ABSTRACT

RUCKER, CARTER A. Improving the Accuracy of a Real-Time ADCIRC Storm Surge Downscaling Model. (Under the direction of Joel Dietrich.)

During major storm events such as hurricanes, emergency managers rely on fast and accurate forecasting models to make important decisions concerning public safety. These models can be computationally costly and cannot quickly make predictions at the highest geospatial resolution. However, model output can be post-processed to mimic high-resolution results with minimal additional computational cost. This research proposes methods for improvement in the accuracy of downscaling (enhancing the resolution of) a real-time storm surge forecasting model. Such improvements to downscaling methods include 1) expansion in its spatial applicability, 2) adding physics using water surface slopes, and 3) adding physics using friction losses across the ground surface.

This research builds upon a process that uses maximum water elevation output from the Advanced Circulation (ADCIRC) model and downscales these results to a finer resolution by extrapolating the water levels to small-scale topography. This downscaling process is referred to as the static method. The method was originally designed for use in North Carolina (NC), where results from an ADCIRC model designed specifically for NC were downscaled to a set of NC topographical data. By joining the static method with an ADCIRC output visualization tool, the downscaling process is now able to run faster with the same level of accuracy and can run on any ADCIRC model with downscaling data from any geographical region or given resolution. This process is used to provide extra guidance to emergency managers and decision makers during hurricanes.

The downscaling process is also improved by adding physics using the slopes method and the head loss method. The slopes method incorporates the slopes of the water levels produced by ADCIRC, rather than only the value of the water level. By interpolating ADCIRC output water elevation points into a smooth surface, slopes of this surface can be used to influence the elevations of downscaled water levels. The head loss method adds friction loss due to variations in the ground surface based on land cover types and friction associated with each type. As water travels over any surface, head loss, or a loss in energy, occurs at different rates depending on the surface roughness. This rudimentary hydrologic principle is applied to increase the accuracy of the downscaling process at minimal cost. The downscaling methods are applied for results from an ADCIRC simulation used in real-time forecasting, and then compared with results from an ADCIRC simulation with 10 times more resolution in Carteret County, NC. The static method tends to over-estimate the flood

extents, and the slopes method is similar. However, the head loss method generates a downscaled flooding extent that is a close match to the predictions from the higher-resolution, full-physics model.

By improving the accuracy of downscaling methods at minimal computational cost and expanding the applicability of these downscaling methods, these methods can be used by emergency managers to provide a better estimation of flooding extents while simulating storm events.

Improving the Accuracy of a Real-Time ADCIRC Storm Surge Downscaling Model

by
Carter A. Rucker

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Civil Engineering

Raleigh, North Carolina

2020

APPROVED BY:

Elizabeth Sciaudone

Helena Mitasova

Joel Dietrich
Chair of Advisory Committee

DEDICATION

To my Mom, Dad, and sister, Sydney.

BIOGRAPHY

Carter Rucker was born in Birmingham, AL in 1996 to Amy and Alan Rucker. He spent his early childhood in various locations before settling in Raleigh, NC in 2006, which became his home. Carter grew up going to the beach with his family and developed a passion for the coast. He attended Panther Creek High School and graduated in 2014 before attending North Carolina State University. While completing a B.S. in Civil Engineering, Carter began engaging in various research projects with Dr. Elizabeth Sciaudone. This introduced him to the world of coastal modeling, through which he met Dr. Casey Dietrich. After finishing his undergraduate degree in 2018, Carter began his graduate research work with his advisor, Dr. Dietrich, continuing at North Carolina State University. Beyond the academic and coastal world, Carter enjoys listening to music, playing the bass guitar, and sports.

ACKNOWLEDGEMENTS

I would like to first thank my advisor, Dr. Casey Dietrich. What he has provided for me is far beyond what one could expect from any mentor. He has helped me every step of the way since the day I started graduate school and would never hesitate to go out of his way to make sure I have everything I need to succeed. His down-to-earth, approachable personality along with his knowledge for coastal engineering and understanding of what it takes to be an excellent researcher has not only made him a perfect advisor, but also a perfect professor and perfect member of the NC State coastal team.

I am thankful for my two committee members, Dr. Helena Mitasova, my former professor who taught me nearly everything I could ever need to know about GRASS GIS, and Dr. Elizabeth Sciaudone, with whom I began my research career and completed numerous field visits. Their support and suggestions have played a major role in making me the student and researcher I am today.

Thank you to each and every member of the NC State coastal engineering team. I am so lucky to be a part of a team that is so incredibly helpful and resourceful, while also being so close-knit and kind. It has been wonderful working with all of you and, as I continue my career, I can only hope to have close to as much of an amazing group of friends as my coworkers.

Finally, I would like to thank my parents and my sister, Sydney. You all have provided me with unwavering support throughout my life and inspire me every day to be the best person I can possibly be.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER

1 INTRODUCTION

Tropical cyclones (such as hurricanes) and extra-tropical storms (such as nor'easters) are major threats to coastal communities. These storms can produce wind damage and flooding, which impact coastal properties and put lives at risk. Storm surge, the increase in water levels due to these storms, is particularly dangerous to low-lying coastal and inland regions. Hurricanes Dorian (2019), Michael (2018), Florence (2018), Maria (2017), Irma (2017), Harvey (2017), and Matthew (2016) each caused more than $1 billion in damage during the past four years, totaling in $326 billion for the seven storms [Smith et al., 2020].

Emergency managers near the U.S. Gulf and Atlantic coasts rely on fast and accurate models to predict storm surge and coastal flooding (Figure 1.1). Water level prediction models must be accurate enough to be trusted, fast enough to use in forecasting, and precise enough to predict water levels at key infrastructure. These models are used by emergency managers for many purposes such as issuing evacuations, implementing safety measures, and allocating disaster relief funds.

For precise and accurate storm surge predictions, numerical model operators increase resolution to represent the flow interactions with smaller-scale features. However, any increase in model resolution also has an increase in computational cost. It is easier to increase resolution for hindcasts and flood zone mapping, when the increased time of the simulation is not a limiting factor. However, numerical models used for real-time forecasting

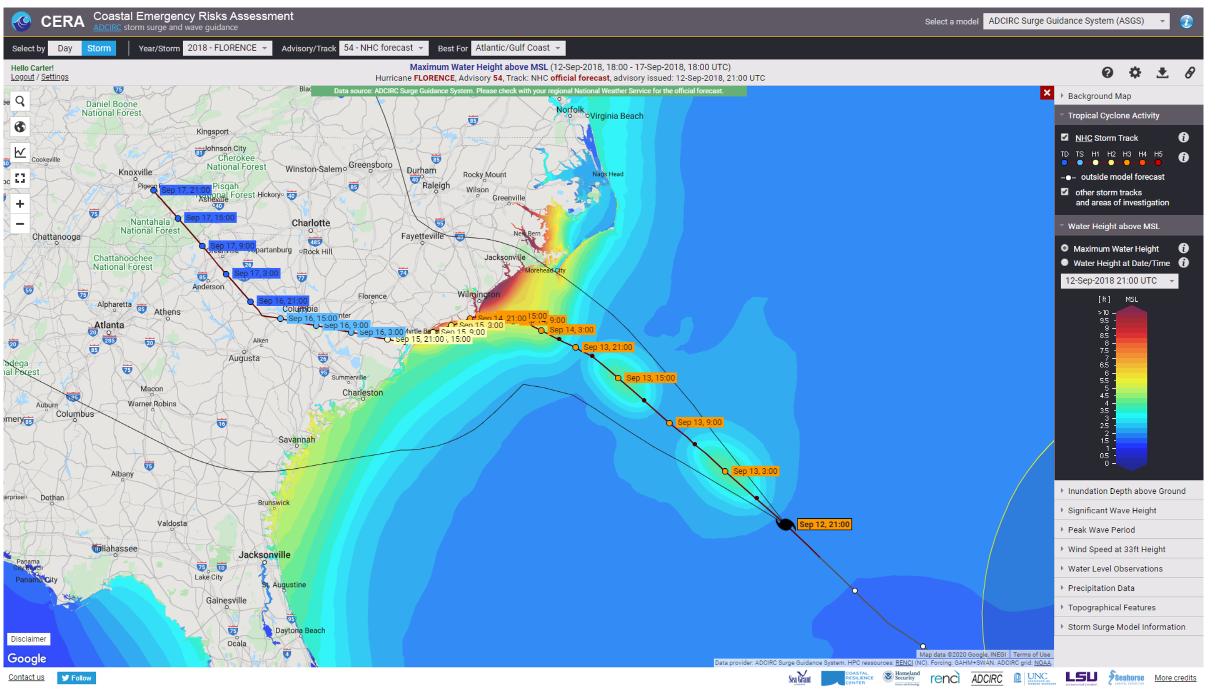

Figure 1.1 Storm surge forecast for Hurricane Florence (2018). This forecast was developed using the ADCIRC Prediction System (APS) based on the NHC Advisory 54, released on September 12th at 2100 UTC (about 2 days prior to landfall). The screen capture is a visual representation of APS results from the Coastal Emergency Risks Assessment (CERA) website (http://cera.coastalrisk.live). The APS predictions displayed in CERA are some of the most trusted and frequently viewed storm surge forecasts by emergency managers and coastal residents.

have time as a limiting factor, so the modeler must balance the resolution and cost.

The computational cost can be reduced by parallelization, in which models are run on high-performance computing clusters. This can allow for higher resolution and accuracy in an acceptable amount of time for use in forecasting. However, although relatively high-resolution results can represent topographic effects to the order of about 100 m [Cyriac et al., 2018], these models are still too coarse to represent small-scale infrastructure. Emergency managers are interested in storm surge overland flooding effects at spatial scales of neighborhoods, roads, and even building parcels to the order of about 15 m when making decisions regarding evacuation and various safety measures [Tull, 2018]. Fortunately, there are simplified methods that use geospatial post-processing to downscale the flooding predictions, without the need to increase resolution in a full-physics numerical model.

This study builds upon an existing method for downscaling, or "enhancing the resolution," of real-time storm surge forecasts, presented in Tull [2018]. The existing method used output from ADCIRC (Advanced Circulation), the current state-of-the-art system for forecasting storm surge, and downscaled results to 50-ft (15 m) resolution using open-source mapping software. The method was limited, however, in that it could only be applied in North Carolina (NC) at 50-ft resolution using a specific type of ADCIRC model output and considered only the predicted water elevations.

Multiple methods are proposed in this study to improve the applicability of the existing method and add simple physics to improve the accuracy of storm surge maxima estimates at small-scale infrastructure. The existing method has been modified to allow users to downscale ADCIRC flooding extents in any region throughout the world using any resolution with different versions of coarse output. Most of the preparation steps for creating a new downscaling method have been automated to reduce the user's need for prior Geographic Information Systems (GIS) and coding knowledge.

Two methods for adding simple physics to the existing method are proposed in this study: 1) the slopes method and 2) the head loss method. These proposed methods are compared to the existing method from Tull [2018], which is referred to as the static method. The static method only uses water elevations predicted by ADCIRC, while the proposed methods use additional information about the coastal region and/or the computed water levels. The slopes method uses both the value and the slopes of the ADCIRC-predicted water surface. The head loss method uses land classification data from the National Land Cover Database (NLCD) to account for energy losses in the storm surge as it travels over land cover types with various resistance factors.

To use the proposed methods in forecasting, the downscaled guidance must be generated at the desired resolution with high accuracy and be distributed to end-users within a minimal period of time. All proposed methods take computational cost and distribution time into account and have the ability to run automatically as forecasting data are received in preparation for a major storm. Additionally, each method is tested within a specific region of North Carolina but is designed to be easily replicable for use in any region of interest.

Chapter 2 provides background on data and resources used throughout this research, existing real-time storm surge forecasting models, tools used to visualize model output, existing post-processing methods in literature, and the objectives of this research. Chapter 3 describes an existing downscaling method, introduces two new downscaling methods, and explains models used to verify the accuracy of each method. Storm surge model performance and results of each downscaling method are described in Chapter 4, along with discussions regarding performance of the storm surge forecasting models and downscaling methods. Finally, key findings and importance of this study are discussed in Chapter 5 along with suggestions for future work.

CHAPTER

2

BACKGROUND

This research is not the first to model and downscale the predicted flooding extents for a coastal region. This chapter will present background information relevant to the state of existing knowledge prior to this research. Coastal regions are described with a wealth of observed data, at and below the spatial scales of infrastructure (Section 2.1). These data are then used for numerical model predictions (Section 2.2) and visualizations (Section 2.3) of flooding extents. Then these flooding extents can be downscaled as a post-processing step. Section 2.4 will present other existing geospatial post-processing methods, and Section 2.5 will outline the objectives for this research.

2.1 Geospatial Data and Software

2.1.1 DEMs and Land Cover Data

For a storm surge downscaling method, the user must identify data sources that are appropriate for a given study area. The downscaling methods presented in this research (static, slopes, and head loss) require a Digital Elevation Model (DEM) raster; the head loss method also requires land cover classification data.

DEMs are derived from LiDAR surveys and are available at high resolutions. The DEM used in Tull [2018] was obtained from NC Emergency Management (NCEM) and covers 32 NC coastal counties. It was derived from the Quality Level 2 LiDAR survey of NC in 2014 and has a 50 ft (15 m) resolution [Tull, 2018].

This research will also consider DEMs from the U.S. Geological Survey (USGS) The National Map (TNM) [*TNM Download*]. This source allows users to download DEMs from the National Elevation Database (NED) at various resolutions and provides DEM coverage at 1/3 arc-second resolution (approx. 10 ms) and 1 arc-second (approx. 30 m) throughout the entire U.S. except for parts of Alaska [USGS, 2015].

In addition to the DEM, the head loss method requires a raster containing land cover classifications that can be converted to friction coefficients (discussed later in Section 3.4). The source used in this research is the National Land Cover Database (NLCD), which provides nationwide (U.S.) data on land cover at 30-m resolution and classifies land cover into 16 types. These data can be downloaded for multiple different years, with 2016 being the most recent year and the year used in this study [*MRLC Viewer*].

2.1.2 GRASS

GRASS (Geographic Resources Analysis Support System) GIS is an open-source software [GRASS Development Team, 2017]. In this research, GRASS is used to enhance water elevation forecasts because it can be automated using Python scripts and is efficient at processing raster data. GRASS data are organized into "locations," which are directories containing all project data, and "mapsets," which are subdirectories in the location. All data within one location contain the same projection (coordinate system, datum, units, etc.), and each location must contain a mapset named "PERMANENT," which includes important information regarding the projection. Operations that alter the boundaries or resolution of a given mapset are done using the module `g.region`. Methods discussed in this thesis use regions chosen with state plane coordinates; this type of coordinate system allows the user to make horizontal measurements in feet or meters, rather than longitude and latitude. As discussed in Chapter 3, the methods utilize built-in GRASS functions to extrapolate raster data, perform numerical calculations, and convert results to a more user-friendly format.

2.2 Real-Time Storm Surge Forecasting Models

2.2.1 ADCIRC

Storm surge is the increase in water levels during a high-energy storm such as a hurricane due to sustained high-velocity winds and changes in atmospheric pressure. To model water levels due to storm surge, one must account for interactions between multiple factors such as tides, multidirectional wind fields with various velocities over different fetches, wave energy and its interaction with bathymetry, and physical interactions in overland flooding. These interactions all occur at different spatial scales. Tides must be modeled along global or ocean-size scales to capture interactions with ocean circulation and tidal constituents. In relation to storm surge, the effects of winds and waves depend on nearshore bathymetry, which requires understanding small-scale interactions. Overland flooding depends on even smaller-scale model features such as land cover variations, changes in topography, and hard structures that impede flow [Westerink et al., 2008].

Depth-averaged shallow water equations are used to model the interaction between ocean circulation, tides, and surge. These shallow water equations are derived from the Navier-Stokes equations of fluid motion by averaging out turbulence and vertical variations. These equations are solved to predict water levels and current velocities by using multi-resolution physics-based numerical models. The model used in this research is ADCIRC. ADCIRC was built for modeling tides, but its addition of water level setup from winds and its coupling with wave models has allowed ADCIRC to become state-of-the-art for modeling storm surge [Dietrich et al., 2010].

Water elevations are obtained by ADCIRC from the generalized wave-continuity equation (GWCE) throughout the domain [Luettich et al., 2004]. ADCIRC uses a set of established points for which computations are completed at set time steps. These computational points create an unstructured mesh (Figure 2.1) of triangular finite elements connected by vertices (intersecting points) where the computations occur. The unstructured mesh allows for variability in element size and shape, which gives users the freedom to alter the orientation of elements to align with key features and vary the mesh resolution [Westerink et al., 2008].

Alignment of mesh vertices is one of the most important factors in creating an efficient ADCIRC model for a given storm scenario. Ideally, an ADCIRC mesh would have high resolution, or small spacing between vertices, throughout the entire domain. However, increasing the mesh resolution will increase the computational cost of the model and it will take longer for the model to run to completion [Bunya et al., 2009; Cyriac et al., 2018; Dietrich et al., 2010; Graham et al., 2017]. There are scenarios in which high-resolution models are feasible,

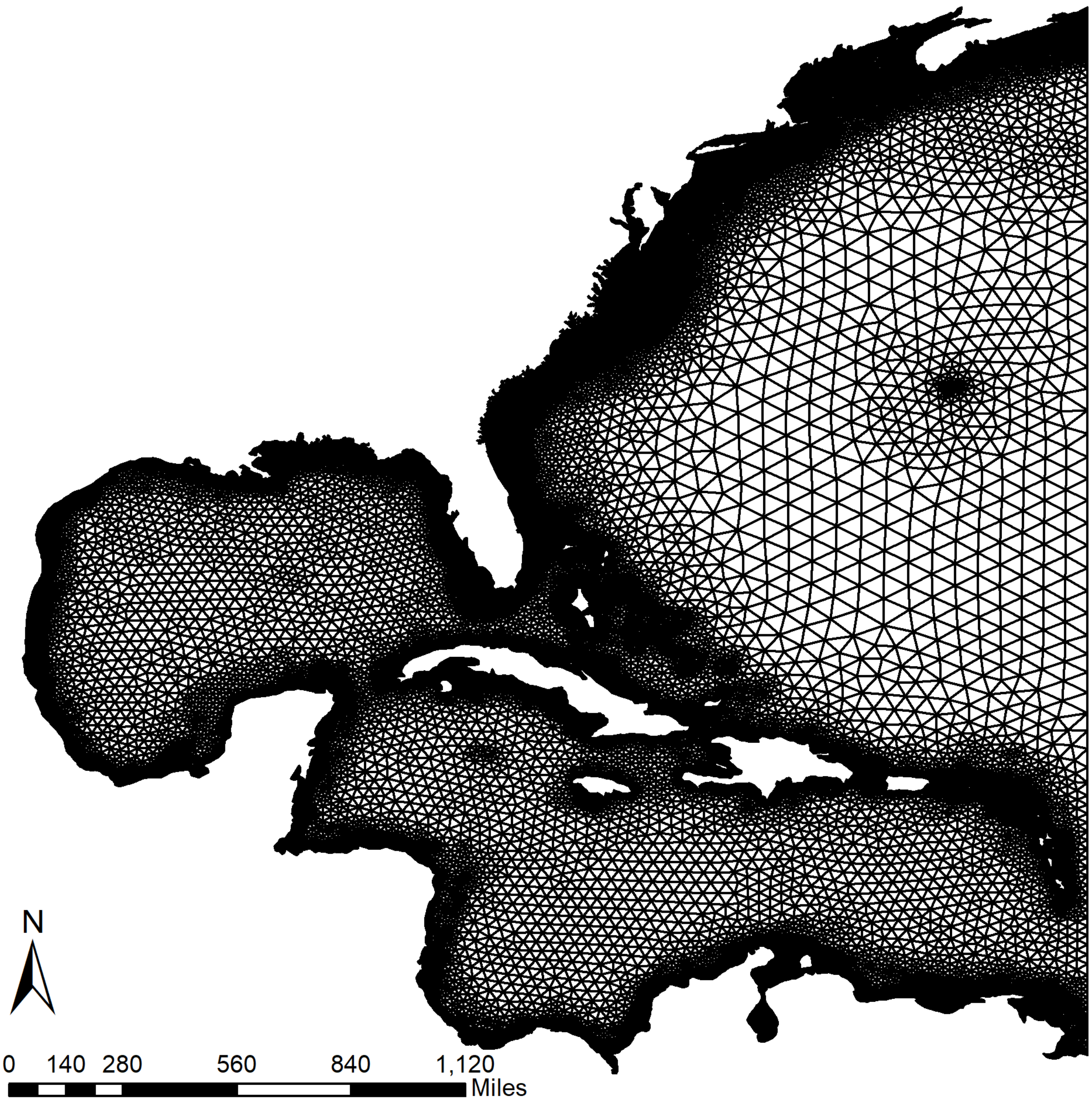

Figure 2.1 Example of an ADCIRC mesh. This ADCIRC mesh is a triangular, unstructured, finite-element mesh developed for flooding predictions in North Carolina. Spacing between vertices (points where mesh element corners intersect) is large in the open ocean and becomes smaller as interactions between vertices become more complex.

but in most cases and especially forecasting, modelers will prefer less computational cost. The freedom to place vertices anywhere can allow modelers to add computations to areas where smaller-scale interactions are important or remove computations from areas where they can be neglected. For instance, a tidal inlet might have vertex spacing at the scale of tens of meters to represent flows in and out of the back bay area, while vertex spacing in the open ocean portion of the same mesh might be at the scale of tens of kilometers. One of the main factors affecting vertex spacing is bathymetry, because, as depths become shallower or elevations surpass mean sea level to reach the overland flooding regime, interactions become more complex.

