MID-AMERICA TRANSPORTATION CENTER

Report # MATC-UI: 145-4

Final Report
WBS: 25-1121-0005-145-4

Real-Time Flood Forecasting for River Crossings - Phase IV

Witold Krajewski, PhD
Rose & Joseph Summers Chair in Water Resources Engineering
Faculty Researcher Engineer, IIHR - Hydroscience & Engineering
Director, Iowa Flood Center
The University of Iowa

Nicolas Velasquez, PhD
Assistant Researcher, Iowa Flood Center
Department of Civil and Environmental Engineering
The University of Iowa

2022

A Cooperative Research Project sponsored by
U.S. Department of Transportation- Office of the Assistant Secretary for Research and Technology

MATC

Real-Time Flood Forecasting for River Crossings – Phase IV

Witold Krajewski, PhD, P.I.
Rose & Joseph Summers Chair in Water Resources Engineering
Faculty Research Engineer, IIHR - Hydroscience & Engineering
Director, Iowa Flood Center
The University of Iowa

Nicolas Velasquez, PhD, P.I.
Assistant Researcher, Iowa Flood Center
Department of Civil and Environmental Engineering
The University of Iowa

A Report on Research Sponsored by

Mid-America Transportation Center

University of Nebraska–Lincoln

January 2020

Technical Report Documentation Page

1. Report No. 25-1121-0005-145-4	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle Real-Time Flood Forecasting for River Crossings – Phase IV		5. Report Date November 2022	
		6. Performing Organization Code	
7. Author(s) Witold Krajewski, PhD ORCID: 0000-0002-3477-9281 Nicolas Velasquez, PhD ORCID: 0000-0001-8207-5492		8. Performing Organization Report No. 25-1121-0005-145-4	
9. Performing Organization Name and Address Mid-America Transportation Center Prem S. Paul Research Center at Whittier School 2200 Vine St. Lincoln, NE 68583-0851		10. Work Unit No. (TRAIS)	
		11. Contract or Grant No. 69A3551747107	
12. Sponsoring Agency Name and Address Office of the Assistant Secretary for Research and Technology 1200 New Jersey Ave., SE Washington, D.C. 20590		13. Type of Report and Period Covered Final Report, September 2020 – December 2021	
		14. Sponsoring Agency Code MATC TRB RiP No. 91994-81	
15. Supplementary Notes			
16. Abstract We have developed a generic prototype of a flood-forecasting model that is transferable to other locations around the Midwest to provide monitoring and forecasting flood potential at critical infrastructure points, such as bridges, where streamflow gauges are not available. A real-time web-based visualization platform to display the model predictions has been implemented. The platform will display the river network upstream from a point of interest and a time control slider that will allow exploring the evolution of flows everywhere in the network over the past several days, and about a week into the future. The model uses in-house developed radar-rainfall maps updated every 5 minutes with the spatial resolution of about 0.5 km currently covering the Iowa domain and extending some 100 km into the neighboring states. For the future rainfall, we use predictions for the National Weather Service High-Resolution Rapid Refresh (HRRR) forecasting system. The system provides hourly accumulation products for up to 20 hours ahead. Our system expands the forecasting capabilities of the current NWS by providing predictions at locations that have not been historically gauged.			
17. Distribution Statement			
18. Security Classif. (of this report) Unclassified	19. Security Classif. (of this page) Unclassified	20. No. of Pages 50	21. Price

Table of Contents

List of Figures

List of Tables

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated in the interest of information exchange. The report is funded, partially or entirely, by a grant from the U.S. Department of Transportation's University Transportation Centers Program. However, the U.S. Government assumes no liability for the contents or use thereof.

Abstract

We have developed a generic prototype of a flood-forecasting model that is transferable to other locations around the Midwest to provide monitoring and forecasting flood potential at critical infrastructure points, such as bridges, where streamflow gauges are not available. Our efforts have centered around creating tools and protocols that would facilitate the implementation of the hydrological model in any of the four MATC states. The protocols include 1) a methodology to use existing regional data to determine the parameters in the runoff routing equation along the river network, 2) a methodology to determine the infiltration parameters that control rainfall-runoff transformation, and 3) Technology transfer between the University of Iowa and the University of Nebraska. In this phase, we focus our work in the validation of the methodologies previously developed to estimate the routing and infiltration parameters.

Chapter 1 Preliminaries: The Iowa Flood Center HLM hydrological model

The Iowa Flood Center hydrological model, Hillslope-Link Model (HLM), is a distributed hillslope-scale rainfall-runoff model that partitions Iowa into over three million individual control volumes following the landscape decomposition outlined in Mantilla and Gupta (2005). The model is parsimonious, using ordinary differential equations to describe transport between adjacent control volumes. This characteristic reduces the computational resources needed by capturing the most essential features of the rainfall runoff transformation; it uses only a few parameters to obtain acceptable results. The model partitions the river network into river links (the portion of a river channel between two junctions of a river network) and the landscape into hillslopes (adjacent areas that drain into the links).

(a) (b)

Figure 1.1 (a) illustration of landscape decomposition into hillslopes and decomposition of the river network into channel link and (b) vertical soil profile and control volumes included in the hydrological model

Mass conservation equations give rise to the system of coupled nonlinear ordinary differential equations that represent changes in the water storage in the hillslope surface (s_{surf}), top soil (s_{tops}), and deep soil (s_{deeps}) given by,

$$\begin{cases} \dfrac{ds_{surf}(t)}{dt} = p(t) - q_{runoff}(t) - q_{infil}(t) - e_{surf}(t) & (1.1) \\ \dfrac{ds_{tops}(t)}{dt} = q_{infil}(t) - q_{percol}(t) - e_{tops}(t) & (1.2) \\ \dfrac{ds_{deeps}(t)}{dt} = q_{percol}(t) - q_{baseflow}(t) - e_{deeps}(t) & (1.3) \end{cases}$$

Fluxes in, across, and out of the vertical hillslope control volumes include precipitation $p(t)$, overland runoff $q_{runoff}(t)$, infiltration into the topsoil q_{infil}, percolation from the topsoil into the deeper soils $q_{percol}(t)$, baseflow into the channel $q_{baseflow}(t)$, and evaporation from the ponded, topsoil, and deep soil layers ($e_{surf}(t)$, $e_{tops}(t)$ and $e_{deeps}(t)$, respectively). The model assumes that percolation flux is a linear function of the amount of water stored at time t in the topsoil $q_{percol}=k_{percol}\cdot s_{tops}$ and that the baseflow is a linear function of the water stored in deep soil $q_{baseflow}=k_{baseflow}\cdot s_{deeps}$. Overland runoff is a power function of the water stored on the hillslope surface (consistent with Manning's equation) given by,

$$q_{runoff} = k_{runoff} s_{surf}^{1.67} \quad (1.4)$$

and infiltration is a nonlinear function of soil moisture content (s_{tops}/T_{tops}), where T_{tops} is the thickness of the topsoil layer (i.e., A-horizon) and a linear function of hydraulic head s_{surf} given by,

$$q_{infil} = k_{dry}\left(1 - \frac{s_{tops}}{T_{tops}}\right)^{\phi} s_{surf} \quad (1.5)$$

where k_{dry} corresponds to the case of dry soil and, similarly to k_{runoff}, k_{percol}, and $k_{baseflow}$ can be interpreted as time constant (residence time) of the respective storage component. The hillslope area (a_h) for the elements in the distributed model is, on average, 0.05 km², and link length (l_{link}) is, on average, 400 m. Note that $a_h/(2l_{link})$ is the hillslope length. The exponent φ is a nonlinearity introduced by the change in the potential matric of the soil column as soil moisture changes with time.

The HLM should be thought of as a modeling system rather than a single specific model. As the equations describing hillslope-scale processes are separated from the numerical solver, it is rather easy to explore different mathematical descriptions for water fluxes. For example, one can consider such simplifications as constant runoff coefficient or water transport velocity, or as an alternative, one can formulate these components based on the available physical characteristics.

Water transport through the river network is nonlinear and governs how channel links propagate flow through the river network. Formulated in the context of a mass conservation equation developed by Gupta and Waymire (1998), it uses the water velocity parameterization given by Mantilla (2007) as,

$$\frac{dq_{link}(t)}{dt} = \frac{v_0 q_{link}^{\lambda_1}(t) A^{\lambda_2}}{(1-\lambda_1)l}\left[a_h\left(k_{runoff}s_{surf}^{1.67}(t) + k_{baseflow}s_{deeps}(t)\right) - q_{link}(t) + q_1(t) + q_2(t)\right] \tag{1.6}$$

where q_{link} is the discharge from the link at time t, a_h is the total hillslope area draining to the link, $q_1(t)$ and $q_2(t)$ are the incoming flows of the upstream tributaries, A is the upstream basin area, and λ_1, λ_2, and v_0 are global parameters of the water velocity component of the model and are set to 0.2, -0.1, and 0.3, respectively. The model can capture the main features of the

hydrographs including the maximum stage. We used the model in several studies (e.g., Ayalew et al. 2014; Cunha et al. 2012). We also discuss the model performance in Krajewski et al. (2017). The model is driven by radar-rainfall estimated from Level II NEXRAD data from seven WSR-88D weather radars covering the state of Iowa. The maps of rainfall intensity have spatial resolution of about 0.25 km^2 and are updated every five minutes. The algorithms are described in Krajewski et al. (2013) and Seo and Krajewski (2015).

An important aspect of our modeling approach is the avoidance of calibration. Instead, we rely on detailed information of the physical properties we model. This includes the topography, land use and land cover, soil properties, and details of the main forcing, i.e., precipitation. Comparing simulation results to streamflow observations across Iowa validates the model formulation and parameterization. Therefore, we can view the model as data-intensive and calibration-free when used in forecast-mode. This, in turn, implies that with more detailed, relevant, and accurate data, including model states and physical domain characterization as well as the driving inputs, the model will work better. The model is fully automatic in the sense that no corrections are applied to the model as it moves forward in time once initial and boundary conditions are imposed.

The model predicts the streamflow fluctuations associated with storm events over the catchment of interest using current observations of rainfall, and rainfall forecasts. The effect of storms on riverways is usually delayed ranging from days to weeks. Each point of interest in the landscape (bridge, culvert) can then be categorized according to the maximum warning time. The web interface will provide a visual tool to show when a particular location will be impacted, and it will provide an inundation map associated with the particular peak flow expected for that

location. Inundation maps are more effective tools in communicating the effects of flooding than crest stages at specific locations.

Chapter 2 Downstream data assimilation

The latest advances include a data assimilation scheme into the model, and the technological transfer to the Nebraska university. The data assimilation is a simple yet powerful scheme that uses upstream streamflow observations to increase the performance downstream. On the other hand, we also worked in the technological transfer developing the streamflow rating curves for a set of streamflow gages in the Elkhorn watershed.

Hydrologic distributed models are subject to large uncertainty, affecting their prediction skill. The uncertainty is due to errors in the modeling components. These include: (1) the errors from inadequate representation (model structure) of physical processes (e.g., rainfall-runoff transformation); (2) the errors in the meteorological inputs (e.g. rainfall, evapotranspiration, snowmelt); and (3) the errors in the model parameters and initial conditions that reflect the state of the water in the ground. These uncertainties lead to inaccurate predictions of streamflow, and consequently reduce our capacity to produce reliable warning of floods.

Data assimilation can help to obtain better streamflow predictions by either altering the initial states of the hydrologic model (Dechant and Moradkhani 2011), changing the values of the model parameters (Moradkhani et al. 2005), or improving the model structure (Nearing and Gupta 2015). The nature of the observations assimilated in the models can range from small scale in-situ measurements (e.g. streamflow gages, soil moisture probes, etc.) to large scale remotely sensed variables. Examples of data that can be assimilated into models include streamflow (e.g., Mazzoleni et al. 2018), water depth (e.g., Neal et al. 2009), soil moisture (e.g., Brocca et al. 2014) or snow cover (e.g., Griessinger et al. 2016). In the current work we assimilate streamflow data, which is a cost-effective approach to improve model prediction accuracy (e.g., Noh et al. 2018).

