



## Research papers

## Prediction models for urban flood evolution for satellite remote sensing

Roderick Lammers<sup>a,1,\*</sup>, Alan Li<sup>b</sup>, Sreeja Nag<sup>b</sup>, Vinay Ravindra<sup>b</sup><sup>a</sup> College of Engineering, University of Georgia, 712 Boyd Grad Studies Bldg 200 DW Brooks Dr, Athens, GA 30602, United States<sup>b</sup> Bay Area Environmental Research Institute, NASA Ames Research Center, Moffett Field, CA 94035, United States

## ARTICLE INFO

This manuscript was handled by Emmanouil Anagnostou, Editor-in-Chief, with the assistance of Qiang Dai, Associate Editor

**Keywords:**  
WRF-Hydro  
LSTM  
Urban flooding  
Adaptive sensing  
Remote sensing

## ABSTRACT

Accurate and timely flood forecasts are critical for protecting people and infrastructure in a changing climate. Satellite remote sensing provides the necessary wide area coverage and period revisits to measure episodic heavy precipitation and resultant urban floods. We propose two methods to assimilate satellite-observed precipitation into hydrologic models in real time to update flood forecasts, bypassing two previous barriers in this technology: infrequent satellite overpasses and long model run times. Constellations of small satellites overcomes the first barrier by providing frequent flights over an area of interest; however, these constellations require coordination and planning to capture precipitation data where it is most needed to inform flood forecasts. The primary purpose of this paper is to address the second barrier – that is high computational costs that make it infeasible to run flood forecast models on-board these satellites so that they can re-orient to measure where most needed. We develop a simple regression-based approach and a machine learning framework (Long Short-Term Memory (LSTM) models) to provide reliable flood forecasts using satellite-observed precipitation at a fraction of the computational cost of physics-based hydrologic models. We apply these approaches to a test case for the Atlanta metropolitan region using the Weather Research and Forecasting model hydrologic modeling system (WRF-Hydro) to simulate flooding across the model domain for several precipitation events. We compare the accuracy of our proposed approaches to the WRF-Hydro model using different spatial extents and temporal frequencies of precipitation observations to examine different plausible satellite constellation scenarios. The LSTM approach trades performance accuracy and adaptability for computational efficiency, which can be important in a time and resource constrained scenario. The LSTM model reduces total error up to 38% from an initial flood forecast. Additionally, this approach correctly classified flooding to within one flood magnitude category in ~90% of cases. These new forecasting algorithms can be used onboard constellations of small satellites to observe ongoing flood events, update short term predictions, and schedule observations to maximize useful measurements and thereby improve flood warning systems for protecting residents and properties.

## 1. Introduction

Remotely sensed observations from satellites play a critical role in today's disaster forecast systems. The observations made by satellites along with those made by terrestrial sensors are assimilated into global and regional Numerical Weather Prediction (NWP) models to provide improved forecasts and give actionable intelligence to the local authorities.

Satellites and their corresponding instruments are expensive resources. For example, the Soil Moisture Active Passive (SMAP) mission ([NASA, 2020](#)) which consists of a single satellite with a radar and radiometer instrument (radar unit currently non-functional), costs

nearly 1 billion USD and takes 3 days to map the entire world. Efforts are underway to develop cheaper observational platforms consisting of several smaller satellites in a distributed layout, where constellations are the most common geometric configuration. Examples of hydrology-relevant constellation missions are the CYGNSS ([Ruf et al., 2018](#)) (in orbit), TEMPEST ([Reising et al., 2018](#)) (in development), TROPICS ([Blackwell et al., 2018](#)) (in development), and RainCube ([Peral et al., 2015](#)) (in orbit). The schedule of locations and times for the observations to be made by a satellite is calculated by ground computers, with human input, and is uploaded to the satellites via ground stations during intermittent overpasses.

Currently, many satellites can obtain infrequent data and coverage of

\* Corresponding author.

E-mail address: [rod.lammers@cmich.edu](mailto:rod.lammers@cmich.edu) (R. Lammers).

<sup>1</sup> Present address: School of Engineering & Technology, Central Michigan University, ET Building 100, Mt. Pleasant, MI 48859, United States.

specific areas. For example, the GEOS-5 model uses amalgamated data from a variety of sources, such as NOAA's polar-orbiting weather satellites, as well as the Aqua and SSM/I instruments which have a spatial resolution of upwards of tens to hundreds of km, depending on the backscatter frequency (Rienecker et al., 2008). The satellites that carry these instruments are multi-generational governmental assets, and thus are few in number, allowing them to take only precise, localized observations due to orbital constraints. With the advent of satellite swarms and constellations, it has become possible to observe different regions and to gather data globally at larger spatial extents, while sacrificing some of the precision afforded by highly calibrated instrumentation. Although the latter characteristic lends itself poorly to detailed modeling, the highly parallel nature of distributed satellite operations allows for quick reaction and agile maneuvering that were previously not possible.

Due to the dynamic nature of satellite and ground point access opportunities, observations made by satellites are expected to be sparse even as constellations grow. Given this sparse opportunity and cost of resources to make observations (power, data limits), it is always more efficient to capture data which has been predicted to be useful rather than to first capture an observation and later determine its utility (Ravindra et al., 2020). This motivates the concept of adaptive sensing (Nag et al., 2020), as illustrated in Fig. 1, wherein past observations are processed and information extracted from them is used to forecast the utility of future observations, which then informs the observation planner – all onboard the satellite without humans or ground stations in the loop. While this basic framework is relevant to remote observation of a variety of phenomena (e.g., tropical storms, forest fires, etc.), this paper focuses on transient precipitation and urban floods.

Generally, adaptive sensing requires a simulator (of the phenomenon of interest) which can rapidly perform data assimilation and forecasting, so that the observation plan of the satellites can be updated with minimal latency. Physics-based NWP hydrological models such as the Weather Research and Forecasting hydrologic modeling system (WRF-Hydro) provide (state of art) accurate forecasts, but in turn require larger computing resources than those available onboard satellites. The most rapidly adaptive set-up would be a simulator onboard the satellite which can assimilate the previous observations on the fly and produce updated forecasts which dictate the next set of observations to be made. To enable adaptive sensing by future satellites, onboard autonomy is required in processing instrumental data, in predicting future observables, and in planning and implementing observation schedules to maximize science utility. Such a sense-plan-act feedback loop would run in real time (i.e., make the satellite completely autonomous), and is thus

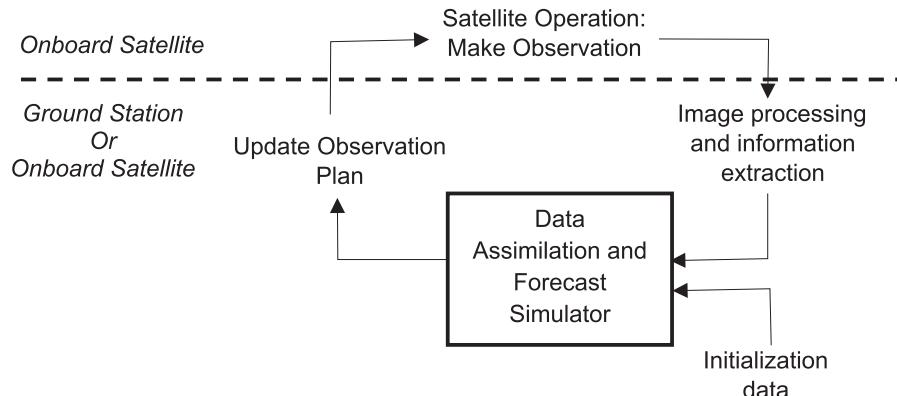
constrained by limited satellite resources as well as physical processes, such as orbital dynamics (Nag et al., 2019). It is also possible to run the sense-plan-act in conjunction with the ground station, wherein the satellite observations would be downlinked to the ground station, which can process them, extract information, assimilate the data, issue forecasts, and update the observation plan for the satellite. The new plan would be uplinked to the satellite upon the next overpass. While moving the bulk of the computation to ground computers reduces the need for onboard resources and still keeps the human out of the loop, this sense-plan-act loop has much longer delay times, due to the sparsity of satellite-ground station contacts.

This paper focuses on the forecast module of the adaptive sensing loop (the box in Fig. 1). For the presented application, satellites make radar-based precipitation measurements as they fly over urban regions and use these measurements as direct input to predict flood levels in small to medium-sized streams, which are otherwise not directly observable from space. Onboard planners can then use these flood predictions to schedule and focus satellites' observations upon areas of higher urgency (e.g., more flooding) and save the opportunity cost of observing locations with negligible flood activity. Previously, large spacecraft (e.g. EO-1) have been tasked to observe large floods (e.g., in Thailand) using ground-based infrastructure for processing, assimilation and scheduling (Chien et al., 2019). Recently, improvements in onboard autonomy on smaller satellites (Doubleday et al., 2015) and full body re-orientation agility on larger satellites have created the potential to run the scheduler on the satellites themselves followed by re-orientation of the instruments or entire satellite to track scheduled ground points (Nag et al., 2018). While ground support will always remain as a backup for planning onboard operations, the limited contact frequency is better utilized in communicating high-level goals and data a few times a day, while lower-level decisions in observational planning can be migrated to the satellites' discretion over time. The benefits further amplify when multiple satellites can coordinate their observations by communicating inferences and plans (Nag et al., 2020b), thus circumventing the lengthy delay of communicating with ground-based relays while observing fast evolving or transient phenomena (Linnabary et al., 2019) such as urban floods.

This paper has two objectives:

Propose and investigate alternative flood simulation models that are less computationally expensive than a standard physics-based model (WRF-Hydro), such that these models can run on satellites as part of the onboard predictor and scheduler that enable adaptive and responsive remote sensing

Compare the prediction accuracy of these alternative flood models to



**Fig. 1.** Adaptive sensing concept for application of flood monitoring from satellites. The planned observations to be made by the satellite are continually updated with new forecasts of the phenomenon (flood) triggered by previously made observations. This paper targets the block called "Data Assimilation and Forecast Simulator". The lower half of the figure may be run onboard (rapid autonomy) or on the ground using downlinked data followed by uplinked plans (slower autonomy).

WRF-Hydro for a range of satellite observation frequencies and spatial coverages, with the goal to show that loss in accuracy is within acceptable levels that justify the benefits of near real-time planning and execution.

The document is organized as follows. In [Section 2](#), we provide background on flood forecasting and machine learning techniques. In [Section 3](#), we discuss the test case application to modeling riverine flooding in the Atlanta metropolitan region, including the development of two computationally efficient flood prediction approaches. [Section 4](#) describes the results of this modeling and [Section 5](#) discusses the significance of these findings and suggests areas for improvement.

## 2. Background

### 2.1. Flood forecasting

Flooding is one of the most common and expensive natural disasters worldwide, and flood risks are increasing due to climate change, development, and population growth ([Andersen and Shepherd, 2013](#)). Traditionally, flood monitoring is achieved using networks of real-time stream gaging stations, such as the network operated by the U.S. Geological Survey. These data are also used in conjunction with hydrologic models to forecast river discharge and flood risk, at a variety of temporal scales (i.e. hours, days, or weeks). These hydrologic models require recent and forecasted precipitation as an input – data that may come from a variety of sensors and weather models. Observed precipitation data are typically obtained from gauges, radar, or satellite sensors ([Li et al., 2016](#)). Regardless of the source, these data can be used to update flood forecast models over time and improve accuracy.

Flood forecasting using coupled atmospheric-hydrologic-hydraulic modeling is maturing and can be applied at a variety of scales. For example, the Global Flood Monitoring System uses satellite-based precipitation and a hydrologic model to simulate river flooding for most of the globe ([Wu et al., 2014](#)) at a coarse spatial (12 km) and temporal scale (3 h) that may miss short flood events on small and medium sized streams. In the U.S., the National Weather Service continuously updates streamflow forecasts at millions of points throughout the country ([Salas et al., 2018](#)), while another system uses radar-derived precipitation data to simulate flash floods ([Gourley et al., 2017](#)). These tools tend to be complex and data-intensive, requiring significant resources to run. Of particular concern is the availability of reliable precipitation data to generate initial flood forecasts and to update those forecasts as additional precipitation data become available. Satellite derived precipitation estimates can be useful for this purpose and are especially important for developing countries without the resources to develop extensive ground-based sensor networks to monitor both precipitation and river flooding ([Rahman and Di, 2017](#)).