2.2.2 Using ADCIRC in North Carolina

An unstructured, finite-element ADCIRC mesh named NC v9.98 (herein referred to as NC9) was created for forecasting in North Carolina with a vertical datum as NAVD88 and horizontal datum as NAD83 [Blanton et al., 2008]. The domain of the mesh covers the entire Western North Atlantic, the Gulf of Mexico, and the Caribbean Sea to represent open-ocean boundary conditions and track hurricane movement. The mesh contains 622,946 computational vertices and 1,230,430 elements, with more than 90 percent of this resolution applied within coastal NC. Larger elements have a mesh spacing of 50 to 100 km in the Gulf of Mexico and open Atlantic, and elements decrease in size as the bathymetry transitions to shallower, near-shore conditions. Mesh spacing within the deeper portions of Albemarle and Pamlico Sounds ranges from 1500 to 1800 m and reduces to 100 to 300 m near river channels and shallower regions bordering the sounds. Resolutions of narrow river channels and areas along the NC coastline are generally less than 50 m [Cyriac et al., 2018]. The NC9 mesh is shown in Figure 2.2 at multiple zoom levels to highlight its variations in resolution. This mesh has been optimized to maximize resolution throughout NC while minimizing the total computational cost by minimizing the number of vertices [Blanton et al., 2010].

The NC9 mesh is the current standard used by emergency managers in NC [Blanton et al., 2008]. Emergency managers rely on the use of this mesh with the ADCIRC Prediction System (APS), specifically the ADCIRC Surge Guidance System (ASGS), a scripting system designed to automatically produce storm surge forecasts in the event of an approaching tropical cyclone [Dietrich et al., 2013; Dresback et al., 2013; Fleming et al., 2008; Forbes et al., 2010; Mattocks et al., 2006, 2008]. The ASGS receives data from the National Hurricane Center (NHC) containing parameters for storm size, intensity, and location through time in each forecast advisory every six hours during a developing storm. When these data are issued, the ASGS uses them to create a vortex wind model that is then used in ADCIRC to

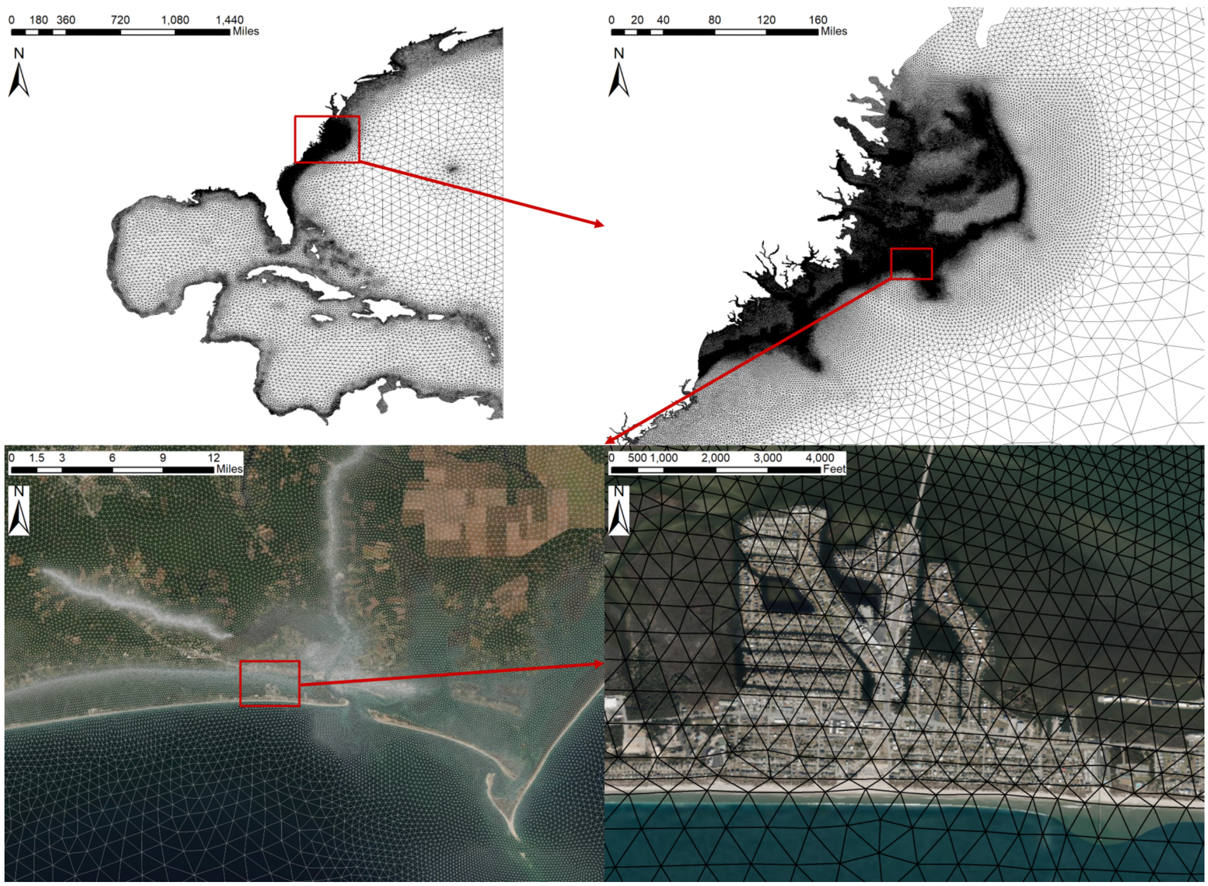

Figure 2.2 Examples of resolution in the NC9 mesh. The NC9 mesh was developed with high resolution in North Carolina with minimal resolution elsewhere. Three levels of zoom are displayed: the full NC9 mesh [top left], the NC9 mesh zoomed to the North Carolina coastal region [top right], the NC9 mesh zoomed to Carteret County, NC [bottom left], and the NC9 mesh zoomed to Atlantic Beach, NC [bottom right].

drive forecasts of storm surge. The ASGS is operated in several locations throughout the U.S., including NC. In NC, the system is referred to as the North Carolina Forecast System (NCFS) and uses the NC9 mesh. NCFS is managed and run using dedicated resources at the Renaissance Computing Institute (RENCI) [Blanton et al., 2012].

2.3 Kalpana

Kalpana is a Python code to visualize ADCIRC output as either ArcGIS-compatible shapefiles (ESRI shapefiles) or Google Earth-compatible KMZ files [Cyriac et al., 2018]. Kalpana can accept ADCIRC outputs in NetCDF format for maximum water levels, wind speeds, wave heights, and peak wave period. In this research, Kalpana is used to convert ADCIRC maximum water levels into an ESRI shapefile (Figure 2.3). These water levels represent the maxima achieved throughout the duration of a storm, and similar water levels are binned to form polygons and create a shapefile.

Kalpana can process data for any ADCIRC mesh within any region because it uses the World Geodetic System 1984 (WGS84) as its horizontal datum. This datum covers the entire globe and uses units of latitude and longitude, which are easily converted into units of feet or meters using GIS. Because Kalpana can convert mesh data into shapefiles that can be used universally, this eliminates the mesh and region dependency of the static method along with methods proposed in this research.

2.4 Geospatial Downscaling Methods

2.4.1 Static Methods

Concern for coastal and riverine flooding has led to a variety of studies that aim to provide flood forecasts at local and regional scales. The task of increasing the resolution of, or downscaling, flooded water level forecasts has become more feasible due to the increased availability of high-resolution elevation data. One technique for downscaling water levels is by using a static method, sometimes referred as a “bathtub” method.

Using the static method, water levels from a numerical model prediction are extrapolated horizontally, until the water surface intersects the topography, thus filling the surrounding topography with a constant water level (Figure 2.4). Because DEMs or other topographical data represent a higher resolution than numerical models, the static method produces results at a higher resolution. In addition to using this method for flood modeling purposes, the method has been used in studies associated with determining the effects of

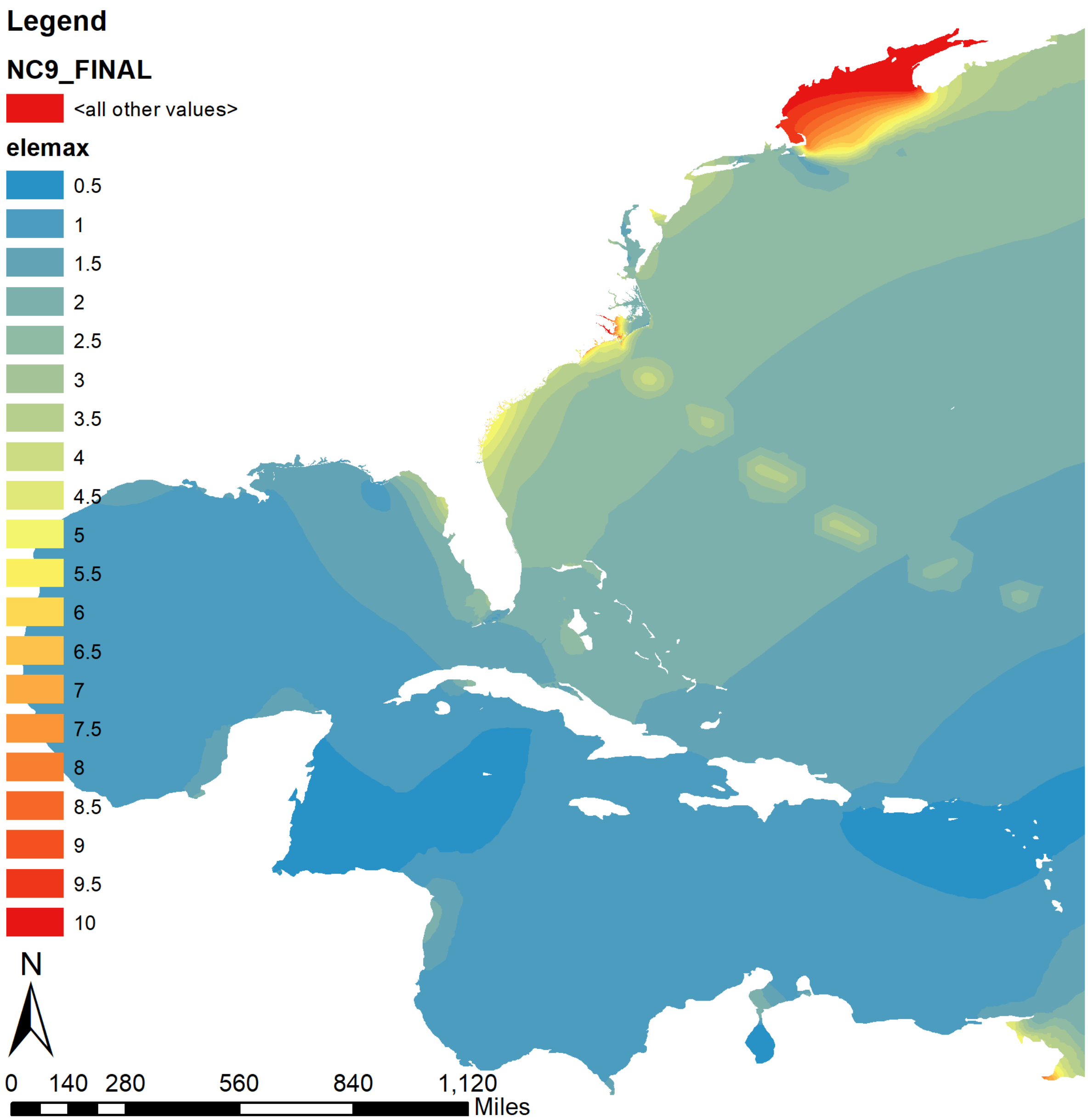

Figure 2.3 Example of Kalpana output. Water elevation forecasts for Hurricane Florence (2018) predicted by ADCIRC using the NC9 mesh, visualized with Kalpana as an ESRI shapefile. Elevations are binned to the nearest 0.5 ft, ranging from 0.5 to 10 ft.

various sea level rise scenarios [Brown, 2006; Heberger et al., 2009; Knowles, 2010; Lichter et al., 2012; Poulter et al., 2008].

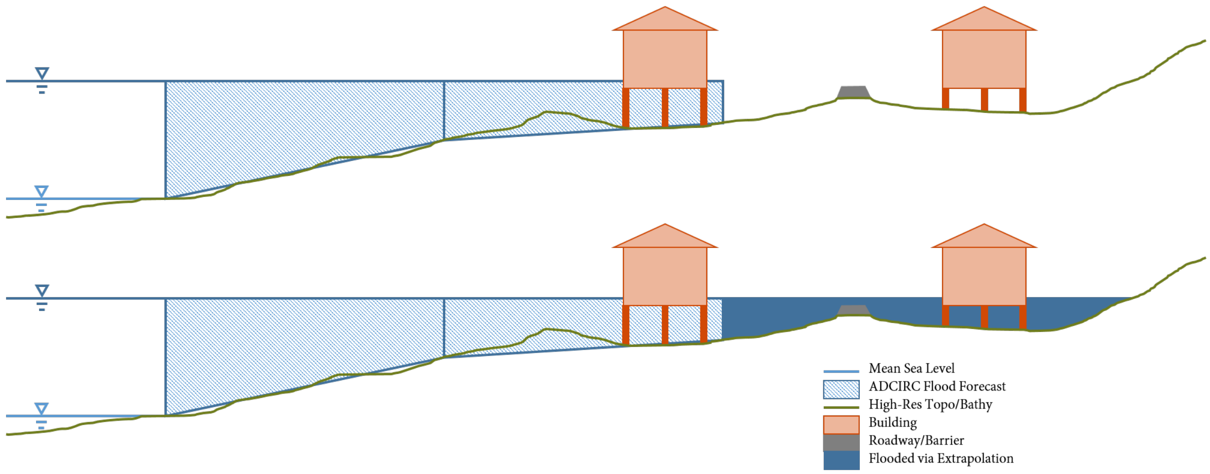

Figure 2.4 Schematic of the static method. The static method is represented in 1D, where the top image represents water elevations predicted by ADCIRC, and where the bottom image represents water elevations after downscaling. The two light blue polygons represent two flooded ADCIRC elements over a high-resolution DEM, represented by the green line. The ADCIRC water elevations are extrapolated horizontally until they intersect the DEM, adding the dark blue area to the predicted flood extents.

Although the static method can add resolution to numerical model predictions, the lack of physics incorporated in the method can lead to errors. Extrapolating the water levels statically will create a flat, horizontal water surface that may not be realistic. A flat surface is a good approximation at small scales, but physical processes such as wind velocities, interactions with waves and currents, and surface friction have an effect on water surfaces at medium and large scales.

Studies by Ramirez et al. [2016], Bates et al. [2000], and Bates et al. [2005] suggest that simple, physics-based methods outperform the static method. Gallien *et al.* (2011) explain that this method can instantaneously and erroneously flood entire neighborhoods if a single water level boundary point becomes higher than the neighborhood's flood protection structure. However, depending on the environment and the spatial scale, the static method can be sufficient for storm surge forecasting. Aerts et al. [2013] used ADCIRC to simulate water levels along the coast of New York City for several synthetic events. That study used a high-resolution mesh with small elements surrounding the coast and flooded area. Because the mesh allowed for relatively high-resolution results, a shorter extrapolation distance was required to fulfill the static method. This led to a simple and effective way to represent local inundation at an acceptable accuracy.

This research builds upon methods developed in Tull [2018], which form the basis to a method herein referred to as the "static method." The static method can downscale ADCIRC water level forecasts using high-resolution DEM rasters. A variation of the method is being used to forecast flood extents in North Carolina. Each time a new hurricane advisory is released, data are processed and downscaled versions are sent to emergency managers via email. Emergency managers use these results for a variety of reasons including evacuation plans, distribution of emergency resources, and allocation of disaster relief funds.

The static method works by downscaling ADCIRC results using various raster operations in GRASS. The method uses output from ADCIRC, specifically the maximum water levels throughout the duration of a storm within the entire domain of the ADCIRC mesh. These maximum water levels represent the maximum at each mesh vertex and therefore represent water levels occurring at different times. The maximum water elevations are then extrapolated horizontally (statically) across the domain until an equal or greater DEM elevation is reached (Figure 2.5).

The static method is an effective tool for downscaling ADCIRC data to a desirable scale for emergency managers, but leaves room for improvement. As developed by Tull [2018], the method could only be used in the state of North Carolina using a mesh developed specifically for NC, with only one NC DEM set to the geographic projection of NC. Also, for the method to maintain minimal computational cost, these data were processed in parallel on 16 CPUs. This made the method even less user-friendly and more difficult to replicate in other regions. To surpass these issues, this research uses the Python script *Kalpana*, which allows users to implement the static method for any region, DEM, or ADCIRC mesh.

Another issue with the static method is that it applied minimal physics; it only accounts for water levels predicted by ADCIRC and DEM data on a horizontal plane. This research will add to these physics by using slopes of water elevations in ADCIRC and applying energy reductions due to land cover along potential flow paths of the flooding water.

2.4.2 Methods with Slopes and Friction

Although static (or bathtub) methods are used often to downscale water elevations, these methods rely on a simple horizontal extrapolation with no mass or momentum balance. Addition of simple physics to downscaling methods can increase their accuracy at low cost and decrease the likelihood of large, non-physical flooded areas.

Storm surge approaching the shore can be conceptualized by considering water levels in terms of a surface slope $\Delta\zeta/\Delta x$, or a change in water surface elevation ζ over a lateral

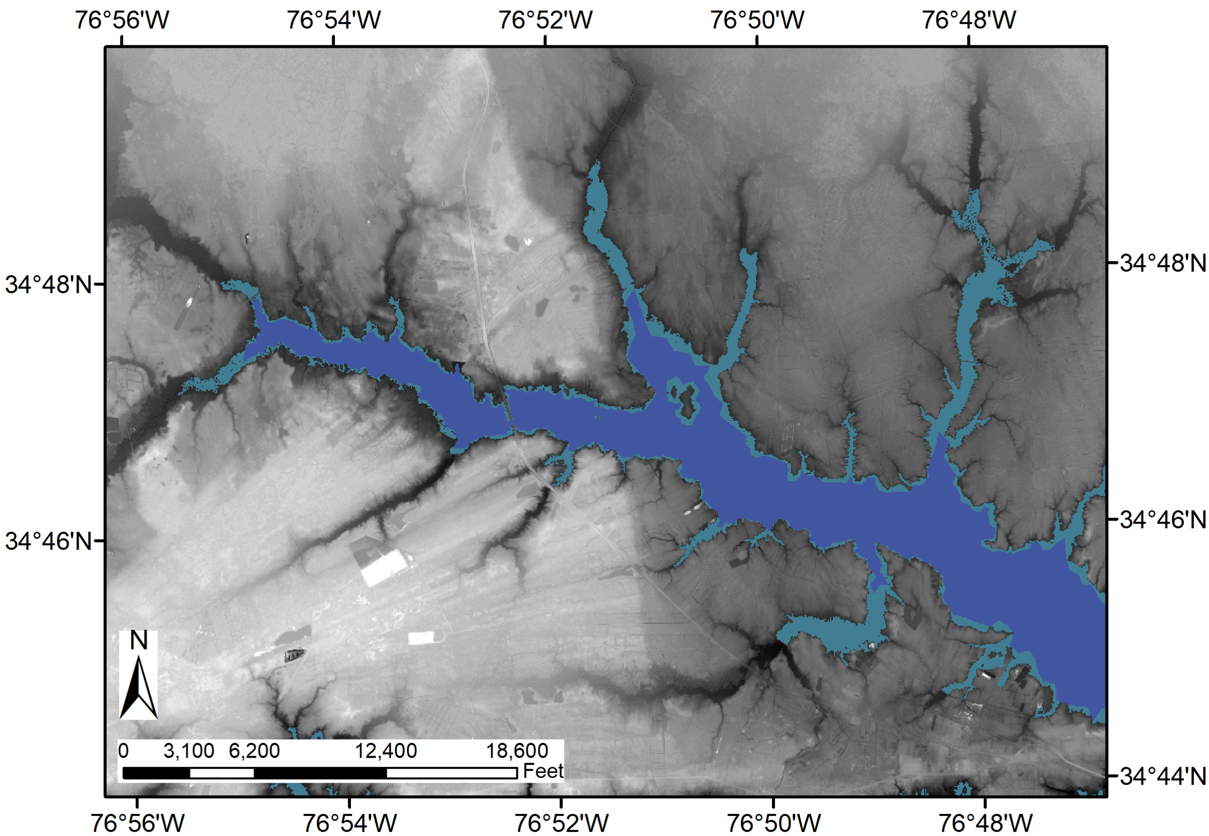

Figure 2.5 Comparison of flood extents from ADCIRC and the static method. Maximum water level extents predicted using the ADCIRC NC9 mesh [indigo, top layer] overlay the same water level forecast, but downscaled to the NCEM DEM using the static method [lighter blue, bottom layer].

distance interval x [Dean et al., 1984]. This relationship is a complex function of wind stress, bottom shear stress, and gravity among other factors. However, the slope can be extracted from the ADCIRC water elevation output by creating a continuous surface from model output vertices. The resulting surface can then be used to extrapolate water levels using the water surface and its slopes, rather than extrapolating the water surface horizontally (as done in the static method). Existing methods were not found to use this technique in literature, but this addition of basic physics can increase the downscaling method's accuracy at low cost.

