A Machine Learning Approach for Forecasting and Visualizing Flood Inundation Information

Citation for published version:

Digital Object Identifier (DOI):
10.1680/jwama.20.00002

Link:
Link to publication record in Heriot-Watt Research Portal

Document Version:
Peer reviewed version

Published In:
Proceedings of the Institution of Civil Engineers: Water Management

Publisher Rights Statement:
Copyright © ICE Publishing 2020, all rights reserved

General rights
Copyright for the publications made accessible via Heriot-Watt Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
Heriot-Watt University has made every reasonable effort to ensure that the content in Heriot-Watt Research Portal complies with UK legislation. If you believe that the public display of this file breaches copyright please contact open.access@hw.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.
Accepted manuscript

As a service to our authors and readers, we are putting peer-reviewed accepted manuscripts (AM) online, in the Ahead of Print section of each journal web page, shortly after acceptance.

Disclaimer
The AM is yet to be copyedited and formatted in journal house style but can still be read and referenced by quoting its unique reference number, the digital object identifier (DOI). Once the AM has been typeset, an ‘uncorrected proof’ PDF will replace the ‘accepted manuscript’ PDF. These formatted articles may still be corrected by the authors. During the Production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal relate to these versions also.

Version of record
The final edited article will be published in PDF and HTML and will contain all author corrections and is considered the version of record. Authors wishing to reference an article published Ahead of Print should quote its DOI. When an issue becomes available, queuing Ahead of Print articles will move to that issue’s Table of Contents. When the article is published in a journal issue, the full reference should be cited in addition to the DOI.
Submitted: 02 January 2020

Published online in ‘accepted manuscript’ format: 12 May 2020

Manuscript title: A Machine Learning Approach for Forecasting and Visualizing Flood Inundation Information

Authors: Syed Kabir¹, Sandhya Patidar¹ and Gareth Pender²

Affiliations: ¹Institute for Infrastructure and Environment, Heriot-Watt University, Edinburgh, UK and ²Deputy Principal Research and Innovation, Heriot-Watt University, Edinburgh, UK

Corresponding author: Syed Kabir, Gait 4, Heriot-Watt University, William Arrol Building, EH14 4AS, Edinburgh, UK. Tel.: +447442772463.

E-mail: Syed.Rezwan.Kabir@gmail.com
Abstract

This paper presents a new data-driven modelling framework for forecasting probabilistic flood inundation maps for real-time applications. The proposed end-to-end (rainfall-inundation) method combines a suite of Machine Learning (ML) algorithms to forecast discharge and deliver probabilistic flood inundation maps with three-hour lead time. The study applies Random Forest (RF)-based rainfall-discharge models on top of Multi-layer Perceptron (MLP)-based classifiers to classify wet/dry cells. The concept of an ML-based hybrid modelling framework is tested using two subsets of data created from an observed fluvial flood event in a small flood prone town in the UK. The results show that the model can effectively emulate the outcomes of a hydrodynamic model (i.e. Flood Modeller) with considerably high accuracy measured in terms of flood arrival time error and classification accuracy. The mean arrival time difference between the proposed model and the hydrodynamic model is of order 1 hour 53 minutes. The classification accuracy is measured against a radar image and the accuracies read 88.22% and 86.58% for the proposed data-driven and Flood Modeller (FM) model, respectively. The key features of the proposed modelling framework are that it is simple to implement, detects flooded cells effectively and substantially reduces computational time.

Keywords: Hydrology and water resource; Computational mechanics; Machine learning; Inundation modelling
1 Introduction

Due to rapid urbanisation and unprecedented climate change impacts the need for hydrodynamic modelling for urban flood management in real-time is greater than ever before (Miller and Hutchins, 2017). Conventionally used two dimensional (2D) hydrodynamic models transform the bulk river discharge into potential flood hazards, specifying the extent, depth, and velocity. Thus, they provide most of the information required by the decision makers working in the areas of flood risk management (Bates et al., 2014). However, for the purpose of operational flood forecasting, hydrological modelling is limited to the generation of the discharge hydrographs without the simulation of 2D hydrodynamic models (Bhola et al., 2018). This is mainly because the 2D hydrodynamic models, in general, have complex model structure with high parameter dimensionality, and thus they require a considerably high computational time for producing dynamic real-time flood maps. The ideal solution to these various technical challenges would be to develop simple models that could significantly reduce the computational burden while still being able to generate practically useful and statistically significant information in real-time.

Recently, researchers have developed computationally efficient data-based mechanistic models (Romanowicz et al., 2008). These models are integrated with a simplified 2D hydrodynamic model (i.e. LISFLOOD-FP) to produce real-time flood inundation maps (Leedal et al., 2010). To further reduce the computational time for simulation of 2D hydrodynamic models during a real-time event, some offline methods have also been proposed (Bhola et al., 2018; Henonin et al., 2013). However, these methods have certain limitations, for example: i) they require multiple platforms working in parallel; ii) they generate scenarios to build the large databases of probabilistic maps which could be labour intensive and demand large storage capacities; iii) the infrastructure requires regular maintenance, a major issue in operational applications; iv) the database needs to be regularly updated for any topographical changes, especially for urban areas.

