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

Carter Rucker

Committee

Advisor – Dr. Casey Dietrich

Committee members – Dr. Helena Mitasova and Dr. Beth Sciaudone

COASTAL RESILIENCE CENTER
A U.S. Department of Homeland Security Center of Excellence

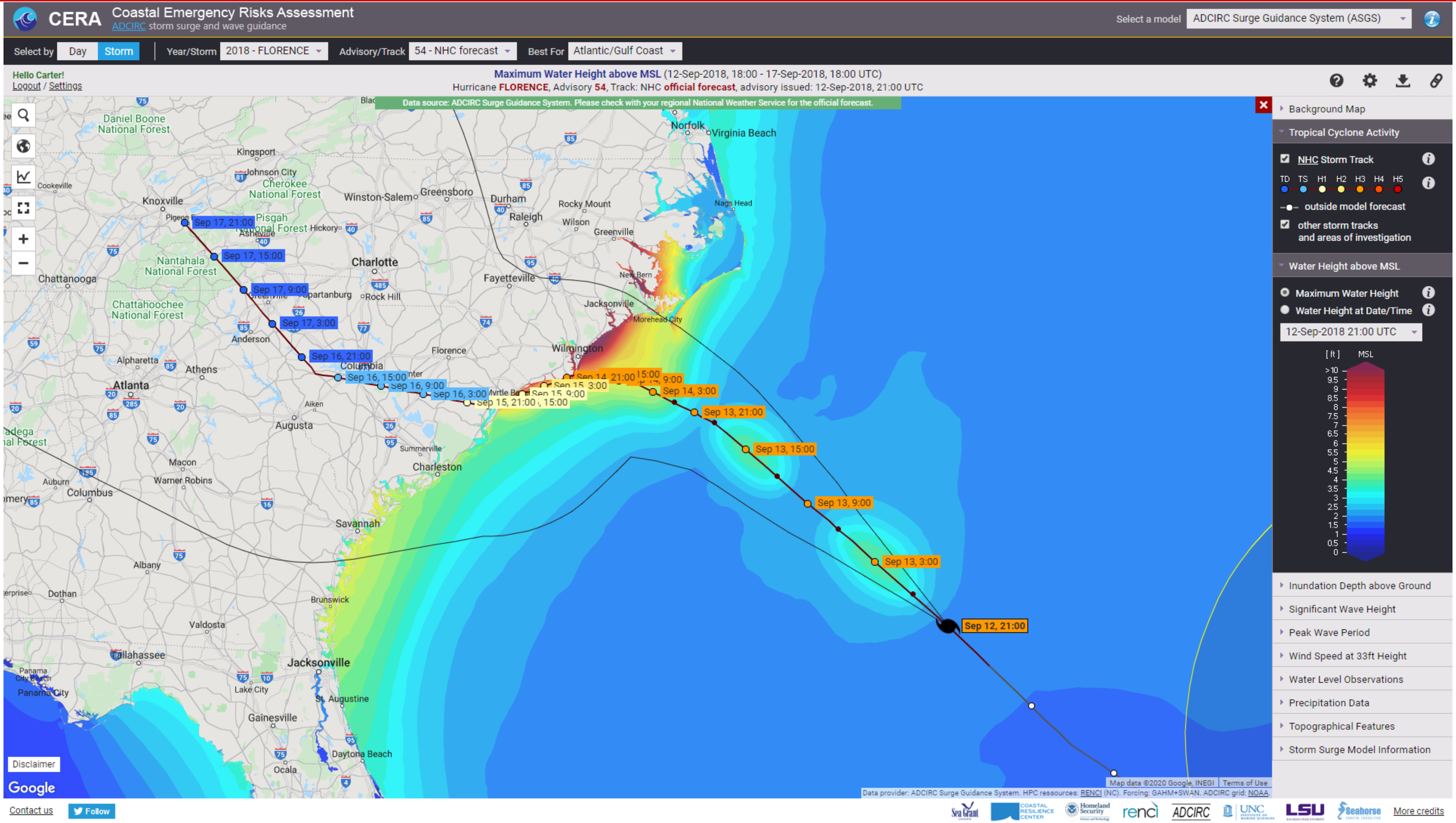
My Background

- From Raleigh, NC
- Grew up going to NC beaches
- Attended NC State University
  - Received a B.S. in Civil Engineering in 2018
  - Participated in two research projects with Beth Sciaudone and helped collect field data

- **Motivation and Background**
- Existing Method
  - Static
- Improving Existing Method
- New Downscaling Methods
  - Slopes
  - Head Loss
- Evaluation of Accuracy
- Conclusions and Future Work

Importance of Storm Surge Predictions

- Storm surge is one of the greatest threats to coastal communities and infrastructure
- Hurricanes Dorian, Michael, Florence, Maria, Irma, Harvey, and Matthew totaled $326 billion in damage from 2016-2019
- Emergency managers rely on models to predict storm surge and coastal flooding
- Models must be **accurate** enough to be trusted, **fast** enough to use in forecasting, and **precise** enough to predict at key infrastructure

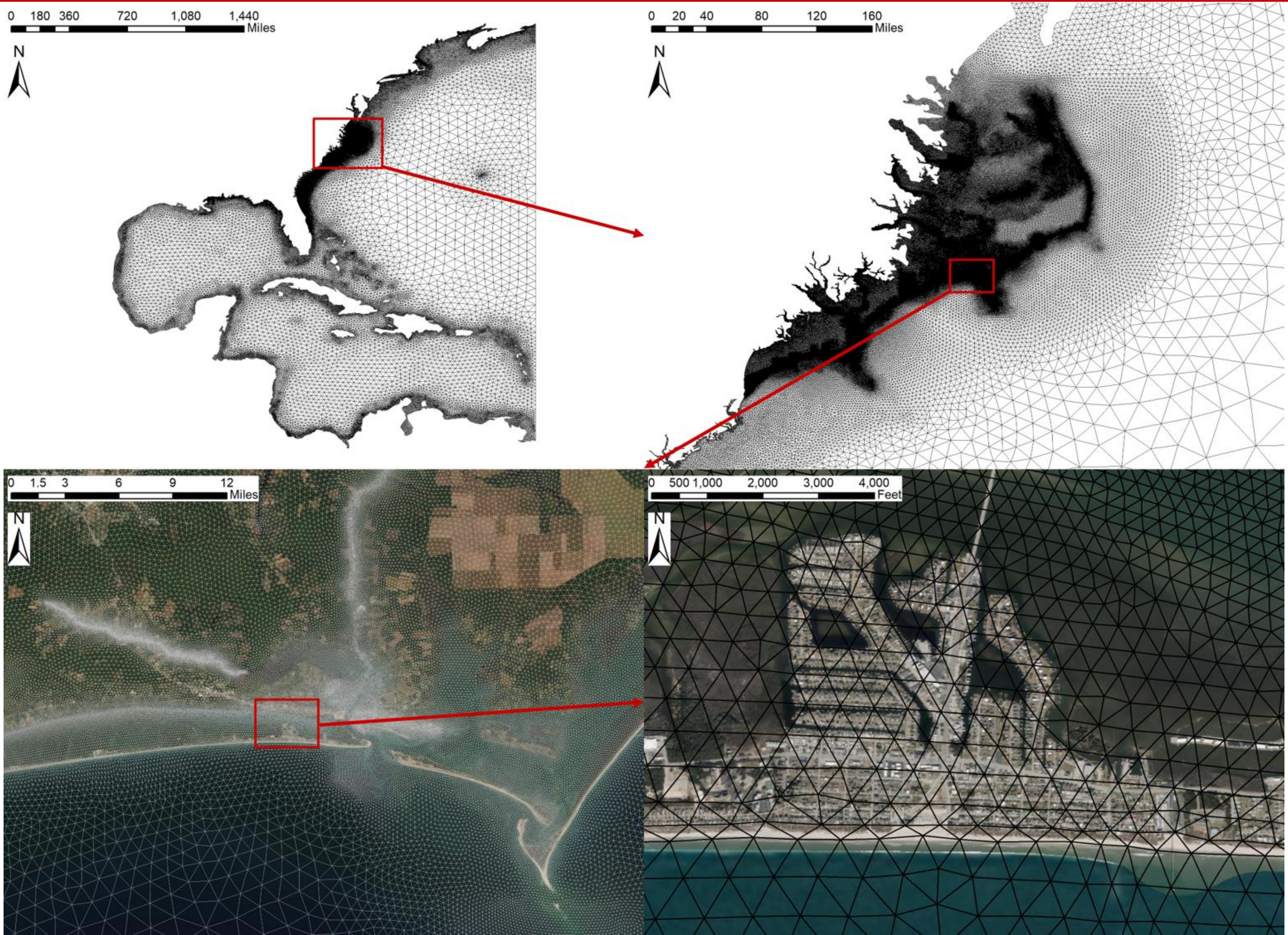


Maximum water levels for Hurricane Florence, advisory 54 – visualized on the CERA website

ADCIRC

Full NC9 Mesh

NC9 in North Carolina

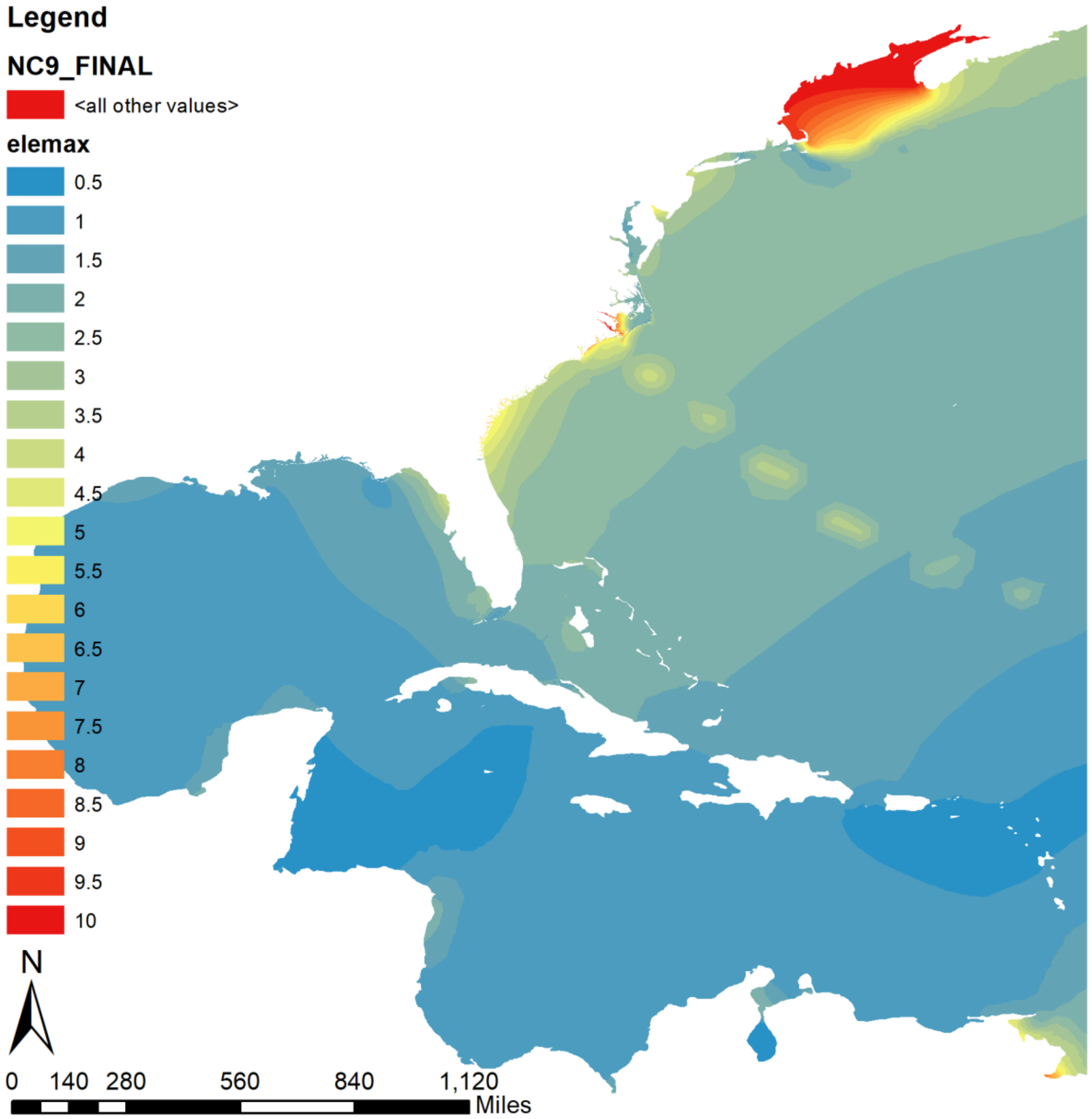


Carteret County

Atlantic Beach

Kalpana

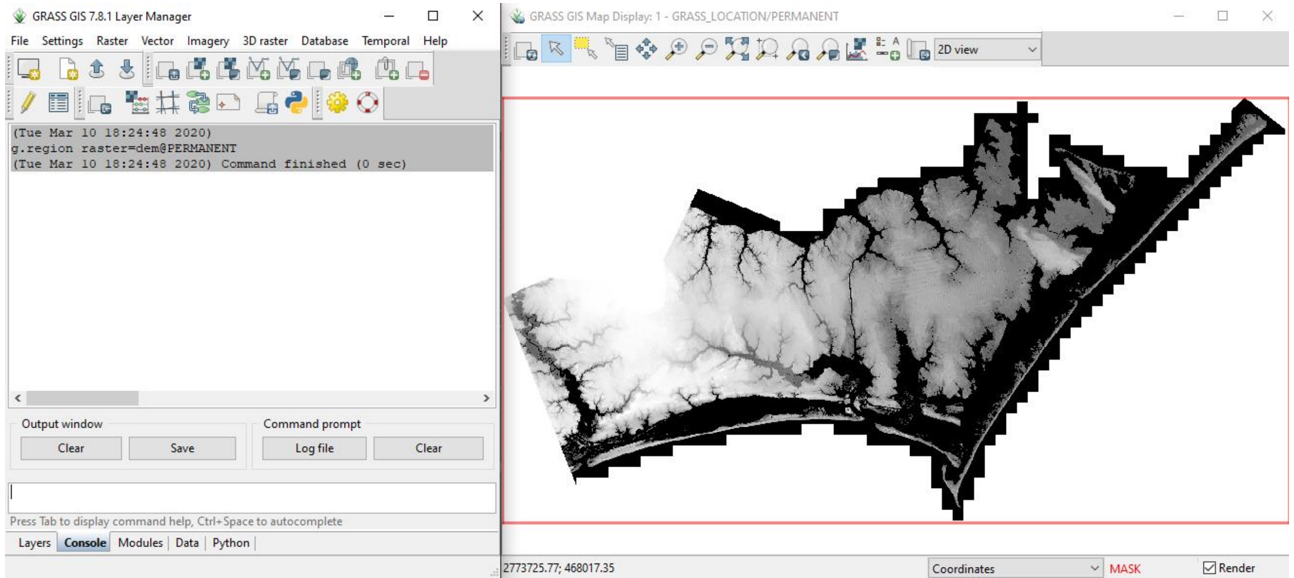
- Python code for visualizing ADCIRC output
- Creates binned ESRI shapefiles or KMZ files
  - Easily cross-reference ADCIRC results with GIS data
- Accepts ADCIRC outputs for max water levels, wind speeds, wave heights, and peak wave period
- This research uses max water level output



Maximum water elevations, visualized as an ESRI shapefile using Kalpana

GRASS GIS

- Geographic Resource Analysis Support System
- Open-source software, free to download
- Used for all GIS operations in this research
- Processes raster data efficiently and is easy to use with Python



Objectives

To achieve the goal of improving the accuracy and applicability of real-time storm surge downscaling methods:

1. Evaluate the accuracy of the existing static downscaling method
2. Increase the applicability of the downscaling code
3. Develop and evaluate a method that downscales water levels using the water surface slope
4. Develop and evaluate a downscaling method that includes head losses due to land cover

What does this study add to the literature?

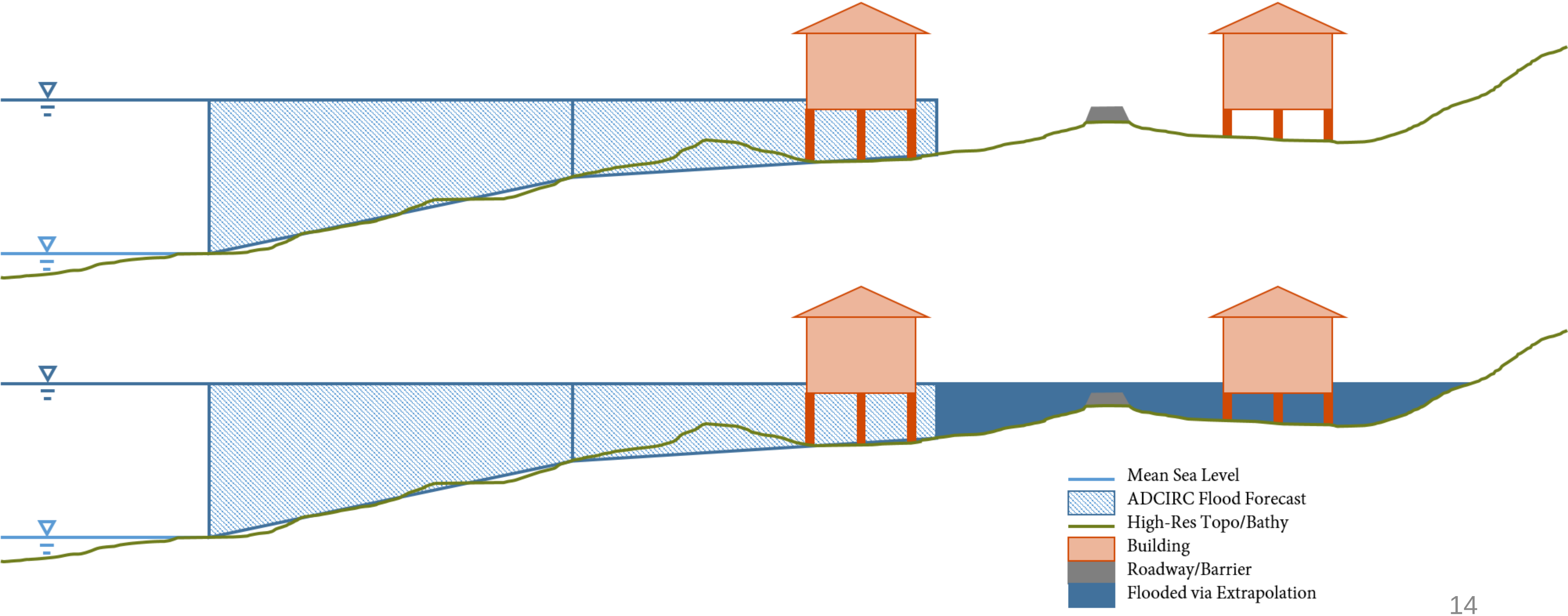
Two novel aspects:

1. Downscaling can be applied **globally** in real-time
    - Methods can be applied anywhere on earth
    - Computational time is sufficient for forecasting
2. Downscaling incorporates basic measures of **physics**
    - Measures of physics are not used in downscaling among literature
    - Accounts for elevation changes, water surface slope, and/or head loss
    - Improves accuracy and allows for parameter flexibility

- Motivation and Background
- **Existing Method**
  - **Static**
- Improving Existing Method
- New Downscaling Methods
  - Slopes
  - Head Loss
- Evaluation of Accuracy
- Conclusions and Future Work

