The outcome of this study shows the improvement of the probabilistic flood approach compared to the single simulation approach. The new Monte Carlo flood risk modeling framework has the ability to provide improved accuracy of flood risk information and in general greater insight into the spatial distribution of flood risk useful in making decisions. By using a single case study area, it was observed that simulating different flooding scenarios with different flow hydrographs enables accounting for many possible flooding scenarios with varying flood depths and velocities that could result in significantly different flood risk compared to a single flood simulation. This has significant implications in the estimation of flood damage and potentially on the emergency management.

CHAPTER 4

ANNUALIZED RISK ANALYSIS APPROACH TO RECOMMEND APPROPRIATE LEVEL OF FLOOD CONTROL

4.1 Introduction

With more than 2.8 billion of the world's population living within 15 km of rivers, flood risk is a major challenge facing municipal and government planning and development agencies (Small and Cohen, 2004). In the future, flood risk and damages are expected to continue to increase as population grows, people move into at-risk locations, and climate changes (McCarthy et al., 2001; Montz and Grunfest, 2002). To mitigate flood risk, a variety of best management practices are available, including structural flood control measures like detention basins, levees, dams and nonstructural measures like flood proofing, permanent evacuation and relocation, land use management, flood insurance, building codes, flood warning and education. In the United States (US), flood risk management has primarily focused on structural flood control programs since the passage of the Flood Control Act of 1936 (IFMRC, 1994). In the 1960s, nonstructural alternatives were encouraged and eventually became the preferred approach in most cases because of their cost effectiveness. Recently, integrated approaches to flood risk management, which acknowledge the interrelationships between structural and

nonstructural measures, have emerged as the preferred path forward in managing flood risk (Hall and Solomatine, 2008). Widespread implementation of integrated flood risk management still lags, but is gradually evolving (Barredo and Engelen, 2010).

Design of any flood risk management system is based on the concept of return period or exceedance probability (Tung, 1996). Usually a 100-year (1% exceedance probability) flood event is used as a standard design criterion for flood control, protection and mitigation systems such that the system performs without failure for flood magnitudes up to or less than magnitude of the design event.

While evaluating design alternatives for flood risk management, reduction in potential flood damage indicates the effectiveness of a design alternative. Economic analysis plays a significant role in this process (Wurbs, 1996). One of the commonly used hydro-economic models is the U.S. Army Corps of Engineers' (USACE) Hydrologic Engineering Center (HEC) Flood Damage Analysis (HEC-FDA) framework for determining the Expected Annual Damage (EAD). EAD is the average damage determined from floods of different annual exceedance probabilities over a long period (NRC, 2000). Details on this approach, which is also referred to as "frequency method," is widely available in the literature (e.g., Tung, 1996; USACE, 1996; NRC, 2000; Hardmeyer and Spencer, 2007; Xu et al., 2007). An appropriate alternative for flood damage reduction is selected based on certain decision criteria after comparing the EADs for the various alternatives and their corresponding capital costs.

As a standard practice in flood damage estimation, flood events at various return periods (e.g., 2 yr, 5 yr, 10 yr, 25 yr, 50 yr, 100 yr, 250 yr and 500 yr) are selected (Ahmad and Simonovic, 2001; Qi and Altinakar, 2011). These discrete return periods

represent the flood damages distribution that is often assumed to be a continuous curve for determining the EAD. The effect of this assumption on the EAD needs to be verified. A better representation would be to directly use the historical stream flow data and designate these flows with their annual exceedance probabilities. This would result in a continuous flow distribution instead of being represented by discrete return periods and enable historical peak flow conditions to be explicitly considered in flood damage estimation. If the length of historical flow records is large enough to capture a wide range of possible flows at a certain river reach, a detailed continuous flow distribution can be derived by sampling from the range of flows, using Monte Carlo sampling.

While EAD is not used for formulating design alternatives, the annualized risk concept can be applied to analyze and compare design alternatives. Annualized risk is the product of the flood damage and the probability of a flood event:

$$R_{Annual} = D_{Flood} \times P_T \quad (4-1)$$

where R_{Annual} is the annualized risk (\$/year), P_T is the exceedance probability of a flood event for a return period T and D_{Flood} is the flood damage associated with a flood event for the corresponding exceedance probability. In other words, it is the instantaneous EAD from the flood damage frequency curve. Using this metric in a flood damage reduction project, preliminary flood damages (D_{Flood}) can be calculated and annualized risk can be determined using equation 4-1. An annualized risk frequency curve can be plotted using exceedance probability, a sample of which is shown in Figure 4.1. Generally, smaller magnitude events (represented by exceedance probability) result in smaller damages and

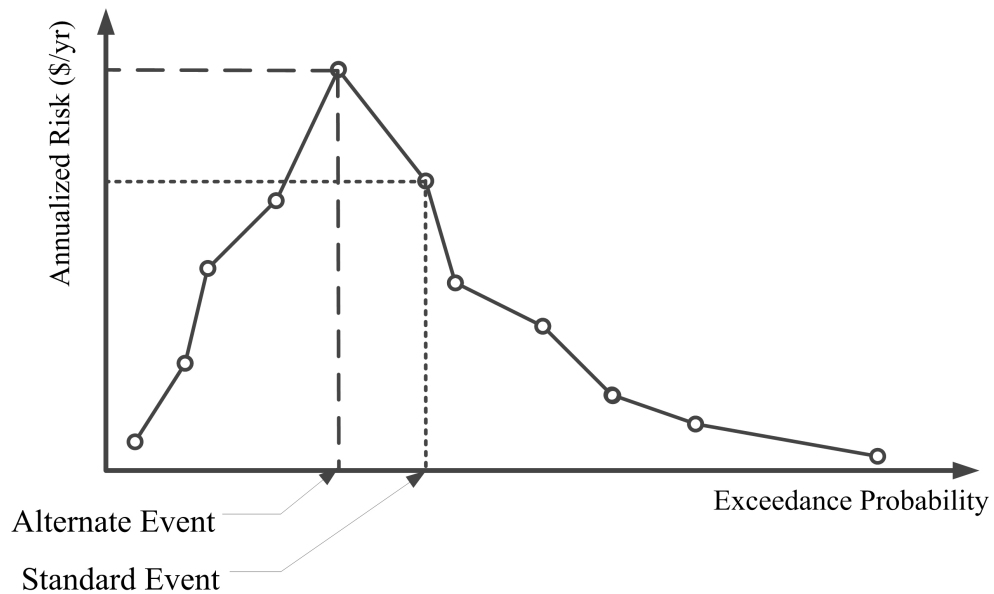


Figure 4.1. An example of annualized risk curve

thus have smaller annualized risk. As the flood magnitude increases, the annualized risk also increases, but it reaches a threshold value where it is maximized and then it starts to decline. In the declining trend of the curve, even though the flood damage is much higher, calculated annualized risk is smaller because of its associated smaller exceedance probability.

From this annualized risk curve, the design event associated with the maximum annualized risk could be considered as an alternative to an arbitrarily selected design event that is suspected to maximize flood control investment benefits. Generally, a 1% flood event is arbitrarily selected as the design event (Williams and Swanson, 1989; Marco 1994; Watt 2000, Petrow et al., 2006). However, in many situations, reasons such as financial limitations of the project, low priority for risk, etc., may encourage the use of design event lesser than standard design event. This significantly reduces the capital costs involved in implementing the project. As an example, Figure 4.2 presents a sample of the

annual cost estimates considered for improving an urban drainage system in Hong Kong, mentioned in Tung (2002). Almost 50% reduction in total cost was observed for improving the system for a 5% protection level in lieu of a standard 1% flood protection level. Thus, the design standards used in flood management significantly influence the project costs.

An integrated approach to flood risk management acknowledges that completely eliminating flood risk and flood damage is not feasible and it aims to apply multiple solutions like the combination of structural and nonstructural management practices, to cumulatively reduce the risk. In this context, it may not always be necessary to use a standard design flood event for all flood risk management alternatives for a floodplain. For example, let us consider that an integrated flood risk management study for a

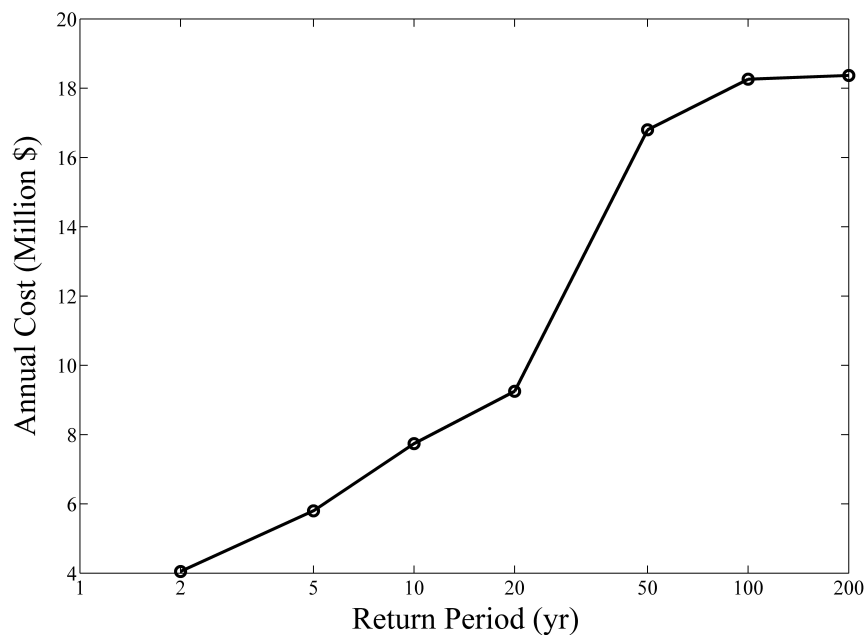


Figure 4.2. Annual Cost estimation for urban drainage system

floodplain recommends implementing a combination of levees, dams, urban storm water drainage and flood proofing, using the standard 1% design conditions, to reduce the resultant flood risk to an acceptable risk level that was determined by floodplain managers and decision makers. Now, if implementing all the above solutions except flood proofing brings down the total flood risk in the floodplain to just above the acceptable risk level, then there are two options. The first option is to implement flood proofing using the standard 1% design condition and reduce the resultant risk to much lower than the floodplain's acceptable risk level. The second option is to use a lower magnitude design flood event such that the resultant risk is less than or equal to acceptable risk level. Here, significant financial savings could be achieved by using a lower design event for flood proofing.

It is financially prudent to consider various design alternatives, including designing the systems for higher exceedance probability flood events. This study addresses the question: Is it beneficial to implement flood risk management alternatives using an alternate design event other than the standard 1% flood event? To answer this, the following evaluations are performed for the Swannanoa River using flood proofing as the flood risk management alternative:

1. Comparison of a continuous flow distribution with the standard discrete return period approach
2. Analysis of benefits in terms of flood damage reduction using annualized flood risk
3. Evaluation of reduction in flood damages by implementing flood proofing at a lower magnitude event compared to the standard 1% flood event

4.2 Methodology

4.2.1 General Flood Damage Analysis Approach

Figure 4.3 presents the process flow of the general methodology used in the estimation of flood damages to compare flood mitigation alternatives. The process requires estimation of input flood flows which can be performed using techniques like at-site frequency analysis, regionalized flood frequency analysis and hydrologic modeling (Merwade et al., 2008). In flood frequency analysis, annual peak flow discharge data are observed, preferably over a long range of historical data, and statistical information such as mean, standard deviation and skew are calculated. They are used to estimate the magnitude and frequency of peak flows and presented in terms of frequency distribution curves. For ungaged basins with less or no historical stream flow data, regional regression equations that relate flows with watershed drainage area, slope, storage and routing characteristics are used to estimate design flows (Ries and Crouse, 2002). There are also software packages available like PeakFQ by US Geological Survey (USGS) and HEC-SSP by HEC that can be used to perform flood frequency analysis. Flood frequency analysis is well established and abundant literature is found in standard manuals, publications and textbooks (IACWD, 1982; McCuen, 1998; Blazkova and Beven, 2002). Rainfall runoff models (e.g., HEC-HMS, TOPMODEL) may also be used to determine the design flood events and their magnitudes. This would require using design storms from an intensity-duration frequency relationship and generating design hyetographs and runoff is simulated using a calibrated rainfall-runoff model to determine discharge hydrograph for a design flood event.