One of the most common ways to increase the physical basis for the downscaled predictions of flood extents is to incorporate friction due to land cover. Different types of land cover are often associated with different magnitudes of energy loss as water travels over the surface. This is one of the most prevalent hydrologic principles and is used in state-of-the-art models such as HEC-RAS [Liu et al., 2018] and ADCIRC [Mayo et al., 2014; Passeri et al., 2011].

The Hydrologic Engineering Center's (HEC) River Analysis System (HEC-RAS) is a software developed by the U.S. Army Corps of Engineers and performs one-dimensional steady and 1D and 2D unsteady flow river hydraulics calculations. Using flow conditions as input, HEC-RAS is capable of predicting water surface profiles along stream corridors and their respective floodplains. The model uses resistance factors associated with the Manning equation (discussed in detail in Section 3.4), which are based on land cover variations when applied to overland flooding [Brunner, 2016b]. This model operates similarly to downscaling methods presented in this thesis, as water levels are propagated from the river source along adjacent floodplains.

The resistance factor associated with the Manning equation is referred to as Manning's roughness coefficient (n). This value n is lumped and distributed in hydraulic models to represent roughness, but can be a source of error for small-scale models. This research uses Manning's n values correlated to land cover classifications from the NLCD, which estimates classifications at a raster resolution of 30 m. Therefore, roughness is generalized into 30-by-30 m cells within the NLCD raster and some of the small-scale roughness features are smoothed out. The NLCD would not be a good source for small-scale applications of Manning's roughness coefficients but, over medium to large study areas such as the North Carolina coastline, this smoothing will still provide sufficient results [Kalyanapu et al., 2009].

Developing an ADCIRC mesh to forecast storm surge requires the user to provide input data associated with each mesh vertex. Some of the most important parameters assigned to each vertex that relate to overland flooding include the Manning's n coefficient, minimum depth in the wetting and drying algorithm, and spatially constant horizontal eddy viscosity

[Garzon et al., 2016].

Choosing the correct Manning's n value is an important step in mesh generation and can make a significant difference in forecasting overland flooding. Sensitivity studies have been completed for possible variations in Manning's n selection, as there is a range of acceptable values for each land cover classification. Passeri et al. [2011] completed a study on this topic in Florida's Big Bend Region. This study determined sensitivity using simulations with astronomic tides, a hindcast of Hurricane Dennis storm surge, and synthetic hurricane storm surge. Simulations used Manning's n varying from its minimum to its maximum values accepted by literature, using a separate model with constant quadratic bottom friction formulation for comparison. The study found that ADCIRC is sensitive to Manning's n in the study region and noted that inland areas, areas in close proximity to rivers, and areas near the wet-dry front show the most sensitivity. The maximum difference in water level peaks between the low and high Manning's n runs was 1.92 m for the synthetic storm and 1.33 m for Hurricane Dennis. For reference, Hurricane Dennis produced a storm surge of 6-9 ft above normal tides within the study region at Apalachee Bay, Florida [Beven, 2005]. Over a low-lying region such as this study area in Florida, differences in predicted flood elevations of this magnitude will make a large impact in total prediction of flood area.

The NLCD is perhaps the most frequently utilized source for developing Manning's n coefficients for hydraulic models [Bunya et al., 2009; Graham et al., 2017; Medeiros et al., 2012; Passeri et al., 2011], but any source of these empirically-derived roughness coefficients will come with error due to the high variability in land surfaces at a small-scale. Therefore, research is currently being done to develop methods to manipulate these values by means of data assimilation and statistics. Research done by Mayo et al. [2014] and Graham et al. [2017] suggest methods for changing Manning's n values in ADCIRC based on comparisons between hindcast or synthetic data and an associated "truth".

Existing literature was not found on studies that involve downscaling water levels with head loss due to land cover friction. However, the desire for achieving higher-resolution hydrologic model output has led researchers to develop "subgrid" models. Using subgrid models, lower-resolution models are able to operate while taking into account high-resolution underlying physics [Brunner, 2016a]. The manner in which subgrid models address head loss is similar to procedures used to develop the head loss method. Rather than incorporating these data within the model itself like a subgrid model, this research performs small-scale operations using the lower-resolution model's output after completion of the simulation.

Research presented in Volp et al. [2013] uses raster operations to create a finite volume hydrodynamic model for shallow water flow. The model computes on a coarse grid, but accounts for small-scale physics occurring on an underlying subgrid. This subgrid accounts for high-resolution bathymetry and roughness variations based on a nonlinear friction coefficient and velocity. HEC-RAS can also operate using a subgrid by pre-processing detailed underlying terrain data. This allows elements within the structured or unstructured mesh to become partially wet, rather than only wet or dry [Brunner, 2016a].

2.5 Objectives

The purpose of this study is to improve the accuracy and applicability of real-time storm surge forecast downscaling methods using GIS techniques. To achieve this goal, the following objectives and tasks must be met:

1. Evaluate the accuracy of the existing static method.
2. Increase the applicability of the static method by eliminating its dependency on a specific mesh, DEM, and geographical region.
3. Develop and evaluate a method that downscales water levels by slope, rather than statically.
4. Develop and evaluate a method that downscales water levels while including friction losses due to land cover.

CHAPTER 3

METHODOLOGY

This chapter will give detailed explanations of the methods used to construct each storm surge downscaling method presented in this research. Section 3.1 will review parts of the static method from Tull [2018] that are relevant to this research, Section 3.2 will present improvements made to the static method, Sections 3.3 and 3.4 will describe implementation of the slopes and head loss methods, and Section 3.5 will explain methods used to verify the accuracy of each downscaling method.

3.1 Static Method

This research will build on the prior work by Tull [2018], who implemented a static method for North Carolina using a DEM raster obtained from NC Emergency Management (NCEM). This static method can enhance the resolution of ADCIRC forecasts by (1) interpolating ADCIRC water levels to a raster at the same resolution as the DEM, (2) extrapolating this water level outward until it intersects the DEM, and (3) converting the enhanced raster to polygon format for convenient, efficient distribution [Tull, 2018]. This method is described briefly in the following subsections, because it is the basis for the developments in this research.

3.1.1 Direct Mesh-to-Raster Interpolation

The static method was developed specifically for the NC DEM provided by NCEM (Figure 3.1) along with the NC9 ADCIRC mesh [Tull, 2018]. The original method interpolated ADCIRC water levels to the DEM without any conversion to a shapefile or another format. This interpolation used an Inverse Distance Weight (IDW) method, in which raster cells not positioned in-line with ADCIRC vertices were filled using linear interpolation. Using the given DEM and mesh, this required interpolation of about 600,000 mesh vertices to a 430 million-cell raster.

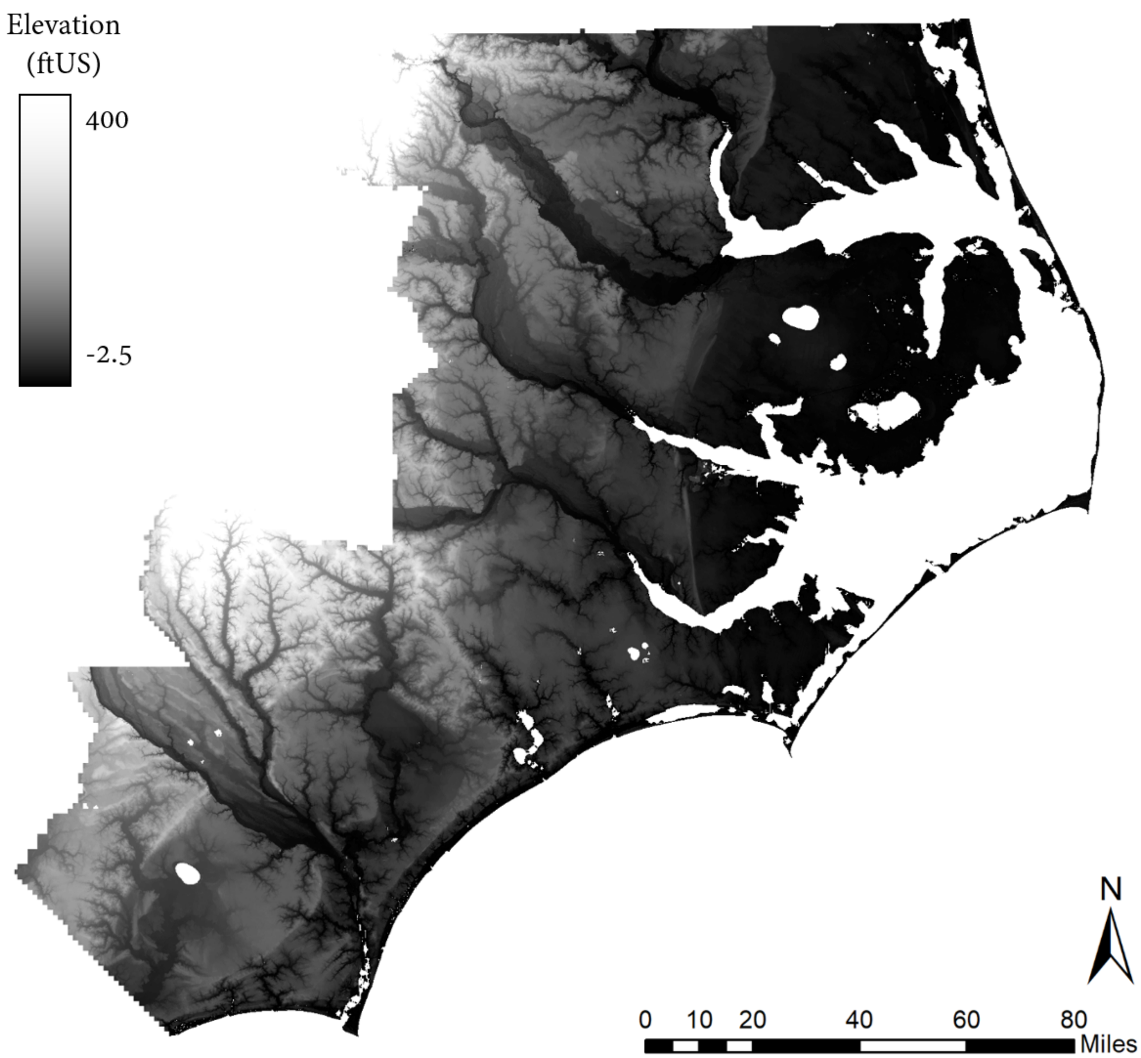

Figure 3.1 NCEM DEM for the 32 counties in coastal NC, with ground surface in a 430-million-cell raster.

Interpolation to a high-resolution raster over the entire coastal region of NC is computationally intensive, so IDWs were pre-computed to allow this process to be used in forecasting. A large text file contained six “weight” values of each raster cell. Each cell contains the numbers of the three closest ADCIRC mesh vertices, as well as the distances from the cell to each of those vertices. Pre-computing the distances to the nearest vertices reduced computational time, because, after receiving water elevation input corresponding to each mesh vertex, only a simple algebraic expression was required to interpolate linearly from water elevations at each vertex to each raster cell. Still, this method required a “weights” file that was unique for each mesh/raster combination, and thus it was not extendable to other ADCIRC applications.

This process of pre-computing and using interpolation weights has been superseded by using a shapefile of maximum water levels, as described in Section 2.3.

3.1.2 Extrapolation as Flat Surface

After ADCIRC water elevation values were interpolated onto the raster, they were ready to be downscaled. The newly-created raster surface had cell resolutions of 50 ft, but the flooding extended only to the resolution of overland ADCIRC elements within the NC9 mesh (~100 m). The water levels in this raster were then extrapolated to intersect the DEM surface (Figure 2.4) using GRASS GIS techniques.

The main GRASS module was `r.grow`, in which `r` signifies a raster operation. The `r.grow` module is also used extensively in methodology presented in Sections 3.3 and 3.4. The module operates by iterating through each raster cell in a given region. Whenever a null cell is found, `r.grow` can assign the value of the nearest non-null cell to the null cell. Using this GRASS module, the static method operates with the following workflow:

1. Extrapolate ADCIRC water levels statically using `r.grow`
2. Remove dry cells using a raster calculation
3. Remove hydraulically-disconnected cells using clumping techniques and a specified maximum flood distance radius
4. Convert resulting water elevations from a raster to a shapefile

First, ADCIRC water elevations are extrapolated using `r.grow`; therefore, for each raster cell that *does not* contain a water elevation interpolated from the ADCIRC mesh, the cell is assigned to contain the value of the nearest cell which *does* contain a water elevation. Water elevations are extrapolated horizontally throughout the domain to each cell outside the original ADCIRC flood extents. Because `r.grow` uses the water level from the nearest non-null cell, the downscaled water surface is effectively flat.

Then, after extrapolation using `r.grow`, a raster calculation is used to remove all dry cells. Because the ADCIRC water levels are given in *elevations* relative to the NAVD88 vertical datum, these values can be compared directly to the DEM. If the water elevation is higher than the ground surface, then the water elevation is written to the output raster. If not, then the raster cell is assigned as null.

Next, all cells containing a water elevation less than the topographical elevation found in the DEM are considered to be dry and removed from the output raster. This raster now contains all of the desired water elevation values, but also typically contains physically-illogical flooding extents, i.e. newly-wet regions that are disconnected hydraulically from the ADCIRC-predicted flooding.

Within the region of North Carolina used in Tull [2018], along with many coastal regions throughout the world, the topography is generally low-lying and can have low elevations without being in close proximity to the coastline. These low-lying areas can be of similar elevation to areas closer to the coastline, but are separated by topographical features or barriers, thus raising the issue of hydraulic connectivity.

The static method considers hydraulic connectivity in two ways: 1) by clumping adjacent flooded raster cells and removing the smallest clumps, and 2) by setting a maximum radius of lateral flooding extents originating from the predicted ADCIRC flooding extents. The clumping procedure removes small areas of predicted inundation that are not connected to the largest areas (i.e. the ocean, sounds/bays, rivers). The maximum radius is set as a threshold where, if water is extrapolated beyond this extent, these values are not considered to be flooded. The raster now contains only extrapolated water elevation values that exceed topographical elevations and are connected hydraulically.

Finally, to reduce the size of the data set and for convenience of use by emergency managers, the raster is then converted to a polygon shapefile. Because the raster contains many unique values, converting it directly to a shapefile would create separate polygons at nearly every raster cell, so water elevations are binned to the nearest 0.5 ft. For example, adjacent water elevations between 0 and 0.5 ft, between 0.5 and 1 ft, etc. will become their own separate polygons. This shapefile is much smaller in size than its raster counterpart, making it easier to send via email to end-users and to store for processing. End-users can use this shapefile in GIS mapping software to compare predicted water elevations and flooding extents to spatial data sets to quickly determine buildings, infrastructure, or other areas of interest at risk of flooding during a storm.

A similar process of clumping to remove hydraulically disconnected cells, binning water levels to a set interval, and converting these bins to shapefiles will be used in each of the three downscaling methods (static, slopes, and head loss) when used in forecasting.

3.2 Improved Applicability of Downscaling Methods

The static method is an excellent tool for downscaling ADCIRC water elevations, but the work presented in Tull [2018] was applicable only in North Carolina. The interpolation process used a set of IDWs that were specific to the NC9 ADCIRC mesh and Python scripts that were hard-coded for the DEM provided by NCEM and the North Carolina state plane geographical region. This section will present methods that eliminate regional limitations and allow any mesh, DEM, or raster resolution.

The first contribution of this research is the integration of the static method with Kalpana, so the enhanced guidance can be produced with this well-documented, open-source, visualization tool. The new, extended version of Kalpana can be downloaded from GitHub (https://github.com/ccht-ncsu/Kalpana). Users can run the static method for any DEM, resolution, geographical region, and/or ADCIRC mesh. Step-by-step instructions for creating a simulation can be found by visiting the NC State University Coastal & Computational Hydraulics Team website Kalpana page (https://ccht.ccee.ncsu.edu/kalpana/) or following the hyperlink: **Downscaling ADCIRC Flooding Inundation Extents Using Kalpana**.

3.2.1 Importing ADCIRC Data with Kalpana

The first improvement was to address the interpolation of ADCIRC water elevations onto the DEM raster. This process was computationally-costly and the most time-consuming step. Interpolating ADCIRC values directly to the DEM using IDWs in serial takes approximately 39 minutes out of the total 64 minutes required for forecasting using the IDW method. Time restrictions required this method to be parallelized into 16 processes to be used in forecasting. Additionally, a full set of IDWs had to be generated for each ADCIRC mesh. This was not an automated process, and it required an experienced GIS user.

Now the interpolation is done with Kalpana. Using the existing Kalpana functionality, the ADCIRC water elevations are binned and converted to a shapefile. Then, using GRASS commands within Kalpana, the shapefile is converted into a raster. Now using Kalpana, this direct interpolation can be performed with any ADCIRC mesh and any DEM, for any coastal region. The only downside is the binning of the water elevations, and thus a loss of precision. However, the water elevations were binned already in the final shapefile so the loss in precision is negligible.

This direct interpolation via shapefiles with Kalpana saves time. For a typical ADCIRC mesh and DEM, the water elevations are converted to a shapefile, then imported to GRASS and converted to a raster in about one additional minute. This is a significant reduction as compared to the IDW interpolation, which took 39 minutes. Thus, this integration with Kalpana is an improvement in both usability and efficiency.

3.2.2 Removing Regional Hard-Code

The next improvement was to generalize the GRASS location, which was hard-wired previously for the NCEM DEM and the North Carolina state plane coordinate system. This issue is resolved with another Python script that automates the creation of a GRASS location for use with the static method. The script, which has been added to the newest version of Kalpana, accepts user input for the desired DEM(s), geographical region, and resolution and requires minimal-to-no prior GIS knowledge to operate.

The static method requires two GRASS locations: one uses the WGS84 datum, and another uses a state plane coordinate system. The WGS84 location does not contain any information other than the WGS84 global datum and can be downloaded directly from GitHub. As discussed in Section 2.3, this location is necessary for importing shapefiles generated by Kalpana, because the shapefiles are created using the WGS84 datum. Aside from this step, all other operations are done in the state plane coordinate system location.

While the WGS84 GRASS location does not contain any region-specific information and will be the same for every user, the state plane coordinate system GRASS location must be created by the user with the aforementioned Python script. After accepting user input, a zip file will be created that contains the user's GRASS location. This location will contain user-specified information regarding the desired state plane coordinate system and raster resolution along with the DEM which will be used with the static downscaling process.

3.2.3 Examples of Use of Improved Static Method

The static method [Tull, 2018] was only capable of being used in North Carolina using the NCEM DEM and the NC9 mesh. To use the static method for a different region, DEM, and/or ADCIRC mesh, the user was required to create a new IDW file corresponding to each new region, DEM, and mesh. After slight modifications and by using Kalpana, the static method is now capable of being used anywhere in the world. The method can also be run in serial for forecasting in a similar amount of time required to run the [Tull, 2018] version on 16 parallelized cores. Automating the GRASS location creation feature and creating an instructional website has further increased the ease of using the static method.

After integration with Kalpana, the static method is capable of being used globally with any desired DEM, resolution, and ADCIRC mesh. In addition, the time required to interpolate ADCIRC data onto a raster has been greatly reduced; the entire downscaling process now takes 30 minutes to run to completion in serial for the NCEM DEM. This downscaling time now meets the standards of NC emergency managers and parallelization is no longer required in forecasting. Kalpana was run in serial to downscale ADCIRC results to the NCEM DEM and distribute resulting shapefiles to emergency managers in North Carolina during the 2019 hurricane season.