### 2.2. Neural networks

In the last decade, neural networks have been at the forefront of machine learning (ML) and artificial intelligence research. Their successful application in a wide range of problems from image classification ([Krizhevsky et al., 2012](#)) to self-enabled reinforcement learning ([Mnih et al., 2013](#)) have paved the way for widespread adoption of these methods, and have started to influence data-driven methods within Earth science as well ([Chen et al., 2016; Li et al., 2020](#)). We will provide here an overview of artificial neural networks, particularly long short-term memory (LSTM) models, which will form the basis of our ML methodology.

Neural networks are loosely modelled after the structure of the human brain by mimicking layers of cascading neurons transporting electrical signals. Neurons that activate together tend increasingly to do so over time, which gives rise to complex capabilities such as pattern recognition and perception. The basic equation for each “neuron” within a neural network is described as:

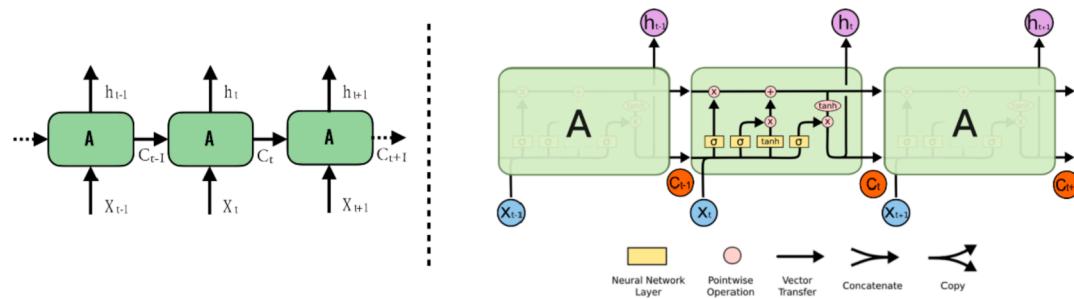
$$z = f(W^*x + b) \quad (1)$$

where  $z$  is the output,  $W$  the weights to be learned,  $x$  the input, and  $b$  the bias, with the  $*$  symbol representing multiplication (although this can be extended to the Hadamard product or convolution). Generally,  $W$  can also represent a tensor, with a set number of filters  $k$ , such that if  $x$  is a vector of length  $n$  and  $W$  is of size  $k \times n$ , the resulting  $z$  will be of length  $k$ . The nonlinear activation function  $f$  relates the importance of the linear output, and although historically the sigmoid function has been utilized to bound the output between 0 and 1, the Rectified Linear Unit (ReLU) is more widely adopted due to its numerical and backpropagation properties ([Nair and Hinton, 2010](#)). Often, these “neurons” are stacked in layers such that the output of one layer acts as the input to the next layer.

Of particular interest is the recurrent neural network (RNN) and its variant the LSTM, depicted in [Fig. 2](#). The structure of the RNN works well for sequential data, and one of the more successful applications of this was in language translation tasks ([Wu et al., 2016](#)). At its core, the RNN allows for chain-like structures consisting of identical recurring cell blocks  $A$ , which takes an input  $X$  and outputs a result  $h$ . However, an additional output  $C$  (known as the cell state) is also generated, which is passed into the next recurring block and is meant to retain some of the “memory” that lingers from previous cell blocks. In this way, certain features from previous states that might be important are encoded within the cell state, which may affect later outputs.

The LSTM is an improvement on the original RNN ([Hochreiter and Schmidhuber, 1997](#)), by incorporating additional interactions per cell block. Namely, we include the sigmoid and hyperbolic tangent functions to the equations such that we can manipulate the importance of the input, allow for previous cell states to expire or be forgotten, add to the existing cell states, and to generate an output relevant to the current input ([Olah, 2015](#)). Further variants of the LSTM have been developed, such as the gated recurrent unit (GRU), which increases the complexity of each cell ([Chung et al., 2015](#)). However, the adaptation of this network structure can be readily obvious: given the inputs  $X$  (precipitation and static physical characteristics), we can generate flood outputs  $h$  per time step, while keeping the internal weights and biases of each cell state identical across all time steps. Furthermore, if we can generalize the model of the river flow encapsulated and described by cell  $A$ , then we can apply the LSTM across all river flows. Furthermore, concepts such as water retention and delayed flows can be represented by the cell state  $C$ . While not perfect, this model is much simpler than physics-based hydrological models and thus has the potential to be ported onto satellite systems which are resource constrained.

The idea of utilizing RNNs and LSTMs for modelling hydrological processes is not a new one. A variety of machine learning approaches have been successfully applied to forecast river flows ([Yaseen et al., 2019](#)), including LSTM models ([Kratzert et al., 2018](#)). However, in many cases, LSTMs have been used to forecast flows in larger rivers over long time scales (e.g. daily or monthly flows) ([Kao et al., 2020; Le et al., 2019](#)). One study applied LSTMs to short, flash flood events, through a large, mountainous watershed ([Song et al., 2020](#)), while others have incorporated LSTMs into an urban flood warning system ([Lee et al., 2020](#)). LSTMs have also been used to improve rainfall forecasts which were then fed into a hydrologic model for flood prediction ([Nguyen and Bae, 2020](#)). In our work, we applied LSTMs to forecast sub-hourly streamflow in an urban setting using a bare-bones approach requiring minimal computational complexity. It is our goal to demonstrate the validity and potential for fast and reliable onboard prediction of flash flooding events on satellites without ground-based communication, allowing for agile maneuvering of satellite resources in a time-sensitive environment.



**Fig. 2.** (Left) Recurrent neural network structure; (Right) Internal structure of a long short-term memory network. Adapted from (Olah, 2015). Recurring cell blocks (A) takes inputs (X) and outputs a result (h). Cell states (C) are also generated and passed to subsequent cell blocks.

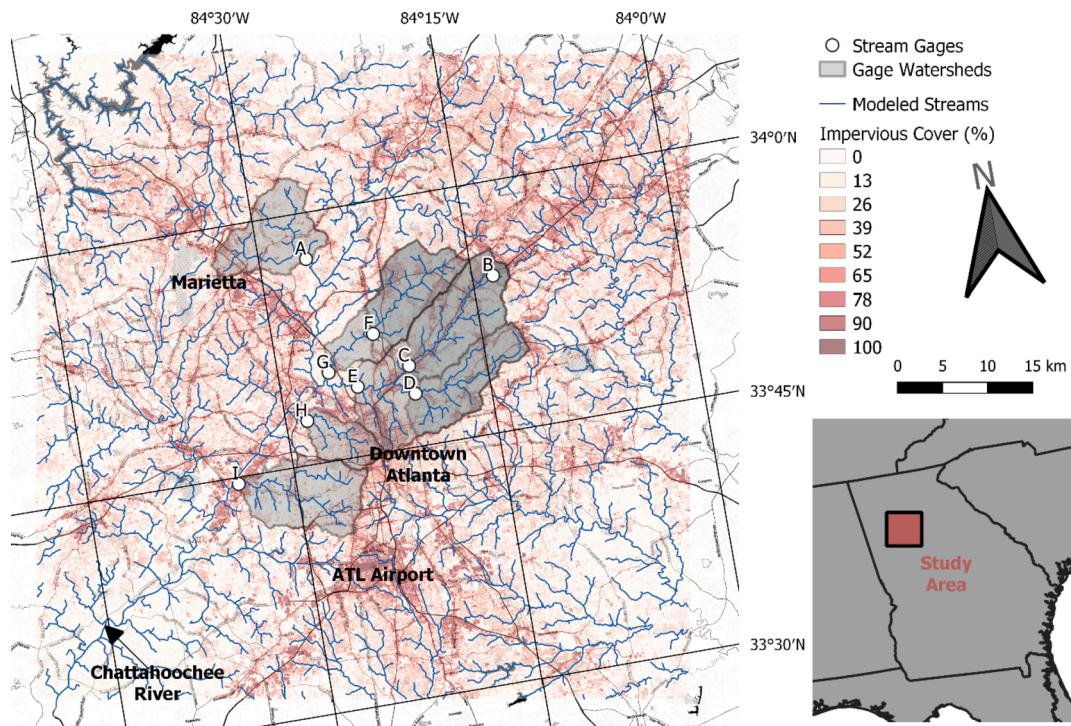
### 3. Methodology

This section proposes methods for data assimilation and forecasting to inform the satellite scheduler's future plans. The models are built to enable adaptive sensing such that the location of future satellite observations can be determined as a function of past observations. Furthermore, these models can either be run in a centralized ground-based location with the results uplinked to the constellations, or run on every satellite in the constellation instead. The spatial resolution is a function of the geographic extent of locations of interest and the instrument field of view (e.g. rain radar). The temporal resolution or re-run cadence is a function of the size of the constellation since more satellites allow more measurements, and therefore more updates. Our proposed methods are generalizable to many natural phenomena where satellite observations of ongoing events can be incorporated into forecast models (e.g., wildfires or hurricanes). The subsections below use representative examples of a constellation adaptively sensing a flood event in a single city. They are also easily extendable to multiple floods in various locations around the globe.

#### 3.1. Observing system simulations

The goal of this paper is to report the accuracy of forecast algorithms that can run near real time onboard satellites. Specifically, we are looking to predict river flooding as a function of observed transient precipitation over a specific city over a predefined duration of time, wherein parts of the city can be sporadically observed by satellite overpasses. To decouple satellite dynamics from the predictor problem, the proposed flood forecasting models are applied for custom, variable samples in space and time. From a satellite's perspective, all regions of interests across the globe can be spatially discretized into grid cells of any spatial resolution. Any of these grid cells can be observed any time a satellite overpasses. Onboard autonomy models can then be "updated" with new predictions for every set of observations. While the results presented in this paper are unique to our chosen test case, the basic sampling approach is generalizable across any urban region with small streams, and can be adapted to any combination of sampling frequency and spatial extent, i.e. any constellation topology.

We chose the Atlanta metropolitan area as our test case. We modeled riverine flooding using the WRF-Hydro hydrologic model version 5 (Gochis et al., 2018) (Fig. 3). WRF-Hydro is a distributed hydrologic model that links a land surface model (Noah-MP) (Niu et al., 2011) with



**Fig. 3.** Map of the study area including percent impervious cover for the model domain (NLCD 2016 data), stream network, and the nine USGS gages used for model calibration and validation.

terrain and channel flow routing modules to simulate streamflow. While WRF-Hydro can be coupled to the Weather Research and Forecasting (WRF) weather model (Skamarock et al., 2008), we ran WRF-Hydro in uncoupled mode, instead supplying it with meteorological forcing data (primarily precipitation) from the North American Land Data Assimilation System (NLDAS) (hourly time step,  $4 \times 4$  km spatial resolution). The model was run for a  $72 \times 72$  km area, with a grid resolution of 900 m for the land surface model and 30 m for the terrain and channel routing. While this grid resolution is relatively coarse, especially for urban areas, we were in part limited by the resolution of model inputs (e.g., digital elevation model, land cover). Furthermore, terrain and channel routing resolutions as coarse as 250 m have been shown to be adequate for modeling flooding in urban areas (Kim et al., 2021). We incorporated both surface and subsurface flow routing and used the Muskingum-Cunge (Cunge, 1969; Gochis et al., 2018) reach-based channel routing option. Model streamflow was output at 15-minute time steps. This temporal resolution was chosen because a satellite overpass for any ground target can be as frequent as every 15 min for a constellation with dozens of satellites in Low Earth Orbit.

Most major model inputs (e.g., soil parameters, land use, and land surface model variables) were obtained from the WRF Preprocessing System (WPS) data download website (NCAR, 2020). Hydro-enforced digital elevation models (DEM) representing local topography were obtained from the USGS National Hydrography Dataset v2 (NHD v2). We manually edited some of this DEM to remove bridges and allow accurate flow routing. WRF-Hydro was calibrated by trial and error by adjusting several static landscape and soil variables (Table 1). All parameters except *refkdt* were simply scaled higher or lower from their starting value (a function of land use or soil type). *Refkdt* was calibrated in a spatially explicit way based on average percent impervious cover for each grid cell estimated from 2016 National Land Cover Database (NLCD) data (Yang et al., 2018). Areas with higher imperviousness were given lower *refkdt* values (less infiltration). We therefore account for some of the high spatial variability in land cover present in urban areas.

