A Parallel Computation and Web Visualization Framework for Rapid Large-scale Flood Risk Mapping

Min Wang*, Ruixun Lai, Runliang Xia, Ming Wang and Ming Yang

Yellow River Institute of Hydraulic Research, Shunhe Road 45th, Zhengzhou, China, 450003
Email: yrcc_wm@163.com

Abstract. For rapid flood risk mapping, a key aspect is transferring the results of flood simulations for web visualization. The challenges here include: (1) large-scale and complicated modelling; (2) the need for fast computation to solve flood numerical models; and (3) effective web visualization. This paper tackles these challenges by introducing a web-based framework that can transfer the results of parallel flood simulation to a web visualization application. Flood depth data, velocity and flood arrival time are calculated using a parallel shallow water model with a predicted input flow. This is automatically transferred into shapefile data using ArcObjects components. Other data relating to the flood risk mapping is managed and served through a REST (Representational State Transfer) interface. The web visualization is performed using the ArcGIS for JavaScript API. We applied the framework to the floodplain of the lower Yellow River in China. The results show basic geo-information about the domain, the flood depth classified by color and economic loss through inundation of villages and farmland. Although the simulated domain was large and the boundary conditions complicated, the whole process from flood risk simulation to web visualization took less than 3 hours, which is enough time to increase flood preparedness.

1. Introduction

Flood risk maps are crucial tools in helping to increase preparedness for floods and to reduce their detrimental impact [1,2]. With a prediction of extreme precipitation due to climate change, a rapid flood risk mapping is a key component of effective emergency management and risk reduction.

For the rapid flood risk mapping, there are several significant challenges to face. First of all, areas for flood simulation are typically large and the numerical modelling is complicated. River floods in particular usually happen in floodplains that cover a large area. Additionally, the boundary conditions for a floodplain are complex. There can be roads, railways, obstructions and a variety of man-made buildings. At the same time, the boundary conditions affect the propagation of the flood, so they have to be considered in the simulation [1]. To model large-scale floods, there is little choice but to apply parallelization so that hyper-resolution simulations can be realized [3].

A second important challenge is the need for a fast prediction of the extent of possible inundation. Fast simulations need to provide a map several hours before the occurrence of real flooding to service a variety of applications [4,5]. Fast simulations also play an important role in flood risk management by triggering reduction of impact measures and strategies across a range of organizations [6].

There are three common approaches to fast simulation. The first uses precipitation-runoff processes derived from hydrological models to simulate inundation using a hydraulic model [7]. The hydraulic models used for this are typically simplified, e.g. 1D numerical models. This helps to improve efficiency and calculation of water depth and delineation of flood zones can be brought about with GIS support [8–10]. Another approach is to apply a 2D hydraulic or coupled numerical model with the
help of parallel computing [11]. 2D hydraulic models need more computational resources but result in hyper-resolution modelling that can predict water movement across complicated surfaces [12]. The third possible approach is the monitoring of flooded areas using remote sensing. The products here have the advantage of representing actual flooded areas, but real-time imagery is often not available because of the bad weather conditions during an event [13].

A third core challenge confronting rapid flood risk mapping is the effective visualization of the flood simulation. The choice of a suitable visualization is critical for properly conveying which areas are most at risk of being flooded. To be truly effective, flood risk maps need to provide resources for interpreting the flood risk. Traditional flood risk maps are static maps printed on a paper-based medium. They are visually designed to work as 2D colorized maps that convey the uncertainty present in flood simulations [14]. Recently, however, web technology and geographical web services have become an efficient way of accessing, sharing and visualizing flood-related information [15–17].

To date, there has been a lack of progress regarding the transformation of flood simulation results into web visualizations that can meet the demands of rapid flood simulation. This paper aims to build up a framework that can rapidly transfer parallel computation-based inundation results in ways that will facilitate effective web visualization. The flood risk simulation uses predicted rainfall data to simulate the extent and depth of inundation. To make the processing as efficient as possible, the simulation solves two-dimensional shallow water equations with parallel computation. The simulated results, including water depth, velocity and flood arrival time, are transformed into a shapefile-type format by using ArcGIS ArcObjects components. All of the data related to the flood risk is managed by ArcGIS Server software and is served in the form of REST (Representational State Transfer) data. Interactions with the flood risk results can be performed on the web using applications based on the ArcGIS for JavaScript API.