2.1 Methodology

We take benefit of the mathematical formulation of HLM to include streamflow assimilation. In our implementation, the variable q in equation (2.1) is treated as an initial value of the ordinary differential equation. Streamflow data is assimilated into HLM using the direct insertion method (Daley 1991) described by

$$q_{t,l}^{+} = z_t^o \tag{2.1}$$

where z_t^o and q are the observation value and the updated model state at time step t at a particular channel link l . The updated state value is indicated with the superscript+.

The streamflow observed in real-time with 15-minute intervals is averaged to hourly values, and inserted as initial value, updating the state variable q in equation (2.1). The averaging to hourly values reduces the chances of inserting null values in the model states, that might result in crashing the simulations. The numerical solver Asynch then resolves the differential equation of hydrologic (i.e., channel) routing to the downstream links. Consequently, the upstream links of the insertion points do not undergo any effects from the assimilation, while the assimilation effects should propagate downstream from the gaging point.

We configured the HLM to simulate streamflow at every channel link of the river network, using observed radar rainfall and evapotranspiration as inputs. We setup the HLM simulation for the period from April 1 to December 1, 2016 and defined initial conditions for the values of all the state variables of the model in April 1 by spinning up the model for one month. We ran HLM with two simulation modes: (1) open loop simulation that is not associated with USGS observations; and (2) simulation with streamflow data assimilation. The results from the

open loop mode are used as reference to assess the effect of data assimilation. In the data assimilation mode, we define assimilation points as the gauges where streamflow observations are assimilated into HLM, and evaluation points as the downstream gauges where streamflow observations are used to assess model simulated streamflow propagated from their upstream assimilation points. We study the influence of the drainage area ratio between the assimilated and evaluated points in the performance of the simulations (Figure 2.1). We performed the experiment in the Cedar river Watershed. In it, we first asses the performance of the simulation when increasing the upstream gauges (Figure 2.1b). Then, we analyze how the performance changes across the main channel using just one upstream assimilation point (Figure 2.1c).

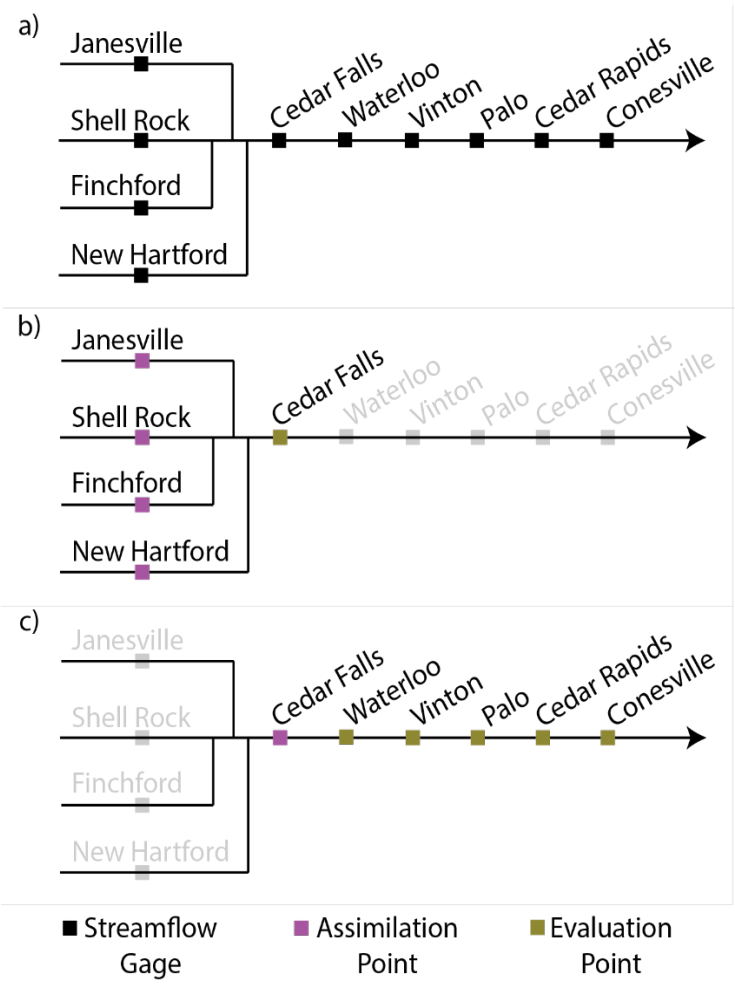

Figure 2.1 Schematics of the location of USGS gages in the Cedar River basin. The arrow indicates the direction of flow.

In the first experiment we assimilate one by one the data from the gauge stations upstream of Cedar Falls. According to table 2.1, the contributing area, and the distance to Cedar Falls of the upstream stations is highly variable.

Table 2.1 Values of drainage area ratio for the configuration of the model described in Figure 3b.

Site (USGS code)	Upstream Area (km^2)	Drainage Area Ratio (%)	Distance to Cedar Falls (km)
New Hartford (05463000)	349	0.07	12
Finchford (05458900)	852	0.18	14
Janesville (05458500)	1676	0.35	12
Shell Rock (05462000)	1711	0.36	20
Cedar Falls (05463050)	4714	1	0

2.2 Data assimilation results

In the first experiment the USGS gauges at Janesville, Shell Rock, Finchford and New Hartford are used as assimilation points, and the observations at Cedar Falls are used as evaluation points (see fig. 2.1b and table 2.1). The upper four panels in figure 2.2 shows the effect of assimilating one point at a time, and the bottom panel shows the result of using the four assimilation points simultaneously. The panels are sorted in order of drainage area ratio (see table 2.1); Hartford has the smaller ratio and Shell Rock has the larger. As the ratio of drainage area increases, the contribution of each gauge is more important to the increase in performance of the simulation, compared to the open loop simulation (no data assimilation). The results for Hartford and Finchford in the first two panels do not show much difference between the simulation with data assimilation compared to the open loop. The results from the contribution of Janesville and Shell Rock show improvements in the ability of the model to better reproduce the low flows and falling limb of the hydrographs. However, the Janesville assimilation creates an

over-estimation of the peak flow. Moreover, we obtain the best representation of the observations when we use all the upstream gauges (lower panel).

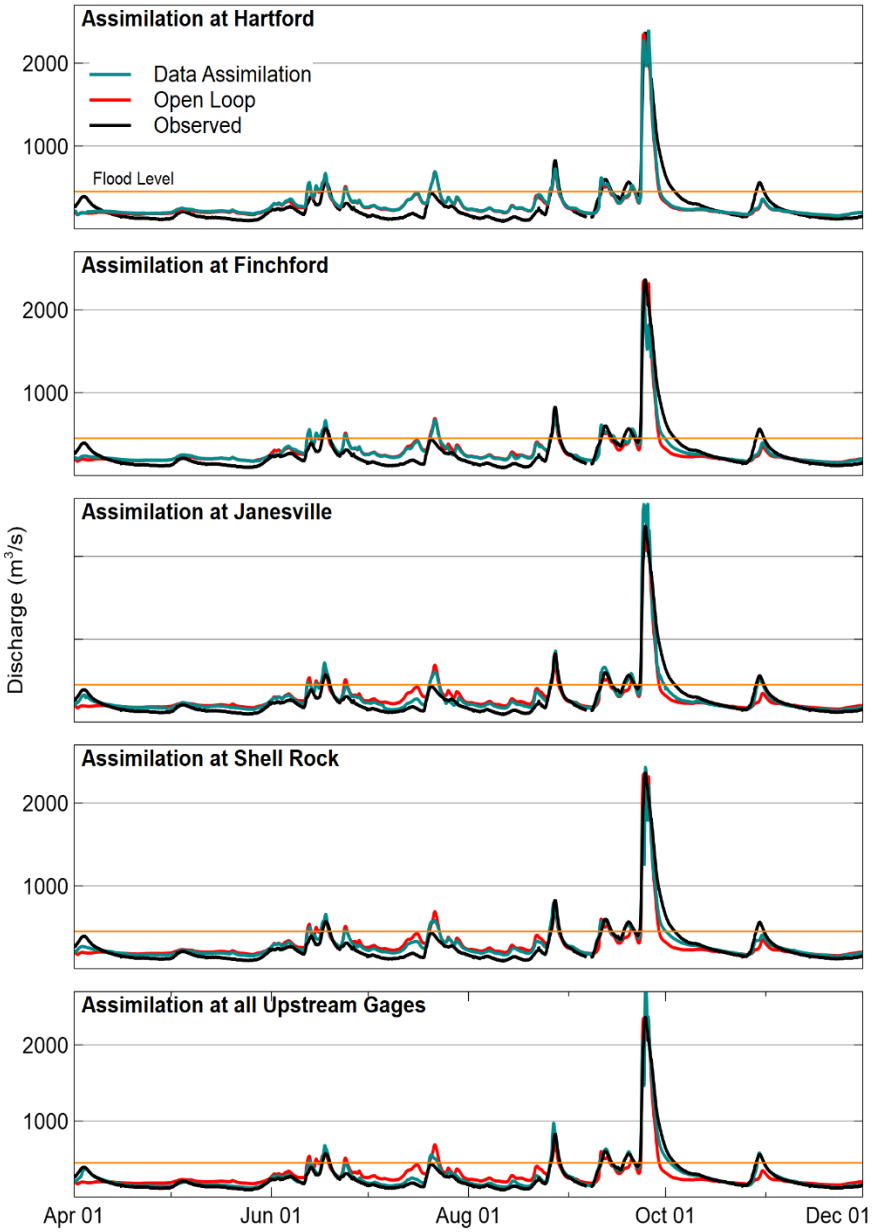

Figure 2.2 Results of the assimilation experiment at Cedar Falls using the configurations shown in figure 1.1.

Our second experiment assess the role of the drainage area ratio in the performance of the simulations. We use another configuration of the model where we assimilate the data observed at Cedar Falls and use the downstream gages as evaluation points (see fig. 2.1c). The upper left panel of figure 2.3 shows that when performing streamflow assimilation at Cedar Falls (the most

upstream gage in this experiment configuration), there is an increase of the model performance at all the downstream gages, in terms of KGE. The increase is proportional to the drainage area ratio; Waterloo being the gage closer to Cedar Falls and with larger area ratio increased its KGE from 0.77 to 0.97; Conesville is the gage with the smaller area ratio in the configuration and increased the KGE value from 0.7 to 0.83. We looked at other metrics that are also relevant to the problem of floods. The upper right panel shows the improvement in total volume estimation through the percent bias. In this metric, the optimal value of percent bias is zero. The streamflow assimilation reduces the volume over estimation produced by the model, as shown in the comparison with the results of the open loop simulations. The lower left panel shows the effect of streamflow assimilation in reducing the error of time to annual peak. In this experiment the assimilation reduces the time to peak error by about 10 hours. The lower right panel shows the effect of streamflow assimilation in reducing the peak flow magnitude error.

