Comparing NFIE RAPID Modelled Discharge with In-Situ Measured Discharge in the Lower Colorado River, Texas.

Introduction

Forecasting the timing and magnitude of flooding is of vital importance to the preservation of lives and property, particularly in flood-prone regions such as central Texas. RAPID (Rapid Application for Parallel computation of Discharge) is a river routing model that can be applied to a stream network given information about stream connectivity of the stream segments within the network. NFIE (National Flood Interoperability Experiment) has coupled land and atmospheric based hydrologic models with the RAPID system to produce flood forecasts for each stream segment in the United States. Hourly discharge is forward modelled for 14 hours from the starting time of the model and a new model is generated every 3 hours. The water budget modelling parameters that feed into consist of a suite of land based hydrologic models called Noah-MP that models a variety of physical factors, including evapotranspiration, precipitation, runoff, groundwater recharge, and many others (figure 1). Perhaps one of the most important model parameters for forward modelling is precipitation forecasting. The NFIE RAPID modelling system uses NOAA HRRR (High Resolution Rapid Refresh) as its precipitation modelling parameter. HRRR is "a real-time 3km resolution, hourly updated, cloud-resolving, convection-allowing atmospheric model, initialized by 3km grids with 3km radar assimilation" (NOAA 2015).

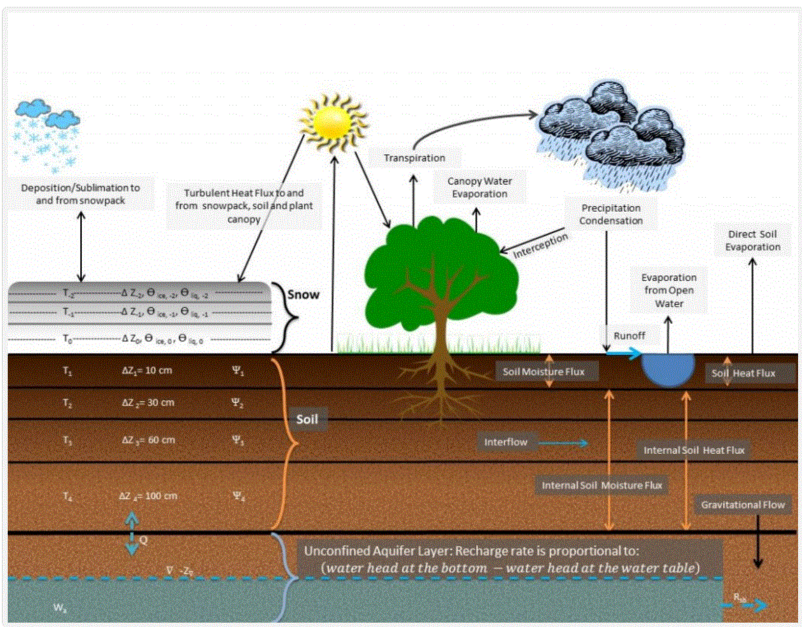

Figure 1-*Noah-MP model parameters (JSG 2015)*

The success of these models at predicting flow conditions is ultimately dependent upon field based "ground-truthing" to calibrate the model under differing hydrologic conditions. Of particular importance is in-situ estimates of discharge, such as those generated from USGS gages. Often, however, these gages are spaced too far apart to assist in model calibration on a small spatial scale. This investigation compares flow predictions from the NFIE RAPID model outputs with in-situ data collected at two field sites along the Lower Colorado River (LCR) between Austin and Bastrop, Texas. The field sites lie between two USGS gages and thus increase the spatial resolution of measured discharge along the river reach.

Methods

Study reach

The study area covers an approximately 90km reach of the Lower Colorado River, bounded upstream by the Longhorn Dam in Austin and the downstream by the city of Bastrop (figure 2). Mean discharge at the USGS 183 gage just downstream of longhorn dam is 47.85m^3/s and the total area drained is 11,024 km^2 (USGS 2015).We have been monitoring a series of four different study sites along this river reach over the last several years to investigate the impacts of natural and controlled flood pulses and how they impact the storage, temperature, and chemistry of surface waters and ground waters as they travel downstream. For the present investigation, data from in-stream pressure transducers installed at two sites (Hornsby Bend and Webberville park) were used to create hydrographs.