To test its viability, the proposed framework was applied to the lower Yellow River in China, which has a length of 786 km along its river channel. The test imposed the challenge of producing an effective rapid flood risk mapping with the whole process from simulation to web visualization not taking more than 3 hours.

In the next section we shall introduce the structure for the large-scale flood risk mapping, including the parallel simulation, data transformation and web visualization of the flood risk results. Section 3 presents background information about the application area and shows the results of the web visualization. In the final section, we review the approach and conclude that the framework meets the needs of large-scale and rapid flood risk mapping.

2. Method
The proposed framework for large-scale and rapid flood risk mapping is built upon a web GIS structure. It contains four levels: flood prediction and calculation; data transformation to web GIS; a web service layer; and web visualization on the client (Fig. 1).

The flood prediction and calculation layer is a fundamental part of risk mapping. For the calculations, the predicted input flow is used to derive the conditions for two-dimensional shallow water equations that are solved using a parallel computation platform. The simulation results, including inundation zone, depth, velocity, and arrival time are then transformed into shapefile-type data using ArcObjects components.

Other data, apart from the flood simulation results, consists of geographical information data and image data. Geographical data is typically used to supply background geo-information such as cross-sections, villages, floodplains, river training structures, floodplain embankments and river levees.

All of the data used to compose the flood risk map is managed by an ArcGIS Server, which itself serves various GIS servers according to data-type and usage of the data source [18]. The geographical data and flood risk results are key. They are served as a map service for assembling the maps on the web. Similarly, the image data is served by an image server to provide background information to the application.

The web server and web adaptor work together. The web server provides optional security and load-balancing to server sites and serves data from a physical computer out to the web. The web
adaptor integrates the web server and the GIS servers to receive web service requests through a common URL and to send them to the various GIS servers in the system.

**Web visualization**

- Flood depth
- Basic geographical information
- Find by keywords
- Interaction with flood risk data
- Identify feature attributes

**Web service layer**

- Desktop user
- Web service and web adaptor
- Administrator
- GIS server
- Geographic data
- Flood risk results
- Image data

**Data transformation to Web GIS**

- ArcObjects

**Flood calculation**

- Flood risk simulation
- Parallel computation
- Hydrological prediction

---

**Figure 1.** Outline of the framework for rapid flood risk mapping and its web visualization.

To access and process the data on the web, a REST API interface is used to interact with the data services and to provide data-rich web-mapping applications. There are several types of APIs. In this framework, we adopt the REST API for JavaScript.

In addition to the fundamental components of the web GIS, there are two parts that are critical for large-scale and rapid flood risk mapping: (a) the parallel simulation and its data transformation; (b) the web visualization. We will examine these two parts in greater detail below.
2.1. Parallel Simulation of Flood Propagation and its Data Transformation

The flood simulation is accomplished by solving two-dimensional shallow-water equations. These equations conserve balance of momentum and mass as fundamental principles. In practice, the equations governing natural water bodies such as broad and mixed rivers and lakes can be simplified into two-dimensional vertically-averaged equations [19].

The simulation area is spatially discretized as a set of triangular cells to form an unstructured computational mesh. In the mesh, the elevations of each mesh node represent the topography of the simulated domain. On the basis of these meshes, the two-dimensional shallow-water equations are numerically discretized using a finite volume method [20]. In the equations, the convective fluxes are formed according to Roe’s flux-difference splitting scheme [21]. The gradients of the flow variables (e.g., depth and velocity) are computed using a Green-Gauss approach [22]. The temporal discretization of the equations uses an explicit multistage scheme [23].

After discretization, a large-scale sparse matrix is formed, based on the number of nodes. To accelerate the simulation, a Message Passing Interface (MPI) [24] is used to execute a set of processes and to exchange data. The simulation uses graph-partitioning tools from the Metis library for domain decomposition [25]. Using Metis, the model domain can be partitioned into designed sub-domains in which individual processes can then be executed. The calculated data used to solve the matrix is exchanged between the sub-domains (see Fig. 2(a)).