We use WRF-Hydro's results to train computationally efficient models that can run onboard satellites, introduced in the next section. This paper does not seek to verify the accuracy of or improve WRF-Hydro, and our approach would hold if we replaced WRF-Hydro with a different physics-based model with better forecast accuracy because our simpler forecasting methods could be trained on that instead. The purpose of the WRF-Hydro model was to provide a test case for which we could develop and apply these alternative, simpler flood forecasting approaches. The goal was not to perfectly simulate a specific flood event, but instead to capture physically realistic spatial and temporal patterns of flooding in an urban area. We could use only observed river discharge data for this analysis; however, these data are available only at a few discrete monitoring sites. Modeling, on the other hand, provides simulated river discharge at all points along the entire stream network. Still, the WRF-Hydro model does not account for many hydrologic complexities in urban areas – such as stormwater pipes, reservoirs, or other infrastructure. Despite this simplification, the model results provide a

**Table 1**

WRF-Hydro parameters included in the manual calibration. Ranges of final calibrated values are shown for spatially-variable parameters. Subsurface flow slope was the same value across the whole model domain.

WRF-Hydro Parameter	Description	Calibrated Value
<i>OV_ROUGH2D</i>	Overland flow roughness coefficient [unitless]	0.02–0.8
<i>dksat</i>	Saturated hydraulic conductivity [m/s]	1.3e-7–5.3e-7
<i>bexp</i>	Pore size distribution index [unitless]	2.8–3.5
<i>refkdt</i>	Surface runoff parameter [unitless]	0.1–4.0
<i>LKSAT</i>	Lateral hydraulic conductivity [m/s]	0.3–3.4e-8
<i>subsurface flow slope</i>	Parameter controlling percolation versus lateral subsurface flow [unitless]	0.1

useful and physically realistic flood case (based on model calibration and verification) that enables us to test our proposed flood forecasting methods.

WRF-Hydro was calibrated on a single storm event during April 5–6, 2017. Modeled streamflow was compared to hydrographs from nine U.S. Geologic Survey (USGS) gages in the Atlanta metro area (Table S1). Details on USGS stream gaging methods can be found in (Carter and Davidian, 1968). The calibrated model was then validated on a separate storm event on June 20–21, 2017. For both model calibration and validation, a roughly 4 month model spin up period was used.

Model performance was assessed using two metrics: Nash-Sutcliffe Efficiency (NSE) (Nash and Sutcliffe, 1970) and percent bias (PBIAS). NSE is a common measure of hydrologic model performance and ranges from  $-\infty$  (worst) to one (best):

$$NSE = 1 - \frac{\sum_{t=1}^T (Q_0^t - Q_m^t)^2}{\sum_{t=1}^T (Q_0^t - \bar{Q}_0)^2} \quad (2)$$

where  $\bar{Q}_0$  is the mean of observed discharges,  $Q_m^t$  is the modeled discharge at time t, and  $Q_0^t$  is the observed discharge at time t. Percent bias ranges from  $-\infty$  to  $\infty$ , with negative values indicating under-prediction of discharge, positive values indicating over-prediction of discharge:

$$PBIAS = \frac{\sum_{t=1}^T (Q_m^t - Q_0^t)}{\sum_{t=1}^T Q_0^t} * 100 \quad (3)$$

The calibrated model results were also used to develop “flood values” to be used by the satellite scheduling optimizer. This flood value was representative of the areas where satellite-measured precipitation would be most valuable for assessing flood risk. Modeled channel discharge was scaled by an estimate of what constituted a “flood” at each location. For simplicity, we used the 2-year recurrence interval flow rate ( $Q_2$ , the flow that has a 50% chance of happening in any year) as this is a rough estimate of the flow that completely fills a river channel (Williams, 1978). For each stream reach,  $Q_2$  was estimated from the USGS regional regression equations for urban streams in the region (Feaster et al., 2014). Flood value was thus defined as:

$$flood.val = \frac{Q_m}{Q_2} \quad (4)$$

Each stream reach has a single value of  $Q_2$  (based on watershed area and percent of impervious cover). However, the modeled flow rate ( $Q_m$ ) is output every 15 min for each stream reach. Therefore, both  $Q_m$  and *flood.val* vary through time at each location. We also examined categorical flood magnitude. The National Weather Service (NWS) defines four flood stages based on impacts to surrounding infrastructure and property: action, minor, moderate, and major floods. For all active stream gages in Georgia with NWS flood stages, we converted these stages to discharges based on the published USGS gage rating curve and then calculated their corresponding *flood.val*. Using these gage-derived values, we estimated *flood.val* cutoffs that corresponded to each of these NWS flood stages. This provided a more realistic assessment of relative flood impacts. Separation of the flooding values based upon the NWS categorization of flood intensity into categories is as follows:

$$0 \leq \frac{Q_m}{Q_2} < 0.7 \rightarrow No flood$$

$$0.7 \leq \frac{Q_m}{Q_2} < 1 \rightarrow Action$$

$$1 \leq \frac{Q_m}{Q_2} < 1.6 \rightarrow Minor$$

$$\begin{aligned} 1.6 \leq \frac{Q_m}{Q_2} < 2.4 &\rightarrow \text{Moderate} \\ 2.4 \leq \frac{Q_m}{Q_2} &\rightarrow \text{Major} \end{aligned} \quad (5)$$

We used the output from the WRF-Hydro model to develop two simplified models to update flood predictions with the satellite-observed precipitation data. The first is a simple regression-based approach and the second uses a machine learning framework. Results from both methods were compared to the calibrated results of WRF-Hydro, which is taken as the “true” flood magnitudes for the study area.

### 3.2. Prediction using a Regression Model

Initially, we developed a simple linear regression model to update flood values across the model domain based on new precipitation data. To explore the relationship between precipitation and flood value, several WRF-Hydro simulations were run. The precipitation from the April 5–6, 2017 event was taken as baseline rainfall and was varied (artificially) to give several sets of precipitation inputs. Precipitation was uniformly scaled by values from 2 to  $10 \times$  and  $1/(2 - 10 \times)$  (e.g.,  $1/10 \times$ ,  $1/9 \times$ ,  $1/8 \times$ , ...,  $8 \times$ ,  $9 \times$ ,  $10 \times$ ) the original precipitation. This captured how increasing or decreasing rainfall would translate to changes in river discharge and flood magnitude and provided a larger dataset to fit the regression model. There was a clear non-linear relationship between change in precipitation and changes in flood magnitude across the stream network; with all reductions in precipitation below  $\frac{1}{2}$  essentially reducing discharge to the same, low value (e.g., scaling precipitation by  $1/3$  or  $1/5$  essentially resulted in no flooding). Therefore, further analysis focused only on cases where precipitation was scaled by  $1/2$  or greater.

For the cases where  $(\frac{\text{precip. obs}}{\text{precip. mod}} \geq 0.5)$ , a simple regression function was fit to the data to predict new *flood.val* based on the old *flood.val*, the magnitude of the precipitation change, and the static watershed characteristics (*DA* – drainage area, *refkdt* – infiltration parameter, and *ksat* – saturated hydraulic conductivity). For cases where  $(\frac{\text{precip. obs}}{\text{precip. mod}} < 0.5)$ , *flood.val\_new* is assumed to be simply  $\frac{\text{precip. obs}}{\text{precip. mod}}$  times *flood.val.old*. The equations that describe this regression model are as follows:

$$\begin{aligned} \text{flood.val\_new} = \text{flood.val\_old} + \\ \left( \frac{\text{precip. obs}}{\text{precip. mod}} - 1 \right) (1.746 * \text{flood.val\_old} + 5.63 \times 10^{-4} * \text{DA} - 3.6 \times 10^{-2} * \text{refkdt} + 4.26 \times 10^5 * \text{ksat}), \\ \text{for } \frac{\text{precip. obs}}{\text{precip. mod}} \geq 0.5 \end{aligned} \quad (6)$$

$$\begin{aligned} \text{flood.val\_new} = \frac{\text{precip. obs}}{\text{precip. mod}} * \text{flood.val\_old}, \\ \text{for } \frac{\text{precip. obs}}{\text{precip. mod}} < 0.5 \end{aligned} \quad (7)$$

In all cases, *precip.mod* is the initial estimate/forecast of precipitation and *precip.obs* is the observed precipitation. The ratio  $\frac{\text{precip. obs}}{\text{precip. mod}}$  represents the ratio of how much higher or lower this observed precipitation is compared to the initial estimate. This regression model requires an initial prediction of flooding (*flood.val.old*). That is, it takes some forecast of streamflow and updates this forecasted value based on differences in observed and forecasted precipitation. This is different than the machine learning approach (see below), which predicts stream discharge

through successively iterating upon the cell state, requiring an initial condition of no previous precipitation and streamflow for the very first timestep.

### 3.3. Prediction using machine learning

The overarching goal for developing ML-based models for flood prediction was that this simpler approach could produce comparable outputs to that of high fidelity hydrology models such as WRF-Hydro, but with only a fraction of the necessary computational resources and runtime. Our models do not aim to be as accurate as WRF-Hydro; rather, the aim is to produce results that are good enough and are able to be run on constrained satellite platforms. These quick onboard estimates can direct satellites to observe locations where additional data are most valuable, without the need for ground-based planning and delayed data transfers.

The general framework for the model is such that each stream is represented as a node. Each node takes an input *I*, consisting of the precipitation and any stream specific domain parameters (Table 2), and outputs the flow rate *Q* (subscripts *u* and *d* represent upstream and downstream, respectively). WRF-Hydro reports the total flow rate *Q<sub>total</sub>*, which consists of all upstream flows as well as the flow within the current stream due to local precipitation. Thus we are able to create a graph network structure for the entire river system: each node represents the local flow *Q* due to local precipitation, which when combined with other local flows generate *Q<sub>total</sub>*. The building blocks of this network structure is shown in Fig. 4. During training, the nodes are trained upon a randomized selection of streams from WRF-Hydro, using the average precipitation over the grid points where the stream flows through.

Initially, we recognize there are two types of streams: headwaters and non-headwaters. Headwater nodes are essentially where the entire streamflow network “begins”, and thus have no input flow apart from possible groundwater and minute sources (which we ignore), whereas all other nodes are non-headwater. The distinction is important since WRF-Hydro does not report *Q* for headwater nodes, and thus their output must be inferred. Continuing to organize the entire streamflow structure, we separate the types of inflow combinations into three types: HH, NH, or NN, where H represents headwater nodes and N represents non-headwater nodes (i.e., NH would represent a non-headwater node combining with a headwater node). The difference between these three

---

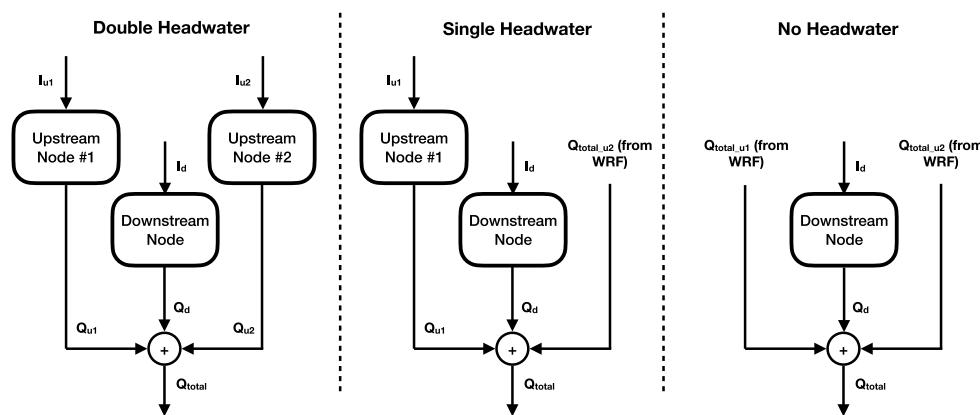
combinations is mainly due to the training structure: HH will require three sets of inputs, NH will require two sets of inputs, while NN will only require one set of inputs. In addition, HH will only require *Q<sub>total</sub>* for the downstream node, while NH and NN will also require WRF-generated *Q<sub>total</sub>* values for the respective upstream nodes during training. Continuing this logic, one can construct the entire stream flow network from these basic building blocks as a graph network, shown in Fig. 5. Here we make an important definition: the order of a stream node is the maximum distance of the node to any headwater node, including itself. Thus, as the network flows downstream, the order continuously increases, with large rivers associated with higher-order values.