General Downscaling Workflow

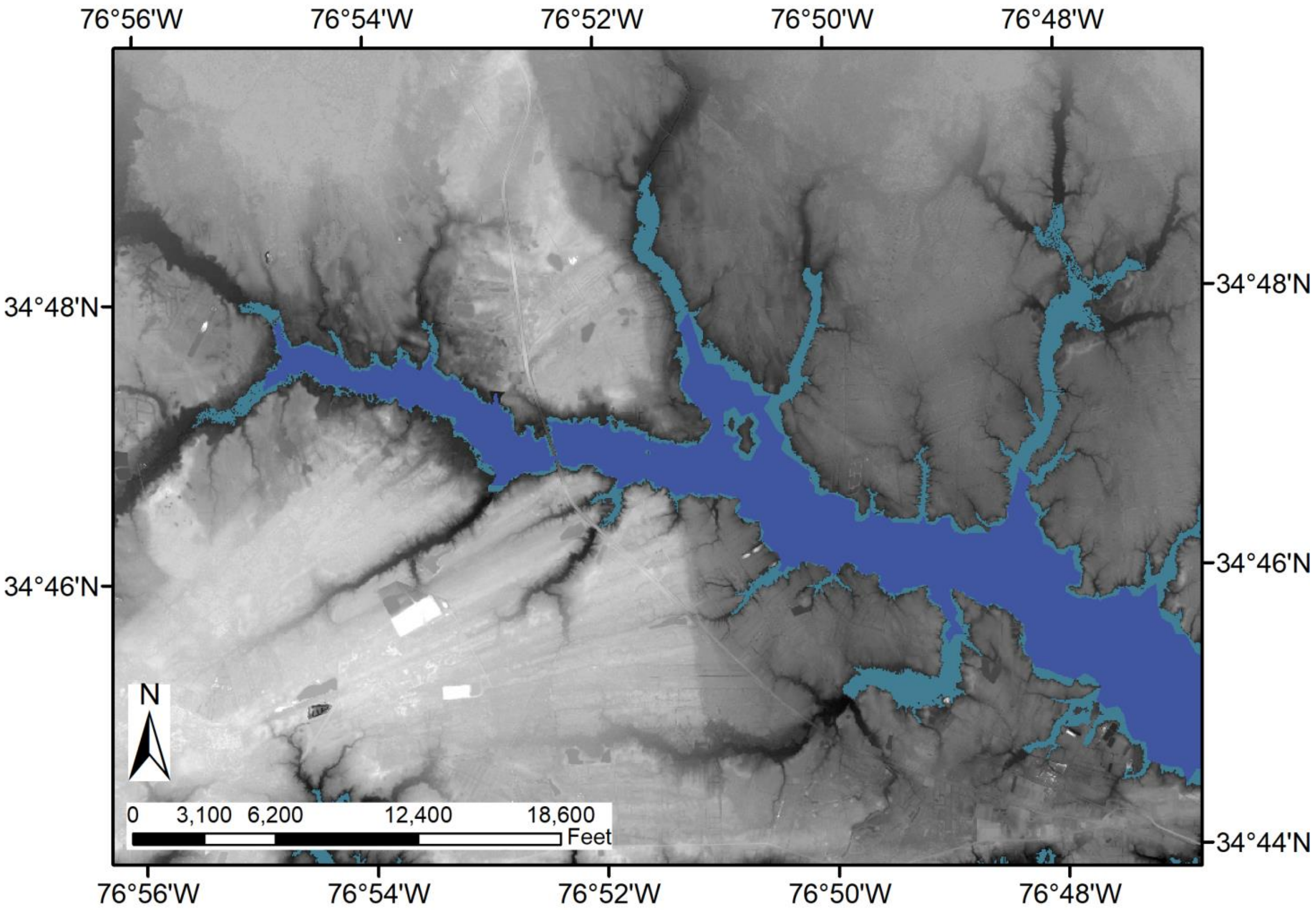
Static Method



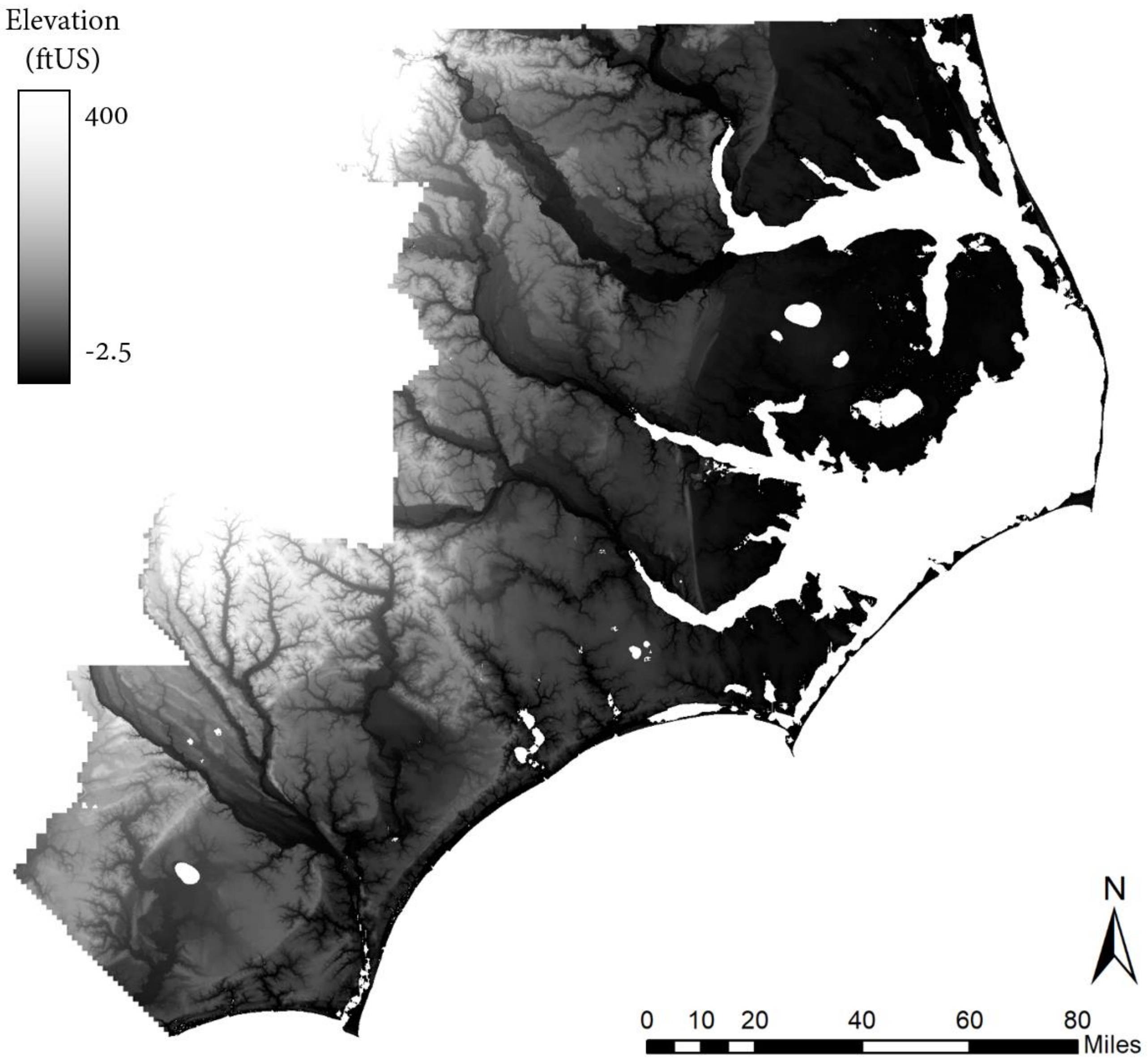
Legend

ADCIRC (NC9 mesh)

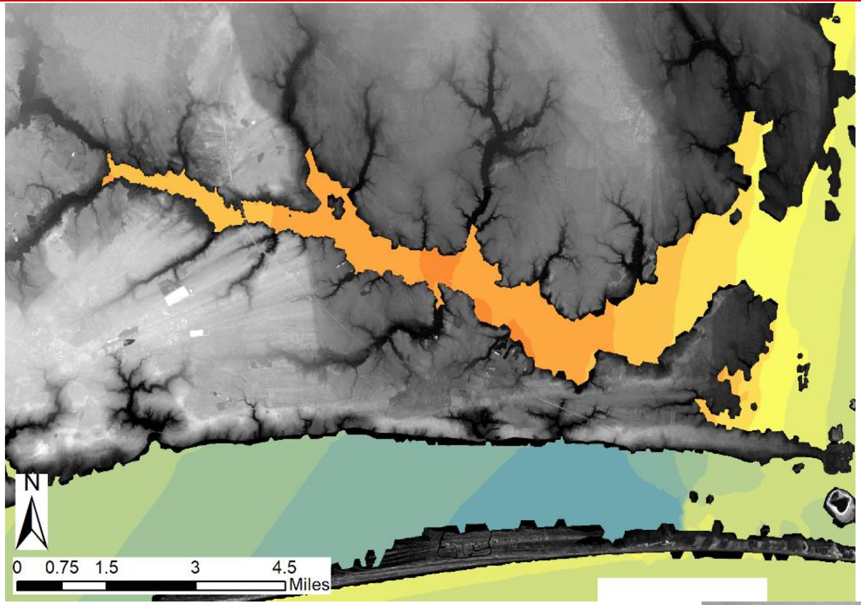
Downscaled using static method



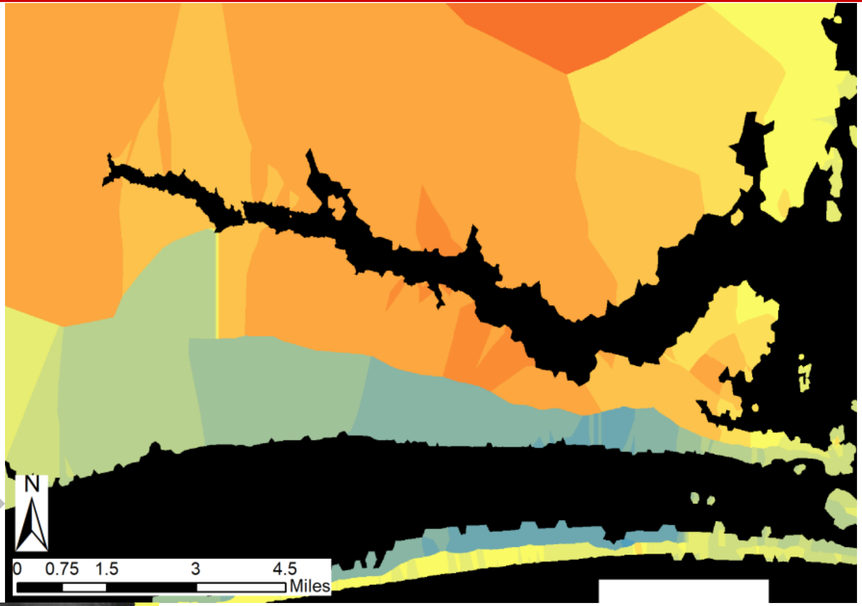
ADCIRC (NC9) results vs. downscaled results using the static method for Hurricane Florence (2018).



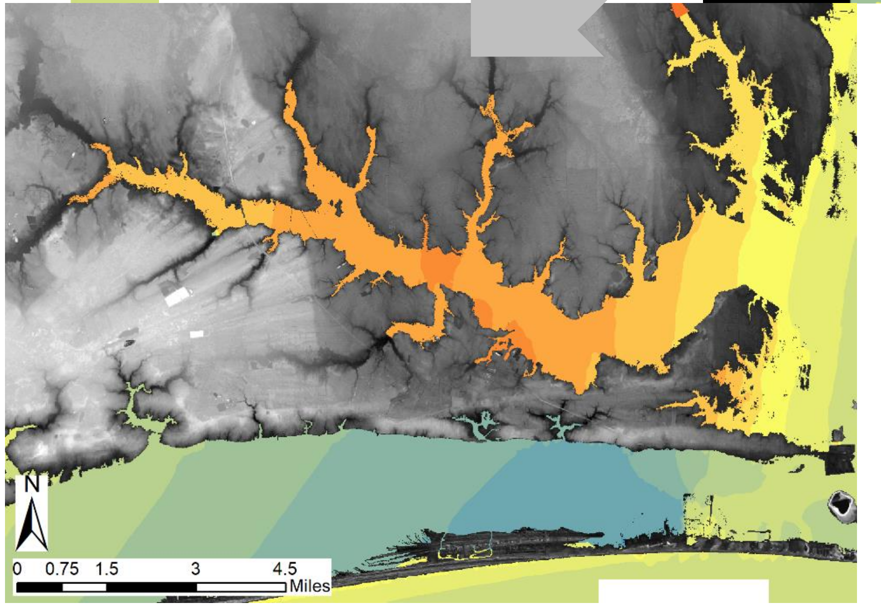
NC Emergency Management (NCEM) DEM



ADCIRC (NC9) water elevations & DEM



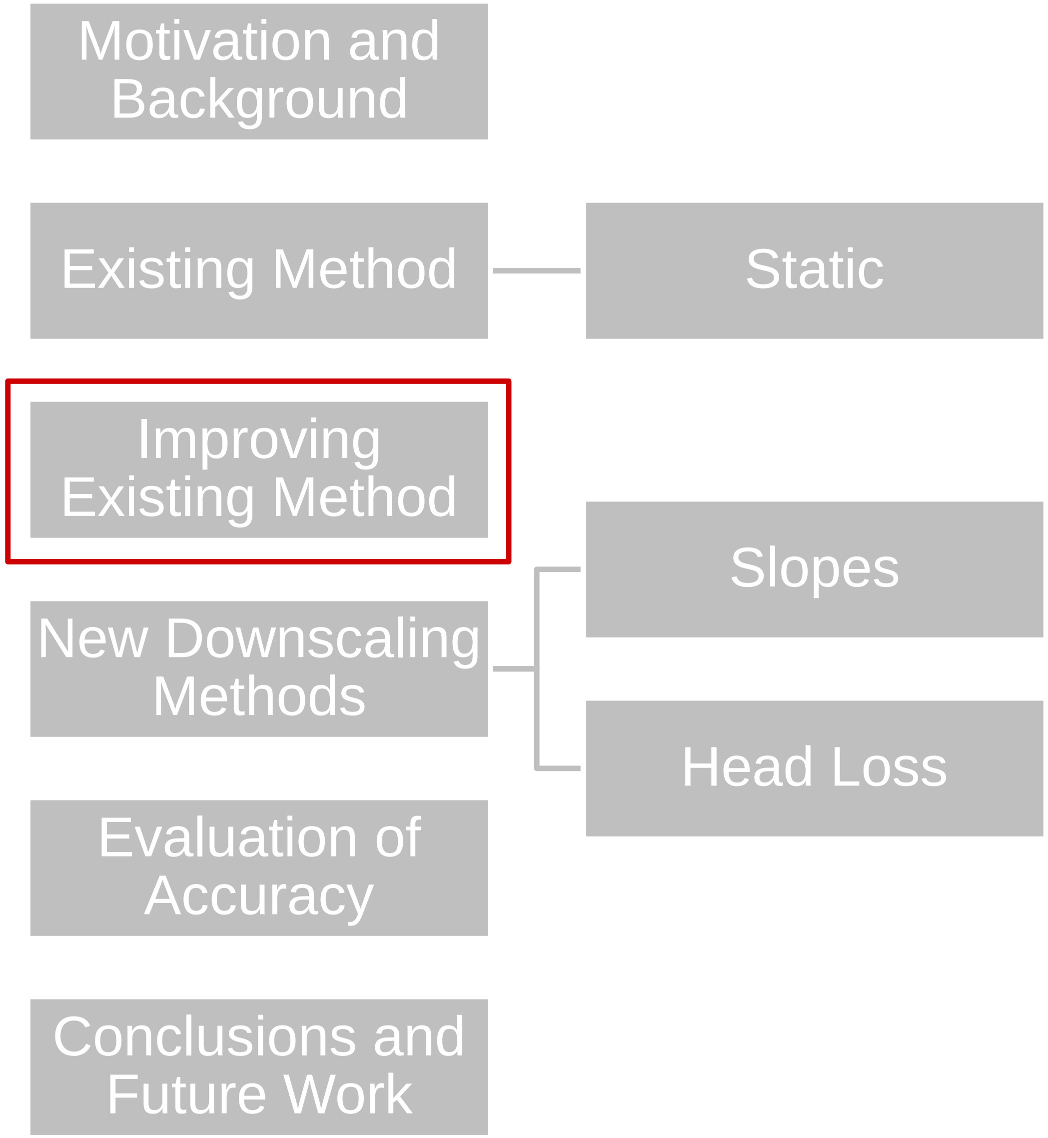
ADCIRC water elevations *extrapolated*



Downscaled water surface elevations

Limitations to Original Methods

- Interpolated water elevations directly from the NC9 ADCIRC mesh to the NC raster using an Inverse Distance Weight (**IDW**) process
- Interpolated 623,000 mesh vertices to the 430 million-cell raster
- IDW process is costly; takes around 39 out of the total 64 minutes required to downscale in serial
  - Requires parallelization over large domains
- New set of IDWs is required for each new ADCIRC mesh or DEM
  - Reduces applicability of the downscaling method
  - Requires an experienced GIS user


Motivation and Background
Existing Method
Static
Improving Existing Method
Slopes
New Downscaling Methods
Head Loss
Evaluation of Accuracy
Conclusions and Future Work

Using Kalpana to Increase Applicability

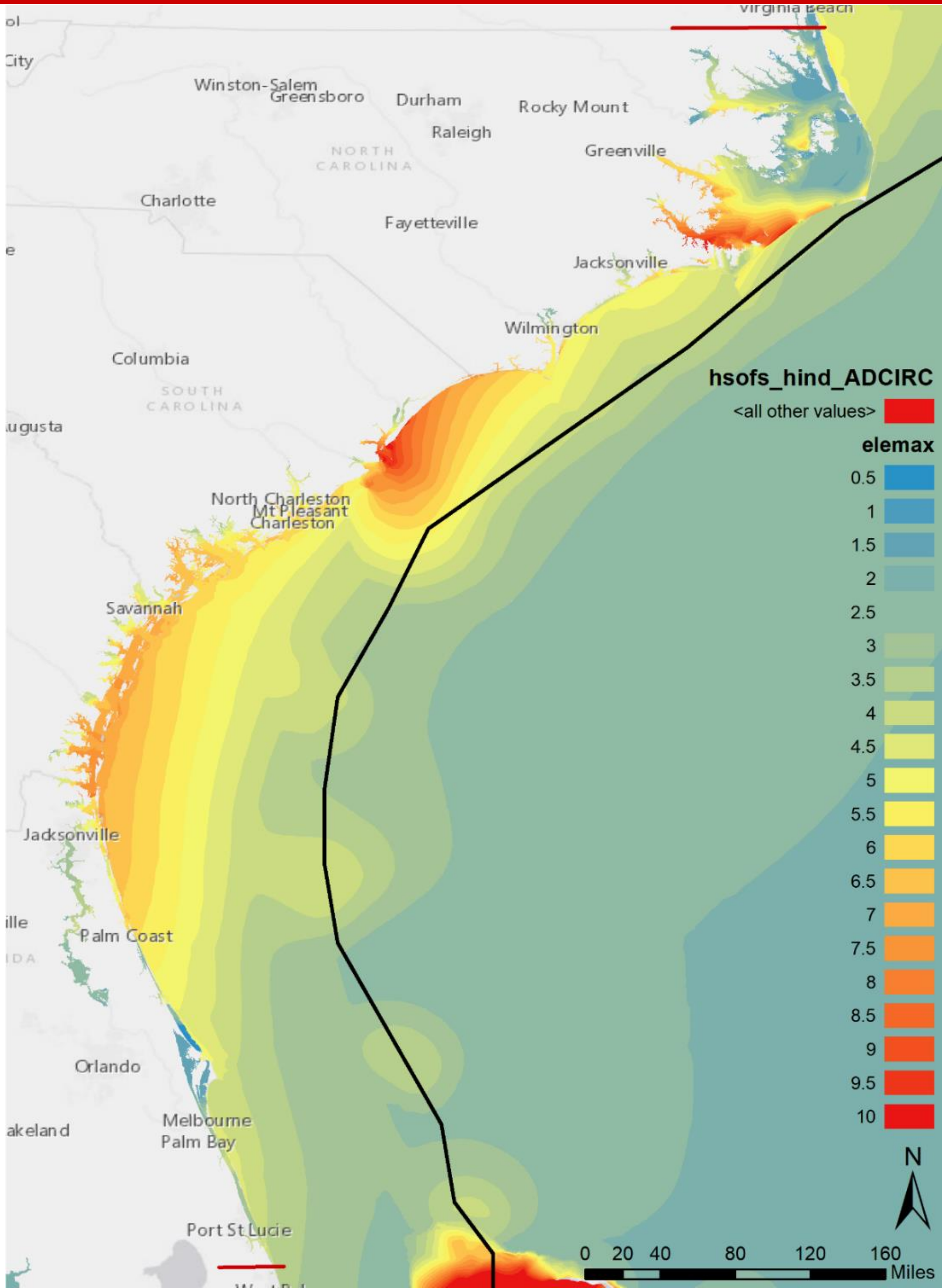
- Kalpana converts ADCIRC data to a shapefile, then the shapefile is converted to a raster in GRASS GIS
- IDW process is no longer necessary
- Process takes 2 min for the NC DEM with NC9 (compared to 39 min using IDW interpolation)
- Kalpana is capable of accepting ADCIRC input from any mesh
- Downscaling methods can now be applied anywhere in the world

Static Method in Kalpana

- Kalpana has been integrated with the static method
  - Can now visualize ***and*** downscale ADCIRC forecasts in Kalpana
- A user-friendly interface was created so users can create their own GRASS location without much prior GIS knowledge
- The automatically-generated GRASS dataset contains:
  - A DEM provided by the user; accepts any resolution and allows users to import multiple DEMs
  - Information about the geographical region

Examples of Downscaling with Kalpana

- Used in forecasting with the static method during the 2019 hurricane season in NC
- Used at George Mason University to assess estuarine flood predictions in Chesapeake Bay
- Assisted Taylor Engineering in decision-making for flood map development
- Created hindcast for FEMA using Hurricane Dorian (2019) along the coasts of FL, GA, SC, and NC using HSOFS

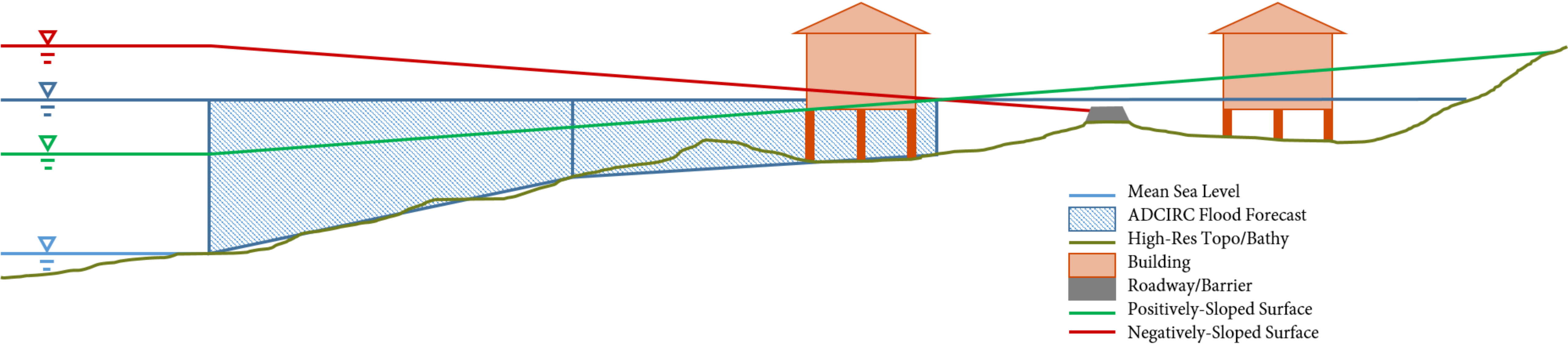


- Motivation and Background
- Existing Method
  - Static
- Improving Existing Method
- **New Downscaling Methods**
  - **Slopes**
  - Head Loss
- Evaluation of Accuracy
- Conclusions and Future Work

Geospatial Downscaling – Slopes Method

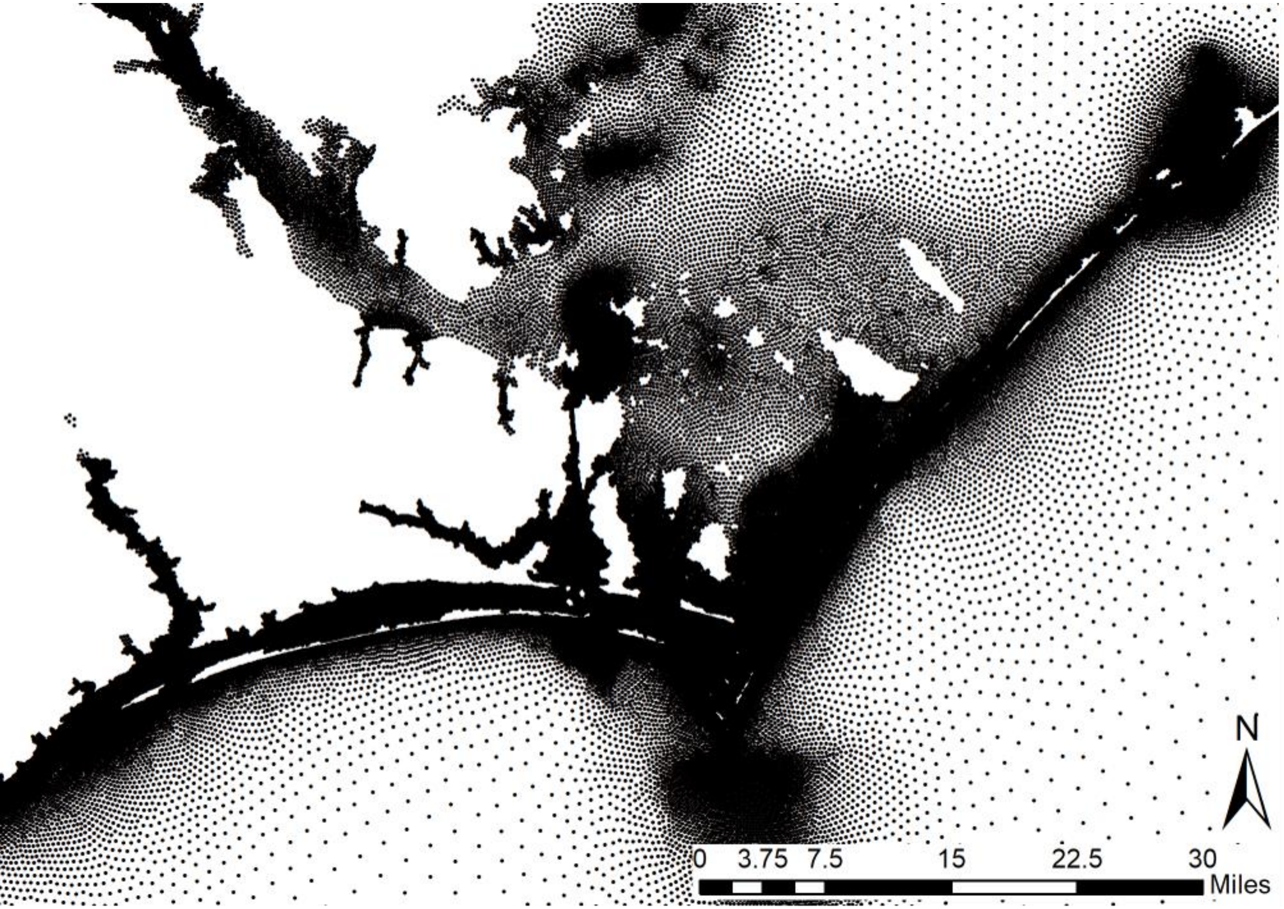
- Water surface slopes are present in water level output from numerical models
- Change in surface elevation per unit distance ($\Delta\zeta/\Delta x$)
- Related to calculation of storm surge
- Not represented in literature