Since the input is an unsteady flow in which the discharge is changing over time, the parallel simulation process needs to calculate all the results for each time interval and each time interval has an output result that stores depth and velocity specifically for that time. For the purposes of simplification, the flood risk map only needs the maximum area of the inundation zone and the maximum depth for each mesh (see Fig. 2(b)). This being so, the results for all time intervals can be compressed into a single time. Taking a mesh as an example, as soon as this mesh is flooded, the mesh will be marked as part of the flood area. The maximum depth for this mesh is the maximum depth value for all the associated time intervals.

To publish the flood results to the GIS server, the flood inundation data needs to be transformed into two shapefile data files (see Fig. 2(c)). One of the shapefile files stores the maximum depth as a polygonal data-type while the other stores the maximum velocity as a point data-type. The actual transformation is accomplished by programing ArcGIS ArcObjects components. The ArcObjects are the building blocks of Microsoft’s COM (Component Object Model)-based technology that is used to program GIS applications on the basis of object-oriented concepts such as classes, properties and methods [26,27].

The two shapefile data files are created sequentially. Once again, taking a mesh as an example, the mesh will be composed of one cell and three nodes. The three-node data is created first, then the cell is created using the existing points which constitute the mesh. The principal ArcObjects components for creating shapefile data are the IWorkspaceFactory and IFeatureWorkspace interfaces. The interface for the IFeatureWorkspace provides an OpenFeatureClass option that can return the interface for an IFeatureClass. First of all, using the IFeatureClass, an interface for an IFeatureBuffer can be obtained by selecting the option CreateFeatureBuffer and an IFeatureCursor can be obtained by selecting Insert. The geometrical shape of the node is acquired by using a Point object to input the x and y coordinates. The value of the node is acquired by using the value attributes in IFeatureBuffer to get the information for the maximum depth and velocity. After this, the node can be stored using InsertFeature in the IFeatureCursor. The other two nodes in the cell are created in the same way.

The process to create a new cell data file is the same as it is for the node, by using the IWorkspaceFactory and the IFeatureWorkspace interfaces. For a cell, the geometry is generated using the three created nodes across the collected IPoint, IPoi ntCollection and IPolygon interfaces. The depth value of the cell is the average depth value of the three nodes.

2.2. Web Visualization of the Computational Results

Almost all of the data related to flood risk can be integrated with map layers. The map layers are generated using a service from the ArcGIS Server. The web application connects the map and the data using the ArcGIS JavaScript API, which is a JavaScript library, to draw the map in the browser and to
make it possible to pan, identify and interact with the map. The ArcGIS JavaScript API is built on the top of the Dojo framework in which the coding pattern employs Asynchronous Modular Definition [28,29].

![Diagram](image)

**Figure 2.** Parallel computation of the flood extent and its data transformation.

The layout of the web visualization is under the control of Dojo [30] and is shown in Fig. 3. In our framework, there are four main functions that can be performed: (1) showing the background geographical information; (2) visualizing the flood inundation area and depth using classified colors; (3) finding geo-informational locations by using keywords; and (4) interacting with the flood risk data.

The background geographical data is managed by map layer APIs to refer to data sources that are geometric, symbolic and even tabular data. The layer APIs support two types of layers: ArcGISDynamicMapServiceLayer and ArcGISTiledMapServiceLayer. In the flood risk mapping system, the geometric data, flood depth and velocity are shown by the ArcGISDynamicMapServiceLayer. The dynamic map means that its content is updated whenever the map is refreshed. Moreover, in a dynamic map, the data can be re-projected to fit on another map layer.
that does not have the same spatial reference. This is very useful when the spatial references for the
dynamic map layer and the image layer are different. The image data is supported by the
ArcGISTiledMapServiceLayer, which allows image data to be tiled. When the tiled map service is
published, the content in the ArcGISTiledMapServiceLayer is already drawn into scaled images. This
helps to improve the efficiency because these pre-rendered tiles can be served quickly with little effort
on the part of the server.

Figure 3. Using the ArcGIS API for JavaScript to visualize the data.