Based on communication with end-users, the static method has already been applied in other regions. For example, a researcher is using the static method to assess estuarine flood predictions in the Chesapeake bay and a company is using the static method to assist decision-making for flood map development.

Recently, I developed a downscaled hindcast for Hurricane Dorian to assist FEMA with post-hurricane disaster relief planning. This hindcast was developed using DEMs from USGS TNM that covered the U.S. Atlantic coast from southern Florida to the northern border of North Carolina (Figure 3.2). Because different state plane projection systems were required for each of the four states covered (Florida, Georgia, South Carolina, and North Carolina), a separate guidance product was produced for each state. These data were derived from an ADCIRC hindcast with the HSOFS (Hurricane Surge On-Demand Forecasting System) mesh; an additional run was completed using the NC9 mesh for North Carolina. To minimize computational cost, the western extent of the DEMs was limited to the edge of the HSOFS and NC9 meshes, while the eastern extent was created by using a small buffer that extended eastward from the land-sea interface. This buffer is necessary for capturing ADCIRC data over water where ADCIRC does not predict overland flooding.

3.3 Downscaling by Water Elevation Slopes

After improving the static method, this research has developed two methods for improving how the physics of overland flow are represented in the enhanced guidance. The first method is downscaling by water elevation slopes (referred as the slopes method). Instead of extrapolating statically, the slope of the water elevation surface is considered (Figure 3.3). By creating a maximum water elevation surface from the ADCIRC results, the slopes can be computed and used for downscaling.

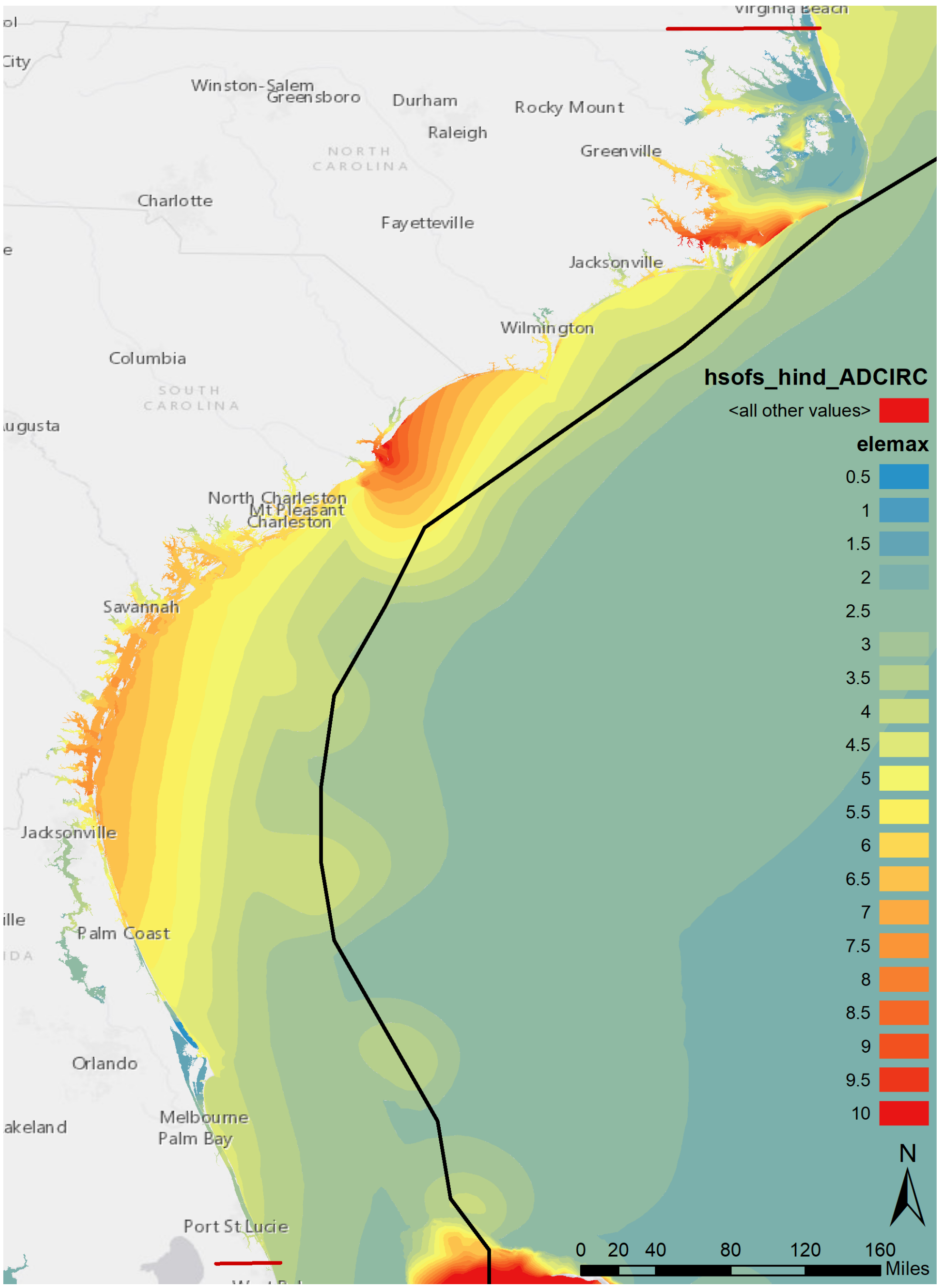

Figure 3.2 Example of downscaled Results for Hurricane Dorian (2019). Downscaled results produced for FEMA using hindcasts of Hurricane Dorian (2019) are shown, where water levels in blue represent 0.5 ft elevations and water levels in red represent 10+ ft elevations. The upper and lower limits of the downscaled extents are marked using red lines along the northern border of NC and a mid-latitude in FL. The NHC best track is represented by the black line.

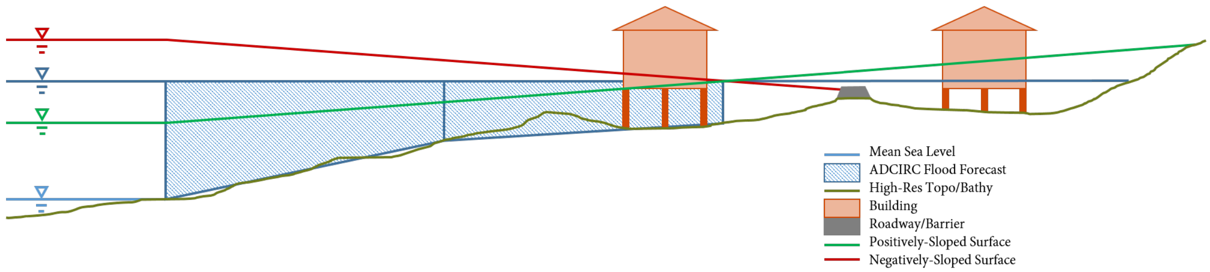

Figure 3.3 Schematic of the slopes method. The dark blue line and light blue polygons represent an ADCIRC forecast with a horizontal surface slope, which can be extrapolated horizontally with the static method, similar to Figure 2.4. Now the downscaling method can also consider downward (negative) and upward (positive) slopes, shown in red and green lines, respectively. Note the difference in flooding extents at the right of the figure, depending on the slope of the water surface.

3.3.1 Generating a Continuous Water Elevation Surface

As noted above, Kalpana was used in the static method to convert ADCIRC data to a shapefile for use in GRASS GIS. However, because the water elevations were binned in these shapefiles, they cannot be used to compute slopes of the water elevation surface. Thus, a new method is proposed to create a surface of water elevations that contain all original slopes between ADCIRC vertices.

To generate a more-realistic, binless surface, ADCIRC maximum water elevation values are converted to a comma-separated-variable text file, in which the latitude, longitude, and water elevation are provided at every mesh vertex. Then these data are used to generate a shapefile with points at all mesh vertices. Finally, a surface is made using the `v.surf.rst` GRASS module (Figure 3.4), in which `v` signifies a vector operation. This module uses a regularized spline with tension to approximate a raster surface based on given vector data, as described in the GRASS GIS manual [*v.surf.rst*].

By using `v.surf.rst` to generate a surface using a regularized spline, the surface does not precisely match the water elevations computed by ADCIRC. However, the average difference between raster surface values and each corresponding ADCIRC water elevation was 3.8 mm, and thus smaller than the bins in the final shapefile.

3.3.2 Generating Water Elevation Slopes and Downscaling

After a raster surface is created with continuous water elevations, new raster surfaces can be developed to contain values for water elevation slopes by using the `r.slope.aspect` GRASS module. With the slopes method, the user must consider both positive and neg-

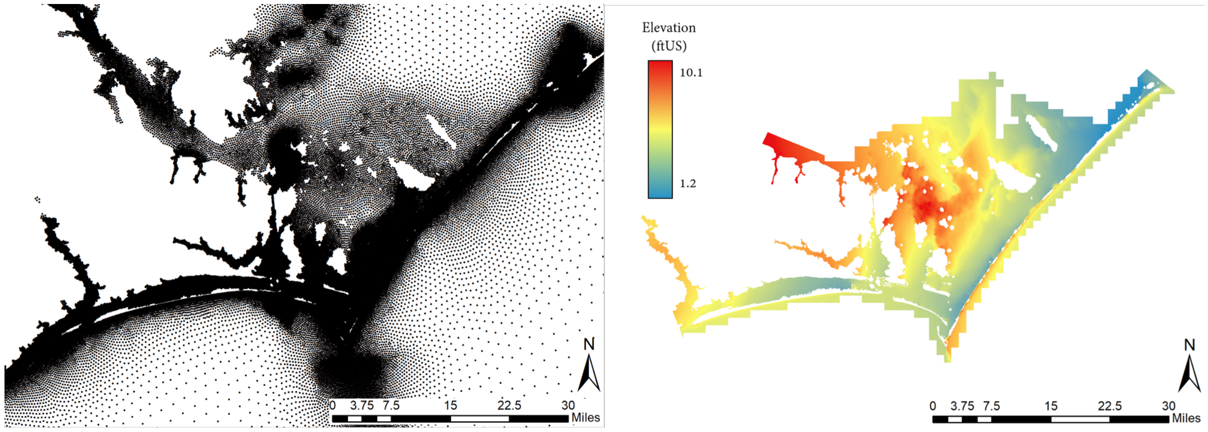

Figure 3.4 Conversion of maximum water elevations from ADCIRC mesh vertices to a continuous raster surface. The maximum water elevations are converted from ADCIRC vertices to points [left]. GRASS uses these data to convert the points to a raster surface [right], spatially limited to the region of interest, which is set using the DEM representation of Carteret County.

ative slopes in the water elevation surface. Additionally, it is important to determine the direction of each slope value before extrapolating these slopes. The typical output from `r.slope.aspect` is omnidirectional slope and only exists as a magnitude. Thus, directional slope is found using the `slopedx` and `slopedy` functions, which are partial derivatives of the water surface slope in the x and y directions. Because these rasters are oriented such that cells roughly align with the cardinal directions, these are east-west and north-south slopes.

In the downscaled portion of the surface resulting from the slopes method, new water elevations are determined by a simple function of the slope multiplied by the horizontal distance traveled, added to the static extrapolation value, as shown in Equation 3.1 below.

$$\zeta_{\text{slopes}} = \zeta_{\text{static}} \pm c\left(m_x \Delta x\right) \pm c\left(m_y \Delta y\right) \tag{3.1}$$

in which ζ_{slopes} is the water level at a null cell as computed with the slopes method, ζ_{static} is the water level at the same null cell as computed with the static method, c is a slope exaggeration multiplier (set to 1 for this research), m_x and m_y are the slopes from the `slopedx` and `slopedy` functions, and Δx and Δy are the distances between the null cell and the extents of the ADCIRC prediction based on the state plane coordinate system. In terms of cardinal directions, Δx is the distance traveled in the east-west direction, and Δy is the distance traveled in the north-south direction.

The slopes method is similar to the static method in that both are a series of `r.grow` operations and raster calculations. Because the slopes method must consider horizontal

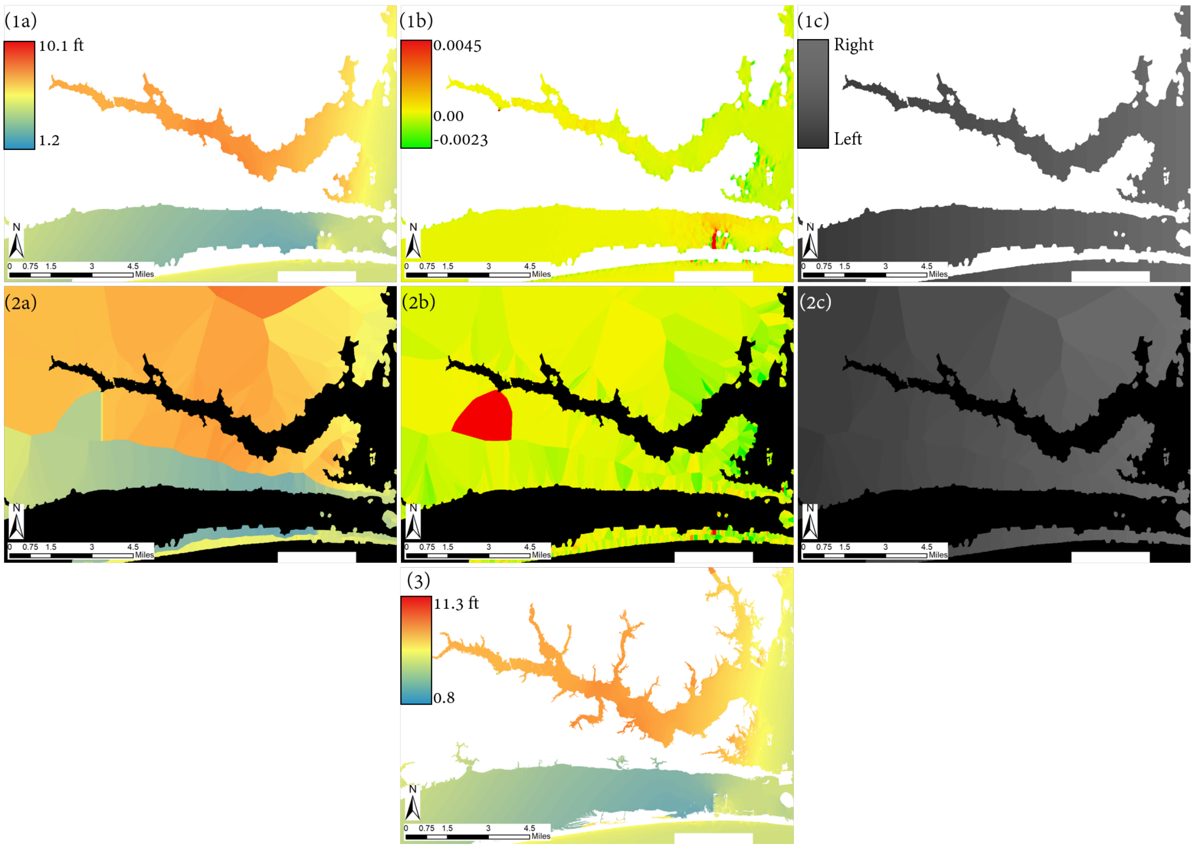

Figure 3.5 Slopes method sequence. Steps for extrapolating water elevations by slope are shown for the Newport River and Bogue Sound, NC area. The first row [1a-1c] displays the ADCIRC maximum water elevation surface (Section 3.3.1, Figure 3.4), the slopes of the ADCIRC water surface (x direction only), and the x value of each raster cell in the ADCIRC water surface. The second row [2a-2c] is each raster from the first row extrapolated to fill each nearest null cell using `r.grow`. Original ADCIRC values in this row are colored black to help the reader differ between ADCIRC values and extrapolated values. The final result from the slopes method is shown in image [3]. Please note that, because only `slopedx` and x values are shown, similar calculations are done in the y direction to produce the final raster in [3].

distances and slopes in both the x and y directions after computing a continuous raster surface using splines, this method is more computationally-intensive than the static method. Added computational cost might require the slopes method to be parallelized over large domains by simultaneously operating on smaller sections of the full DEM raster.

3.4 Downscaling with Head Loss due to Land Cover

The second method of adding simple physics to the downscaling process is to include energy head loss due to friction from land cover variations (referred as the head loss method). Head loss due to land cover is incorporated into many hydrologic studies and in ADCIRC to simulate overland flooding (Section 2.4.2). As water floods over a surface, energy is dissipated as it comes into contact with any form of roughness that might interfere with the flow. Overland flow caused by storm surge is hindered most by hard structures and dense vegetation, while developed surfaces and wooded wetlands allow the surge to flow more freely (although energy is still being dissipated). Rather than extrapolating the water elevations until they intersect with a DEM surface (as done in the static and slopes methods), the head loss method extrapolates water elevations to an energy cost surface (Figure 3.6), which represents the total energy required (elevation plus head loss) to reach a flood extent.

3.4.1 Selecting Land Cover Variation Data

In ADCIRC, along with many other hydrological models, the amount of friction associated with certain types, or patterns, of land cover is represented using a Manning's n coefficient. This coefficient is typically used with Manning's equation (3.2):

$$U = \frac{k}{n} R^{2/3} S^{1/2} \tag{3.2}$$

in which U is the flow velocity, k is a unit conversion ($k = 1$ for SI units; $k = 1.49$ for U.S. customary units), n is Manning's coefficient, R is the hydraulic radius, and S is the slope of the energy grade line. Likewise, Equation 3.3:

$$Q = \frac{k}{n} R^{2/3} S^{1/2} A \tag{3.3}$$

can be used to estimate channel flow Q, where A is the cross-sectional area of the channel.

This research uses data from the USGS National Land Cover Database (NLCD) to determine land cover classifications throughout the study area. These classifications are available in raster format with 30 m resolution throughout the United States [USGS, 2019].

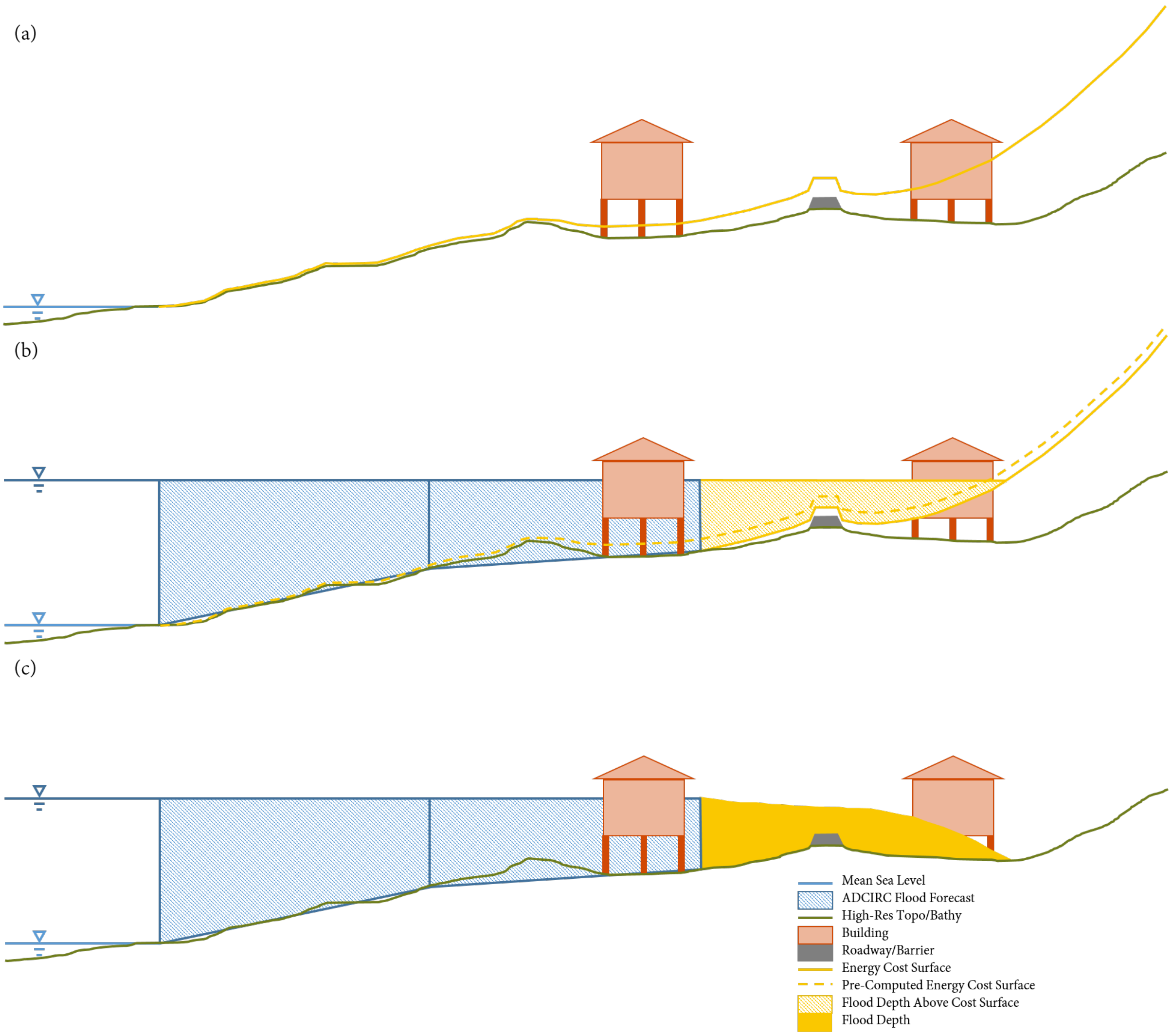

Figure 3.6 Downscaling with head loss. A 1D representation of the head loss method is presented, where the energy cost surface [yellow line] is the elevation [static head] plus head loss. The yellow area is the flood depth above the cost surface and will be used to determine the actual flood depth (Section 3.4.4). The downscaled water levels in the lower panel are then included in the enhanced guidance.