On the other hand, application of Machine Learning (ML) techniques for rainfall-runoff forecasting has been investigated for decades (Solomatine and Ostfeld, 2008). Mosavi et al. (2018) have reviewed 180 articles that have used multiple ML techniques for both short and long term flood prediction. Yaseen et al. (2015) reviewed AI-based models for streamflow forecasting from 2000 to 2015, with a focus on current challenges and opportunities for prospective research and future applications. Furthermore, Raghavendra and Deka (2014) have comprehensively reviewed the Support Vector Machine (SVM) applications in the field of hydrology. However, research on the applications of ML for flood inundation modelling/simulation is, to the best of authors knowledge, significantly understudied. Liu et al. (2009) shifted the development of the fast flood modelling paradigm towards ML, possibly for the first time. Their work has shown that outputs from a Coarse Grid Model (CGM), which are computationally less expensive than the Fine Grid Models (FGMs), can be used with a non-linear regression model, i.e. Support Vector Regression (SVR), to predict the outputs from an FGM. And by doing so, it is possible to emulate the outputs of an FGM without having to run FGM simulations. Their idea was to generate water depth and velocity values at some selected locations, using both the CGM and FGM runs for multiple inflow hydrographs. To further reduce the dependency upon CGM runs, Liu and Pender (2015) proposed another SVR based modelling approach, which ignores the outputs from CGMs in the training process. They only used flow observation time and flow rates as input variables.
and FGM outputs (water depth and velocity) as the target variable. In this method, water depth/velocity time series of selected locations are normalised to its arrival time to overcome the complexity associated with discontinuity of the flood wave arrival time. The results showed that the proposed model could sufficiently predict the water depth and velocity values when compared against the FGM generated outcomes. The ISIS-2D inundation model was used to generate FGM training and validation samples.

With a different concept, Chang et al. (2010) proposed a hybrid ML based inundation model that uses a clustering (K-means) step to categorize the different flooding characteristics (e.g. water depth, surface elevation) in the study area. The centre of the clusters is identified as control points. Once the control points are detected, a separate Back Propagation Neural Network (BPNN) is trained for each control point using rainfall data from the nearby stations and water depth values at these points (extracted from 2D hydrodynamic model simulations). Linear regression models are fitted to the grids that are highly correlated to the control points in each cluster, and a multi-grid BPNN is used for the grids that do not correlate. The multi-grid BPNN models use rainfall, water depth values of the corresponding control point at time \( t \) and \( t+1 \), Manning’s roughness value, the elevation of the grid, distance from control points to the grid, and water depth at \( t \) and \( t-1 \). As it is not possible to have observed depth values during an event, the models use time-lag water depths values as an input. Therefore, the authors have used the output from the BPNN models as an input for the next time step. They constructed five single-grid BPNNs for five control points, five multi-grid BPNNs for 1474 loosely correlated grids (nonlinear), and 1022 linear regression models for highly correlated grids to forecast 1h ahead inundation depths. This hybrid approach was tested in the Dacun Township in central Taiwan and results showed that the model has the ability to forecast 1h ahead flood inundation maps at 80m x 80m spatial resolution, continuously and adequately.

Pan et al. (2011) used only rainfall data to forecast flood depth at 19 sites in Yunlin County, Taiwan. They used rainfall data from nearby stations with different order as inputs and Principal Component Analysis (PCA) for dimensionality reduction; then BPNN models were trained (the target was the water depth values simulated from a 2D model) to forecast \( t+1 \)h ahead water depth.

In many previous studies, it has been discussed and suggested that it is not an efficient approach to recursively adopt forecasted values for many time-steps ahead in the future, which could result in accumulation of model error and performance degradation at each time step (Chen et al., 2013, Parlos et al., 2000). To reduce the error accumulation, Shen and Chang (2013) investigated the possibility of applying multiple Recurrent Neural Network (RNN) structures on 13 sites in Taiwan. They concluded that the recurrent configuration of the Nonlinear Autoregressive Exogenous (R-NARX) network can significantly inhibit error accumulation when implemented for generating multi-step ahead flood inundation forecasts for a longer period in the future.

Chang et al. (2014) presented a new Artificial Neural Network (ANN) based hybrid model for real-time multistep-ahead flood inundation mapping. In their approach, they first divided the study area into several sub-areas according to features such as villages, towns, and cities. In the next stage, the spatial distribution of flood depths information (collated from 989 datasets) was clustered into 25 inundation topological maps using a Self-organizing Map (SOM), a special type of ANN designed for the clustering operation. These topological maps represented the spatial distribution of flooding for 25 types of hydrological conditions. Then the R-NARX network was constructed (one for each sub-area) using rainfall and total inundated volume. The total inundated volume is the value that quantifies the inundation condition of the sub-area and can be calculated by aggregating the depths of all the grids in
the sub-area. The output of the R-NARX model is the total inundated volume that quantifies a specific hydrological condition. The forecasted total inundation depth was then compared to the total inundation volume of the 25 topological maps. The closely matched map was selected, and the total inundation volume of the map was adjusted using the forecasted total inundation volume. Further, an updated 40m x 40m map of a sub-area was produced from the adjusted total inundation volume through a proportional interpolation method. The results were compared against the hybrid model proposed by Chang et al. (2010) for a design storm event in the Yilan County, Taiwan, and provided better forecasting performance in terms of smaller mean absolute errors and maximum absolute errors. Using the same idea, Chang et al. (2018a) developed a SOM-RNARX based regional flood inundation model for up to 12h ahead forecasting in the Kemaman River Basin, Malaysia. This ML based modelling approach was further improvised in Chang et al. (2018b), where they have proposed an Intelligent Hydroinformatics Integration Platform (IHIP) to provide a user-friendly web interface for improved online forecasting capability and flood risk management.

Thus, to summarise, the ML paradigm in flood inundation modelling has followed two major approaches. Approach 1 aims to estimate water depth values at different locations within a floodplain using inflow hydrographs as inputs to the model or rainfall from nearby gauging stations. Approach 2, however, applies a clustering method to group flood zones under different hydrological conditions and linear/nonlinear models are used to generate water depth values for a particular hydrological condition (rainfall condition). Both approaches have produced promising results by utilizing the advantages of 2D hydrodynamic models and simple ML algorithms. However, the proposed modelling approaches have limitations and need to be researched further. For example, none of these studies attempted to quantify the model uncertainties. Specifically, in approach 2, using the forecasted water depth values as an input recursively to forecast depths for the next time-step results in the accumulation of forecasting error and thus gradually increases model uncertainty. Application of such models for generating deterministic flood maps in real-time for a longer temporal horizon can be misleading (significantly over- or underestimating water depths). Furthermore, approach 2 requires many inundation maps generated from 2D hydraulic models under a multitude of hydrological conditions for the clustering purpose. Producing a large number of inundation scenarios at very high spatial resolution can be extremely laborious and computationally expensive. Moreover, to reduce computational costs a ground resolution of about 40 meters was used, which is considerably large, specifically for the urban areas. While so far, spatial resolution and recurrent use of water depth values have been two key limitations of approach 2, approach 1 has followed a different strategy to extract water depths. The approach 1 methods do not require a large number of 2D model runs and are simpler than approach 2, but these approaches have certain limitations as well: i) the use of linear interpolation method to spatially distribute the water depth values, ii) their ability to generate multi-step ahead forecasting in real-time has not been fully tested yet, and iii) the application of these approaches requires the upstream boundary conditions to be forecasted/measured before this method can be applied.