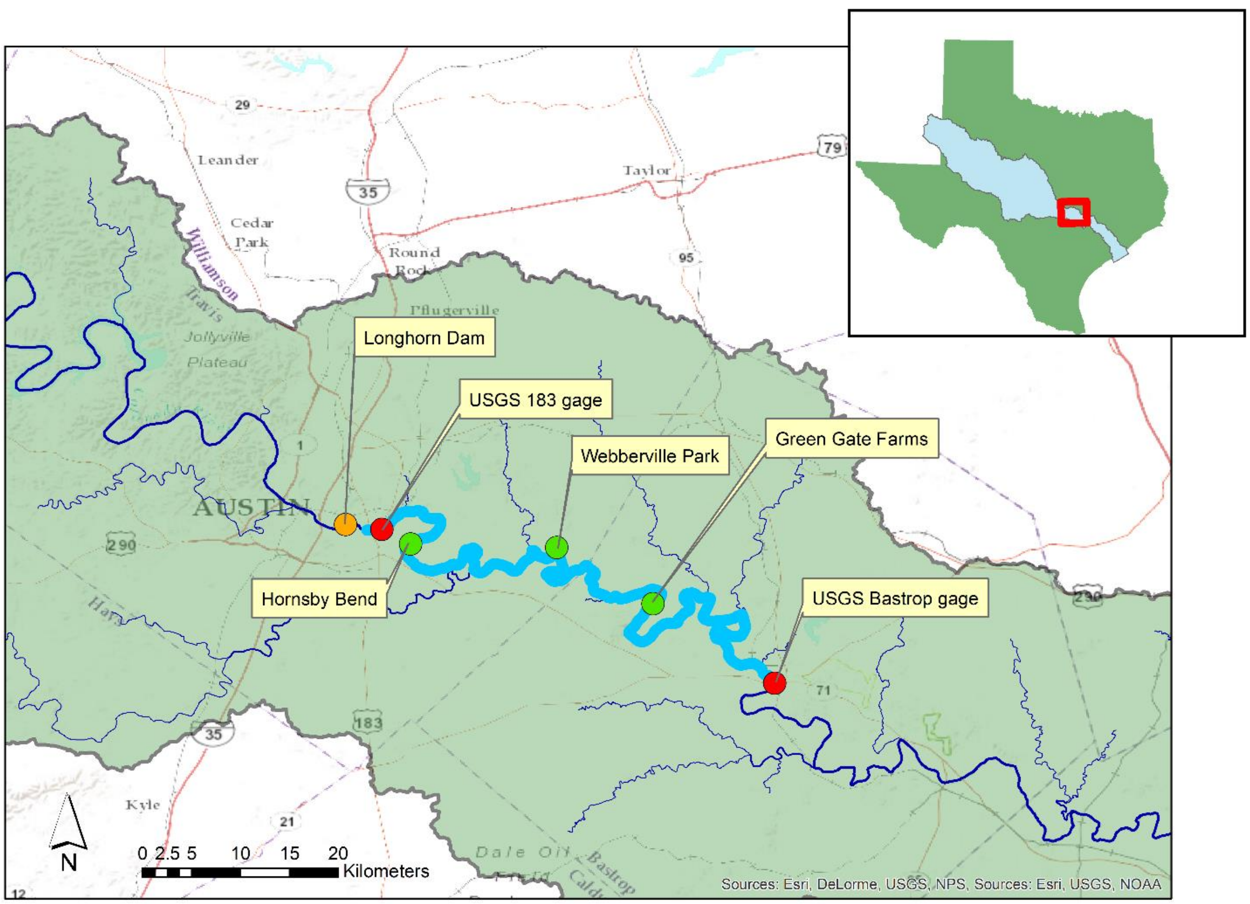

Figure 2*-Map of the study reach along the Colorado River with locations of field sites, USGS gages, and Longhorn Dam. Study reach in light blue.*

October 2015 Floods

Two large flood pulses travelled down the study reach spaced apart by approximately one week of time (figure 3). The first pulse arrived at the upstream part of the reach early morning on Oct 24 and the second on the morning of Oct. 30. Both pulses took approximately 24 hours to travel the full length of the study reach. Peak discharge from upstream to downstream was relatively unchanged during the Oct. 24 flood event. For the Oct. 30 flood event peak discharge nearly doubled from upstream to downstream (937-1727m^3/s from upstream to downstream)(figure 4). This was likely due to large flow inputs from the Onion Creek tributary which joins the Colorado between the two gages. The USGS onion creek gate reported a peak discharge of over 2,800 m^3/s.

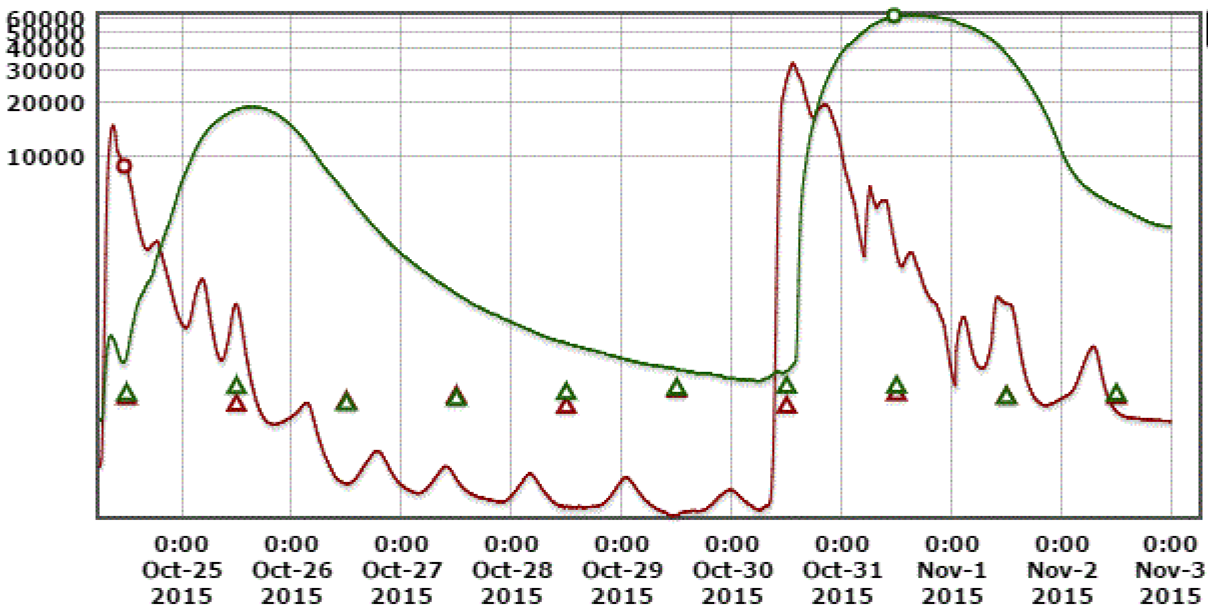

Figure 3*-USGS gage discharge estimates over the study period. 183 gage (SiteID: 08158000) in red, Bastrop gage (SiteID: 08159200) in green. (USGS 2015)*

In-situ stage discharge measurement

In-stream pressure transducers were installed at two sites along the study reach (Hornsby Bend and Webberville Park, figure 1) during the October 2015 floods. I surveyed river water levels before and after flood pulses and tied them to reference benchmarks to convert transducer pressure measurements to elevation above sea level. I then applied previously developed stage-discharge equations to the river elevation data to convert it to discharge (figure-stage discharge curve). The stage discharge rating curves for each field site were developed in May 2015 from discharge and survey measurements taken by researchers at Texas A&M (Knappet 2015, pers comm).