Table 3.1 Manning's n Values Associated with NLCD Land Cover Classifications

Land Cover	Description	Manning's n
11	Open water	0.001
21	Developed, open space	0.0404
22	Developed, low intensity	0.0678
23	Developed, medium intensity	0.0678
24	Developed, high intensity	0.0404
31	Barren land	0.0113
41	Deciduous forest	0.36
42	Evergreen forest	0.32
43	Mixed forest	0.40
52	Shrub/scrub	0.40
71	Grassland/herbaceous	0.368
81	Pasture/hay	0.325
82	Cultivated crops	0.037
90	Woody wetlands	0.086
95	Emergent herbaceous wetlands	0.1825

The data are generated by assembling and preprocessing Landsat imagery and geospatial ancillary datasets, then passing these through a decision tree that is entrained to determine the land cover classification. Land cover classifications are easily related back to Manning's equation by associating a Manning coefficient value to each classification, as shown in Table 3.1 and Figure 3.7. There is a range of acceptable Manning's n values associated with each land cover, so the values used in this research are derived from Liu et al. [2018] and Kalyanapu et al. [2009].

3.4.2 Friction Losses in Downscaling

Manning's equation is used typically for estimating channel flow velocity, but can be manipulated for use with overland flooding due to storm surge. In downscaling with raster operations, we consider Equation 3.3 with a wide channel ($R \approx$ depth of flow) and flow area equal to the width of one raster cell multiplied by the depth of flow. The Manning coefficient will be the value assigned to each cell based on NLCD data, while S is the slope of the energy grade line between cells. Now a manipulated form of Manning's equation is utilized to calculate the energy grade line slope:

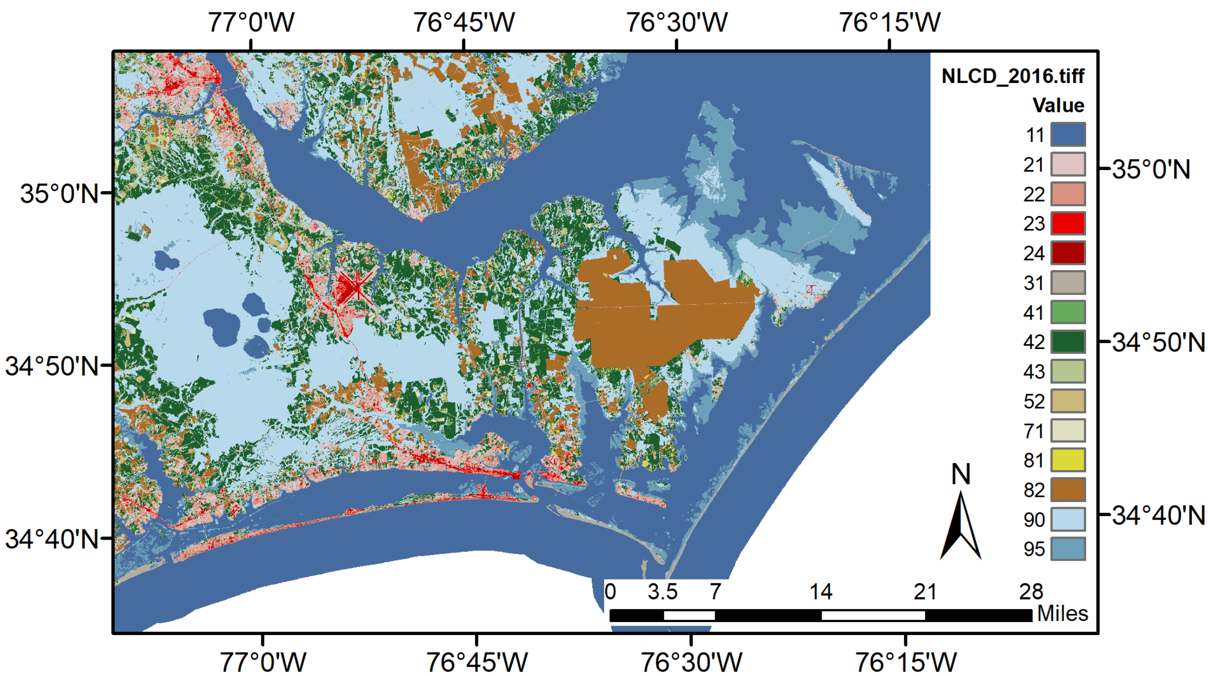

Figure 3.7 NLCD Land Cover Classifications. A portion of the USGS NLCD raster is pictured, where each color represents a different land cover classification and each number corresponds to a land cover type and a Manning's n friction coefficient (see Table 3.1).

$$S = \left(\frac{n * U}{k * R^{2/3}} \right)^2 \tag{3.4}$$

Using Equation 3.4, the slope of the energy grade line S can be interpreted as the friction slope h_L/L [Rubin et al., 2001], where h_L is head loss and L is the lateral distance traveled. Head loss between raster cells can be calculated using Equation 3.5:

$$h_L = L \left(\frac{n * U}{k * R^{2/3}} \right)^2 \tag{3.5}$$

in which L represents the lateral distance between the center of each cell, and h_L represents the vertical reduction in water elevation over distance L.

3.4.3 Calculating Accumulation of Head Loss

Equation 3.5 makes it possible to determine the head loss occurring over each individual raster cell for a given water velocity U and hydraulic radius R, by using each cell's assigned n and k with a direction-dependent L. However, there are two more issues that arise when attempting to apply this method to forecasting: (1) water velocities and hydraulic radii are unknown prior to receiving input from ADCIRC; and (2) cumulative head loss (i.e. the energy lost in comparison to static, horizontal extrapolation) must be calculated based on an accumulation of frictional losses over the flow paths, which are unknown. To overcome these issues, this research uses the `r.walk` module to pre-compute an energy surface that includes both gains in potential energy due to elevation changes in the DEM and head losses due to surface friction associated with land cover classifications.

The `r.walk` module creates a raster map showing the anisotropic cumulative cost of moving between different geographic locations [*r.walk*]. In this research, cost represents the total energy needed to reach a given location based on both elevation changes and head loss due to land cover (Figure 3.6). One way to pre-compute the cost surface would be to loop over each DEM raster cell, use `r.walk` to compute the costs to all cells at MSL, and then use the minimum cost at each raster cell. However, this method is both costly and redundant, as the costs will be very similar at neighboring cells.

Thus, instead of pre-computing the cost at every raster cell, we instead do a pre-computation at only an array of selected points, and then interpolate at all cells. For each point throughout the array, `r.walk` creates a raster surface representing the least amount of energy required to travel from MSL to the point. The least amount of energy, or "least cost," is determined by iterating over different possible paths and choosing each "least cost path" from MSL to the end point until a full raster of values is generated. Differences in cost between raster cells are calculated as follows:

$$\text{cost}_{\text{total}} = \text{cost}_{\text{movement}} + \lambda\, \text{cost}_{\text{friction}}\, \Delta S \tag{3.6}$$

in which λ is a constant multiplication factor, friction costs are the cost of moving laterally over each cell (Equation 3.5), and ΔS is lateral distance (L from Equation 3.5). Movement time cost is calculated as:

$$\text{cost}_{\text{movement}} = a\Delta S + b\Delta H_{\text{uphill}} + c\Delta H_{\text{downhill,moderate}} + d\Delta H_{\text{downhill,steep}} \tag{3.7}$$

in which ΔH is change in elevation according to the DEM, and a, b, c, and d are multiplication factors. Because the desired cost raster will be in units of energy head and will not consider time (as originally intended by `r.walk`), a, b, c, and d are set respectively to 0, 1, -1, and -1. Therefore, for each vertical unit in elevation gain or loss, there is a unit of head gain or loss in the form of potential energy. The coefficient a is set to 0 because units of head loss due to friction are accounted for in Equation 3.6. In this equation, movement time cost has now become potential energy cost (Δz), while $[\lambda\, \text{cost}_{\text{friction}}\, \Delta S]$ represents frictional head loss. Now, combining Equations 3.5 and 3.6, where λ is set to 1 and $\Delta S \equiv L$:

$$\text{cost}_{\text{total}} = \Delta z + \sum L \left(\frac{nU}{kR^{2/3}} \right)^2 \tag{3.8}$$

Because Δz is based on the given DEM and L is built-in to the `r.walk` calculations, the remaining portion of Equation 3.8 corresponds to the "$\text{cost}_{\text{friction}}$" raster from Equation 3.6. As previously stated, the value n is based on NLCD data corresponding to Manning's coefficient and k is a unit conversion coefficient, but U and R are unknown without input water velocities and elevations from ADCIRC. Therefore, to pre-compute accumulation of head losses in respect to flow paths, a constant ratio of $U/R^{2/3}$ is assumed throughout the pre-computational process to entrain flow paths based on both elevation and friction losses. This ratio, referred to as UR_{const}, can now be used to fill the missing piece of Equation 3.8, as shown below.

$$\text{cost}_{\text{friction}} = \left(\frac{nU}{kR^{2/3}} \right)^2 = \left(\frac{n(UR_{\text{const}})}{k} \right)^2 \tag{3.9}$$

Each `r.walk` operation generates a least cost raster from MSL to one end point. All least-cost paths within the raster represent the least cost heading in the direction of the end point; these paths will likely not represent the least cost when heading in the direction of another point. For this reason, `r.walk` is placed into an iterative function, which generates a least cost raster between MSL and each point within an array of points. A new raster is created for each end point and a final least cost surface is created where the smallest value from each raster cell amongst every raster is taken as the final cost (Algorithm 1). A sample of this iterative process is displayed in Figure 3.8.

Algorithm 1 Find least cost from MSL to an array of end points

```
Assign large value to FinalCost
for each end point in array of points do
    execute r.walk to generate walkRast
    for each cell in walkRast do
        if walkRast cell < corresponding FinalCost cell then
            overwrite FinalCost cell with walkRast cell
        end if
    end for
end for
```

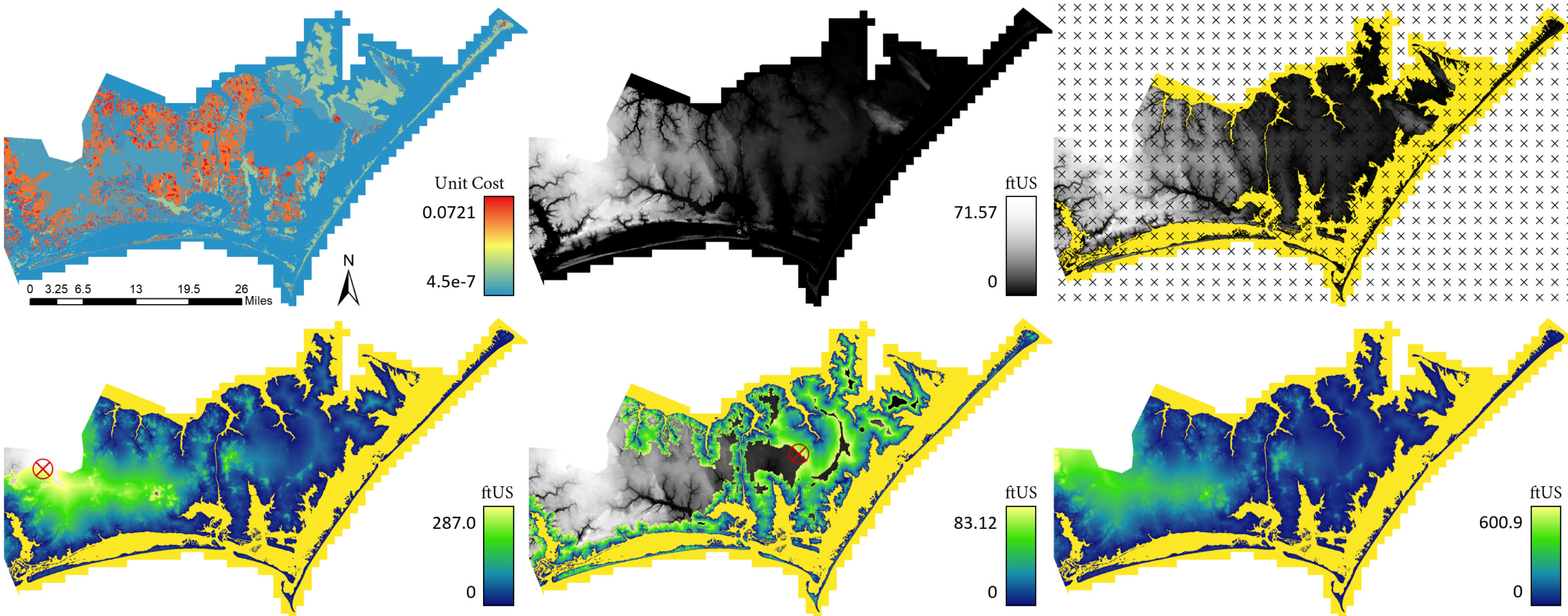

Figure 3.8 Sequence for pre-computation of head loss method. Figures above represent a sample of the process required for developing the energy cost surface for the head loss method. The unit head loss raster [top left] has values to represent the energy head lost per lateral unit traveled over each cell, calculated using Equation 3.9. For the DEM [top middle], a set of endpoints [top right] are used for each `r.walk` operation (Algorithm 1) with the MSL raster (yellow), which is the starting point. Examples of the `r.walk` operations [bottom left, bottom middle] for two different endpoints (marked with an encircled red X) are then used to pre-compute the total cost [bottom right] (Algorithm 1).

3.4.4 Forecasting with Friction Losses

A raster now exists that contains estimated total cost values based on changes in elevation and head loss due to friction based on an assumed constant ratio of $U/R^{2/3}$ (denoted as UR_{const}). To prepare this raster for forecasting, UR_{const} must be removed so the raster is able to replace the synthetic constant value with actual forecast data from ADCIRC. From Equations 3.8 and 3.9, UR_{const} can be removed by subtracting the DEM from the total cost raster and dividing by $(UR_{\text{const}})^2$, leaving only a raster referred to as "raw cost" in the form of:

$$\text{cost}_{\text{raw}} = \sum L \left(\frac{n}{k}\right)^2 \tag{3.10}$$

The raw cost raster is the cumulative head loss values based on flow paths, stripped of U and R. After U and R values are imported from ADCIRC, a raster calculation is executed to multiply these values into the head loss portion of Equation 3.8 and add the DEM.

Similarly to the static method, ADCIRC water elevations are extrapolated horizontally until reaching a corresponding raster surface. In the static method, this surface is the DEM, but in the head loss method, this surface is the total cost. Because one of the factors contributing to total head loss is the hydraulic radius, this radius must be calculated for each cell. In forecasting, the cumulative cost from MSL to each cell is calculated using the average radius R, based on the depth according to static extrapolation. For simplicity, the average R between MSL and the cell of interest is the average between the storm surge depth at MSL predicted by ADCIRC and the static depth at each cell of interest (Equation 3.11). This method does not account for variations in topography causing changes in R between the flow path's initial and final locations, but is the best method for estimating R throughout the flood path without the time-consuming process of recalculating flood paths.

$$R_{\text{avg}} = \frac{1}{2}\left((\zeta_{\text{ADCIRC}} - z_{\text{DEM}})_{\text{raster}} + (\zeta_{\text{ADCIRC}} - z_{\text{DEM}})_{\text{MSL}}\right) \tag{3.11}$$

in which ζ_{ADCIRC} are the water elevation predictions and z_{DEM} are the topographical elevations, each taken at a null cell of interest and the cell's nearest MSL location. However, the horizontally extrapolated ADCIRC values are the same at the MSL location and the null cell, and the DEM value at the MSL location is equal to zero by definition, and thus:

$$R_{\text{avg}} = \zeta_{\text{ADCIRC}} - \frac{1}{2} z_{\text{DEM,raster}} \tag{3.12}$$

For the velocity U, this quantity is a common output from ADCIRC. However, its predicted values cannot be trusted at the ADCIRC flooding extents, because of a no-normal flow condition at this "boundary." Thus, for this friction method, U is assumed as an arbitrary constant. This constant can be controlled as a uniform value throughout the domain by multiplying the raw cost raster by an assumed U^2 value, or can be added manually as similar multipliers to the pre-computed cost raster.

Flood extents of maximum water elevations predicted by ADCIRC do not align typically with MSL, which is the starting point for all head loss accumulation paths. ADCIRC flood predictions almost always extend beyond the lateral coverage of MSL, meaning ADCIRC is already taking into account some energy losses throughout the overland flooding process (Figure 3.9c). Because water elevations are extrapolated horizontally from the extents of the ADCIRC prediction using `r.grow`, the cost overcome by ADCIRC predictions within the given extent is also extrapolated simultaneously (Figure 3.9f). This creates two rasters: one with extrapolated ADCIRC water elevations and another with costs overcome by ADCIRC corresponding to the predicted water elevations.

Extrapolating cost overcome by ADCIRC water elevation predictions is necessary because now ADCIRC elevations are able to be compared in terms of the energy cost surface, rather than just topographical elevation. For instance, ADCIRC may predict a water elevation of 6 ft over a topography with a 4 ft elevation, but has already overcome 3 ft of energy losses. Comparing the water elevation to the topography, this "wall of water" will continue to flood outward. However, comparing the water elevation to the energy cost surface, the water elevation is already below the energy cost (7 ft). By extrapolating the cost overcome by ADCIRC and adding this to the predicted water elevation, the water elevations are able to be compared in terms of the energy cost.

Rather than extrapolating the total cost raster values already accounted for by ADCIRC, raw cost values are extrapolated. Sometimes the inundation depth (water elevation minus DEM) from ADCIRC is very small, which makes the total cost very high. This means that where inundation depths (approximately equal to hydraulic radii, R) are small, the cost overcome by ADCIRC is perceived to be high and adds this total cost value to the inundation depth after extrapolation, which leads to instabilities. The raw cost raster is used because this raster does not account for depths until multiplied back into the total cost equation; extrapolation can occur before introducing the hydraulic radius into the head loss equation, thus avoiding instabilities.

In summary, a version of Manning's equation (Equation 3.5) has been introduced into the ADCIRC water elevation downscaling method to account for energy head loss due to land cover friction in the water elevation extrapolation process. Rather than extrapolating water elevations to a DEM surface, they are extrapolated to an energy surface. Flow paths accounting for accumulation of head loss are pre-computed using the `r.walk` module in GRASS and made into a surface, stripped of its synthetic hydraulic radius (R). When maximum water elevation forecasts are received from ADCIRC, average hydraulic radii (R_{avg}) are calculated and used to create an energy cost surface over which water elevations are extrapolated. Now a downscaled water elevation surface is produced that uses energy head losses due to friction as overland flooding due to storm surge occurs.