To address these challenges, the current paper proposes a new data-driven framework that combines a non-linear regression based hydrological model, a computationally efficient wet/dry cell classification module, and a data visualisation schematic for generating dynamic flood inundation maps. High-resolution data visualisation is achieved by applying a spatial interpolation technique, which uses auxiliary information available from a high-resolution Digital Elevation Model (DEM). Conceptually, the framework can be viewed as a 1D-2D
linked model, where the initial part (regression) of the system acts as a 1D model and output from the model is turned into a geospatially varied extent map (2D). The research activity focuses on the town of Upton-upon-Severn, which is situated on the west bank of the River Severn in the Malvern Hills District of Worcestershire, England. The analysis is done for the time frame covering the flooding event of October-November 2000.

2 Methodology

2.1 Models and materials
The proposed framework (described in Section 2.2), at its core, makes use of multiple ML techniques and a 2D hydrodynamic model. The ML techniques include Random Forests (RF) and Multi-layer Perceptron (MLP).

2.1.1 Random Forest
The RF is one of the most popular and powerful supervised ML algorithms that, from a computational standpoint, is capable of both regression and classification tasks, is relatively fast to train, and depends on only a small number of tuning parameters (Cutler et al., 2012). The RF, first introduced by Leo Breiman (Breiman, 2001), is a decision-tree-based ensemble method. Each decision-tree, a flowchart-like structure, can be created through a binary recursive partitioning procedure, which generates binary splits of distinct individual variables in the predictor space (input space). The entire predictor space is often referred to as the root node and is held within the topmost node of the decision-tree. Other internal nodes of the decision-tree, in general, split into two descendant nodes. The decision-tree (originating from the root node) selects the next descendant node based on an attribute testing algorithm that intends to optimise decision making. The end nodes of the decision-tree are called leaf or terminal nodes. The splitting process continues recursively until a user-defined stopping criterion is reached.

RF technique uses the so-called bootstrapping method to create random datasets from the training data. Within the RF algorithm, the size of the bootstrapped dataset is the same as the original dataset. The decision-trees (base learners of RF) are fitted to the datasets (generated from the bootstrapping method) for a random subset of predictor variables at each step. Using a bootstrapped sample and randomly selecting a subset of the predictor variables at each step results in a wide variety of different forms of decision-trees. The variety is what makes RF more effective than individual decision trees. When a new sample is passed to the trees created from bootstrapped data, the responses from these individual trees are combined by unweighted voting (classification) or unweighted averaging (regression) to make the final optimised decision.

2.1.2 Multi-layer Perceptron
The MLPs, the most common form of ANN, consist of several weighted connections between nodes and are arranged in three different layers: input, hidden and output layers. The processing units (neurones) in the hidden and output layers multiply the input signals propagating through the layers with a set of weights from the weight matrix and transform it into the overall system output/response using an activation function.
The activation functions are differentiable functions and depending on the problem under investigation these functions can be both linear and non-linear in nature (Hagan et al., 2002). An activation function maps the response of a neuron for a given set of input variables. These functions are selected to meet the requirements of the system and the problem under investigation, accordingly. For example, for a binary classification problem the ‘sigmoid’ function is used in the output layer, while the Rectified Linear Unit (ReLU) function is widely used in the hidden layers. Following an iterative procedure, the network computes an output error with respect to the target values for a given set of input vectors and modifies the weight and bias values according to a learning rule, commonly referred to as the training of the neural network. The training process is continued until a stopping criterion is reached (error goal, and/or a maximum number of iterations/epochs). A more detailed explanation of the learning rules and other forms of ANNs can be found elsewhere, e.g., Hagan et al. (2002).

2.1.3 Hydrodynamic modelling

One-dimensional or 1D modelling solves the 1D Shallow Water Equations (SWE) of flow in the channel. Such modelling is fast to run and is good at representing in-channel water level and flows, and is efficient for modelling point features such as the bridges, weirs, sluices etc. However, 1D modelling has some disadvantages, too. For example, the calibration of a 1D model requires detailed information on various parameters such as full identification of major flow routes. In addition, no information is provided on velocity distribution, and simulation outcomes can be poor in urban areas. On the other hand, 2D modelling solves the 2D SWE and calculates water depth and ‘depth averaged’ velocity on a grid. 2D modelling does not require predefined flow routes and is thus easy to set up and is more accurate for urban flow routing. However, 2D models are slow to run due to increased complexity. To address some of these limitations, a linked 1D and 2D model, combining the key features of 1D and 2D modelling, is preferred. This study utilizes the 1D-2D linked modelling feature of the Flood Modeller (FM) software for simulating the flood events in order to generate training data for the ML models.

2.2 Framework for data driven flood inundation mapping

This section presents a detailed description of the proposed data-driven modelling framework for generating real-time probabilistic flood inundation maps from observed rainfall and upstream flow data. The modelling framework consists of two core components: A. developing the pre-trained/fitted model databases; and B. generating real-time flood inundation maps (Fig. 1).