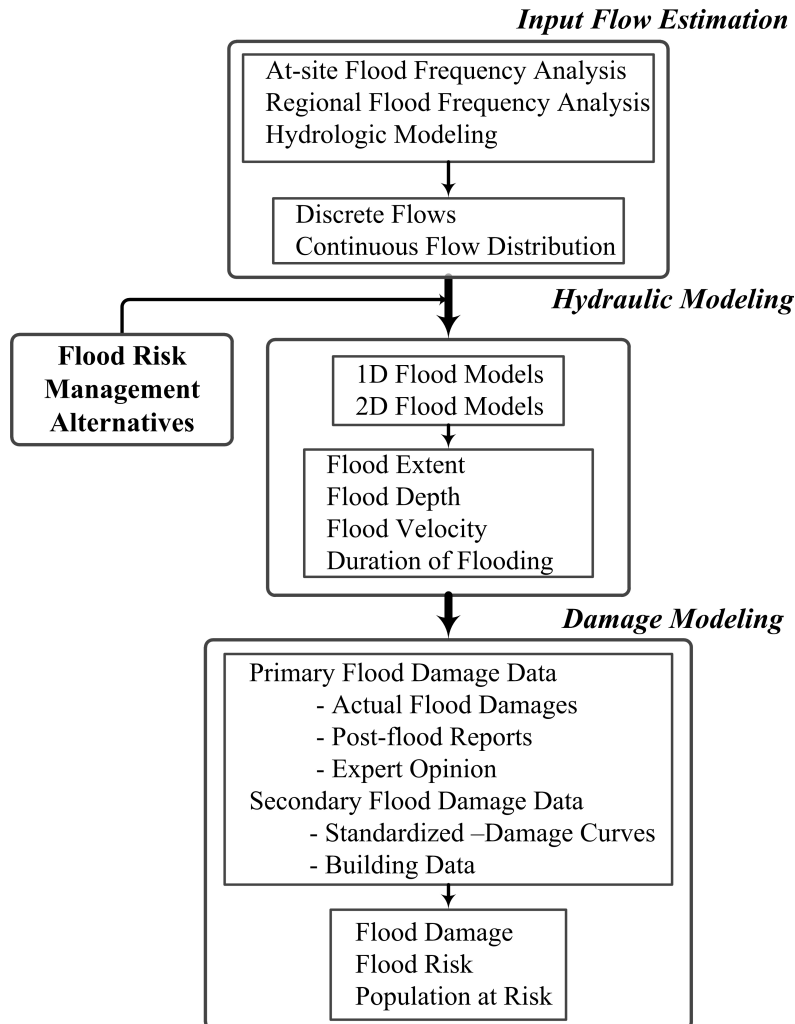


Figure 4.3. General flood damage assessment framework

The resultant flows may be in the form of discrete representation of flows or a continuous flow distribution curve. As mentioned previously, the standard practice is to perform flood modeling and flood damage calculation for flows with discrete return periods (e.g., 2 yr, 5 yr, 10 yr, 25 yr, 50 yr, 100 yr, 250 yr and 500 yr) or their corresponding discrete exceedance probabilities.

The estimated flows serve as inputs to 1D hydraulic models such as HEC-RAS, MIKE 11; 2D hydraulic models like RMA, Flo2DH, FIT2D, TELEMAC-2D, Flood2D-

GPU; integrated 1D-2D hydraulic models like LISFLOOD-FP and MIKE FLOOD, to estimate flood inundation extent, flood depths and velocities. Hydraulic modeling can be performed using either the steady flow assumption or unsteady flow analysis. The simulated flood extent, depths and velocities are combined with primary or secondary flood damage data and flood damage, flood risk or population at risk are derived.

Many computer programs and techniques, including HEC-FDA (USACE, 1996), ANUFLOOD (Smith and Greenaway, 1988), the Blue Manual (Penning-Rowsell and Chatterton, 1977) and HOWAD (Neubert et al., 2009) are available to estimate these damages in terms of direct flood damages (Viljoen et al., 2001; Hardmeyer and Spencer, 2007). To select appropriate flood risk management alternatives, this damage calculation process is repeated for different alternatives.

4.2.2 Flood Damage Approach in This Study

The following section presents the specific approach followed in this study for flood damage calculation and Figure 4.4 graphically depicts the process.

4.2.2.1 Input Flow Estimation

In this study, the input flows are estimated using regional regression equations developed by flood frequency analysis technique for ungaged basins (Weaver et al., 2009). The peak flows are generally considered at various discrete return periods or recurrence intervals as a standard way of design using hydraulic models (e.g., Ahmad and Simonovic, 2001; Morita, 2008; Qi and Altinakar, 2011). A continuous flow distribution is used in this study to show the impact of this choice.

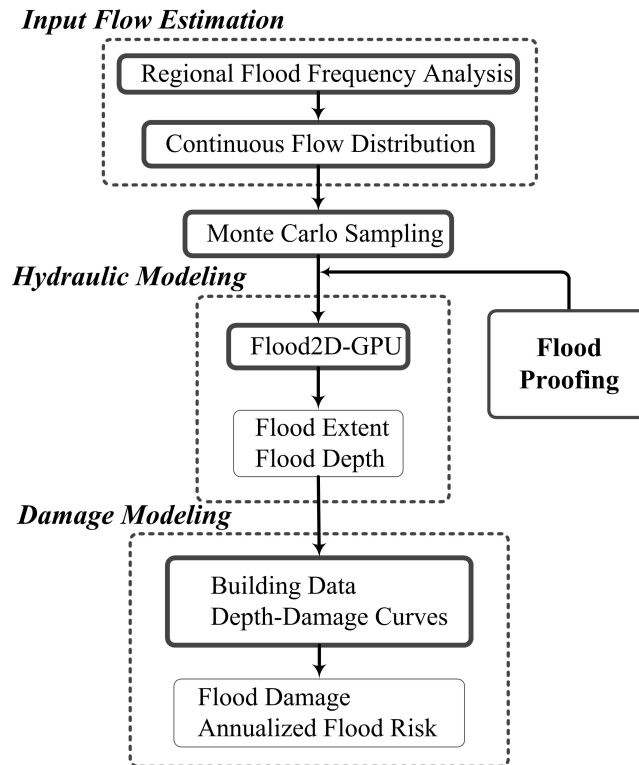


Figure 4.4. Methodology adopted in this study

The continuous flow distribution is synthesized for an ungaged basin by assuming that the regression flow estimates represent the peak flows experienced in the basin. Exceedance probabilities are identified ranging from 99.99% to 0.1% and their corresponding flows are calculated using the USGS regression equations for selected probabilities (Weaver et al., 2009). These calculated peak flows and their probabilities, which are estimated using log-Pearson Type III distribution, are plotted on a probability paper and a straight line is fit to the data. From this fitted straight line, numerous data points are extracted throughout the range of probabilities. These data points are redrawn on a plot with linear x and y axes and a synthetic continuous flow distribution curve is derived as presented in Figure 4.5. This curve represents all the possible peak flows at the location with their associated probabilities.

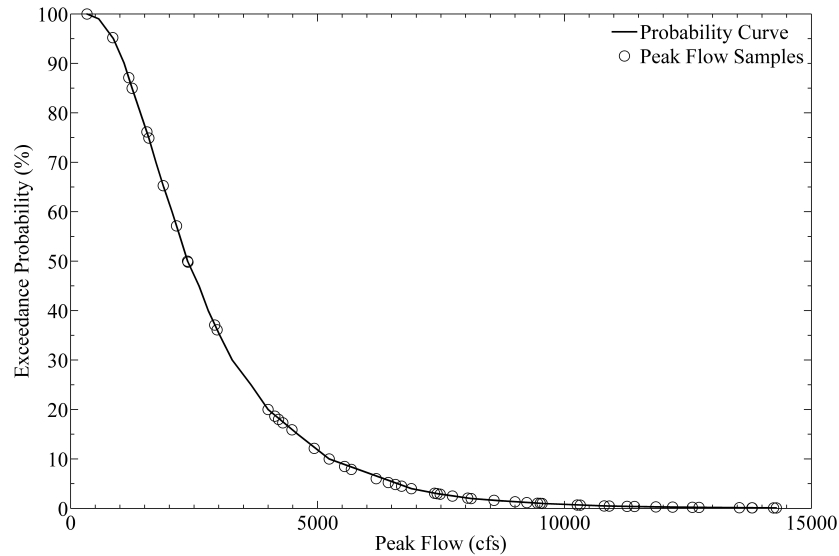


Figure 4.5. Flow distribution curve

Since a continuous flow distribution curve represents infinite flows, a sampling strategy is needed to sample from the distribution. In this study, a Monte Carlo sampling strategy is adopted to randomly sample flows. This sampling strategy is implemented by a Monte Carlo based flood risk modeling framework that was presented in Chapter 3.

From the flow distribution curve, the required number of peak flows, each representing flood events with certain exceedance probabilities, are selected. The peak flows derived from the curve can be used in steady flow modeling based on the assumption that the flood flow has been constant for a sufficiently long period so that all the area that could be flooded at that flow has in fact been flooded (Bales and Wagner, 2009). In this study, unsteady flow modeling is used, which requires the temporal variation of flow in terms of a hydrograph. Thus, the peak flow needs to be represented as a hydrograph. To do this, a generic hydrograph shape can be created using hydrologic models such as HEC-HMS and TOPMODEL and the peak flow is adjusted into a synthetic hydrograph, as depicted in Figure 4.6. For every random sample of peak flow

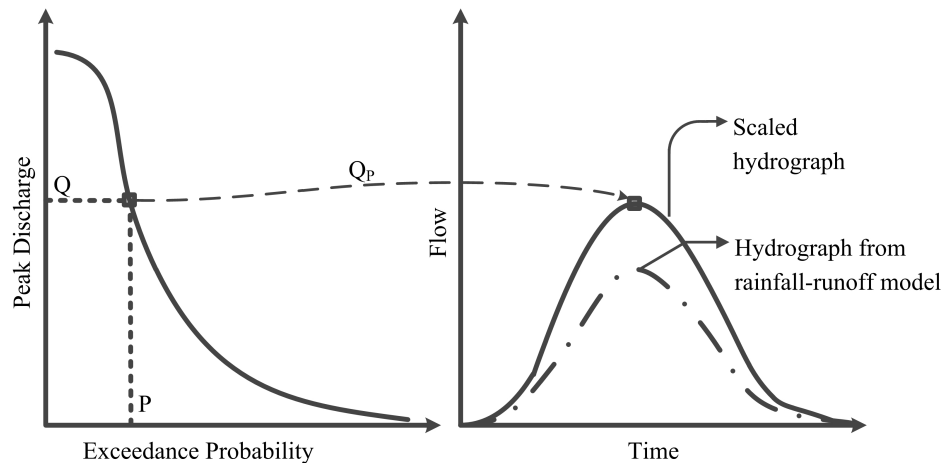


Figure 4.6. Conversion of peak flow discharges to discharge hydrograph

(Q_p) at an exceedance probability P , a scaled hydrograph shape is created using the generic hydrograph. It is to be noted that the hydrograph flood volume is not conserved through this process. The objective of this conversion is to recreate flow condition with peak flow from the probability curve and corrected hydrograph shape to simulate the hydrological response in terms of the time to peak and recession.

4.2.2.2 Hydraulic Modeling

The hydraulic model used in this study is a 2D shallow water wave-based flood model called Flood2D-GPU. A 2D flood model is selected in this study due to the advantages compared to 1D flood models, including better representation in flood flows (especially in floodplains), simultaneous flood extent delineation and instantaneous flood velocities and depths at all grid cells in the computational domain (Judi et al., 2010).

Flood2D-GPU is a GPU flood model developed in NVIDIA's CUDA programming environment (Kalyanapu et al., 2011). Flood2D-GPU is an unsteady flow model that solves the nonlinear hyperbolic shallow water equations using a first-order accurate finite

difference scheme. These equations are developed from Navier-Stokes equations by integrating the horizontal momentum and continuity equations over depth, often referred to as the “depth averaged” shallow water equations or Saint Venant equations. Equations 4-2 – 4-4 below present the nonconservative form of the partial differential equations:

$$\frac{\partial h}{\partial t} + u \frac{\partial h}{\partial x} + v \frac{\partial h}{\partial y} = 0 \quad \text{Continuity Equation (4 - 2)}$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + g \frac{\partial H}{\partial x} + g S_{fx} = 0 \quad \text{Momentum Equation in x - direction (4 - 3)}$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + g \frac{\partial H}{\partial y} + g S_{fy} = 0 \quad \text{Momentum Equation in y - direction (4 - 4)}$$

where, h is the water depth, H is the water surface elevation, u is the velocity in the x-direction, v is the velocity in y-direction, t is the time, g is the acceleration due to gravity, S_{fx} is the friction slope in the x-direction and S_{fy} is the friction slope in the y-direction. The model uses an upwind finite difference numerical scheme for discretizing governing equations (4-2 – 4-4), as it yields nonoscillatory solutions through numerical diffusion (Patankar 1980; Ferziger and Peric, 2002). Flood2D-GPU requires a digital elevation model (DEM) representing topography, Manning’s n for surface roughness and a flow hydrograph. The model has been validated and observed to be very fast (designed specifically for Monte Carlo analysis), with computational speedups between 80x – 88x compared to a regular CPU model (Kalyanapu et al., 2011). The flood depths and flood inundation extent simulated by Flood2D-GPU for each of the randomly sampled flow events is used in the damage modeling step.

4.2.2.3 Damage Modeling

The calculation of total flood damage (in \$) and annualized flood risk (in \$/year) is performed in a geographic information system (GIS) environment using ESRI® ArcGIS software. All the spatial data is represented in a raster grid cell format with the same spatial resolution as the input DEM. The flood depths simulated from hydraulic models, building data containing information about the property values of flood-prone buildings, and depth-damage relationship relating the damage and flood depths, are used in calculating flood damage. Figure 4.7 illustrates the process.

To estimate flood damages, the building data containing building footprints and their property value are required. In the absence of building footprints, land parcel data with building values may be used and the footprint of the land parcel is assumed to approximate the building(s). Based on the values in the building footprints or land parcels, a building damage density is calculated by dividing the value with the area of the grid cell. As a result, a damage density layer is created in which each grid cell represents the potential flood damage per area. In this layer, some of grid cells which do not contain buildings or which are a part of empty land are assigned a zero value.