Each stream node is modeled as an LSTM block containing 128 filters, followed by 2 dense layers of filter sizes 16 and 1, respectively, applied upon the LSTM output, shown in Fig. 6. In total, this constitutes 71,201 total trainable parameters, a total of roughly 300 kB in size. It is

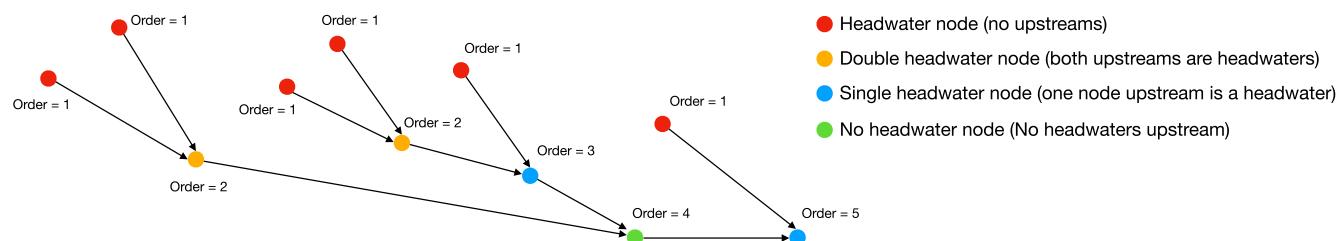
**Table 2**

Inputs and outputs to each stream node in the LSTM model.

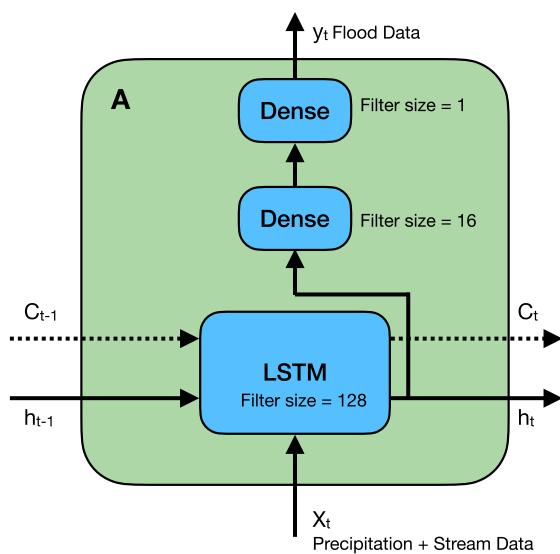
Inputs						Outputs	
Stream parameters (network)	Precipitation – $P$	Stream Length - $L$	Stream Slope – $S$	Drainage area – $A$	$Q$ – Flow rate		
Domain parameters used (grid)	Saturated soil hydraulic conductivity - dksat	Parameter in surface runoff parameterization - refkdt					



**Fig. 4.** Basic stream network structures during the LSTM training process (2 headwaters, 1 headwater, or no headwaters). WRF always outputs  $Q_{total}$  values, and so does not exist when the upstream node is a headwater node. The output  $Q_{total}$  is derived from summing the output of all upstream nodes and the current node. In the case of 2 headwater nodes, all inputs must pass through their respective node networks before the summation, while non-headwater nodes can proceed with training by taking  $Q_{total}$  directly from WRF. Note that all stream nodes are identical within the LSTM structure, and hence must be able to capture a variety of scenarios given appropriate inputs.



**Fig. 5.** Sample graph network representation of stream network, consisting of combinations of headwater and non-headwater nodes. Each stream node is represented as an LSTM + associated dense layers. Order is defined as the maximum distance of the current node to any headwater node.



**Fig. 6.** Detailed representation of a stream LSTM node, consisting of an LSTM and 2 dense layers.

also important to note that all stream nodes are represented as identically structured LSTMs across all types of stream flow structures. The reasoning behind this is to only require saving one set of variables that can be applied across all stream flows (being as general as possible), while keeping the overall memory and computational requirements manageable. However, due to the availability of output data (recall that WRF does not produce  $Q$  for headwater nodes), training the LSTM is accomplished by considering the three types of stream flow structures separately, each with its own appropriate number of inputs & outputs. For example, the case with no headwaters is arguably the simplest, as both  $Q_{total\_u1}$  and  $Q_{total\_u2}$  from upstream nodes can be taken directly from WRF, and only one set of inputs is required for the node to generate its own  $Q$  before summation and comparison to the WRF output  $Q_{total}$ .

The LSTM we applied consists of 40 time steps (i.e., 10 h at 15-min time steps) for the purposes of training, starting from an initial condition of no precipitation and ideally no (or low) stream flow. Theoretically, we are not limited to a set number of time steps, since we can continue to add LSTM nodes per additional timestep as necessary. We randomly select portions of the entire stream flow network during training, taking note and feeding relevant stream parameters into its corresponding stream flow structure. All inputs are normalized against their respective maximum and minimum values, such that they ideally range from -1 to 1. The only exception here is  $Q$ , since the summation of two negative values will yield a larger negative value, and so it is instead normalized from 0 to 1. The final error is defined as the difference between the WRF generated  $Q_{total}$  and the LSTM model-generated  $Q_{total}$ .

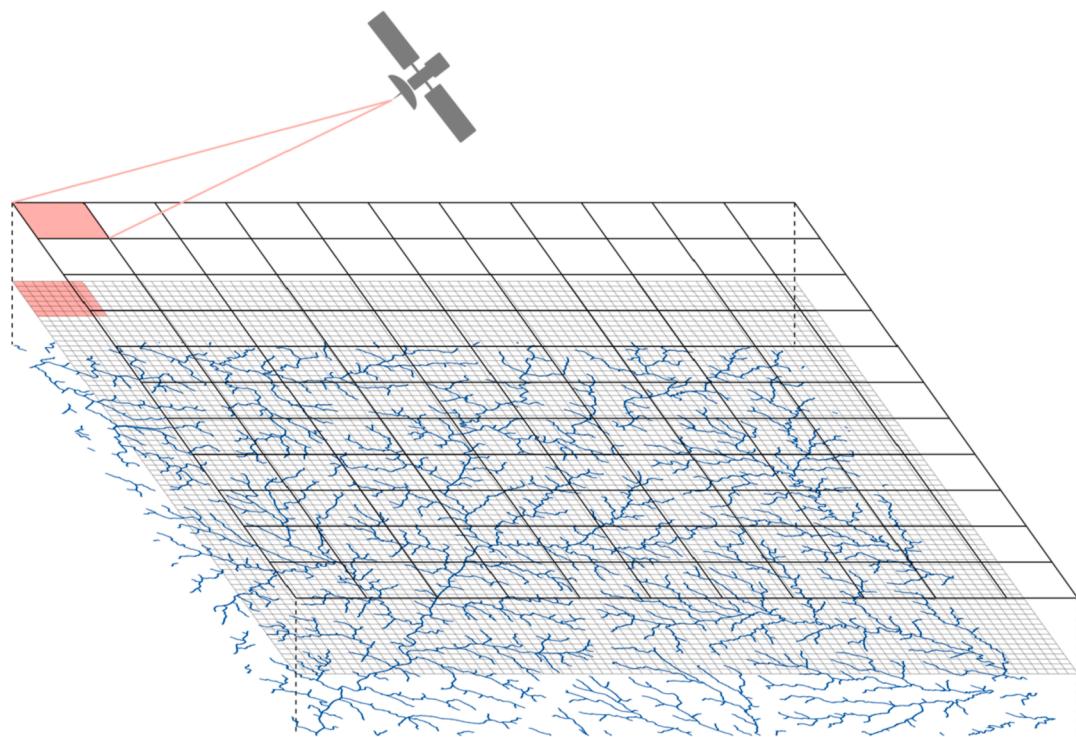
Since the problem is one of regression, we tested both mean squared error as well as absolute squared error as error metrics. During prediction, we added an artificial lag of one timestep (15 min) to the downflow  $Q$  during summation for every five stream nodes, such that it mimicked the nature of downstream flow. Although this is a simplification, encoding such a step within the LSTM would require training a larger LSTM that can capture this type of delay as well as scaling flood flows accordingly between two very different streams. As such, we opted for a simpler solution here that does not require additional training.

### 3.4. Simulation setup

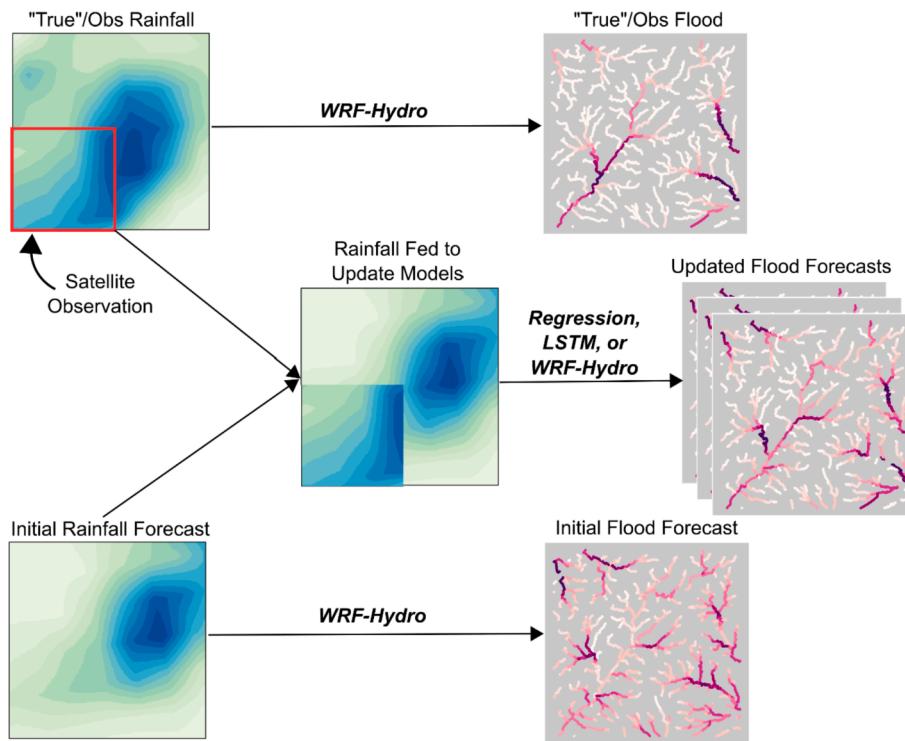
To assess the prediction accuracy of the update models (regression and LSTM), we must first standardize the method in which we discretize the landscape from the satellite's perspective. The regression model is run for every modeled stream reach based on average observed parameters (i.e., precipitation, saturated hydraulic conductivity ( $ksat$ ), infiltration parameter ( $refkdt$ )) in the contributing watershed. Similarly, every stream reach is treated as its own LSTM neural network. While this can yield detailed results, the satellite needs to be scheduled to point toward a vector of spatial points at scheduled times. The landscape is therefore discretized into equally spaced, regular grid cells (of a size comparable to the satellite's field of view). We assigned the stream reach-based predictions of flood magnitude from the regression and LSTM models to the appropriate grid cell (or cells) nearest to this reach. If more than one stream reach intersects a grid point, the highest predicted flood value is used. These grid-based flood values would then be sent to the satellites to inform where to make future observations. Note that satellites can only observe the amount of precipitation falling over an area, and thus must infer the flow rate  $Q$  from models. We created an initial precipitation forecast for the region (and subsequent flood forecast from WRF-Hydro) but with every observation in space and time, this precipitation map is updated and fed to either the regression or LSTM model to predict new river flood values.

The stream network consists of a total of 7,013 reaches, spread over a total of  $79 \times 79$  high resolution grid cells ( $0.9 \times 0.9$  km) (Fig. 7). The satellite is scheduled to observe a low-resolution version of the area with approximately  $8 \times 8$  km cells, thereby the entire map is spread over  $11 \times 11$  grid cells. When the satellite observes one of these  $11 \times 11$  grid-points, all high resolution gridpoints within the larger satellite observation area observed (and may have different precipitation rates). Satellites may be scheduled to observe a varying number of grid cells each time they pass over an area, as well as varying timing between these observation cycles. Since we treat every 15 min as one timestep within our forecast simulation, the highest observable scenario is one in which every one of the 121 grid cells are observed every 15 min. Varying these observational parameters, we discretize our simulation to observe a varying 30, 50, 75, 100, or 121 grid cells per observation cycle, with observation intervals occurring either 15, 30, 60, 120, or 180 min apart. Observed grid cells were randomly selected, but remained consistent among all models tested. The lower bound of sampling frequency is limited to once every 3 h because ground stations of commercial satellite constellations are able to establish space-ground contact between their assets every 4–6 h, and we wanted to demonstrate the benefits of on-board autonomy if updates can be made at higher frequency.