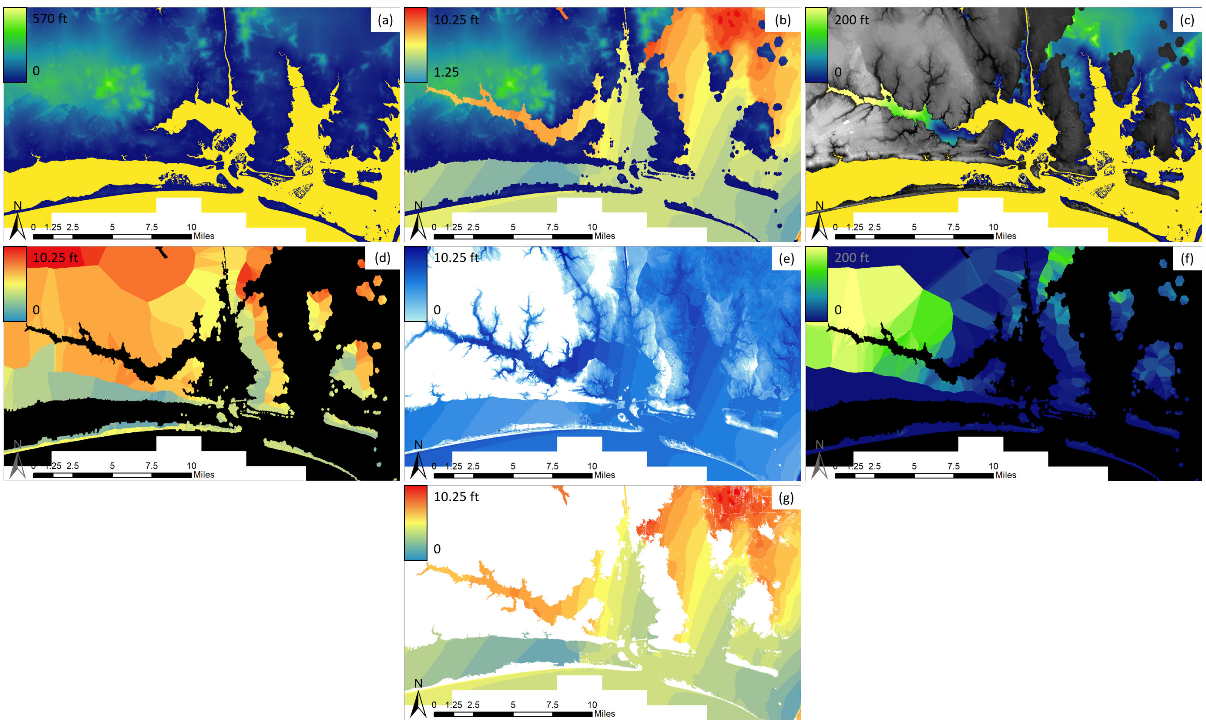

Figure 3.9 Forecasting with the head loss method. Intermediate steps for producing a downscaled water level forecast using the head loss method, ordered as steps a-g. Panel [a] shows the pre-computed data, where the viridis (blue-to-green scale) raster is the raw cost (Equation 3.10) and the yellow raster is MSL. Next, the maximum water elevation results from the NC9 ADCIRC model are processed by Kalpana and shown in panel [b]. The raw cost accounted for by ADCIRC is made into its own raster in panel [c] for use later in the code. In panel [d], the water elevations are extrapolated using `r.grow`; these values are used along with Equation 3.12 to determine R_{avg}. Black rasters are used in panels [d] and [f] to represent the extents of the NC9 model output. The extent of R_{avg} is limited to the extents of the static method because, beyond the point where the water elevations intersect the DEM, R_{avg} becomes negative. To account for raw cost from [c] in downscaling, these values are extrapolated to form [f]. Finally, [f] is subtracted from [a] and multiplied by [e], then added to the DEM to form the final cost surface. Output from the head loss method is shown in panel [g].

3.5 Evaluation with a High-Resolution ADCIRC Model

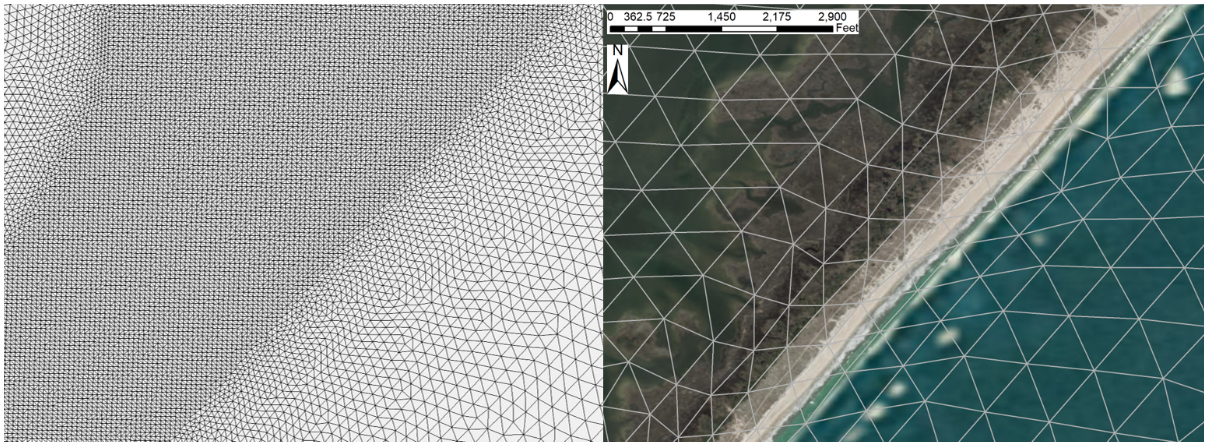

Figure 3.10 High-Resolution Mesh vs. NC9. The high-resolution mesh developed is shown on the left and the NC9 mesh is on the right. Both are shown at the same zoom level over a barrier island along the Cape Lookout National Seashore in Carteret County. Along the overland portion, the high-resolution mesh vertices are aligned with the 50 ft resolution rasters used for downscaling. Moving away from the overland portion, spacing between mesh vertices becomes larger until eventually merging with the NC9 mesh. It is visually evident that the NC9 mesh (right) has a much lower resolution than the high-resolution mesh.

To verify the accuracy of these downscaling methods, there must be accurate data to cross-reference the forecasted inundated area. The reliability of the downscaling methods is dependent on the accuracy of ADCIRC output, which is the most important input to the downscaling methods. A high-resolution ADCIRC hindcast was chosen to represent the "truth" in this research, because ADCIRC accounts for physics in overland flooding and there are not enough supporting reliable data sources within the regime of downscaled flood extent forecasts.

In Tull [2018], a high-resolution ADCIRC mesh was developed for Carteret County, NC. The high-resolution mesh was developed using the NC9 mesh and has vertices aligning with each raster cell in a 50-ft resolution DEM covering the extents of Carteret County. Aligning vertices with square raster cells causes the mesh to become structured and misaligned with the rest of the NC9 mesh, which is unstructured. However, mesh generation software automatically interpolates between these sections of mesh to account for the differences in structure patterns. The high-resolution mesh contains a total of 6,772,170 vertices and 13,528,879 elements, which makes using this mesh implausible in forecasting.

In comparison, the NC9 mesh is commonly used for forecasting and only contains 622,946 vertices.

The accuracy of methods presented in this research is evaluated for Hurricane Florence (2018). This hurricane was chosen because it made landfall close to Carteret County as a Category-1 hurricane and generated storm surge that led to large amounts of overland flooding and property loss. According to NOAA, the total damage estimates due to Hurricane Florence were around $24 Billion [Smith et al., 2020].

To keep consistency among the high-resolution ADCIRC predictions and the downscaled predictions from the NC9 mesh, ADCIRC simulations were run for Hurricane Florence with each mesh. All input parameters were kept the same between the two ADCIRC simulations, and mesh data were interpolated spatially to remain consistent. Therefore, output from the NC9 mesh simulation was able to be used effectively as input for each of the three downscaling methods. Results in Chapter 4 compare high-resolution ADCIRC output to downscaled low-resolution ADCIRC output, both from Hurricane Florence.

CHAPTER

4

RESULTS AND DISCUSSION

4.1 Hurricane Florence

Hurricane Florence (2018) was a powerful storm. It reached Category-4 strength on the Saffir-Simpson Hurricane Wind Scale as it moved across the Atlantic Ocean, and it made landfall as a Category-1 hurricane in southeastern North Carolina (near Wrightsville Beach) with 80-kt wind speeds at 1115 UTC on September 14. At and after landfall, it was a slow-moving storm with a 5-kt westward motion, and its rainfall totals exceeded 30 in in some areas. Florence was downgraded to a tropical storm by 0000 UTC on September 15 while the storm was located just north of Myrtle Beach, South Carolina. From there, the storm moved inland and continued losing energy while traveling north until finally dissipating over Massachusetts shortly after 1200 UTC on September 18 [Stewart et al., 2019].

Maximum storm surge heights were estimated in Stewart et al. [2019] to reach 8-11 ft above ground level in North Carolina along the Neuse River and its tributaries. The Neuse River, which empties into the Pamlico Sound, is located along the northern edge of the main area of focus for this research (Carteret County). USGS sensors deployed in downtown New Bern reached 10.08 ft above NAVD88 (North American Vertical Datum of 1988); a post-storm simulation suggests that maximum inundation reached 11 ft upstream of downtown New Bern. A USGS sensor installed in a wave-protected area of Emerald Isle on Bogue Sound (within the study area) measured a peak water level of 7.76 ft above NAVD88 [Stewart et al., 2019].

4.2 ADCIRC Predictions

4.2.1 Simulation Parameters

ADCIRC was run using two different meshes: (1) NC9, which was used as the base mesh and whose results provided input for the downscaling methods, and (2) the high-resolution mesh with an improved representation of Carteret County, as discussed in Section 3.5.A simulation without winds was completed for 15 days prior to the storm (August 22 to September 7, 2018) to allow the model to generate tides; this process is referred to as a “cold start.” The coastal ocean conditions from the end of this first simulation were then passed into a follow-on ADCIRC+SWAN simulation of the storm, with meteorological wind and pressure data for the duration of the storm, lasting 11.5 days from 0000 EST on September 7 to 1200 EST on September 18.

4.2.2 ADCIRC performance

ADCIRC performed well in Carteret County for this study. Peak water levels measured at the NOAA Beaufort tide gauge reached 5.15 ft, while water levels predicted by ADCIRC at the gauge were 4.37 ft for NC9 and 4.32 ft for the high-resolution mesh. Figure 4.1 presents time series comparisons of water levels predicted at the Beaufort gauge for the NC9 mesh, high-resolution mesh, and measured data. ADCIRC under-estimated water elevations at the Beaufort gauge in comparison to the measured value, but the difference is less than 1 ft, and the time series of the ADCIRC simulations were nearly identical to one another.

4.3 Validation of Downscaling Techniques

4.3.1 Visual Validation

Output from each of the downscaling methods can be compared to predictions from the high-resolution mesh, which are considered as “truth.” In summary, the static and slopes methods performed similarly, while the head loss method was significantly different and the best match to the true flooding extents.

Because the water surface produced by ADCIRC was relatively flat, downscaling using slopes of this surface did not differ much compared to the static method. The effectiveness

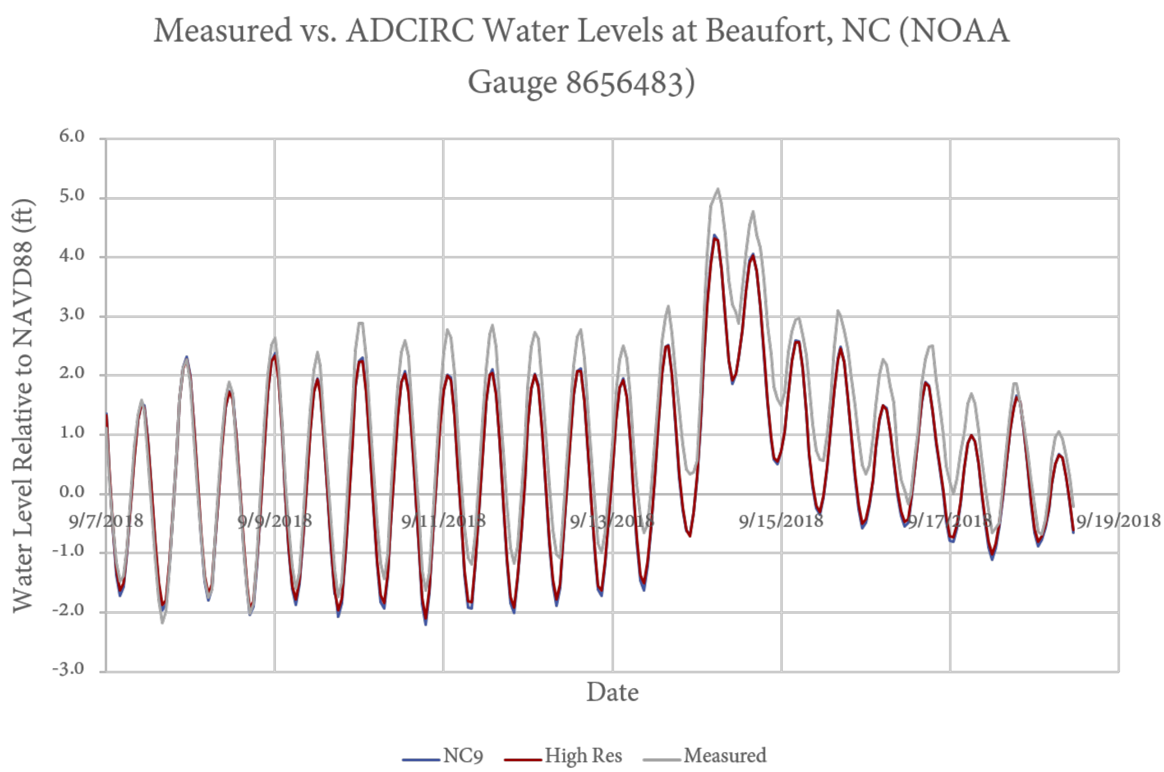

Figure 4.1 Time Series Comparisons. Water levels at the Duke Marine Lab in Beaufort, NC are shown for each ADCIRC simulation and measured values throughout the duration of Hurricane Florence. Measured values [grey] show slightly higher water levels than those produced by the NC9 mesh [indigo] and high-resolution mesh [red] ADCIRC simulations. The time series is shown for the duration of ADCIRC runs using meteorological wind and pressure data (i.e. excluding the "cold start").

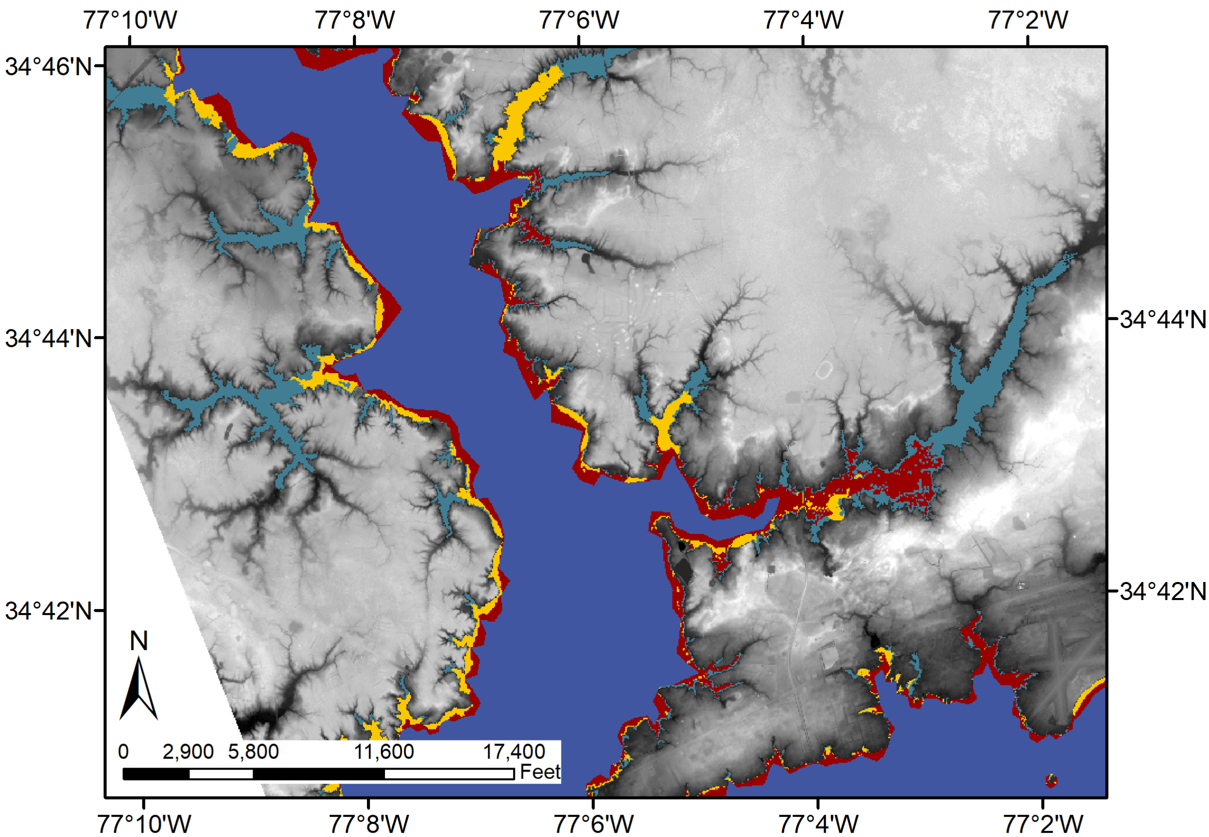

Figure 4.2 Example of flooding extents from the ADCIRC and the downscaling methods. Rasters are displayed representing the output for each model/method discussed in this research. From the top layer to the bottom layer, the rasters are outputs from NC9 (indigo), the high-resolution mesh (red), the head loss method (yellow), and the static method (blue). The slopes method raster is not shown because the results did not differ enough from the static method. All rasters shown overlay the DEM used for downscaling (grayscale).

of the slopes method can be tuned by using the slope exaggeration parameter (set to unity for the purpose of this study), which works as a multiplier to the addition or subtraction of total water elevation from the extrapolated surface. Going back to Equation 3.1, the slope exaggeration term c can be used to increase or decrease the slope's effect on the downscaled water elevation. This tuning was not explored in this study, but it can be considered in future work.

The static method tends to over-estimate the flood extents (Figure 4.2). Because the head loss method operates similarly to the static method but uses an energy cost surface (DEM plus head loss) rather than only a DEM, it is impossible for flood extents predicted by the head loss method to exceed those of the static method. Therefore, unless the head loss method under-estimates the truth, this method will outperform the static method.

There are some areas, as shown in Figure 4.3 near the Newport River, where the head loss method under-estimates flooding. Areas surrounding this river were difficult to model using the head loss method because, during the pre-computational steps where flow paths are traced from MSL, a large portion of the river was not recognized as a perennial bodies of water, because the resolution of the DEM raster exceeded the typical stream width. This led to inaccuracies because higher costs were perceived to have been required along the river than would have been realistic if the stream had been properly identified. This issue can be fixed either using a higher resolution raster or manually delimiting perennial water sources.

Another area where the methods performed unfavorably was the low-lying farmland region in eastern Carteret County (between Merrimon, Sealevel, and Smyrna, NC). As shown in Figure 4.4, the simulation using NC9 predicts most of this area to be flooded, while simulation using the high-resolution mesh does not. The downscaling methods only make changes to the water levels *within* the extents of the NC9 forecast if the DEM exceeds the forecasted water elevation, so this leads to a major discrepancy in overall flooded cells between the truth and each downscaling method. For this reason, the main focus of water level comparisons is outside the extents of the NC9 forecast.

An area that exemplifies the benefits of the head loss method is the area surrounding the White Oak River (Figure 4.5). The White Oak River is a much wider river than the Newport River and can easily be delineated automatically without increasing the raster resolution. This allows the head loss method to appropriately calculate accumulation of head loss along flow paths and develop a more accurate cost surface. As shown in Figure 4.5, the head loss method was able to reduce the flood extents of the static method and produce results more similar to those reached by the high-resolution ADCIRC model.

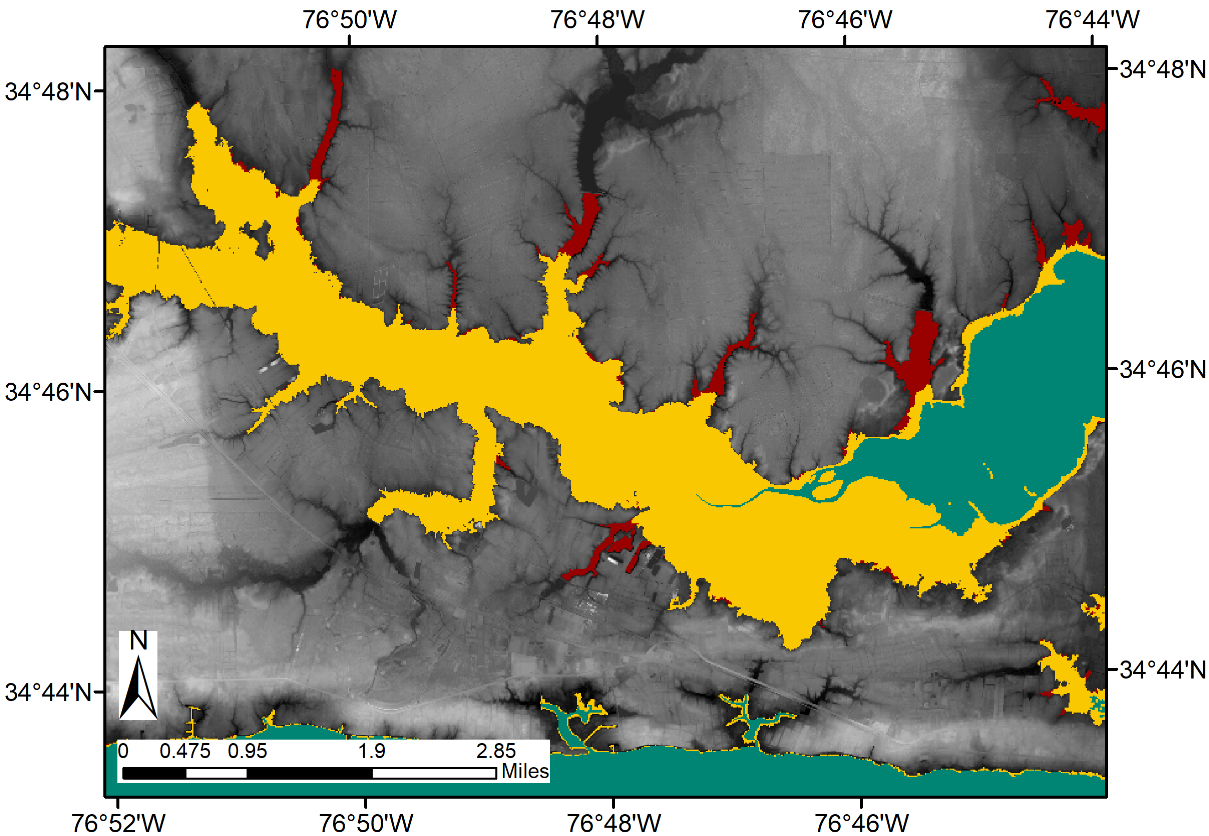

Figure 4.3 Example of underestimation of flooding extents with the head loss method. Listing from the top layer to the bottom layer, the rasters are MSL (blue-green, used as the initial points for the `r.walk` head loss accumulation calculations), the head loss method (yellow), and the high-resolution mesh (red).