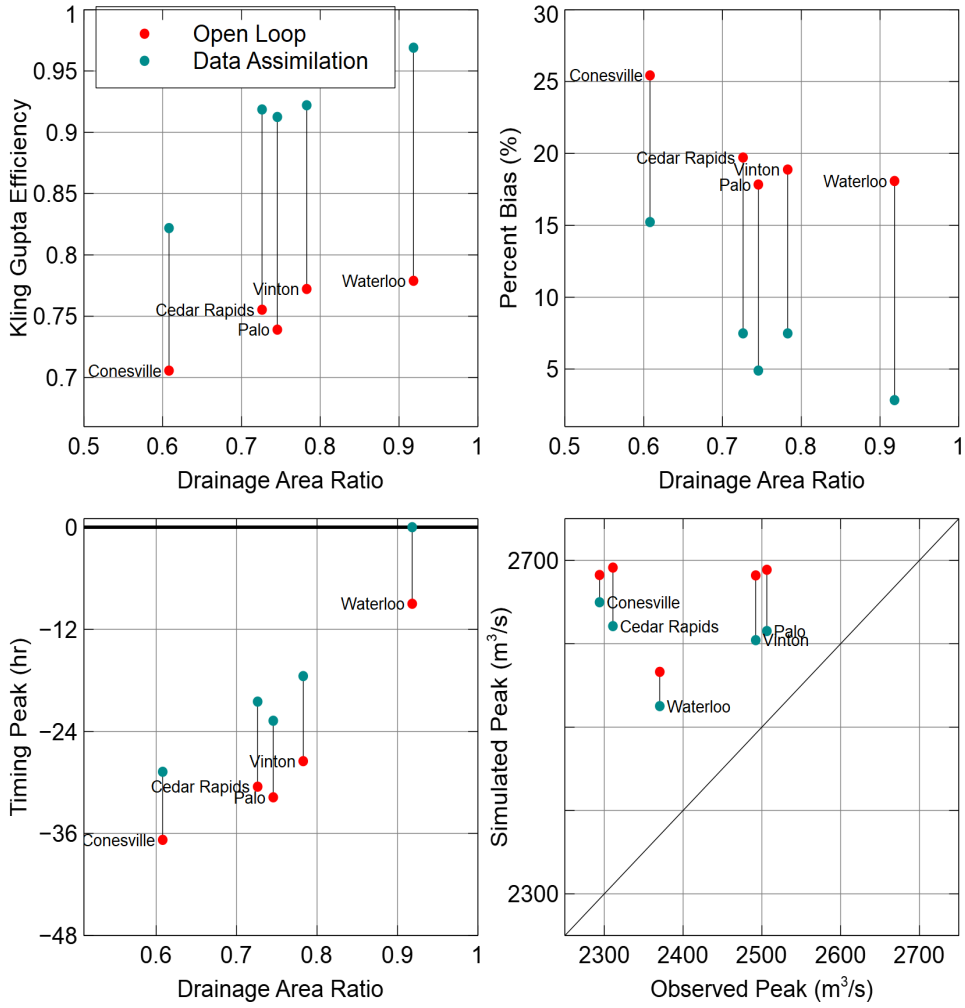

Figure 2.3 Results of the experiment showing the effect of the drainage area ratio in the performance of the simulations. The panels show the improvement of the data assimilation experiments compared to the open loop simulation in terms of Kling Gupta Efficiency, Percent Bias, Timing of Peak Flows, and Magnitude of Peak Flows.

Chapter 3 Implementation of a snow parameterization in HLM

During the winter to spring transition in 2019, southwest Iowa was primarily covered in snow. In March of that year, the bomb cyclone brought precipitation and increased temperatures to the region, detonating a significant flood. A forecast of the described event involves representing the snow water equivalent (SWE) processes and the runoff increase due to frozen soils. Additionally, for operational purposes, it may be parsimonious. With the described considerations, we implemented a new module for the Hillslope Link Model (HLM) called HLM-Snow, which uses the threshold temperature index (TI) method to accumulate and melt snow water equivalent (SWE). We tested the module by simulating the flood event in the Nishnabotna River in Iowa in March 2019. HLM-Snow outperformed other HLM modules with a more straightforward consideration of snow processes. The average skill of HLM-Snow to simulate flow in the selected event was KGE ≈ 0.74, and the skill in reproducing the SWE temporal variability was NSE ≈ 0.87. Our results suggest that a simple TI method can keep model parsimony while capturing flow and SWE.

The implementation of the snow parameterization is the result of a collaborative work between the University of Iowa and the University of Nebraska.

3.1 Introduction

Southwestern Iowa was one of the areas more severely affected by the bomb cyclone event in the U.S. Midwest in March 2019. The event was a combination of frozen ground conditions, rain-on-snow (ROS), temperatures above the melting point, and snowmelt resulting in a massive flood (Flanagan et al., 2020). The described mechanism is uncommon and happens mainly in the early stages of spring (McCabe et al., 2007). To better understand this flooding mechanism, it is essential to account for the weather processes before and during this event. In

the winter to spring transition, temperatures usually oscillate around the freezing point with chances of melting the snowpack (E A Anderson, 2006; Eric A Anderson, 1973; Follum et al., 2019) and shifting precipitation between snowfall and rainfall (Gray & Prowse, 1993). Also, the soil surface is typically frozen, blocking water infiltration into the soil and increasing the runoff ratio (Quinton et al., 2004; Suzuki et al., 2006). The mentioned factors describe a complex system that is challenging to represent. To simulate snow processes, modelers have used approaches with different levels of complexity and data requirements.

Snow accumulation and melting model's complexity rely on how the processes are represented. Energy balance (EB) models simulate more processes through physically-based equations. Some examples of EB models are Isnobal (Marks et al. 1999), SHAW (N. Flerchinger & E. Saxton, 1989), UEB (Tarboton & Luce, 1996), and SNOWPACK (Lehning et al., 2002). EB models have proven to be skillful when there is enough data (Kumar et al., 2013; Shakoor et al., 2018; Skaugen et al., 2018). However, EB family models require forcing data that is usually scarce or uncertain (Franz et al., 2008) and may have higher equifinality (Beven, 2006). On the other hand, temperature index (TI) models compensate the data limitation by using empirical relationships with the air temperature. Some TI methods include the Cemaniege module (Valéry, 2010), the threshold method, the linear method (Jordan, 1991), and the sigmoid method (Wang et al., 2019).

Choosing an EB or TI approach will depend on the study goal, the terrain conditions (Zaramella et al., 2019), and the available resources (Shakoor et al., 2018). EB models give more insights into the mechanisms and are usually more skillful in mountainous and forested regions (Fierz et al., 2003). On the other hand, TI models are parsimonious and have a comparable performance where the terrain is less complex (Lehning et al., 2006; Lundquist & Flint, 2006).

Considering the advantages and limitations of each approach, we used a TI model to reproduce the snow accumulation and melting process that caused the devastating flood of southwestern Iowa in March 2019.

This study's primary goal is to test the suitability of the TI model for simulating coupled processes of snow accumulation-melting and runoff generation by including those snow processes in the structure of a hydrologic model. For this purpose, we use the Hillslope Link Model (HLM) (Krajewski et al., 2017). HLM includes a module called HLM-FSnow that allows forcing snowmelt estimates to include the effect of this component in the runoff generation processes during the winter and spring months. Koya et al. (2022) proposed the formulation to include the TI model in HLM. In this study, we implement the Koya et al. (2022) formulation to develop and evaluate a new module called HLM-Snow that simulates snow-related processes using the TI model. We evaluate the suitability of the HLM-Snow module by simulating the flood event of March 2019 at the Nishnabotna River basin in southwestern Iowa. We compare HLM streamflow simulations to observations at five USGS gauges. We also evaluate the accuracy of HLM-Snow SWE spatially distributed estimates by comparing them to observed SWE fields in the watershed. The novelty of the study is that it explores the feasibility of using a parsimonious TI approach to improve the forecast skills of HLM, which has been operational for the last ten years (Krajewski et al., 2017; Quintero et al., 2020). Moreover, we present a comprehensive validation at multiple locations with observed streamflow and SWE.

3.2 Materials and Methods

The Nishnabotna River watershed is in southwest Iowa and has 7,300 km^2 (see fig. 3.1). The watershed has two main branches identified as the East and West Nishnabotna Rivers. The

land use is dedicated to farmlands, and a significant portion of its main river channels have been artificially straightened. The Nishnabotna River is a tributary to the Missouri River.

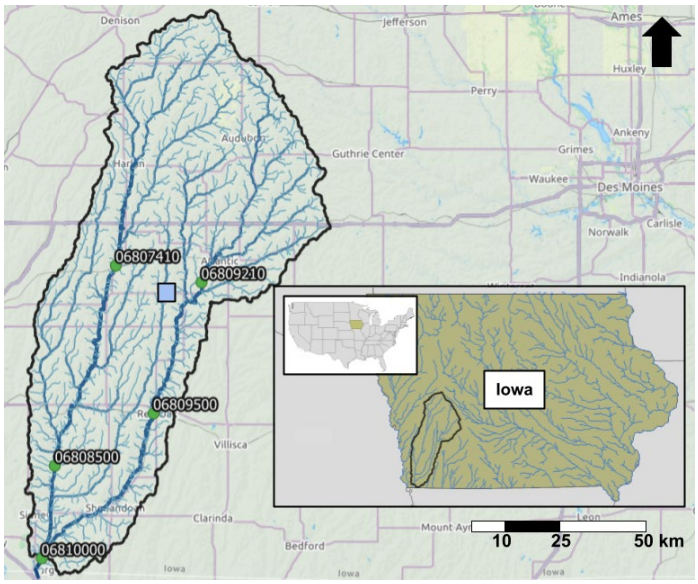

Figure 3.1 Nishnabotna watershed localization. Green dots correspond to the USGS gauges. The blue square corresponds to the ISU Mesonet Lewis Amstrong weather station.

3.2.1 The flood event of March 2019

In March of 2019, a deep surface cyclone detonated a series of devastating events over the central United States (Flanagan et al., 2020). Over eastern Nebraska and western Iowa, the flood event increased due to the surface conditions. Over the Nishnabotna River, SWE packs around 60mm melted, bringing devastating floods to Hamburg (Iowa). The SWE accumulation and melting in the watershed can be explained by temperature and rainfall observed oscillations (fig. 3.2a and b, respectively). Two successive precipitation events took place, one (March 9) under freezing conditions (blue area in figure 3.2) and another (March 13) under melting conditions (orange area in figure 3.2). The combination between accumulated SWE, rainfall and higher temperatures generated significant runoff, setting the conditions for high flows in the

watershed. We focused our work on understanding the described event using different setups of the parsimonious HLM model.

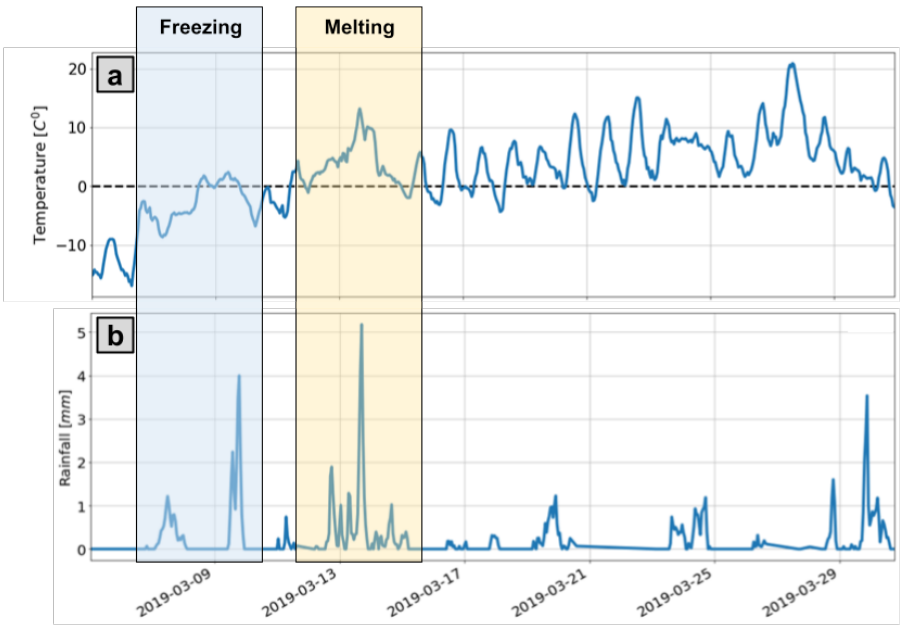

Figure 3.2 a) Temperature records at the Lewis Armstrong meteorological station. b) Mean areal rainfall obtained using MRMS (b). The colored boxes correspond to the two rainfall events over the Nishnabotna watershed under freezing (blue) and melting (orange) conditions.