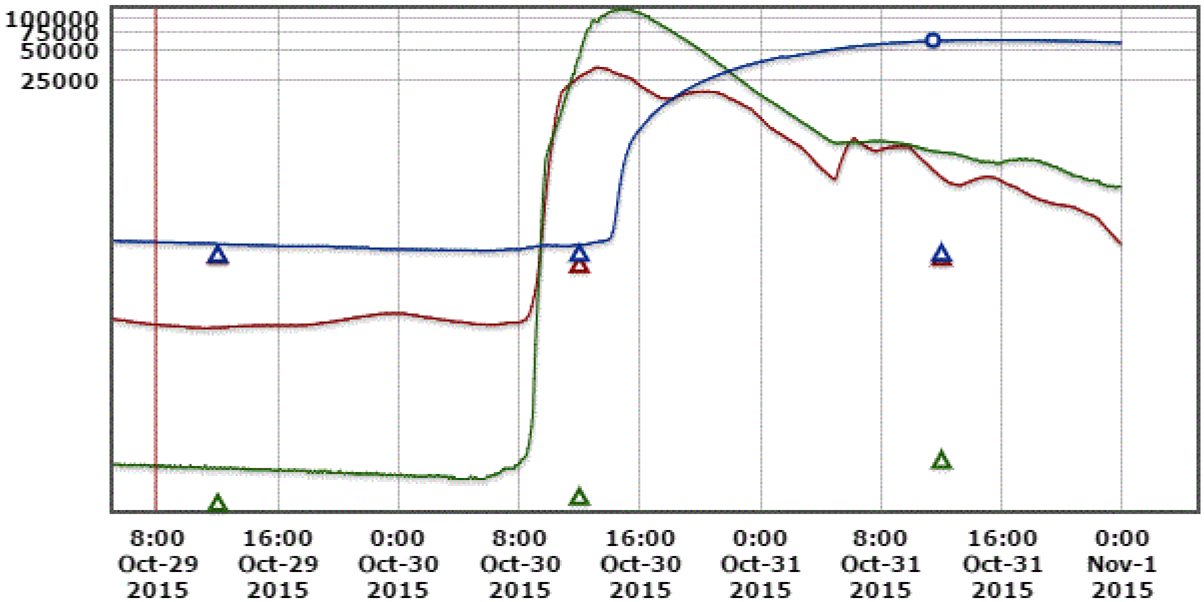

Figure 4*-USGS gage discharge estimates over the study period. 183 gage (SiteID: 08158000) in red, Onion creek gage (SiteID: 08159000) in green, Bastrop gage (SiteID: 08159200) in blue. (USGS 2015)*

Extraction and plotting of NFIE RAPID data

I obtained NFIE RAPID model datasets in netCDF (nc) format from Peirong Lin. Each .nc file contains discharge measurements linked with stream reach COMIDS. COMIDs of stream reaches overlapping with the LCR in-situ measurement sites were identified in ArcMap 10.3.1 (figure 5). Once the COMIDs were identified I wrote a script in Matlab (version R15b) which utilized the suite of Matlab NetCDF functions to identify and extract the model predicted discharges from each .nc RAPID model associated with the COMIDs. The script can be found in Appendix A of this publication.

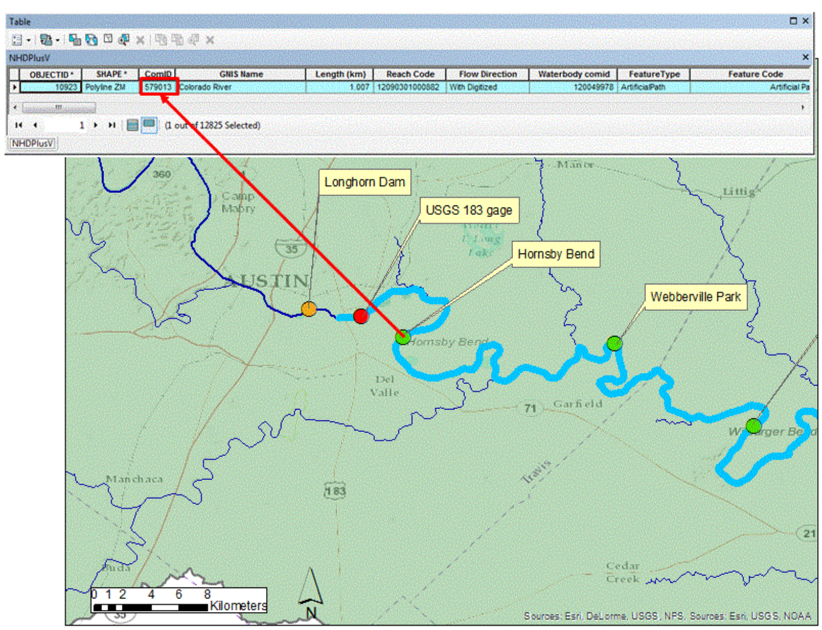

Figure 5*-Stream reach identification in ArcMap*

Analysis of RAPID and measured hydrographs

Hydrographs generated from the RAPID dataset were analyzed and compared with hydrographs generated from pressure transducer measurements. The magnitude of flood pulses was calculated from the following formula:

$$Peak\ flood\ discharge - preflood\ discharge\ (baseflow)$$

Where peak flood discharge is the maximum modelled discharge during the flood. By removing the pre-flood discharge the modelled flood peaks are not distorted by potential modelling overestimates of baseflow. The timing of the flood pulse was noted from the latest model in the dataset as it was assumed to have the most up-to-date rainfall models in its model input.

