Computational Enhancements for the Virginia Department of Transportation Regional River Severe Storm (R^2S^2) Model

http://www.virginiadot.org/vtrc/main/online_reports/pdf/17-r18.pdf

MOHAMED M. MORSY
Graduate Research Assistant
Department of Civil and Environmental Engineering
University of Virginia

GINA L. O'NEIL
Graduate Research Assistant
Department of Civil and Environmental Engineering
University of Virginia

JONATHAN L. GOODALL, Ph.D., P.E.
Associate Professor
Department of Civil and Environmental Engineering
University of Virginia

GAMAL HASSAN, P.E.
President and Senior River Mechanics Engineer
Hassan Water Resources, PLC

Final Report VTRC 17-R18

VIRGINIA TRANSPORTATION RESEARCH COUNCIL
530 Edgemont Road, Charlottesville, VA 22903-2454
vtrc.virginiadot.org

Standard Title Page - Report on Federally Funded Project

1. Report No.: FHWA/VTRC 17-R18	2. Government Accession No.:	3. Recipient's Catalog No.:
4. Title and Subtitle: Computational Enhancements for the Virginia Department of Transportation Regional River Severe Storm (**R^2S^2**) Model		5. Report Date: May 2017 6. Performing Organization Code:
7. Author(s): Mohamed M. Morsy, Gina L. O'Neil, Jonathan L. Goodall, Ph.D., P.E., and Gamal Hassan, P.E.		8. Performing Organization Report No.: VTRC 17-R18
9. Performing Organization and Address: Virginia Transportation Research Council 530 Edgemont Road Charlottesville, VA 22903		10. Work Unit No. (TRAIS): 11. Contract or Grant No.: 107898
12. Sponsoring Agencies' Name and Address: Virginia Department of Transportation 1401 E. Broad Street Richmond, VA 23219	Federal Highway Administration 400 North 8th Street, Room 750 Richmond, VA 23219-4825	13. Type of Report and Period Covered: Final Contract 14. Sponsoring Agency Code:
15. Supplementary Notes:		

16. Abstract:

Climate change introduces infrastructure flooding challenges, especially for coastal regions with low topographic relief. More frequently occurring intense storms and sea level rise are two projected impacts of climate change that will lead to increased flooding risks. These changing conditions make having the ability to forecast accurately potential flooding impacts to transportation infrastructure critical. The Virginia Department of Transportation (VDOT) Hampton Roads District worked with Hassan Water Resources, PLC, a consulting firm, to create a flood forecasting model called the Regional River Severe Storm (R^2S^2) model for, among other purposes, flood warning applications. The model was built for watersheds within the district that cover approximately 2,230 square miles and include 493 bridges and culverts.

This report describes work by researchers at the University of Virginia to complete computational enhancements to the R^2S^2 model so that it might ultimately be implemented by VDOT for flood forecasting applications. Specific project tasks were to (1) design, implement, and test software for automating rainfall forecast inputs from the National Weather Service; (2) speed up the model execution using a graphics processing unit (GPU); and (3) automate the visualization of model output through an online, map-based system and automate emails of flood impacted locations to decision makers within VDOT.

Task 1 resulted in software for automating the access, download, and transform of rainfall forecast data produced by the National Weather Service High-Resolution Rapid Refresh (HRRR) model into the inputs required by the R^2S^2 model. The heart of the R^2S^2 model, and the most time-consuming part of the model, is a high-resolution hydrodynamic model called the Two-dimensional Unsteady Flow (TUFLOW) model. Task 2 resulted in speeding up the TUFLOW model by 50x from over 100 hr when the model is run using a central processing unit to just over 2 hr when using a GPU. Finally, Task 3 resulted in software for automating tasks including extracting maximum water depth from the model output for a set of bridges, creating a Google Maps–based website showing impacted bridges, and having the system notify decision makers via email of bridges at risk of being overtopped.

The study recommends that VDOT's Hampton Roads District Hydraulics Engineer undertake the following: (1) with the help of research staff at the Virginia Transportation Research Council, coordinate with the University of Virginia research team to begin using the R^2S^2 model to determine the accuracy of the output derived using the new GPU-based TUFLOW solver and begin the calibration process using real rainfall data and streamflow information; and (2) based on the information gathered, determine what additional R^2S^2 model modifications are required to make the model a fully automated flood forecasting system. These modifications could include automating the data exchange between HEC-HMS and TUFLOW and making further enhancements to the workflow for visualization and notification of model results based on feedback from potential VDOT users.

17 Key Words: flood model, flooding risks, GPU processing, TUFLOW		18. Distribution Statement: No restrictions. This document is available to the public through NTIS, Springfield, VA 22161.	
19. Security Classif. (of this report): Unclassified	20. Security Classif. (of this page): Unclassified	21. No. of Pages: 22	22. Price:

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

FINAL REPORT

COMPUTATIONAL ENHANCEMENTS FOR THE VIRGINIA DEPARTMENT OF TRANSPORTATION REGIONAL RIVER SEVERE STORM (R^2S^2) MODEL

Mohamed M. Morsy
Graduate Research Assistant
Department of Civil and Environmental Engineering
University of Virginia

Gina L. O'Neil
Graduate Research Assistant
Department of Civil and Environmental Engineering
University of Virginia

Jonathan L. Goodall, Ph.D., P.E.
Associate Professor
Department of Civil and Environmental Engineering
University of Virginia

Gamal Hassan, P.E.
President and Senior River Mechanics Engineer
Hassan Water Resources, PLC

VTRC Project Manager
G. Michael Fitch, Ph.D., Virginia Transportation Research Council

In Cooperation with the U.S. Department of Transportation
Federal Highway Administration

Virginia Transportation Research Council
(A partnership of the Virginia Department of Transportation
and the University of Virginia since 1948)

Charlottesville, Virginia

May 2017
VTRC 17-R18

DISCLAIMER

The project that is the subject of this report was done under contract for the Virginia Department of Transportation, Virginia Transportation Research Council. The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the Virginia Department of Transportation, the Commonwealth Transportation Board, or the Federal Highway Administration. This report does not constitute a standard, specification, or regulation. Any inclusion of manufacturer names, trade names, or trademarks is for identification purposes only and is not to be considered an endorsement.

Each contract report is peer reviewed and accepted for publication by staff of the Virginia Transportation Research Council with expertise in related technical areas. Final editing and proofreading of the report are performed by the contractor.

ABSTRACT

Climate change introduces infrastructure flooding challenges, especially for coastal regions with low topographic relief. More frequently occurring intense storms and sea level rise are two projected impacts of climate change that will lead to increased flooding risks. These changing conditions make having the ability to forecast accurately potential flooding impacts to transportation infrastructure critical. The Virginia Department of Transportation (VDOT) Hampton Roads District worked with Hassan Water Resources, PLC, a consulting firm, to create a flood forecasting model called the Regional River Severe Storm (R^2S^2) model for, among other purposes, flood warning applications. The model was built for watersheds within the district that cover approximately 2,230 square miles and include 493 bridges and culverts.

This report describes work by researchers at the University of Virginia to complete computational enhancements to the R^2S^2 model so that it might ultimately be implemented by VDOT for flood forecasting applications. Specific project tasks were to (1) design, implement, and test software for automating rainfall forecast inputs from the National Weather Service; (2) speed up the model execution using a graphics processing unit (GPU); and (3) automate the visualization of model output through an online, map-based system and automate emails of flood impacted locations to decision makers within VDOT.

Task 1 resulted in software for automating the access, download, and transform of rainfall forecast data produced by the National Weather Service High-Resolution Rapid Refresh (HRRR) model into the inputs required by the R^2S^2 model. The heart of the R^2S^2 model, and the most time-consuming part of the model, is a high-resolution hydrodynamic model called the Two-dimensional Unsteady Flow (TUFLOW) model. Task 2 resulted in speeding up the TUFLOW model by 50x from over 100 hr when the model is run using a central processing unit to just over 2 hr when using a GPU. Finally, Task 3 resulted in software for automating tasks including extracting maximum water depth from the model output for a set of bridges, creating a Google Maps–based website showing impacted bridges, and having the system notify decision makers via email of bridges at risk of being overtopped.

The study recommends that VDOT's Hampton Roads District Hydraulics Engineer undertake the following: (1) with the help of research staff at the Virginia Transportation Research Council, coordinate with the University of Virginia research team to begin using the R^2S^2 model to determine the accuracy of the output derived using the new GPU-based TUFLOW solver and begin the calibration process using real rainfall data and streamflow information; and (2) based on the information gathered, determine what additional R^2S^2 model modifications are required to make the model a fully automated flood forecasting system. These modifications could include automating the data exchange between HEC-HMS and TUFLOW and making further enhancements to the workflow for visualization and notification of model results based on feedback from potential VDOT users.