The prediction accuracy of the simple regression and machine learning approaches was assessed based on precipitation observations at the various number of grid cells and observation cycles. These two approaches were also compared to flood forecast updates using the WRF-Hydro model at these same update frequencies (e.g., these updated precipitation forecasts were fed through the WRF-Hydro model to generate new flood forecasts at each time step). These tests were run on the same April 5–6, 2017 rainfall event used in the WRF-Hydro calibration. The NLDAS precipitation data was taken to be the “true” precipitation values for this event. We developed a synthetic initial forecast of precipitation for this event by perturbing the “true” values using a Gaussian random filter. Both these precipitation data were run through WRF-Hydro to yield a “true” flood scenario and an initial flood forecast



**Fig. 7.** Schematic showing the coarse  $11 \times 11$  grid the satellite observes (top), the fine  $79 \times 79$  grid that WRF-Hydro is run on, and the underlying stream network. If a satellite observes precipitation in one grid cell (red), it observes precipitation in all the underlying fine grid cells. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)



**Fig. 8.** Schematic of the relationship between rainfall data and flood prediction for a single time step. “True”/Obs precipitation is from NLDAS and initial rainfall forecast is an artificially perturbed version of these data. Each of these precipitation time series are run through WRF-Hydro to generate “true” flood values and an initial flood forecast. For the satellite prediction system, the satellite observes some subset of this “true” precipitation. This is combined with the initial rainfall forecast data for areas not observed by the satellite and fed to the update models (regression, LSTM, and WRF-Hydro) to create an updated flood forecast for that time step. If no precipitation is observed by a satellite in a given time step, the initial rainfall forecast is fed to the update models.

(Fig. 8). The error between the forecast and “true” flooding is the initial error. The simple regression and LSTM update approaches should then reduce this error by incorporating different amounts of observed “true” precipitation. Areas that were not observed by the satellite are assigned their initial forecasted precipitation values. We applied our update methods to a 10 h period (April 5 10:00 – April 5 20:00) where most of the flooding occurred.

Error was quantified based on differences in predicted and true categorical flood magnitudes based on NWS flood categories (Eq. (5)) as well as calculated flood values ( $Q_m/Q_2$ ). Using normalized values is useful when comparing streams of different sizes and behaviors, since large rivers will consistently have higher flow rate  $Q$  when compared to smaller streams. Furthermore, using flood categories allows us to generalize the importance of different degrees of flooding, while ignoring the smaller errors that exist within the prediction of the raw stream flow itself. Finally, flood categories are most representative of the severity of flood impacts and are standard forecast communication tools used in the U.S. While categorical data should generally not be treated as numeric, flood categories are ordered and are nearly linearly related (Eq. (5)). Computing average categorical errors (see below) is therefore an appropriate and useful metric for comparing performance between our models.

These errors can be calculated on stream-based or grid-based results. Stream-based error is calculated for every stream reach simulated in the model. Grid-based error is calculated based on flood values assigned from these modeled stream reaches to the  $79 \times 79$  model grid. Both stream-based and grid-based error is calculated as follows:

$$E(x, t) = |\hat{F}_{x,t} - F_{x,t}| \quad (8)$$

where  $E(x, t)$  is error and  $\hat{F}$  and  $F$  represent the predicted (e.g., LSTM) and real categorical or numeric flood values, respectively. This provides error for every location ( $x$ , either stream reach or grid cell) and time step ( $t$ ). We can also average error over time and/or location. Mean flood error at each time step can be calculated as follows:

$$E(t) = \frac{1}{N} \sum_x |\hat{F}_{x,t} - F_{x,t}| \quad (9)$$

where  $N$  is the total number of stream reaches or grid cells. We can average across both space and time to get a single error metric for a given model scenario. This mean flood error is calculated as:

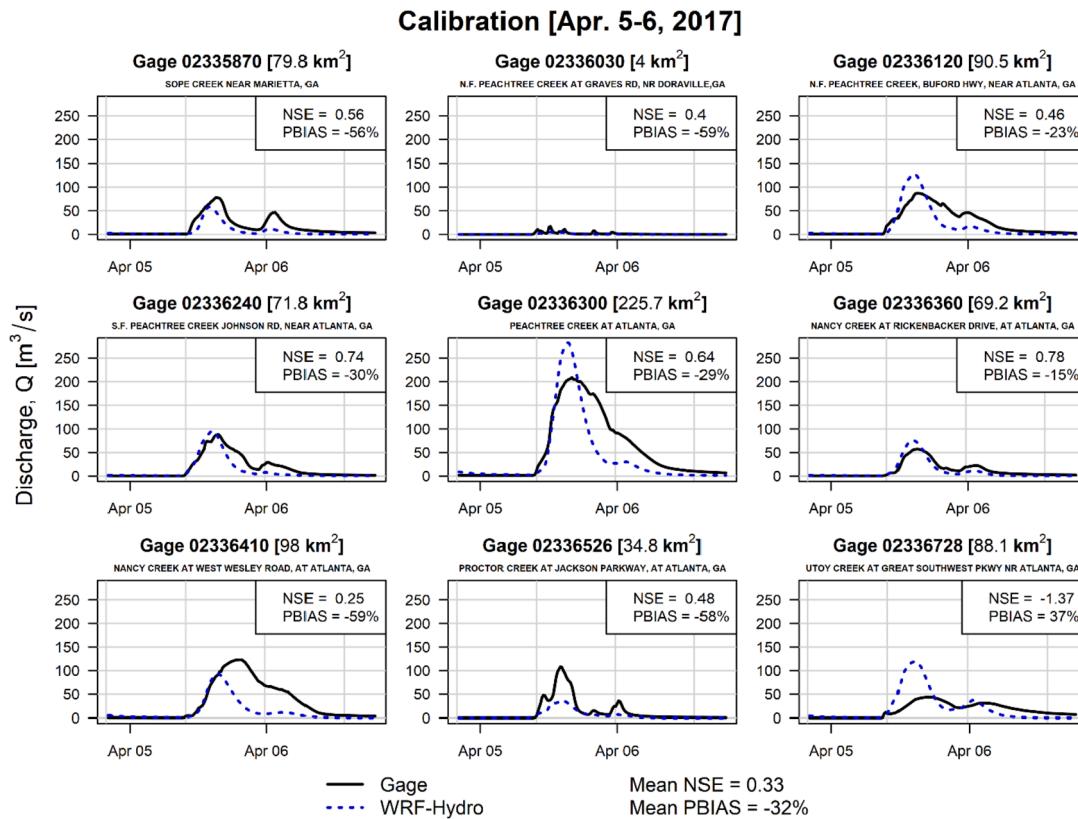
$$\bar{E} = \frac{1}{TN} \sum_t \sum_x |\hat{F}_{x,t} - F_{x,t}| \quad (10)$$

where  $T$  is the total number of prediction cycles. We use this average error metric to compare model errors across different satellite observation parameters (i.e., number of cells observed and observation frequency).

## 4. Results

### 4.1. WRF-Hydro model calibration and validation

The WRF-Hydro model was calibrated to match observed hydrographs from nine USGS stream gages in the Atlanta metro area for a storm event on April 5–6, 2017. Mean NSE was 0.33 and mean percent bias was −32% (Fig. 9). Modeled flows matched observed flows well for some gages, but under- or over-predicted peak flow rates at others. Gages with multiple peaks were reasonably well captured by WRF-Hydro. Validation performance was similar (June 20–21, 2017 storm event), with a mean NSE of 0.51 and percent bias of −42% (Fig. S1). In both the calibration and validation, negative bias indicates a consistent under-prediction of discharge. This is primarily because the model seems to be under-predicting runoff volume (i.e., area under the hydrograph) even if the peak flow rates were relatively well simulated for most sites. Model errors may be a result of simplifications in the model itself or incorrect parameterization. WRF-Hydro is a relatively coarse scale model (30 × 30 m resolution for overland flow routing) and we do not incorporate any stormwater infrastructure, reservoirs, or other complexity in this highly urban area that affects runoff rates and volumes. Additionally, the under-prediction of runoff volume may



**Fig. 9.** WRF-Hydro calibration results comparing measured and modeled discharge for nine USGS gages. For validation data, please refer to the Supplemental Materials (Fig. S1). Gage numbers and names shown along with watershed areas. NSE and PBIAS are shown for each gage, as well as average values for all gages.

suggest that too much infiltration is being simulated. However, additional tuning of calibration parameters did not result in significantly improved model performance.

Despite this systematic under-prediction of runoff volume and errors in the absolute value of flood peaks, the calibrated WRF-Hydro predicts the relative magnitude of flooding for most gages. The model captures where high flows occur and where they do not (especially visible in the validation results), which is important for our interests of determining when and where floods are occurring. Although accuracy would ideally be higher, these model results still serve as a useful realistic test case for comparison to our proposed flood forecasting methods.

#### 4.2. Flood forecast update performance

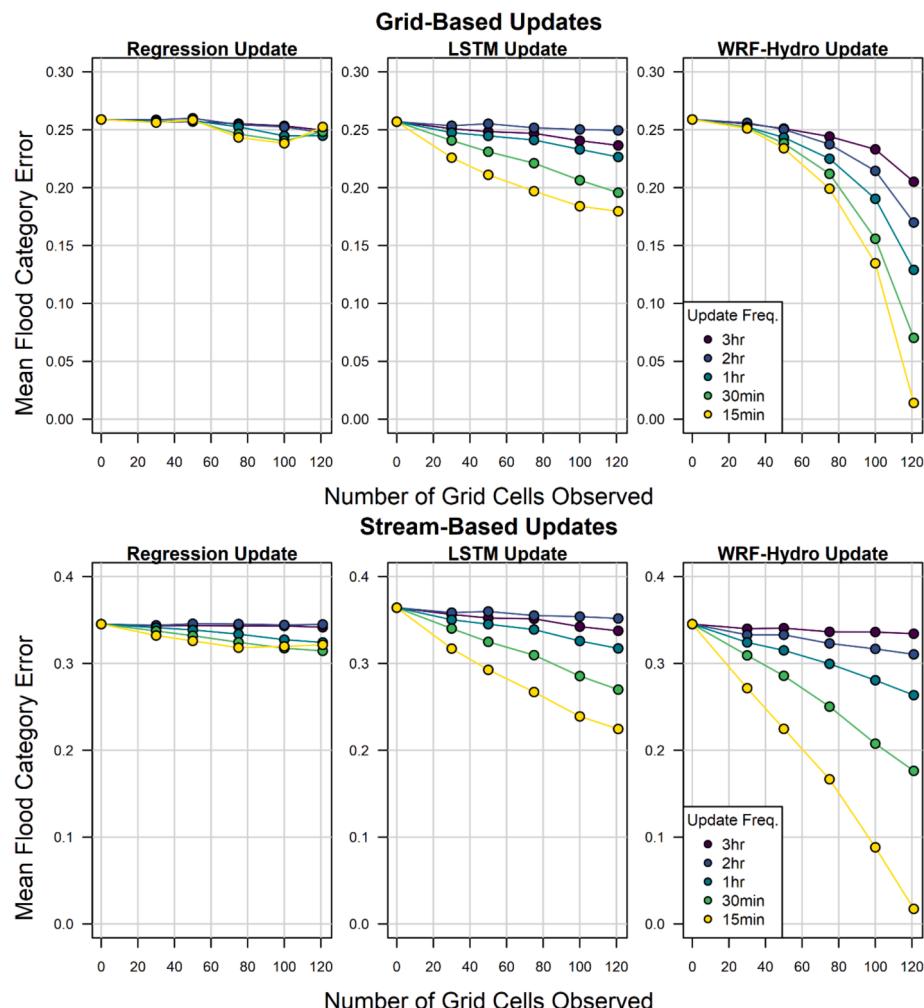
We compared the performance of the simple regression model update and the LSTM update approach to the performance of the WRF-Hydro model for forecasting flooding for the April 5–6, 2017 storm event. The accuracy of the flood update function generally improved as more precipitation data were incorporated (Fig. 10). The grid-based categorical flood error (averaged over space and time) decreased as the spatial and temporal coverage of satellite-observed precipitation increased. Overall improvement, however, was small for the regression model update, especially if 50 or fewer grid cells were observed. The WRF-Hydro update, as expected, showed considerable reduction in flood errors, especially as more observed precipitation data were incorporated. The LSTM update showed moderate error reduction, with much better performance than the regression update but not as good as the WRF-Hydro update. Fig. S2 shows similar results, but using mean flood value ( $Q_m/Q_2$ ) error instead of flood categories. The error reductions are lower for both the regression update and LSTM update using this metric, but the general trends are the same.