Results

Figures 6 and 7 present the hydrographs generated from the first flooding event (Oct. 24-25) at the upstream and downstream field sites (Hornsby bend and Webberville Park respectively). Figures 8 and 9 are the hydrographs generated from the second flooding event. Figure 10 presents a hydrograph generated from data during the base flow condition between the two flooding events where river stage had returned close to its pre-flood condition. Table 1 summarizes the magnitude of peak flood arrival times and discharge with pre-flood base flow discharge subtracted.

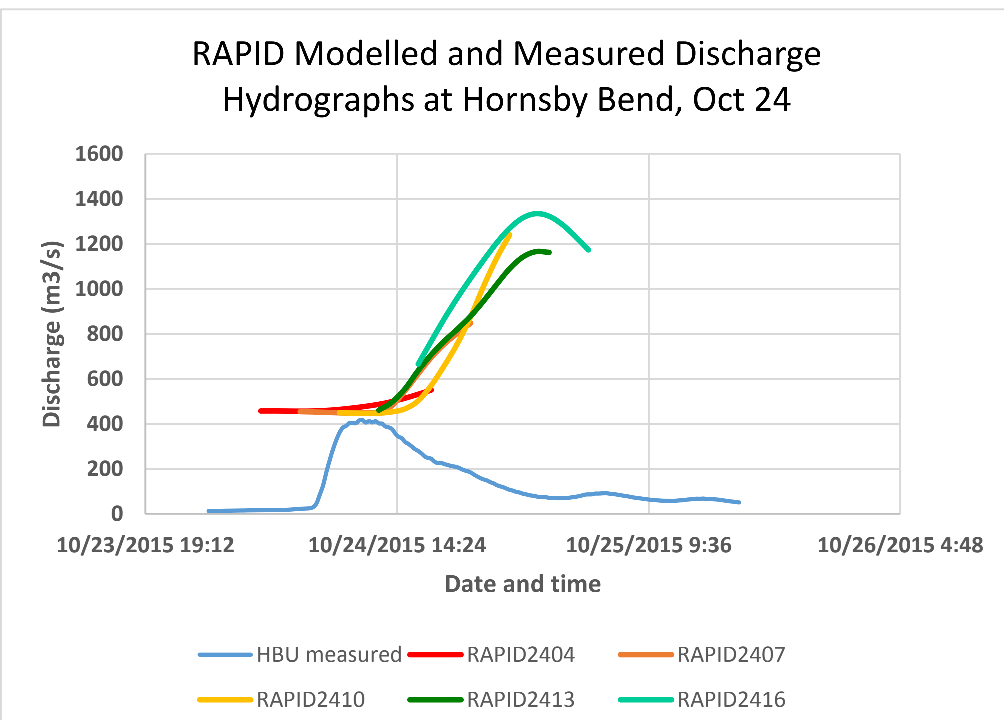

Figure 6*-Hornsby Bend hydrographs for Oct. 24 flood event*

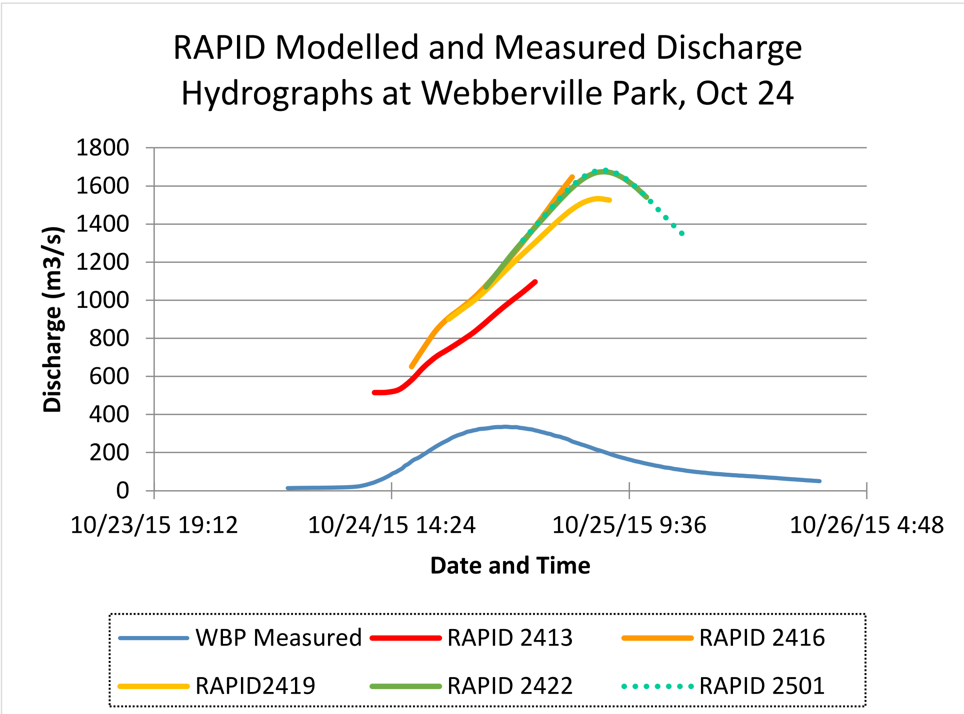

Figure 7*- Webberville Park hydrographs for Oct. 24 flood event*

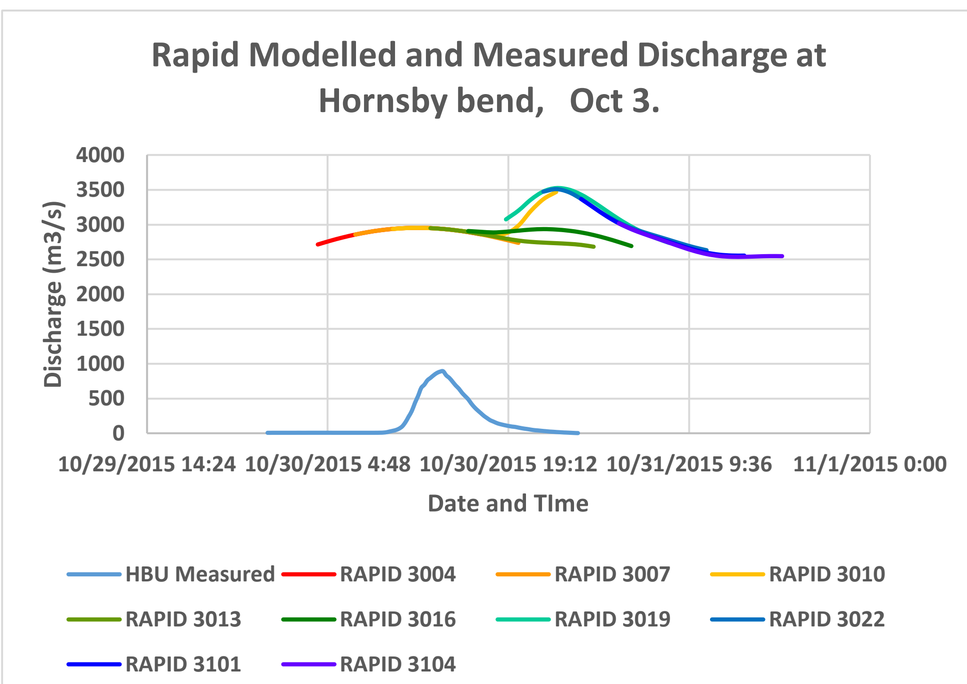

Figure 8*- Hornsby bend hydrographs for Oct. 30 flood event*

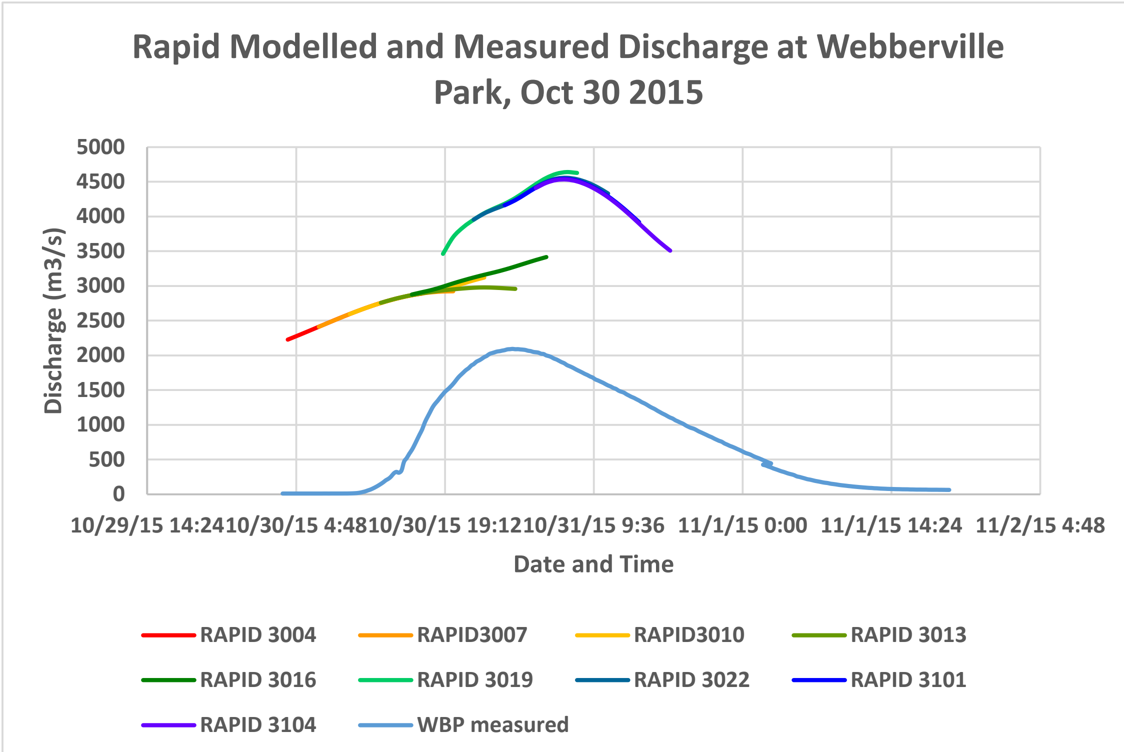

Figure 9*- Webberville Park hydrographs for Oct. 30 flood event.*

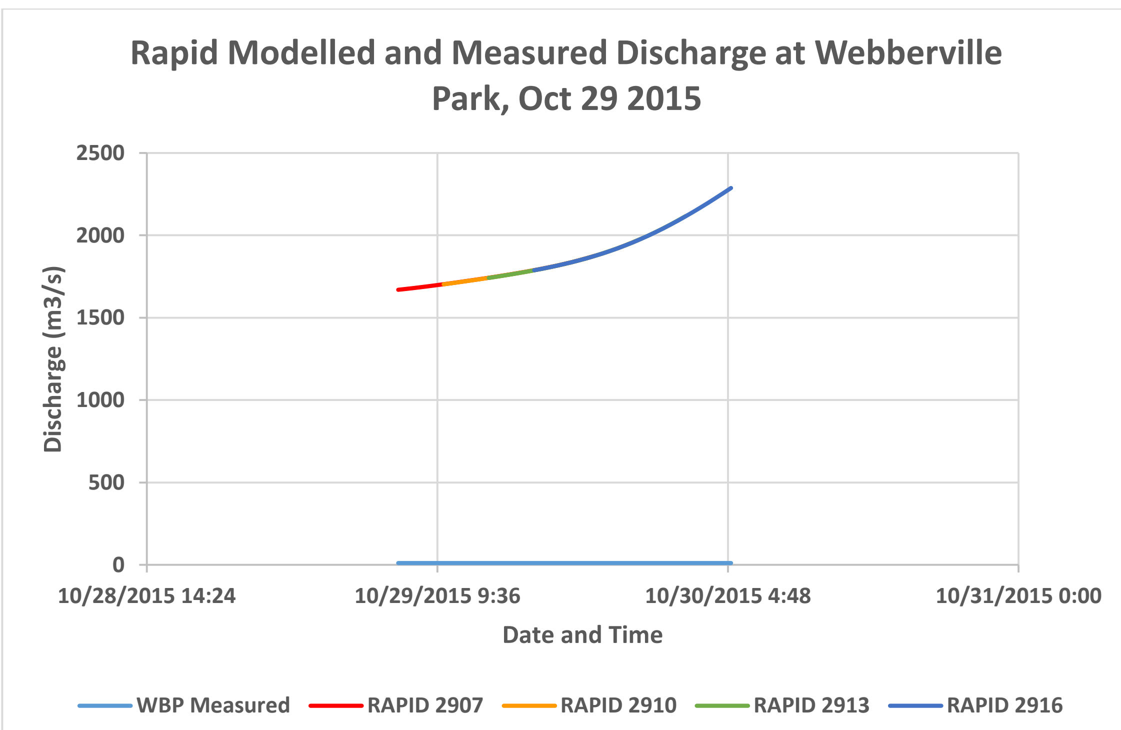

Figure 10*- Webberville Park hydrographs during pre-flood conditions, Oct. 29.*

Measured peak flow (m3/s)	Measured Peak arrival	Modelled peak flow (m3/s)	Modelled peak arrival	Peak flow offset	% peak flow difference	Peak arrival offset (hours)
414	10/24;11:45	878	10/25; 1:00	464	71.83	13.25
336	10/24;23:30	1155	10/25; 9:00	819	109.86	9.5
878	10/30; 14:00	1740	10/30; 23:00	862	65.85	9
2079	10/31; 1:45	2244	10/31; 7:00	165	7.63	5.25

Table 1*-Summary of measured and modelled peak discharge data after removing base flow component of discharge.*

For all hydrographs the RAPID modelled discharge significantly over predicted measured discharge (450-2000 m^3/s) (figures 6-10). Analysis of flood peak timing revealed that the RAPID model consistently predicted delayed timing of peak flood discharge arrival (5.25-13.25 hours) (table 1). The peak flood discharge was over estimated by the model for both sites during the Oct. 24 flood event and for the Hornsby bend site for the Oct. 30 flooding event (66-109% between measured and modelled discharge). Modelled and measured peak discharge for the Oct. 30 flood event at Webberville park site were within 7.6% (table 1).