3.3 Hillslope Link Model

HLM represents the hydrological processes at the hillslope scale and routes the streamflow through the channel network. The model is highly flexible, allowing the representation of rainfall-runoff processes using different modules, but is consistent in that the hydrologic response units are the hillslopes and the water is moved across the river network. In this work, we represent the hillslope runoff processes using three storages: ponded surface (S_p $[m]$); topsoil ($S_T[m]$); and subsurface storage ($S_s[m]$). Rainfall ($P(t)[mm^3 \cdot s^{-1}]$) goes into the ponded storage of each hillslope. The water from the ponded storage can either infiltrate the

topsoil ($q_{pT}[m \cdot min^{-1}]$) or flow as runoff to the channel link ($q_{pL}[m \cdot min^{-1}]$). The water in the topsoil percolates ($q_{Ts}[m \cdot min^{-1}]$) to the soil storage. Finally, the water in the soil storage seeps into the channel link as subsurface runoff ($q_{sL}[m \cdot min^{-1}]$). Evaporation ($e_i[m \cdot min^{-1}]$) occurs from the three storages as removal of volume from the model. A schematic of the model structure is presented in figure 3.3a. The differential equations describing the changes in storage are given by:

$$\frac{dS_p}{dt} = P - q_{pL} - q_{pT} - e_p \tag{3.1}$$

$$\frac{dS_T}{dt} = q_{pT} - q_{Ts} - e_T \tag{3.2}$$

$$\frac{dS_s}{dt} = q_{Ts} - q_{sL} - e_s \tag{3.3}$$

The subsurface flow q_{sL} has a linear ql_{sL} and an exponential qe_{sL} component. Fonley et al. (2020) describe the exponential formulation, validated for Iowa by Velasquez et al. (2021). Once in the river network, HLM transports the channel water $q[m^3 \cdot s^{-1}]$ downstream. A detailed description of the streamflow routing and hillslope flows processes can be found in (Mantilla & Gupta, 2005; Quintero et al., 2019).

In this study, we used three modules of HLM with incremental complexity levels: HLM-NoSnow, HLM-FSnow, and HLM-Snow (fig. 3) described in the following subsections. The three modules use the formulation described before in equations (3.1), (3.2), and (3.3), and the channel routing scheme. They only differ in the way snow processes are represented. We use the incremental complexity approach to analyze the relevance of the snow process during the flood event and determine the strengths and limitations of HLM-Snow.

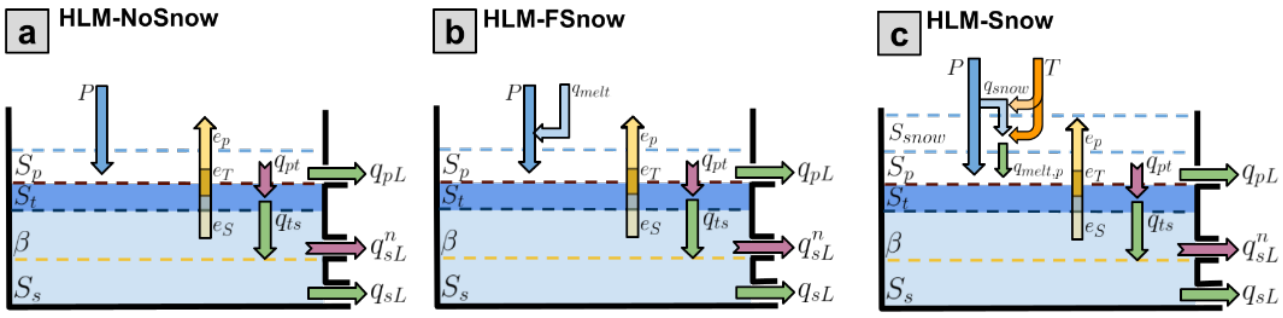

Figure 3.3 Representation of the hillslope processes in HLM. a) HLM-NoSnow, b) HLM-FSnow, and c) HLM-PSnow model scheme. Blue arrows represent precipitacion P and q_{snow}. Yellow arrows represent evapotranspiration from S_p, S_t, and S_s. Green arrows represent linear fluxes between storages. Purple arrows represent nonlinear fluxes.

3.3.1 HLM-NoSnow

HLM-NoSnow is the more straightforward configuration of the three studied modules, and it does not include a representation of snow processes. In this module, the P term in equation (3.1) is the input from liquid precipitation. We used this configuration as a reference to contrast the effects caused by the inclusion of snow processes.

3.3.2 HLM-FSnow

This module includes inputs from liquid precipitation and snowmelt, changing equation (3.1) to

$$\frac{dS_p}{dt} = P + q_{melt} - q_{pL} - q_{pT} - e_p \tag{3.4}$$

where P is liquid precipitation and $\mathrm{q_{melt}}$ is snowmelt forcing (fig. 3.3b). The term $\mathrm{q_{melt}}$ is obtained as the difference of sequential SWE fields. For each time interval with SWE records greater than zero, we estimate the snowmelt rate ($q_{melt}[m \cdot s^{-1}]$) as follows:

$$\mathrm{q_{melt}} = SWE(t) - SWE(t-1) \tag{3.5}$$

where $SWE(t)$ is the snow water equivalent at time interval t estimated with the High-Resolution Rapid Refresh (HRRR) model (Benjamin et al. 2016). In HLM-FSnow, the added complexity is the additional forcing. However, this module does not include a representation of SWE accumulation and melting processes.

3.3.3 HLM-Snow

Following the formulation proposed by Koya et al. (2022), we implemented HLM-Snow to represent snow accumulation and melting processes at the hillslopes. HLM-Snow adds a storage S_{snow} to represent SWE accumulation and uses the temperature threshold method (TT) to represent SWE accumulation and melting. Compared to other TI schemes, TT is one of the most straightforward (Kienzle, 2008), adding just two parameters to the model and keeping its parsimony. In TT, precipitation is assumed to fall as liquid rainfall or SWE depending on a threshold temperature, T_b. The precipitation accumulates in the snow storage S_{snow} as SWE if the air temperature $T(t)$ is lower than T_b. Moreover, snowmelt takes place at S_{snow} with a rate D if $T(t)$ is greater than T_b. The given description can be expressed as follows:

$$q_{snow} = \begin{cases} P & , \quad T < T_b \\ 0 & , \ T \geq T_b \end{cases}, \tag{3.6}$$

In equation (3.5), q_{snow} represents the rainfall P becoming SWE (S_{snow}). In the threshold scheme P becomes equals to zero when $T < T_b$. Moreover, snowmelt, $q_{melt,p}$ subtracts water from S_{SWE} when $T \geq T_b$as follows:

$$q_{melt,p} = min(D \cdot T, S_{SWE}) \tag{3.7}$$

After $q_{melt,p}$ is computed, S_{SWE} is updated as follows:

$$\frac{dS_{SWE}}{dt} = q_{\text{snow}} - \text{q}_{\text{melt,p}} \tag{3.8}$$

The melted water from S_{SWE} updates the ponded storage (S_p) modifying equation (3.1) as follows:

$$\frac{dS_p}{dt} = P - q_{pL} - q_{pT} + q_{melt,p} - e_p, \tag{3.9}$$

Additionally, when $T < T_b$ we assume zero flow from the ponded to the topsoil ($q_{pT} = 0$), representing that the ground is frozen. Figure 3.3c shows the described processes at the hillslope scale.

3.3.4 Model inputs and configuration

The precipitation inputs are MRMS hourly rainfall estimates with a resolution of 4km (Zhang et al., 2016). Monthly average evapotranspiration estimates for the basin were obtained from MODIS (Running et al., 2017). Temperature records were obtained from the Lewis Armstrong station operated by Iowa Mesonet (https://mesonet.agron.iastate.edu/) (blue square in figure 3.1). Observed streamflow data for hydrologic validation was obtained from five USGS gauges located inside the watershed (green dots in figure 3.1). Observed SWE daily fields were obtained from the Snow and Ice Data Center to perform validation of HLM-Snow. The basin river network used in HLM was extracted using NHDplusV2 and a DEM with a resolution of 10 m (USGS, 2017).

The three HLM modules have the same parameterization to represent runoff and routing processes. The runoff generation parameters follow the parameters described in Fonley et al. (2021) and Velasquez et al. (2021). HLM-Snow has two additional parameters: the threshold temperature, T_b set to 5 ℃ and the melting rate, D set to 0.2 $[mm \cdot ℃^{-1} \cdot h^{-1}]$. The selected T_b value is within the range reported by several authors as described by Kienzle, (2008). The melting rate is in the range suggested in the National Engineering Handbook Hydrology of 1.6 to 6 $[mm \cdot ℃^{-1} \cdot day^{-1}]$ (Mockus et al., 2007).

We performed discharge simulations of the selected basin with HLM for the period between March 5 and March 30 of 2019. For the three configurations, we use the same initial conditions setting $S_p = 0.0$, $S_t = 0.05$, and $S_s = 1.6$. In the case of HLM-Snow, we assumed S_{snow} equal to the mean value of the NDSIC accumulated SWE before the event (around 60cm).

3.4 Validation Experiment

We conducted three experiments, each experiment using the HLM modules described before. In the three experiments, MRMS precipitation was used as input. In the experiment using HLM-FSnow, we forced snowmelt estimates from HRRR SWE retrievals (Benjamin et al., 2016). For the experiment with HLM-Snow, we used hourly temperature records from the meteorological station shown in figure 3.1.

We compared the simulated flows to observations at five USGS stations. To measure the performance of each module, we computed the Kling Gupta Efficiency (KGE) (Gupta et al., 2009), Nash Sutcliffe efficiency (NSE) (Nash & Sutcliffe, 1970), peak flow difference (ΔQ_p), and the volume bias (P_{bias}) at each gauge using the following performance indexes:

$$KGE = \sqrt{(1-\mu)^2 + (1-\sigma)^2 + (1-\gamma)^2} \tag{3.10}$$

$$NSE = 1 - \frac{\sum_t (Q_s(t) - Q_o(t))}{\sum_t (Q_o(t) - \overline{Q_o})} \quad (3.11)$$

$$\Delta Q_p = 100 \cdot \frac{Qo_p - Qs_p}{Qo_p} \quad (3.12)$$

$$P_{bias} = 100 \cdot \frac{\sum_{t=1}^{N} (Qo(t) - Qs(t))}{\sum_{t=1}^{N} Qo(t)} \quad (3.13)$$

In equation (3.10), μ, σ, and γ correspond to the mean bias, deviation bias, and correlation, respectively. From equations (3.10) to (3.13), $Qo[m^3 \cdot s^{-1}]$ are the observed flows and $Qs[m^3 \cdot s^{-1}]$ the simulated. In equation (3.13), Qo_p and Qs_p correspond to the observed and simulated peak flows, respectively.

We also validated the HLM-Snow SWE estimates by comparing the average basin values from NSIDC observations. In this comparison, we compute the performance indexes of KGE, NSE, and P_{bias} using a daily timestep. We included a visual comparison of the observed and simulated SWE fields.

3.5 Results

3.5.1 Flow Simulations

Figure 3.4 shows the simulated and observed flows at the five USGS gauges with significant differences among the three HLM realizations. Moreover, the lack of snow accumulation and snowmelt processes limits the HLM-NoSnow performance. HLM-NoSnow (blue line) underestimates peak flows at all the gauges compared with the observations. At the upstream gauges (fig. 3.4a to c), HLM-NoSnow simulates hydrographs with two peaks instead of one peak. Downstream, the hydrographs exhibit a large duration with oscillations underestimating the observations (fig. 3.4d and e). We attribute the described behavior to the

lack of a mechanism to store precipitation as SWE and then release it as runoff. The low performance of HLM-NoSnow was expected; nevertheless, we included it as a reference. Moreover, HLM-FSnow improves the peak flow estimations by using the HRRR ΔSWE as an additional input. Like HLM-NoSnow, HLM-FSnow also exhibits a double peak behavior at the upstream gauges (fig. 3.4a to c) and an early rising limb downstream (fig. 3.4d and e). Additionally, its hydrographs have a third peak after the observed recession. The described early and late peaks lead to an overestimation of the volume and a decrease in the model performance attributed to the HRRR input.

Finally, HLM-Snow exhibits the highest skill in representing the hydrographs, accurately estimating the rising and falling limbs and the peak flows. Besides, the double peaks are eliminated from the simulations. Nevertheless, there are some discrepancies. Upstream, we can appreciate slight peak flows underestimations (fig. 3.4a and b) and overestimations downstream (fig. 3.4e). Moreover, there are also peak flow timing differences upstream (fig. 3.4a to c). HLM-Snow successfully recreates the observed hydrographs using temperature records as an additional input.