FINAL REPORT

COMPUTATIONAL ENHANCEMENTS FOR THE VIRGINIA DEPARTMENT OF TRANSPORTATION REGIONAL RIVER SEVERE STORM (R^2S^2) MODEL

Mohamed M. Morsy
Graduate Research Assistant
Department of Civil and Environmental Engineering
University of Virginia

Gina L. O'Neil
Graduate Research Assistant
Department of Civil and Environmental Engineering
University of Virginia

Jonathan L. Goodall, Ph.D., P.E.
Associate Professor
Department of Civil and Environmental Engineering
University of Virginia

Gamal Hassan, P.E.
President and Senior River Mechanics Engineer
Hassan Water Resources, PLC

INTRODUCTION

Climate change introduces significant challenges for departments of transportation (Committee on Climate Change and U.S. Transportation, 2008). In Virginia, the Virginia Department of Transportation (VDOT) Hampton Roads District is already facing these challenges due to sea level rise, storm surge, and more extreme weather events. Increased flooding is one of the impacts to transportation infrastructure caused by these changing climatic conditions. Severe weather results in flooding in coastal regions from both storm surge and precipitation runoff. Sea level rise exacerbates the flooding problem by increasing the impact of storm surge and increasing tailwater elevations for streams and stormwater infrastructure designed to alleviate flooding from increased runoff.

Having the ability to forecast potential impacts to transportation infrastructure accurately and quickly due to forecast weather events will become more critical as climate change continues. VDOT's Hampton Roads District has already started to address this issue by creating a flood warning system called the Regional River Severe Storm Model (R^2S^2). The purpose of R^2S^2 is to serve as a planning tool to assist residency administrators in efficiently allocating scare resources to close roads and to assist first responders with accessing flood prone areas. Future enhancements may provide flooding predictions directly to the public. The system makes use of forecast rainfall conditions from the National Weather Service. It then uses the rainfall forecasts to drive a hydrology and hydraulics model that identifies specific locations where roads or bridges may flood in the 2,230 square mile watershed.

The current implementation of R^2S^2 is computationally demanding and data-intensive. Some of the data translation steps in the current implementation, in particular the translation of forecast rainfall datasets to model input files, are currently done manually. Flood warning is time sensitive, and so there is a desire to reduce the time from when forecasts are available to on-the-ground projections of road closures. This requires automation of data processing workflows to access, transform, and load rainfall forecasts into the model. Also, the model itself is computationally demanding and there may be opportunities for speeding up the model execution time through adoption of high performance computing techniques and infrastructure.

PURPOSE AND SCOPE

The purpose of this study was to speed up the R^2S^2 execution so that it might ultimately be implemented by VDOT as a tool for flood forecasting applications. This was accomplished as three primary tasks: (1) automating the workflow for accessing, transforming, and loading rainfall forecasts from federal data providers into R^2S^2; (2) investigating and providing solutions for speeding up the hydrology and hydraulic models behind R^2S^2 through parallel computing using graphic processing units (GPUs); and (3) demonstrating and providing methods for automatically sending road closure forecasts to decision makers.

METHODS

Study Area

The study area is in the portion of the Chowan River basin that is within VDOT's Hampton Roads District, which is about 2,230 square miles (Figure 1) and includes the Blackwater River, the Nottoway River, and the Meherrin River. The study area includes 493 georeferenced VDOT bridges and culverts. Due to a high portion of the study area consisting of low-relief terrain, R^2S^2 uses a two-dimensional (2D) hydrodynamic commercial model called Two-dimensional Unsteady Flow (TUFLOW). The area upstream of the project domain is modeled by using Hydrologic Engineering Center–Hydrologic Modeling System (HEC-HMS), a lumped hydrology model that is less computationally intensive, to generate the inflow boundary conditions for the study area. Including these upstream watersheds, the project domain grows to about 4,240 square miles.

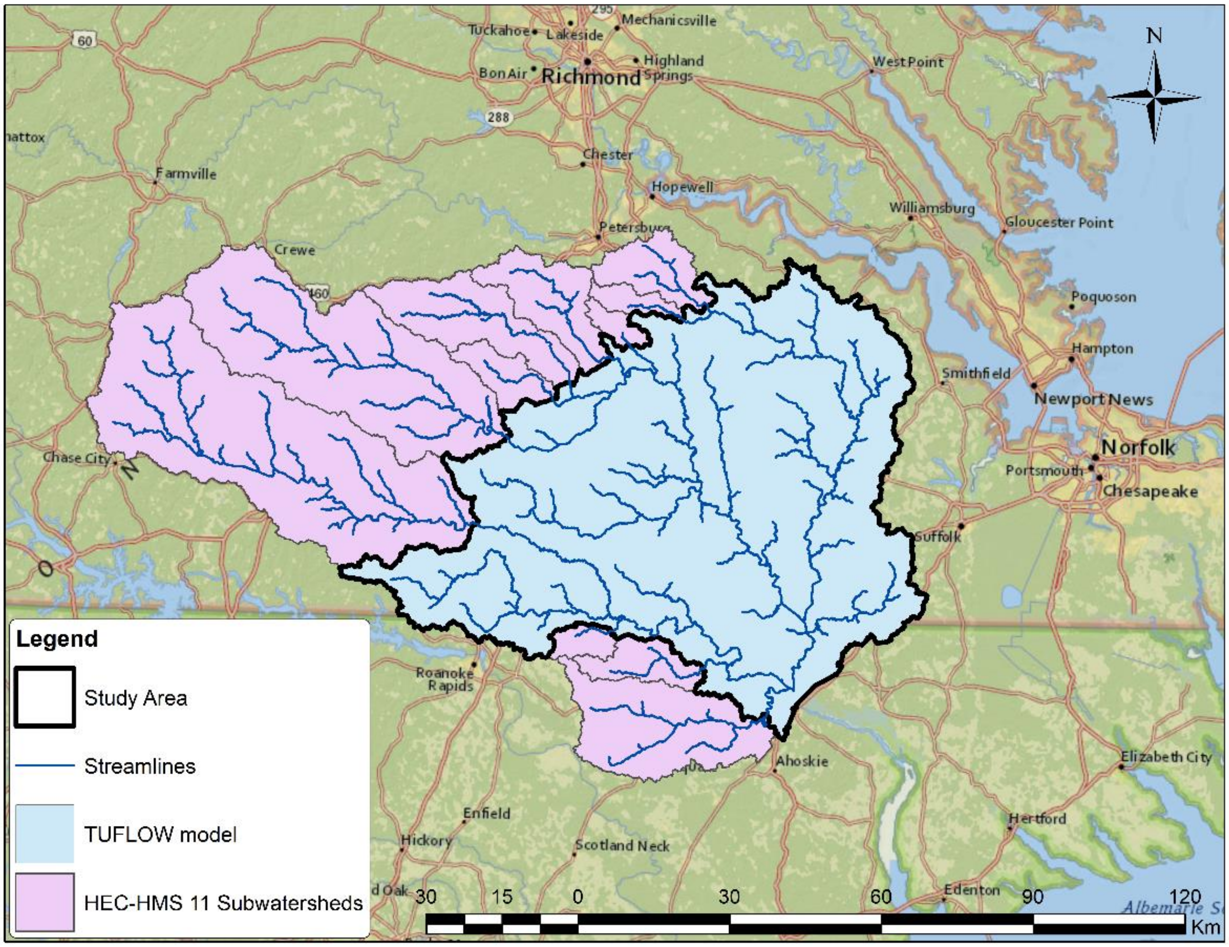

Figure 1. R^2S^2 Domain Composed of Study Area and 11 Subwatersheds That Contribute the Inflow to the Study Area Boundary

Task 1: Forecast Data Automation and Preparation

R^2S^2 uses real-time and forecast products for rainfall and streamflow conditions. Prior to this study, these data were manually downloaded, reformatted, and then manually loaded into the modeling system. In this study, these procedures to collect and process the rainfall data were automated.

Prior to completing this workflow automation step, the research team reviewed the current R^2S^2 implementation to understand how input files are obtained and reformatted for use by the model. Then the research team created a copy of the model at the University of Virginia (UVA) to use the university computational resources for creating the workflow tools.

TUFLOW and HEC-HMS require input rainfall data in different formats. TUFLOW has three approaches for applying the rainfall directly to the computational cells: (1) polygons covering multiple cells are assigned a rainfall time series (used in the original model provided); (2) gridded rainfall is created as ASCII files for each time step or as one NetCDF file (recently available through the updated TUFLOW model); and (3) a rainfall control file that allows the user to specify point time series over the model and specify how the rainfall is interpolated to the model cells. HEC-HMS uses a rainfall time series for each basin stored in a DSS file as an input for the model.