Discussion

When comparing the RAPID modelled discharge with measured in-situ discharges the first question that needs to be answered is: are the in-situ discharge estimates representative of actual discharges? If the in-situ estimated flow values are not correct than it is not reasonable to use them to calibrate the RAPID modelling results.

To provide a qualitative answer to this question I compared measured Hornsby Bend discharges with USGS 183 (Site ID:08158000) gage discharges and the Webberville park in-situ values with the USGS Bastrop (Site ID; 08159200) gage values (see figure 1 for gage locations). The USGS 183 gage is 11.6km upstream of the Hornsby bend site and there is one significant tributary (Walnut creek) between the gage and measurement site. Summing the discharges from the Walnut creek (SiteID: 08158600) and USGS 183 gages should provide a reasonable estimate for flow just upstream of Hornsby bend. The Bastrop gage is approximately 54 km downstream of the Webberville park site and there are few significant tributaries along this reach, and should provide a reasonable coarse approximation of Webberville park peak discharge. Table 2 summarizes a comparison between estimated in-situ peak discharge and USGS peak gage discharge. All in-situ peak discharges were within 25% of USGS gage discharges except for the Webberville park discharge from the Oct. 24 flooding event, which was 45% lower than the USGS gage at Bastrop. With the exception of the latter measurement it appears that the in-situ measurements are providing a reasonable ‘Ball park’ estimate of discharge. They are all within 45% of the USGS discharge measurements, and considering that the RAPID model discharge estimates differed by more than