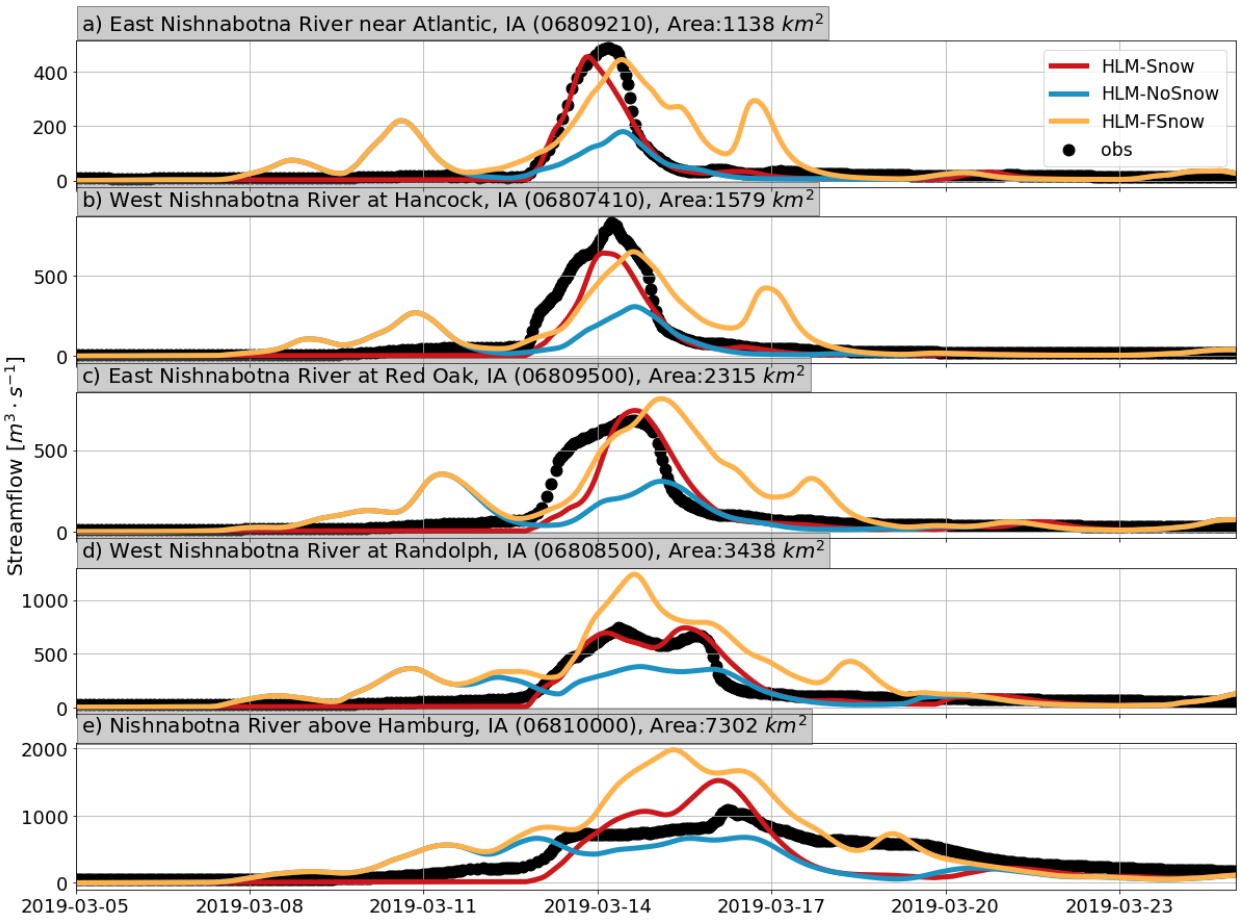

Figure 3.4 HLM flow simulations (color lines) and USGS gauges flow observations (black dots) during the flood of March 2019. Blue lines correspond to HLM-no-snow, yellow to HLM-F-snow, and red to HLM-Snow.

HLM-NoSnow and HLM-FSnow exhibited a double peak signal that is not present in the observations. We attribute the simulated double peak to the presence of two rainfall events over the watershed (fig. 3.2). The first event happened under freezing conditions and the second during melting. Neither HLM-NoSnow nor HLM-FSnow can store precipitation as SWE, so both modules immediately release the rainfall as runoff. HLM-Snow stored the first event precipitation as snow due to the low temperature records. As a result, their discharge simulations exhibited one peak instead of two, agreeing with the observations.

We summarized the described behavior of the streamflow simulations by computing the KGE index and the peak flow difference, ΔQ_p at each gauge (fig. 3.5). HLM-NoSnow (fig. 3.5a and d) has low performance, with KGE values oscillating around 0.2 and with peak flow underestimations around -100%. The results from this setup slightly improve near the outlet with ΔQ_p values to around -20%. On the other hand, HLM-Fsnow (fig. 3.5b and e) exhibits more variability in both indexes. In this case, the KGE (fig. 3.5b) oscillates between -0.1 (downstream) and 0.6 (upstream), with values around 0.1 in the middle. ΔQ_p (fig. 3.5e), indicating accurate peak flow estimations at Atlantic and Red Oak (East Nishnabotna) with values around 10%. However, downstream at Randolph and Hamburg, HLM-FSnow exhibits a significant ΔQ_p overestimation with values around 100%. Finally, in figure 3.5c and f, we present the indexes for HLM-Snow. Here, the KGE values (fig. 3.5c) oscillate between 0.5 and 0.9, where Randolph is the station with the highest score. Regarding ΔQ_p, HLM-Snow has an accurate estimation of the peak flow at Randolph, Red Oak, and Atlantic. However, it underestimates peak flows at Hancock (around 20%) and overestimates at Hamburg (around 30%).

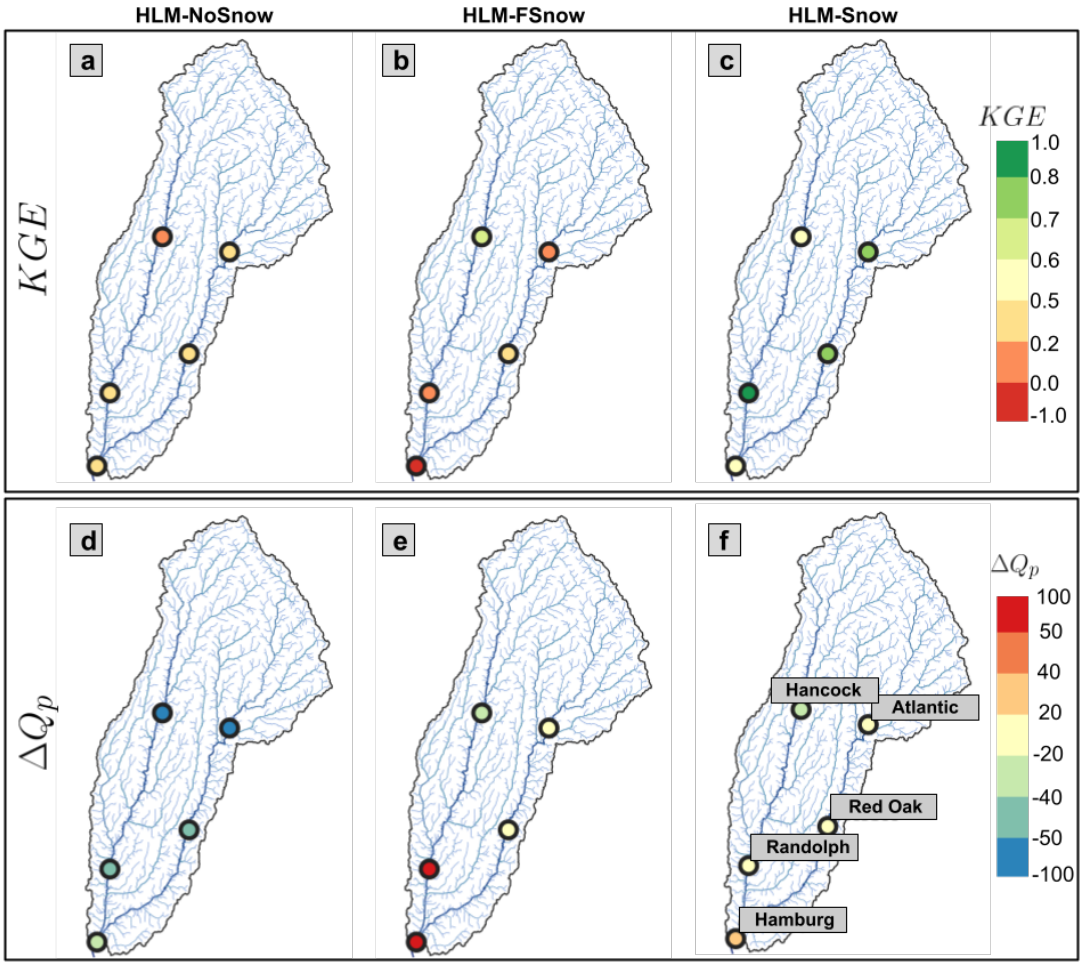

Figure 3.5 KGE (first row) and ΔQ_p (second row) computed for each HLM setup and each USGS gauge. Frames a) to c) correspond to the KGE of HLM-NoSnow, HLM-FSnow, and HLM-Snow, respectively. Frames d) to f) correspond to the ΔQ_p of of HLM-NoSnow, HLM-FSnow, and HLM-Snow, respectively.

Table 3.1 shows the median value of the performance indexes using the five USGS gauges. We included two additional performance indexes in the table: NSE and the volumetric difference,P_{bias}. According to the table, HLM-NoSnow has a KGE and NSE values that are above their respective benchmarks of -0.41 (see Knoben et al., 2019) and 0.0, respectively. However, both can be considered low. Besides, HLM-NoSnow has the most significant bias in ΔQ_p with a value of -49%. On the other hand, HLM-FSnow has lower KGE and NSE values and

the most considerable P_{bias} overestimation (50%). Finally, HLM-Snow exhibits the best performance at all the indexes. Its KGE and NSE values are 0.74 and 0.86, respectively, and its ΔQ_p is 8% indicating a slight overestimation.

Table 3.1 The median value of the performances obtained by HLM-no-snow, HLM-F-snow, and HLM-Snow.

Performance Index	HLM-NoSnow	HLM-Fsnow	HLM-Snow
KGE	**0.31**	**0.18**	**0.74**
NSE	**0.36**	**0.14**	**0.86**
$\boldsymbol{\Delta Q_p}$**[%]**	**-49**	**21**	**8**
$\boldsymbol{P_{bias}}$**[%]**	**-30**	**50**	**22**

3.5.2 SWE estimates

We compared the basin average SWE observed and simulated with HLM-Snow daily. We obtained an NSE of 0.87, a KGE of 0.93, and a P_{bias} of -1.25%. According to the described indexes and figure 3.6a, HLM-Snow reasonably estimates the SWE oscillations for the simulated event. Nevertheless, there are spatial distribution differences. In figures 3.6b and 3.6c, we present four SWE snapshots of the observed and simulated SWE. We choose the snapshots to compare the initial SWE storage and its initial accumulation (Dates 1 and 2), SWE values after the precipitation event of March 9 (see fig. 3.2b) (Date 3), and SWE during the flood event around March 13 (Date 4). Besides, Date 4 also coincides with the snowmelt of the SWE. In Date 1, the observations suggest higher SWE accumulation near the watershed outlet, while

simulations indicate a homogeneous distribution. In Date 2, SWE observations and simulations suggest higher accumulations downstream and upstream over the northwest side of the watershed. Date 3 indicates increased SWE accumulations over the same regions with lower accumulations downstream. To this date, the simulated SWE exhibits higher accumulations towards the North-East region of the watershed. Finally, during Date 4, the simulated SWE overestimates the observed one. This overestimation is more remarkable on the East side of the watershed.