To estimate flood damages from flood depths, depth-damage functions are used. These functions are derived through systematically applied survey procedures, and they can also be generated from insurance claims data analysis or historical flood data analysis, considering the possible damage ratio based on the given flood depths (Middleman-Fernandes, 2010). Generic depth-damage functions are developed by USACE relating flood depth (or stage height) to the damage or loss to building structures for nationwide use in flood damage reduction studies in the US (USACE, 2000). This

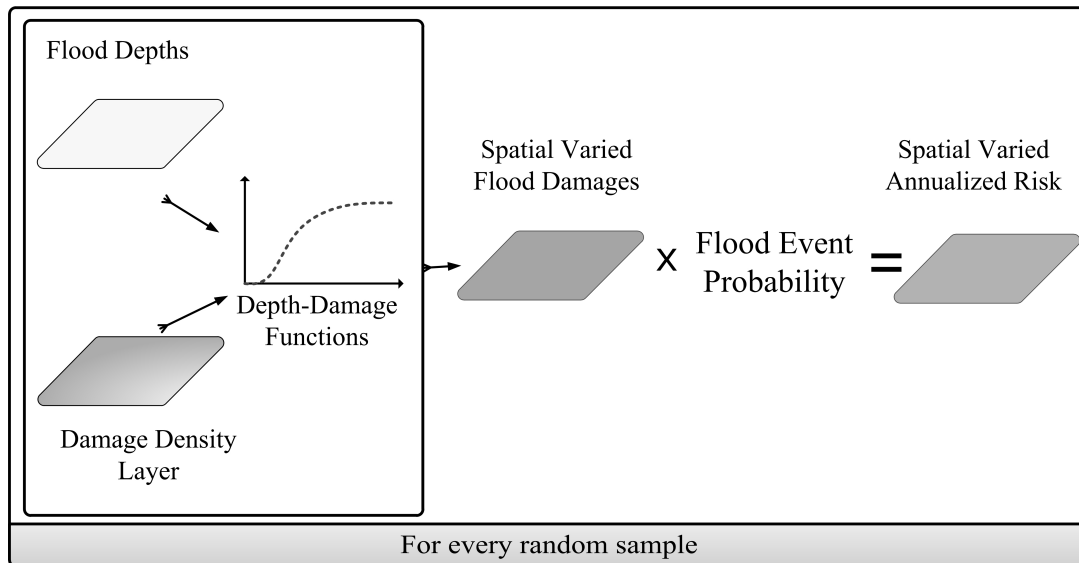


Figure 4.7. Estimation of flood damages and annualized risk

damage is expressed in terms of the percentage of the building value.

For every model realization corresponding to the random flood event from the flow distribution curve, maximum depths at all grid cell locations in the river reach and floodplain are derived. Total flood damage in a grid cell is estimated by multiplying the flood damage density, percent flood damage for the corresponding flood depth and the area of the grid cell. It is assumed that the DEM elevation in a grid cell represents the elevation of the lowest floor of the building.

Annualized risk is calculated by multiplying the flood damage with exceedance probability, as mentioned in equation 4-1. This is a measure of flood risk from a flood event with certain recurrence interval. The flood damage frequency curve and annualized risk frequency curve are generated by plotting the flood damage and annualized risk respectively, with their corresponding flood events represented by their exceedance probabilities. The area under the flood damage frequency curve is the EAD. The

annualized risk frequency curve is used to select the flood event with maximum flood risk as an alternate design event.

The flood risk management alternative used in this study is Flood Proofing. In the US, it is one of the nonstructural flood risk reduction measures that is applied to frequently flooded properties and is generally less disruptive to the environment (USACE, 1995). According to Federal Emergency Management Authority (FEMA), flood proofing is “any structural or nonstructural implementation, changes or adjustments to structures which reduce or eliminate the flood damages to real estate or improved real property, water and sanitary facilities, structures and their contents” (FEMA, 2002). It has been incorporated into several regional flood control plans (SCS, 1987, 1990, 1992, 1994) and is specifically mentioned in the Water Resources Development Act of 1996. The advantage of flood proofing is that it can be undertaken by individual property owners without waiting for government action and it can provide protection in areas where large structural projects, such as construction of dams or major waterway improvements, are not warranted (NFPC, 2011). Flood proofing can be classified into three categories: a) raising or moving the structure; b) constructing barriers to stop floodwater, and c) wet flood proofing. This study is focused on the flood proofing by raising the structure, commonly referred to as “Elevation” where the structure is raised in place so that its lowest floor is above the expected level of floodwaters, thus reducing frequency and/or depth of flooding.

To assess the reduction in flood damages, the flood damage frequency curve and annualized risk frequency curve are estimated without implementing flood proofing. Let us call this the ‘No flood proofing’ alternative. Then, these curves are generated after

implementing flood proofing for the standard 1% flood event. Let us call this the '1% flood proofing' alternative. To select the lower magnitude flood event as an alternative to 1% flood proofing the annualized risk curve from 'No flood proofing' is used. The flood event with the maximum annualized risk from this curve is selected as the alternative. Using this alternative, flood proofing is implemented and flood damage and annualized risk curves are generated. Let us call this the 'X% flood proofing' alternative, where X is its exceedance probability of the identified design event. Comparing the flood damage frequency curves, the calculated EADs and the annualized risk curves of the three cases show the reduction in flood damages and the effectiveness of the two flood proofing alternatives.

4.3 Case Study

The Swannanoa River watershed is located in the mountains of western North Carolina in Buncombe County (Figure 4.8). It is part of the larger French Broad River Basin. These two rivers are essential to the citizens and economy of Buncombe County and the entire Western North Carolina region. Communities in the Swannanoa River watershed have been severely affected by flooding during Hurricanes Francis and Ivan in 2004, including cities of Montreat, Black Mountain, Swannanoa and Asheville. Consequently, the General Assembly of North Carolina enacted the Hurricane Recovery Act of 2005, also known as Senate Bill 7 (SB 7) that jumpstarted recovery activities for the affected communities and funds to examine and implement measures to reduce flood risk and loss of life, resulting in forming the Swannanoa Flood Risk Management Project

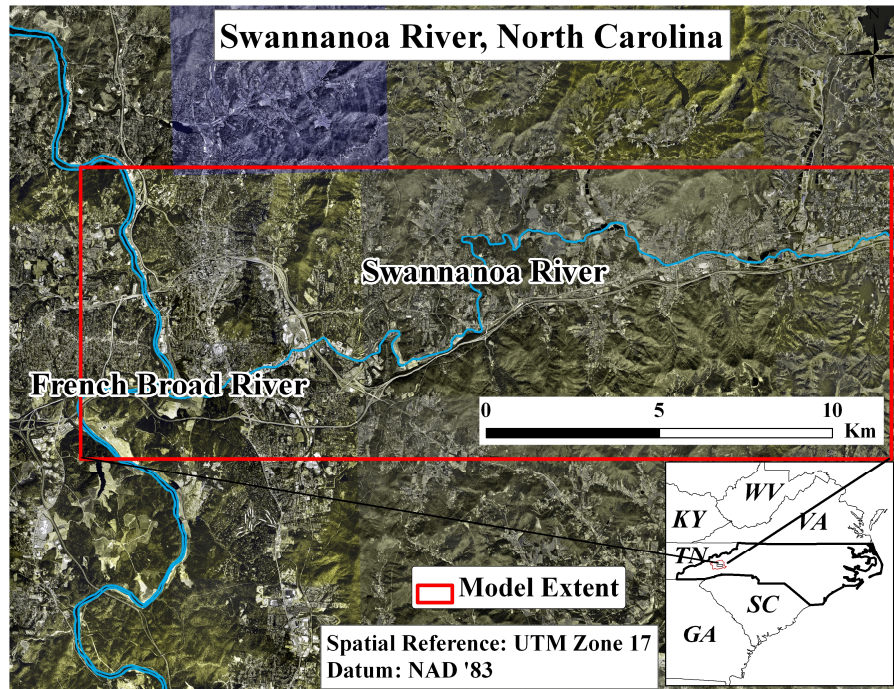


Figure 4.8. Swannanoa River flood study area

(SFRMP). More information can be found at <http://www.swannanoafloods.org/>. Detailed information, including the topographic data and economic data, are available for the watershed, making it a unique case study.

The 32 km Swannanoa River is used in this study. This reach is bounded by 133.1 sq. km upper Swannanoa watershed. The USGS has a stream flow gage (USGS 03449500) in this area. However, no historical data are available since 1931; hence, it is treated as an ungaged basin. Based on the regional regression equations developed for the study area (Weaver et al., 2009), the continuous flow distribution curve for the Swannanoa River reach is generated as explained in the Methods section. From this flow distribution curve, 50 flow samples are randomly selected with different exceedance probabilities through the Monte Carlo sampling framework. To convert the peak flow

values of the continuous flow distribution curve, a generic flow hydrograph is created for the watershed using HEC-HMS and the 100-year, 24-hr SCS type II design rainfall event with a depth of 155.7 mm from the National Oceanographic and Atmospheric Administration, Atlas 14 Precipitation–Frequency Atlas of the United States (Bonnin et al., 2004). As a result, 50 different flow hydrographs are generated to be used in Flood2D-GPU.

The Flood2D-GPU is applied using a USGS DEM at 20 m spatial resolution (obtained from <http://seamless.usgs.gov>) and a Manning's roughness value of 0.11 (McCuen, 1998) is used to represent the vegetation and light turf along the floodplain, consistent with the roughness values used in hydraulic model development by the North Carolina Floodplain Mapping Program (NCFMP, 2011). Building footprint data for the floodplain was unavailable so the Buncombe County tax parcel data (Buncombe County, 2011) is used as the starting GIS dataset. These parcel data contain the land use in each parcel along with its land value and the value of any buildings it contains. Based on the building values, damage density (\$/area) is calculated for every grid cell. Generic depth-damage relationships from Economic Guidance Memorandum (USACE, 2000), are used in this study. The generic depth-damage curve corresponding to two or more stories with no basement is used to relate the flood depths and the percent damage to the structure. Thus, to estimate the flood damage due to inundation in a grid cell, the percent damage is multiplied with damage density and grid cell area. By aggregating the damages for all the grid cells, the total flood damage for a flood event is calculated. From this value, annualized risk is derived. Damage to public infrastructure, including pipelines, roads, bridges, etc., and damage to vehicles are not considered in this study as their information

is not available. This is a simple flood damage estimation at a river scale. A detailed estimation would involve incorporating individual building details, which is very cumbersome, data intensive and would require very high spatial resolution (at least 5 m). The flood damage to contents within the structure is not considered here, as this study implements a coarser approach of flood damage calculation.

To implement flood proofing using the elevation approach for a certain design event, the floodplain corresponding to that design event is considered. All the grid cells within the design floodplain that are prone to flood damage (i.e., damage density > \$/area) are considered to be flood proofed. For example, if the flood proofing is designed for a 1% flood event, then the 1% floodplain is selected and all the grid cells within that floodplain prone to flood damage are identified to be flood proofed. The height of flood proofing for these cells is selected as the maximum simulated flood depth for the design event plus an additional 1ft of freeboard, such that the resultant elevation of these cells would be above the flood level of the design event, per FEMA guidelines (FEMA, 2002).

The maximum flood depths corresponding to the 50 different flood events that are used in estimating damages from the depth-damage curves are reduced by the height of flood proofing for these cells. For instance, if the height of flood proofing for a cell is 3 ft, and the maximum flood depth at that cell for a flood event is determined to be 10 ft, then the resultant flood depth used in flood damage calculation is 7 ft. Depending on the flood event that is used as the design condition for flood proofing, the height of flood proofing for cells vary, with typically larger magnitude events requiring higher flood proofing heights.

4.4 Results and Discussion

This section presents the results of flood damage calculations for the Swannanoa River in terms of flood damage frequency and annualized risk curves generated for three alternatives: No flood proofing, 1% flood proofing and X% flood proofing. The 'No flood proofing' alternative does not include flood proofing. The 1% flood proofing is the alternative where flood proofing is designed for the 1% flood event. From the annualized risk curve of the No flood proofing alternative, the X% flood proofing alternative is identified to be 12% (see below), thus it is called the 12% flood proofing alternative and flood proofing is designed for the maximum flood depths simulated for that event.

4.4.1 No Flood Proofing Alternative

Figure 4.9 presents the flood damage calculated using 50 various flood events represented by their exceedance probabilities for this alternative. It also presents the discrete return periods for comparison.