There are several available forecast datasets available from the federal government. These datasets include (1) the High-Resolution Rapid Refresh (HRRR), a higher-resolution nest inside the hourly updated Rapid Refresh (RAP) provided by the National Oceanic and Atmospheric Administration (NOAA) and the National Center for Environmental Prediction (NCEP), (2) the North American Mesoscale Forecast System (NAM), also provided by NCEP; and (3) the National Digital Forecast Database (NDFD), provided by the National Weather Service. These forecast datasets were compared in terms of their spatial resolution, temporal resolution, and model cycle in order to determine which of these datasets would be the best for use within R^2S^2. Once this determination was made, then software was built to automate the workflow of downloading and reformatting the forecast rainfall data to meet the requirements of the different parts of R^2S^2.

Task 2: Speeding Up R^2S^2 Execution

R^2S^2 consists of processing many input files for the TUFLOW model, running HMS to establish boundary conditions for TUFLOW, and processing output files from TUFLOW to determine inundated bridges and culverts (Figure 2). Within this overall workflow, TUFLOW is the bottleneck in terms of workflow execution time. The original TUFLOW model in R^2S^2 used a central processing unit (CPU) for computation and took more than 3 days to execute. This is due to the data and computational demands of using a 2D hydrodynamic model at the scale required for predicting flooding of bridge and culvert infrastructure. Thus, the objective of this task was to focus on speeding up the TUFLOW model execution. The use of multiple CPUs and GPUs has been investigated as a means of speeding up the model execution time for 2D hydrodynamic models (Kalyanapu et al., 2012; Brodtkorb et al., 2012; Rostrup and Sterck, 2010; Castro et al., 2011; Lacasta et al., 2013; Sanders et al., 2010; Garcia et al., 2015). Using CPU clusters is expensive and requires continuous maintenance (Vacondio et al., 2014). Using GPUs offers the performance of smaller clusters at a much lower cost (Jacobsen et al., 2010). Therefore, using GPUs was investigated for speeding up the TUFLOW model.

TUFLOW comes with a GPU Module for speeding up model execution. The research team explored the use of both UVA's computational resources and commercial clouds such as Amazon Web Services (AWS) for providing GPU computational resources. Given the event-based nature of flood warning systems, which require no computational resources between extreme events (other than for testing and development), the cloud computing paradigm of renting computational and storage resources by the hour may be an attractive option from a cost perspective. For this reason, testing and development using cloud infrastructures was a top priority. To implement this step, the TUFLOW model was updated to run with the latest version of the TUFLOW GPU Module.

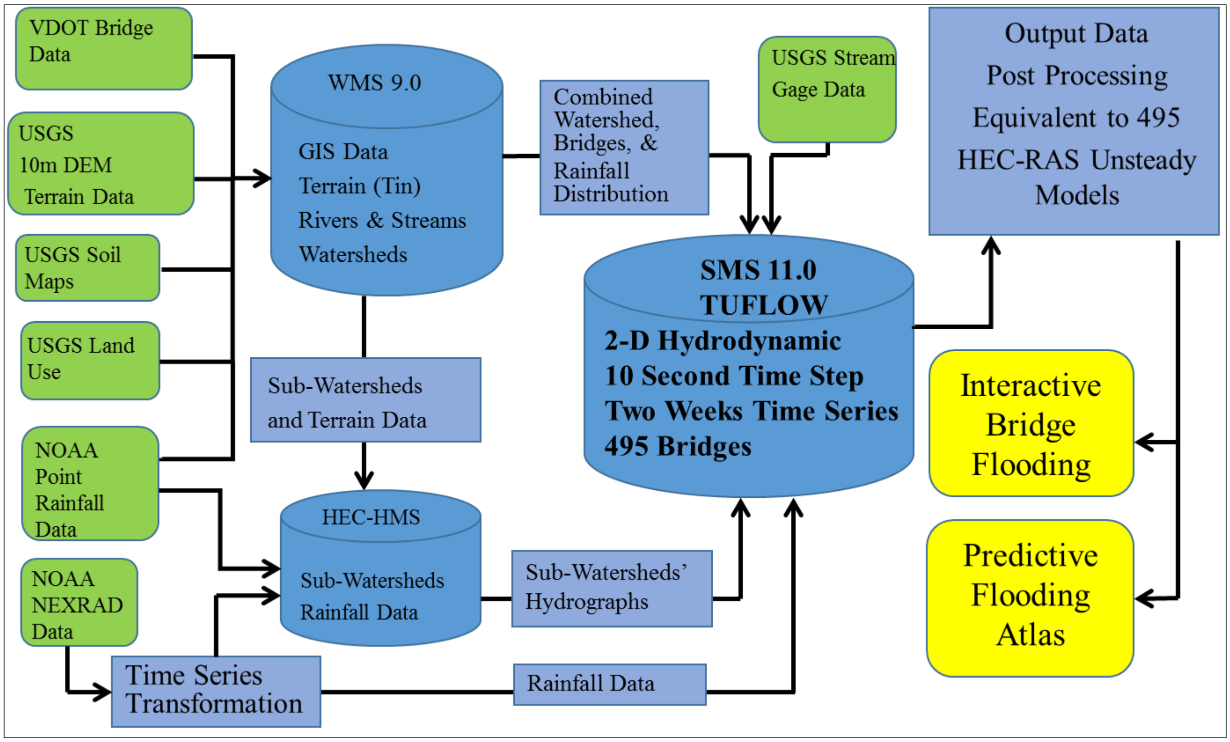

Figure 2. R^2S^2 Workflow

TUFLOW's GPU Module is a powerful solver that is built into the TUFLOW software. The GPU Module uses an explicit solver which is different than the CPU TUFLOW solver used in the original R^2S^2 model, TUFLOW Classic, which uses an implicit solver. A GPU has substantial parallel computing ability that can be used to run large 2D hydrodynamic models from 20 to 100 times faster than using a CPU (Huxley and Syme, 2016; Garcia et al., 2015). Using the power of the modern GPUs, very large models with fine grid resolution can be run within a more useful timeframe for flood warning scenarios.

The UVA Hydroinformatics Lab has a workstation with a modest GPU, shown as M1 in Table 1. The Viz Lab, a facility for UVA students, staff, and faculty to explore and investigate the power of visualization in research and education, has four powerful workstations that include GPUs. The Escher, M2 in Table 1, is a high-end graphics workstation with 64 GB RAM and two NVidia GeForce Titan Graphics cards and is the most powerful workstation in the facility. The research team used M1 and M2 to investigate the effect of using a powerful GPU, M2, versus a modest one, M1 to speed up the TUFLOW model (Table 1).

Table 1. Local Computers With GPUs Used to Investigate TUFLOW Model Execution Times

ID	Type	CPU	RAM (GB)	GPU	GPU RAM
M1	Desktop Dell OptiPlex 990	3.40 GHz, 4 Core(s)	16	NVIDIA Quadro K2000	2.00 GB, 384 SMX CUDA parallel processing cores
M2	Desktop Viz Lab ESCHER	3.20GHz, 3201 Mhz, 6 Core(s)	64	Two units of NVIDIA GeForce GTX TITAN	6.00 GB, 2688 CUDA parallel processing cores for each

GPU = graphics processing unit; CPU = central processing unit.

The Amazon Elastic Compute Cloud (Amazon EC2) is part of AWS, a leading commercial cloud provider, and is designed to make web-scale cloud computing easier for developers. There are several types of the EC2 instance that are each designed for specific purposes. For GPU-based computations, AWS offers G2 instance types. G2 instances provide powerful machines ideal for many applications including computational fluid dynamics (Table 2). Therefore, the G2 instances were used in this analysis.

Table 2. Comparison Between G2 EC2 Instances Performance and Cost

EC2 Instance	Model	GPUs	vCPU	Memory (GiB)	GPU Memory	Storage (GB)	Hourly Fee
G2	g2.2xlarge	1	8	15	4 (GB)	SSD 1 x 60	$0.767
	g2.8xlarge	4	32	60	16 (GB)	SSD 2 x 120	$2.878

GPU = graphics processing unit; vCPU = virtual central processing unit

Task 3: Post-processing and Automating Model Output Dissemination

The TUFLOW model computes the maximum water level at each computational cell within the study area throughout the simulation duration. By use of these maximum water levels and the VDOT bridge locations, a post-processing workflow was created to automate sending an email with the flooded bridges location and generate a visualization for the flooded bridge locations. Web resources such as Google Maps and Geosheets were used to provide a real-time visualization for the flooded bridges in the Hampton Roads District. Google Maps has the capability to generate a simple visualization of uploaded KMZ files, which is a quick and simple method to visualize the flooded bridge locations. Geosheets, an add-on to Google Sheets, has more advanced visualization capabilities than using Google Maps directly without modification. Geosheets can visualize location data with specified attributes provided in a Google sheet. Using the capabilities of the Google API and Geosheet in the post-processing workflow, advanced real-time visualization of the flooded bridge locations can be generated.