2.2.1 Developing the pre-trained/fitted model databases

In the first step, the RF-based three hours ahead flow forecasting models are trained. These models are trained using antecedent rainfall and flow data. It is worth mentioning that an initial investigation revealed that the upstream flow forecasting models based on RF technique outperformed the models based on SVR technique. Therefore, RF is selected as the preferred flow forecasting modelling approach in this study. Secondly, the MLP-based models that predict the states (wet/dry) of the sampled cells, i.e. 150 Ground Control Points (GCPs), within the floodplain are trained. These classifiers are trained using the outputs from FM at a coarse grid resolution (30m in this study) as the target and upstream flow as the input variables. One of the major benefits of using MLP is that the model can generate multiple outputs for a specified input dataset. That is, a single MLP classifier (i.e. a global MLP) can classify states (wet/dry) of all the GCPs for a given upstream boundary condition at a time. However, in this study, a separate model is trained for each GCP (local MLP) instead of a global model. This is because an initial investigation showed that overall results obtained using local MLPs were slightly better than the outputs.
from a global MLP. However, a global model can be used for large scale modelling as it reduces model training time significantly without a major drop in accuracy. The processes of training the RF and MLP-based models are further described in Section 3. The validated flow forecasting and classifier models are stored in the databases for future applications.

2.2.2 Generating real-time flood inundation maps
In the third step, for real-time operation, the pre-trained RF-based models are loaded for forecasting three-hour ahead upstream flows using observed rainfall and discharge. That is, the models forecast the discharge at the upstream with three-hour lead time for every new observation. During a flood event, the forecasted discharge crosses a certain threshold which loads the MLP-based cell classifiers. These models predict the states of the GCPs (i.e. flooded/dry). Finally, these classified cells (wet/dry) are then used to predict the state of the unsampled cells using Regression Kriging (RK) method (Hengl et al., 2007). The so-called RK is a spatial interpolation technique that uses distance as well as other auxiliary information between sampled points to predict the values for the unsampled points. In this study, surface elevation is used as auxiliary data to predict states of the unsampled cells. The final outcomes are moderately higher resolution (9m) interpolated probabilistic maps with three-hour lead time which are sequentially generated for every new rainfall and flow observations (hourly). The probabilistic maps could be made available through ‘WebGIS’ for real-time decision making.

2.3 Evaluation metrics
To assess robustness and reliability of the modelling framework, forecasted results generated from the integrated framework are compared using a range of goodness of fit statistics (Table 1). To quantify the uncertainty in flood arrival times, the residual variations are analysed. The residuals are the predicted time differences between the FM and ML classifiers at sampled locations in the first arrival of the flood wave. Prediction of flood water presence/absence at sampled pixels is spatially interpolated to produce moderately high-resolution probabilistic maps. These probabilistic maps are visually compared with the reference deterministic maps produced by the FM simulations. Finally, the classification accuracy between the FM and proposed data-driven method is compared against a reference satellite image captured on 8th November 2000.

3 Experimental setup

3.1 Study area
The town of Upton-upon-Severn is situated on the west bank of the River Severn in the Malvern Hills District of Worcestershire, England. According to the Environment Agency (EA, 2015), since 1970, the area has seen over 70 flood events and has often been dubbed as the most flooded town in the UK. The study domain covers an area of about 5.27 km² within the town of Upton-Upon-Severn. To develop the proposed methodology, at first 150 locations as GCPs within the study area are selected randomly. These locations covered a good mix of cells that had experienced flooding and those that were not flooded due to higher surface elevation (Fig. 2). The spread and selection of these samples is strategic to facilitate a robust model development.
3.2 Hydrometeorological data pre-processing

The hydrometric data used in this study consisted of hourly flow at upstream, near Severn Stoke (British National Grid coordinates: X: 384733, Y: 249922) for the October - November 2000 flood event. The time series of observed flow starts on 28/10/2000 at 13:45 (as time step 0) and stops at 15:45 on 17th November 2000 (time step 485). The hydrograph of the data has a peak (899.19 m$^3$/s) recorded on 03/11/2000 at time 04:45 and a second peak (832.49 m$^3$/s) recorded on 08/11/2000 at 20:45. Figure 3A presents the complete observed flow hydrograph that includes corresponding rainfall intensities, and Figure 8.3B and 8.3C show the subsetted hydrographs created from the observed hydrograph with corresponding rainfall values. Hourly rainfall data were downloaded from the Met Office Integrated Data Archive System (MIDAS) for Great Malvern (src id 670) and Pershore (src id 657) meteorological stations. Hourly data from both stations were aggregated and used as a proxy for the catchment average rainfall.

In addition, four synthetic hydrographs with different peak and duration were created to train the MLP-based classifiers (Fig. 4). Synthetic flow series can be created in different ways. For this study, synthetic hydrographs were generated using a Hidden Markov Model (HMM) analogous to Patidar et al. (2018) and Pender et al. (2016). Table 2 lists the areas of the model development covered by all six hydrographs along with their peak values.

3.3 Inflow hydrograph forecasting

The upstream boundary conditions were unavailable at the river cross section node MC033 (Fig. 2). Therefore, it was not possible to use the observed volume of incoming channel water at the exact upstream of the floodplain. The Severn Stoke station (Node- MC010) is located approximately 9 km in the North from the study area. Since the purpose of the study was to investigate the applicability of ML for inundation extent modelling of the Upton-Upon-Severn town in real-time the complete hydraulic modelling of the region was ignored. Therefore, there was no estimated flow budget (volume of water transported to and from the channel to the land) from Severn Stokes to Node MC033. Instead, the flows were forecasted at section MC033 using FM 1D unsteady simulations. This provided the approximated flow hydrographs for the river cross section node at top of the floodplain. This step is void for large scale modelling where required upstream values are made available.