For the Swannanoa River reach, flood damage increases with decrease in the probability of the event. They range between \$72 only and \$21.3 million for exceedance probabilities ranging from 99.9% (peak flow ~331 cfs) to 0.1 % (peak flow ~14,289 cfs). While there is a significant increase in flood damage, it should be noted that there is a higher chance of lower magnitude events occurring more frequently than the higher magnitude events. The discrete points in Figure 4.9 show the standard approach of depicting flood damages in terms of discrete return period compared to the continuous flood damage frequency curve. Here, the discrete return period approach shows the general trend in the increase of flood damages. However, flood damages are quantified

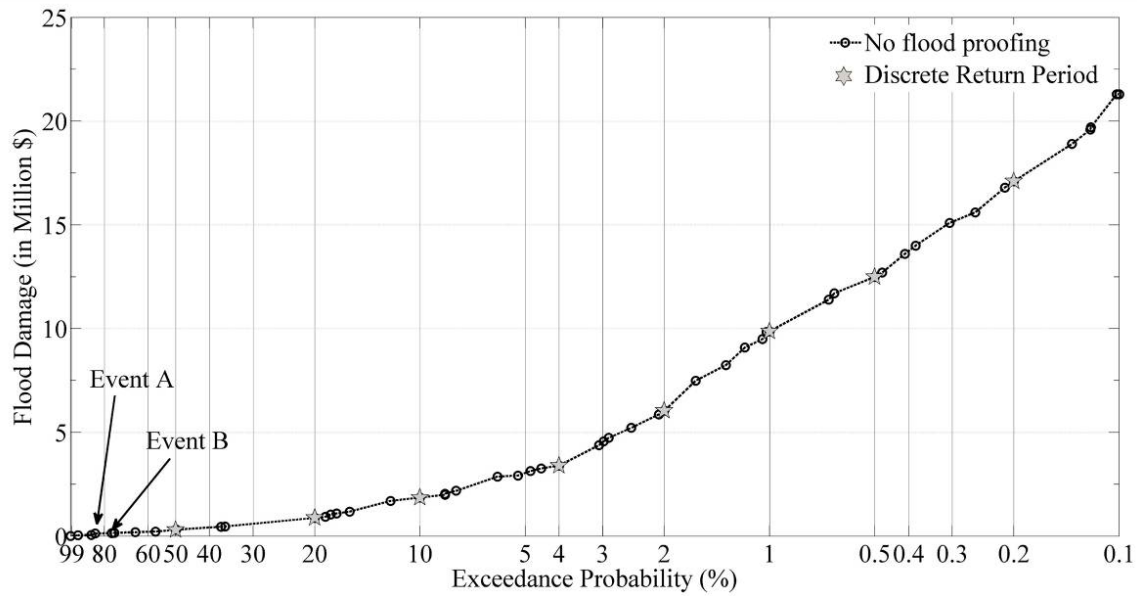


Figure 4.9. Total flood damage frequency curve for No flood proofing alternative

using EAD for both the discrete return period approach and the continuous curve approach. The EAD for the continuous approach is calculated to be \$811,000 while EAD for the discrete approach is calculated as \$619,000, underestimating by 23.6%. This shows that using the continuous curve representing flood damages gives detailed flood damage quantification compared to the standard approach using discrete return periods.

Figure 4.10 presents the annualized risk calculated for the Swannanoa River. An increasing trend is observed for higher probability events and then a decreasing trend for lower probability events. Few outliers are noticed within these general trends of annualized risk which are affected by the magnitude of the probability of event. For example, in the increasing trend region, consider two events A and B from Figure 4.10 corresponding to 84.9% and 76.1% exceedance probabilities, respectively. The flood damages for events A and B are \$ 127,000 and \$ 138,000, respectively (Figure 4.9), as expected. However the annualized risks calculated for these two events are \$108,000/yr

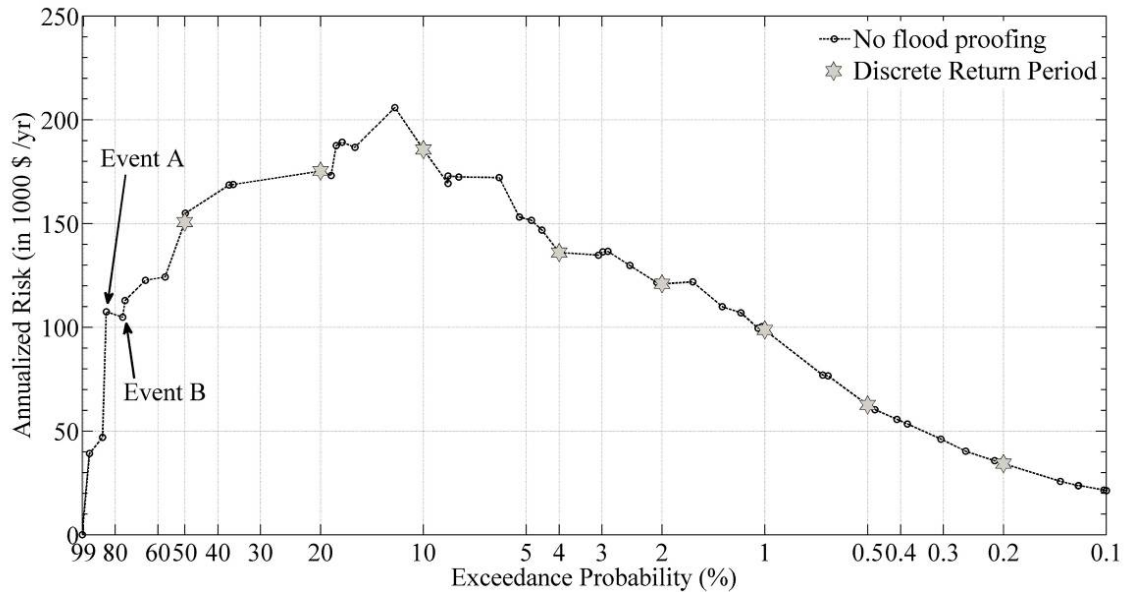


Figure 4.10. Annualized flood risk frequency curve for No flood proofing alternative

and \$105,000/yr, respectively (Figure 4.10). Thus, the increase in flood damage (and flood magnitude) from event A to event B has a lesser effect on the annualized risk than the decrease in the exceedance probability from event A to B. Conversely, few outliers with increasing trend are also noticed in the lower probability region. Discrete return period events are also plotted in Figure 4.10. Even though discrete return period events present the general trend of the variation of annualized risk, they do not capture the maximum annualized risk corresponding to the 12% event. Thus, a continuous distribution of annualized risk would be needed to clearly pinpoint the event with the maximum risk in addition to the underestimation discussed above.

The annualized risk is estimated to be about \$72/year for a 99.99% event. It increases to about \$206,000/year for the 12% event and has a decreasing trend towards lower probability events with about \$ 21,00/year for a 0.099% flood event. It is partly due to the various definitions of flood risk that are used (Simonovic and Ahmad, 2007). It is

also due to the fact that flood risk is inversely proportional to exceedance probability. Counterintuitive to the notion of “higher flood magnitude yields higher risk,” it could also cause situations where lower magnitude events due to their high frequency of occurrence result in significant annualized flood risks. This has financial implications for implementing flood proofing, in terms of the significant difference of capital costs involved between designing for a larger magnitude versus a lower magnitude flood event.

To assess the impact of selecting the design flood event for flood proofing on the reduction of flood damage and annualized risk, two cases are considered based on Figure 4.10. The 1% flood proofing alternative is selected to be the standard 1% flood event with a peak discharge of 9,550 cfs. The 12% flood proofing alternative is selected to be the 12% exceedance probability event corresponding to the highest annualized risk, with a peak discharge of 4,934 cfs.

The 1% flood proofing alternative includes locations within the 1% floodplain that experienced flood damage and is subjected to flood proofing (raising the elevation of the structure, such that its elevation is at least 1ft above the flood depth of the 1% floodplain). The 12% flood proofing alternative includes the locations identified within the 12% floodplain and subjected to flood proofing. The reasoning behind using this scenario is that the annualized risk can be reduced by selecting the level of protection from floods to a higher exceedance probability event with a lower flood magnitude.

4.4.2 Flood Proofing Alternatives

To simulate flood proofing by elevation, all the grid cells within the design floodplain are identified, and their flood depths are adjusted by the design event flood

depth at those grid cells. For the 1% flood proofing alternative, there were 4000 grid cells within the 1% floodplain that experienced flood damage and flood proofing by elevation is performed on them. Similarly, for the 12% flood proofing alternative, there were 1381 grid cells within the 12% floodplain experiencing the flood damage and flood proofing is performed on them. Both of the scenarios are simulated with the flood model using the same 50 flow hydrographs. The simulated flood depths are used to calculate the flood damages and annualized risk. Figure 4.11 plots the distribution of the flood damage for various probabilities for the three cases.

A significant reduction in flood damage is observed for the 1% flood proofing alternative across the range of exceedance probabilities compared to the No flood proofing alternative. Implementing flood proofing based on the 1% design event not only reduced flood damage for events equal to or smaller than the 1% event, but it also reduced the damages for larger events significantly. For example, the flood damage

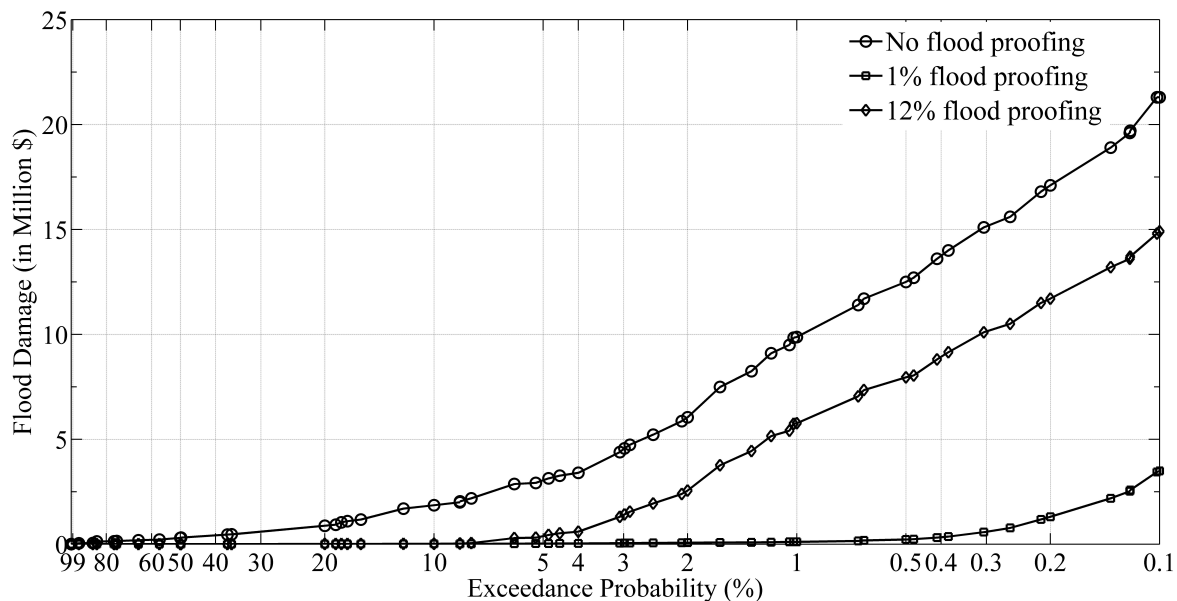


Figure 4.11. Total Flood Damage frequency curve after Flood Proofing

corresponding to the 2% probability event reduced from \$6.02 million to \$ 70,000, from No flood proofing to 1% flood proofing. The maximum total flood damage reduced from \$21.3 million to \$3.48 million for the 0.1% probability event.

Significant flood damage reduction is also observed in the 12% flood proofing alternative, comparable to the 1% flood proofing alternative. For the 0.1% probability event, the maximum flood damage reduced from \$21.3 million to \$14.9 million.

It is seen that there is a significant decrease in the annualized flood risks by incorporating flood proofing versus the No flood proofing alternative (Figure 4.12) for both the design alternatives. The damages are "shifted" or "attenuated" towards the lower probability events. By implementing the 1% flood proofing alternative, it is observed that the annualized flood risk has a continuous increasing trend until 0.102% probability (equivalent to peak flow of 14,235 cfs).

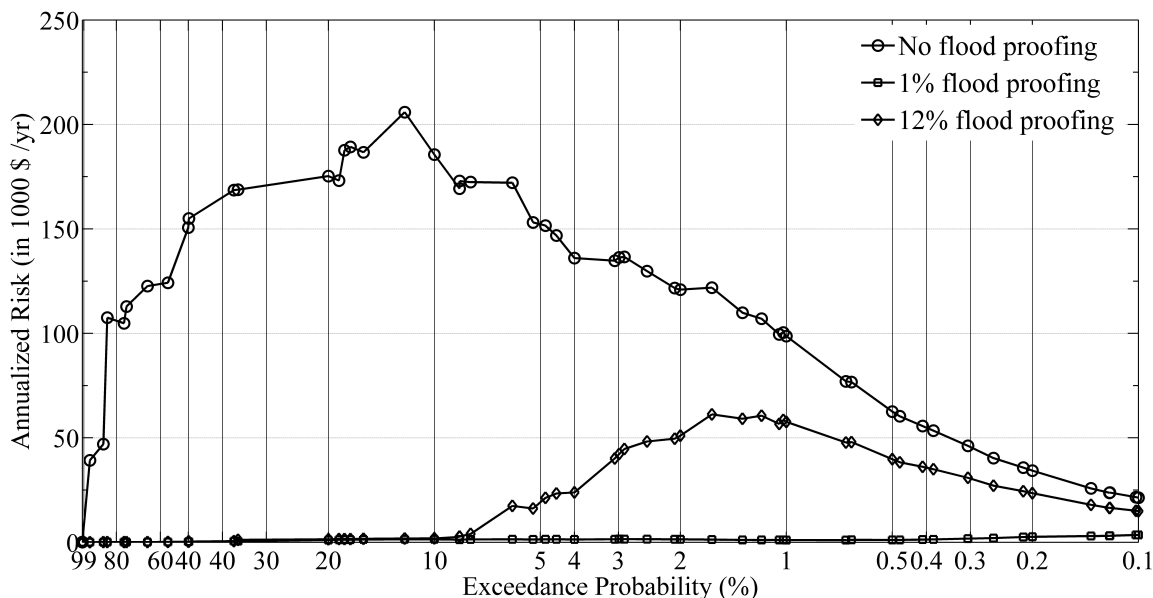


Figure 4.12. Annualized flood risk frequency curve after Flood Proofing

Compared to the No flood proofing alternative, the annualized risk has reduced significantly for the 1% flood proofing alternative from about \$206,000 to \$ 3,500, a decrease of 98.3%. For the 12% flood proofing alternative, the annualized risk starts to increase significantly at about 7.88% (peak flow of 5,684 cfs) and the maximum annualized risk is observed at 1.6% (peak flow of 8,575 cfs) and then follows a decreasing trend for the lower probability events. By implementing the 12% flood proofing alternative, maximum annualized risk reduced from \$206,000/year to \$61,000/year, a decrease of 70.3%.