Slopes Method

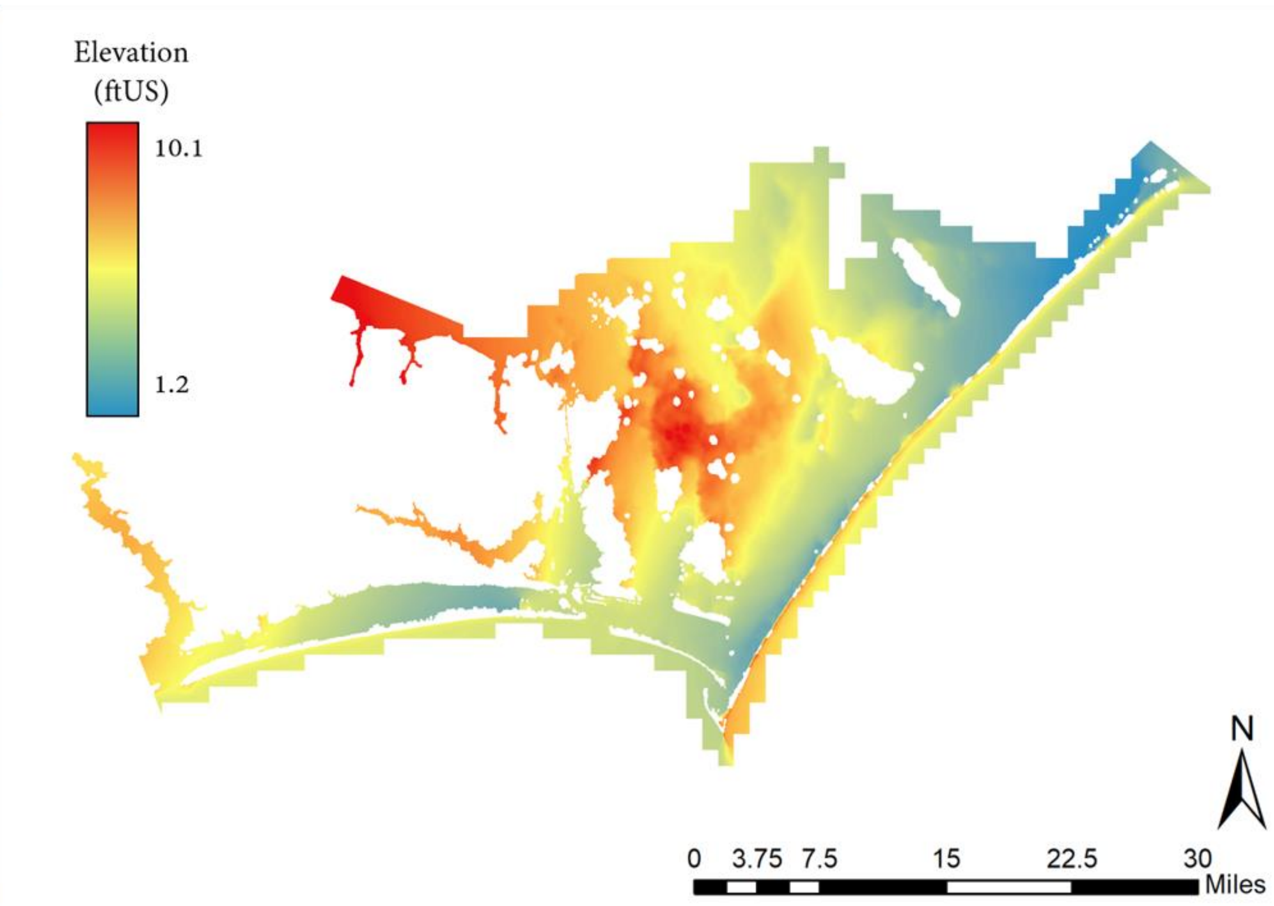


Generating a Continuous Water Elevation Surface

- Cannot use binned polygon; these slopes are 0
- Import ADCIRC maximum water elevations as points
- Interpolate using v.surf.rst GRASS module to create a continuous surface
- Does not perfectly match every point, but had an average vertical error of 3.8 mm



ADCIRC maximum water elevations
of flooded vertices



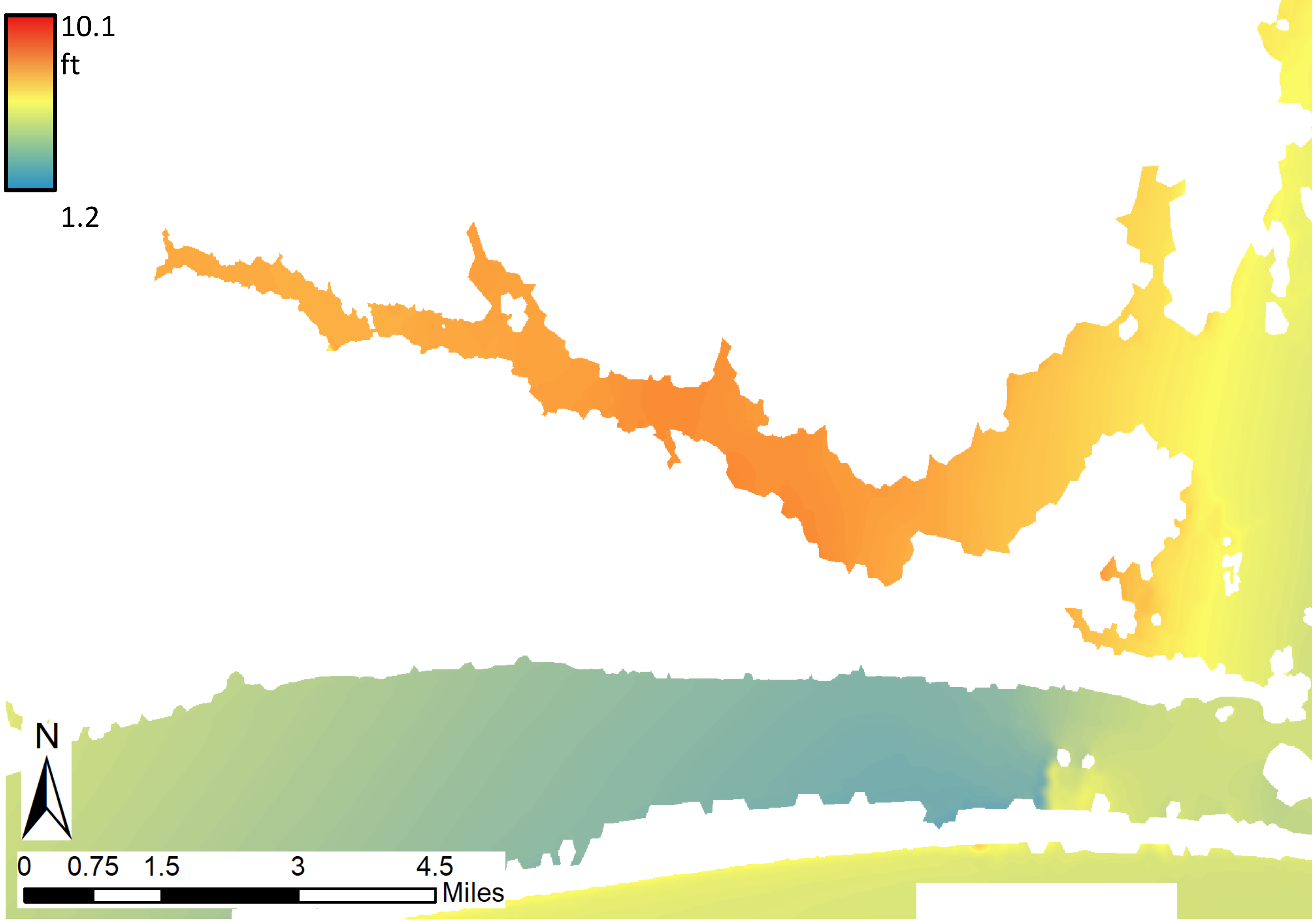
Water surface elevations
(from v.surf.rst)

Downscaling with Water Elevation Slopes

- Similarly to the static method, the slopes method extrapolates ADCIRC data to null cells
- Rather than extrapolating as a horizontal surface, slopes are taken into account:

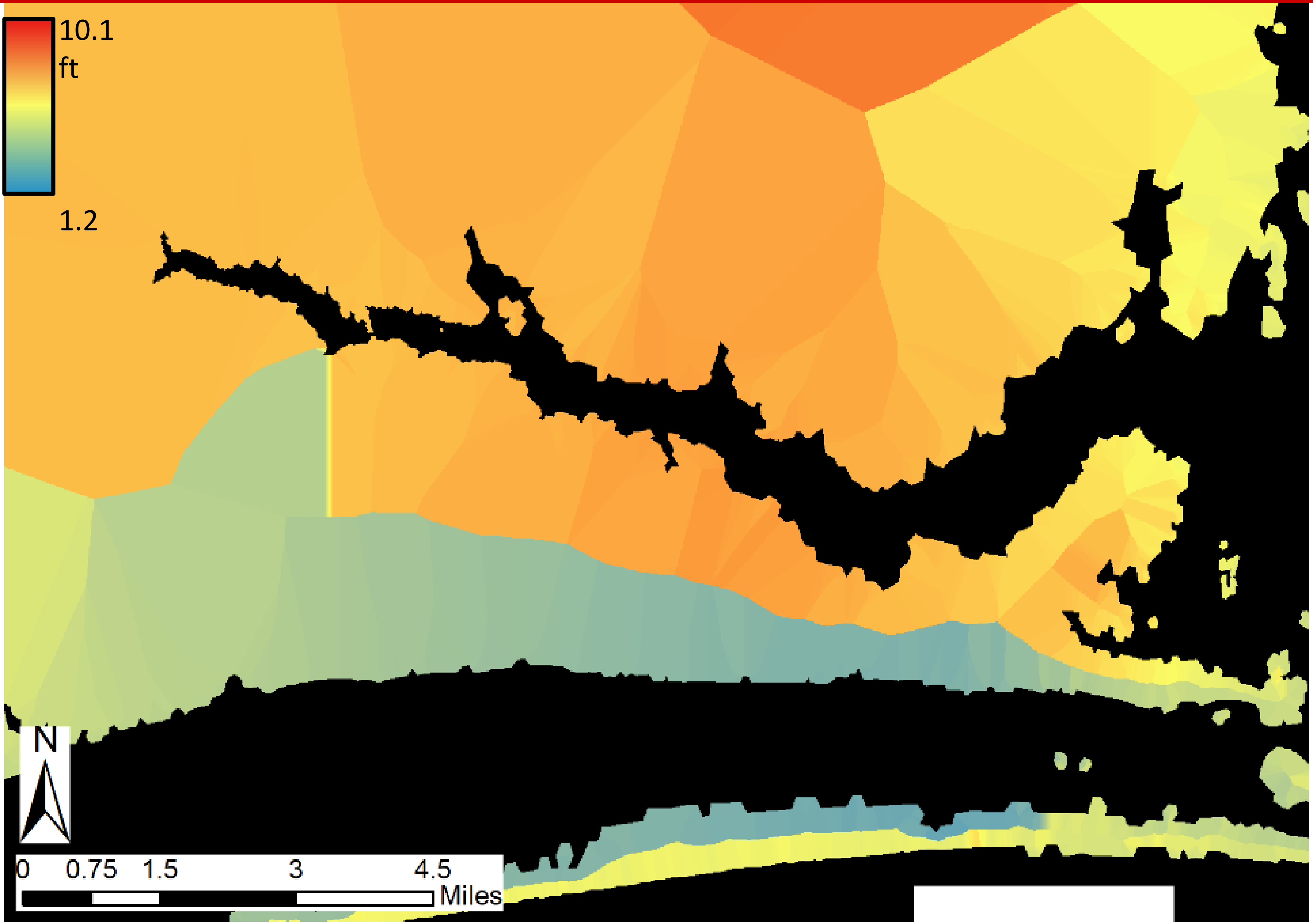
$$\zeta_{slopes} = \zeta_{static} \pm c(m_x \Delta x) \pm c(m_y \Delta y)$$

  - ζ: water elevation
  - c: exaggeration factor
  - m: surface slope
  - Δx: change in horizontal distance in the East-West direction
  - Δy: change in horizontal distance in the North-South direction



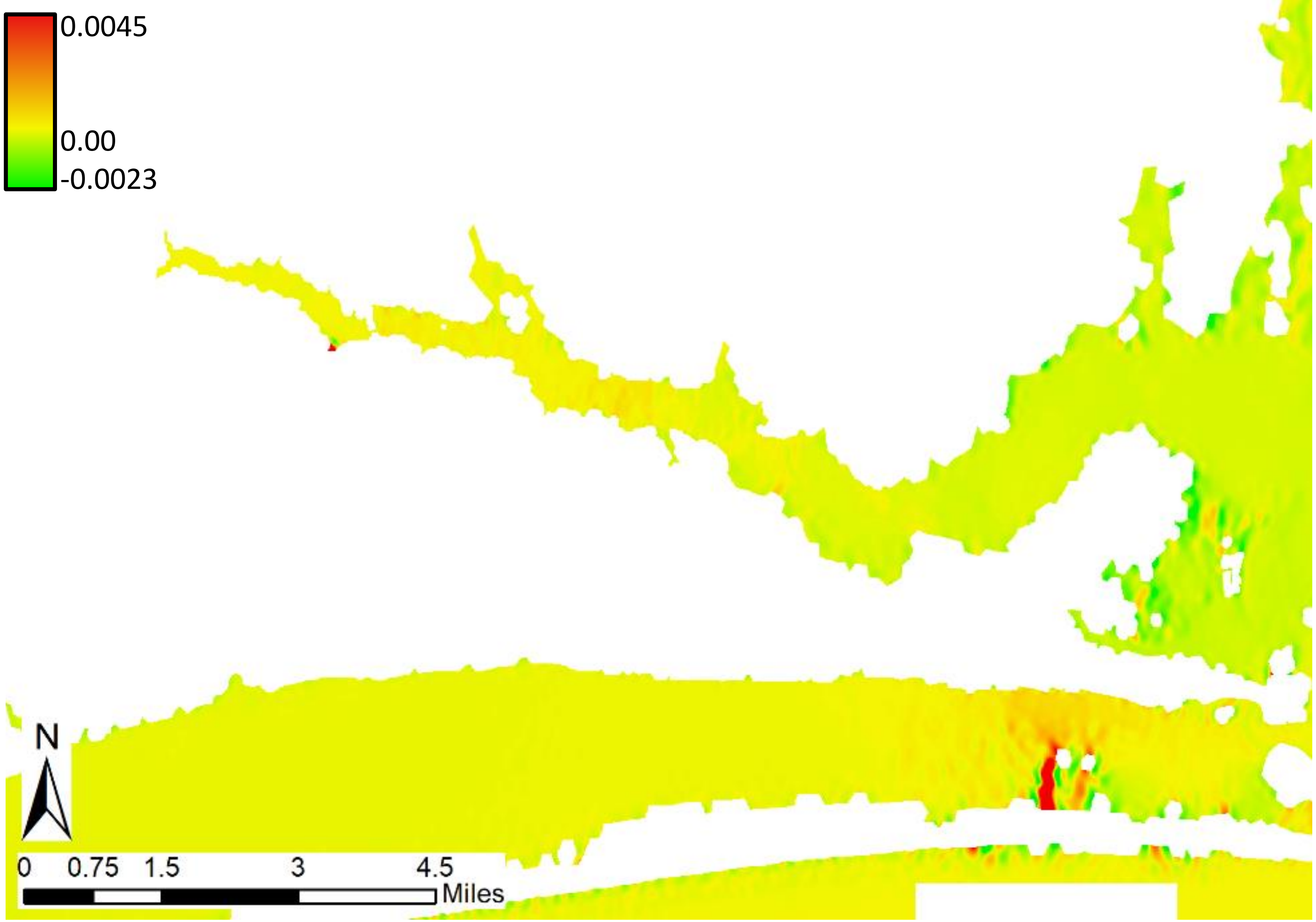
Water surface elevation (from v.surf.rst)

$$\zeta_{slopes} = \boxed{\boldsymbol{\zeta_{static}}} \pm c(m_x \Delta x) \pm c(m_y \Delta y)$$



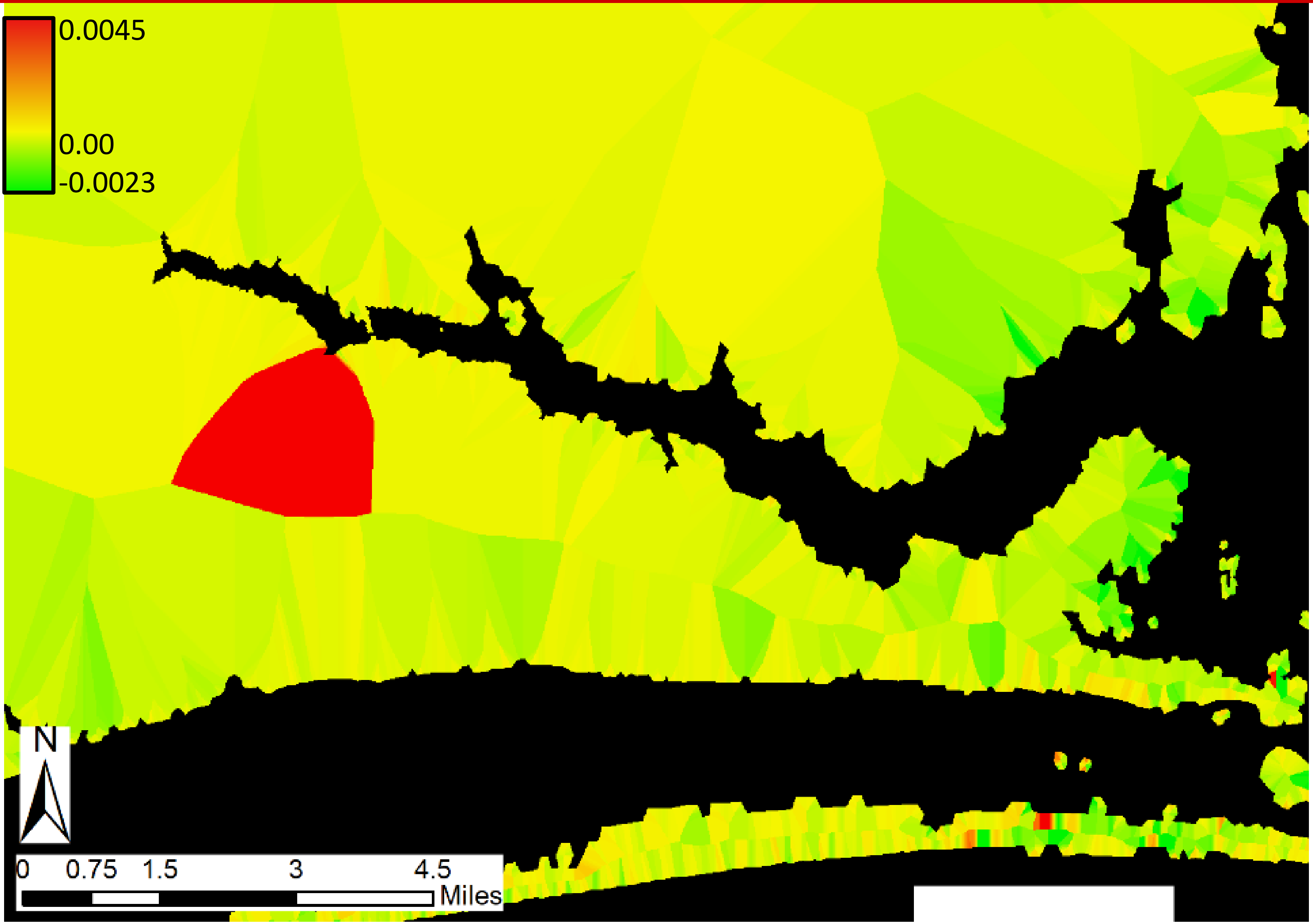
Water surface elevation, *extrapolated* horizontally

$$\zeta_{slopes} = \boxed{\boldsymbol{\zeta_{static}}} \pm c(m_x \Delta x) \pm c(m_y \Delta y)$$



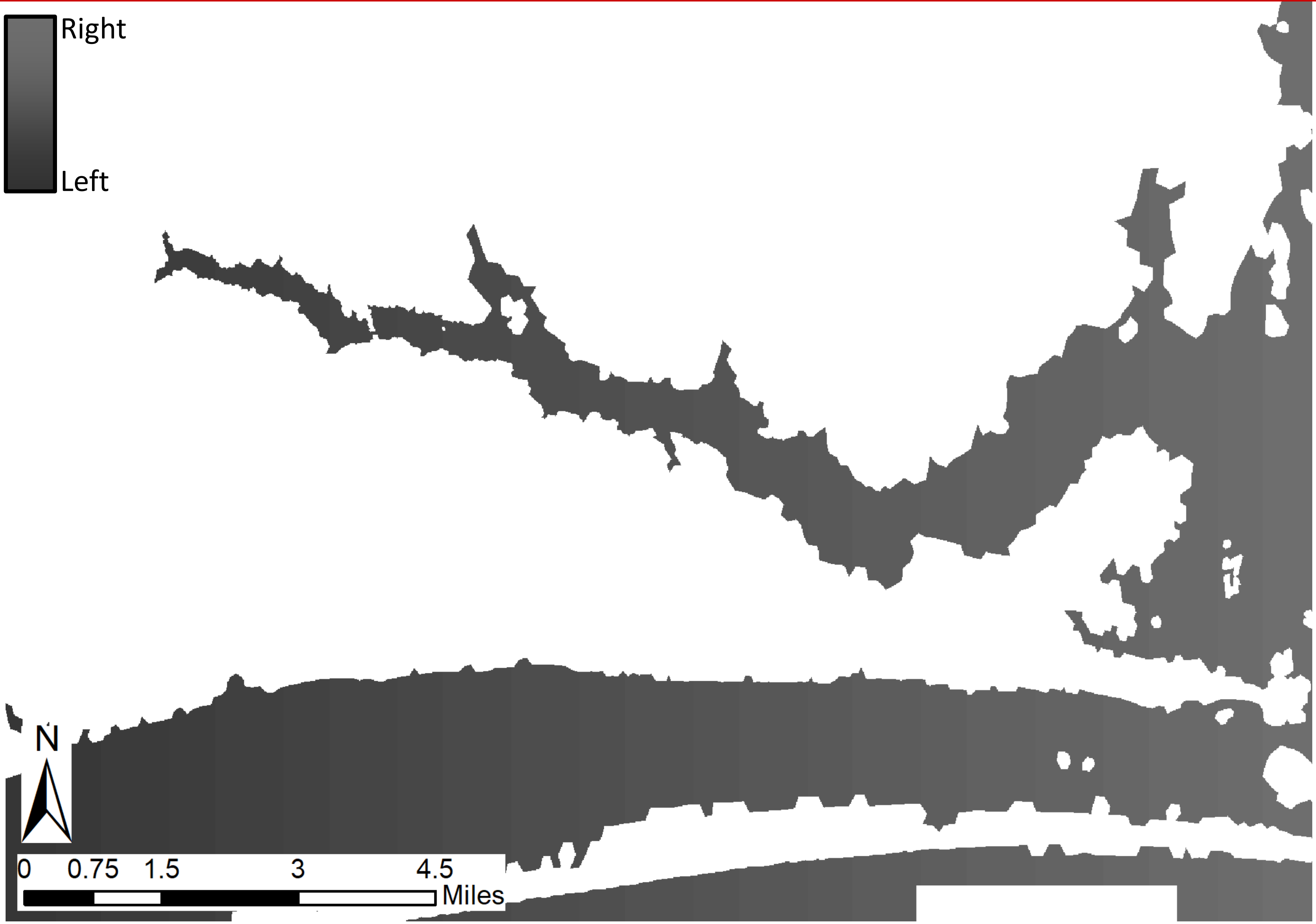
Water surface slope in the x (East-West) direction

$$\zeta_{slopes} = \zeta_{static} \pm c(\boldsymbol{m_x}\Delta x) \pm c(m_y \Delta y)$$