an order of magnitude from the measured estimates, the measured data provides a reasonable baseline of flow estimates. Furthermore, because for much of our analysis we are interested in relative changes in discharge over a flooding event rather than absolute discharge, comparisons between the in-situ and RAPID modelled flow estimates are reasonable for qualitative comparison.

USGS gage discharge (m3/s)	In-situ estimated discharge (m3/s)	% difference
534	417	24.60
1133	893	23.69
535	335	45.97
1741	2090	18.21

***Table 2**- Summary of USGS gage data and in-situ measured data.*

It is difficult to understand why the NFIE RAPID model consistently over predicted discharge for all of the hydrographs without being able to see the relative contributions of the different model inputs to total flow. A quantitative analysis of these inputs is beyond the scope of this investigation, but qualitative analysis of the peak flood discharge versus modelled base flow discharge may provide some insight into these overestimations. RAPID base flow estimates on Oct. 29 during the inter-flood period estimated a base flow of between 1670 and 2288 m^3/s (figure 10). These model estimates followed a 5 day period where no rainfall had occurred and thus should represent a base-flow condition. These estimates are 2 orders of magnitude higher than the flows reported by the USGS along the Colorado during the same time period (~15m^3/s), suggesting that base flow values are being greatly over estimated. If subtracted from the total model flow estimates they bring the model significantly closer to observed conditions.

Taking this information into account it follows that one possible culprit for the overestimation is the model's estimation of groundwater inputs. The Noah-MP model uses a simple 1D unconfined aquifer model (JSG 2015). If the groundwater levels were modelled by Noah-MP to be excessively high, this would cause an overestimation of groundwater inputs into the river.