We also examined how error changes over the course of the simulation (i.e., averaged over space, but not time). Fig. 11 shows grid-

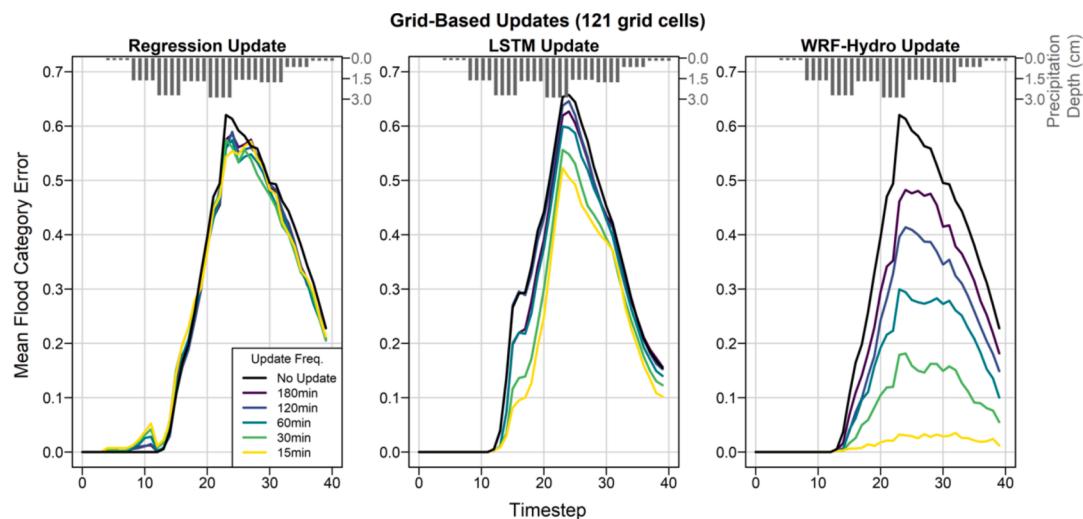
averaged flood categorical error over time for different update frequencies, if all 121 grid cells are observed by the satellite. All update approaches showed little error initially, with a peak around time step 23, which coincides with the greatest observed-modeled flooding across the study domain. The full WRF-Hydro update displayed the lowest error, especially as more observed precipitation data are incorporated (15-min update frequency). The simple regression update only marginally reduced error, although the improvement from the no update scenario was highest at the flooding peak, where reducing error was the most important. Notably, there was a small spike in error around time step 10, where the regression update was forecasting flooding before it actually occurred. Since the update function used a precipitation ratio, it was sensitive to small differences in precipitation that may have resulted in a large ratio. Again, the LSTM update performed better than the simple update, but cannot reduce error as much as the full WRF-Hydro update. For all updates, error increased as more precipitation falls, but decreased again once precipitation (and flooding) abated.

While the results shown so far are all averaged across all grid cells (because the satellite breaks the landscape into a grid for observation), the update methods themselves are applied to stream reaches. Looking at categorical (Fig. 10, bottom) and numeric (Fig. S2, bottom) flood error averaged over time and all modeled stream reaches shows slightly better performance for the LSTM update, compared to the grid-based results. The maximum error reductions (i.e., lowest error value from Fig. 10 compared to no update) using this stream-based averaging was 9% for the regression update and 38% for the LSTM update. These are slightly better than the maximum error reductions using the grid-based averaging (8% for the regression update and 30% for the LSTM update). Maximum error reduction for the WRF-Hydro update was slightly better for the grid-based averaging (99%) compared to the stream-based averaging (95%).

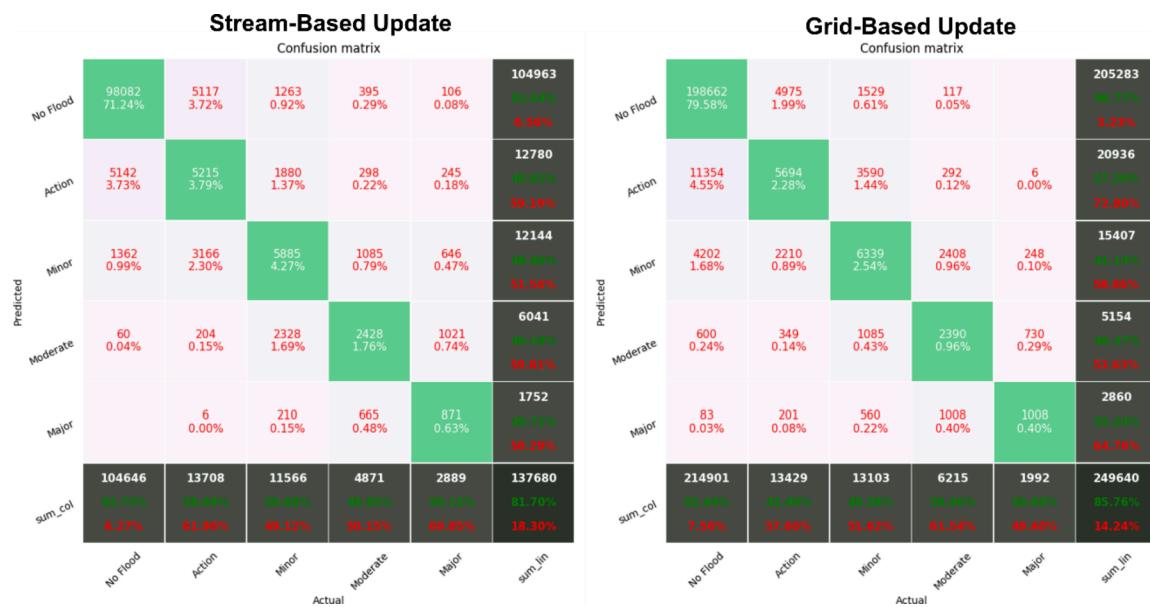
Since the LSTM update approach performed better than the regression update, we will examine this approach in more detail. Since our



**Fig. 10.** Grid-average (top) and stream-averaged (bottom) flood categorical error (Eq. (10)) for different update frequencies and spatial scales for the simple regression model, LSTM model, and WRF-Hydro update.



**Fig. 11.** Grid-averaged flood category error over time for the three update approaches at different update frequencies. All data are for the scenarios where precipitation is observed at all 121 grid cells. Bars at top show average “true” precipitation depth across the whole modeled grid to illustrate the general temporal pattern of the storm.



**Fig. 12.** Confusion matrix for categorical results for streams (left) and grid (right). Each cell shows the number and percent of observations (stream or grid location for each time step) for each predicted and actual flood category. Green cells show accurate predictions. Row and column sums are shown in gray, with correct (green text) and incorrect (red text) prediction percentages. Although ‘no flood’ dominates the majority of the data, we note that for our purposes that as long as the predicted value is within 1 category away from its actual category, it is considered acceptable for satellite tasking. If there are no values in a cell then there were no observations with that combination of predicted and actual flood category. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

error metrics are based on categorical flood values, it was useful to construct a confusion matrix of the LSTM update results for both stream and grid-based results (15 min update frequency, all 121 grid cells observed), shown in Fig. 12. The LSTM correctly classified flood categories for 81.7% (stream-based update) and 85.8% (grid-based update) of cases. If we assume that it was acceptable to be at most one flood category off from our predictions for satellite tasking purposes, we can recalculate our metrics for only situations when a flood was actually occurring (since the majority of streams over most of the time had no flooding). Using this approach, the LSTM model had an accuracy of 89.8% (stream-based update) and 96.3% (grid-based update). That is, the LSTM approach accurately predicted flooding within one category for at least 90% of cases (time steps and locations).

However, the accuracy of the LSTM approach was not the same for all streams. Fig. S3 shows sample predictions from LSTM compared to WRF-Hydro outputs for low order streams (close to headwaters). The LSTM models tracked those of WRF fairly closely, at least for streams of lower order. The results tended to diverge for larger, higher order streams and rivers since they have different rainfall-runoff dynamics compared to small streams.

To evaluate the ability of the LSTM to handle varying scenarios, we applied it to several artificial precipitation events. First, we doubled the amount of observed precipitation from the base test case (April 5–6, 2017 event) (Fig. S4). Although the error increased compared to the original case, the average categorical error was still relatively low. Similarly, we also applied the LSTM to a case where input precipitation was  $\frac{1}{2}$  of the original. Error overall is lower (Fig. S5) likely because flooding magnitude was also lower. The LSTM performs particularly well if it has the most possible precipitation data (15-min update frequency, yellow lines). Both of these cases are for the same basic storm event that the LSTM was trained on (albeit with different total amounts of rainfall).

We also applied the LSTM to the June 20–21, 2017 storm event that was used to validate the original WRF-Hydro model (and which was not used at all to train the LSTM). For this storm event, the LSTM approximated the WRF-Hydro hydrographs in most cases (Fig. S6). Looking only at the categorical predictions, the results show it performed almost

identically to WRF-Hydro. The mean absolute categorical error of the LSTM method (LSTM prediction compared to the observed gage) was 0.37, identical to the WRF-Hydro error. At most, the LSTM predicts flood categories no worse than that of WRF, at a fraction of the runtime.

## 5. Discussion

This paper builds out a predictor or forecast module for automated scheduling of satellite constellations, which can adaptively observe global urban floods caused due to transient precipitation events. Small satellites with precipitation radars can measure rain through space and time (e.g., RainCube), but they need to infer flood severity in observed locations in the absence of direct measurements. Forecasting flood severity is a key input to the satellite’s scheduler so that it can prioritize observations of areas with maximum flooding, for more accurate measurements, better understanding of the evolving phenomena, and thereby improved disaster management. The predictor module is important for adaptive sensing even when physical parameters are directly observable but dynamic (e.g., soil moisture) because future plans will be a function of the rapidly changing environment and their forecasts.

We proposed and compared three tools for predicting flood severity across a metropolitan area using dynamic precipitation measurements by satellites – a simple regression-based update model, a LSTM machine-learning algorithm, and a fully distributed hydrologic model (WRF-Hydro) – for the purpose of forecasting and adaptive sensing. These tools range from low to high complexity and low to high computational cost. The simple regression model can produce flood updates quickly (on the order of seconds) but only marginally reduces error from an initial flood forecast. The LSTM produces much less error, but does require more computational time (on the order of minutes). Finally, the full WRF-Hydro has the least error, but has by far the highest run times (on the order of hours), which is currently infeasible for onboard satellite platforms. To update flood forecasts in real-time based on the latest available precipitation data, it is important to balance performance (error reduction) with computational cost. This is especially important for running these flood forecast models on-board a satellite, which has

limited computational power.

Given these competing constraints, it appears the LSTM approach represents an appropriate balance by providing reasonably accurate flood forecasts relatively rapidly. The simple regression update does have the lowest cost, but it cannot adequately reduce flood forecast error. This perhaps is not surprising since it is a relatively naïve modeling approach that attempts to relate changes in flood magnitude to only relative differences in cumulative observed precipitation (albeit with some adjustment for watershed characteristics). Complex rainfall-runoff processes are not mechanistically accounted for in this approach. The LSTM approach was also shown to be relatively robust, performing well when precipitation depth was increased and decreased (Figs. S4 and S5) and when applied to a different rainfall event than was used in model training (Fig. S6). For this validation event, the LSTM predicted actual flood categories with similar accuracy as WRF-Hydro. Percent error for WRF-Hydro (compared to gage-measured flood peak) was lower (median 7%) than for the LSTM (median 40%). Comparatively, LSTMs used by other flood models (Kao et al., 2020; Le et al., 2019) display maximum percentage errors of up to 28.2% and 50%, respectively, for 3 and 6 timesteps away from the point of prediction. We note that our LSTM does not have access to upstream inflows like these other models, and must predict 40 timesteps of highly dynamical flow scenarios over many different types of streams. Furthermore, the LSTM showed appreciable error reduction in predicting flood categories with increasing precipitation data (Fig. 10), even if numeric flood value error reduction was smaller (Fig. S2). Overall, these results suggest that the LSTM model can predict the general flood severity (as flood category), but has difficulty predicting the exact flood discharge.