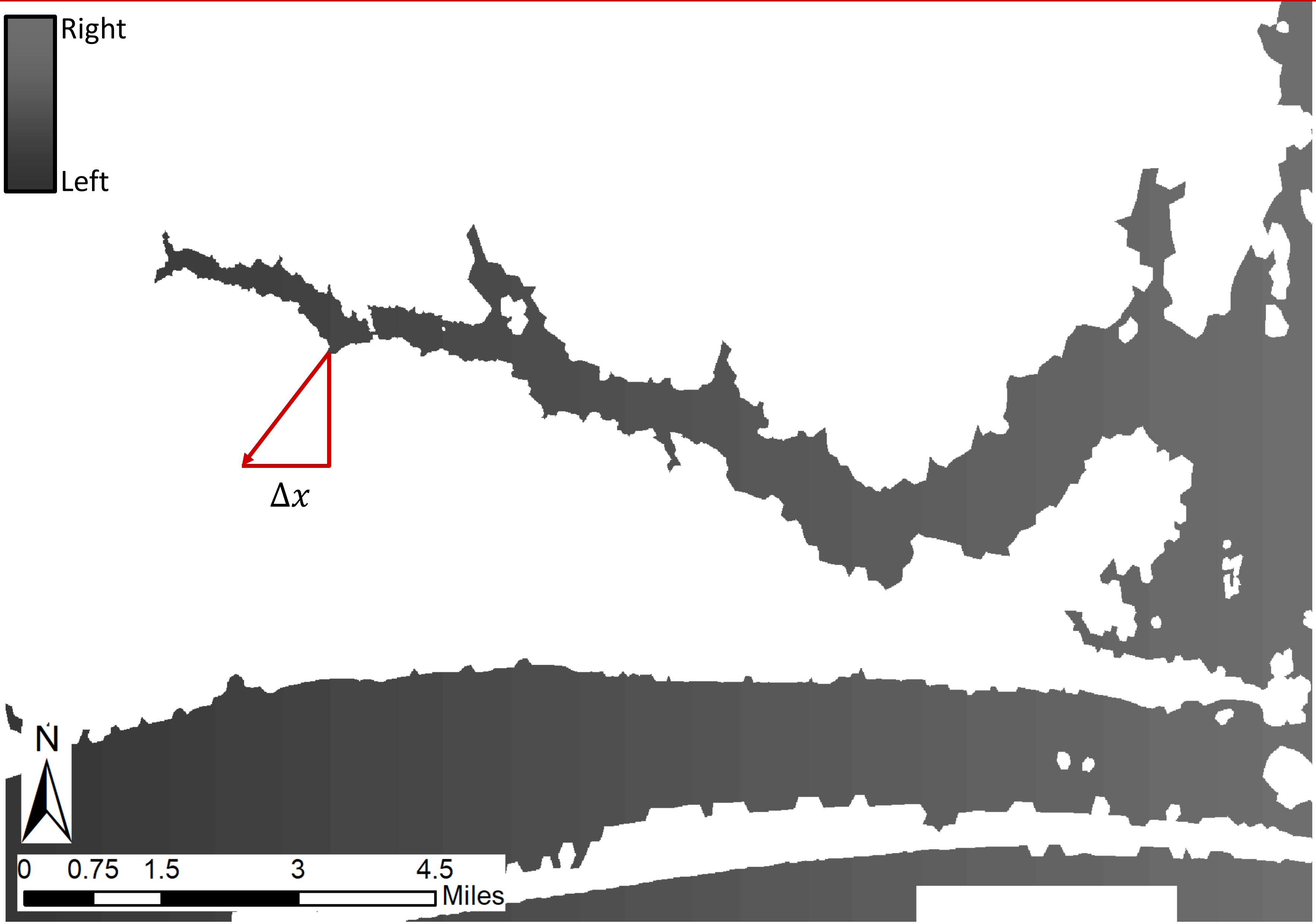
Water surface slope in the x (East-West) direction, *extrapolated*

$$\zeta_{slopes} = \zeta_{static} \pm c(\boldsymbol{m_x}\Delta x) \pm c(m_y \Delta y)$$



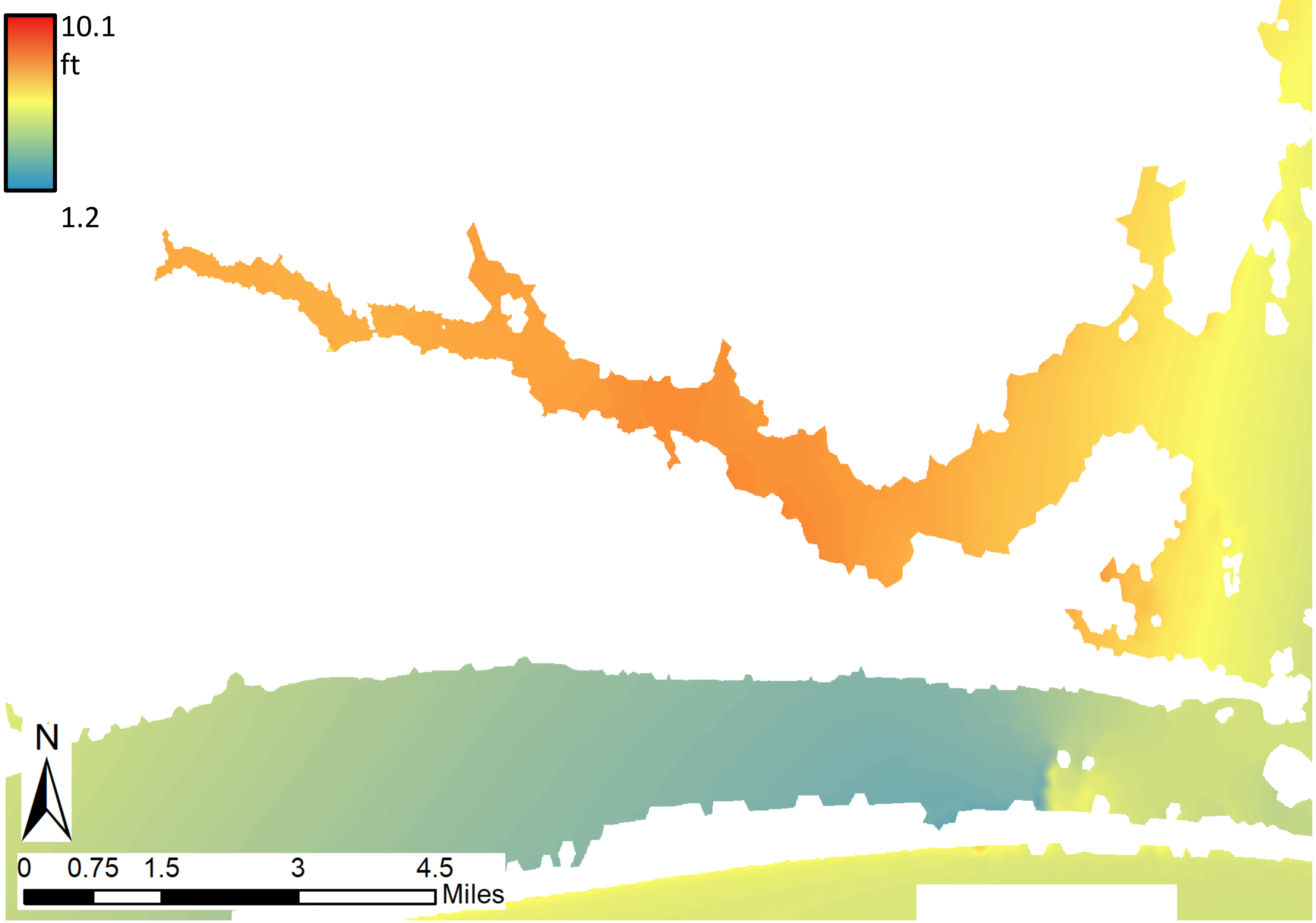
ADCIRC x values

$$\zeta_{slopes} = \zeta_{static} \pm c(m_x \Delta \boldsymbol{x}) \pm c(m_y \Delta y)$$



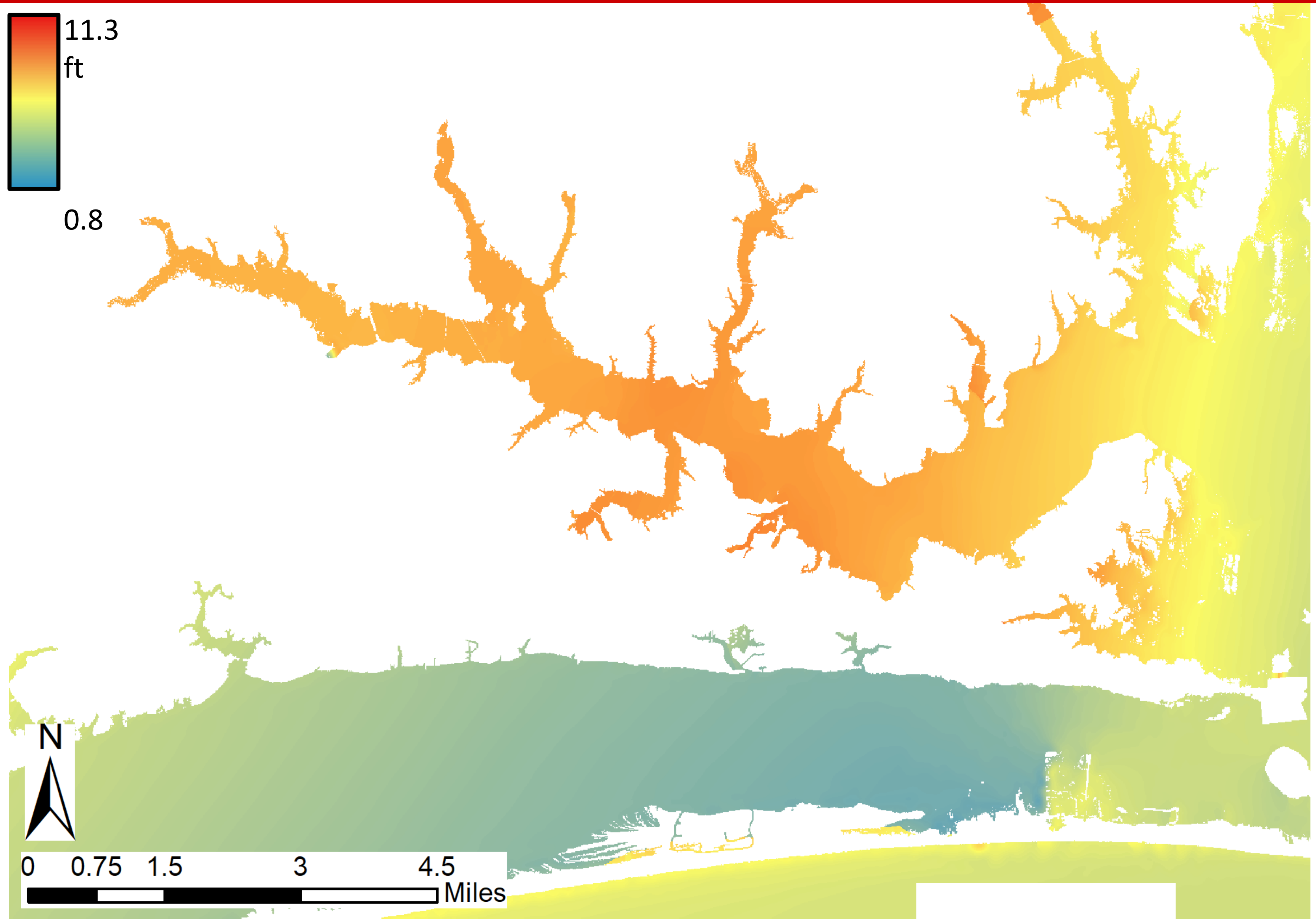
Δx

$$\zeta_{slopes} = \zeta_{static} \pm c(m_x \boldsymbol{\Delta x}) \pm c(m_y \Delta y)$$



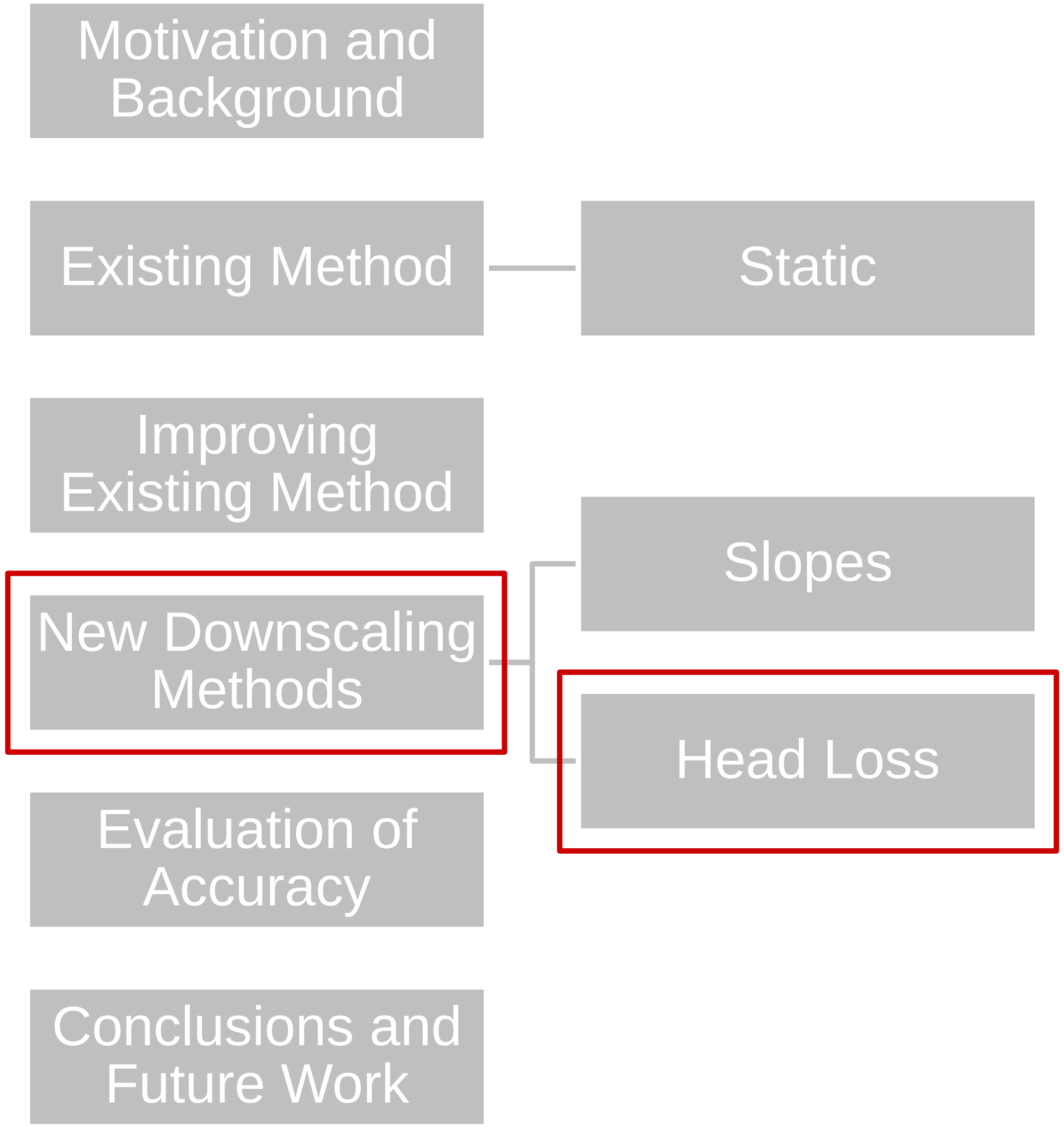
Water surface elevation (from v.surf.rst)

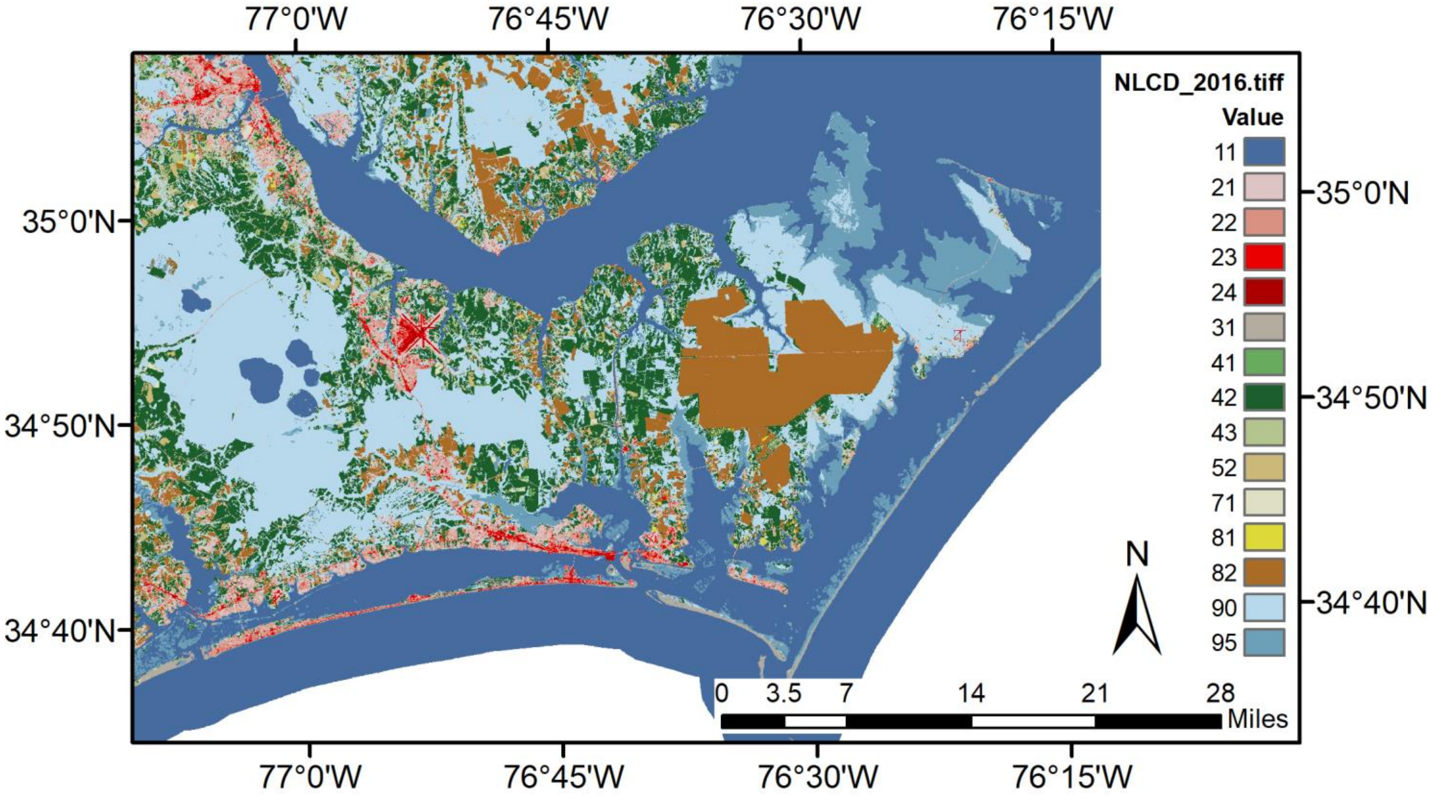
$$\zeta_{slopes} = \boxed{\boldsymbol{\zeta_{static}}} \pm c(m_x \Delta x) \pm c(m_y \Delta y)$$