Analysis of the timing and magnitude of flood peak arrival and discharge revealed that the RAPID model consistently predicted late arrival of the flood peak and over predicted flood peak discharge. This was after the pre-flood base flow component of discharge was removed so that only relative change in discharge was being analyzed. Again, without knowing the water inputs from the Noah-MP model it is not possibly to identify exactly which model parameters are causing the discordance between modelled and measured data. But one possible model input that could cause them is the precipitation model and a lack accounting for urbanization in the runoff model. Noah-MP utilizes the NOAA HRRR forward modelled rainfall predictions. Accurate forward modelling of intensity the spatial distribution of rainfall within a watershed is a difficult task to accomplish. If the precipitation models were over predicting rainfall within the LCR watershed the RAPID model would predict higher discharge measurements than what was observed.

The smallest delay in flood peak arrival, as well as the closest agreement between modelled and measured discharge, occurred at Webberville Park during the Oct. 30 flooding event (table 1). The key difference between this flooding event and the first event was that the majority of flow was coming from the Onion Creek tributary (figure 4). Because the confluence of Onion Creek occurs between the two field sites, only the Webberville Park stream reach would be subject to modelling differences due to the different source waters of the flood. So one possible explanation for the high level of agreement between modelled and measured flooding along this reach could be that the model is better at predicting runoff from within the Onion Creek watershed, and worse at predicting it in the LCR watershed upstream of the confluence. The Onion Creek watershed is significantly less urbanized than the LCR watershed upstream, which could explain why the model was better able to predict flooding for Onion Creek. Additionally, runoff from precipitation upstream of Longhorn Dam is subject to impoundment and sudden release from dams in the upper Colorado River, which complicates flow modelling.

Conclusions

There were significant differences between modelled and measured flow along river reach studied in this investigation. These differences could be a result of failure of the model to accurately predict groundwater inputs to base flow, overland flow, the intensity and spatial distribution of rainfall, or a combination of these and many other model inputs. Without access to the model inputs that went into discharge predictions it is impossible to know which of the input model parameters is to blame. In light of this, a more in-depth investigation of these model inputs is recommended. The NFIE RAPID forward flow modelling system has the potential to be a powerful tool for the forecasting of future flood conditions. Future investigations such as this one will be required to calibrate the model and improve forecasting.

Acknowledgments

Thanks to Peirong Lin for taking the time to share selected NFIE RAPID data files with me, and for clarifying aspects about how the model works. Thanks to Peter Knappett's Texas A&M research group for developing stage-discharge ratings for the sites and to the graduate students in Bayani Cardenas' research group who assisted with LCR field work.

References

Knappett, P, 2015, Personal communication, data received via e-mail July 2015.

JSG, 2015, Noah-MP Users Site, accessed 4 Dec. 2015 < http://www.jsg.utexas.edu/noah-mp/>

NOAA, 2015, Earth System Research Laboratory, accessed 4 Dec. 2015 < http://ruc.noaa.gov/hrrr/>

USGS, 2015, USGS water data for the nation, accessed 4 Dec. 2015 < http://waterdata.usgs.gov/nwis/>