To quantify flood risk in terms of an annual expected damage, EAD (in \$ per year) for the three cases is also estimated as the area under the flood damage curve (Figure 4.11). This is presented in Table 4.1. The 1% flood proofing alternative significantly reduced the EAD by 95.2% compared to No flood proofing while the 12% flood proofing alternative reduced EAD by 75.3%.

4.4.3 Capital Cost Estimates

The capital cost involved in implementing flood proofing for the two scenarios are estimated. Based on the unit costs cited in the USACE (2004) report on nonstructural

Table 4.1. Estimated annual damages calculated for the three cases

| Alternative | EAD (\$) | Benefits | |
|--------------------|-----------------|----------------------|--------------------|
| | | EAD Reduction | % Reduction |
| No flood proofing | \$811,000 | - | - |
| 1% flood proofing | \$39,000 | \$771,000 | 95.2 |
| 12% flood proofing | \$200,000 | \$610,000 | 75.3 |

flood damage reduction assessment, the cost to elevate 1 sq. ft of area is considered to be \$26.60 including foundation, extending utilities, and miscellaneous items such as sidewalks and driveways. Table 4.2 presents the cost estimates for implementing the 1% flood proofing and 12% flood proofing alternatives. It is observed that the total estimated cost for the 1% flood proofing alternative is about \$607.9 million, which is almost three times the total cost for the 12% flood proofing alternative. To quantify the implementation costs on an annual basis, an expected annual cost (EAC) is estimated, calculated using the following equation (Sullivan et al., 2003):

$$A = P \left[\frac{i(1+i)^N}{(1+i)^N - 1} \right] \quad (4-5)$$

where, A is the EAC or it is a uniform series of annual costs occurring at the end of each year in (\$/year), P is the total cost (or present worth) of installation in (\$), N is the total number of interest periods in years and i is the interest rate in %. The quantity in the brackets is called the capital recovery factor, which brings the present worth of the installation costs to an annual basis. EAC is estimated using an interest rate of 5% and an

Table 4.2. Calculation of cost estimates for implementing flood proofing

| Alternative | Elevation | | Temporary Housing (\$) | Total Estimated Cost (\$) | EAC (\$ per year) |
|--------------------|----------------------|-----------------|------------------------|---------------------------|-------------------|
| | Unit Cost (\$/sq.ft) | Final Cost (\$) | | | |
| 1% flood proofing | \$ 26.60 | \$599,160, | \$8,800,000 | \$607,960,000 | \$33,302,000 |
| 12% flood proofing | \$ 26.60 | \$206,860,000 | \$3,038,200 | \$209,898,000 | \$11,498,000 |

estimated life time period of 50 years.

It is observed that EAC is high for the 1% flood proofing, \$33.3 million per year, and implementing 12% flood proofing reduced the EAC by 65.5%. This shows that even though the 1% flood proofing alternative reduces the maximum annualized risk by 98.3%, it comes at a high price. However, significant reduction in maximum annualized risk, about 70.3% from the No flood proofing alternative, can be obtained by using 12% flood proofing. This shows that implementing 1% flood proofing significantly reduces the flood damage and risk. But, it also shows that even 12% flood proofing significantly increased the benefits from flood proofing by 75% in terms of EAD.

For every dollar spent on implementing flood proofing for the 1% flood proofing, there is a reduction of 2.32 cents in flood damages every year. For the 12% flood proofing, there is a 5.31 cents reduction of flood damages every year, more than twice the reduction from the 1% flood proofing. While implementing flood proofing is a costly affair and depends on the availability of financial resources and many other factors, this study intends to show that designing a flood risk management alternative for a smaller magnitude but more frequent flood event could reduce the financial costs compared to a standard design flood event.

The results from this single case study raise important questions to the decision makers: Would you be willing to take a risk by designing one of many integrated flood risk management solutions at a high frequency (lower magnitude) flood event? Or would you rather stay with the standard design. This has significant implications on the decision makers and floodplain managers to carefully take a second look at the “acceptable” risk when designing flood protection.

4.5 Summary

This paper presents an analysis approach using a Monte Carlo flood risk modeling framework to compare annualized risk reductions from flood control alternatives targeting different average recurrence interval events. The benefits for implementing flood proofing is assessed for the Swannanoa River floodplain located in Buncombe County, North Carolina, in terms of reduction in annualized risk and flood damage. Fifty different flood model simulations were performed by randomly sampling from the continuous flow distribution curve developed for Swannanoa River reach. For every model simulation, flood damage and annualized risk are calculated, resulting in two continuous curves: a flood damage frequency curve and an annualized risk curve. Firstly, this study found that the general approach of using discrete return periods to calculate EAD underestimated by 23.6% compared to EAD calculated using a continuous flood damage frequency curve. Secondly, from the annualized curve, it is found that there is higher annualized risk at a probability of 12% (~4,930 cfs) and then the annualized risk decreases with the probability of event. Along with No flood proofing alternative, two flood proofing alternatives are also compared: 1% flood proofing and 12% flood proofing alternative. There was 95.2% reduction in EAD from No flood proofing alternative to 1% flood proofing alternative. Using the 12% flood proofing alternative, there was 75.3% reduction from No flood proofing alternative. Even though 1% flood proofing reduces the maximum annualized risk by 98.3%, it comes at a high price, while significant reduction in maximum annualized risk, about 70.3%, can be obtained by using the cheaper 12% flood proofing. Annually, for every dollar spent on implementing flood proofing, 12% flood proofing alternative resulted in twice the reduction in flood damages compared to

the 1% flood proofing alternative. Finally, the use of an annualized risk curve approach in selection of flood risk management design alternatives is demonstrated. Thus, this preliminary study shows that significant reduction in flood damage is possible by designing flood proofing, one of the many flood risk management alternatives, for a lower magnitude and more frequent design flood event compared to the standard 1% flood event.

CHAPTER 5

CONCLUSIONS

The objectives of this dissertation research were to improve flood risk management by enhancing model computational capability and incorporating uncertainty in better representing flood risk. This is accomplished by (1) developing a computationally efficient GPU-based 2D flood model by using an efficient and robust upwind numerical scheme to solve the complete 2D Saint Venant equations, (2) developing a Monte Carlo based probabilistic framework to incorporate data and parameter uncertainties and generate probability weighted flood risk, and (3) applying the Monte Carlo based framework to study the benefits of implementing flood proofing.

The first objective resulted in developing a new GPU-based dynamic flood model Flood2D-GPU in the NVIDIA CUDA framework. The model is validated using a lab scale dam experiment and a flood event resulting from the Taum Sauk dam break failure in Missouri. The computational advantage of using GPU versus an equivalent CPU model is presented in three different ways. First, the computational enhancement of using this parallel programming technique is presented with computational speedups ranging between 82x and 88x compared to a CPU model implementing the same numerical algorithms. Second, the computational domain is minimized to include only the flood extent, further reducing the computational intensity. It is observed that while the domain

reduction increased the CPU speedup, it is less compared to the speedup from GPU. The GPU model presents modelers with more flexibility to be less precise with modeling domain extent, thus reducing additional time in preprocessing flood models, which is significant for emergency operations. Third, the effect of spatial resolution on speedups is studied. It is observed that the parallel processing power of GPU is more evident at higher spatial resolution with a larger number of grid cells, which better incorporates complex topography and flow characteristics and is preferred for flood studies. Overall, the Flood2D-GPU flood model provides a useful parallelization approach implementing the full dynamic wave, permitting more accurate and faster flood simulation results to be obtained. The future of GPU implementation in flood modeling has great potential with developments in GPU hardware, software and ever increasing availability expected.

The significant performance boost by Flood2D-GPU is applied in developing a Monte Carlo based probabilistic flood inundation framework. The framework consists of three modules: (i) User-defined random sampling module for flood model parameters and input variables, (ii) Geospatial Output Analysis module, to quantify the inundation probability and flood model statistics, and (iii) Risk Map Development module to develop a probability weighted flood risk map using a Water Depth-Velocity Hazard Classification Diagram. The importance of incorporating uncertainty in flood models is demonstrated by applying this framework to a 1% flood event in Swannanoa River, North Carolina. Flood risk maps were developed using a single simulation (deterministic) and a multiple simulation (probabilistic) approaches. The probabilistic flood risk map was developed from simulating 50 random flow hydrographs in Flood2D-GPU. The deterministic approach underestimated the flood risk by 32.5% relative to the

probabilistic approach. As the number of samples increased, compared to the deterministic approach, probabilistic approach estimated areas with low hazard and high hazard increased by 3.8% and 21.2%, respectively. This difference in flood risk translates into significant underestimation of lives-in-jeopardy in populated areas. This is quantified by calculating the lives-in-jeopardy, where the probabilistic approach simulated 1101 lives are lost while deterministic simulated 925, an underestimation of 16%, which can be very significant if the floodplain is highly populated. Application of this framework on a single case study demonstrates the improvement of the probabilistic flood modeling approach compared to the deterministic approach. Thus, the framework has the ability to provide improved accuracy of flood risk information and in general greater insight into the spatial distribution of flood risk useful in making decisions. This has significant implications in the estimation of flood damage and on the floodplain emergency management.

Finally, Monte Carlo based framework was also applied to assess the benefits of flood risk management. Selecting flood proofing as the flood risk management alternative, this study focused on determining the effect of using different design flood events to implement flood proofing on its associated annualized risk, compared to using the standard 1% flood event. Using the annualized flood risk concept, this study assesses benefits for implementing flood proofing in Swannanoa River floodplain in terms of reduction in annualized risk and flood damage. The probabilistic flood modeling framework is used to simulate floods at various exceedance probabilities. Fifty different flood model simulations were performed by randomly sampling from the continuous flow distribution curve developed for Swannanoa River reach (Base Case). These simulations

are used in calculating flood damage annualized risk curves. Firstly, this study found that the general approach of using discrete return periods to calculate EAD underestimated by 23.6% compared to EAD calculated using a continuous flood damage frequency curve. From the annualized curve, it is found that there is higher annualized risk at an exceedance probability of 12%. It was hypothesized that designing flood proofing using 12% probability event may not only yield significant reduction in annualized risk but also significantly reduces the implementation costs, compared to a 1% flood event. To study this, two cases are tested where flood proofing is simulated by designing at two flood events, 1% flood event (Standard Design) and 12% flood event (Alternate Design), respectively. The results from the Standard Design showed significant reduction in flood damage with the maximum flood damage reducing from \$20.7 million to \$8.11 million. The EAD calculated show 95.2% reduction from implementing the Standard Design. It is also observed that designing flood proofing for the Alternate Design significantly reduced the maximum flood damage from \$20.7 million to \$15.5 million. The EAD reduced by 75.3% when implementing the Alternate Design. It is observed that even though the Standard Design reduces the maximum annualized risk by 98.3%, it comes at a high capital cost of \$33.3 million annually, while reduction in maximum annualized risk, about 70.3%, can be obtained by using the Alternative Design that only costs \$11.5 million annually. Also, this shows the use of the annualized risk curve approach in selecting design alternatives. In terms of implementation costs, every dollar spent on flood proofing annually in the Alternate Design resulted in twice the reduction in flood damages compared to the Standard Design. Thus, this preliminary study shows that significant reduction in flood damage is possible by designing flood proofing for a lower

magnitude and more frequent design flood event compared to the standard 1% flood event.

The outcomes of the research are improved flood modeling and simulation capability by providing increased speed, improved quantification of flood inundation uncertainty, and a newer understanding of flood risk management and decision making.

Future work extending this study should include:

- Developing a second-order accurate upwinding numerical solution on the GPU framework including its validation and verification to flood modeling applications.
- Incorporate physical process components including infiltration, evapotranspiration, and erosion mechanics, especially long-term flood simulations.
- Study the feasibility of the Flood2D-GPU to emergency management operations including flood warning and flood fighting operations.
- Extend the application of the Monte Carlo flood risk modeling framework to various applications including the impacts on climate change and urbanization on increasing flood risk.