As mentioned earlier, RF-based models were used to forecast T+3h flows using catchment average rainfall. The RF-based model for forecasting T+3h ahead flow was first developed and cross validated at upstream (Severn Stoke, Section ID MC010) and then a separate RF-based model was developed and cross validated for generating forecasts at T+3h ahead flow at section ID MC033. It is indeed possible to select other lead times as well. However, 3h were assumed to be a good starting lead time considering the size of the study domain. The input-output structure of the model can be defined by the following equation:

$$f[(Q_{t+3}, (R_t, R_{t-1}, R_{t-2}, R_{t-3})]$$

Where, target variable $Q_{t+3}$ is the flow at $t+3$ hours; $f(.)$ is the model function; $R_t$ is the current flow with its lags (in hours) and $R_{t-3}$ is the rainfall with its lags (in hours). The skgarden package in Python programming language was used to configure the RF-based models.
3.4 Simulating the linked 1D-2D FM model to generate training samples for MLP-based classifiers

To generate the target data (water depth) for training and testing the MLP-based models, FM 1D-2D linked model was applied. To keep computational time comparatively low, a 30m DEM and larger time steps of 5 seconds were used. The alternating direction implicit solver was selected due to its shorter runtime features and robustness. In addition, it did not require any supercritical flow calculation. The channel and floodplain roughness values used to run the model were 0.025 and 0.05, respectively (see the FM manual (JACOBS, 2017) for 1D-2D linked model setup guidance). Six separate 1D-2D linked model runs were performed with six hydrographs (used as the upstream boundary conditions at Severn Stoke station (FM cross section node MC010) to estimate water depth values at 150 sampled locations (GCPs). Note that this section is situated further to the North of the study domain. Water depth values from these sampled locations were then used to train and validate the MLP-based classifiers. Furthermore, to visually compare the final interpolated probabilistic flood inundation maps, deterministic flood inundation maps at 9m resolution were also generated for subset 1 and 2 hydrographs.

3.5 Configuring the MLP-based classifier and forecasting inundated cells

For training MLP-based classifiers, the water depth values obtained from four synthetic hydrographs were binarized. To delineate wet (‘1’) and dry (‘0’) cells, a threshold value was applied: water depth > 0.09m = wet and water depth < 0.09m = dry. The threshold value of 0.09m was selected to ignore very low water depths (seen as outliers) during the MLP training process. To cover the spatial range, 150 separate MLP-based models were trained to forecast for T+3h ahead wet/dry state exploiting the Keras data framework in Python. A summary of the model architecture and parameters is given in Table 3. About 80% of the total training samples were used to train the models and 20% of the samples were used for validation. Then, the efficiency of trained models was estimated by comparing the predicted results against the binarized data (reference data) obtained from subset 1 and 2 hydrographs. That is, the flooded cells were predicted using the fitted MLP-based classifiers for subset 1 and 2 hydrographs, and then compared them against the outputs from FM. The trained MLP-based models developed in this step were used to build the database of hydraulic classifier.

3.6 Developing the pre-trained/fitted model databases

Two databases were developed, as described in Section 2.2, after training the RF and MLP-based models. The database of hydrological models contained the trained RF based three hours ahead flow forecasting models and the database of hydraulic classifiers contained the trained MLP-based models that predict the states (wet/dry) of the GCPs within the floodplain.

3.7 Forecasting wet/dry cells in real-time from rainfall data

To forecast inundated areas within the floodplain, first, T+3h ahead river discharges at upstream and MC033 were forecasted using the pre-trained RF-based models. Then the T+3h flow values were fed into the pre-trained MLP-based classifiers to forecast ‘wet/dry’ state of the sampled locations. The probabilistic outputs generated by the MLP-based classifiers were converted into 1s and 0s. Cells having a probability of less than 0.5 were encoded as ‘0’ and cells with values greater than 0.5 were encoded as ‘1’. To assess the robustness of the MLP-based classifiers, the classification task was conducted for both subset 1 and 2 and compared against the corresponding reference data (see results and analysis). Thus, each of the sampled locations was assigned a binary value during the entire flood event at hourly time-steps. Initially, all the sampled pixels were classified as dry and then the pixels started to change their states subsequently with the increase in flow rate.
3.8 Inundation map generation

Finally, to predict the state of unsampled locations as time progressed, RK interpolation method was employed using surface elevation as the auxiliary information. To select the best fitting variogram model for the RK, a robust analysis was conducted using Pykrig package in Python. 80% of the total data points were used to fit the RK model and the rest of the samples were used for validation. Five variogram models, specifically Spherical, Exponential, Linear, Power and Gaussian were tested. Best accuracy was found using the Exponential model. The coefficient of determination (R-squared) was found to be 0.314 when the presence and absence of water were regressed against corresponding elevation values. The fitted variogram was then applied to a moderately high-resolution DEM (9m) to predict the presence/absence of water at unsampled locations. The results obtained were in the form of a probability value of flood occurrence throughout the study area based on 150 samples at 9m resolution.

4 Results and analysis

4.1 Inflow hydrograph forecast

The first step towards the framework development involved the fitting of a robust ML based hydrological model to forecast flow at T+3h for upstream (MC010) and at the MC033 node. For this, RF-based rainfall-discharge models were trained and tested. A two-fold cross validation method was adopted: Case 1 - the models were trained on subset 1 and validated on subset 2; and Case 2 - the models were trained on subset 2 and validated on subset 1. For a systematic quantification of forecasting accuracy, three error statistics were calculated (Table 4): i) The R-squared value – it compares the goodness of fit of the forecasting models; ii) The RMSE – it measures the standard deviation of prediction errors; and iii) the NSE – it quantifies how well the model simulation predicted the output variable (Nash and Sutcliffe, 1970). For both cases, high values for R-squared and NSE and low values for RMSE indicate that RF-based models were highly accurate in forecasting 3h ahead flows (Table 4).

4.2 Wet/Dry cell forecast

The MLP simulated results were first compared against the FM outputs in terms of arrival time (Table 5, columns 2-8). The results from all the sampled locations show that for Case 1, the mean error observed in the flood arrival time was about +1h 51min (95% CI [-2h, +6h]). Thus, on average, there could be an up to 1h55m delay in predicted arrival times. For case 2 (RF forecasted flows for subset 1), the observed mean error was of order +1h 51min (95% CI [-2h, +6h 27min]).