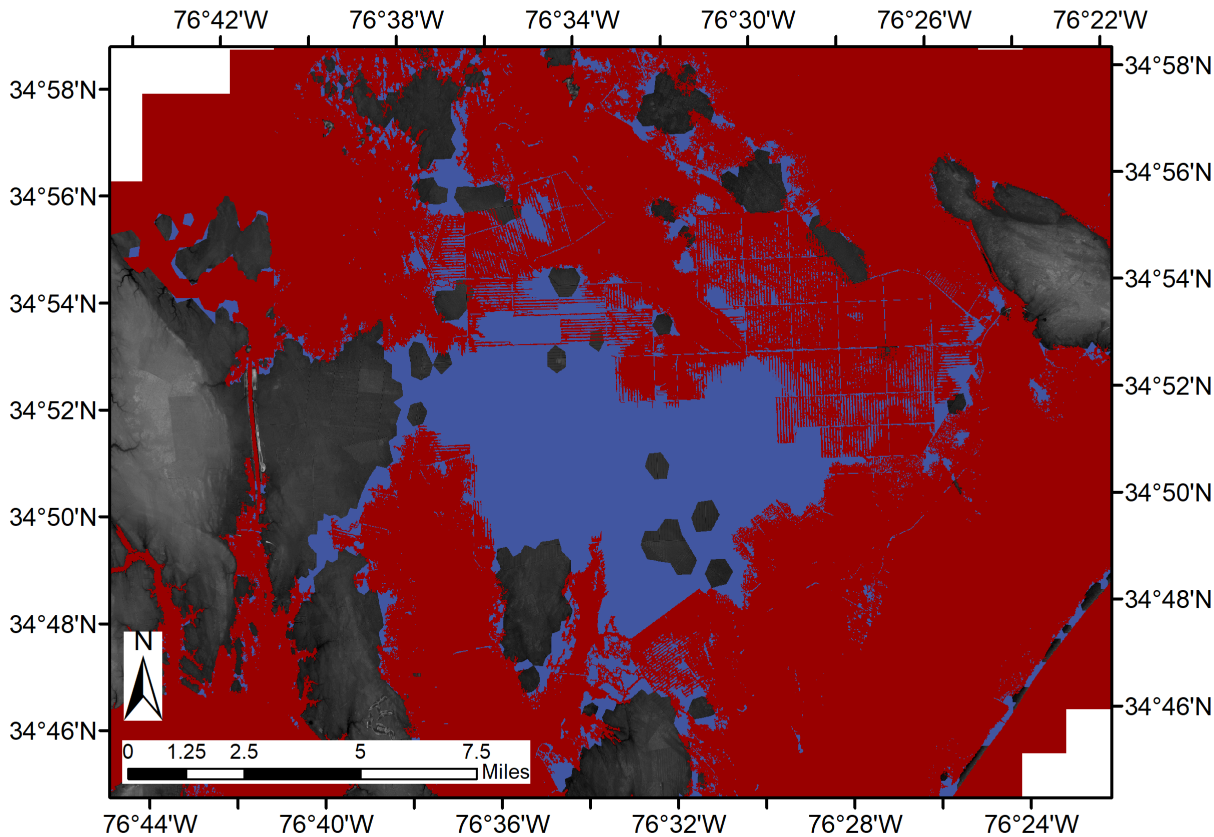

Figure 4.4 Example of over-estimation of flooding extents in a simulation on the NC9 mesh. ADCIRC maximum water elevation results from simulations with the NC9 (indigo, bottom layer) and high-resolution (red, top layer) meshes are shown throughout a low-lying farmland region in eastern Carteret County. This image highlights the magnitude to which the NC9 mesh over-estimates flood extents in this region, whereas the NC9 mesh would typically under-estimate flood extents in other regions.

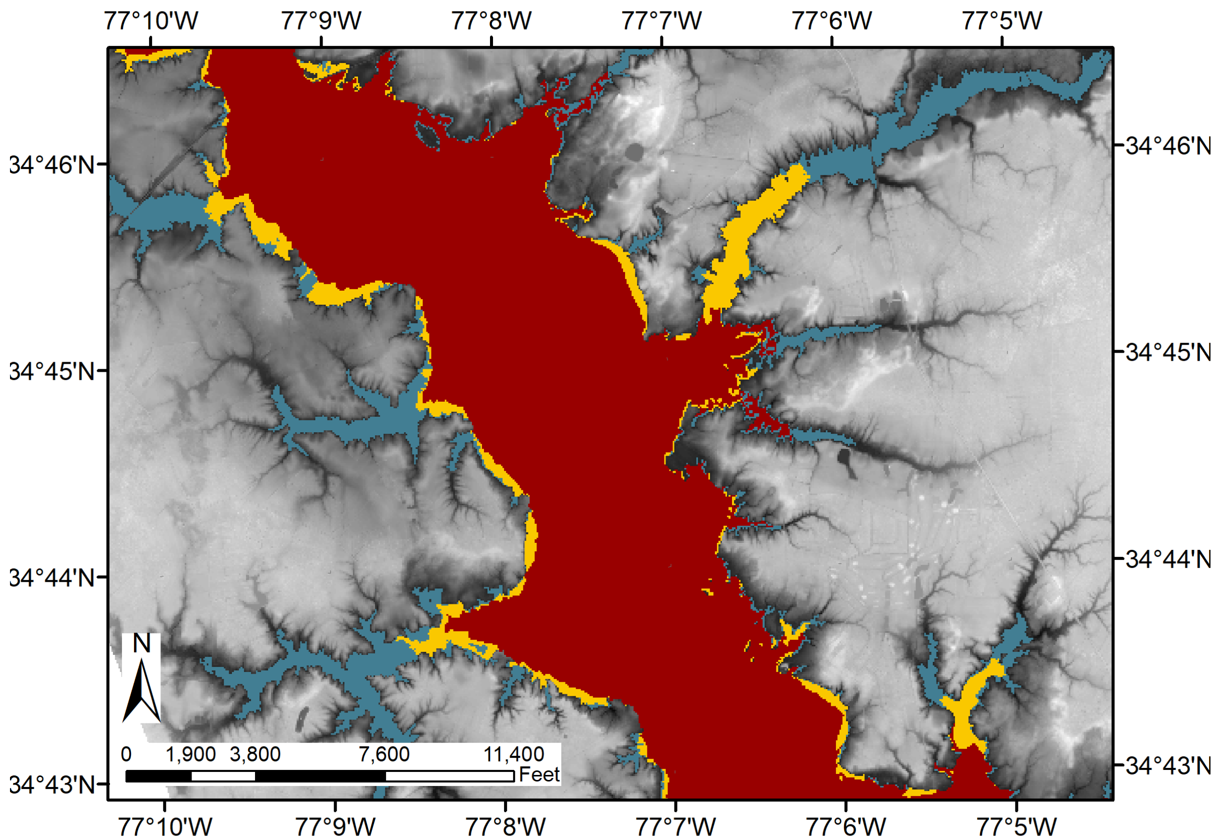

Figure 4.5 Example of good performance using the head loss method. The head loss method (yellow) had good performance in the areas surrounding the White Oak River in comparison to the static method (blue), with the high-resolution ADCIRC model (red) representing the truth. This image shows that the flood extents from the head loss method matched closely with the flood extents of the high-resolution model and outperformed the static method.

4.3.2 Quantitative Validation

Each method can be validated quantitatively by comparing different flood extent metrics. The flooding extents from simulations using the NC9 and high-resolution meshes are compared with the downscaled flooding extents from the static, slopes, and head loss methods. Each flooding extent is converted to raster format and then compared based on numbers of flooded cells. The total number of flooded cells, or raster cells occupied by water that have an elevation greater than zero (MSL), is quantified for all five models and methods. Next, this same metric is used only for cells outside the extent of NC9. As discussed in the previous subsection, this is done to eliminate errors made within the extents of NC9 that are not accounted for in the downscaling methods. The final two metrics compare over- and under-estimations, both outside the extents of NC9. Over-estimations are cells flooded by a downscaling method but not the high-resolution mesh, while under-estimations are cells flooded by the high-resolution mesh but not a downscaling method. These results are displayed in Table 4.1.

Table 4.1 Comparisons of flooding extents from simulations (NC9 and high-resolution meshes) and downscaling methods (static, slopes, and head loss). All values represent cell counts for each raster in Carteret County, NC.

Mesh	Downscaling Method	Flooded	Flooded (outside NC9)	Over-Estimation (outside NC9)	Under-Estimation (outside NC9)
NC9	—	2,741,038	—	—	—
NC9	Static	3,035,320	406,403	243,747	1,370
NC9	Slopes	3,055,440	418,285	255,341	1,083
NC9	Head Loss	2,832,769	204,360	97,101	56,767
High Resolution	—	2,205,760	164,026	—	—

Data from Table 4.1 show that the slopes method performed poorest, producing more flooded cells outside NC9 (418,285) than the already over-estimated static method (406,403). The head loss method performed best, predicting 204,360 flooded cells outside the extents of NC9 compared to the truth of 164,026 cells. Looking at the "Flooded" column, it is evident that the overall performance of each downscaling method is largely dependent upon the performance of the NC9 mesh. Due primarily to the topographically-low farmland region in eastern Carteret County (Figure 4.4), the NC9 simulation over-predicted the total number of flooded cells with a predicted 2,741,038 compared to the high-resolution results of 2,205,760. Although the NC9 simulation mostly under-estimated flood extents elsewhere besides the farmland region, this led to an over-estimation that subsequently led to over-estimations in the total number of flooded cells for each downscaling method.

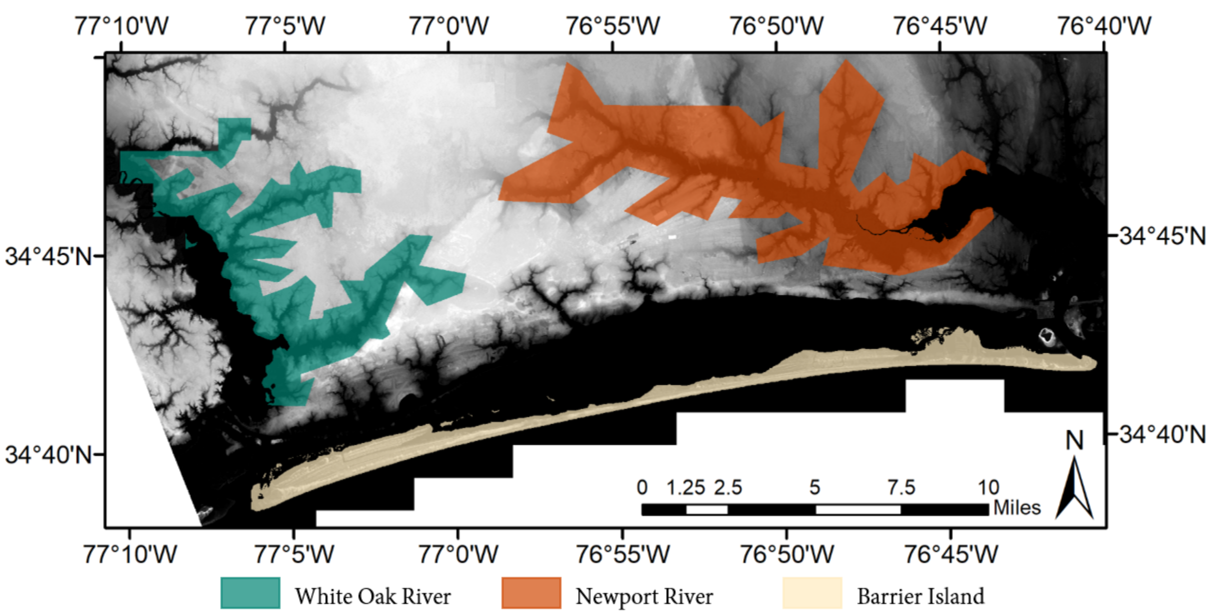

Figure 4.6 Regional performance comparison. The three regions used for comparing downscaling method performance are the White Oak River region (blue-green), the Newport River region (orange), and the Barrier Island region (tan).

To further illustrate regional differences in downscaling method performance, three separate regions were analyzed individually (Figure 4.6). These regions were chosen because each illustrates a different downscaling environment. The White Oak River region contains a large river system with a large river width and well-defined tributary streams. The Newport River region is still a large river system, but the river is not as wide and its tributary streams

Table 4.2 Regional cell count comparisons between the high-resolution mesh simulation results and the head loss and static downscaling method results.

Mesh	*Barrier Island Region* Downscaling Method	Flooded	Flooded (outside NC9)
NC9	Static	39,196	20,972
NC9	Head Loss	30,988	12,912
High Resolution	—	27,048	11,248

Mesh	*Newport River Region* Downscaling Method	Flooded	Flooded (outside NC9)
NC9	Static	97,834	31,730
NC9	Head Loss	83,434	17,330
High Resolution	—	82,745	18,167

Mesh	*White Oak River Region* Downscaling Method	Flooded	Flooded (outside NC9)
NC9	Static	42,893	28,006
NC9	Head Loss	24,892	10,009
High Resolution	—	26,610	12,134

are slightly smaller. Finally, the barrier island region relies on shorter extrapolation distances but contains more hard structures that create sea walls, canals, and other barriers.

As shown in Table 4.2, the head loss method outperformed the static method in each region. For example, within the White Oak River region, the head loss method produced 10,009 flooded cells outside the extents of the NC9 model output compared to 12,134 from the high-resolution model, while the static method produced 28,006 cells. Results from Table 4.2 also show that the head loss method tends to under-predict flood extents in riverine environments. This is likely due to the fact that, within smaller tributary stream areas, flood extents generated by ADCIRC extend further away from MSL than they do in other regions. Therefore, the cost accounted for by ADCIRC output (Figure 3.9c) is much higher and can lead to inaccuracies as these costs are taken into account in downscaling. These under-predictions can be managed by either manually delineating perennial water sources to reduce the cost accounted for by ADCIRC output or reducing the friction values associated with these tributary stream areas.

4.4 Downscaling Method Parameter Flexibility

The new methods proposed in this research (slopes and head loss) are flexible, and their parameters can be tuned. As mentioned previously, the slopes method can be tuned by changing the slope exaggeration factor (c in Equation 3.1) to increase or decrease the effect of the water surface slope on the extrapolated water elevations. If necessary, the slopes method can be applied in conjunction with the head loss method.

The head loss method allows for the most flexibility. With many of the raster operations occurring as pre-computational steps, computational time is not an important factor, and parameters can be optimized. The delineation of important small-scale perennial water sources (such as the Newport River) can eliminate the need to increase resolution to capture these features. Similarly, small-scale hard structures such as sea walls or levees that are not detected within the resolution of the DEM can be added. Most importantly, users can tune friction values to more accurately match high-resolution ADCIRC output or measured results based on historical storm data or synthetic flood simulations. For instance, if an area is being over-estimated, the user can increase the friction in the area until the true flood extent is achieved. These methods also allow for the possibility of applying machine learning techniques or statistical techniques such as those presented in Mayo et al. [2014].

A major uncertainty in the head loss method is the selection of Manning's n values for each land cover type. This is a common research topic for which many sensitivity studies are performed [Kalyanapu et al., 2009; Liu et al., 2018; Medeiros et al., 2012; Passeri et al., 2011]. As described in Passeri et al. [2011], each land cover type may have a range of acceptable n values, which can affect the results of a storm surge model. Modeling storm surge using ADCIRC for a synthetic storm, the difference in maximum water elevations between a simulation run using the highest and lowest acceptable n values was 1.92 m. Due to the uncertainty in Manning's n, the head loss method would be best optimized by adjusting pre-computed friction values to match high-resolution ADCIRC output or measured values.

Another way to tune the pre-computational process of the head loss method is to alter the UR_{const} parameter from Equation 3.9. Increasing UR_{const} will increase the effect of land cover variations in determining flow paths, while decreasing UR_{const} will increase the effect of elevation changes in determining flow paths. Finally, if the head loss method is universally over- or under-predicting flood extents, friction can be universally increased or decreased by applying a simple constant multiplication factor.

In summary, some of the possible methods for tuning the two downscaling methods (slopes and head loss) include, but are not limited to:

- Alter the slope exaggeration factor

- Use the slopes and head loss methods together
- Manually delineate small-scale water features and flow-impeding hard structures
- Manually increase or decrease raster cell friction in areas of over- or under-estimated flood extents
- Use machine learning or statistical methods to determine ideal friction values
- Manipulate UR_{const} from Equation 3.9 to dictate flow paths in pre-computational steps
- Use a constant friction multiplication factor (or change the constant velocity U value) to increase or decrease friction universally throughout the domain

CHAPTER

5

CONCLUSIONS AND FUTURE WORK

Accurate, timely predictions of storm surge and flooding are critical to emergency managers in the event of an approaching storm. To provide sufficient accuracy, numerical modelers often use computationally-heavy, parallelized models, thus requiring a balance of computational time with model resolution. These models are not able to simulate flooding at the finest scales of available DEMs, which can resolve small-scale infrastructure such as roads, buildings, and neighborhoods of interest to emergency managers. This study presents GIS-based techniques for downscaling (or increasing the resolution of) ADCIRC storm surge predictions in real-time forecasting. Building upon work by Tull [2018], this research improves the downscaling method's applicability, provides two new methods for adding simple physics to downscaling (Figure 5.1), and validates each downscaling method using a high-resolution ADCIRC model. These techniques can provide emergency managers with accurate forecast results at high resolution in a reasonable time for decision-making during an impending storm event.