APPENDIX A

FIRST-ORDER UPWIND NUMERICAL DISCRETIZATION OF
THE ST.VENANT EQUATIONS

A.1 Numerical Flood Model

The following section gives a description of the 2D model that is developed in this study. The numerical algorithm used is a first-order accurate upwind difference scheme that solves the nonlinear hyperbolic shallow water equations. The model is based on the flood inundation model developed by Judi (2009). These equations are developed from the Navier-Stokes equations by integrating the horizontal momentum and continuity equations over depth. Thus, these equations are often referred to as the depth-averaged or depth-integrated shallow water equations. The form of partial differential equations shown here is the nonconservative form of the equations.

$$\frac{\partial h}{\partial t} + \frac{\partial uh}{\partial x} + \frac{\partial vh}{\partial y} = 0 \quad \text{Continuity Equation (A - 1)}$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + g \frac{\partial H}{\partial x} + gS_{fx} = 0 \quad \text{Momentum Equation in x - direction (A - 2)}$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + g \frac{\partial H}{\partial y} + gS_{fy} = 0 \quad \text{Momentum Equation in y - direction (A - 3)}$$

where, h is the water depth, H is the water surface elevation, u is the velocity in the x-direction, v is the velocity in the y-direction, t is the time, g is the acceleration due to gravity, S_{fx} is the friction slope in the x-direction and S_{fy} is the friction slope in the y-direction.

The friction slope terms are estimated based on the Manning's formula as follows:

$$S_{fx} = \frac{n_x^2 u \sqrt{(u^2 + v^2)}}{h_x^{4/3}} \quad (\text{A - 4}) \text{ and,}$$

$$S_{fy} = \frac{\overline{n}_y v \sqrt{v^2 + u^2}}{h_y^{4/3}} \quad (\text{A - 5})$$

where \overline{n}_x and \overline{n}_y are the average Manning's roughness values at the location of the velocity vector in x and y direction, respectively, and h_x and h_y are the average depths at the location of the velocity vector in x and y direction, respectively.

Boundary conditions are treated by creating cells in the computational domain. Zero gradients (free boundary) are specified by assigning these ghost cells to mirror the values of depth and velocities of the boundary cells.

A.1.1 Spatial Discretization

The first-order upwind finite difference numerical scheme was used for discretizing the governing equations because it yields nonoscillatory solutions through numerical diffusion (Patankar, 1980; Ferziger and Peric, 2002; Judi, 2009). A staggered grid stencil is used to define the computational domain with the water depth (h) in the center of the cell and u and v velocities on the cell edges, as shown in Figure A.1.

Let us look at the continuity equation first to determine the updated h values by rearranging equation A-1 as follows:

$$\frac{\partial h}{\partial t} = - \left[\frac{\partial uh}{\partial x} + \frac{\partial vh}{\partial y} \right] \quad (\text{A - 6})$$

$$h_N = \begin{cases} h_{i,j} & \text{if } (v_{i,j} > 0) \\ h_{i+1,j} & \text{if } (v_{i,j} < 0) \end{cases} \quad (\text{A - 11})$$

$$h_S = \begin{cases} h_{i,j} & \text{if } (v_{i,j-1} > 0) \\ h_{i,j} & \text{if } (v_{i,j-1} < 0) \end{cases} \quad (\text{A - 12})$$

The momentum equation in x-direction is rearranged here for determining the updated u velocities as follows:

$$\frac{\partial u}{\partial t} = - \left[u_{i,j} \frac{\partial u}{\partial x} + \overline{v}_{i,j} \frac{\partial u}{\partial y} + g \frac{\partial H}{\partial x} + g S_{fx_{i,j}} \right] \quad (\text{A - 13})$$

The convective terms are discretized using either a forward or backward finite difference depending on the direction of the local velocity.

$$\frac{\partial u}{\partial x} = \begin{cases} \frac{u_{i,j} - u_{i-1,j}}{\Delta x} & (u_{i,j} > 0) \\ \frac{u_{i+1,j} - u_{i,j}}{\Delta x} & (u_{i,j} < 0) \end{cases} \quad (\text{A - 14})$$

$$\frac{\partial u}{\partial y} = \begin{cases} \frac{u_{i,j} - u_{i,j-1}}{\Delta y} & (\overline{v}_{i,j} > 0) \\ \frac{u_{i,j+1} - u_{i,j}}{\Delta y} & (\overline{v}_{i,j} < 0) \end{cases} \quad (\text{A - 15})$$

where, $\overline{v}_{i,j}$ is the local velocity in the y-direction taken from the average of the surrounding cells.

$$\bar{v}_{i,j} = \frac{v_{i,j} + v_{i+1,j} + v_{i+1,j-1} + v_{i,j-1}}{4} \quad (\text{A-16})$$

The nonconvective term representing the water surface gradient is resolved as a second-order accurate central-difference scheme:

$$\frac{\partial H}{\partial x} = \frac{H_{i+1,j} - H_{i,j}}{\Delta x} \quad (\text{A-17})$$

The friction term is discretized in the x-direction, using Manning's equation as:

$$S_{fx} = \frac{\bar{n}_{x^2,i,j} u_{i,j} \sqrt{(u_{i,j}^2 + v_{i,j}^2)}}{(\bar{h}_x + h_{extra})^{4/3}} \quad (\text{A-18})$$

Similarly, discretization is carried out for the momentum equation in the y-direction.

A.1.1.2 Temporal Discretization

The continuity and momentum equations are discretized explicitly to march forward in time as follows:

$$h_{i,j}^{t+1} = h_{i,j}^t - \Delta t \left[\frac{\partial uh}{\partial x} + \frac{\partial vh}{\partial y} \right]_{i,j}^t \quad (\text{A-19})$$

$$u_{i,j}^{t+1} = u_{i,j}^t - \Delta t \left[u_{i,j} \frac{\partial u}{\partial x} + \bar{v}_{i,j} \frac{\partial u}{\partial y} + g \frac{\partial H}{\partial x} + g S_{fxi,j} \right]_{i,j}^t \quad (A-20)$$

$$v_{i,j}^{t+1} = v_{i,j}^t - \Delta t \left[\bar{u}_{i,j} \frac{\partial v}{\partial x} + v_{i,j} \frac{\partial v}{\partial y} + g \frac{\partial H}{\partial y} + g S_{fyi,j} \right]_{i,j}^t \quad (A-21)$$

The explicit finite difference approach solves unknown values sequentially at a time step from one grid location to the next, and can be unstable for larger increments of Δt for the numerical procedure to converge. The scheme may be stable if the errors generated from discretization are not propagated to the future time steps. A necessary but insufficient condition for stability of the explicit scheme, called Courant condition, is applied for stability and numerical convergence (Liggett and Woolhiser, 1967) as follows:

$$\Delta t = C_n \frac{\Delta n}{u_{\max}} \quad (A-22)$$

where, C_n is the Courant number, u_{\max} is the maximum velocity in both x and y directions and Δn is the smallest spatial dimension for a cell in x or y direction.

The u_{\max} is maximum velocity (both x and y directions) in any cell including the wave celerity 'c', $u_{\max} = \max(u) + c$. The Courant condition needs the time step to be less than the time for a wave (diffusive) to travel the distance Δn . If Δt is large enough that the condition is not satisfied, then there is in effect, an accumulation or piling up of water, causing instability. In our study, Δt is determined at each time step that meets the Courant condition computed at all grid cells and the smallest value (which means highest

velocity) is taken and reduced by 90 % (because, $C_n = 0.1$ is adapted). It should be noted that Courant condition does not confirm stability but is just a guideline (Chow et al., 1988).

APPENDIX B

IMPLEMENTATION STEPS FOR CPU AND GPU FLOOD MODEL FRAMEWORKS

B. 1 Implementation of CPU and GPU Flood Models

B.1.1 CPU Computation

1. The input DEM and a flow hydrograph are provided. The DEM array is padded with an extra layer of cells on all four sides.
2. Based on the DEM size, the h, u and v arrays are allocated and initialized. As with the DEM array, these three arrays are also padded accordingly.
3. Based on the source coordinates (user input), the corresponding source row and column values are calculated.
4. The program iterates over the interior cells to find the updated h, u and v values. These are calculated, based on solving the continuity and momentum equations. The maximum u, v and the corresponding h values are updated for every iteration.
5. The extra cells are populated with the updated h, u and v values.
6. Using the Courant condition, the new Δt value is obtained.
7. The source location is updated with the new h, u and v values, using the hydrograph.
8. Steps 4 to 6 are iterated through a certain number of times, till the desired simulation time is achieved.

B.1.2 GPU Computation

The GPU model has been implemented in CUDA. In this model, steps 1 to 3 remain the same, and are executed on the CPU. Step 4 is executed as a kernel (function

executed on the GPU). The computational domain is divided into a set of blocks, and the blocks are scheduled onto the Symmetric Multi Processing (SMP) cores. Only the inner cells are considered when generating the blocks (computation done only for interior cells, extra cell values are copied from the interior cells). Corresponding to every block in the domain, a 2D shared memory array is allocated to store cell values. The array dimensions are two more than the block size in both directions, to account for extra cells required during computation. The interior cells in the array are populated using corresponding threads, and the extra cells are populated using a method by Micikevicius (2009). This method does not populate the four extra cells in the corners. This issue is addressed by directly reading from the global memory, thereby reducing any further divergence. After computing the updated h , u and v values, they are populated back into the shared memory, and one thread writes into the global memory. Thread synchronization calls ensure that all threads write to the shared memory first, before the global memory is updated.

Step 5 involves updating the extra cells. To ensure that threads do not go idle by creating extra blocks, two kernel calls are made, one for updating the extra rows, and the other for updating the extra columns. The block size for the kernel calls is equal to the extra row and extra column size, respectively. This is illustrated in Figure B.1. To find the maximum u , v and the corresponding h values, a kernel has been implemented which performs parallel reduction. Similar to Step 4, the computational domain (corresponding to the interior cells) is divided into blocks. The data corresponding to a block is copied to a shared memory array. Within a block, a method of comparison by Harris (2007) is used to find the column maximum. The method ensures less divergence, and a faster execution

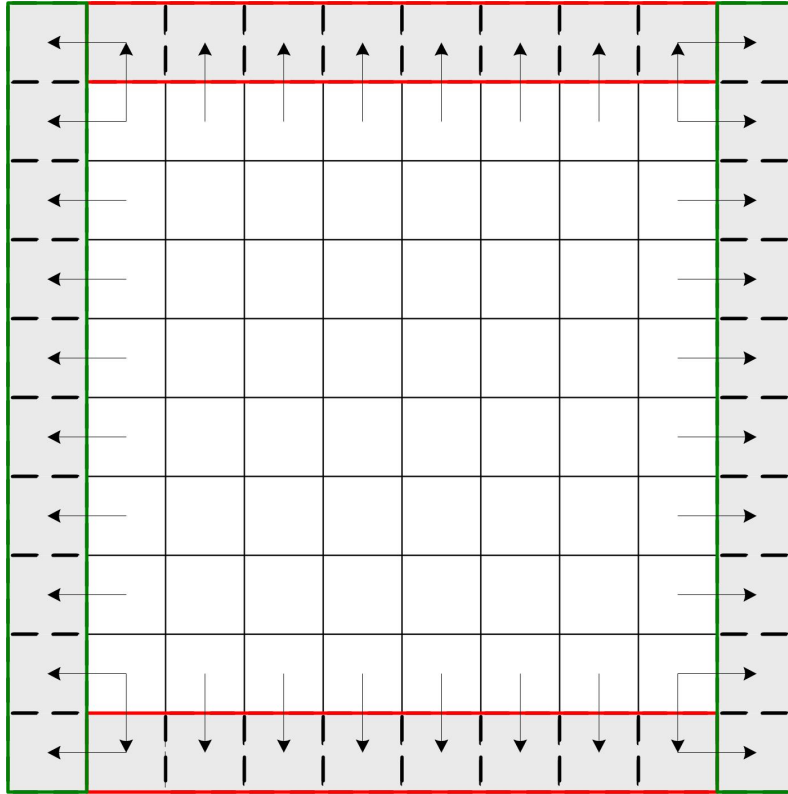


Figure B.1: A layer of extra rows and columns, scheduled as a single block on to the CUDA cores.

time. The column maximums are written at index 0 of their corresponding rows, and index 0 locations of the corresponding rows are then compared to find the maximum value in the block.

The maximum value in a block is stored at the array location, corresponding to the thread id 0. The 0 thread then writes back to the global memory, at the location corresponding to the block id. This ensures all maximum values are written into a subarray, within the larger global memory block. This is illustrated in Figure B.2.