The function for finding objects by keyword is triggered by a user clicking on the find button. This
can help with finding and zooming in on desired objects by using keywords such as the name of a
river training structure, a cross-section, a village, or the name of a floodplain. The click event is
listened for by the feature getElementById from the document class. Findtask executes the task by
using findParams, which defines the search target including layers, fields and search text. Finally,
Findtask zooms in on the location of the found objects.

The function that enables interaction with the flood risk data lets users query the impact of
the flood and assess the property lost in flooded villages. The property lost includes farmland areas and
agricultural output. This function is triggered once the web map has been loaded. It is accomplished by using the FeatureTable and FeatureLayer classes. Users can query the flood risk data and zoom in on a flooded area that has been selected.

Other functions available in the web visualization include identification, layer management and a home button. The ‘identify’ task is triggered when users click anywhere on the map. It is accomplished by both the IdentifyTask and IdentifyParameters features. IdentifyParameters defines the geometry and extent of the clicked objects while IdentifyTask executes the task and calls back an array of identified results. Legend and layer management is achieved by using the LayerList class. The home button lets users immediately pull back to the full extent of the map.

3. Application and Results

3.1. The Study Area and Data Acquisition

The lower reaches of the Yellow River extend from the Xiaolangdi reservoir to the river’s delta, an overall distance of about 786 km. The channel of the Yellow River in its lower course is compound. There are more than 20 large floodplains that change as the channel evolves. The floodplains in the lower river encompass 15 cities and 43 counties, covering an area of 3956 km², with 2052 villages and 250000 hectares of farmland. As the river floodplains form the main expressway during flooding, floods occur here often [31].

![Figure 4. The lower Yellow River from Xiaolangdi to its delta.](image)

Roads and other manmade obstacles have been built in the floodplains, which makes flood propagation more complicated to calculate. About 1.81 million people live in the floodplains around the lower course of the river. To maintain their livelihood, flood embankments have been built in the floodplains to protect farmland and reduce the threat of flooding. As a result, sedimentation happens around the flood embankments, raising the riverbed and shrinking the channel every year. Flooding of the floodplains remains a threat to people’s lives and agricultural production, resulting in an underdeveloped economy and forming an impoverished area along the river.

For these reasons, rapid flood risk mapping for the lower Yellow River is crucial to protect people’s lives and to reduce agricultural loss. Rapid flood risk mapping for this area needs to accommodate 470 river training structures, 487 cross-sections, and 391 flood embankments with a length of 1331 km. As the simulation domain is large and the boundary conditions are complicated, high-resolution modelling with parallel computation is required.
For this paper we take a scenario that has three input flows from the Yellow River and its tributaries, the Qin River and the Yiluo River (see Fig. 4). The topography of the domain was interpolated using measured cross-sections[32]. The data for the three peak flows and their volume is shown in Table 1. The scenario is intended to illustrate what happens when the flow from the main channel is larger than that from the tributary rivers. For this reason, the peak flow and volume from the Qin and Yiluo Rivers are lower than those from the Yellow River.

Table 1. Input flows and their features from the three rivers.

| River      | Input location | Peak flow (m$^3$/s) | Volume (100 million m$^3$) |
|------------|----------------|---------------------|----------------------------|
| Yellow River | Xiaolangdi     | 4949                | 30.34                      |
| Qin River   | Wuzhi          | 1682                | 3.67                       |
| Yiluo River | Heishiguan     | 4133                | 10.69                      |

Figure 5. A quick view of the results for the flood inundation area.
3.2. Results of the Web Visualization

The whole process from the start of the simulation to the web visualization took less than 3 hours, which meets the need for flood management. In the process of simulation, the computational domain was divided into 8 sub-domains to facilitate the parallel simulation.

Figs. 5 and 6 show the results of the scenario. Fig. 5 provides a quick view of the main information regarding the flood inundation area and an introduction to the simulation domain. The introduction gives users some background regarding the lower Yellow River, including the floodplain area, the size of the population, the proportion of farmland, etc. At the left of the web page, there is a legend that shows the symbols for flood embankments, river training structures and their names.