RESULTS AND DISCUSSION

Task 1: Forecast Data Automation and Preparation

After comparing the spatial resolution, temporal resolution, and model cycle of each dataset (Table 3), it was concluded that HRRR was the best choice to implement in R^2S^2. HRRR is a weather prediction system composed of a numerical forecast model and an analysis/assimilation system to initialize the model. HRRR is a higher-resolution model nested inside the hourly updated RAP data. Although RAP can provide upper-level analyses and short-range forecasts, HRRR is best used to examine surface and near-surface parameters, such as surface precipitation. The HRRR model is run every hour of the day and forecasts out to 18 hr on a 1-hr time-step for each cycle. It provides a surface total precipitation product in units of millimeters of precipitation depth at a horizontal resolution of 3 km (NOAA, 2012). Surface total precipitation can be accessed as gridded data with dimensions of longitude, latitude, and time. Longitude and latitude are provided in the World Geodetic System (WGS) 1984 coordinate system, and time is in units of decimal days such as 1-1-1 00:00:0.0 (NOAA, 2017a). HRRR data are distributed as a part of the NOAA Operational Model Archive and Distribution

System (NOMADS) project. NOMADS is a network of data servers that uses Open Source Project for a Network Data Access Protocol (OPeNDAP) as the framework used to distribute real-time HRRR data (NOAA, 2017a).

An automated workflow was created to retrieve the real-time and predicted forecast rainfall data from the HRRR database and prepare them as an input to the hydrologic models in the R^2S^2 (Figure 3). Doing so reduces human translation errors and decreases the time between when new rainfall forecasts are available and when the R^2S^2 model produces forecasts.

Table 3. Comparison of Available Forecast Datasets

Dataset	Data Provider	Relevant Data Product	Spatial Resolution (km)	Temporal Resolution (hours)	Forecast Hours	Model Cycle
HRRR	NCEP	Surface total precipitation	3	1	18	24/day
RAP	NCEP	Surface total precipitation	13	1	18	24/day
NDFD	NWS	Quantitative precipitation forecast	5	6	72	8/day
NAM	NCEP	Surface total precipitation	12	1	36	4/day

Source: NOAA (2017b).

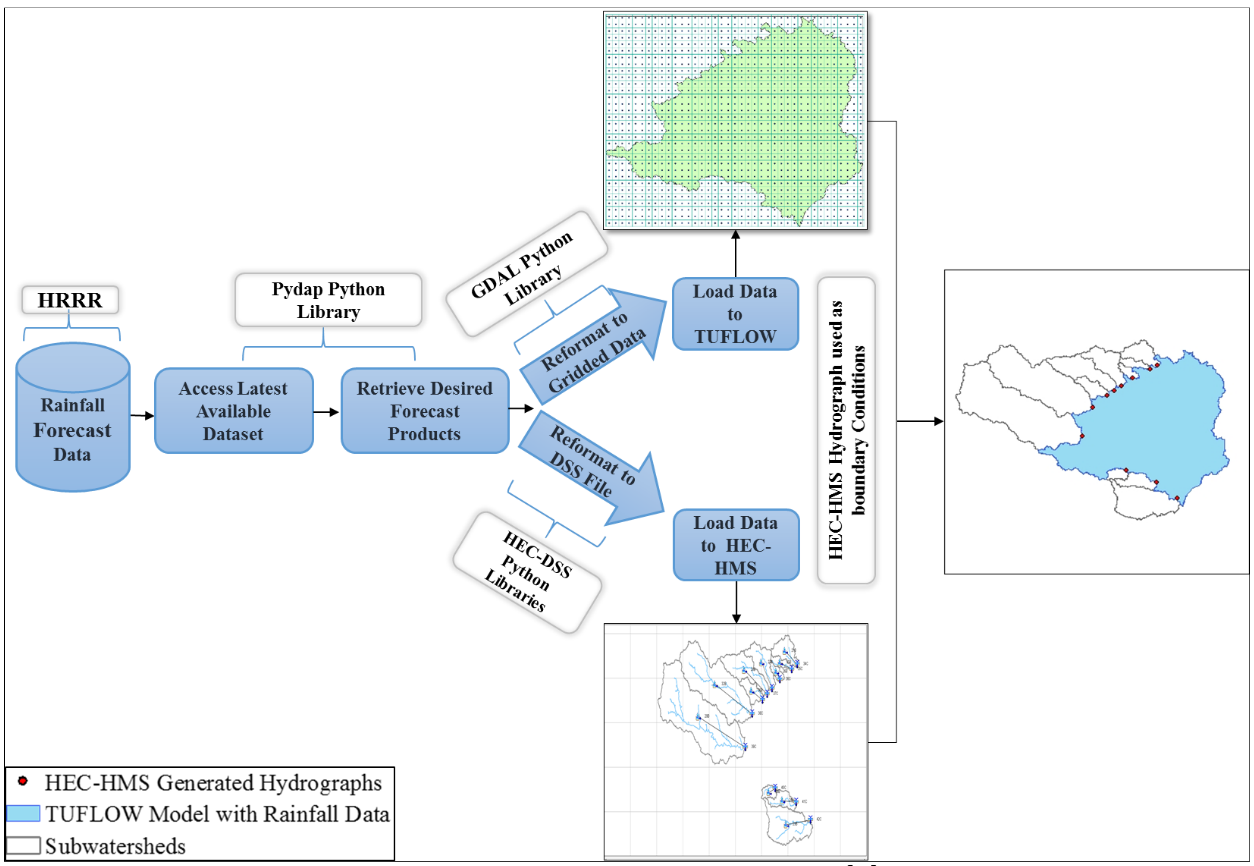

Figure 3. Forecast Data Workflow From Source, HRRR, to R^2S^2 Hydrologic Models

Figure 3 shows the workflow for downloading and reformatting the forecast rainfall data. Pydap, a pure Python library implementing the OPeNDAP, is used to retrieve the desired forecast data for the study area. The automated workflow consists of three main parts: (1) access the latest available forecast data from the HRRR database, (2) retrieve the forecast surface total precipitation with a horizontal resolution of 3 km x 3 km in WGS 1984 coordinate system, and (3) reformat the forecast data for model input in the NAD83 UTM 18N projected coordinate system. These rainfall data are reformatted in two ways: gridded rainfall data for TUFLOW (Figure 4) and subwatershed time series for HEC-HMS.

To include these direct rainfall data in TUFLOW, an event file, TEF, was created to define the storm event properties. For example, using the new TEF, the user can run the model for Super Storm Sandy (Sandy event) using Recorded data (historical rainfall data if available) and Forecast data (forecast rainfall data).

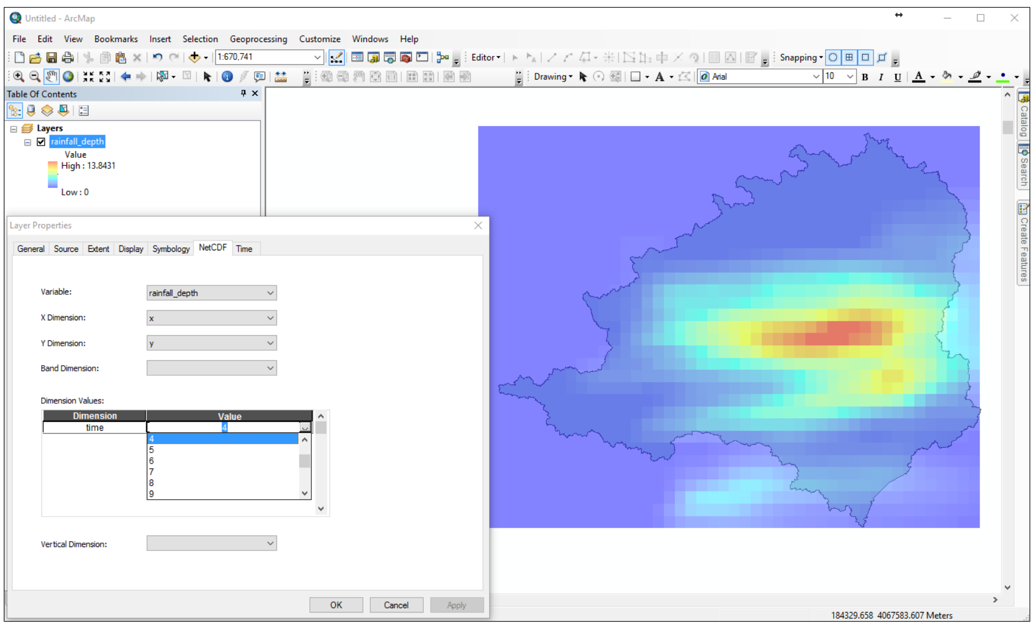

Figure 4. Sample of Forecast Rainfall Data Within a NetCDF File at Time 4 Hours

Task 2: Speeding Up the R^2S^2 Execution

In this task, the original TUFLOW model was enhanced and a new version was created for the speeding up purposes. Both the original model (OM) and updated model run scenarios were based on the Hurricane Sandy storm event that was provided as part of the OM for testing purposes.