The major conclusions of this research are summarized below:

1. *Integrating the downscaling methods with Kalpana allows users to apply the methods throughout the world, using any mesh or DEM without requiring parallelization.* Methods presented in Tull [2018] used pre-computed IDWs to interpolate ADCIRC maximum water elevation data from the NC9 mesh onto the 50-ft resolution DEM,

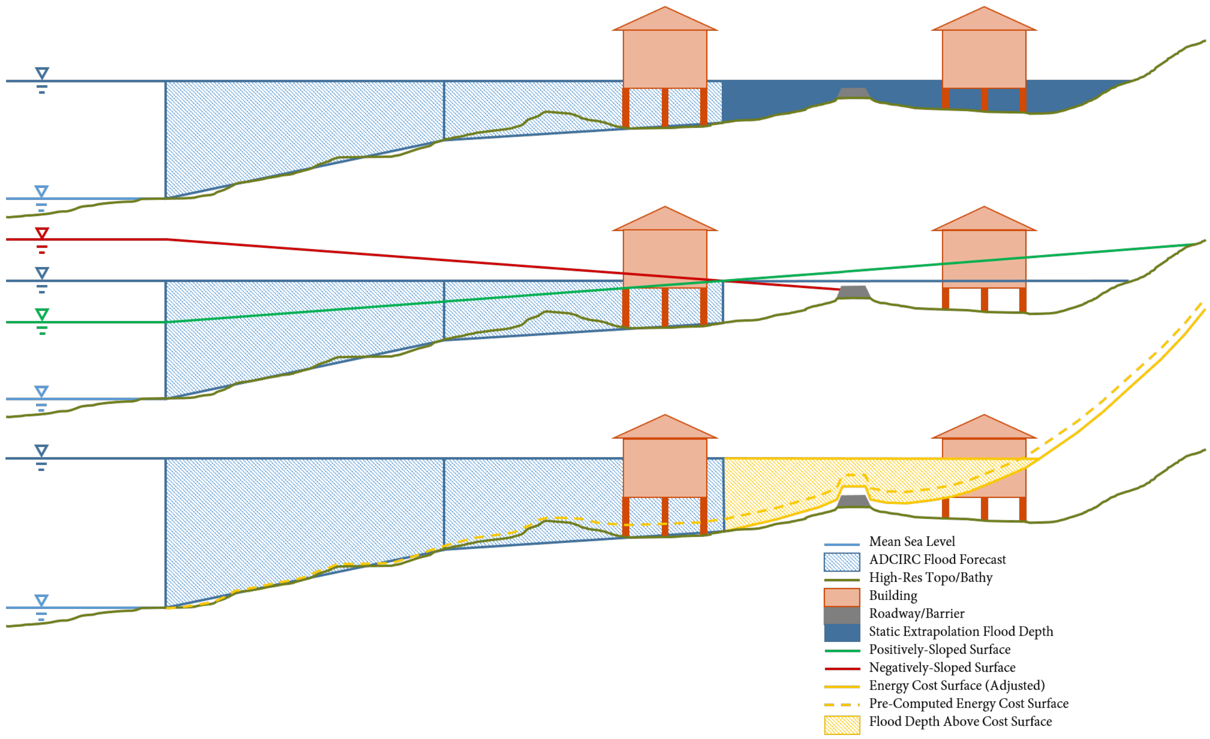

Figure 5.1 Schematic of all downscaling methods. Each method is shown: static (top), slopes (middle), and head loss (bottom).

located in a GRASS location using the NC state plane projection. The IDWs were specific to each mesh, DEM, and region and had to be created for any new combination of the three. Additionally, the interpolation of ADCIRC data to a raster using IDWs was the most computationally costly step of the methods presented in Tull [2018] and required users to parallelize the downscaling process.

Now, Kalpana allows users to convert ADCIRC data from any mesh within any region throughout the world into a shapefile without IDWs. Using Kalpana, ADCIRC water elevation data can be converted to shapefile format, which is then converted to raster format using GRASS. This allows users to surpass the IDW process and apply downscaling methods anywhere in the world, using any ADCIRC mesh and any DEM at any resolution without needing parallelization. A version of Kalpana can be downloaded online, complete with added modules for downscaling and a user-friendly interface for creating a GRASS location with data required for downscaling.

2. *The static method over-predicts water level extents.* Outside the extents of the NC9 water elevation forecast for Hurricane Florence (2018), the static method predicted 406,403 cells to be flooded throughout Carteret County, while an ADCIRC simulation

on the high-resolution mesh (taken as the truth) predicted 164,026 cells to be flooded. Although the method over-predicts flood extents, it can still provide useful information to emergency managers. The static method typically causes over-estimated flood extents and can lead to issues with hydraulic connectivity but, with a minimal amount of physics, the method is also the least costly in computational time.

3. *Downscaling using the slopes method did not improve the downscaling simulations*The water elevation surface slopes were small, so extrapolating while using these slopes did not affect the downscaled water elevations. In relation to the static method, which already over-predicts flood extents with a total of 406,403 cells flooded outside the extents of the NC9 results, the slopes method added to this over-prediction with 418,285 cells flooded. The slopes method also has a significantly higher computational cost compared to the static method; forecasting over large domains would require parallelization.

4. *The head loss method performed best and allows for the most flexibility.* Outside the extent of the NC9 model results, the head loss method had a total of 204,360 flooded cells while the high-resolution model had 164,026 flooded cells. The head loss method extrapolates water levels to an energy cost surface, consisting of energy head due to changes in elevation, added to head loss due to friction from land cover. Because the DEM surface used for extrapolation in the static method only contains elevation data and does not account for head loss, the energy cost surface must always have values greater than those within the DEM. Therefore, unless the head loss method is under-estimating water level extents, this method will always out-perform the static method in terms of total flooded cell count. Most of the computations for creating the energy cost surface are done prior to receiving input from ADCIRC models, so, because time is not a limiting factor in the pre-computational process, the head loss method allows for the most flexibility.

This study focused on developing methods for improving the accuracy of real-time ADCIRC storm surge downscaling methods. Results from each method were compared to results from an ADCIRC simulation using a high-resolution mesh in Carteret County, NC, for Hurricane Florence (2018). These downscaling methods were run without altering any of the available parameters (i.e. no slope exaggeration, no changes to the pre-computed rasters, and UR_{const} and U were set to 1 with no constant friction multiplication factor). These parameters can be optimized by analyzing downscaling method results for different regions using different storms and ADCIRC results.

Further research may involve running series of storm simulations using high-resolution ADCIRC models (or using another source as truth) and tuning parameters to best match the desired results. This can be done using ADCIRC results from both historical storms and synthetic storms. In addition to changing parameters manually, parameters and pre-computed rasters can be manipulated using machine learning and/or statistical methods.

The accuracy of storm surge simulations will continue to improve, as storm surge will continue to be critical to emergency managers for decision-making before, during, and after storms. The methods presented in this research will provide important tools for emergency managers evaluating storm surge predictions at small scales. With continued development toward improving the accuracy and speed of storm surge models, these will become a standard practice in storm preparation.

BIBLIOGRAPHY

Aerts, J C J H et al. (2013). "Low-Probability Flood Risk Modeling for New York City". *Risk Analysis* **33**, pp. 772–778.

Bates, P D et al. (2000). "A simple raster-based model for flood inundation simulation". *Journal of Hydrology* **236**, pp. 54–77.

Bates, P D et al. (2005). "Simplified two-dimensional numerical modelling of coastal flooding and example applications". *Coastal Engineering* **52**, pp. 793–810.

Beven, J (2005). *Tropical Cyclone Report, Hurricane Dennis, 4 - 13 July 2005*. Tech. rep. National Hurricane Center.

Blanton, B O et al. (2012). "Urgent Computing of Storm Surge for North Carolina's Coast". *Proceedings of the International Conference on Computational Science*. Vol. 9, pp. 1677–1686.

Blanton, B.O. et al. (2008). *North Carolina Coastal Flood Analysis System: Model Grid Generation*. Tech. rep. TR-08-05. Renaissance Computing Institute.

— (2010). *North Carolina Floodplain Mapping Program, Coastal Flood Insurance Study: Water Level Validation Study*. Tech. rep. TR-10-06. Renaissance Computing Institute.

Brown, I (2006). "Modelling future landscape change on coastal floodplains using a rule-based GIS". *Environmental Modelling and Software* **21**, pp. 1479–1490.

Brunner, G W (2016a). *HEC-RAS 2D Modeling User's Manual, Version 5.0*. U.S. Army Corps of Engineers. URL: https://www.hec.usace.army.mil/software/hec-ras/documentation/HEC-RAS%205.0%202D%20Modeling%20Users%20Manual.pdf.

— (2016b). *HEC-RAS User's Manual, Version 5.0*. U.S. Army Corps of Engineers. URL: https://www.hec.usace.army.mil/software/hec-ras/documentation/HEC-RAS%205.0%20Users%20Manual.pdf.

Bunya, S. et al. (2009). "A High-Resolution Coupled Riverine Flow, Tide, Wind, Wind Wave, and Storm Surge Model for Southern Louisiana and Mississippi. Part I: Model Development and Validation". *Monthly Weather Review* **138**.2, pp. 345–377.

Consortium, Multi-Resolution Land Characteristics. *MRLC Viewer*. URL: http://https://www.mrlc.gov/viewer/ (visited on 02/15/2020).

Cyriac, R. et al. (2018). "Variability in Coastal Flooding predictions due to forecast errors during Hurricane Arthur". *Coastal Engineering* **137**, pp. 59–78.

Dean, R G et al. (1984). *Water Wave Mechanics for Engineers and Scientists*. Englewood Cliffs, NJ: Prentice Hall, pp. 157–158.

Dietrich, J C et al. (2013). “Real-time forecasting and visualization of hurricane waves and storm surge using SWAN+ADCIRC and FigureGen”. *Computational Challenges in the Geosciences*. Ed. by Dawson, C N et al., pp. 49–70.

Dietrich, J.C. et al. (2010). “Modeling hurricane waves and storm surge using integrally-coupled scalable computations”. *Coastal Engineering* **58**, pp. 45–65.

Dresback, K M et al. (2013). “Skill assessment of a real-time forecast system utilizing a coupled hydrologic and coastal hydrodynamic model during Hurricane Irene (2011)”. *Continental Shelf Research* **71**, pp. 78–94.

Fleming, J et al. (2008). “A Real Time Storm Surge Forecasting System Using ADCIRC”. *Estuarine and Coastal Modeling (2007)*, pp. 893–912.

Forbes, C et al. (2010). “A Retrospective Evaluation of the Storm Surge Produced by Hurricane Gustav (2008): Forecast and Hindcast Results”. *Wea. Forecasting* **25**, pp. 1577–1602.

Garzon, J L et al. (2016). “Storm Surge Modeling in Large Estuaries: Sensitivity Analyses to Parameters and Physical Processes in the Chesapeake Bay”. *Journal of Marine Science and Engineering* **4**.3, pp. 1–32.

Graham, L. et al. (2017). “A Measure-Theoretic Algorithm for Estimating Bottom Friction in a Coastal Inlet: Case Study of Bay St. Louis during Hurricane Gustav (2008)”. *Monthly Weather Review* **145**.3, pp. 929–954.

GRASS Development Team (2017). *Geographic Resources Analysis Support System (GRASS GIS) Software, Version 7.2*. Open Source Geospatial Foundation. URL: http://grass.osgeo.org.

Heberger, M et al. (2009). *The Impacts of Sea-Level Rise on the California Coast*. Tech. rep. California Climate Change Center.

Horn, B K P et al. *r.slope.aspect*. Open Source Geospatial Foundation. URL: https://grass.osgeo.org/grass78/manuals/r.slope.aspect.html.

Kalyanapu, A et al. (2009). “Effect of land use-based surface roughness on hydrologic model output”. *Journal of Spatial Hydrology* **9**.2, pp. 51–71.

Knowles, N (2010). “Potential Inundation Due to Rising Sea Levels in the San Francisco Bay Region”. *San Francisco Estuary and Watershed Science* **8** (1).

Larson, M et al. *r.grow*. Open Source Geospatial Foundation. URL: https://grass.osgeo.org/grass78/manuals/r.grow.html.

Lichter, M et al. (2012). “Assessing the costs of sea-level rise and extreme flooding at the local level: A GIS-based approach”. *Ocean and Coastal Management* **59**, pp. 47–62.

Liu, Z et al. (2018). “Investigating the role of model structure and surface roughness in generating flood inundation extents using one- and two-dimensional hydraulic models”. *Flood Risk Management* **12**.1, pp. 1–19.

Luettich, R A et al. (2004). *Formulation and Numerical Implementation of the 2D/3D ADCIRC Finite Element Model Version 44.XX*. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.675.3043&rep=rep1&type=pdf.

Mattocks, C et al. (2006). *Design and Implementation of a Real-Time Storm Surge and Flood Forecasting Capability for the State of North Carolina*. Tech. rep. University of North Carolina.

Mattocks, C et al. (2008). “A real-time, event-triggered storm surge forecasting system for the state of North Carolina”. *Ocean Modelling* **25**, pp. 95–119.

Mayo, T et al. (2014). “Data assimilation within the Advanced Circulation (ADCIRC) modeling framework for the estimation of Manning’s friction coefficient”. *Ocean Modelling* **76**, pp. 43–58.

Medeiros, S C et al. (2012). “Comparison of floodplain surface roughness parameters derived from land cover data and field measurements”. *Journal of Hydrology* **452-453**, pp. 139–149.

Mitas, L et al. *v.surf.rst*. Open Source Geospatial Foundation. URL: https://grass.osgeo.org/grass79/manuals/v.surf.rst.html.

Passeri, D et al. (2011). “Sensitivity of an ADCIRC Tide and Storm Surge Model to Manning’s n”. *Estuariene and Coastal Modeling* **4**.3, pp. 457–475.

Poulter, B et al. (2008). “Raster modelling of coastal flooding from sea-level rise”. *International Journal of Geographic Information Science*, pp. 167–182.

Ramirez, J A et al. (2016). “Hyper-resolution mapping of regional storm surge and tide flooding: Comparison of static and dynamic models”. *Natural Hazards* **82**, pp. 571–590.

Rubin, H et al. (2001). *Environmental Fluid Mechanics*. 270 Madison Avenue, New York, NY 10016: Marcel Dekker, pp. 280–284.

Shapiro, M. *g.region*. Open Source Geospatial Foundation. URL: https://grass.osgeo.org/grass78/manuals/g.region.html.

Smith, A et al. (2020). *U.S. Billion-Dollar Weather and Climate Disasters 1980-2019*. Tech. rep. NOAA National Centers for Environmental Information (NCEI).

Stewart, S R et al. (2019). *National Hurricane Center Tropical Cyclone Report*. Tech. rep. NOAA and National Weather Service.

Team, GRASS Development. *r.walk*. Open Source Geospatial Foundation. URL: https://grass.osgeo.org/grass76/manuals/r.walk.html.

Tull, N (2018). "Improving Accuracy of Real-Time Storm Surge Inundation Predictions". Master's thesis. Raleigh, NC, USA: North Carolina State University.

USGS. *TNM Download*. URL: http://https://viewer.nationalmap.gov/basic/ (visited on 02/15/2020).

— (2015). *USGS NED 1/3 arc-second n35w077 1 x 1 degree ArcGrid 2015*. URL: https://www.sciencebase.gov/catalog/item/581d213ee4b08da350d52e02 (visited on 02/15/2020).

— (2019). *NLCD 2016 Land Cover Conterminous United States*. URL: https://www.mrlc.gov/downloads/sciweb1/shared/mrlc/metadata/NLCD_2016_Land_Cover_L48.xml (visited on 02/15/2020).

Volp, N D et al. (2013). "The particle problem in the general theory of relativity". *Water Resources Research* **49**, pp. 4126–4135.

Westerink, J.J. et al. (2008). "A basin- to channel-scale unstructured grid hurricane storm surge model applied to southern Louisiana". *Monthly Weather Review* **136**.3, pp. 833–864.

APPENDICES

APPENDIX

A

ACRONYMS

ADCIRC	Advanced Circulation model
APS	ADCIRC Prediction System
ASGS	ADCIRC Surge Guidance System
CERA	Coastal Emergency Risks Assessment
DEM	Digital Elevation Model
FEMA	Federal Emergency Management Agency
GIS	Geographic Information Systems
GRASS	Geographic Resources Analysis Support System
HEC	Hydrologic Engineering Center
HEC-RAS	Hydrologic Engineering Center's River Analysis System
IDW	Inverse Distance Weight
LiDAR	Light Detection and Ranging
NAD83	North American Datum of 1983
NAVD88	North American Vertical Datum of 1988
NC	North Carolina
NCEM	North Carolina Emergency Management

NCFS	North Carolina Forecast System
NC9	North Carolina v9.98 ADCIRC mesh
NED	National Elevation Database
NHC	National Hurricane Center
NLCD	National Land Cover Database
RENCI	Renaissance Computing Institute
TNM	The National Map
USGS	United States Geological Survey
WGS84	World Geodetic System 1984

APPENDIX

B

GRASS MODULES

B.1 g.region

The *g.region* module allows the user to manage the settings of the current geographic region. These regional boundaries can be set by the user directly and/or set from a region definition file (stored under the `windows` directory in the user's current mapset). The user can create, modify, and store as many geographic region definitions as desired for any given mapset. However, only one of these geographic region definitions will be current at any given moment, for a specified mapset; i.e., GRASS programs that respect the geographic region settings will use the current geographic region settings [*g.region*].

https://grass.osgeo.org/grass78/manuals/g.region.html

B.2 r.grow

r.grow adds cells around the perimeters of all areas in a user-specified raster map layer and stores the output in a new raster map layer. The user can use it to grow by one or more than one cell (by varying the size of the **radius** parameter), or like *r.buffer*, but with the option of preserving the original cells (similar to combining *r.buffer* and *r.patch*).

If **radius** is negative, *r.grow* shrinks areas by removing cells around the perimeters of all areas [*r.grow*].

https://grass.osgeo.org/grass78/manuals/r.grow.html

B.3 v.surf.rst

v.surf.rst program performs spatial approximation based on z-values (input vector map is 3D and **zcolumn** parameter is not given), categories (input vector map is 2D and **zcolumn** parameter is not given), or *attributes* (**zcolumn** parameter is given) of point or isoline data given in a vector map named **input** to grid cells in the output raster map **elevation** representing a surface.

As an option, simultaneously with approximation, topographic parameters slope, aspect, profile curvature (measured in the direction of the steepest slope), tangential curvature (measured in the direction of a tangent to contour line) or mean curvature are computed and saved as raster maps specified by the options **slope, aspect, pcurv, tcurv, mcurv** respectively. If **-d** flag is set, *v.surf.rst* outputs partial derivatives f_x,f_y,f_{xx}, f_{yy},f_{xy} instead of slope, aspect, profile, tangential and mean curvatures respectively. If the input vector map have time stamp, the program creates time stamp for all output maps.

User can either use *r.mask* to set a mask or specify a raster map in **mask** option, which will be used as a mask. The approximation is skipped for cells which have zero or NULL value in mask. NULL values will be assigned to these cells in all output raster maps. Data points are checked for identical points and points that are closer to each other than the given **dmin** are removed. If sparsely digitized contours or isolines are used as input, additional points are computed between each 2 points on a line if the distance between them is greater than specified **dmax**. Parameter **zmult** allows user to rescale the values used for approximation (useful e.g. for transformation of elevations given in feet to meters, so that the proper values of slopes and curvatures can be computed).

Regularized spline with **tension** is used for the approximation. The tension parameter tunes the character of the resulting surface from thin plate to membrane. Smoothing parameter **smooth** controls the deviation between the given points and the resulting surface and it can be very effective in smoothing noisy data while preserving the geometrical properties of the surface. With the smoothing parameter set to zero (**smooth=0**) the resulting surface passes exactly through the data points (spatial interpolation is performed). When smoothing parameter is used, it is also possible to output a vector point map deviations containing **deviations** of the resulting surface from the given data.

If the number of given points is greater than **segmax**, segmented processing is used. The region is split into quadtree-based rectangular segments, each having less than **segmax** points and approximation is performed on each segment of the region. To ensure smooth connection of segments the approximation function for each segment is computed using the points in the given segment and the points in its neighborhood which are in the rectangular window surrounding the given segment. The number of points taken for approximation is controlled by **npmin**, the value of which must be larger than **segmax**. User can choose to output vector maps **treeseg** and **overwin** which represent the quad tree used for segmentation and overlapping neighborhoods from which additional points for approximation on each segment were taken.

Predictive error of surface approximation for given parameters can be computed using the **-c** flag. A crossvalidation procedure is then performed using the data given in the vector map **input** and the estimated predictive errors are stored in the vector point map **cvdev**. When using this flag, no raster output maps are computed. Anisotropic surfaces can be interpolated by setting anisotropy angle **theta** and scaling factor **scalex**. The program writes values of selected input and internally computed parameters to the history file of raster map **elevation**.

The user must run *g.region* before the program to set the region and resolution for approximation [*v.surf.rst*].

https://grass.osgeo.org/grass78/manuals/v.surf.rst.html

B.4 r.slope.aspect

r.slope.aspect generates raster maps of slope, aspect, curvatures and first and second order partial derivatives from a raster map of true elevation values. The user must specify the input **elevation** raster map and at least one output raster maps. The user can also specify the **format** for slope (degrees, percent; default=degrees), and the **zscale**: multiplicative factor to convert elevation units to horizontal units; (default 1.0).

The **elevation** input raster map specified by the user must contain true elevation values, *not* rescaled or categorized data. If the elevation values are in other units than in the horizontal units, they must be converted to horizontal units using the parameter **zscale**. *In GRASS GIS 7, vertical units are not assumed to be meters any more. For example, if both your vertical and horizontal units are feet, parameter zscale must not be used* [*r.slope.aspect*].

https://grass.osgeo.org/grass78/manuals/r.slope.aspect.html

B.5 r.walk

r.walk computes anisotropic cumulative cost of moving between different geographic locations on an input elevation raster map whose cell category values represent elevation combined with an input raster map layer whose cell values represent friction cost. *r.walk* outputs 1) a raster map showing the lowest cumulative cost (time) of moving between each cell and the user-specified starting points and 2) a second raster map showing the movement direction to the next cell on the path back to the start point (see Movement Direction). It uses an input elevation raster map whose cell category values represent elevation, combined with a second input raster map whose cell values represent friction costs.

This function is similar to *r.cost*, but in addition to a friction map, it considers an anisotropic travel time due to the different walking speed associated with downhill and uphill movements [*r.walk*].

https://grass.osgeo.org/grass78/manuals/r.walk.html