Downscaled water surface elevations

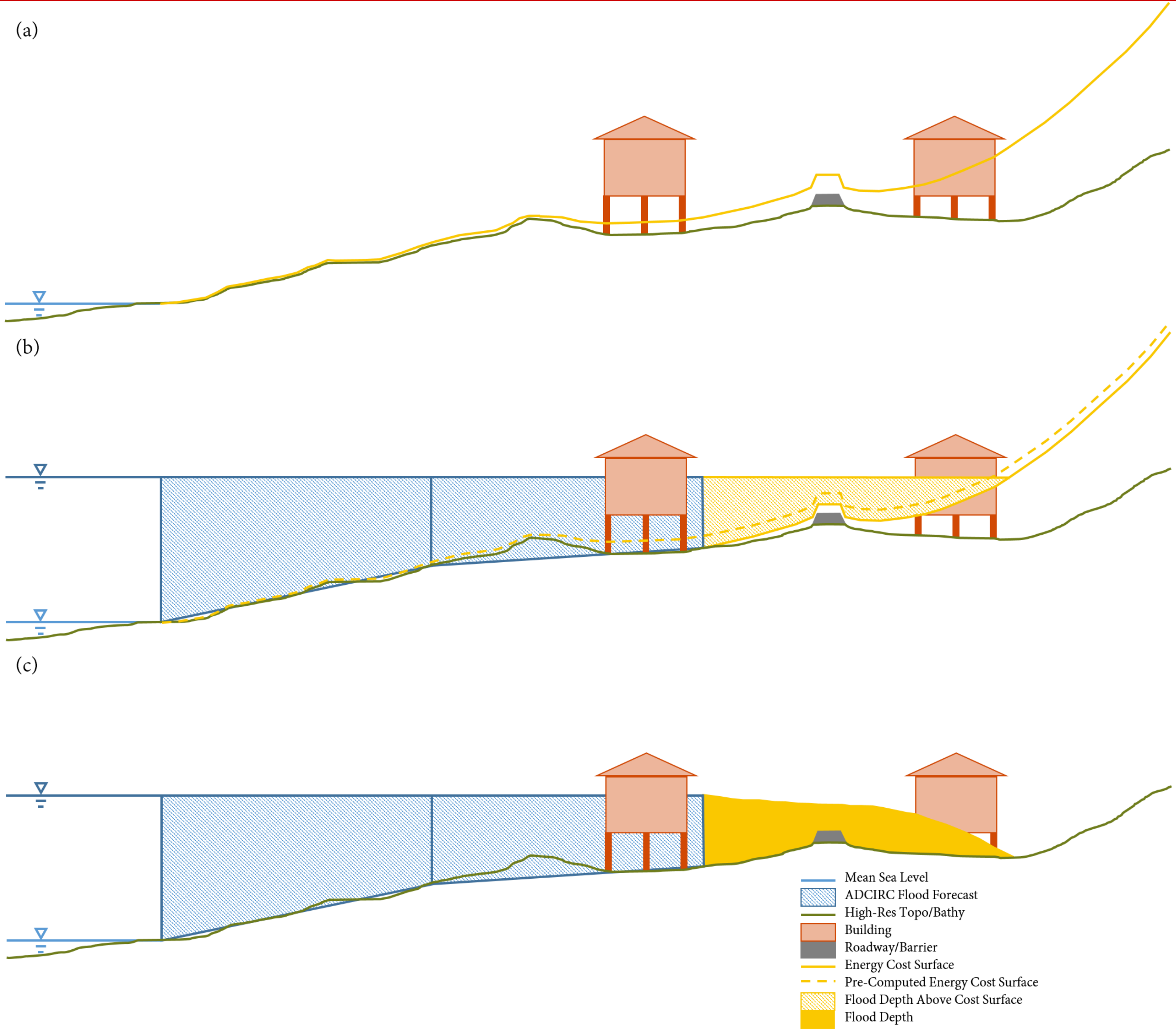
$$\zeta_{slopes} = \zeta_{static} \pm c(m_x \Delta x) \pm c(m_y \Delta y)$$


Motivation and Background
Existing Method
Static
Improving Existing Method
Slopes
New Downscaling Methods
Head Loss
Evaluation of Accuracy
Conclusions and Future Work



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

Liu et al. [2018] and Kalyanapu et al. [2009]


(a)
(b)
(c)
Mean Sea Level
ADCIRC Flood Forecast
High-Res Topo/Bathy
Building
Roadway/Barrier
Energy Cost Surface
Pre-Computed Energy Cost Surface
Flood Depth Above Cost Surface
Flood Depth

Head Loss Using Manning's Equation

- Manning's Equation:

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

  - U: water velocity
  - k: unit conversion factor (1 for SI; 1.49 for empirical)
  - n: Manning's friction coefficient
  - R: hydraulic radius (≈depth of flow)
  - S: slope of the energy grade line

Head Loss Using Manning's Equation

Manning's equation can be manipulated to directly calculate head loss by stating that *S* (slope of the energy grade line) is equal to head loss (h_L) divided by horizontal distance traveled (L)

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

Head Loss Method

Pre-Forecasting

- Before receiving input from ADCIRC
- Computation time is **not** important
- Goal: Create energy cost surface to use in forecasting
- Have: DEM, Manning's n
- Need: Flow paths, flood depths, water velocities

Forecasting

- After receiving input from ADCIRC
- Computation time **is** important
- Goal: Downscale ADCIRC results and distribute to emergency managers
- Have: Cost surface, ADCIRC water elevations
- Need: Water levels, depths, and velocities **from ADCIRC**

Calculating Accumulation of Head Loss

- Paths are entrained using the **r.walk** GRASS module
- A surface is generated containing the "**least cost**" of moving from MSL to any overland point throughout the region
- Uses the following general form:

$$cost_{total} = \Delta z + \sum L \left(\frac{nU}{kR^{2/3}} \right)^2$$

  - $cost_{total}$: energy head required to reach a certain point (ft, m)
  - Δz: change in elevation (ft, m)
  - Summation term: head loss (ft, m)

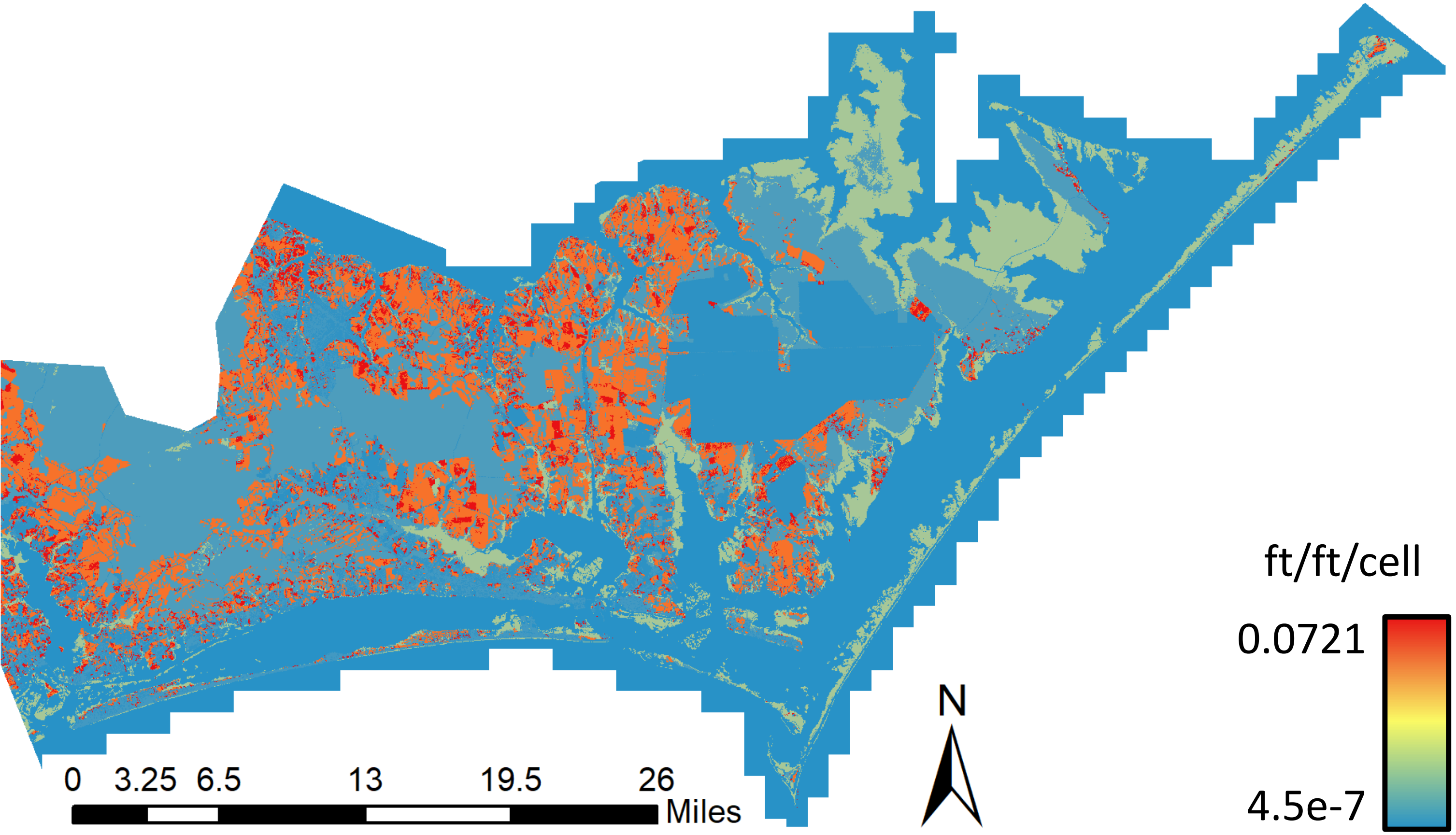
Pre-Forecasting r.walk Steps

- *U* and *R* are unknown during pre-forecasting entrainment
- A synthetic value $\boldsymbol{UR_{const}}$ is used for $U/R^{2/3}$ to entrain flow paths
- This research used UR_{const}=1
- The head loss portion becomes:

$$h_L = L\left(\frac{nU}{kR^{2/3}}\right)^2 = L\left(\frac{n(UR_{const})}{k}\right)^2$$

Pre-Forecasting r.walk Steps

- Each r.walk operation generates a least cost raster from MSL to **one** end point
- We need least cost to **all** possible end points
- Calculating least cost rasters for each overland null cell is redundant
- An array of endpoints with constant spacing is used
- Lowest values from each iteration are kept to create the least cost energy surface



Unit Head Loss (ft h_L/ft distance/cell)

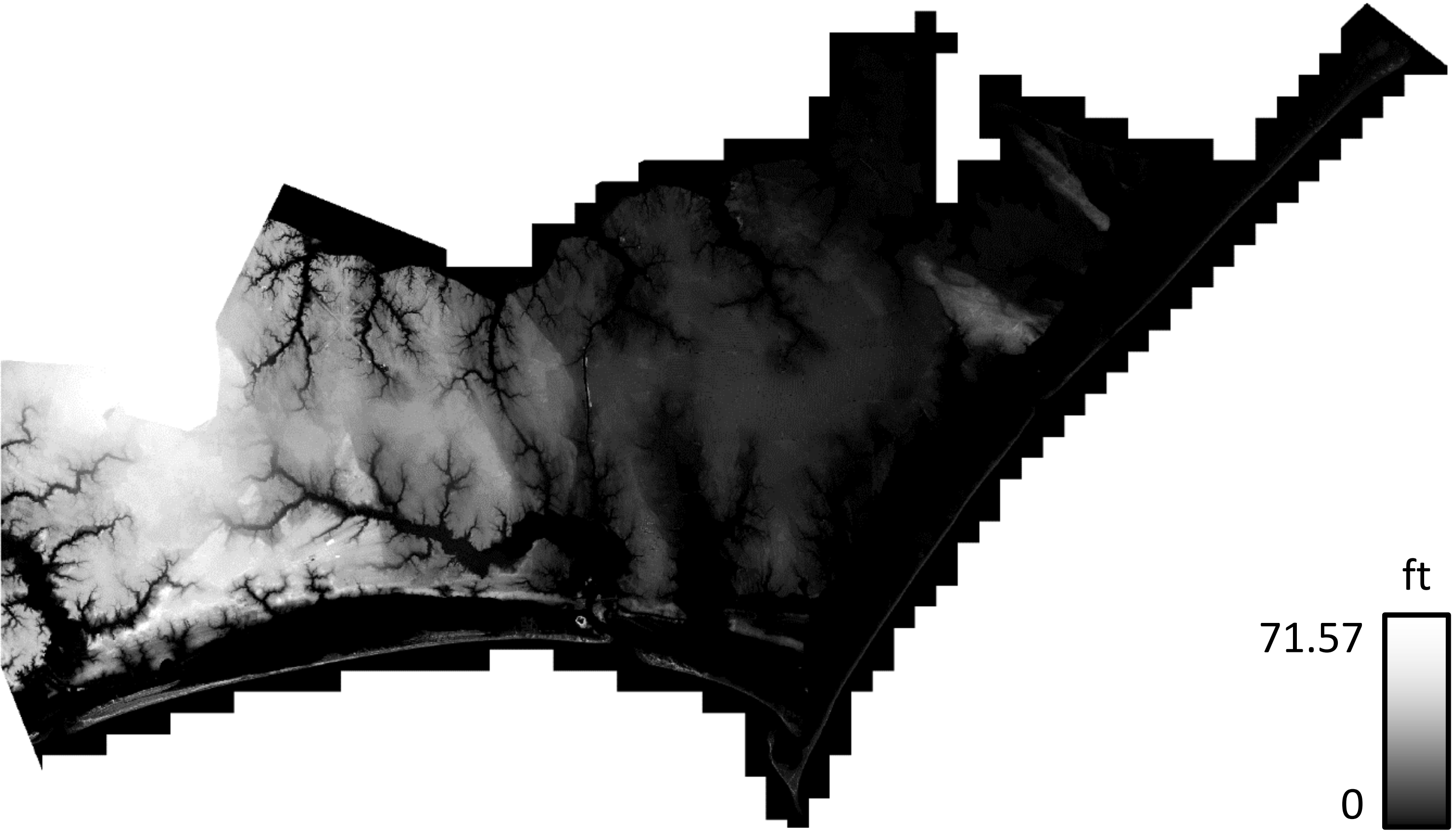
ft/ft/cell

0.0721

4.5e-7

$$cost_{total} = \Delta z + \sum L \left(\frac{\boldsymbol{n}(\boldsymbol{UR}_{\boldsymbol{const}})}{\boldsymbol{k}} \right)^2$$

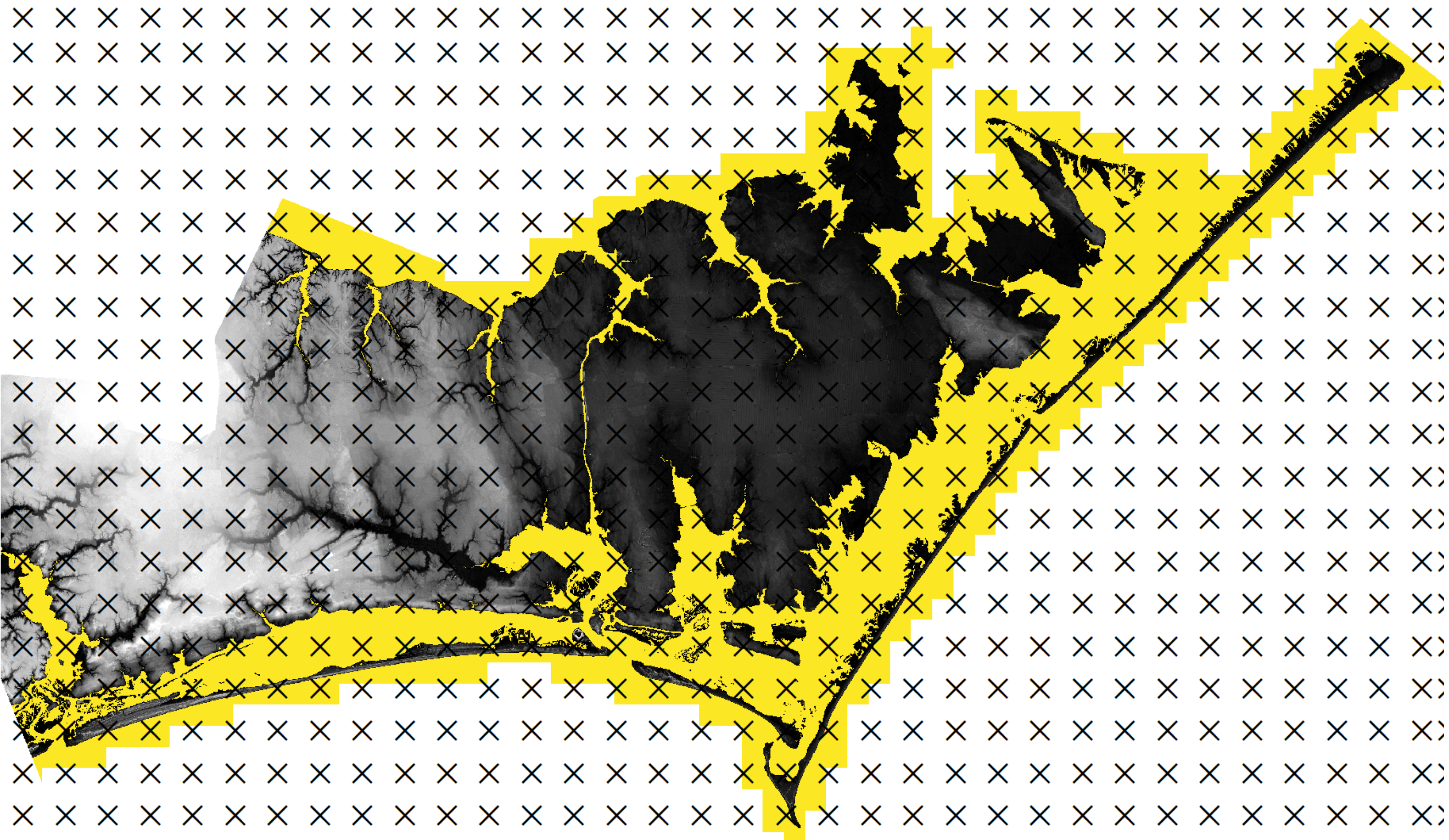
```
FOR each end point DO
    CALCULATE minimum cost from MSL to end point
    IF r.walk value < final cost surface THEN
        WRITE r.walk value to final cost surface
    END IF
END FOR
```



DEM

$$cost_{total} = \boxed{\Delta z} + \sum L\left(\frac{n(UR_{const})}{k}\right)^2$$

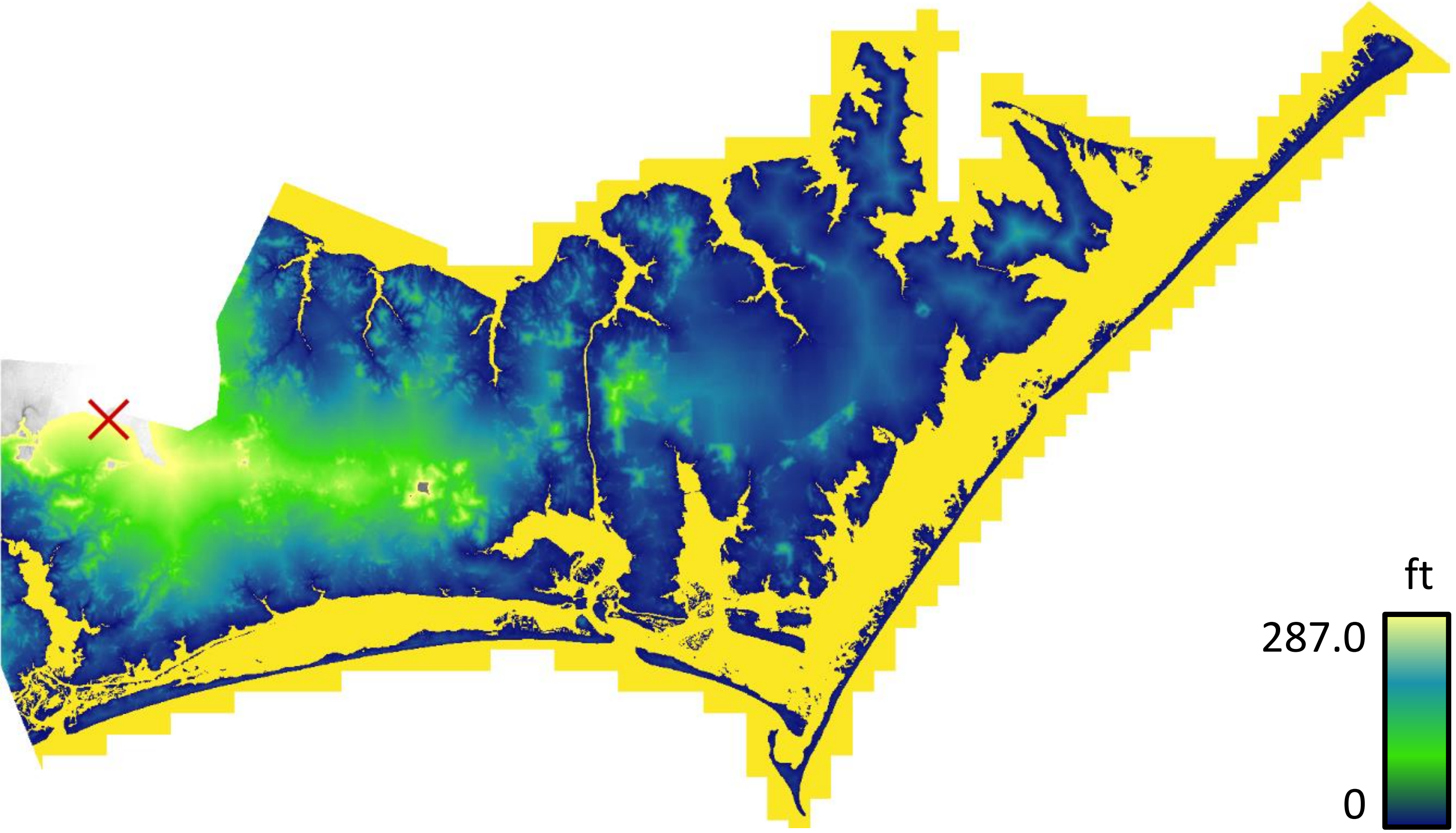
```
FOR each end point DO
    CALCULATE minimum cost from MSL to end point
    IF r.walk value < final cost surface THEN
        WRITE r.walk value to final cost surface
    END IF
END FOR
```



MSL (start, cost=0) and endpoints

$$cost_{total} = \Delta z + \sum L \left(\frac{n(UR_{const})}{k} \right)^2$$

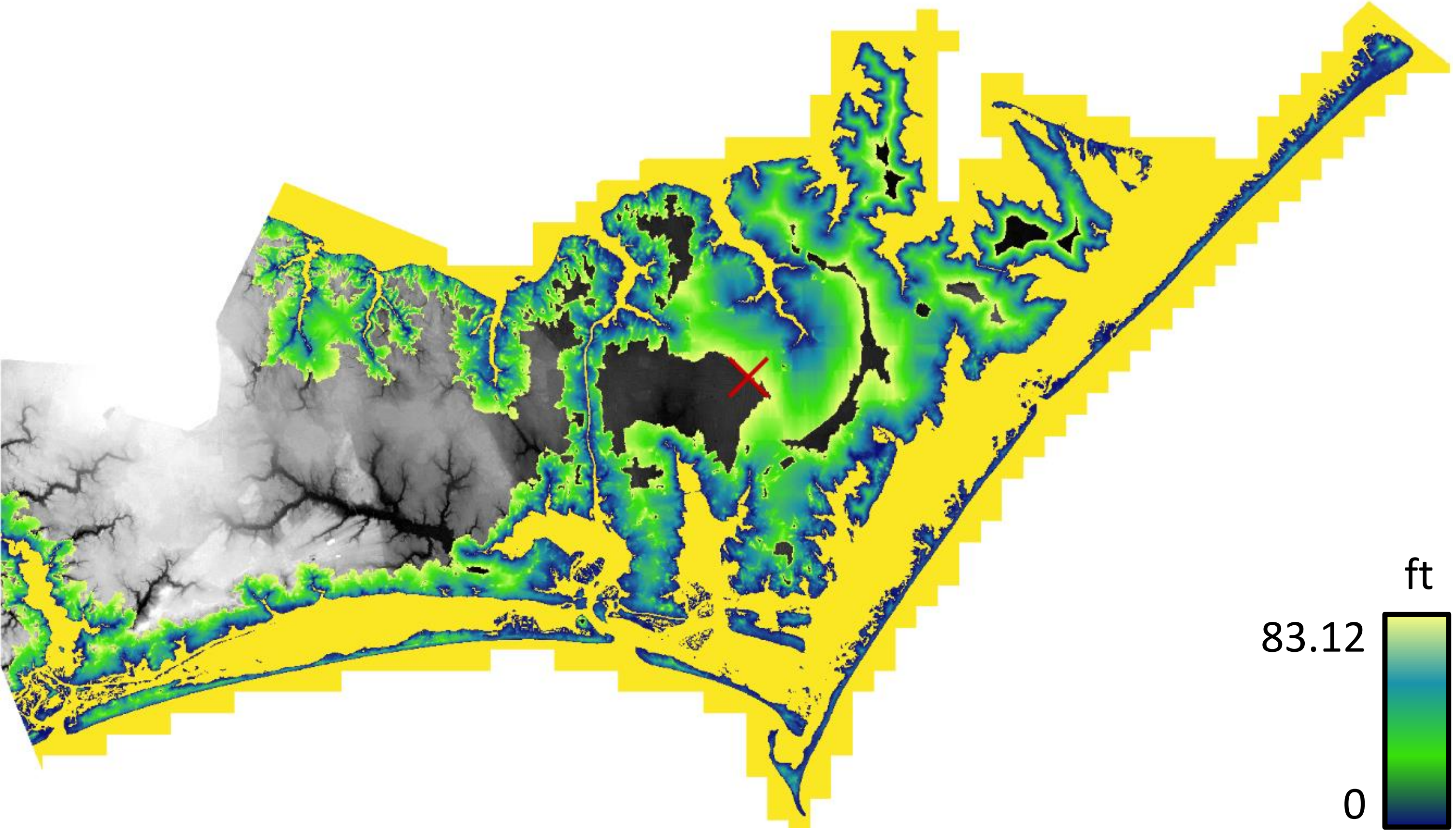
```
FOR each end point DO
     CALCULATE minimum cost from MSL to end point
     IF r.walk value < final cost surface THEN
          WRITE r.walk value to final cost surface
     END IF
END FOR
```



$$cost_{total} = \Delta z + \sum L \left(\frac{n(UR_{const})}{k} \right)^2$$

```
FOR each end point DO
    CALCULATE minimum cost from MSL to end point
    IF r.walk value < final cost surface THEN
        WRITE r.walk value to final cost surface
    END IF
END FOR
```