Table 4 includes the significant changes between the original TUFLOW model used in R^2S^2 and the updated model created through this study. The OM ran through the Surface-water Model System (SMS) that uses an older TUFLOW release (2013-12-AC). The new model is run outside the SMS and is therefore capable of using the latest TUFLOW release (2016-03-AA), which has the capability of using gridded rainfall data as a direct rainfall to the model. The OM used SMS to prepare input data; the new model uses ArcMap to prepare these data. In the OM, the TUFLOW Control File (TCF) was generated automatically by SMS; now, the TCF is created outside of SMS and was enhanced to include a TUFLOW Event File (TEF) and a TUFLOW Log File (TLF). The TEF was created to include the several event scenarios that the model could run. The TLF was created to organize the running scenarios. The current TUFLOW model consists of a Cartesian grid with a cell size of 50 m by 50 m. Using the TLF, finer Cartesian grids with longer computational duration or coarser Cartesian grids with shorter computational duration and a different time step can be implemented in the future to assess the effect of the model resolution and time step on the model output. The TUFLOW manual states that the time step is typically in the range of one-half to one-fifth of the cell size. Based on this recommendation, the updated model time step has been changed to 15 sec rather than 10 sec as this may have a corresponding reduction in the execution time. By default, the TUFLOW model output is generated in a cell size that is one-half the cell size of the input data. First, the model outputs the data in the original cell size of the input data; second, the model does an internal interpolation to output the data in a smaller cell size. In the new model, the output has the same cell size as the input as this may have a corresponding reduction in the execution time too.

Table 4. Comparison of Model Enhancements

Model Specification	Original Model	Updated Model
Input data preparation	SMS	ArcMap—GIS
TUFLOW Release	2013-12-AC	2016-03-AA
Output Cell Size	25 m	50 m
Time-step	10 sec	15 sec
TUFLOW Control File	Auto-generated from SMS	Created directly outside of SMS
Run Method	Run through SMS	Run directly using batch files

Table 5 summarizes the results of the five TUFLOW model scenarios using M1 and M2 machines (Table 1). The OM scenario represents running the OM obtained from VDOT. The OM took 105.7 hr to execute. Four updated model scenarios were run. The R1 scenario represents executing the updated model with a CPU and by using the same parameters of the OM. This scenario took 120 hr to execute, which is slightly longer than the OM running time. The R2 scenario represents running the new model with two GPUs and keeping the same parameters as the OM. This scenario took 2.2 hr to execute, which is a significant reduction in the model execution time by nearly 50x of the OM execution time (105.7 hr). The R3 scenario represents executing the updated model with a larger time step (15 sec) and generated output cell size (50 m), with two GPUs provided by M2. This scenario also took 2.2 hr to execute, indicating that there is no significant increase in execution time caused by these changes in the time step and output cell size. The R4 scenario represents the same setup as the R3 scenario, but executing the model with the more modest GPU provided by M1. This scenario took 5 times longer to execute (11.5 hr) compared to the R3 scenario. This indicates the quality of the GPU is a significant factor in the model execution time.

Table 5. Comparison of New Versus Original Model

Model Specification	Original Model	New Model			
	OM	R1	R2	R3	R4
Machine	M1	M1	**M2**	**M2**	M1
TUFLOW Release	2013-12-AC	**2016-03-AA**	**2016-03-AA**	**2016-03-AA**	**2016-03-AA**
Precision	Single	Single	Single	Single	Single
Time-step (sec)	10	10	10	**15**	**15**
Output Cell Size (m)	25	25	25	**50**	**50**
Processing Units	CPU	CPU	**GPU**	**GPU**	**GPU**
No. of GPUs	-	-	**2**	**2**	**1**
Running Time (hr)	105.7	**120**	**2.2**	**2.2**	**11.51**

The difference in each scenario is in bold.
OM = original model; R1 = run scenario 1; R2 = run scenario 2; R3 = run scenario 3; R4 = run scenario 4; CPU = central processing unit; GPU = graphics processing unit.

The following describes the analysis of differences in the model results when using the updated TUFLOW version and when using a GPU instead of a CPU solver. Figure 5 provides the fraction of computational cells that have differences in the maximum water level (Max. WL) of the OM compared to the updated version of TUFLOW (R1). These results show that although some cells have differences in water levels up to 1.25 m (4 ft), 92% of the computational cells differ by less than 0.5 m (1.6 ft).

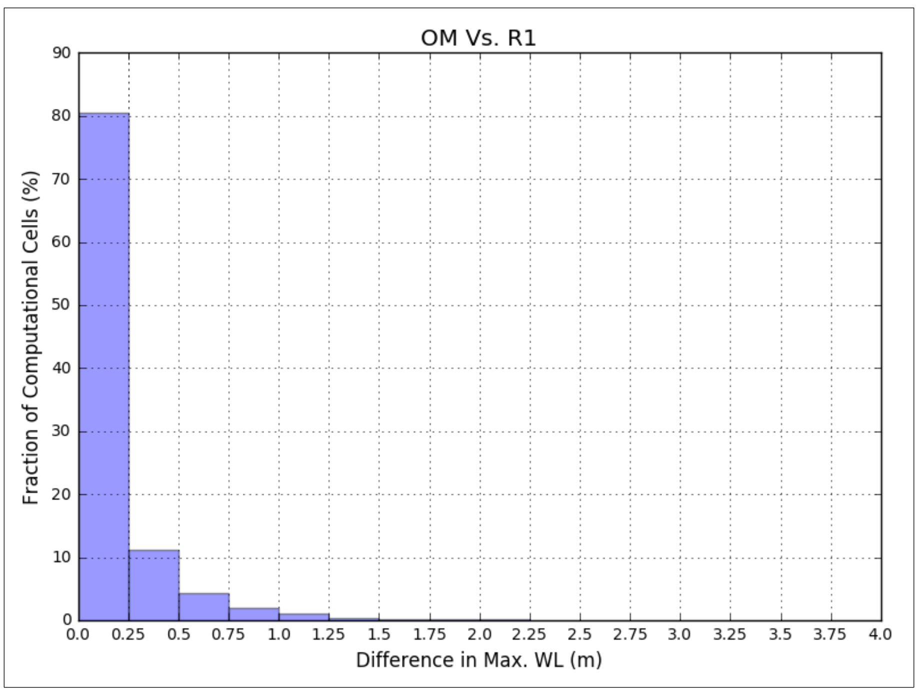

Figure 5. Differences Between Maximum Water Level Generated From OM and R1. OM = original model; R1 = run scenario 1.

Figure 6 provides the differences in Max. WL generated from the OM and the updated model run using GPU solvers (R2). This result shows that differences in the water levels are up to 2.75 m (9 ft), but 80% of the computational cells have differences in the maximum water level less than 0.5 m (1.6 ft). Figure 7 provides the differences in Max. WL generated from the updated model run using the CPU solver (R1) and the updated model run using the GPU solvers

(R2). This result shows that differences in the water levels are up to 2.5 m (8 ft), but again, 92% of the computational cells have differences in the maximum water level less than 0.5 m (1.6 ft). From Figures 6 and 7, it is expected that the two scenarios will produce slightly different results, but explicit solvers using GPUs are less numerically stable compared to implicit solvers using CPUs, so the differences can be large and should be closely checked for consistency.

TUFLOW's GPU Module is able to use multiple GPUs in parallel. A test was conducted to determine how increasing the number of GPUs influenced model execution time. As expected, running the model by using different numbers of GPUs produced the same output results (i.e., no differences in the maximum water levels). Figure 8 provides the results of this test using the updated model and the AWS G2 instances. By using the g2.8xlarge instance with one GPU, the new model takes 4 hr to run. By using the g2.8xlarge instance and increasing the number of GPUs, the optimum execution time is 2.5 hr when three GPUs are used. Using four GPUs on this instance actually increases the execution time compared to using three, which is a known tradeoff caused by data transfers between GPU units.