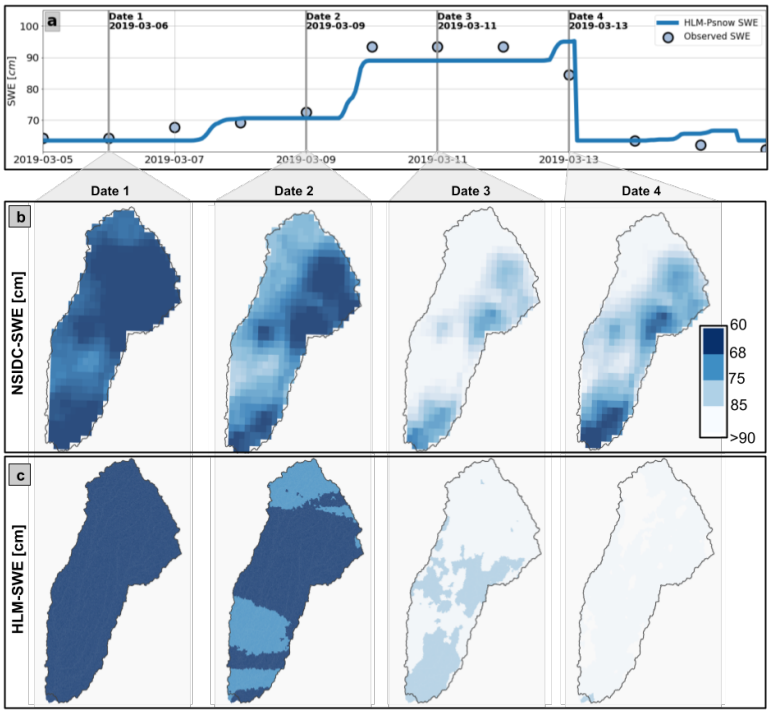

Figure 3.6 Mean SWE accumulation (in cm) during the March 2019 flood event. A) HLM (blue line) and NSIDC (dots) mean SWE over the Nishnabotna watershed. B) NSIDC SWE estimations for March 6, 9, 11, and 13. C) HLM-Fsnow SWE estimations for March 6, 9, 11, and 13.

The HLM-Snow SWE estimations are more spatially homogeneous than the observations. The homogeneity may be attributed to the threshold method to melt snow and to

the use of only one weather station to represent temperatures. Depending on the temperature, the threshold method converts all the precipitation into snow or melts it at a constant rate D. This behavior can be improved by using an s-shaped function (Kienzle, 2008). Other improvements include the hillslopes aspect (Cazorzi & Dalla Fontana, 1996; Zaramella et al. 2019) and the combination of the temperature index with simplified radiation balance (Cazorzi & Dalla Fontana, 1996; Follum et al. 2019; Kustas & Rango, 1994). We are aware that the described modifications may lead to better results. However, they require additional complexities and parameters that may be addressed in future work around HLM that may consider our results as a benchmark.

3.6 Conclusions

We tested a simple approach to simulate snow accumulation and melting processes in HLM using a new HLM-Snow module. We checked the ability of the new module to reproduce the runoff generation processes that took place in March 2019 in the Nishnabotna River, Iowa. We also examined the effects of using a module that does not include representation of snow processes and a module that uses snowmelt data as forcing. The obtained results are promising; however, our conclusions are limited since we only tested our model for a single watershed and one flood event. We validated our results by comparing simulated flows with observations at five USGS gauges. Also, we compared, simulated, and observed SWE in the watershed. Future work may expand the validation to more gauges and events covering several years of records.

Our results suggest that using the TI model allows better runoff generation and SWE estimation representation. There are still limitations in the representation of the spatial SWE accumulations that can be attributed to the fact that we used one weather station to describe the temperature of the domain and the limitations of the threshold approach in the TI model. The

temperature-related limitation may become more relevant in larger domains. In this case, we assumed a spatially uniform temperature to keep the model's simplicity. Future work may include distributed information and expressions that correct surface temperature based on the elevation (Daly et al., 2000; Franz & Karsten, 2013) and the hillslope orientation (Cazorzi & Dalla Fontana, 1996), solar radiation, and land use. Also, using more complex TI models that involve S-shaped functions (Kienzle, 2008) can be explored. The results presented in this study provide the baseline for a regional benchmark for future developments of models including snow processes.

We addressed the relevance of including snow processes in the model by comparing the HLM-Snow and HLM-FSnow setups. Including snowmelt as a force in HLM-FSnow corrected volume underestimations and improved peak flow estimations compared to not including any information about snow. However, HLM-FSnow exhibited flow oscillations not present in the observations, while HLM-Snow produced better flow simulations without artificial oscillations. We attribute the HLM-FSnow drawbacks to errors in the snowmelt estimates and how these are forced in HLM. Nevertheless, we cannot conclude that representing the processes is better than including snowmelt as a force since our results are limited to one basin and one flood event. We will expand our work in this direction since including snowmelt as a force could still be a plausible option.

Using a TI model can provide accurate forecasts of snow-related floods during the early spring. Our results suggest that including the snow storage in the hydrologic model using the threshold method is a simple yet powerful tool for providing flood forecasts. Compared with HLM-NoSnow, HLM-Snow requires only adding one additional forcing of temperature.

Considering this, HLM-Snow is suitable to make forecasts at a regional level, becoming a complement to the current flood warning system of Iowa.

Chapter 4 Technology transfer – HLM in Elkhorn River

The latest advances include the implementation of the HLM model for the Elkhorn River (in Nebraska) and the installation of asynch (the container of HLM) in the computers of the University of Nebraska. With around 18,000km^2, the Elkhorn River is one of the largest and most important watersheds in the state of Nebraska. Using HLM and Stage IV QPEs as the main input, we are able to reproduce some of the most important streamflow fluctuations observed by eight UGSG gauges between 2008 and 2020. However, our results also highlight the need for improvements in the model performance to obtain a better representation at some stations. On the other hand, we achieve a straightforward installation of HLM in a computer of the University of Nebraska using Docker. This report details both items, the implementation of HLM in the Elkhorn River and the steps to install asynch in a machine using Windows 10.

4.1 HLM setup

Using HLM and Stage IV QPEs, we simulated the streamflow in the Elkhorn River between 2008 and 2020. We setup HLM by processing a DEM with a resolution of 16m in TauDEM (Tarboron 1997; Tarboton and Bras 1989). From TauDEM we obtained the definition of the links and their respective hillslopes both saved in the ESRI shape format. Then, using the Python package **ifis_tools** (https://github.com/nicolas998/ifis_tools) and the shapefile containing the river network (fig. 4.1), we extracted the required files for the model. Finally, we created the binary rainfall files containing the Stage IV QPEs and then we ran the HLM model. The results were compared with the records of the eight USGS gauges shown in figure 4.1.

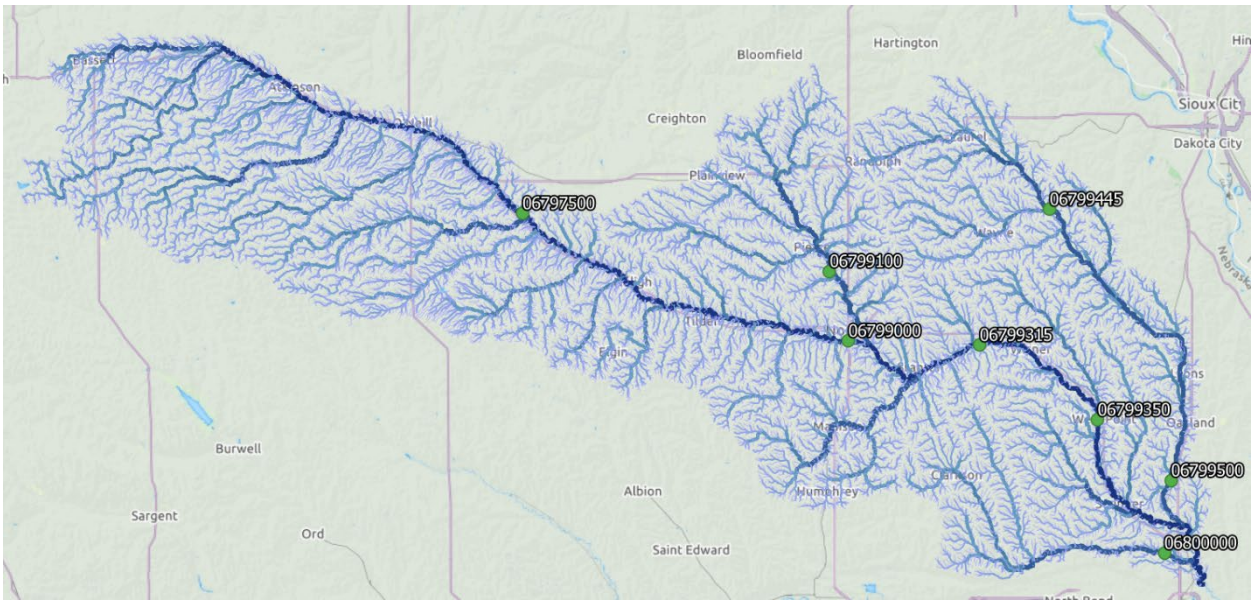

Figure 4.1 Elkhorn Watershed network description (blue lines) and UGSG gauges localization (green).

The following section details the steps for setup and running the HLM model for the Elkhorn Watershed.

4.2 DEM processing

To process the DEM, we use the TauDEM software which can be downloaded from the following link:

https://hydrology.usu.edu/taudem/taudem5/

The DEM was downloaded from the USGS 3D elevation Program (3DEP). The program provides DEMs for the US with a resolution of ~8[m]. For the current work, we resample the DEM to a resolution of 16 [m]. Processing is performed in a HPC computer using a total of 224 computational nodes. In our case the DEM process took about two hours. The total processing time depends on the resolution of the DEM, the number of processors, and the processor speed.

To perform the processing, we import the TauDEM and the GDAL (GDAL/OGR contributors 2021) packages as follows:

```
module load taudem
module load gdal/2.1.3_python-3.7.0
```

During the process, TauDEM follows the network delineation provided by NHD+ for the Elkhorn Watershed. the NHD+ network is burned into the DEM using the following script:

```
gdal_calc.py --calc=A*0 --outfile=network.tif -A dem16m.tif --overwrite
gdal_rasterize -burn 1 network.shp network.tif
gdal_calc.py --calc=A-B*10 --outfile=dem_burn.tif
    -A dem16m.tif -B network.tif --overwrite --type 'Int16'
```

The first line of the script creates the void raster file **network.tif** using the **dem16m.tif** raster as a base. The second line populates the **network.tif** file with values equal to 1 where there are river segments. Finally, the third line burns the **network.tif** file into the **dem16m.tif** file.

After the burn process, we use TauDEM to remove pits, extract flow directions, compute upstream areas, extract network, and delineate hillslopes. This process is done with the following code:

```
# Pitremove
mpiexec -n 224 pitremove -z dem_burn.tif -fel demfel.tif
# D8 flow directions
mpiexec -n 224 d8flowdir -p demp.tif -sd8 demsd8.tif -fel demfel.tif
# Contributing area
mpiexec -n 224 aread8 -p demp.tif -ad8 demad8.tif -o outlet.shp -nc
# Grid Network
mpiexec -n 224 gridnet -p demp.tif -gord demgord.tif -plen demplen.tif -
tlen demtlen.tif
# hillslope Threshold
npixels=590
mpiexec -n 224 threshold -ssa demad8.tif -src demsrc.tif -thresh $npixels
#streamnet
mpiexec -n 224 streamnet -fel demfel.tif -p demp.tif -ad8 demad8.tif
    -src demsrc.tif -ord demord.tif -tree demtree.txt
    -coord demcoord.txt -net demnet.shp -w demw.tif
#Get the hillslopes from the demw.tif file
gdal_polygonize.py demw.tif hills.shp -8 -b 1
    -f "ESRI Shapefile" hills DN
```

Executing the described code, we obtained the **dement.shp** and the **hills.shp** maps which are the basic information required to set up HLM. The details of the used TauDEM functions can be found in the following link:

https://hydrology.usu.edu/taudem/taudem5/TauDEM53CommandLineGuide.pdf

4.2.1 HLM setup

HLM requires the following files to run:

- **Topology file:** Describes the connection between channel links. In it, each link could have no parent links or a couple of parents.
- **Parameter file:** Describes the upstream area, the length, and the hillslope area of each link. It can also contain additional hydrological parameters depending on the model configuration.
- **Global file:** Determines how the model is going to be run. It contains the start and end date of the simulation. Also, it has set up options for the forces of the model, where to write the results, and the formats to use.