Once the values are written, the procedure of reduction is repeated, until it is reduced to an array of size 1×1 , which happens to the maximum values in the computational domain. The reason behind writing back a subarray into the global

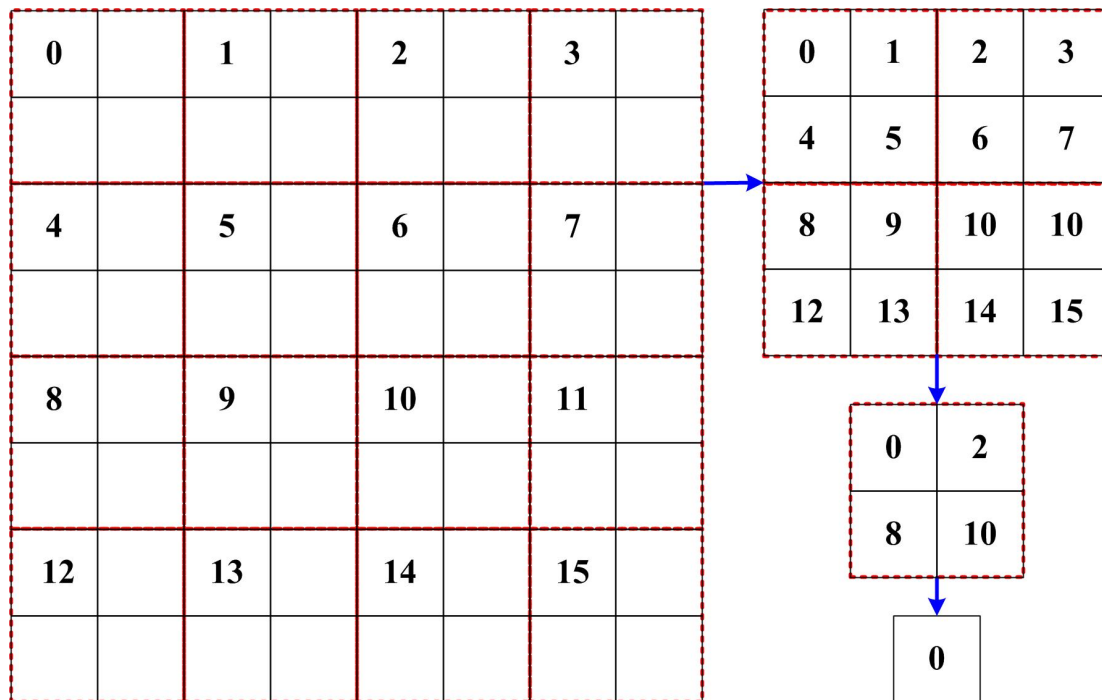


Figure B.2: An example of the recursive parallel reduction technique, with a block size of 2x2.

memory block is that it reduces the size of the computational domain (for the next iteration), and also the number of thread/blocks. The amount of time to perform comparison, and to find out the maximum value, reduces significantly with each successive call. Steps 6, 7 and 8 remain the same, as in the CPU implementation.

B.1.2 GPU Enhancements

Within the GPU model, a *16 cell padding* scheme has been implemented. The computational domain is padded with a layer of 16 extra cells, which ensures memory alignment, and coalesced memory calls by threads in a half warp (32 threads make a warp; 16 are executed as a unit). This is of significance especially on devices with a lower compute capability (NVIDIA, 2009). The other enhancement comes into play

during the execution of the parallel reduction method. In order to ensure that the block size does not exceed the new subarray size, the block size is altered dynamically. The following two equations involve calculating the new subarray and the block size,

$$arrW = \frac{(arrW + blkW - 1)}{blkW} \quad (B-1)$$

The second equation is solved, only if the subarray width is lesser than the block width,

$$blkW = 2^{\log_2(arrW \% blkW)} \quad (B-2)$$

where, $arrW$ and $blkW$ are the subarray and the block size, at any given iteration and ‘%’ sign indicates the remainder of the fraction. The same set of equations is used for calculating the new height.

REFERENCES

- ACER Technical Memorandum No. 11., 1988. Assistant Commissioner—Engineering and Research, Downstream hazard classification guidelines, Denver, Colorado. U.S. Department of the Interior, Bureau of Reclamation.
- Ahmad, S., Simonovic, S. P., 2006. An intelligent decision support system for management of floods. *Water Resources Management*, 20, 391-410.
- Anderson, J., Jones, B., Yang, J. –H., Shaw, M., Watt, C., Koshevoy, P., Spaltenstein, J., Jurrus, E., Kannan, U., Whitekar, R., Mastronarde, D., Tasdizen, T., Marc, R., 2009. Ultrastructural mapping of neural circuitry: a computational framework. *PLoS Biology*, 7(3), 1135-1137.
- Apel, h., Thielen, A. H., Merz, B., Bloschl, G., 2006. A probabilistic modelling system for assessing flood risks. *Natural Hazards*, 38, 79-100.
- Aronica, G., Hanking, B., Beven, K., 1998. Uncertainty and equifinality in calibrating distributed roughness coefficients in a flood propagation model with limited data. *Advances in Water Resources*, 22(4), 349-365.
- Aroninca, G., Bates, P. D., Horritt, M S., 2002. Assessing the uncertainty in distributed model predictions using observed binary pattern information within GLUE. *Hydrological Processes*, 16, 2001-2016.
- Bales, J. D., Wagner, C. R., 2009. Sources of uncertainty in flood inundation maps. *Journal of Flood Risk Management*, 2(2), 139-147.
- Barredo, J. I., Engelen, G., 2010. Land use scenario modeling for flood risk mitigation. *Sustainability*, 2, 1327-1344.
- Bates, P. D., Anderson, M. G., Baird, L., Walling, D. E., Simm, D., 1992. Modeling flood plain flows using a two dimensional finite element model. *Earth Surface Processes and Landforms*, 17, 575-588.
- Bates, P. D., Anderson, M. G., Hervouet, J. -M., 1995. Initial comparison of two two-dimensional finite element codes for river flood simulations. *Proceedings of the Institution of Civil Engineers, Water Maritime and Energy*, 112-238-248.

- Bates, P.D., Stewart, M.D., Siggers, G.B., Smith, C.N., Hervouet, J.M., Sellin, R.H.J., 1998. Internal and external validation of a two-dimensional finite element code for river flood simulations. *Proceedings of the Institution of Civil Engineers, Water Maritime and Energy*, September, 127-141.
- Bates, P. D., De Roo, A. P. J., 2000. A simple raster-based model for flood inundation simulation. *Journal of Hydrology*, 236(1-2), 54-77.
- Bates, P. D., Horritt, M. S., Aronica, G., Beven, K., 2004. Bayesian updating of flood inundation likelihoods conditioned on flood extent data. *Hydrological Processes*, 18, 3347-3370.
- Bates, P. D., Dawson, R. J., Hall, J. W., Horritt, M. S., Nicholls, R. J., Wicks, Hassan, M. A. A. M., 2005. Simplified two-dimensional numerical modeling of coastal flooding and example applications. *Coastal Engineering*, 52, 793-810.
- Beven, K. J., Binley, A., 1992. The future of distributed models – model calibration and uncertainty prediction. *Hydrological Processes*, 6(3), 279-298.
- Blazkova, S., Beven, K., 2002. Flood frequency estimation by continuous simulation of a catchment treated as ungaged (with uncertainty). *Water Resources Research*, 38(8), 1139, 10.1029/2001WR000500.
- Bonnin, G., Todd, D., Lin, B., Parzybok, T., Yekta, M., Riley, D., 2004. NOAA Atlas 14, Precipitation frequency atlas of the United States, Volume 2. U.S. Department of Commerce, National Oceanic and Atmospheric Administration, National Weather Service, Silver Spring, Maryland.
- Bowker, P., 2007. Flood resistance and resilience solutions: an R&D scoping study. Technical Report, Department for Environment, Food and Rural Affairs, London.
- Buchele, B., Kreibich, H., Kron, A., Thiken, A., Ihringer, J., Oberle, P., Merz, B., Nestmann, F., 2006. Flood-risk mapping: contributions towards an enhanced assessment of extreme events and associated risks. *Natural Hazards and Earth System Sciences*, 6, 485-503.
- Buijs, F. A., van Gelder, J. K., Vrijling, J. K., Vrouwenvelder, A. C. W. M., Hall, J. W., Sayers, P. B., Wehrung, M. J., 2003. Application of Dutch reliability-based flood defence design in the UK. In: Bedford, T., and van Gelder, P.H.A.J.M (Eds.), *Safety and Reliability, Proceedings of ESREL 2003*. Taylor and Francis, pp. 311-320.
- Buncombe County, 2011. Geographic Information Systems – Download Digital Data. Available at: <
<http://www.buncombecounty.org/governing/depts/gis/dataDownload.asp>>.

- Burby, R.J., 2001. Flood insurance and floodplain management: the US experience. *Environmental Hazards*, 3, 111-122.
- Carter, N. T., 2009. Federal flood policy challenges: lessons from the 2008 Midwest flood. Congressional Research Service, 7-5700. Washington, DC, Available at: <<http://apps.crs.gov/products/r/pdf/R40201.pdf>>.
- Crawford, N. H., Linsley, Jr., R. K., 1966. Digital simulation in hydrology: Stanford watershed model IV. Technical Report No. 39, Department of Civil Engineering, Stanford University, Stanford, California, USA.
- Di Baldassarri, G., Schumann, G., Bates, P. D., Freer, J. E., Beven, K. J., 2010. Floodplain mapping: a critical discussion of deterministic and probabilistic approaches. *Hydrological Sciences Journal*, 55(3), 364-376.
- Duan, Q., Sorooshian, S., Gupta, V. K., 1992. Effective and efficient global optimization for conceptual rainfall-runoff models. *Water Resources Research*, 28, 1015-1031.
- European Parliament, 2007. Directive 2007/60/Ec of the European Parliament and of the council of 23 October 2007 on the assessment and management of flood risks. Available at: <http://ec.europa.eu/environment/water/flood_risk/key_docs.htm#Directive>.
- Federal Emergency Management Agency (FEMA), 2001. Engineering Principles and Practices for Retrofitting Floodprone Residential Buildings, second ed., Mitigation Directorate, Washington, DC. Available at: <<http://www.fema.gov/library/viewRecord.do?id=1645>>.
- Federal Emergency Management Agency (FEMA), 2002. National Flood Insurance Program: Program Description. Federal Emergency Management Agency/Federal Insurance Mitigation Administration, Available at: <<http://www.fema.gov/library/viewRecord.do?id=1480>>.
- Ferziger, J.H., Peric, M., 2002. Computational Methods for Fluid Dynamics, third ed., Springer, New York.
- Fox, J., Dobson, G., Phillips, M., Pierce, T., 2008. Visualization as a decision support tool - Asheville's North Fork Reservoir (Buncombe County, North Carolina) and the balance between drought and flood mitigation. 2008 Association of State Dam Safety Officials (ASDSO) Southeast Regional Conference, Asheville, North Carolina. 13-16, April 2008.
- Fread, D.L., 1985. Channel routing. In: Anderson, M. G., Burt, T. P. (Eds.), *Hydrological Forecasting*. Wiley, Chichester, pp. 437-503.

- Freer, J., Beven, K.J., Ambrose, B., 1996. Bayesian estimation of uncertainty in runoff prediction and the value of data: an application of the GLUE approach. *Water Resources Research*, 32 (7), 2161–2173.
- Frey, A. E., Olivera, F., Irish, J. L., Dunkin, L. M., Kaihatu, J. M., Ferreira, Edge, B. L., 2010. Potential impact of climate change on hurricane flooding inundation, population affected and property damages in Corpus Christi. *Journal of the American Water Resources Association*, 46(5), 1049-1059.
- García-Navarro, P., Brufau, P., Burguete, J., Murillo, J., 2008. The shallow water equations: an example of hyperbolic system. *Monografías de la Real Academia de Ciencias de Zaragoza*, 31, pp. 89-119.
- Garland, M., Grand, S. L., Anderson, J., Hardwick, J., Morton, S., Phillips, E., Zhang, U., Volkov, V., 2008. Parallel computing experiences with CUDA. *IEEE Micro*, 28(4), 13-27.
- Gommes, R., du Guerny, J., Nachtergaele, F., Brinkman, R., 1997. Potential impacts of sea-level rise on populations and agriculture. Food and Agricultural Organization of the United Nations, Rome. Available at: <http://www.fao.org/sd/eidirect/EIre0045.htm>.
- GPGPU, 2010. General-Purpose Computation on Graphics Hardware. Available at: <http://gpgpu.org/>.
- Hagen, T. R., Hjelmervik, J. M., Lie, K. –A., Natvig, J. R., Henriksen, M. O., 2005. Visual simulation of shallow-water waves. *Simulation Modelling and Practice and Theory*, 13, 716-726.
- Hall, J. W., Meadowcroft, I. C., Sayers, P. B., Bramley, M. E., 2003. Integrated flood risk management in England and Wales. *Natural Hazards Review*, 4(3), 126-135.
- Hall, J. W., Tarantolo, S., Bates, P. D., Horritt, M., 2005. Distributed sensitivity analysis of flood inundation model calibration. *ASCE Journal of Hydraulic Engineering*, 131(2), 117-126.
- Hall, J., Solomatine, D., 2008. A framework for uncertainty analysis in flood risk management decisions. *International Journal of River Basin Management*, 6(2), 85-98.
- Harris, M. J., Coombe, G., Scheuermann, T., Lastra, A., 2002. Physically-based visual simulation on graphics hardware. *Proceedings of the SIGGRAPH/Eurographics Workshop on Graphics Hardware*, 109-118.