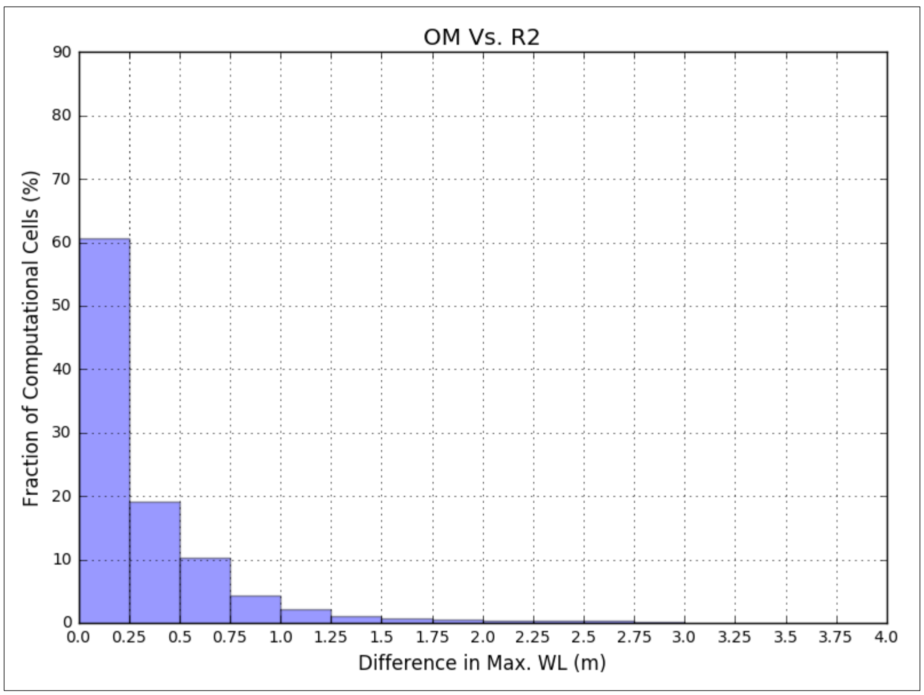

Figure 6. Differences Between Maximum Water Level Generated From OM and R2. OM = original model; R2 = run scenario 2.

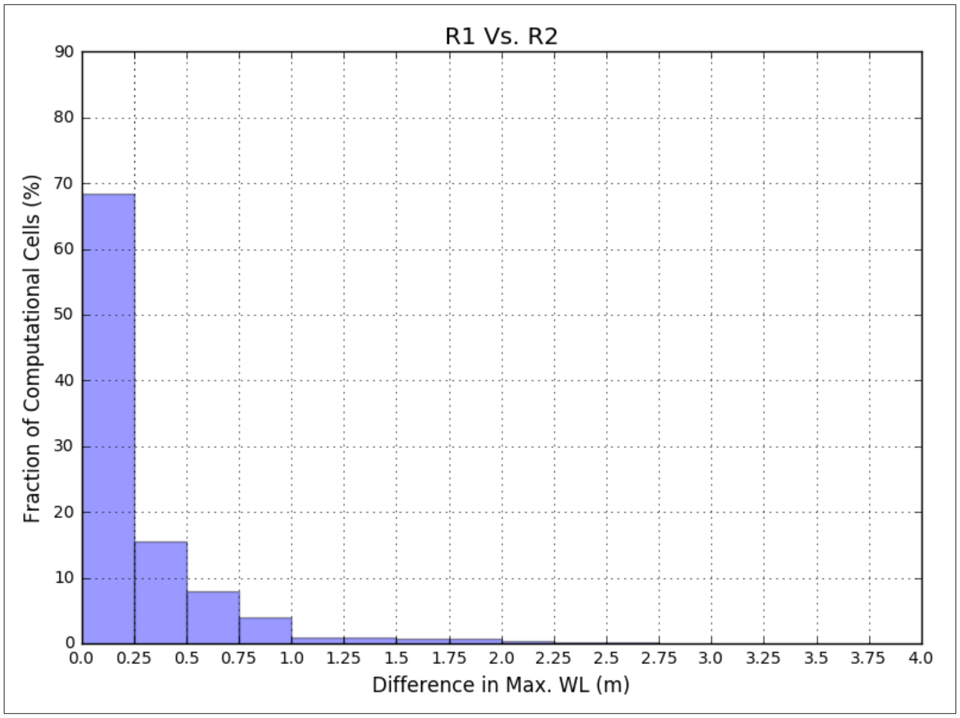

Figure 7. Differences Between Maximum Water Level Generated From R1 and R2. R1 = run scenario 1; R2 =run scenario 2.

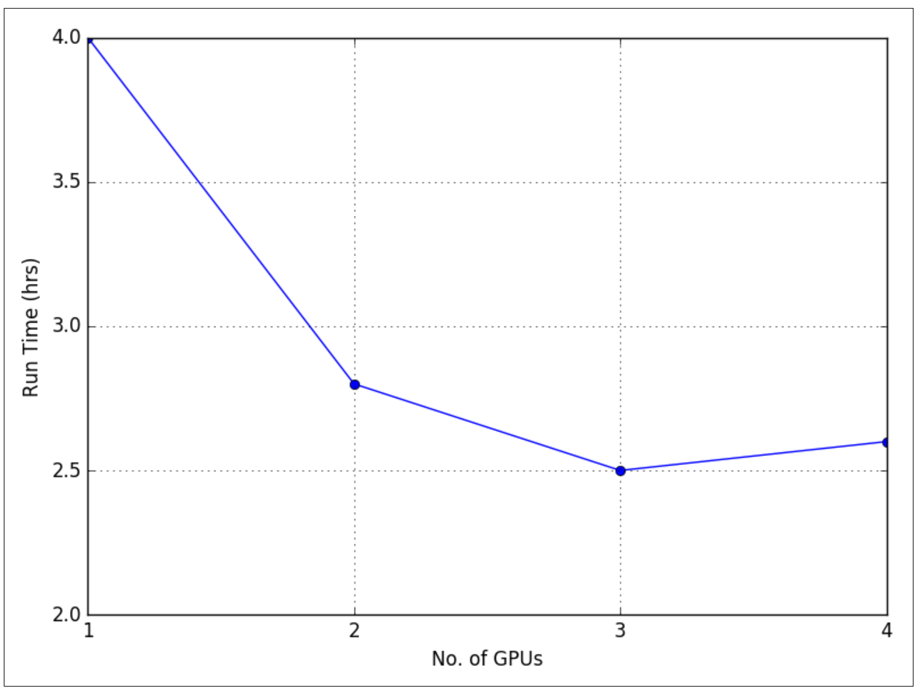

Figure 8. Running TUFLOW Model Through AWS Instances With Different Numbers of GPUs. AWS = Amazon web services; GPUs = graphics processing units.

The optimum execution time on AWS G2 instance, 2.5 hr using three GPUs, is slightly higher than the optimum execution time on M2 machine available at UVA. In addition, the M2 machine used only two GPUs whereas the AWS G2 instance used three. The M2 GPUs are more powerful than the AWS G2 instance GPUs, which accounts for this difference. AWS recently released new P2 instances (Table 6) that have more powerful GPUs, and these machines are expected to provide additional model speedup in AWS. Future work could explore speeding up using the newly released P2 EC2 instances.

Table 6. Comparison Between P2 EC2 Instances Performance and Cost

EC2 Instance	Model	GPUs	vCPU	Memory (GiB)	GPU Memory	Storage (GB)	Hourly Fee
P2	p2.xlarge	1	4	61	12 (GiB)	EBS	$1.084
	p2.8xlarge	8	32	488	96 (GiB)	EBS	$8.672
	p2.16xlarge	16	64	732	192 (GiB)	EBS	$17.344

Task 3: Post-processing and Automating Model Output Dissemination

The created workflow uses different Python libraries such as Geospatial Data Abstraction (GDAL/OGR), Simple KML library (SIMPLEKML), and email library to generate the visualization of the flooded bridge locations and send automatic email with flooded bridges to the decision makers (Figure 9). Figure 9 shows the workflow and its products that could be used with ArcMap, Google Maps, and Google Earth for visualization. This workflow can also generate an online, map-based visualization in real time using Geosheets. There are three products for visualization that can be generated from this workflow: (1) a new shapefile that includes just the flooded bridges; (2) a KMZ file that includes just the flooded bridges that could be used to visualize the results through Google Maps and/or Google Earth besides the ability to sending this file to decision makers through email; and (3) a dynamic and real-time visualization on Geosheets created by automatically uploading the bridges with their flooded status to Google Doc using Google API. Figure 10 shows an example for uploading the generated KMZ file from the workflow to Google Maps and how the flooded bridges show up with corresponding information. Figure 11 shows an example of an advanced visualization for the flooded bridges directly on the Geosheets permanent URL once the workflow runs. This visualization shows the bridges as being not overtopped (green), nearly overtopped (yellow), and overtopped (red) from forecast rain events.