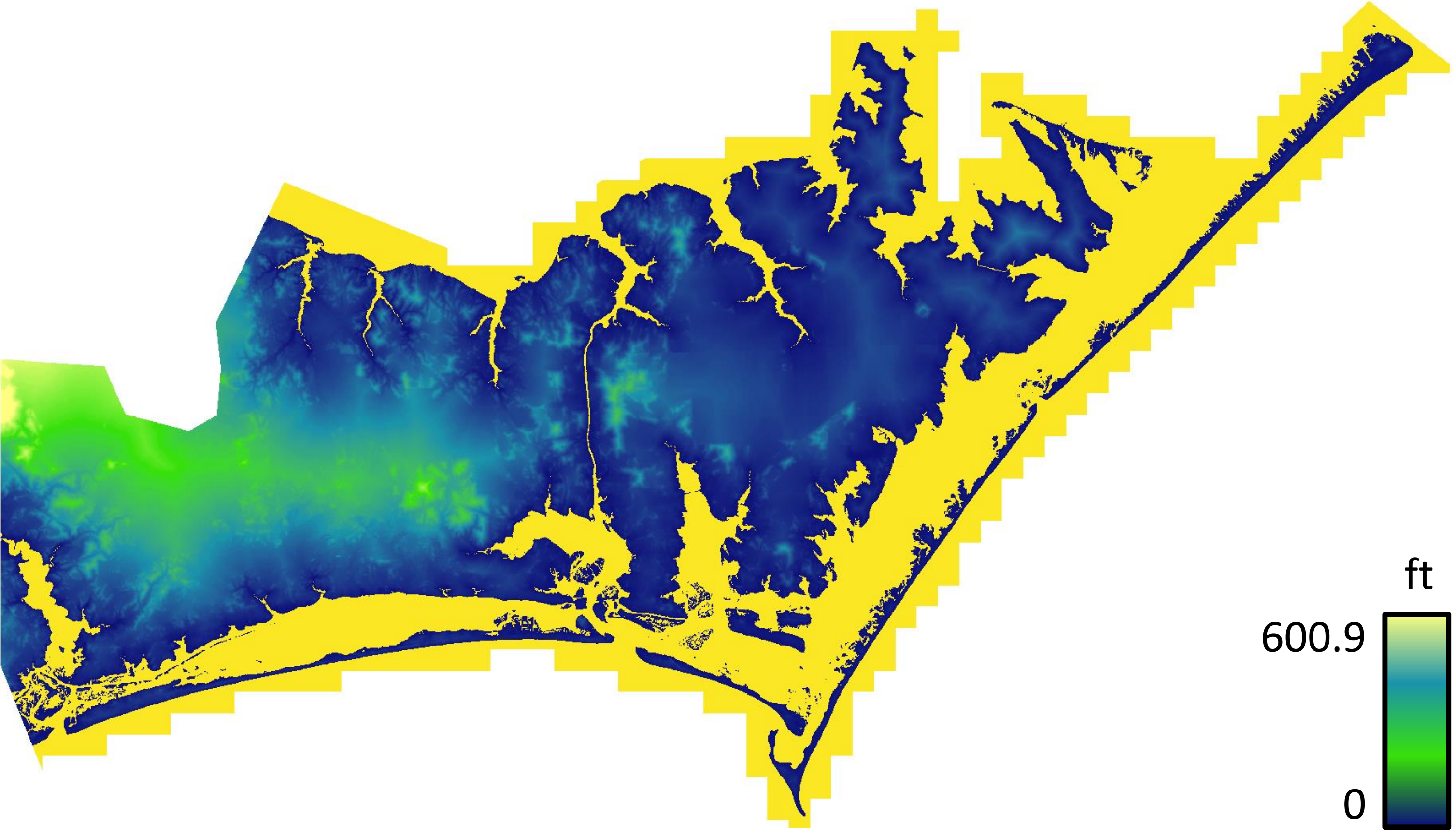
r.walk iteration; start from MSL, end at red X



r.walk iteration; start from MSL, end at red X

$$cost_{total} = \Delta z + \sum L \left(\frac{n(UR_{const})}{k}\right)^2$$

```
FOR each end point DO
    CALCULATE minimum cost from MSL to end point
    IF r.walk value < final cost surface THEN
        WRITE r.walk value to final cost surface
    END IF
END FOR
```



Final cost surface; contains lowest cost for each raster cell over all iterations

$$cost_{total} = \Delta z + \sum L \left(\frac{n(UR_{const})}{k} \right)^2$$

```
FOR each end point DO
    CALCULATE minimum cost from MSL to end point
    IF r.walk value < final cost surface THEN
        WRITE r.walk value to final cost surface
    END IF
END FOR
```

Pre-Forecasting r.walk Steps

- Remove the synthetic UR_{const} values by taking the resulting total cost raster, subtracting the DEM elevations, and dividing by $(UR_{const})^2$
- Now cumulative head loss values are stripped to the following form:

$$cost_{raw} = \sum L\left(\frac{n}{k}\right)^2$$

Head Loss Method

Pre-Forecasting

- Before receiving input from ADCIRC
- Computation time is **not** important
- Goal: Create energy cost surface to use in forecasting
- Have: DEM, Manning's n
- Need: Flow paths, flood depths, water velocities

Forecasting

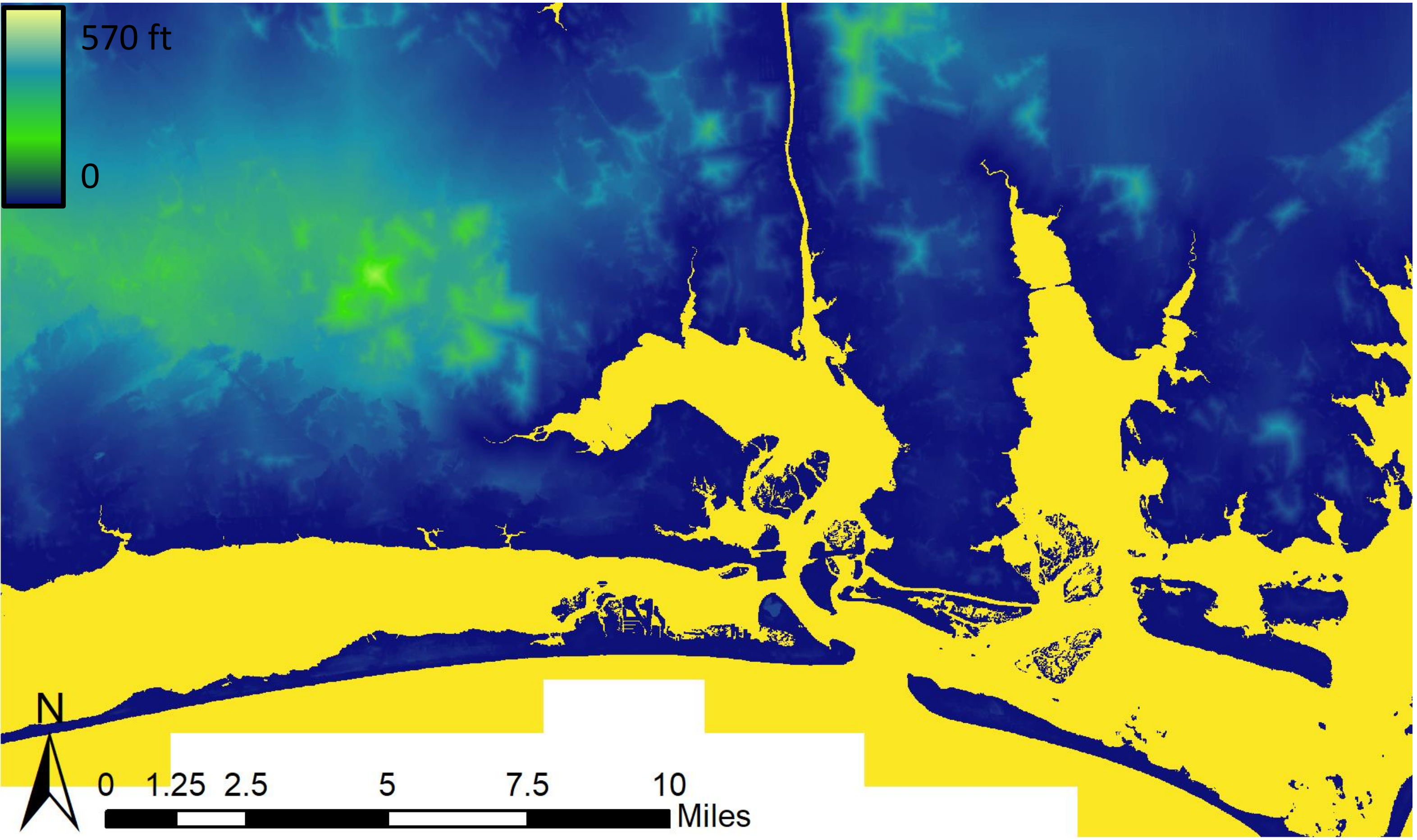
- After receiving input from ADCIRC
- Computation time **is** important
- Goal: Downscale ADCIRC results and distribute to emergency managers
- Have: Cost surface, ADCIRC water elevations
- Need: Water levels, depths, and velocities **from ADCIRC**

Forecasting with Head Loss

- Use ADCIRC water elevations to calculate R_{avg}

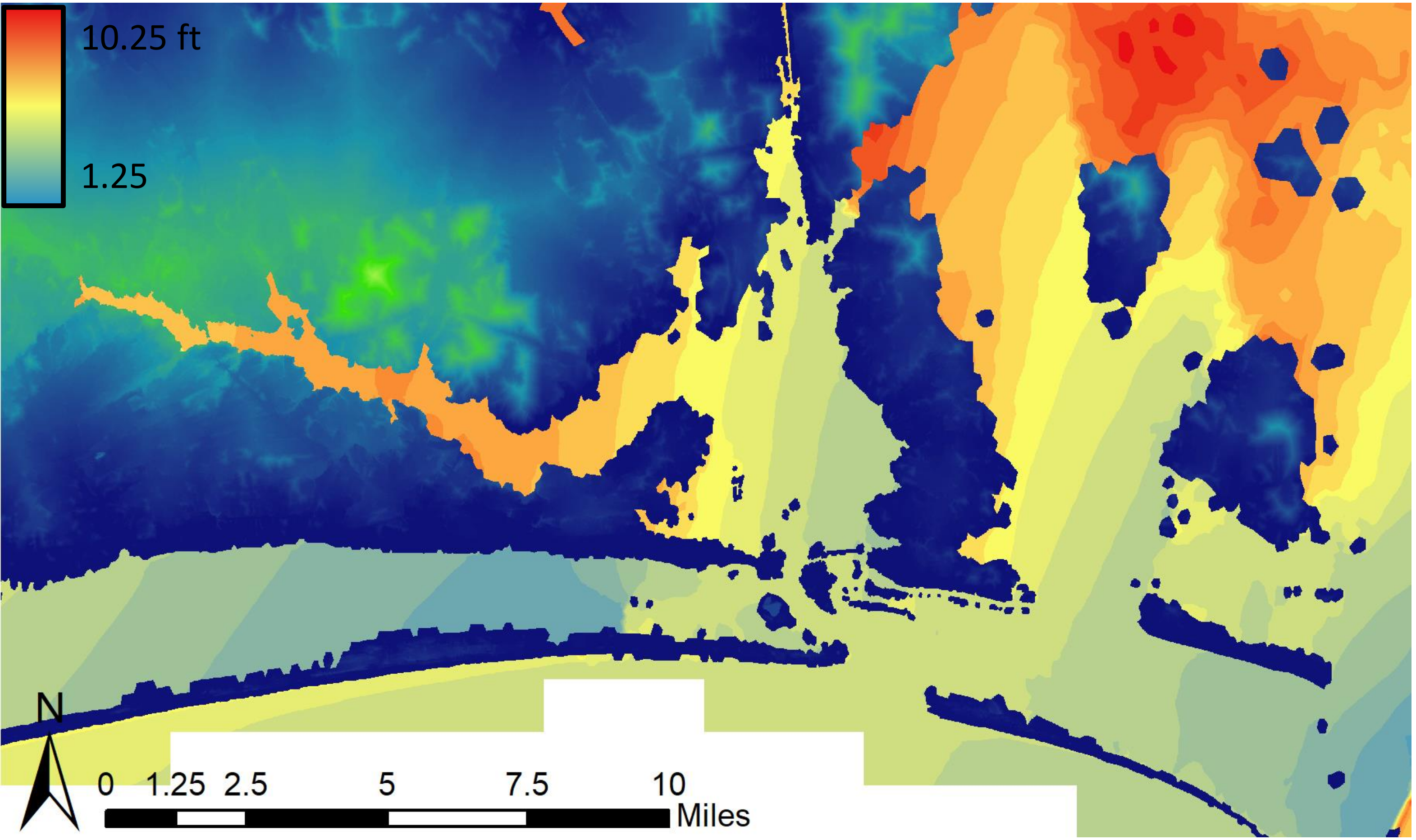
$$R_{avg} = \frac{1}{2}\left((\zeta_{ADCIRC} - z_{DEM})_{raster} + (\zeta_{ADCIRC} - z_{DEM})_{MSL}\right)$$

- No-flow condition exists in ADCIRC at wet/dry boundary; velocities at this divide are negligible
- A constant value is used for *U*; for simplicity, this research uses *U*=1
- Multiply $cost_{raw}$ by $\left(\frac{1}{R_{avg}^{\frac{2}{3}}}\right)^2$ and constant U^2 to get full h_L equation



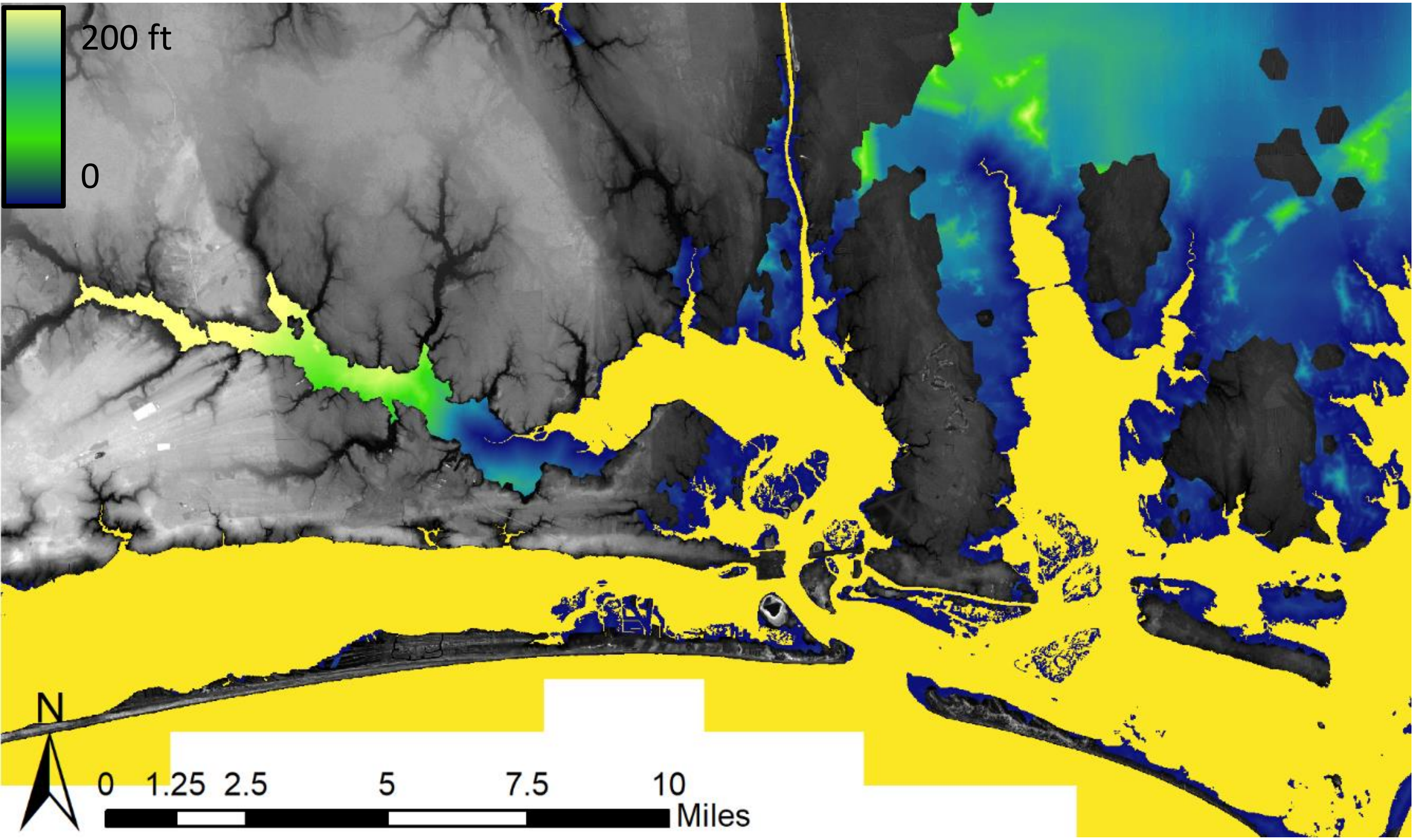
$cost_{raw}$ and MSL

Start with raw cost raster
Import ADCIRC water levels
Extract cost accounted for by ADCIRC
Extrapolate ADCIRC costs
Extrapolate ADCIRC water levels
Calculate R_{avg}
Generate downscaled water levels



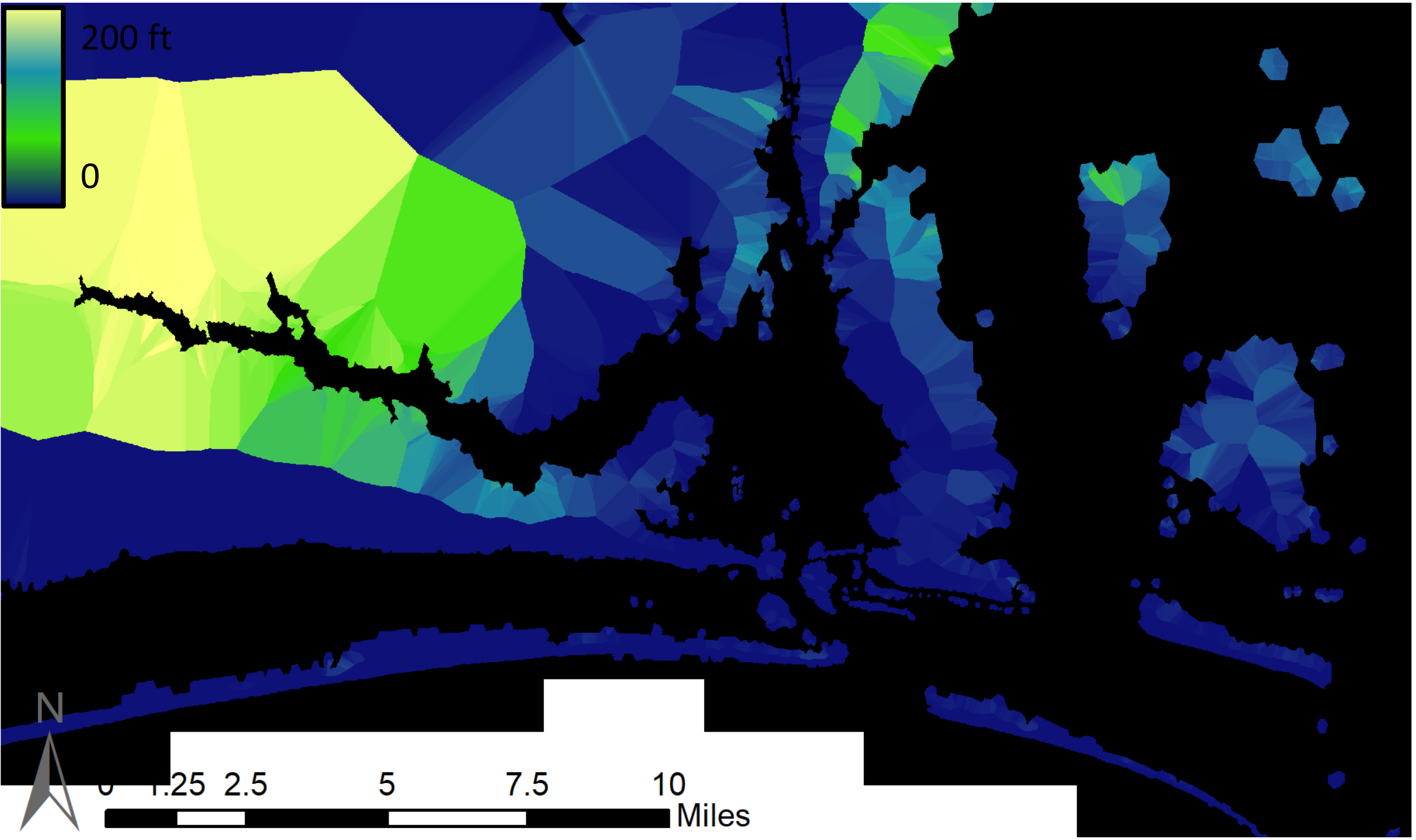
$cost_{raw}$ and ADCIRC water levels

Start with raw cost raster
Import ADCIRC water levels
Extract cost accounted for by ADCIRC
Extrapolate ADCIRC costs
Extrapolate ADCIRC water levels
Calculate R_{avg}
Generate downscaled water levels