To estimate the classification accuracy of the models, three distinct measurements were estimated for both subsets: i) overall accuracy; ii) F1-score; and iii) F-scores. The overall accuracy and F1-score (harmonic mean of recall and precision) were calculated from a confusion matrix and values were averaged across all 150 locations (Table 5, columns 9-10). The overall accuracy measure describes how accurately GCPs were classified during the flood event; the F1-score compares how well the models correctly classified flooded cells during the event. The GCPs which were dry throughout the flood event were excluded from F1-score calculation. The F-scores were calculated using the method described in Section 2.3. They measure how many sampled locations were correctly classified at a particular time step.
To present the yielded $F$-scores, both Case 1 and 2 flood events were divided into several time segments: the first segment consisted of the dry hours prior to the flood arrival, followed by several segments of 48 hours during the event, with the last segment consisting of the remaining hours of the event. The sampled locations started to get flooded at the 32$^{nd}$ hour. Arrival time errors were detected for the very first flooded sampled location. This location experienced first flood water at the 29$^{th}$ hour into the event while classifiers forecasted the location to be flooded at the 32$^{nd}$, resulting in 3 hours of delay. Since it was only one location which was misclassified as a dry cell for 3 hours, the $F$-score was not calculated for this, as it would be a 0 in such cases. The $F$-scores were calculated from the 32$^{nd}$ hour onwards in 48 hours segments. Table 6 shows the averaged $F$-score for the various segments.

The mean overall accuracy and $F1$-score of the two case studies were 0.987 and 0.982, respectively. Average arrival time error estimated as 1h 53min delay with 95% CI interval for both simulations as, Case 1: -2h, 6h and Case 2: -2h, 6h 27min. The classifiers on average yielded 98.51% of accuracy detecting the true state of the sampled locations for every time step.

### 4.3 Interpolated flood extent maps

High accuracies, as noted for wet/dry cell classification using ML models, indicate that the idea of producing probabilistic maps from data points could be the potential solution to the real-time flood inundation visualization problems. Figure 5A and 5B show a forecasted probabilistic flood inundation map from the RK method and deterministic map (water depth measured in meters) from the FM software derived for Case 1 at 122$^{nd}$ hour (flood peak period), respectively. The water depth values from FM are very high at this time-step and the town is severely flooded. The same feature is seen in the probabilistic map. Higher probabilistic values indicate that the proposed method was able to successfully detect the flooded cells.

In addition, an Advanced Synthetic Aperture Radar (ASAR) image which was captured on 8$^{th}$ November 2000 at 12.18pm was acquired. This capture is almost in line with the flood map predicted at 111$^{th}$ hour of the Case 1 runs. Therefore, both the FM simulated water depth and the interpolated map were binarized and compared against the ASAR image. The resulting $F$-scores are given in Table 7. The results show that the proposed method performed comparatively better than the FM. It should be noted that the proposed method predicted this flooding state three hours ahead. This is an encouraging outcome and provides a strong case for the suitability of the proposed data-driven modelling framework.

Flood extent maps are also compared in Figure 6. The figure shows a significant match between the extents derived from all three sources (i.e. FM, data-driven and ASAR). Probabilistic and deterministic maps for Case 1 from two other flooding phases, i.e. flood growing and receding periods, are illustrated in Figure 7. During the training of the MLP models, water depth values > 0.09m were set to 1s. Therefore, the probabilistic map was expected to have high probability values where the water depth values were higher and where flooding was experienced for longer duration. Figure 7A and 7B illustrate the probabilistic and deterministic conditions of the event at 52$^{nd}$ hour. And as expected, the probabilistic map shows very low probability values in the regions where the deterministic map shows no or insignificant water depths. On the other hand, probabilities are higher in the regions where the water depth values are large. However, a slight overestimation can be noticed in the North-West part of the domain. During the flood recession period (323$^{rd}$ hour), fragmented water cells are observed in both maps (7C and D). The dominant regions are clearly visible, and a greater threshold value (> 0.6) can be applied to the probabilistic map to delineate inundated cells.
5 Discussion

5.1 Upscaling spatial dimensionality
The proposed framework was developed and tested for fluvial flooding conditions. The spatial domain of the study area was relatively small. In the future, similar methods could be developed for larger spatial domains. For a large domain, it may not be feasible to train a separate model for each GCP and a global classifier approach should be adopted. Although an ANN can predict multiple outputs, due to computer memory limitation it is also not possible to make predictions for an entire domain containing millions of cells. Therefore, a balance between computer memory and number of GCPs is required. The predicted values at the GCPs can then be simply interpolated using a higher resolution DEM. In this study, probabilistic inundation maps were produced; however, deterministic depth maps can be generated. To do so, one can skip the binarizing steps in the development of the hydraulic classifier database and change the sigmoid activation function in the output layer.

5.2 Training data
The performance of the ML-based models depends on the quality and quantity of the training data. It is an integral part of developing data-driven models to make sure the training data samples are sufficient and encompass all possible hydrological scenarios. In this study the classifiers were trained using four synthetic hydrographs. The number of hydrographs can be increased to provide more training samples. For large domains, topographical features (e.g., land cover data), rainfall etc. can also be included in the training data set.

5.3 Managing uncertainty
The uncertainty in the outputs can severely limit the application of such a data-driven modelling framework. This is because every step of this modelling chain, from the input of the 2D hydrodynamic model simulation to the output of the RK method, leads to a cascade of uncertainty. While it is not possible to eliminate uncertainties completely, they can be diminished and quantified. To reduce model uncertainty, it is essential to generate training samples from a fully calibrated 2D model and to optimise the hyperparameters of the ML-based models. However, a comprehensive quantification of propagated uncertainty was not part of this study. In this study, quantification of uncertainty was limited to flood arrival time. For a more comprehensive uncertainty quantification an ensemble approach can be devised in the future studies.