Additionally, HLM requires files to describe the forces and the control points. A detailed account of the required files can be found in the following link:

https://asynch.readthedocs.io/en/latest/index.html

We have developed the Python package **ifis_tools** that contains a collection of functions to obtain the required files using the result **dement.shp**. The package can be downloaded from the following GitHub repository:

https://github.com/nicolas998/ifis_tools

From **ifis_tools** we use the **from_taudem** package that contains the Python class to write all the required files for the HLM execution. In the following code we detail the steps:

```
#Import ifis tools
from ifis_tools import from_taudem as ft
#Read the TauDEM network as a class of ft
net = ft.network('maps/demnet.shp')
#Set the parameters (for model 608 by default)
net.get_prm()
net.set_prm_for_model()
#Write the topology (rvr) and param (prm) files
net.write_prm('elk_param.prm')
net.write_rvr('elk_topo.rvr')
```

In the code, first we import the package as **ft.** Then we use **dement.sh** to define the parameters of the **ft.network** class that contains the functions used to define the files. After defining the network class, we setup the parameters. In this case, the parameters are set by default for the model 608 (non-linear subsurface model) detailed in (Fonley et al. 2021). Finally, we write the topology (**.rvr**) and the parameters (**.prm**) files.

After defining the parameters and the topology files, we write the global files and the run files for each year. This procedure is shown in the following code:

```
for y in range(2008,2020):
    global_name = 'elk_global_'+str(y)+'.gbl'
    net.write_Global(global_name,
        model_uid=608,
        date1=str(y)+'-02-01 00:00', # Sim Start date
        date2=str(y)+'-12-27 00:00', # Sim End data
        rvrFile='elk_topo.rvr', # Topology file
        rvrLink='', # No out link
        prmFile='elk_param.prm', # Param file
        initialFile='ini'+str(y)+'.uini', # initial conditions file
        rainPath = 'rainfall/'+str(y)+'/', # Binary rainfall path
        controlFile='control.sav', # Control points file
        snapType = 0, # No snapshots
        nComponents = 5, # Number of states to print
        Components = [0,1,2,3,4] # States to print
        datResults='qsim'+str(y)+'.dat', # where to save results
        evpFile='evp_'+str(y)+'.mon') #Evaporation
    #Updates the dictionary with the parameters to setup the run
    Proc={global_name:{'nproc': 14, 'secondplane': False}}
    #Writes the runfile with all the comands
    net.write_runfile('run_'+str(y)+'.sh',
        Proc, jobName='xx_'+str(y), nCores= 14)
```

In it, we write a global file and a run file for each year. The **net.write_Global** function takes several arguments that populate the file. In this case, we leave comments in the function of each argument. After writing the global file, the code uses the **net.write_runfile** to write the executable file.

4.2.2 Rainfall setup

After defining the HLM files, we define the rainfall look up table for Stage IV rainfall using the following code:

```
ft.rainfall_raster_ranks(path_in = 'stageIV_raster.tif',
                         path_out = 'rain_ranks.tif')
net.get_rainfall_lookup('rain_ranks.tif')
net.rain_ranks.to_file('ranked.shp')
```

In the code we first read an example raster containing StageIV QPEs and from it we define a ranked raster. The ranked raster is then used to define a look up table that is written in an ESRI shapefile.

Finally, we write the binary rainfall files using the following code:

```
#Read the rainfall data list and extract the dates
list_rain = glob.glob('stage4/'+str(year)+'*.tif')
list_rain.sort()
dates = [pd.to_datetime(get_date(date), format='%Y%m%d%H') for date in list_rain]
dates.sort()
df = pd.DataFrame(list_rain, dates)
#Itereate through the dates and write the binaries
for date in df[year].index:
    #Get the unix time
    unix = aux.__datetime2unix__(date)
    #Reads and convert the rainfall
    rain = net.rain2links(rain = None, path_rain = df.loc[date].values[0])
    count = rain[rain>1].size
    #Save the binary data
    if rain[rain > 1].size > 10:
        ft.saveBin(rain.index.values,
                   rain.values,count,
                   'rainfall/'+str(year)+'/'+str(unix))
```

In it, we read the list of Stage IV files and then iterate through them reading the QPEs of each time step. At each iteration, we assign the QPEs values to each hillslope and save the results in a binary file with the Unix time as the name.

4.3 Modeling results

Using HLM, we simulate streamflow records for the Elkhorn River between 2008 and 2020. According to our results, the model captures the main oscillations in most of the stations. Moreover, it simulates some of the most significant peak flows observed in the same period. However, there are still shortcomings in its performance. We attribute the shortcomings to abrupt changes in the landscape, the dry weather. Also, the model must include snow and frozen ground processes to improve its performance over the watershed.

To analyze the model's performance, we use the KGE index, the peak magnitude difference (ΔQ_p), and the time to peak difference (Δt_p). The KGE index measures the overall performance

using the correlation (γ), the mean ratio (α), and the standard deviation ratio (σ) as follows:

$$KGE = \sqrt{(\gamma - 1)^2 + (1 - \mu)^2 + (1 - \sigma)^2} \quad (4.1)$$

The index oscillates between $-\infty$ and 1 where negative values are considered simulations with issues, and positive values are acceptable simulations. The peak magnitude difference compares the simulated annual peak flow ($Q_{p,s}$) with the observed ($Q_{p,o}$) using the following equation:

$$\Delta Q_p = \frac{Q_{p,s} - Q_{p,o}}{Q_{p,o}} \quad (4.2)$$

In it, negative values indicate under estimation of the peak flows, and positive values represent over estimations. ΔQ_p values near zero correspond to a good estimation of the peak flow. Finally, we measure the time to peak difference (in hours) between the arrival of the observed peak ($t_{p,o}$) and the simulated one ($t_{p,s}$) with the following equation:

$$\Delta t_p = t_{p,o} - t_{p,s} \quad (4.3)$$

Negative values of Δt_p represent an early simulation of the peak flow, while positive values represent a late simulation of it. In figure 4.2 we show the result of the three indexes. According to figure 4.2a, the model has an acceptable KGE performance in some cases. However, the most frequent KGE value is 0.2 which means that there is room to improve the

model. On the other hand, the simulations have an overall good accuracy representing the peaks, with some cases going towards under-estimations (fig. 4.2b). Finally, there are issues representing the time to peak (fig. 4.2c). The Δt_p frequency of values near 0 is low, and most of the cases correspond to positive Δt_p values representing late peak flows.

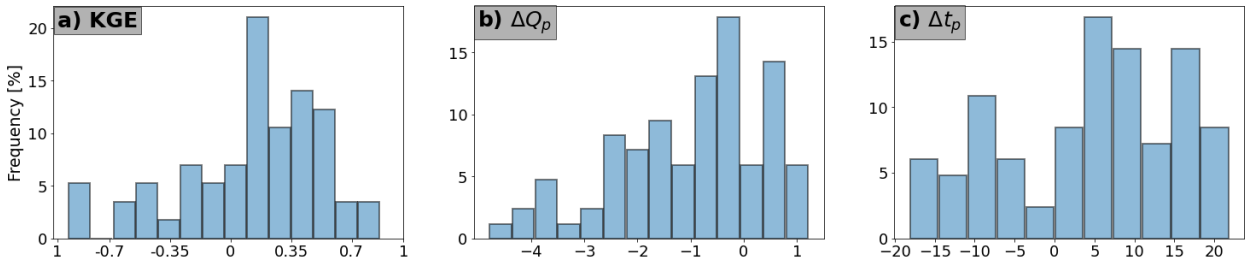

Figure 4.2 Summary of the yearly HLM performance at the USGS stations shown in figure 1.1. a) Kling Gupta efficiency index (KGE). b) Peak flow difference (ΔQ_p), c) Time to peak difference (Δt_p).

There are only two significant peaks in the 2002-2020 records. One in 2010 (fig. 4.3), and the other in 2019. The 2010 peak happened between May and Jun, the 2019 in March. The 2010 event is well explained by the Stage IV QPEs, while the 2019 requires to consider snow melt and frozen ground information. In figure 4.3, we show the simulation results for the 2010 case. According to it, HLM captures some features of the 2010 hydrograph. HLM over-estimates the peakflow over the main channel. Over estimations are observed upstream at gauge 06796500 and downstream at gauge 06799350. Moreover, the model has a better peak flow estimation in the tributaries such as gauge 06797500 and 06799445.

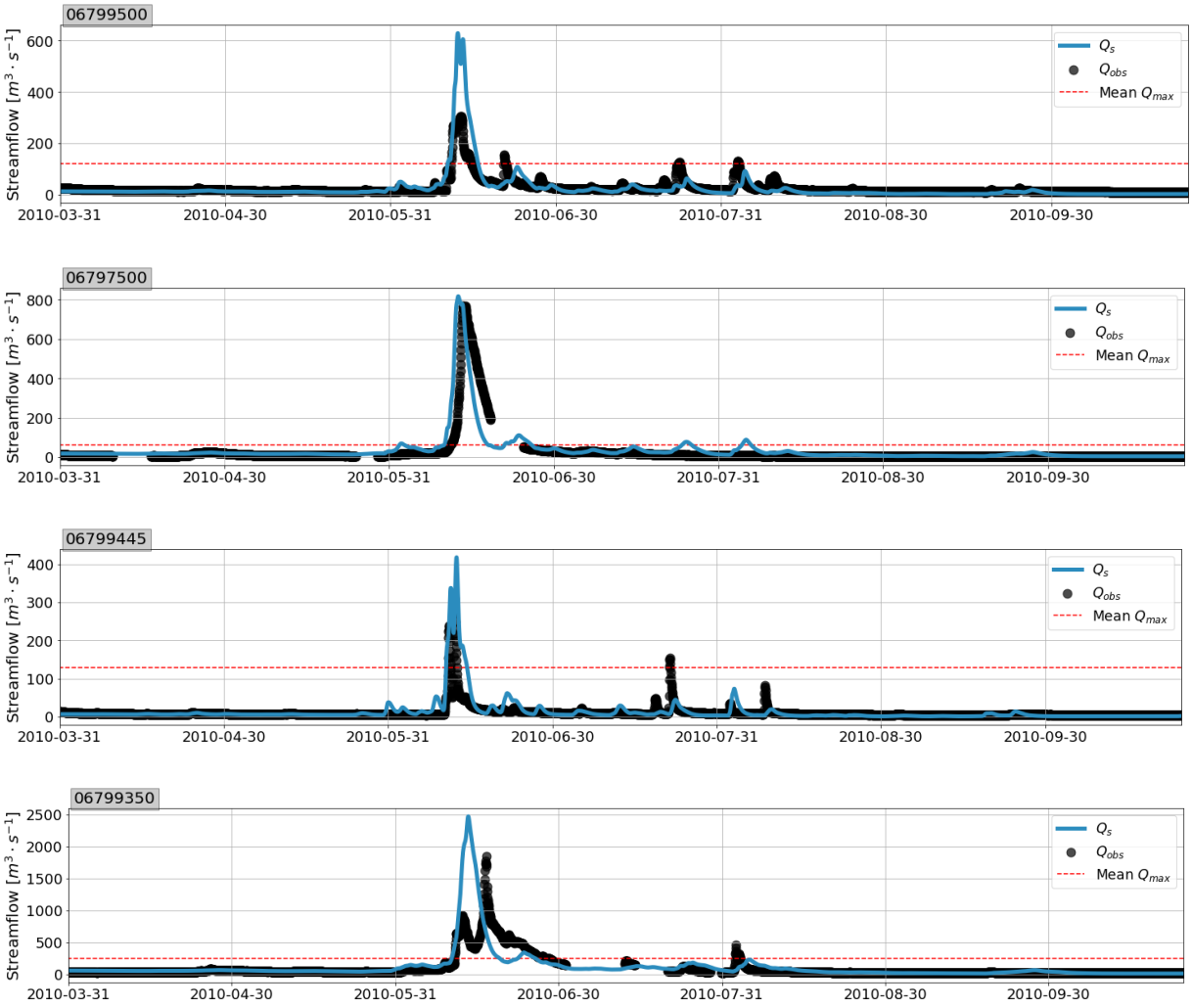

Figure 4.3 Simulated streamflow for the 2010 event. The results correspond to two gauges over the main channel 06799500 and 06799350, and two tributaries: 06797500 and 06799445.