Fig. 6 shows the flood depth and flood risk. The left of the web page contains two panels. One is the legend regarding the background geographical information, the other is the search panel for finding specific features and zooming in using keywords. The depth and velocity in the simulation results are illustrated in the center frame. The feature regarding the flooded villages in the scenario, including provinces, counties, farmland area, and economic loss, are listed at the bottom. Whenever users click on a row in the list, the web page zooms in on that particular location.

4. Conclusions

This paper has introduced a framework for rapid flood simulation mapping and its web visualization when handling large quantities of data. The framework was built on a web GIS architecture and uses the ArcGIS server. Three types of data can be drawn upon: flood risk; background geo-information; and images. The ArcGIS JavaScript APIs were specifically coded for the web visualization of flood risk results.

There are two crucial parts to the framework. One is fast simulation using parallel computation and transformation of the simulation results into structured shapefiles that can be used by the ArcGIS Server. The shapefiles contain the results regarding the inundation area, depth, and velocity, as well as basic geographical information such as river training structures, flood embankments, intersections, and floodplains. The other key part is the web visualization based on the ArcGIS JavaScript APIs, which is used to illustrate the flood risk results, primarily depth by classified colors and the velocity field.

The proposed framework was applied to the lower Yellow River, from the Xiaolangdi reservoir to the river delta, a total of 786 km. More than 1.8 million people live in the floodplains of the lower Yellow River, which covers 3956 km², with 2052 villages and 250000 hectares of farmland. The
framework was tested using a scenario with three input flows from the Yellow River, the Qin River and the Yiluo River. The whole process, from the initial parallel computation to the web visualization took less than 3 hours, which meets the needs of flood management. In the test scenario, the data transformation and its web visualization took place automatically. The web page showing the flood risk map performed well from the point of view of user interaction.

Although the framework works well for large-scale flood simulations, such as in the lower Yellow River, there are limitations that still need to be improved. First of all, because the parallel simulation and the web visualization work on Linux and windows platforms respectively, there is a data interaction process that needs to take place between the two platforms. To avoid this, an open source web GIS and data visualization solution needs to be developed that can perform the geographical data management in the Linux environment, thus sharing the same platform as the parallel simulation of flood propagation. Secondly, more functionality needs to be coded into the web visualization to improve the user experience, for instance providing a comparison of depth between two potential scenarios.

5. Acknowledgments
The authors acknowledge partial support from the National Key Research and Development Program of China (2016YFC0402409-07, 2018YFC0407905) and the National Natural Science Foundation of China (Grant No. 51809108).

6. References
[1] Kalantari Z, Nickman A, Lyon S W, Olofsson B and Folkeson L 2014 A method for mapping flood hazard along roads J. Environ. Manage. 133 69–77
[2] Sahoo S N and Sreeja P 2015 Development of Flood Inundation Maps and Quantification of Flood Risk in an Urban Catchment of Brahmaputra River ASCE-ASME J. Risk Uncertain. Eng. Syst. Part Civ. Eng. A4015001
[3] 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 Adv. Water Resour. 33 1456–67
[4] Jen-Kuo Huang, Ya-Hsin Chan and Kwan Tun Lee 2016 Real-Time Flood Forecasting System: Case Study of Hsia-Yun Watershed, Taiwan J. Hydrol. Eng. 21 5015015–5015031
[5] Wu H, Adler R F, Tian Y, Huffman G J, Li H and Wang J 2014 Real-time global flood estimation using satellite-based precipitation and a coupled land surface and routing model Water Resour. Res. 50 2693–717
[6] Alfierei L, Cohen S, Galantowicz J, Schumann G J P, Trigg M A, Zsoter E, Prudhomme C, Kruczkiewicz A, Coughlan de Perez E, Flamig Z, Rudari R, Wu H, Adler R F, Brakenridge R G, Kettner A, Weerts A, Matgen P, Islam S A K M, de Groeve T and Salamon P 2018 A global network for operational flood risk reduction Environ. Sci. Policy 84 149–158
[7] A. Lagmay A M F, Racoma B A, Aracan K A, Alconis-Ayco J and Saddi I L 2017 Disseminating near-real-time hazards information and flood maps in the Philippines through Web-GIS J. Environ. Sci. 59 13–23
[8] da Cruz e Sousa R and Miranda O L 2018 Incorporating wetlands in hydrologic and hydraulic models for flood zone delineation: An application to Durán, Ecuador Int. J. Disaster Risk Reduct. 28 375–83
[9] Ben Khalifallah C and Saidi S 2018 Spatiotemporal floodplain mapping and prediction using HEC-RAS - GIS tools: Case of the Mejerda river, Tunisia J. Afr. Earth Sci. 142 44–51
[10] Kulkarni A T, Mohanty J, Eldho T I, Rao E P and Mohan B K 2014 A web GIS based integrated flood assessment modeling tool for coastal urban watersheds Comput. Geosci. 64 7–14
[11] Noh S J, Lee J-H, Lee S, Kawaike K and Seo D-J 2018 Hyper-resolution 1D-2D urban flood modelling using LiDAR data and hybrid parallelization Environ. Model. Softw. 103 131–45
[12] Barthélémy S, Ricci S, Morel T, Goutal N, Le Pape E and Zaoui F 2018 On operational flood forecasting system involving 1D/2D coupled hydraulic model and data assimilation J. Hydrol.
[13] Huang X, Wang C and Li Z 2018 A near real-time flood-mapping approach by integrating social media and post-event satellite imagery Ann. GIS 24 113–23