5.4 Key limitations
One of the key limitations of this modelling framework is that the output does not provide depth and velocity information. Some concurrent studies showed that it is possible to apply ML-based models for depth estimation (e.g. Bermúdez et al., 2019, Liu and Pender, 2015), therefore, in future studies, depth estimation model can be embedded to the system. However, developing a model to estimate velocity can be challenging and so far, recent studies reported that the ML-based models do not perform well in predicting velocities (Bermúdez et al., 2019).
In addition, the current approach is only tested for fluvial flooding. Further studies are required to develop and test the capacity of data-driven models for pluvial flooding.
5.5 Compute time

The primary objective of this paper was to implement a data-driven end-to-end inundation mapping system which can be used for operational forecasting. Specific to this study, the proposed method reduces computational time from minutes to seconds. The FM software takes about 40 minutes to generate raster files containing water depths. Note that in this study velocity files were not generated. The model simulation time would be significantly longer if velocity values were also produced. In contrast, once the training samples are prepared the training of RF-based models can take only a few seconds to complete. The training of MLP-based models, however, depends on the selected approach. Naturally, training a separate model for each of the GCPs (local MLP) will take longer than training a global MLP. Both local and global approaches were tested in this study. The global model took ~9 seconds to complete the training process while local approach took ~25 minutes. Once the models are trained, it takes about ~10 seconds to sequentially generate probabilistic inundation maps. An Intel Core i5- 250 GHz CPU with 8 GB RAM system was used to execute both the FM and data-driven model runs.

6 Conclusion

This paper presented the development of a new modelling framework based on ML techniques for operational flood forecasting and inundation visualisation. The concept of deriving an ML-based hybrid network to forecast and produce multi-step ahead sequential probabilistic inundation maps was formulated and tested. The test results obtained for the proposed model from the Upton case study reveal promising outcomes in terms of forecasting inundated areas in real-time. A key finding of this research is that the classifiers can be trained on samples generated from coarse grid 2D hydrodynamic model runs and then be applied to high-resolution DEM for dynamic mapping of flood inundation extents. This study has also shown that the models can be trained using synthetic datasets if there are not enough observed data available.

Major advantages of the proposed modelling framework are that it provides greater computational efficiency and does not require large storage capacity to store fitted ML models. The training process is relatively simple, and the models can be re-trained in case of any significant changes in the topography of the floodplain. In addition, it requires very little expertise to run the trained models. One of the biggest advantages of the ML based models is that they can be scaled and easily modified to add new components. The proposed framework for real-time flood inundation can be further improved by plugging auto-updating features for refining the forecasted maps. For example, fusing live social media information utility can be used to update flood maps.

The quantification of model uncertainty was limited to arrival time, the next stage of development of the proposed real-time inundation modelling system could be a systematic uncertainty quantification. Further research is required to improve forecasting and visualization capabilities of the model (e.g. training on data generated from a calibrated 2D model using observed inflow hydrographs instead of synthetic flow series at regional scale, developing flood inundation warning systems).
Acknowledgement

We would like to thank the Institute for Infrastructure and Environment, Heriot-Watt University for providing us with the flood hydrograph and Dr. Sylvain Néelz for supporting us by providing the initial ISIS-1D Upton-Upon-Severn river network.

List of notations

\( N \) is the number of samples
\( \hat{y}_t \) is the predicted value
\( \bar{y} \) is the mean of observed data
\( q_t \) is the channel flow at time \( t \)
\( \bar{R}_t \) is the catchment average rainfall at time \( t \)
References


Table 1: Model evaluation metrics used in this study.

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Application</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1-score</td>
<td>Ability of the classifier to detect state changes (dry to wet/ wet to dry) of a cell. It compares how well the models correctly classified the GCPs during the event.</td>
<td><em>F1-score</em> which is the harmonic mean of Recall and Pr:</td>
</tr>
<tr>
<td>Mean, 95%</td>
<td>Mean error in predicting flood arrival time and 95% confidence limits.</td>
<td>Descriptive statistics of the residuals to show the classifiers error range.</td>
</tr>
<tr>
<td>F-score</td>
<td>Measures how many GCPs are correctly classified at a particular time step. It shows the percentage of GCPs, whose true state is correctly predicted by the model for every hour and throughout the flood event. GCPs that never get flooded are excluded to reduce accuracy bias in F-score calculation.</td>
<td>The method is described in Aronica et al. (2002)</td>
</tr>
<tr>
<td>Overall accuracy</td>
<td>Describes how accurately GCPs are classified during the flood event. It includes all the GCPs.</td>
<td>Category A: is the total number of the cells correctly classified as dry by both the 2D hydrodynamic model and ML algorithms; Category B: is the number of cells misclassified as dry by the ML algorithms (false negative); Category C: is the total number of cells misclassified as wet by the ML algorithms (false positive); and Category D: is the total number of cells correctly classified as wet by both models. Calculated from a confusion matrix.</td>
</tr>
<tr>
<td>Coefficient of determination (R-squared)</td>
<td>Quantifies the variation of the predicted values defined by the different input values</td>
<td></td>
</tr>
<tr>
<td>Root Mean Squared Error (RMSE)</td>
<td>Measures the spread of residuals.</td>
<td></td>
</tr>
<tr>
<td>Nash- Sutcliffe Efficiency (NSE)</td>
<td>Measures the goodness of fit between observed and modelled data.</td>
<td></td>
</tr>
</tbody>
</table>
Table 2: Summary of hydrographs used in the study.