Comparing the confusion matrices (Fig. 12) of categorical error from the LSTM also yields some interesting insights: we note that for at least 90% of relevant flooding cases, the LSTM predicts flooding that is at most one category away from the actual flooding category. While not perfectly accurate, most of the predictions are not far off from the actual event. The “no flooding” case does dominate our statistics, but is still important to the prediction effort. Given limited satellite passes and observation opportunities, it is critical that we do not overestimate flooding scenarios such that it diverts these resources away from areas that are flooding. From a flood warning perspective, on the other hand, it is better to overestimate a flooding case than the underestimate it, particularly for any category above ‘minor’ flooding.

The LSTM did perform significantly better than the simple regression approach; however, it did not perform equally well across all streams. To visualize where error is highest, it is useful to split the categorical error per stream into their own order groups, shown in Fig. 13. It can be seen that as the streams turn into larger rivers, the error increases. The reason

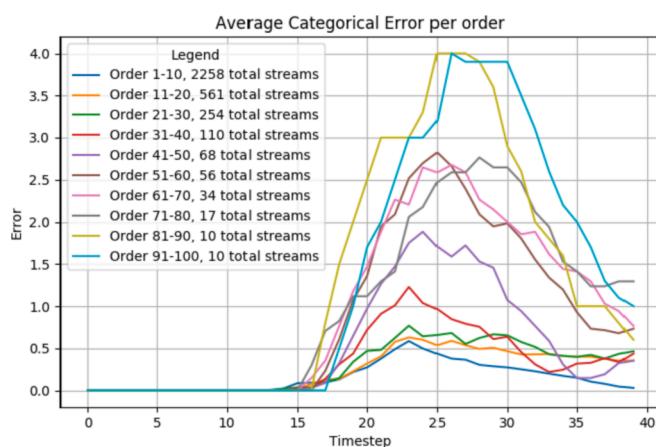
for this is twofold: first, because there are only a few large streams within the training data, the LSTM does a poor job of capturing the different dynamics that dominates the larger flows. Second, any error in the flow rate  $Q$  propagates downstream, compounding many small errors into several large errors, especially when thousands of small streams eventually flow into a few large rivers. As a result, the LSTM tends to overestimate the flow rate  $Q$  in larger rivers, since local precipitation has a smaller effect on these slow-moving streams than the faster flows encountered near headwaters.

These challenges were also observed when training the LSTM on large rivers; because of the different dynamics between streams of different sizes, it was difficult for the models to converge (the weights would oscillate if one trained on small streams, then on large rivers, and vice versa). As a result, much of the training was focused on double or singular headwater flows (since these were most common). This works well even for non-headwater flows in the beginning, but eventually the errors accumulate, especially for downstream flows. To remedy this problem, we propose that separate LSTM models need to be trained depending upon the order of the stream in question. Additionally, summing tributary inflows, although simple, may be inadequate for predicting flows in larger streams.

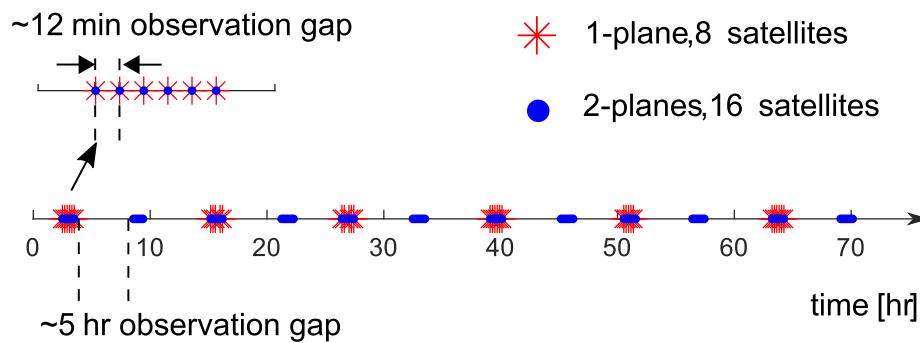
Quick and accurate flood prediction in smaller streams is more important to satellite adaptive sensing than in larger rivers. Small streams tend to respond quickly to rainfall, and this fast hydrologic response can be damaging but does not have the same breadth of flood forecasting tools as larger rivers (Gourley et al., 2017). Lag times for flooding in large rivers are longer, and therefore allows for more computationally intensive and accurate flood modeling to be run on ground-based computers and for adaptive sensing strategies to be uplinked to satellites during overpasses. On the other hand, rapid onboard updating of flood forecasts for small streams using satellite-derived precipitation could be extremely beneficial.

Flood prediction accuracy increases with more observations of precipitation for all three update methods, thereby quantifying sensitivity to the “adaptiveness” of adaptive sensing. We examined this effect both spatially (number of grid cells the satellite observed) and temporally (time between satellite observations). For the WRF-Hydro update, the satellite needs to observe over half of the available grid cells before we see significant reductions in error. In comparison, the LSTM seems to actually have greater error reduction at these low numbers of observed grid cells (for the grid-based errors at least), as long as observation frequency is high (Fig. 10). In our framework, observation scale and frequency are constrained by satellite availability. Our results are useful for scheduling satellite passes over the study area to obtain the necessary frequency and spatial distribution of precipitation to reduce error in flood forecasts below an acceptable level. Importantly, we quantify the value of recurring observations to overall predictive capability, and at what point additional satellite passes may have diminishing returns. This is especially relevant because parallel studies have shown that single satellite overpasses over the Atlanta urban region allow enough time to observe less than 30% of the grid cells (Sin et al., 2021), therefore predicting their relative value and scheduling to observe the most valuable areas maximizes mission value. This observation constraint is due to precipitation radars being narrow field of view sensors which can observe small fractions of Atlanta in a single image.

The primary focus of this paper was the prediction module portion of Fig. 1; however, it is useful to provide more details of satellite constellation topologies as this is relevant to the results discussed above. Low Earth Orbit satellites are dynamic with respect to grid cells on the Earth and rain radars onboard can observe any ground point over an access duration of 10–15 min. It can take several days for a single satellite to revisit that ground point, depending on its orbit and latitude of the ground point. Flying a constellation of satellites improves revisit significantly, but with marginal benefits after a certain number. As discussed in a parallel study (Nag et al., 2019), a 24 satellite can provide revisits as frequently as every 15 min and a maximum gap duration of a



**Fig. 13.** Mean stream-based categorical flood error over time per order group. Error tends to be low for streams near headwaters (low order) and increases as the streams turn into rivers (high order).



**Fig. 14.** Simulated access time of the Atlanta metro region of two satellite configurations (1 or 2 planes with 8 satellites each). All the satellites are at a Sun-synchronous orbit of 705 km altitude, orbital inclination of 98.2° (like the Landsat-8 orbit).

few hours, when arranged in 3 orbital plans. An example of revisit frequencies for different satellite constellations is shown in Fig. 14. A single plane constellation gives a minimum observation gap of 12 min, following by a long 10–12 h gap before the constellation can revisit the region of interest. Additional constellations decrease these revisit gaps, providing more relevant observation data. Small satellite companies have demonstrated commercial success in flying large constellations of satellites (Foster et al., 2018; Stringham et al., 2019) and commercial launch providers can deploy dozens of satellites as rideshare options (e.g., SpaceX). Reducing the satellites or orbital planes increases the time gap between revisits, allowing less frequent re-planning cycles based on the previously collected round of measurements. Therefore, we demonstrate the performance of the forecast models as a function of measurement update frequency varying from 15 min to 3 h in Fig. 10, to capture the variability of various constellation abilities make measurements at said cadence.

Regardless of the observation frequency, the scheduling of satellites is based on conceptualizing the landscape as a grid. However, our update methods performed at least marginally better at the stream-reach scale, compared to the grid-based approach (taking the maximum observed flooding in each cell) due to the loss in information during the transfer. Supplying grid-based values makes the scheduling problem more tractable for satellites, but exploring the best ways to make this conversion to minimize errors is an important area of further research.

The LSTM update approach can and will be applied to other regions to determine its applicability outside of our one model domain, so that the predictive models can be extended to global adaptive sensing. This would require training a new set of LSTMs based on regional topography, soils, stream network, and other local characteristics. As discussed previously, LSTM approaches have been applied to forecast flooding in other regions (Kao et al., 2020; Kratzert et al., 2018; Le et al., 2019). There is therefore growing evidence that the basic structure of LSTMs is suitable for flood prediction in a variety of situations. However, each of these models are trained for specific locations and it remains to be seen whether more generalizable LSTM models can be developed that can be rapidly applied to a variety of locations with minimal site-specific data and training.

There are several opportunities to improve LSTM accuracy in future work. For example, we used a single LSTM model structure for all interactions between streams. One improvement would be to develop a separate LSTM model based on stream type (i.e., size or number of tributaries). Additionally, more model parameters could be incorporated (e.g., more complex urban land cover and stormwater infrastructure). In both cases, there is a risk of overtraining the models, reducing the ability to predict flooding for a variety of storm events. We attempted some of these approaches, including separating the double headwaters, single headwaters, and no headwaters scenarios into their own separate models, as well as including additional domain variables (i.e., elevation and flow accretion), but the results were on par with our original LSTM model.

The results presented in this paper successfully demonstrate the development of methods to accurately predict riverine flooding using space-based measurements of precipitation and an approach that could be run on-board a satellite constellation. This has some major advantages over more computationally intensive and centralized approaches for updating predictions of flood events, which is especially important in regions without ground-based precipitation and streamflow sensors (as long as sufficient data are available to train a LSTM model). The proposed approach deserves further testing to determine when and where it can replace or complement existing flood forecasting approaches. Our results also add to growing evidence that machine learning approaches in general (Yaseen et al., 2019) and LSTM in particular (Kao et al., 2020; Kratzert et al., 2018; Le et al., 2019) are promising methods for accurately and efficiently predicting river flows and flooding without the need for data-intensive and computationally demanding process-based modeling.

## 6. Conclusion

This paper proposes two simplified flood forecast models that are computationally efficient enough to run onboard a constellation of satellites – providing an approach to prioritize where satellite-observed precipitation data will be most informative during ongoing flood events. We show that one method, the LSTM model, is able to mimic the more complicated, physics-based WRF-Hydro model relatively well, correctly classifying flooding within one flood category in 90% of cases. The LSTM approach runs orders of magnitude faster than WRF-Hydro, making it more suitable to run quickly on-board resource limited satellites. Additional testing showed the LSTM model was robust to changes in precipitation and associated flood severity. Finally, we demonstrated continuous reductions in flood prediction error as the spatial extent and temporal frequency of satellite precipitation observations increased. These findings together suggest that LSTM models are appropriate for informing responsive scheduling of satellite constellations – directing these satellites to observe precipitation where it will be most useful for updating flood forecasts during an ongoing flood event. Future work can help refine this approach and develop more general LSTM models that are applicable beyond our study region.

## CRediT authorship contribution statement

**Roderick Lammers:** Conceptualization, Methodology, Software, Formal analysis, Writing – original draft, Writing – review & editing, Visualization. **Alan Li:** Conceptualization, Methodology, Software, Formal analysis, Writing – original draft, Writing – review & editing, Visualization. **Sreeja Nag:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Supervision, Funding acquisition. **Vinay Ravindra:** Conceptualization, Methodology, Formal analysis, Writing – original draft, Writing – review & editing.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgements

This work was supported by the NASA Science Mission Directorate New Investigator Program [RC-CREST Cooperative Agreement #NNX12AD05A]. We are grateful to Dr. Brian Bledsoe for his support with the initial project development and Dr. Sujay Kumar for his review and comments on preliminary results from this work. We are also grateful for the comments of three anonymous reviewers that greatly improved the manuscript.

## Appendix A. Supplementary data

Supplementary data to this article can be found online at <https://doi.org/10.1016/j.jhydrol.2021.127175>.