[14] Seipel S and Lim N J 2017 Color map design for visualization in flood risk assessment Int. J. Geogr. Inf. Sci. 31 2286–309

[15] Kim D, Cho H, Onof C and Choi M 2016 Let-It-Rain: a web application for stochastic point rainfall generation at ungaged basins and its applicability in runoff and flood modeling Stoch. Environ. Res. Risk Assess.

[16] Zavala-Romero O, Ahmed A, Chassignet E P, Zavala-Hidalgo J, Fernández Eguiarte A and Meyer-Baese A 2014 An open source Java web application to build self-contained web GIS sites Environ. Model. Softw. 62 210–20

[17] Chien N Q and Keat Tan S 2011 Google Earth as a tool in 2-D hydrodynamic modeling Comput. Geosci. 37 38–46

[18] Rubalcava R 2015 ArcGIS web development (Shelter Island, NY: Manning Publications Co)

[19] Anastasiou K and Chan C T 1997 Solution of the 2D shallow water equations using the finite volume method on unstructured triangular meshes Int. J. Numer. Methods Fluids 24 1225–1245

[20] Lu X and Xie S 2016 Conventional versus pre-balanced forms of the shallow-water equations solved using finite-volume method Ocean Model. 101 113–120

[21] Roe P L 1997 Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes J. Comput. Phys. 135 250–8

[22] Barth T and Jespersen D 1989 The design and application of upwind schemes on unstructured meshes (American Institute of Aeronautics and Astronautics)

[23] Dick E 2009 Introduction to Finite Volume Methods in Computational Fluid Dynamics Computational Fluid Dynamics ed J F Wendt (Berlin, Heidelberg: Springer Berlin Heidelberg) pp 275–301

[24] Gropp W, Lusk E and Skjellum A 2014 Using MPI: portable parallel programming with the Message-Passing-Interface (Cambridge, Massachusetts: The MIT Press)

[25] Karypis G and Kumar V 1998 A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs SIAM J. Sci. Comput. 20 359–92

[26] Burke R 2003 Getting to know ArcObjects: programming ArcGIS with VBA (Redlands, Calif: ESRI Press)

[27] Chang K-T 2008 Programming ArcObjects with VBA: A Task-Oriented Approach (CRC Press)

[28] Doman K 2015 Mastering ArcGIS Server Development with JavaScript

[29] Vijayaraghavan J and Dhanapal Y 2016 ArcGIS for JavaScript Developers ArcGIS by example

[30] Gill R, Riecke C and Russell A 2008 Mastering Dojo: JavaScript and Ajax Tools for Great Web Experiences

[31] Li G 2009 Questions and answers about the Yellow River (Yellow River Conservancy Press)

[32] Lai R, Wang M, Yang M and Zhang C 2018 Method based on the Laplace equations to reconstruct the river terrain for two-dimensional hydrodynamic numerical modeling Comput. Geosci. 111 26–38