- Harris, M., 2007. Optimizing parallel reduction in CUDA. Available at: http://developer.download.nvidia.com/compute/cuda/1_1/Website/projects/reduction/doc/reduction.pdf.
- Helm, P., 1996. Integrated risk management for natural and technological disasters. *Tephra*, 15, 4-13.
- Hluchy, L., Froehlich, D., Tran, V. D., Astalos, J., Dobrucky, M., Nguyen, G. T., 2002. Parallel numerical solution for flood modeling systems. In: Sloot, P. M. A., Hoekstra, A. G., Tan, C. J. K., and Dongarra, J. J. (Eds.) *Computational Science – ICCS 2002: International Conference, Krakow, Poland, Proceedings, Part I*, Lecture Notes in Computer Science, vol. 2329, Springer-Verlag, Berlin Heidelberg New York, pp. 543-551.
- Hunter, N. M., Bates, P. D., Horritt, M. S., Wilson, M. D., 2007. Simple spatially-distributed models for predicted flood inundation: a review. *Geomorphology*, 90, 208-225.
- Interagency Advisory Committee on Water Data. (IACWD). 1982. Guidelines for Determining Flood Flow Frequency: Bulletin 17B of the Hydrology Subcommittee. Office of Water Data Coordination, U. S. Geological Survey, Reston, VA., 183 p.
- Interagency Floodplain Management Review Committee. (IFMRC), 1994. Sharing the Challenge: Floodplain Management into the 21st century: Report of the Interagency Floodplain Management Review Committee. Administration Floodplain Management Task Force, Washington DC.
- Johnson, G. P., Holmes, Jr., R. R., Waite, L. A., 2004. The Great flood of 1993 on the Upper Mississippi River – 10 years later. USGS Fact Sheet 2004-3024. U.S. Geological Survey.
- Judi, D. R., 2009. Advances to fast-response two-dimensional flood modeling. PhD Dissertation, University of Utah, Salt Lake City, UT, USA.
- Judi, D. R., Burian, S. J., McPherson, T. N., 2010. Two-dimensional fast-response flood modeling: desktop parallel computing and domain tracking. *Journal of Computing in Civil Engineering*, 25(3), 184-191.
- Kalyanapu, A. J., Burian, S. J., McPherson, T. N., 2009. Effect of land use-based surface roughness on hydrologic model output. *Journal of Spatial Hydrology*, 9(2), 51-71.
- Kalyanapu, A. J., Shankar, S., Pardyjak, E. R., Judi, D. R., Burian, S. J., 2011. Assessment of GPU computational enhancement to a 2D flood model. *Environmental Modelling & Software*, 26(8), 1009-1016.

- Knebl, M R., Yang, Z. –L., Hutchison, K., Maidment, D. R., 2005. Regional scale flood modeling using NEXRAD rainfall, GIS, and HEC-HMS/RAS: a case study for the San Antonio River Basin summer 2002 storm event. *Journal of Environmental Management*, 75(4), 325 – 336.
- Knight, D. W., Shiono, K., 1996. River channel and floodplain hydraulics. In: Anderson, M. G., Walling, D. E., Bates, P. D. (Eds.). *Floodplain Processes*, Wiley, Chichester, 139-182.
- Kruger, J., Westermann, R., 2003. Linear algebra operators for GPU implementation of numerical algorithms. *ACM Transactions on Graphics*, 22(3), 908-916.
- Lamb, R., Crossley, A., Waller, S., 2009. A fast two-dimensional flood inundation model. *Proceedings of the Institute of Civil Engineers – Water Management*, 162(6), 363-370.
- Leopardi, A., Oliveri, E., Greco, M., 2002. Two-dimensional modeling of floods to map risk-prone Areas. *Journal of Water Resources Planning and Management*, 128(3), 168-178.
- Levy, J. K., Gopalakrishnan, C., Lin, Z., 2005. Advances in decision support systems for flood disaster management: challenges and opportunities. *Water Resources Development*, 21(4), 593-612.
- Levy, J. K., Hartmann, J., Li, K. W., An, Y., Asgary, A., 2007. Multi-criteria decision support systems for flood hazard mitigation and emergency response in urban watersheds. *Journal of American Water Resources Association*, 43(2), 346-358.
- Marks, K., Bates, P., 2000. Integration of high-resolution topographic data with floodplain flow models. *Hydrological Processes*, 14, 2109-2122.
- McCarthy, J. J., Canziani, O. F., Leary, N. A., Dokken, D. J., White, K. S., 2001. *Climate Change: Impacts, Adaptation and Vulnerability*, Cambridge University Press, Cambridge.
- McCuen, R. H., 1998. *Hydrologic Analysis and Design*, Prentice-Hall, New Jersey.
- Menon, S., 2008. Scientific computing on streaming processors. Masters Thesis, University of Massachusetts, Amherst, MA, USA.
- Merwade, V., Olivera, F., Arabi, M., Edleman, S., 2008. Uncertainty in flood inundation mapping: current issues and future directions. *Journal of Hydrologic Engineering*, 13(7), 608-620.
- Micikevicius, P., 2009. 3D finite difference computation on GPUs using CUDA. *ACM International Conference Proceedings Series, Proceedings of Second Workshop*

- on General Purpose Processing on Graphics Processing Units, GPGPU-2, 383, 79-84.
- Middelmann-Fernandes, M. H., 2010. Flood damage estimation beyond stage-damage functions: an Australian example. *Journal of Flood Risk Management*, 3, 88-96.
- Montz, B. E., Grunfest, E., 2002. Flash flood mitigation: recommendations for research and applications, *Environmental Hazards*, 4, 15-22.
- Morss, R. E., Wilhelmi, O. V., Downton, M. W., Grunfest, E., 2005. Flood risk, uncertainty, and scientific information for decision making: lessons from an interdisciplinary project. *Bulletin of the American Meteorological Society*, 86(11), 1593-1601.
- National Nonstructural Flood Proofing Committee. (NFPC), 2011. US Army Corps of Engineers, Omaha District. Available at: <http://www.nwo.usace.army.mil/nfpc/>.
- National Research Council (NRC), (2000). Risk analysis and uncertainty in flood damage reduction studies. Committee on risk-based analysis for flood damage reduction, Water Science and Technology Board, National Academy Press, 2101 Constitution Avenue, NW Washington, DC.
- National Research Council (NRC), 2009. Mapping the Zone: Improving Flood Map Accuracy. Committee on FEMA Flood Maps, National Research Council, National Academies Press, Washington D. C.
- Neal, J. C., Fewtrell, T. J., Trigg, M. A., 2009. Parallelisation of storage cell flood models using OpenMP. *Environmental Modelling & Software*, 24, 872-877.
- Nguyen, H., 2007. GPU Gems 3. Addison-Wesley, Upper Saddle River, NJ, USA.
- North Carolina Floodplain Management Program. NCFMP, 2011. NCFMP Program Review, Appendix B. Available at: http://www.ncfloodmaps.com/program_review.htm.
- NVIDIA, 2009. CUDA Programming Guide 2.2.1. Available at: http://developer.download.nvidia.com/compute/cuda/2_21/toolkit/docs/NVIDIA_CUDA_Programming_Guide_2.2.1.pdf
- Owens, J. D., Luebke, D., Govindaraju, N., Harris, M., Kruger, J., Lefohn, A. E., Purcell, T. J., 2005. A survey of general-purpose computation on graphics hardware. *Eurographics 2005, State of the Art Reports*, 21-51.

- Pappenberger, F., Beven, K., Horritt, M., Blazkova, S., 2005. Uncertainty in the calibration of effective roughness parameters in HEC-RAS using inundation and downstream level observations. 302, 46-69.
- Pappenberger, F., Matgen, P., Beven, K. J., Henver, J. B, Pfister, L., de Fraipont, P., 2006. Influence of uncertainty boundary conditions and model structure on flood inundation predictions. *Advances in Water Resources*, 29, 1430-1449.
- Patankar, S.V., 1980. *Numerical Heat Transfer and Fluid Flow*. Taylor and Francis.
- Pau, J. C., Sanders, B. F., 2006. Performance of parallel implementations of an explicit finite volume shallow-water model. *Journal of Computing in Civil Engineering*, 20(2), 99 – 110.
- Pielke, Jr., R.A., Downton, M.W., Miller, J. Z. B., 2002. *Flood damage in the United States, 1926–2000: a reanalysis of National Weather Service estimates*. Boulder, CO: UCAR.
- Purvis, M. J., Bates, P. D., Hayes, C. M., 2008. A probabilistic methodology to estimate future coastal flood risk due to sea level rise. *Coastal Engineering*, 55, 1062-1073.
- Qi, H., Altinakar, M. S., 2011. A GIS-based decision support system for integrated flood management under uncertainty with two dimensional numerical simulations. *Environmental Modelling & Software*, 26, 817-821.
- Rao, P., 2005. A parallel RMA2 model for simulating large-scale free surface flows. *Environmental Modelling & Software*, 20(1), 53-57.
- Richardson, A., 2009. *GPU acceleration of HPC applications*. MSc Thesis, University of Edinburgh, Edinburgh, UK.
- Romanowicz, R., Beven, K., 2003. Estimation of flood inundation probabilities as conditioned on event inundation maps. *Water Resources Research*, 39(3), 1073-1085.
- Samuels, P. G., 1990. Cross section location in one-dimensional models. In: White, W. R. (Ed.). *International Conference on River Flood Hydraulics*, Wiley, Chichester, 339-350.
- Sayers, P. B., Hall, J. W., Meadowcroft, I. C., 2000. Towards risk-based flood hazard management in the UK, *Floods – a new approach*. Special Issue One, *Proceedings of the Institute of Civil Engineers*, 150, 36-42.
- Schanze, J, 2006. Flood risk management - a basic framework. In: Schanze, J.; Zeman, E.; Marsalek, J. (Eds.), 2006: *Flood Risk Management - Hazards, Vulnerability and Mitigation Measures*, 149-167. Springer.

- Simonovic, S. P., Ahmad, S. S., 2007. A new method for spatial analysis of risk in water resources engineering management. *The Open Civil Engineering Journal*, 1, 1-12.
- Small, C., Cohen, J., 2004. Continental physiography, climate, and the global distribution of human population. *Current Anthropology*, 54, pp. 269.
- Smemoe, C. M., Nelson, E. J., Zundel, A. K., Miller, W., 2007. Demonstrating floodplain uncertainty using flood probability maps. *Journal of the American Water Resources Association*, 43(2), 359-371.
- Smith, K., Ward, R., 1998. *Floods: Physical Processes and Human Impacts*. John Wiley and Sons, New York.
- Smith, P. J., 2007. Fall 2007 Class Notes - Validation Metrics. Computational fluid dynamics, CHEN 5353/6353, University of Utah, Salt Lake City, UT.
- Sullivan, W. G., Wicks, E. M., Luxhoj, J T., 2003. *Engineering Economics*, twelfth ed., Prentice Hall, Upper Saddle River, New Jersey.
- Tayfur, G., Kavvas, M. L., Govindaraju, R. S., Storm, D. E., 1993. Applicability of St. Venant equations for two-dimensional overland flows over rough infiltrating surfaces. *Journal of Hydraulic Engineering*, 119(1), 51-63.
- Tingsanchali, T., Rattanapitikon, W., 1999. 2-D Modelling of dambreak wave propagation on initially dry bed. *International Journal of Science and Technology*, 4(3), 28-37.
- Tran, V.D., Hluchy, L., 2004. Parallelizing flood models with MPI: approaches and experiences. In: Bubak, M., van Al-bada, G.D., Sloot, P.M.A., Dongarra, J.J. (Eds.), *Computational Science - ICCS 2004: International Conference, Krakow, Poland, Proceedings, Part I, Lecture Notes in Computer Science*, vol. 3036, 425-428, SpringerVerlag, Berlin, Heidelberg, New York.
- United Nations Development Programme, (UNDP), 2004. *A global report: reducing disaster risk- A challenge for development*. John S. Swift Co., United States of America.
- USACE, 1988. *National Economic Development Procedures Manual – urban flood damage*. Water Resources Support Center, Institute of Water Resources, IWR Report 88-R-2, Fort Belvoir, Virginia.
- USACE, 1995. *Engineering and design – hydrologic engineering requirements for flood damage reduction studies*. EM 1110-2-1419.

- USACE, 2000a. Generic depth-damage relationships for residential structures without basements. Economic Guidance Memorandum #01-03.
- USACE, 2000b. Planning manual, planning guidance notebook. Engineer Regulation (ER) 1105-2-100.
- Villanueva, I., Wright, N.G., 2006. Linking Riemann and storage cell models for flood prediction. *Proceedings of the Institution of Civil Engineers: Water Management* 159 (1), 27-33.
- Wagener, T., Gupta, H. V., 2005. Model identification for hydrological forecasting under uncertainty. *Stochastic Environmental Research and Risk Assessment*, 19(6), 378-387.
- Weaver, J. C., Feaster, T. G., Gotvald, A. J., 2009. Magnitude and frequency of rural floods in the Southeastern United States, through 2006 – Volume 2, North Carolina. U. S. Geological Survey Scientific Investigations Report, 2009-5158.
- Werner, M. G. F., Hunter, N. M., Bates, P. D., 2005. Identifiability of distributed floodplain roughness values in flood extent estimation. *Journal of Hydrology*, 314, 139-157.
- Wright, J. M., 2000. The nation's responses to flood disasters: a historical account. Association of State Floodplain Managers. Available at: http://www.floods.org/PDF/hist_fpm.pdf.
- Wurbs, R.A., 1996. Optimal sizing of flood damage reduction measures based on economic efficiency. *International Journal of Water Resources Development*, 12(1): 5–16.
- Yang, J., Reichert, P., Abbaspour, K. C., Xia, J., Yang, H., 2008. Comparing uncertainty analysis techniques for a SWAT application to the Chaohe Basin in China. *Journal of Hydrology*, 358, 1-23.
- Yu, D., 2010. Parallelization of a two-dimensional flood inundation model based on domain decomposition. *Environmental Modelling & Software* 25, 935-945.
- Zhang, W., Cundy, T. W., 1989. Modeling of two-dimensional overland flow. *Water Resources Research*, 25(9), 2019-2035.