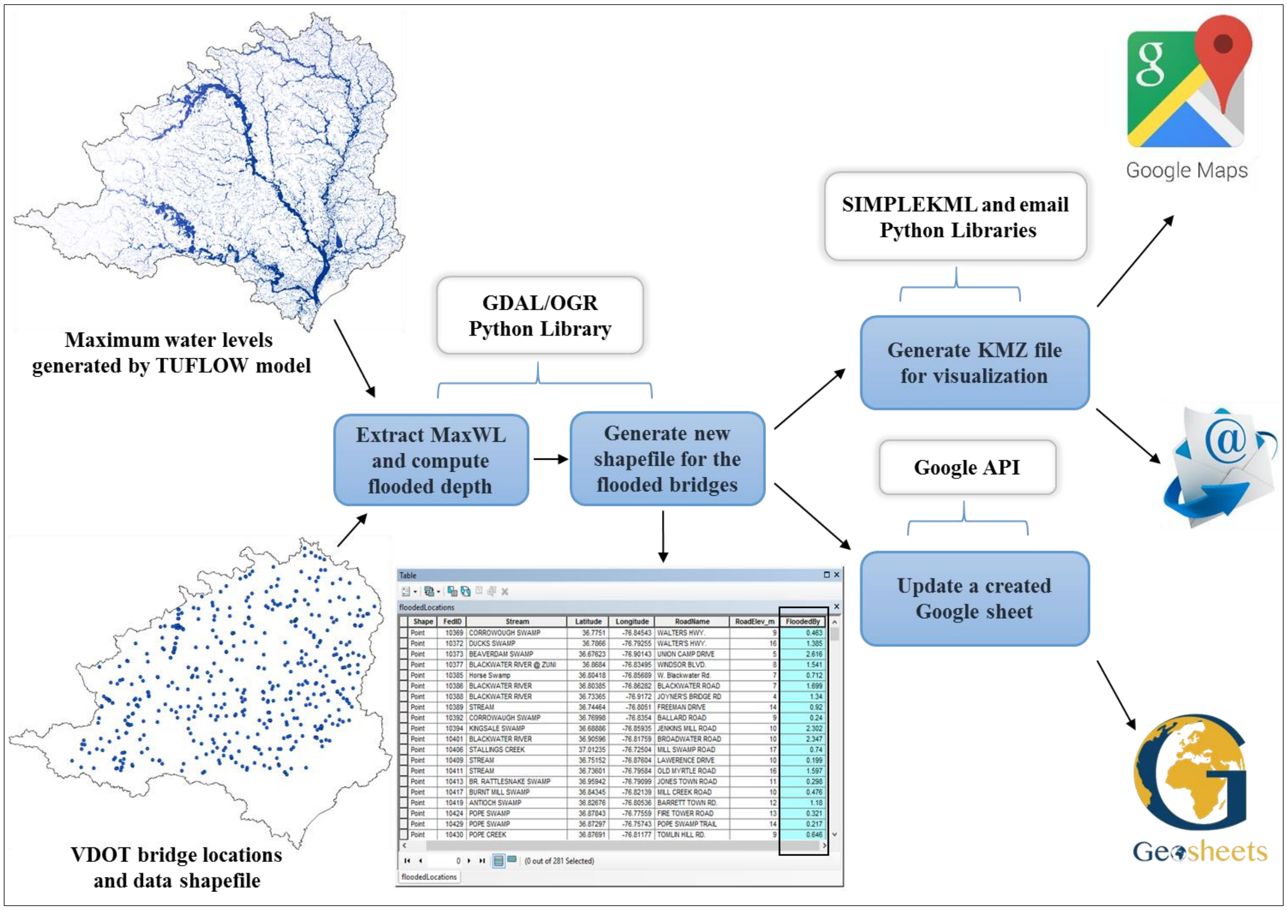

Figure 9. Post-processing Workflow for Producing Different Visualization Resources

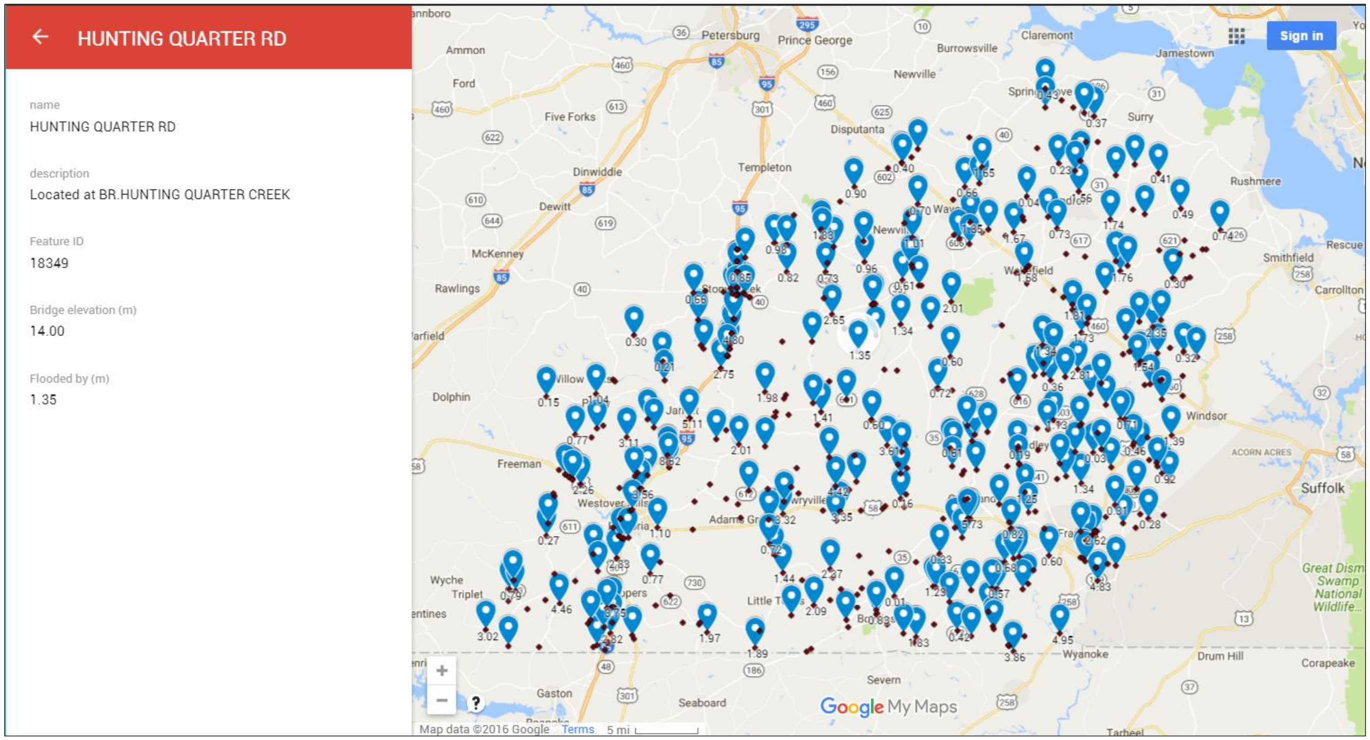

Figure 10. Visualizing the Flooded Bridges Location Using the Generated KMZ File and Google Maps. https://goo.gl/j4aQ7q.