<table>
<thead>
<tr>
<th>Index</th>
<th>Used for</th>
<th>Peak [m³/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syn. Hydrograph 1</td>
<td>Training MLP</td>
<td>1000</td>
</tr>
<tr>
<td>Syn. Hydrograph 2</td>
<td>Training MLP</td>
<td>802.1</td>
</tr>
<tr>
<td>Syn. Hydrograph 3</td>
<td>Training MLP</td>
<td>1150</td>
</tr>
<tr>
<td>Syn. Hydrograph 4</td>
<td>Training MLP</td>
<td>831</td>
</tr>
<tr>
<td>Subset 1</td>
<td>(1) Train and test RF for flow forecasting</td>
<td>899.2</td>
</tr>
<tr>
<td></td>
<td>(2) Test MLP-based classifiers</td>
<td></td>
</tr>
<tr>
<td>Subset 2</td>
<td>(1) Train and test RF for flow forecasting</td>
<td>832.5</td>
</tr>
<tr>
<td></td>
<td>(2) Test MLP-based classifiers</td>
<td></td>
</tr>
</tbody>
</table>

Table 3: MLP parameters used in the training process.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values and functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of total training input-output instances</td>
<td>1219</td>
</tr>
<tr>
<td>No. of input variables</td>
<td>7 (flow hydrographs with their lagged values, i.e. 2 [upstream (MC010) and MC033], and corresponding time, T)</td>
</tr>
<tr>
<td>No. of nodes in input layer</td>
<td>8 (7 for input variables and 1 bias node)</td>
</tr>
<tr>
<td>No. of hidden layer</td>
<td>1</td>
</tr>
<tr>
<td>No. of nodes in output layer</td>
<td>1</td>
</tr>
<tr>
<td>No. of nodes in hidden layer</td>
<td>10</td>
</tr>
<tr>
<td>Activation functions</td>
<td>ReLU function for the input and hidden layer, sigmoid function for the output layer.</td>
</tr>
<tr>
<td>Loss function</td>
<td>binary_crossentropy</td>
</tr>
<tr>
<td>Optimizer algorithm</td>
<td>Adam</td>
</tr>
<tr>
<td>No. training iterations</td>
<td>100</td>
</tr>
<tr>
<td>Batch size</td>
<td>20</td>
</tr>
</tbody>
</table>

Table 4: Error statistics of RF-based models in forecasting 3h ahead flows.

<table>
<thead>
<tr>
<th>Section ID</th>
<th>Method</th>
<th>Case 1</th>
<th>Case 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R-squared</td>
<td>RMSE</td>
<td>NSE</td>
</tr>
<tr>
<td>MC010</td>
<td>0.997</td>
<td>9.659</td>
<td>0.997</td>
</tr>
<tr>
<td>MC033</td>
<td>0.997</td>
<td>11.115</td>
<td>0.996</td>
</tr>
</tbody>
</table>
Table 5: Results of the integrated ML model when compared against reference FM outputs.

<table>
<thead>
<tr>
<th>Study case</th>
<th>Mean</th>
<th>Median</th>
<th>Min</th>
<th>Max</th>
<th>1&lt;sup&gt;st&lt;/sup&gt; Quartile</th>
<th>3&lt;sup&gt;rd&lt;/sup&gt; Quartile</th>
<th>95% CI</th>
<th>Classification accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>1h51m</td>
<td>2h</td>
<td>-2h</td>
<td>7h</td>
<td>0h</td>
<td>4h</td>
<td>[-2h, 6h]</td>
<td>0.989</td>
</tr>
<tr>
<td>Case 2</td>
<td>1h55m</td>
<td>2h</td>
<td>-2h</td>
<td>7h</td>
<td>0h</td>
<td>4h</td>
<td>[-2h, 6h27m]</td>
<td>0.985</td>
</tr>
</tbody>
</table>

Table 6: Average F-scores defining classification accuracies at the different temporal segments.

<table>
<thead>
<tr>
<th>Time step</th>
<th>Case 1</th>
<th>Case 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1&lt;sup&gt;st&lt;/sup&gt; 32 hours</td>
<td>No flooded cells</td>
<td>No flooded cells</td>
</tr>
<tr>
<td>Next 48 hours</td>
<td>90.9%</td>
<td>90.0%</td>
</tr>
<tr>
<td>Next 48 hours</td>
<td>99.3%</td>
<td>99.2%</td>
</tr>
<tr>
<td>Next 48 hours</td>
<td>99.6%</td>
<td>97.5%</td>
</tr>
<tr>
<td>Next 48 hours</td>
<td>98.5%</td>
<td>98.9%</td>
</tr>
<tr>
<td>Next 48 hours</td>
<td>98.8%</td>
<td>97.1%</td>
</tr>
<tr>
<td>Last 52 hours</td>
<td>97.8%</td>
<td>98.5%</td>
</tr>
<tr>
<td>Avg.</td>
<td>98.82%</td>
<td>98.20%</td>
</tr>
</tbody>
</table>

Table 7: Yielded classification accuracies when FM and proposed data-driven method are compared against ASAR image.

<table>
<thead>
<tr>
<th>Model</th>
<th>F-score (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data-driven method</td>
<td>88.22%</td>
</tr>
<tr>
<td>FM</td>
<td>86.58%</td>
</tr>
</tbody>
</table>
Figure Captions

Figure 1. Framework for the data-driven real-time flood inundation forecasting system.

Figure 2. Study domain and ground control points.

Figure 3. (A) Observed inflow hydrograph for total duration including rainfall intensity (reverse x-axes), (B) subset 1 and (C) subset 2.

Figure 4. Synthetic hydrographs used to train the MLP-based classifiers.

Figure 5. (A) Probabilistic flood inundation map forecasted at 122\textsuperscript{nd} hour using the proposed data-driven modelling framework, and (B) Deterministic flood map generated by the FM software.

Figure 6. Flood extent maps derived from the proposed data-driven modelling approach, the FM and the ASAR image captured on 8\textsuperscript{th} November 2000.

Figure 7. (A) Probabilistic flood inundation map forecasted at 52\textsuperscript{nd} hour using the proposed data-driven modelling framework, (B) Deterministic flood map generated by the FM software at 52\textsuperscript{nd} hour (C) Probabilistic flood inundation map forecasted at 323\textsuperscript{rd} hour using the proposed data-driven modelling framework, and (D) Deterministic flood map generated by the FM software at 323\textsuperscript{rd} hour.
Accepted manuscript doi: 10.1680/jwama.20.00002

Figure 1
Figure 3
Figure 4
Figure 7