## References

- Andersen, T.K., Shepherd, J.M., 2013. Floods in a changing climate. *Geographys. Compass* 7, 95–115.
- Blackwell, W.J., Braun, S., Zavodsky, B., Velden, C., Greenwald, T., Herndon, D., Bennartz, R., DeMaria, M., Chirokova, G., Atlas, R., Dunion, J., 2018. Overview of the NASA TROPICS cubesat constellation mission. In: *CubeSats and NanoSats for Remote Sensing II*. International Society for Optic and Photonics, p. 1076908.
- Carter, R.W., Davidian, J., 1968. General procedure for gaging streams, in: *Techniques for Water-Resources Investigations*. U.S. Geological Survey.
- Chen, Y., Jiang, H., Li, C., Jia, X., Ghamisi, P., 2016. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Trans. Geosci. Remote Sens.* 54 (10), 6232–6251. <https://doi.org/10.1109/TGRS.2016.2584107>.
- Chien, S., McLaren, D., Doubleday, J., Tran, D., Tanpipat, V., Chitradon, R., 2019. Using taskable remote sensing in a sensor web for Thailand flood monitoring. *J. Aerospace Inf. Syst.* 16 (3), 107–119. <https://doi.org/10.2514/1.I010672>.
- Chung, J., Gulcehre, C., Cho, K., Bengio, Y., 2015. Gated feedback recurrent neural networks, in: *32nd International Conference on Machine Learning*, ICML 2015. pp. 2067–2075.
- Cunge, J.A., 1969. On the subject of a flood propagation computation method. *J. Hydraul. Res.* 7, 205–230.
- Doubleday, J., Chien, S., Norton, C., Wagstaff, K., Thompson, D.R., Bellardo, J., Francis, C., Baumgarten, E., 2015. Autonomy for remote sensing – Experiences from the IPEX CubeSat. International Geoscience and Remote Sensing Symposium (IGARSS) 5308–5311. <https://doi.org/10.1109/IGARSS.2015.7327033>.
- Feaster, T.D., Gotvald, A.J., Weaver, J.C., 2014. Methods for estimating the magnitude and frequency of floods for urban and small, rural streams in Georgia, South Carolina, and North Carolina, 2011. doi:10.3133/sir20145030.
- Foster, C., Mason, J., Vittaldev, V., Leung, L., Beukelaers, V., Stepan, L., Zimmerman, R., 2018. Constellation phasing with differential drag on plant labs satellites. *J. Spacecraft Rockets* 55, 473–483.
- Gochis, D.J., Barlage, M., Dugger, A., Fitzgerald, K., Karsten, L., McAllister, M., McCright, J., Mills, J., RafieejNasab, A., Read, L., Sampson, K., Yates, D., Yu, W., 2018. The WRF-Hydro modeling system technical description (Version 5.0).
- Gourley, J.J., Flamig, Z.L., Vergara, H., Kirstetter, P.E., Clark, R.A., Argyle, E., Arthur, A., Martinaitis, S., Terti, G., Erlingis, J.M., Hong, Y., Howard, K.W., 2017. The FLASH project - Improving the tools for flash flood monitoring and prediction across the United States. *Bull. Am. Meteorol. Soc.* 98, 361–372. <https://doi.org/10.1175/BAMS-D-15-00247.1>.
- Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. *Neural Comput.* 9 (8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>.
- Kao, I.-F., Zhou, Y., Chang, L.-C., Chang, F.-J., 2020. Exploring a Long Short-Term Memory based Encoder-Decoder framework for multi-step-ahead flood forecasting. *J. Hydrol.* 583, 124631. <https://doi.org/10.1016/j.jhydrol.2020.124631>.
- Kim, S., Shen, H., Noh, S., Seo, D.-J., Welles, E., Pelgrim, E., Weerts, A., Lyons, E., Philips, B., 2021. High-resolution modeling and prediction of urban floods using WRF-Hydro and data assimilation. *J. Hydrol.* 598, 126236. <https://doi.org/10.1016/j.jhydrol.2021.126236>.
- Kratzert, F., Klotz, D., Brenner, C., Schulz, K., Herrnegger, M., 2018. Rainfall – runoff modelling using Long Short-Term Memory (LSTM) networks. *Hydrol. Earth Syst. Sci.* 22 (11), 6005–6022.
- Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. ImageNet Classification with Deep Convolutional Neural Networks. *Adv. Neural Inf. Process. Syst.* 25, 1097–1105.
- Lee, J.H., Yuk, G.M., Moon, H.T., Moon, Y.I., 2020. Integrated flood forecasting and warning system against flash rainfall in the small-scaled urban stream. *Atmosphere* 11, 971. <https://doi.org/10.3390/ATMOS11090971>.
- Le, X.H., Ho, H.V., Lee, G., Jung, S., 2019. Application of Long Short-Term Memory (LSTM) neural network for flood forecasting. *Water* 11, 1387. <https://doi.org/10.3390/w11071387>.
- Li, A.S., Chirayath, V., Segal-Rosenheimer, M., Torres-Perez, J.L., van den Bergh, J., 2020. NASA NeMO-Net's Convolutional Neural Network mapping marine habitats with spectrally heterogeneous remote sensing imagery. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* 13, 5115–5133.
- Linnabary, R.B., Obrien, A.J., Smith, G.E., Ball, C., Johnson, J.T., 2019. Using cognitive communications to increase the operational value of collaborative networks of satellites, in: *2019 IEEE Cognitive Communications for Aerospace Applications Workshop*, CCAAW 2019. 10.1109/CCAAW.2019.8904900.
- Li, Y., Grimaldi, S., Walker, J.P., Pauwels, V.R.N., 2016. Application of remote sensing data to constrain operational rainfall-driven flood forecasting: A review. *Remote Sens.* 8, 456. <https://doi.org/10.3390/rs8060456>.
- Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M., 2013. Playing Atari with Deep Reinforcement Learning. *arXiv preprint arXiv:1312.5602*.
- Nag, S., Aguilar, A., Akbar, R., Azemati, A., Frank, J., Levison, R., Li, A., Moghaddam, M., Ravindra, V., Selva, D., 2020. D-SHIELD: Distributed spacecraft with heuristic intelligence to enable logistical decisions. *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*.
- Nag, S., Li, A.S., Merrick, J.H., 2018. Scheduling algorithms for rapid imaging using agile Cubesat constellations. *Adv. Space Res.* 61 (3), 891–913.
- Nag, S., Li, A., Ravindra, V., Sanchez Net, M., Cheung, K.-M., Lammers, R., Bledsoe, B., 2019. Autonomous scheduling of agile spacecraft constellations with delay tolerant networking for reactive imaging, in: *International Conference on Automated Planning and Scheduling SPARK Workshop*, Berkeley, CA.
- Nag, S., Net, M.S., Li, A., Ravindra, V., 2020b. Designing a disruption tolerant network for reactive spacecraft constellations, in: *ASCEND 2020*. p. 4009.
- Nair, V., Hinton, G.E., 2010. Rectified linear units improve Restricted Boltzmann machines, in: *ICML 2010 - Proceedings*, 27th International Conference on Machine Learning.
- NASA, 2020. SMAP Soil Moisture Active Passive [WWW Document]. URL: <https://smap.jpl.nasa.gov/>.
- Nash, J.E., Sutcliffe, J.V., 1970. River flow forecasting through conceptual models Part I – A discussion of principles. *J. Hydrol.* 10 (3), 282–290. [https://doi.org/10.1016/0022-1694\(70\)90255-6](https://doi.org/10.1016/0022-1694(70)90255-6).
- NCAR, 2020. WPS V4 Geographical Static Data Downloads Page [WWW Document]. URL: [https://www2.mmm.ucar.edu/wrf/users/download/get\\_sources\\_wps\\_geog.html](https://www2.mmm.ucar.edu/wrf/users/download/get_sources_wps_geog.html).
- Nguyen, D.H., Bae, D.-H., 2020. Correcting mean areal precipitation forecasts to improve urban flooding predictions by using long short-term memory network. *J. Hydrol.* 584, 124710. <https://doi.org/10.1016/j.jhydrol.2020.124710>.
- Niu, G.Y., Yang, Z.L., Mitchell, K.E., Chen, F., Ek, M.B., Barlage, M., Kumar, A., Manning, K., Niyogi, D., Rosero, E., Tewari, M., Xia, Y., 2011. The community Noah land surface model with multiparameterization options (Noah-MP): 1. Model description and evaluation with local-scale measurements. *J. Geophys. Res. Atmos.* 116, 1–19. <https://doi.org/10.1029/2010JD015139>.
- Olah, C., 2015. Understanding LSTM Networks [WWW Document]. URL: <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>.
- Peral, E., Tanelli, S., Haddad, Z., Sy, O., Stephens, G., Im, E., 2015. A proposed constellation of precipitation profiling radars in CubeSat. In: *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 1261–1264.
- Rahman, M.S., Di, L., 2017. The state of the art of spaceborne remote sensing in flood management. *Nat. Hazards* 85 (2), 1223–1248. <https://doi.org/10.1007/s11069-016-2601-9>.
- Ravindra, V., Nag, S., Li, A., 2020. Ensemble-guided tropical cyclone track forecasting for optimal satellite remote sensing, in: *IEEE Transactions on Geoscience and Remote Sensing*. doi:10.1109/TGRS.2020.3010821.
- Reising, S.C., Gaier, T.C., Padmanabhan, S., Lim, B.H., Heneghan, C., Kummerow, C.D., Berg, W., Chandrasekar, V., Radhakrishnan, C., Brown, S.T., Carvo, J., 2018. An Earth venture in-space technology demonstration mission for Temporal Experiment for Storms and Tropical Systems (TEMPEST). In: *IGARSS 2018 IEEE International Geoscience and Remote Sensing Symposium*, pp. 6301–6303.
- Rienecker, M.M., Suarez, M.J., Todling, R., Bacmeister, J., Takacs, L., Liu, H.-C., Gu, W., Sienkiewicz, M., Koster, R.D., Gelaro, R., Nielsen, J.E., 2008. The GEOS-5 Data Assimilation System: Documentation of Versions 5.0, 5.1.0, and 5.2.0. NASA Technical Report.
- Ruf, C.S., Chew, C., Lang, T., Morris, M.G., Nave, K., Ridley, A., Balasubramanian, R., 2018. A new paradigm in earth environmental monitoring with the CYGNSS small satellite constellation. *Sci. Rep.* 8, 1–13.
- Salas, F.R., Somos-Valenzuela, M.A., Dugger, A., Maidment, D.R., Gochis, D.J., David, C. H., Yu, W., Ding, D., Clark, E.P., Noman, N., 2018. Towards real-time continental scale streamflow simulation in continuous and discrete space. *J. Am. Water Resour. Assoc.* 54 (1), 7–27. <https://doi.org/10.1111/jawr.2018.54.issue-110.1111/1752-1688.12586>.
- Sin, E., Arcak, M., Nag, S., Ravindra, V., Li, A., 2021. Attitude trajectory optimization for agile satellites in autonomous remote sensing constellations. *AIAA Science and Technology Forum and Exposition (SciTech Forum)*.
- Skamarock, W.C., Klemp, J., Dudhia, J., Gill, D.O., Barker, D., Wang, W., Powers, J.G., 2008. A Description of the Advanced Research WRF Version 3.
- Song, T., Ding, W., Wu, J., Liu, H., Zhou, H., Chu, J., 2020. Flash flood forecasting based on long short-term memory networks. *Water* 12 (1), 109. <https://doi.org/10.3390/w12010109>.

- Stringham, C., Farquharson, G., Castelletti, D., Quist, E., Riggi, L., Eddy, D., Soenen, S., 2019. The Capella X-band SAR constellation for rapid imaging. In: IGARSS 2019 – IEEE International Geoscience and Remote Sensing Symposium, pp. 9248–9251.
- Williams, G.P., 1978. Bank-full discharge of rivers. *Water Resour. Res.* 14 (6), 1141–1154.
- Wu, H., Adler, R.F., Tian, Y., Huffman, G.J., Li, H., Wang, JianJian, 2014. Real-time global flood estimation using satellite-based precipitation and a coupled land surface and routing model. *Water Resour. Res.* 50 (3), 2693–2717. <https://doi.org/10.1002/2013WR014710>.
- Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, L., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., Dean, J., 2016. Google's NMT. ArXiv e-prints.
- Yang, L., Jin, S., Danielson, P., Homer, C., Gass, L., Bender, S.M., Case, A., Costello, C., Dewitz, J., Fry, J., Funk, M., Grammanan, B., Liknes, G.C., Rigge, M., Xian, G., 2018. A new generation of the United States National Land Cover Database: Requirements, research priorities, design, and implementation strategies. *ISPRS J. Photogramm. Remote Sens.* 146, 108–123. <https://doi.org/10.1016/j.isprsjprs.2018.09.006>.
- Yaseen, Z.M., Sulaiman, S.O., Deo, R.C., Chau, K.W., 2019. An enhanced extreme learning machine model for river flow forecasting: State-of-the-art, practical applications in water resource engineering area and future research direction. *J. Hydrol.* 569, 387–408. <https://doi.org/10.1016/j.jhydrol.2018.11.069>.