Chapter 5 Conclusions

An important aspect in providing a safe, efficient, and effective transportation system is anticipating natural hazards that can lead to road closures. Extreme floods can lead to bridge overtopping and/or compromising the structural integrity of river overpasses, including box culverts. The flood forecasting model and information system proposed here provides a tool to anticipate potential hazardous situations related to floods. It would allow time for the activation of action plans to minimize the impact on the overall transportation system. The forecasting model can be used in real time to anticipate floods and to look at past flooding scenarios to determine if all the actions taken were appropriate or can be improved. Our forecasting system will contribute to improving safety and minimizing risk associated with increasing multi-modal freight movements on the U.S. surface transportation system by *enhancing safety* and providing warning of potential road closures.

As part of this project, we have provided a prototype forecasting web platform with four specific innovations: 1) Forecasts at critical river/road intersections, 2) Spatial animated maps of flood evolution into the future, and 3) a measure of forecast accuracy at the newly incorporated forecast bridges. Our developments give us confidence that we can continue moving forward in developing a forecasting system that is transferable to other locations in the Midwest. As floods continue to be the most costly disaster in the nation, it becomes critical that tools are develop to better predict them.

References

Ayalew, T., Krajewski, W., Mantilla, R. (2013). Exploring the Effect of Reservoir Storage on Peak Discharge Frequency. J. Hydrol. Eng. 18, 1697–1708. https://doi.org/10.1061/(ASCE)HE.1943-5584.0000721

Ayalew, T. B., Krajewski, W. F. & Mantilla, R. (2014), "Connecting the power-law scaling structure of peak-discharges to spatially variable rainfall and catchment physical properties," *Advances in Water Resources*, vol. 71, pp. 32–43.

Ayalew, T., Krajewski, W., Mantilla, R. (2015). Insights into Expected Changes in Regulated Flood Frequencies due to the Spatial Configuration of Flood Retention Ponds. J. Hydrol. Eng. 20, 4015010–4015010. https://doi.org/10.1061/(ASCE)HE.1943-5584.0001173

Ayalew, T.B., Krajewski, W.F., Mantilla, R., Wright, D.B., Small, S.J. (2017). Effect of Spatially Distributed Small Dams on Flood Frequency: Insights from the Soap Creek Watershed. J. Hydrol. Eng. 22. https://doi.org/10.1061/(ASCE)HE.1943-5584.0001513

Bhatt, Gopal et al. (2014) (https://pdfs.semanticscholar.org/715d/d10d9c3a356ef61c576bf1afb8f704bfd0ca.pdf)

Choi, C. C., Constantinescu, G., & Mantilla, R. (2015). Implementation of a Hydraulic Routing Model for Dendritic Networks with Offline Coupling to a Distributed Hydrological Model. Journal of Hydrologic Engineering.

Cunha, L.K., P.V. Mandapaka, W.F. Krajewski, R. Mantilla, and A.A. Bradley. (2012). Impact of radar rainfall error structure on estimated flood magnitude across scales: An investigation based on a parsimonious distributed hydrological model, Water Resources Research, 48(10), W10515.

Gupta VK, Waymire EC. (1998). Spatial Variability and Scale Invariance in Hydrologic Regionalization. In Scale Dependence and Scale Invariance in Hydrology, Sposito G

(ed.).Cambridge University Press: Cambridge; 88–135. DOI: 10.1017/CBO9780511551864.005

Krajewski, W.F., A. Kruger, S. Singh, B.-C. Seo and J.A. Smith, Hydro-NEXRAD-2: Real-time access to customized radar-rainfall for hydrologic applications, Journal of Hydroinformatics, 15(2), 580-590, 2013.

Krajewski, W. F., Ceynar, D., Demir, I., Goska, R., Kruger, A., Langel, C., Young, N. C. (2017). Real-time flood forecasting and information system for the state of Iowa. Bulletin of the American Meteorological Society, 98(3), 539–554. https://doi.org/10.1175/BAMS-D-15-00243.1

Laplante and Phillip. (2007). What Every Engineer Should Know About Software Engineering.

Mantilla, R. and V.K. Gupta, (2005). A GIS numerical framework to study the process basis of scaling statistics in river networks, IEEE Geoscience and Remote Sensing Letters, 2(4), 404-408.

Mantilla, R., Gupta, V. K. & Mesa, O. J. (2006).Role of coupled flow dynamics and real network Mantilla, R. (2007). Physical basis of statistical self-similarity in peak flows on random self-similar networks, PhD dissertation, University of Colorado, Boulder.

Mantilla, R. (2007). Physical basis of statistical self-similarity in peak flows on random self-similar networks, PhD dissertation, University of Colorado, Boulder.

Mantilla, R., Cunha, L. K., Krajewski, W. F., Small, S. J., Jay, L. O., Fonley, M. & Curtu, R. (2012), “Simulation of a Distributed Flood Control System using a Parallel Asynchronous Solver for Systems of ODEs,” Applied Simulation and Modeling, June, Napoli, Italy, http://dx.doi.org/10.2316/P.2012.776-042.

Moser, B. A., Gallus, Jr., W. A., & Mantilla, R. (2015). An Initial Assessment of Radar Data Assimilation on Warm Season Rainfall Forecasts for Use in Hydrologic Models. Weather and Forecasting, vol. 30, no. 6, pp. 1491-1520.

Newton, D.W., and Herrin, J.C. (1982). Assessment of commonly used methods of estimating flood frequency, Transportation Research Record, 896, 10-30.

NWS – National Weather Service. Climate Reports: August 2016 & summer 2016. [Online] 2016. http://www.weather.gov/dvn/Climate_Monthly_08_2016 (accessed Mar 21, 2017).

Seo, B.-C., W.F. Krajewski and L.K. Cunha, Impact of radar-rainfall uncertainty on hydrologic response for the eastern Iowa flood of 2008, Water Resources Research, 49, 2747–2764, 2013.

Small, S. J., Jay, L. O., Mantilla, R., Curtu, R., Cunha, L. K., Fonley, M., & Krajewski, W. F. (2013). An asynchronous solver for systems of ODEs linked by a directed tree structure. Advances in Water Resources, 53, 23–32. https://doi.org/10.1016/j.advwatres.2012.10.011

State of Iowa. (2009). https://www.legis.iowa.gov/docs/ico/section/466C.1.pdf

Ali, J., Khan, R., Ahmad, N., & Maqsood, I. (2012). Random Forests and Decision Trees. *International Journal of Computer Science Issues*, *9*(5), 272–278.

Andrews, F. T., Croke, B. F. W., & Jakeman, A. J. (2011). An open software environment for hydrological model assessment and development. *Environmental Modelling and Software*, *26*(10), 1171–1185. https://doi.org/10.1016/j.envsoft.2011.04.006

Breiman, L. (1994). Bagging predictors: Technical Report No. 421. *Department of Statistics University of California*, *2*, 19. https://www.stat.berkeley.edu/~breiman/bagging.pdf

Devia, G. K., Ganasri, B. P., & Dwarakish, G. S. (2015). A Review on Hydrological Models. *Aquatic Procedia*, *4*(Icwrcoe), 1001–1007. https://doi.org/10.1016/j.aqpro.2015.02.126

Dinnes, D., Jaynes, D. B., Kaspar, T., & Hatfield, J. (2002). Nitrogen Management Strategies to Reduce Nitrate Leaching in Tile-Drained Midwestern Soils PhD research project on soil desinfection with ozone in substitution to synthetic pesticides View project Enhancing Soil Begins with Soil Biology and a Stable Soil Microclimate View project. *Article in Agronomy Journal*, *December 2014*. https://doi.org/10.2134/agronj2002.0153

Goodall, J. L., Horsburgh, J. S., Whiteaker, T. L., Maidment, D. R., & Zaslavsky, I. (2008). A first approach to web services for the National Water Information System. *Environmental Modelling and Software*, *23*(4), 404–411. https://doi.org/10.1016/j.envsoft.2007.01.005

Gray, J., Chaudhuri, S., Bosworth, A., Layman, A., Reichart, D., Venkatrao, M., Pellow, F., & Pirahesh, H. (1997). Data cube: A relational aggregation operator generalizing group-by, cross-tab, and sub-totals. *Data Mining and Knowledge Discovery*, *1*(1), 29–53. https://doi.org/10.1023/A:1009726021843

Gupta, H. V., Kling, H., Yilmaz, K. K., & Martinez, G. F. (2009). Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling. *Journal of Hydrology*, *377*(1–2), 80–91. https://doi.org/10.1016/j.jhydrol.2009.08.003

Holden, J., Chapman, P. J., Kay, P., Mitchell, G., & Pitts, C. S. (2019). Surface water quality. *Water Resources*, 99–150. https://doi.org/10.4324/9780429448270-4

Kam Ho, T. (1995). Random Decision Forests Perceptron training. *AT&T Bell Laboratories*.

Klaus, J., & Zehe, E. (2010). Modelling rapid flow response of a tile-drained field site using a 2D physically based model: Assessment of "equifinal" model setups. *Hydrological Processes*, *24*(12), 1595–1609. https://doi.org/10.1002/hyp.7687

Landerer, F. W., & Swenson, S. C. (2012). Accuracy of scaled GRACE terrestrial water storage

estimates. *Water Resources Research*, *48*(4), 1–11. https://doi.org/10.1029/2011WR011453

Li, H., Sivapalan, M., Tian, F., & Liu, D. (2010). Water and nutrient balances in a large tile-drained agricultural catchment: A distributed modeling study. *Hydrology and Earth System Sciences*, *14*(11), 2259–2275. https://doi.org/10.5194/hess-14-2259-2010

Loritz, R., Hassler, S. K., Jackisch, C., Allroggen, N., van Schaik, L., Wienhöfer, J., & Zehe, E. (2016). Picturing and modelling catchments by representative hillslopes. *Hydrology and Earth System Sciences Discussions*, 1–56. https://doi.org/10.5194/hess-2016-307

Maidment, D. R. (2002). Arc Hydro: GIS for water resources. In *Agriculture*.

Moussa, R., Chahinian, N., & Montpellier, H. (2009). Comparison of different multi-objective calibration criteria using a conceptual rainfall-runoff model of flood events. *Earth*, 519–535.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2012). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning*, *12*, 2825--2830.

Quinlan, J. R. (1986). Induction Fo Decion Trees. *Machine Learning*, *1*, 81–106.

Sasaki, Y. (2007). The truth of the F-measure. *Teach Tutor Mater*, 1–5. http://www.cs.odu.edu/~mukka/cs795sum09dm/Lecturenotes/Day3/F-measure-YS-26Oct07.pdf

Schilling, K. E., & Helmers, M. (2008). *Effects of subsurface drainage tiles on streamflow in Iowa agricultural watersheds : Exploratory hydrograph analysis*. *4506*(May), 4497–4506. https://doi.org/10.1002/hyp

Sivapalan, M. (2018). From engineering hydrology to Earth system science: Milestones in the

transformation of hydrologic science. *Hydrology and Earth System Sciences*, *22*(3), 1665–1693. https://doi.org/10.5194/hess-22-1665-2018

Tallaksen, L. M. (1995). A review of baseflow recession analysis. *Journal of Hydrology*, *165*(1–4), 349–370. https://doi.org/10.1016/0022-1694(94)02540-R