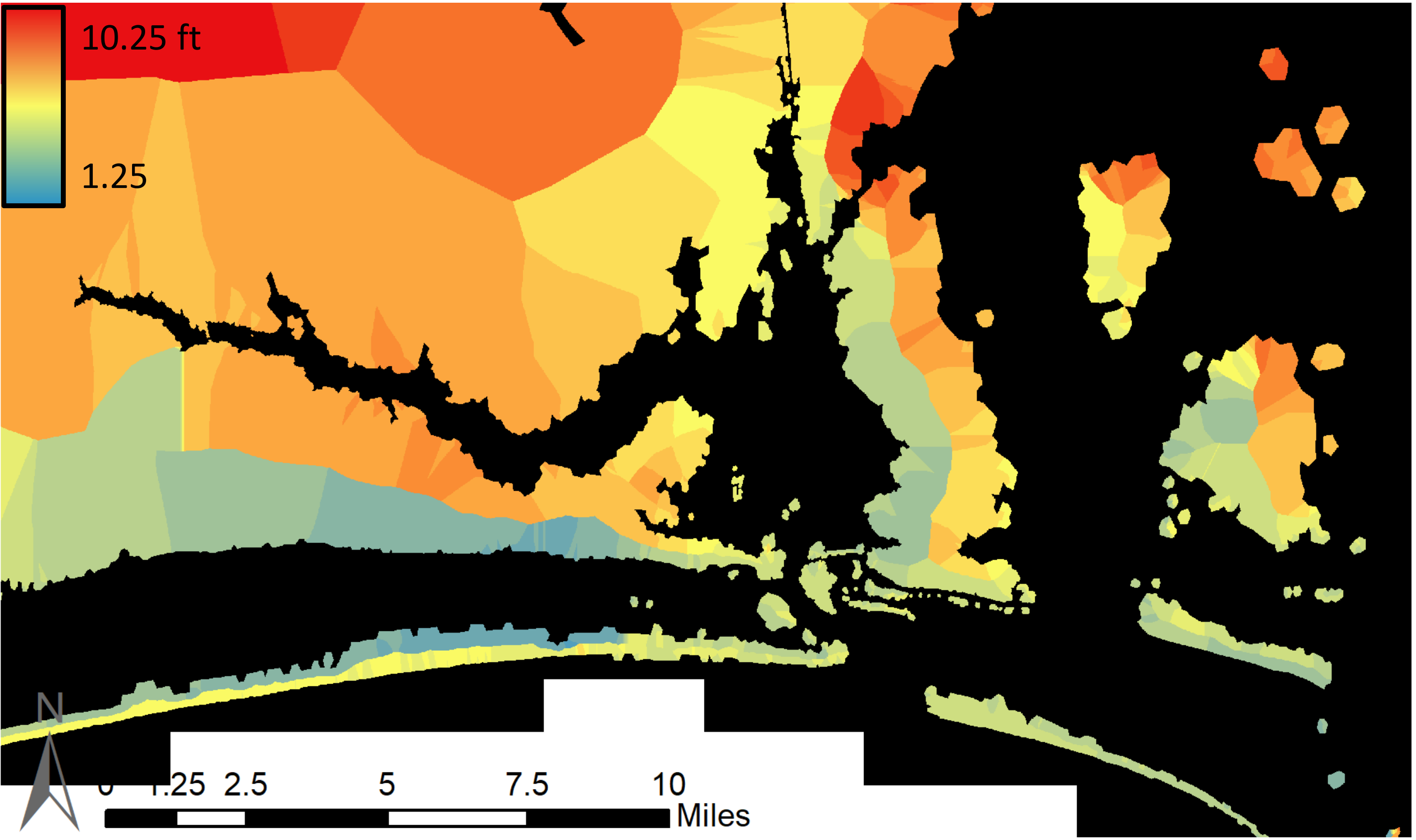
$cost_{raw}$ covered by ADCIRC

Start with raw cost raster
Import ADCIRC water levels
Extract cost accounted for by ADCIRC
Extrapolate ADCIRC costs
Extrapolate ADCIRC water levels
Calculate R_{avg}
Generate downscaled water levels



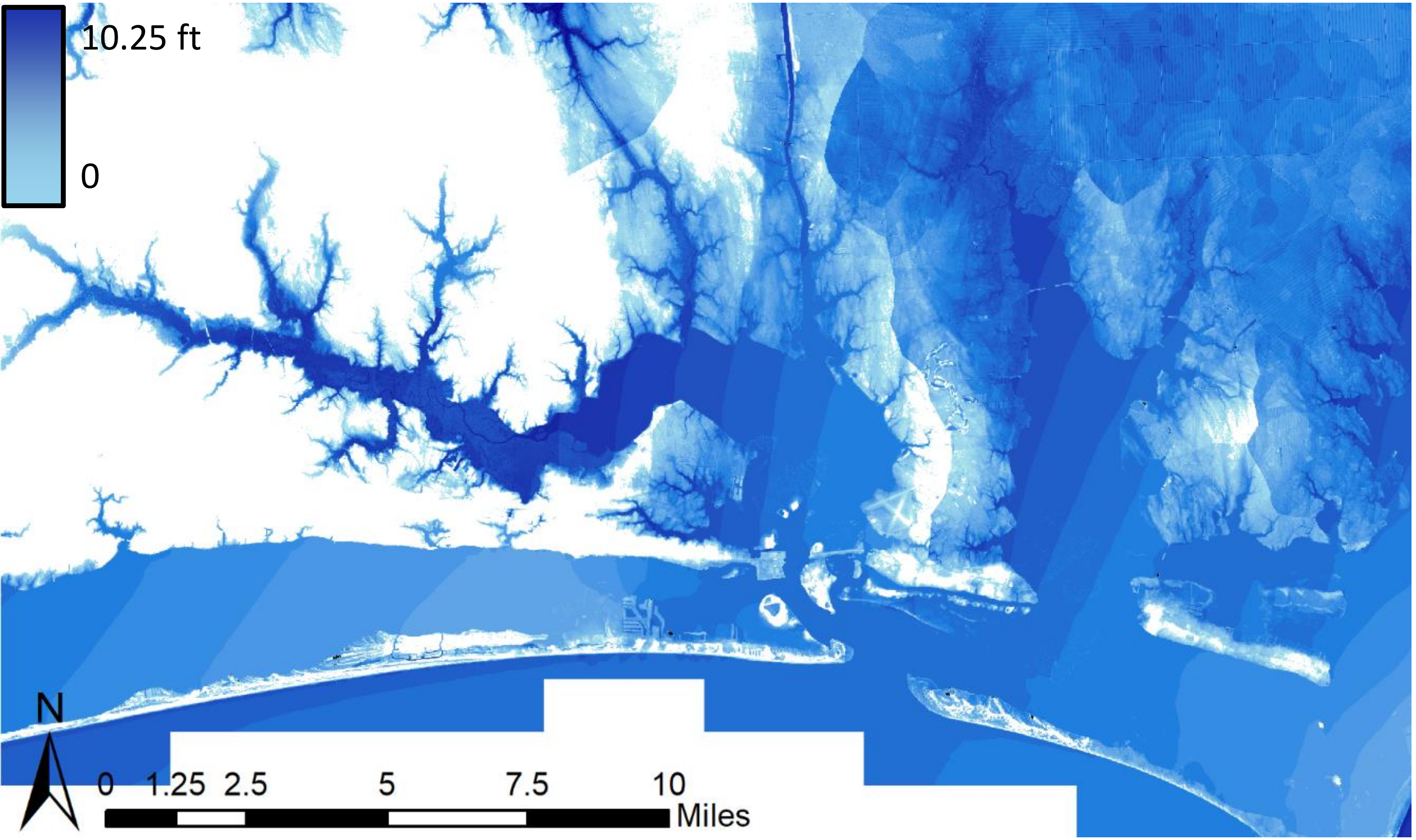
$cost_{raw}$ covered by ADCIRC, *extrapolated*

Start with raw cost raster
Import ADCIRC water levels
Extract cost accounted for by ADCIRC
Extrapolate ADCIRC costs
Extrapolate ADCIRC water levels
Calculate R_{avg}
Generate downscaled water levels



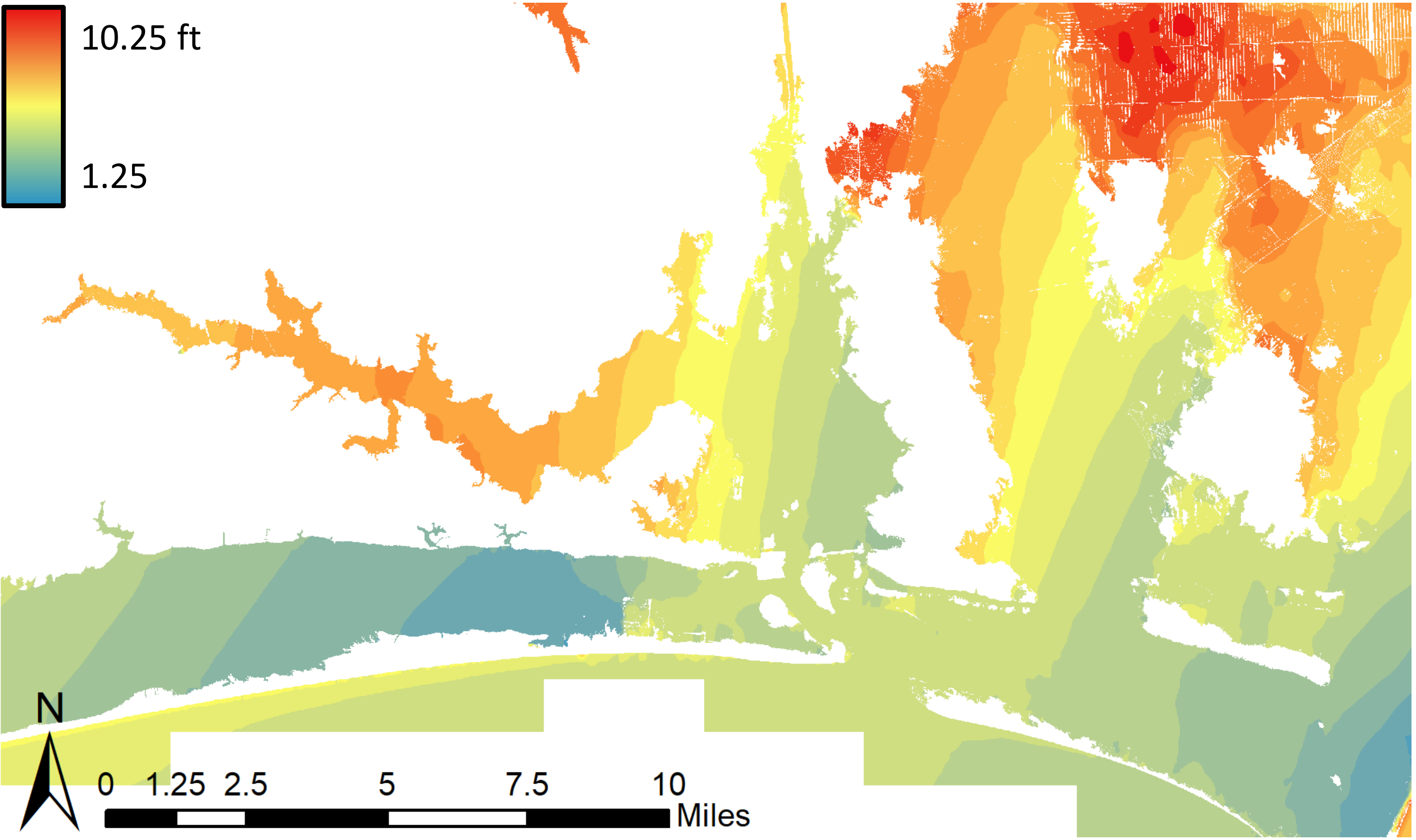
ADCIRC water levels, *extrapolated* horizontally

Start with raw cost raster
Import ADCIRC water levels
Extract cost accounted for by ADCIRC
Extrapolate ADCIRC costs
Extrapolate ADCIRC water levels
Calculate R_{avg}
Generate downscaled water levels



R_{avg}

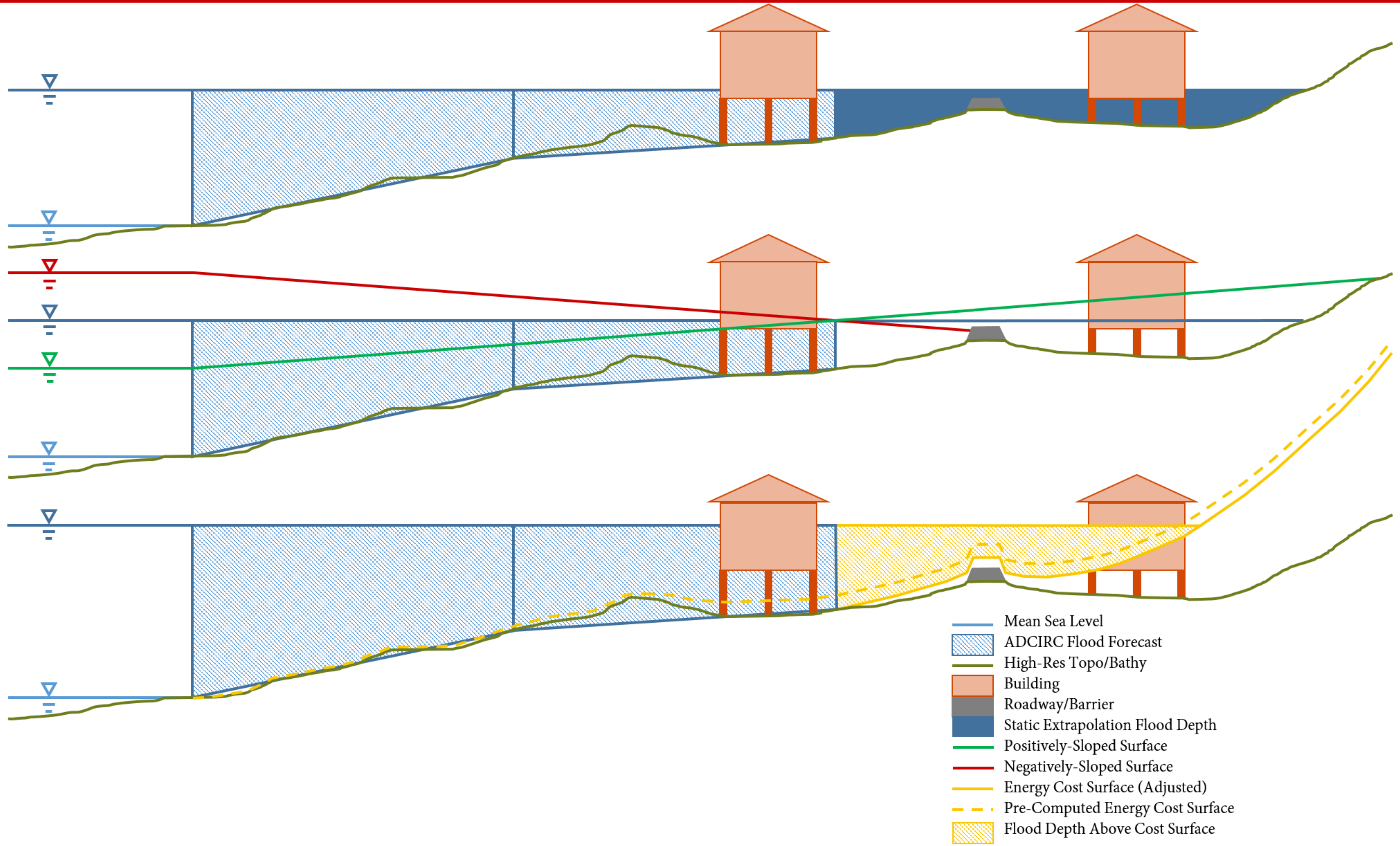
Start with raw cost raster
Import ADCIRC water levels
Extract cost accounted for by ADCIRC
Extrapolate ADCIRC costs
Extrapolate ADCIRC water levels
Calculate R_{avg}
Generate downscaled water levels



Downscaled water surface elevations

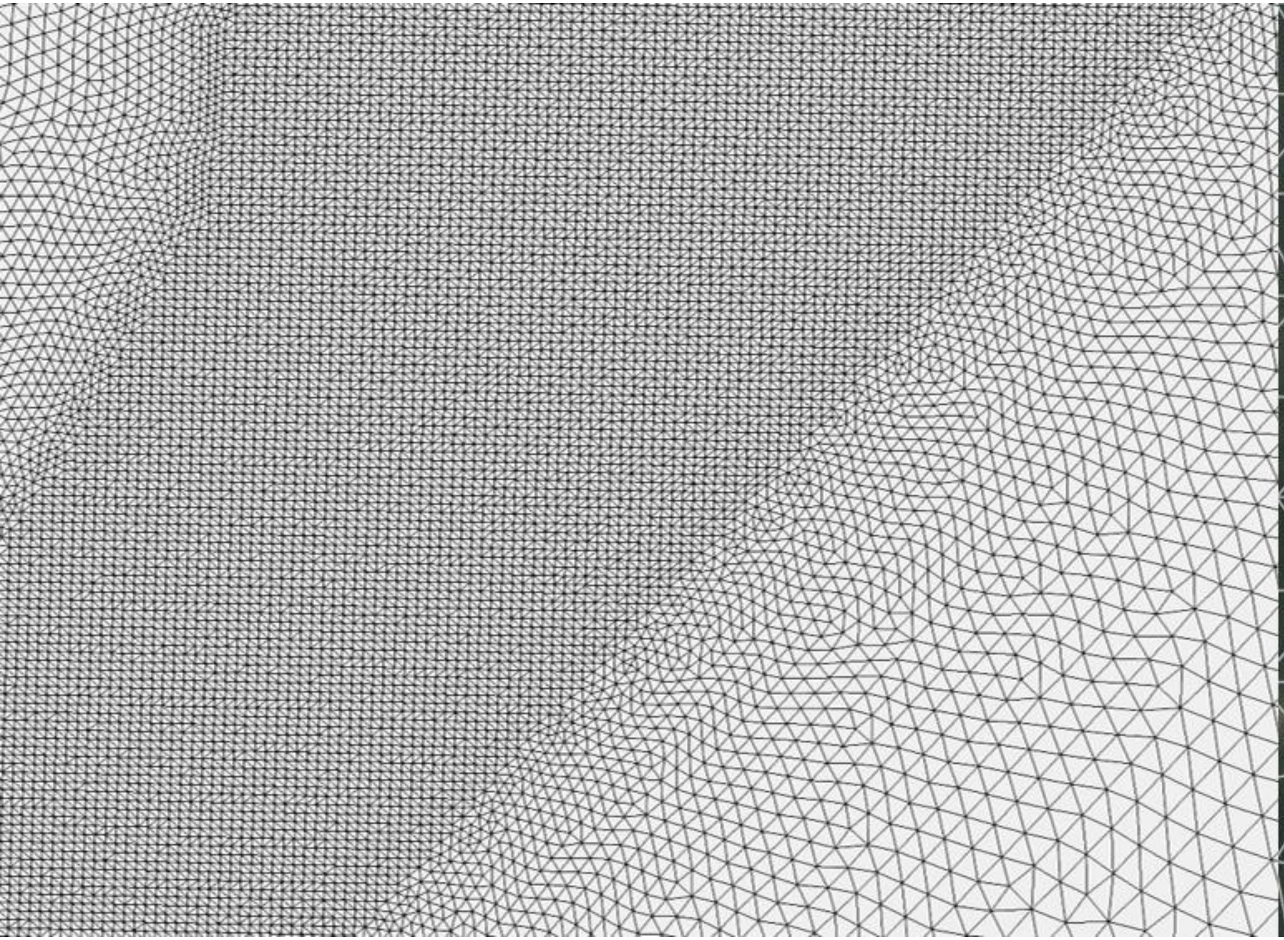
Start with raw cost raster
Import ADCIRC water levels
Extract cost accounted for by ADCIRC
Extrapolate ADCIRC costs
Extrapolate ADCIRC water levels
Calculate R_{avg}
Generate downscaled water levels

- Motivation and Background
- Existing Method
  - Static
- Improving Existing Method
- New Downscaling Methods
  - Slopes
  - Head Loss
- **Evaluation of Accuracy**
- Conclusions and Future Work

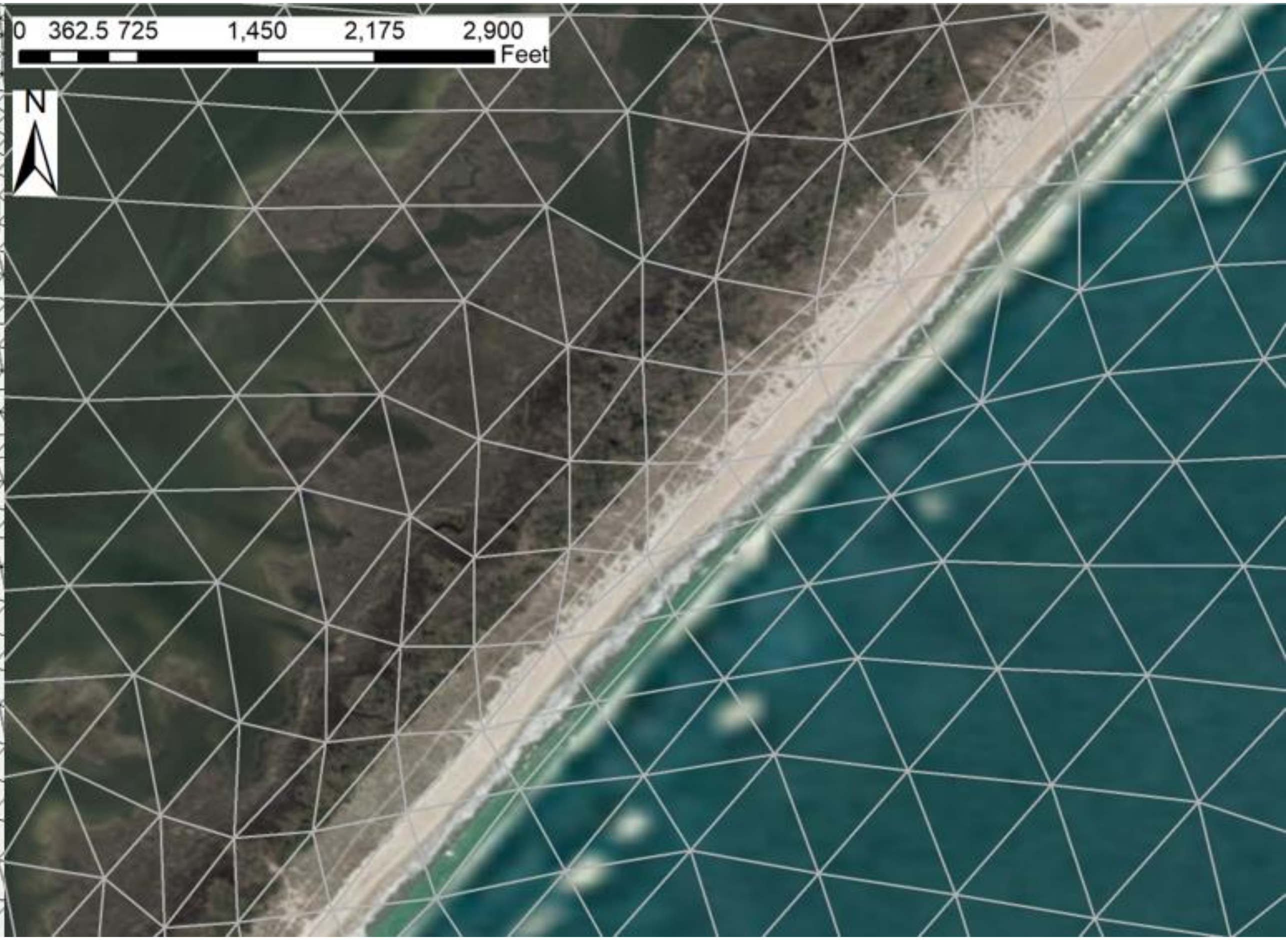

Mean Sea Level
ADCIRC Flood Forecast
High-Res Topo/Bathy
Building
Roadway/Barrier
Static Extrapolation Flood Depth
Positively-Sloped Surface
Negatively-Sloped Surface
Energy Cost Surface (Adjusted)
Pre-Computed Energy Cost Surface
Flood Depth Above Cost Surface