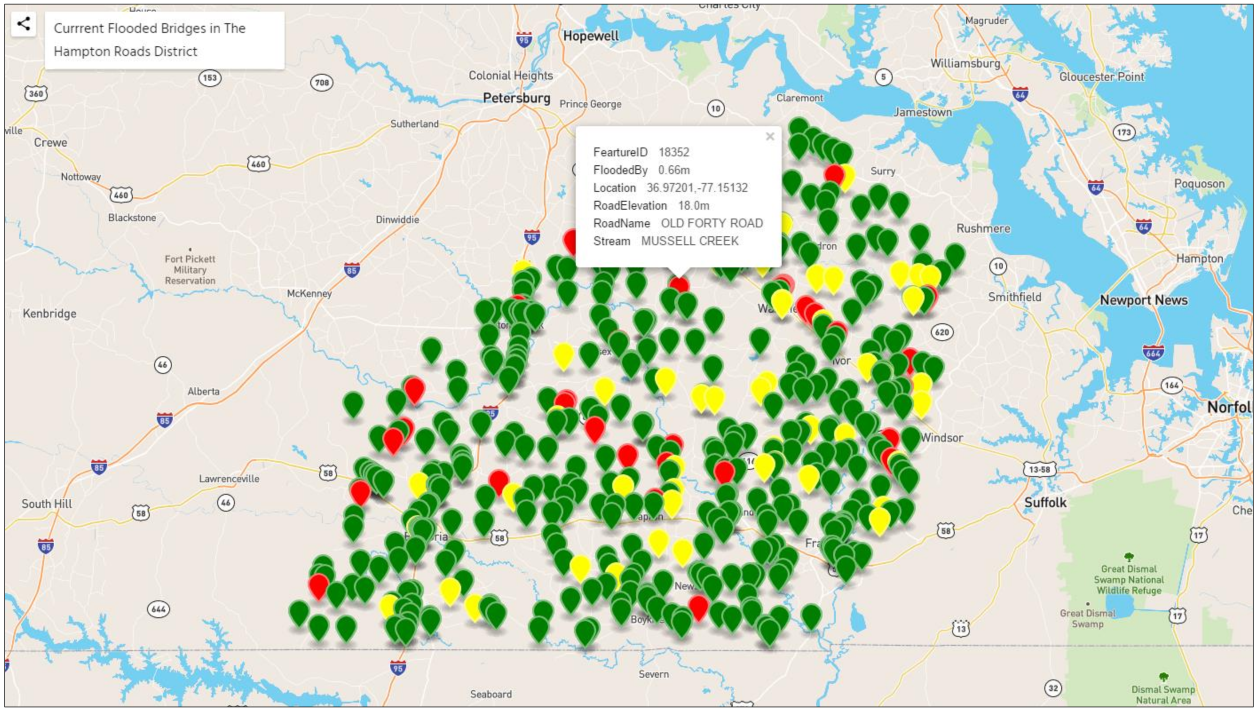

Figure 11. Real-Time Visualization With Permanent URL for Visualizing the Flooded Bridges Location Using Geosheets. https://www.geosheets.com/map/s:Lo6Wq0Jl/Currrent-Flooded-Bridges-in-The-Hampton-Roads-District.

CONCLUSIONS

This study resulted in the following computational enhancements to the R^2S^2 model:

- New software now exists for automating the data gathering and processing steps required to create forecasted rainfall inputs to the model. With the use of this new software, it is now possible to obtain and process automatically the latest rainfall forecast data from federal data providers.

- A new approach has been identified that uses multiple GPUs in parallel to execute the hydrodynamic model at the core of R^2S^2. Using this new approach, a 50x speedup was achieved, reducing the model execution time from over 100 hr to just over 2 hr.

- A new prototype visualization tool was created to allow for real-time dissemination of model results to end-users and decision makers. This online, map-based visualization shows bridge inundation conditions based on the model output and can be generated in a semi-automated fashion when new model results become available.

RECOMMENDATIONS

1. *VDOT's Hampton Roads District Hydraulics Engineer, with the help of research staff at the Virginia Transportation Research Council, should coordinate with the UVA research team to begin using the R^2S^2 model to determine the accuracy of the output derived using the new GPU-based TUFLOW solver and begin the calibration process using real rainfall data and streamflow information.*

2. *Based on information gained when implementing Recommendation 1, VDOT's Hampton Roads District Hydraulics Engineer should determine what additional R^2S^2 model modifications are required to make the model a fully automated flood forecasting system. These modifications could include (1) automating the data exchange between HEC-HMS and TUFLOW, and (2) making further enhancements to the workflow for visualization and notification of model results based on feedback from potential VDOT users.*

BENEFITS AND IMPLEMENTATION

Benefits

With regard to Recommendation 1, the benefit will be a better understanding of the current accuracy of the model and adjustments to the model made through model calibration could improve its accuracy. Knowing the accuracy of the model and improving its accuracy to the extent possible with current data inputs will benefit the agency as they use the tool for decision support.

With regard to Recommendation 2, the benefit will be a clearer understanding of next steps to be taken that will improve the accuracy and utility of the tool. This will aid in directing further resources to targeted priority areas that most improve the tools adoption within the agency for decision support before, during, and following extreme weather events.

Implementation

VDOT's Hampton Roads Deputy District Administrator and VDOT's Hampton Roads District Hydraulics Engineer met with the UVA research team to discuss the plan for evaluating the model output for upcoming precipitation events as suggested in Recommendation 1. Both expressed interest in further refinements to the model to enhance its potential operational value for the district. These specific enhancements will be identified as the model begins being used during the 2017 summer and fall seasons (Atlantic hurricane season) and would move a fully automated flood forecasting system to one that could be used for operational decisions about road and bridge closures in response to expected flood events.

ACKNOWLEDGMENTS

The Regional River Severe Storm (R^2S^2) model was originally built by Hassan Water Resources, PLC, in collaboration with VDOT's Hampton Roads District Hydraulics Engineer Andrew B. Scott. This study greatly benefited from both the data and insights that resulted from this prior work.

REFERENCES

Brodtkorb, A.R., Sætra, M.L., Altinakar, M. 2012. Efficient Shallow Water Simulations on GPUs: Implementation, Visualization, Verification, and Validation. *Computers & Fluids*, Vol. 55, pp. 1-12.

Castro, M.J., Ortega, S., de la Asunción, M., Mantas, J.M., and Gallardo, J.M. 2011. GPU Computing for Shallow Water Flow Simulation Based on Finite Volume Schemes. *Comptes Rendus Mécanique*, Vol. 339, No. 2-3, pp. 165-184.

Committee on Climate Change and U.S. Transportation. 2008. *Potential Impacts of Climate Change on U.S. Transportation.* Transportation Research Board Special Report 290. http://www.trb.org/Main/Public/Blurbs/156825.aspx. Accessed September 30, 2015.

Garcia, R., Restrepo, P., DeWeese, M., Ziemer, M., Palmer, J., Thornburg J., and Lacasta, A. 2015. Advanced GPU Paralellization for Two-Dimensional Operational River Flood Forecasting. In *36th International Association for Hydro-Environment Engineering and Research World Congress.* The Hague, The Netherlands.

Huxley, C., and Syme, B. 2016. TUFLOW GPU—Best Practice Advice for Hydrologic and Hydraulic Model Simulations. In *Proceedings of the 37th Hydrology and Water Resources Symposium (HWRS),* Queenstown, New Zealand.

Jacobsen, D., Thibault, J.C., and Senocak, I. 2010. An MPI-CUDA Implementation for Massively Parallel Incompressible Flow Computations on Multi-GPU Clusters. In *American Institute of Aeronautics and Astronautics (AIAA) 48th Aerospace Science Meeting Proceedings.* Orlando, FL.

Kalyanapu, A.J., Shankar, S., Pardyjak, E.R., Judi, D.R., and Burian, S.J. 2012. Assessment of GPU Computational Enhancement to a 2D Flood Model. *Environmental Modelling & Software,* Vol. 26, No. 8, pp. 1009-1016.

Lacasta, A., García-Navarro, P., Burguete, J., and Murillo, J. 2013. Preprocess Static Subdomain Decomposition in Practical Cases of 2D Unsteady Hydraulic Simulation. *Computers & Fluids,* Vol. 80, pp. 225-232.

National Oceanic and Atmospheric Administration. 2012. *Rapid Refresh*. https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/rapid-refresh-rap. Accessed January 24, 2017.

National Oceanic and Atmospheric Administration. 2017a. *Advanced Data Access Methods*. http://nomads.ncdc.noaa.gov/guide/?name=advanced. Accessed January 24, 2017.

National Oceanic and Atmospheric Administration. 2017b. *Data Products*. https://www.ncdc.noaa.gov/nomads/data-products. Accessed January 24, 2017.

Rostrup, S., and Sterck, H.D. 2010. Parallel Hyperbolic PDE Simulation on Clusters: Cell Versus GPU. *Computer Physics Communications*, Vol. 181, No. 12, p. 2164.

Sanders, B.F., Schubert, J.E., and Detwiler, R.L. 2010. ParBreZo: A Parallel, Unstructured Grid, Godunov-Type, Shallow-Water Code for High-Resolution Flood Inundation Modeling at the Regional Scale. *Advances in Water Resources*,Vol. 33, Issue 12, December, pp. 1456-1467.

Vacondio, R., Dal Palù, A., and Mignosa, P. 2014. GPU-enhanced Finite Volume Shallow Water Solver for Fast Flood Simulations. *Environmental Modelling & Software*, Vol. 57, pp. 60-75.