Evaluation Using High Resolution ADCIRC Mesh

- **High resolution ADCIRC mesh was used as “truth”**
- Developed using the NC9 mesh, which is input for downscaling
- Completely identical, except high resolution mesh vertices align with each cell in the DEM raster for Carteret County, NC
  - NC9 mesh: **622,946** vertices, 1,230,430 elements
  - High resolution mesh: **6,772,170** vertices, 13,528,879 elements
- Both models were run for Hurricane Florence (2018)
- Each model uses the same exact input parameters



High Resolution Mesh



NC9 Mesh

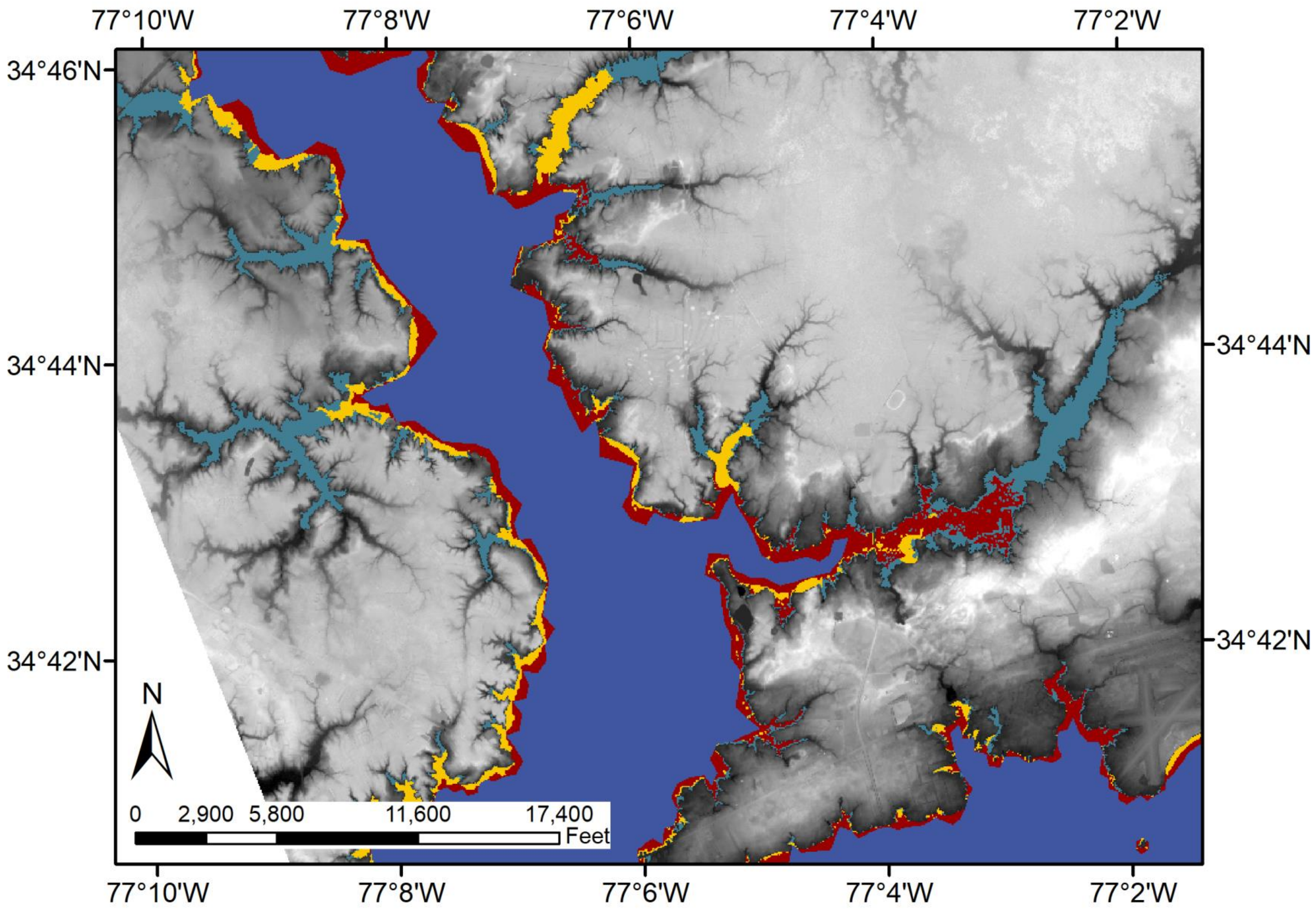
Legend

ADCIRC (NC9)

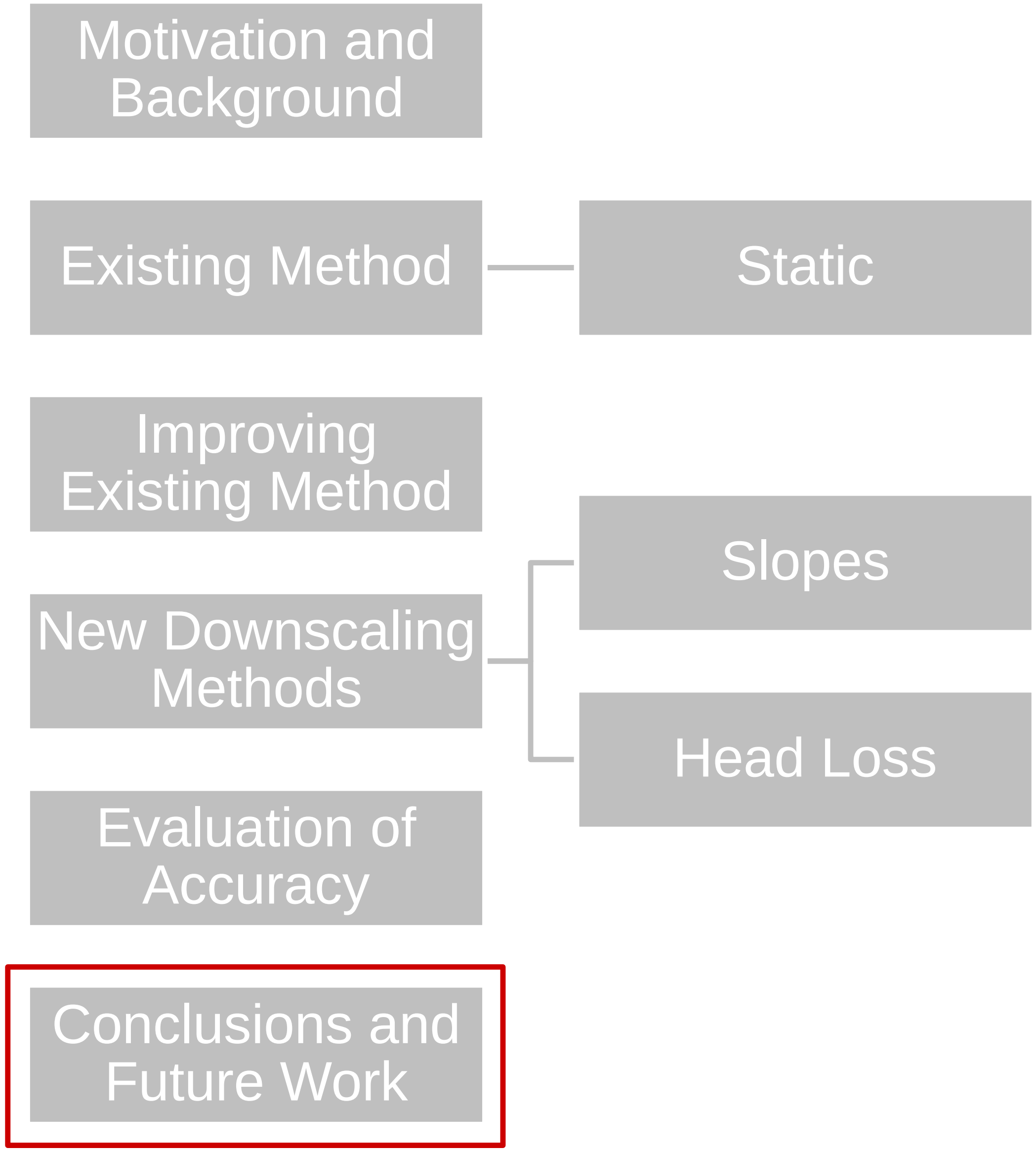
ADCIRC (high resolution)

Head loss method

Static method



Mesh	Downscaling Method	Flooded (acres)	Flooded, outside NC9 (acres)	Over-estimation, outside NC9 (acres)	Under-estimation, outside NC9 (acres)
NC9	---	157,314	---	---	---
NC9	Static	174,203	**23,324**	13,989	79
NC9	Slopes	175,358	**24,006**	14,655	62
NC9	Head Loss	162,579	**11,729**	5,573	3,258
High Resolution	---	126,593	**9,414**	---	---


Motivation and Background
Existing Method
Static
Improving Existing Method
Slopes
New Downscaling Methods
Head Loss
Evaluation of Accuracy
Conclusions and Future Work

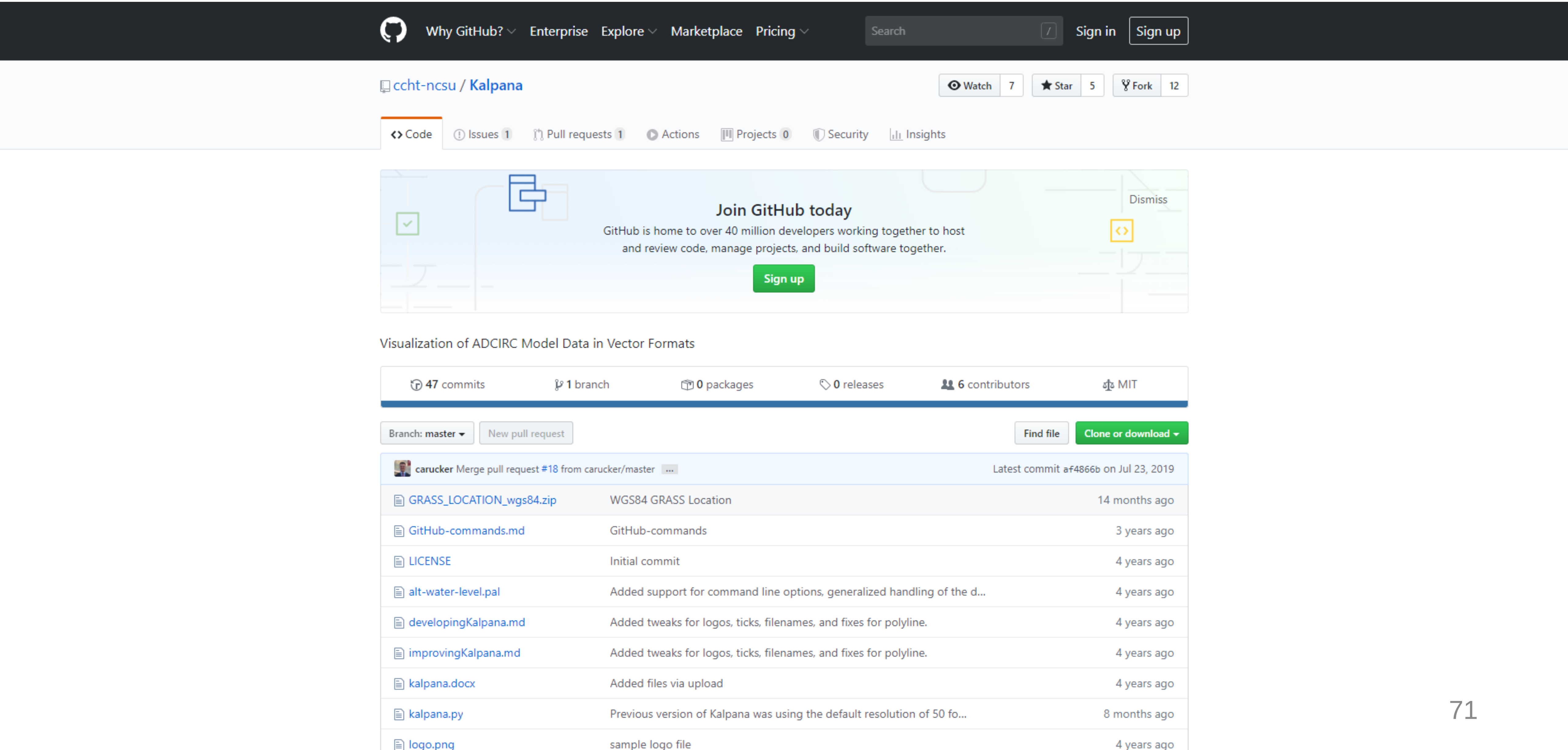
Conclusions

1. Integrating the downscaling methods with Kalpana allows users to apply methods throughout the world, using any mesh or DEM
2. The static method over-predicts water level extents
3. The slopes method did not improve the downscaling simulations, but could be useful in conjunction with other methods
4. The head loss method performed best and allows for the most flexibility

Future Work

- Apply new downscaling methods in **forecasting**
- Optimize downscaling method parameters
  - Test downscaling methods in other regions and for different storms
  - Adjust parameters manually or by using machine learning or statistical methods
  - **Parameters were not adjusted as a part of this thesis**; only one set of parameters was tested

github.com/ccht-ncsu/Kalpana

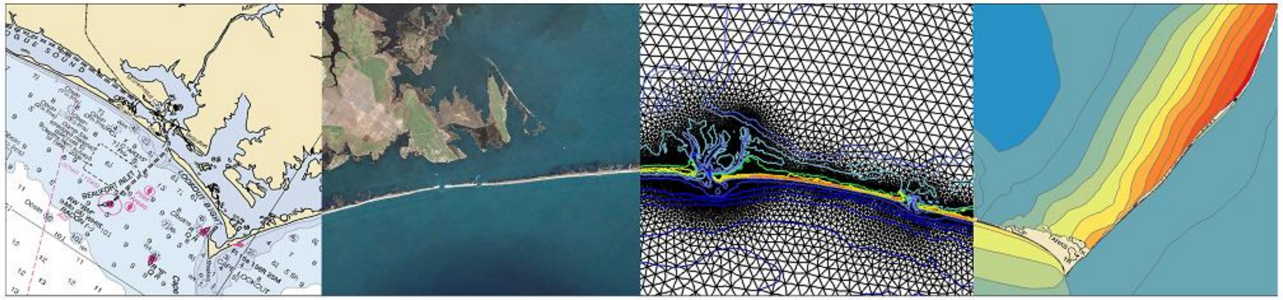


NC STATE UNIVERSITY

ccht.ccee.ncsu.edu/kalpana

NC STATE UNIVERSITY

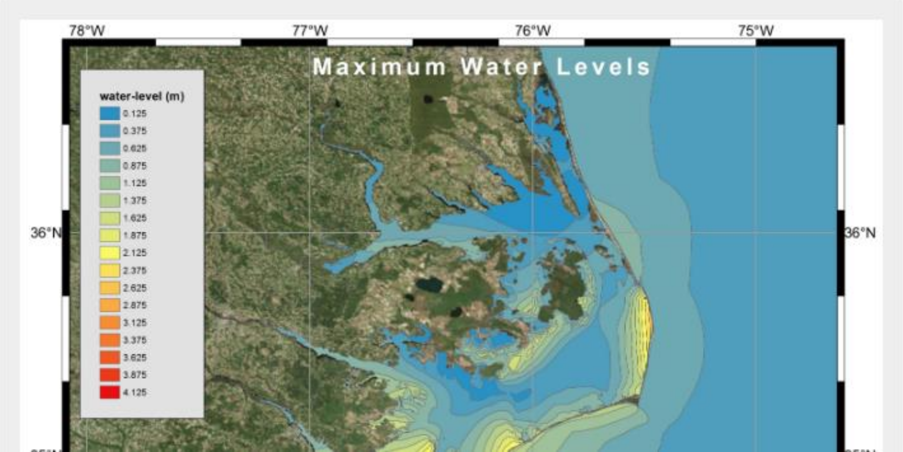
Coastal & Computational Hydraulics Team



What We Do | Join Our Team | FigureGen | **Kalpana** | SWAN+ADCIRC

Kalpana

Kalpana is a Python script that converts ADCIRC output files to ArcGIS compatible shapefiles and Google Earth compatible KMZ files. The code accepts NetCDF formatted ADCIRC outputs for maximum water levels, wind speeds, wave heights and peak wave period and converts these to polyline/polygon shapefiles and polygon KMZ files. The code is also capable of converting timeseries ADCIRC outputs for water levels, wind speeds and wave heights into polygon shapefiles.



MEET OUR TEAM

Faculty

- **Casey Dietrich** | Posts | CV

Post-Doctoral Researchers

- **Dylan Anderson** | Posts | Web

Graduate Students

- **Ajimon Thomas** | Posts | CV
- **Alireza Gharagozlou** | Posts | CV
- **Johnathan Woodruff** | Posts | CV
- **Autumn Poisson** | Posts | CV
- **Carter Rucker** | Posts | CV
- **Brooke Rumbaugh** | Posts | CV

Researchers

- Sheppard Medlin

Undergraduate Students

- Chloe Stokes
- Carter Howe
- Jessica Gorski

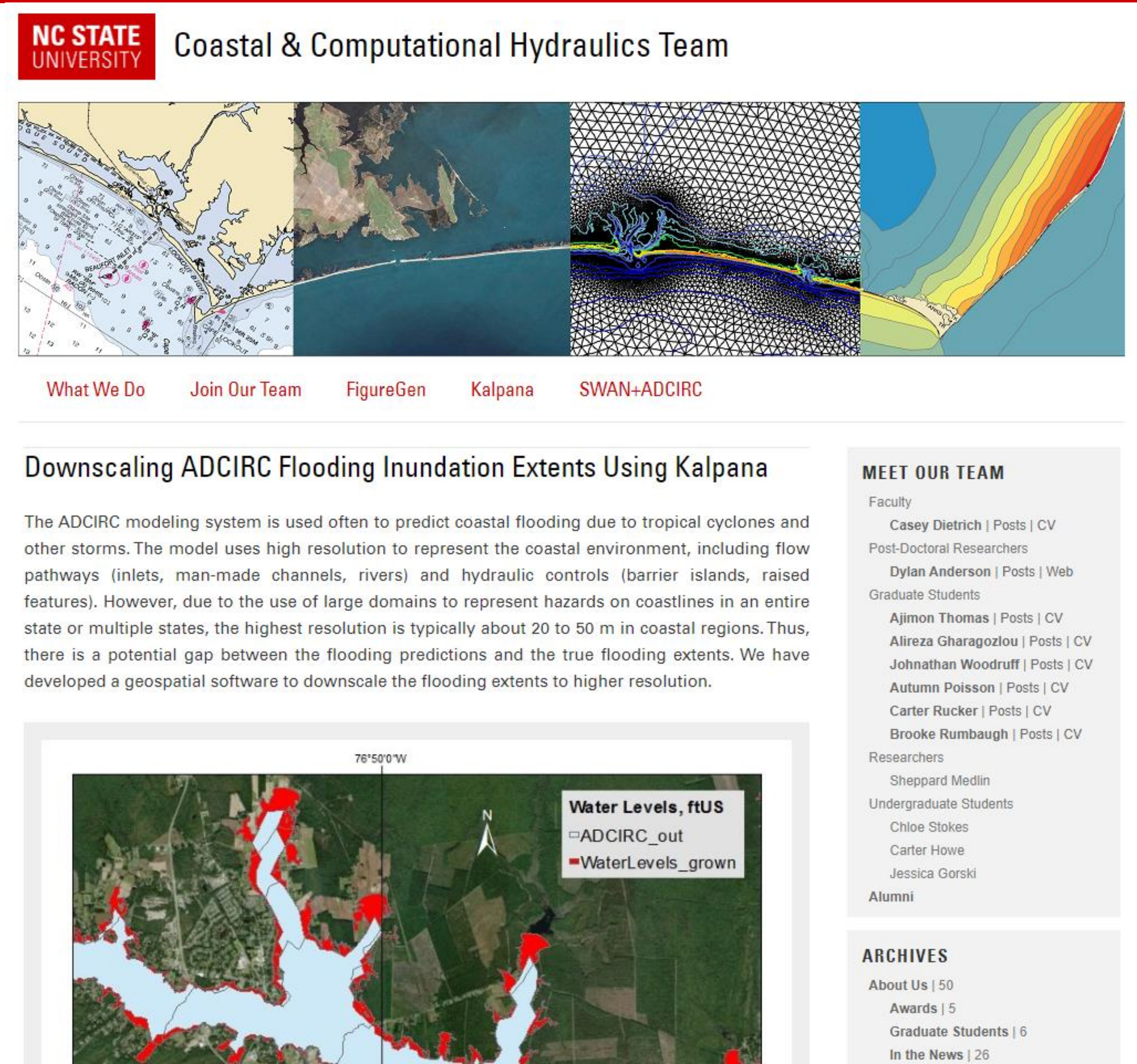
Alumni

ARCHIVES

About Us | 50

- **Awards** | 5
- **Graduate Students** | 6
- **In the News** | 26

NC STATE UNIVERSITY


NC STATE UNIVERSITY

Coastal & Computational Hydraulics Team

What We Do | Join Our Team | FigureGen | Kalpana | SWAN+ADCIRC

Downscaling ADCIRC Flooding Inundation Extents Using Kalpana

The ADCIRC modeling system is used often to predict coastal flooding due to tropical cyclones and other storms. The model uses high resolution to represent the coastal environment, including flow pathways (inlets, man-made channels, rivers) and hydraulic controls (barrier islands, raised features). However, due to the use of large domains to represent hazards on coastlines in an entire state or multiple states, the highest resolution is typically about 20 to 50 m in coastal regions. Thus, there is a potential gap between the flooding predictions and the true flooding extents. We have developed a geospatial software to downscale the flooding extents to higher resolution.

MEET OUR TEAM

Faculty
- Casey Dietrich | Posts | CV

Post-Doctoral Researchers
- Dylan Anderson | Posts | Web

Graduate Students
- Ajimon Thomas | Posts | CV
- Alireza Gharagozlou | Posts | CV
- Johnathan Woodruff | Posts | CV
- Autumn Poisson | Posts | CV
- Carter Rucker | Posts | CV
- Brooke Rumbaugh | Posts | CV

Researchers
- Sheppard Medlin

Undergraduate Students
- Chloe Stokes
- Carter Howe
- Jessica Gorski

Alumni

ARCHIVES

About Us | 50
- Awards | 5
- Graduate Students | 6
- In the News | 26

Thank You.
Questions?