Appendix A- Matlab script for extracting Qout discharge variable from selected gage locations

```
%Script for importing 13 hr forward modelled discharge data from the NFIE
%RAPID dataset.
%% Open CDFs of interest
cd('C:\Users\Jeff\Box Sync\nfie_data')
wbpnc2213=netcdf.open('ncRapid2015102213.nc');
wbpnc2413=netcdf.open('ncRapid2015102413.nc');
wbpnc2416=netcdf.open('ncRapid2015102416.nc');
wbpnc2419=netcdf.open('ncRapid2015102419.nc');
wbpnc2422=netcdf.open('ncRapid2015102422.nc'); %opening nc file associated with timestep closest to flood peak:
10pm Oct 24, Approx 2 hrs before peak flood arrival
wbpnc2907=netcdf.open('ncRapid2015102907.nc');
wbpnc2910=netcdf.open('ncRapid2015102910.nc');
wbpnc2913=netcdf.open('ncRapid2015102913.nc');
wbpnc2916=netcdf.open('ncRapid2015102916.nc');
wbpnc2919=netcdf.open('ncRapid2015102922.nc');
wbpnc2922=netcdf.open('ncRapid2015102922.nc');
wbpnc2501=netcdf.open('ncRapid2015102501.nc');
wbpnc3004=netcdf.open('ncRapid2015103004.nc');
wbpnc3007=netcdf.open('ncRapid2015103007.nc');
wbpnc3010=netcdf.open('ncRapid2015103010.nc');
wbpnc3013=netcdf.open('ncRapid2015103013.nc');
wbpnc3016=netcdf.open('ncRapid2015103016.nc');
wbpnc3019=netcdf.open('ncRapid2015103019.nc');
wbpnc3022=netcdf.open('ncRapid2015103022.nc');
wbpnc3101=netcdf.open('ncRapid2015103101.nc');
wbpnc3104=netcdf.open('ncRapid2015103104.nc');
wbpnc3107=netcdf.open('ncRapid2015103107.nc');
hbunc2404=netcdf.open('ncRapid2015102404.nc');
hbunc2407=netcdf.open('ncRapid2015102407.nc');
hbunc2410=netcdf.open('ncRapid2015102410.nc');
hbunc2413=netcdf.open('ncRapid2015102413.nc');
hbunc2416=netcdf.open('ncRapid2015102416.nc');
hbunc3004=netcdf.open('ncRapid2015103004.nc');
hbunc3007=netcdf.open('ncRapid2015103007.nc');
hbunc3010=netcdf.open('ncRapid2015103010.nc');
hbunc3013=netcdf.open('ncRapid2015103013.nc');
hbunc3016=netcdf.open('ncRapid2015103016.nc');
hbunc3019=netcdf.open('ncRapid2015103019.nc');
hbunc3022=netcdf.open('ncRapid2015103022.nc');
hbunc3101=netcdf.open('ncRapid2015103101.nc');
hbunc3104=netcdf.open('ncRapid2015103104.nc');
hbunc3107=netcdf.open('ncRapid2015103107.nc');
hbunc2907=netcdf.open('ncRapid2015102907.nc');
hbunc2910=netcdf.open('ncRapid2015102910.nc');
hbunc2913=netcdf.open('ncRapid2015102913.nc');
hbunc2916=netcdf.open('ncRapid2015102916.nc');
%%
%%WBP2413
Qwbp2413=netcdf.getVar(wbpnc2413,1); %variable 1= Discharge (m3/s)
CID=netcdf.getVar(wbpnc2413,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp2413=Qwbp2413(CIDwbp,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 2015
clear Qwbp2413
clear CID

%%WBP2416
Qwbp2416=netcdf.getVar(wbpnc2416,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc2416,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp2416=Qwbp2416(CIDwbp,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 2015
clear Qwbp2416
clear CID

%%WBP2419
Qwbp2419=netcdf.getVar(wbpnc2419,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc2419,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp2419=Qwbp2419(CIDwbp,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 2015
clear Qwbp2419
clear CID
%%WBP2422
Qwbp2422=netcdf.getVar(wbpnc2422,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc2422,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp2422=Qwbp2422(CIDwbp,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 2015
clear Qwbp2422
clear CID
%%WBP2501
```

```
Qwbp2501=netcdf.getVar(wbpnc2501,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc2501,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp2501=Qwbp2501(CIDwbp,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 2015
clear Qwbp2501
clear CID
%%WBP2907
Qwbp2907=netcdf.getVar(wbpnc2907,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc2907,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp2907=Qwbp2907(CIDwbp,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 2015
clear Qwbp2907
clear CID
%%WBP2910
Qwbp2910=netcdf.getVar(wbpnc2910,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc2910,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp2910=Qwbp2910(CIDwbp,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 2015
clear Qwbp2910
clear CID
%%WBP2913
Qwbp2913=netcdf.getVar(wbpnc2913,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc2913,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp2913=Qwbp2913(CIDwbp,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 2015
clear Qwbp2913
clear CID
%%WBP2916
Qwbp2916=netcdf.getVar(wbpnc2916,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc2916,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp2916=Qwbp2916(CIDwbp,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 2015
clear Qwbp2916
clear CID
%%WBP3004
Qwbp3004=netcdf.getVar(wbpnc3004,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc3004,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp3004=Qwbp3004(CIDwbp,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 2015
clear Qwbp3004
clear CID
%%WBP3007
Qwbp3007=netcdf.getVar(wbpnc3007,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc3007,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp3007=Qwbp3007(CIDwbp,:); %stores vector of hourly dischar
clear Qwbp3007
clear CID
%%WBP3010
Qwbp3010=netcdf.getVar(wbpnc3010,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc3010,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp3010=Qwbp3010(CIDwbp,:); %stores vector of hourly dischar
clear Qwbp3010
clear CID
%%WBP3013
Qwbp3013=netcdf.getVar(wbpnc3013,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc3013,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp3013=Qwbp3013(CIDwbp,:); %stores vector of hourly dischar
clear Qwbp3013
clear CID
%%WBP3016
Qwbp3016=netcdf.getVar(wbpnc3016,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc3016,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp3016=Qwbp3016(CIDwbp,:); %stores vector of hourly dischar
clear Qwbp3016
clear CID
%%WBP3019
Qwbp3019=netcdf.getVar(wbpnc3019,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc3019,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp3019=Qwbp3019(CIDwbp,:); %stores vector of hourly dischar
clear Qwbp3019
clear CID
%%WBP3022
Qwbp3022=netcdf.getVar(wbpnc3022,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc3022,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp3022=Qwbp3022(CIDwbp,:); %stores vector of hourly dischar
```

```
clear Qwbp3022
clear CID
%%WBP3101
Qwbp3101=netcdf.getVar(wbpnc3101,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc3101,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp3101=Qwbp3101(CIDwbp,:); %stores vector of hourly dischar
clear Qwbp3101
clear CID
%%WBP3104
Qwbp3104=netcdf.getVar(wbpnc3104,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc3104,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp3104=Qwbp3104(CIDwbp,:); %stores vector of hourly dischar
clear Qwbp3104
clear CID
%%WBP3107
Qwbp3107=netcdf.getVar(wbpnc3107,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc3107,0); % variable 0=COMIDS
CIDwbp=find(CID==5790132) %identifies row of desired COMID stream reach
Qpeakwbp3107=Qwbp3107(CIDwbp,:); %stores vector of hourly dischar
clear Qwbp3107
clear CID
%%HBU2404
Qhbu2404=netcdf.getVar(hbunc2404,1); %variable 1= timestep
CID=netcdf.getVar(hbunc2404,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu2404=Qhbu2404(CIDhbu,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 201
clear Qhbu2404
clear CID
%%HBU2407
Qhbu2407=netcdf.getVar(hbunc2407,1); %variable 1= timestep
CID=netcdf.getVar(hbunc2407,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu2407=Qhbu2407(CIDhbu,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 201
clear Qhbu2407
clear CID
%%HBU2410
Qhbu2410=netcdf.getVar(hbunc2410,1); %variable 1= timestep
CID=netcdf.getVar(hbunc2410,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu2410=Qhbu2410(CIDhbu,:); %stores vector of hourly discharge s
clear Qhbu2410
clear CID
%%HBU2413
Qhbu2413=netcdf.getVar(hbunc2413,1); %variable 1= timestep
CID=netcdf.getVar(hbunc2413,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu2413=Qhbu2413(CIDhbu,:); %stores vector of hourly discharge s
clear Qhbu2413
clear CID
%%HBU2416
Qhbu2416=netcdf.getVar(hbunc2416,1); %variable 1= timestep
CID=netcdf.getVar(hbunc2416,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu2416=Qhbu2416(CIDhbu,:); %stores vector of hourly discharge s
clear Qhbu2416
clear CID
%%HBU3004
Qhbu3004=netcdf.getVar(hbunc3004,1); %variable 1= timestep
CID=netcdf.getVar(hbunc3004,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu3004=Qhbu3004(CIDhbu,:); %stores vector of hourly discharge s
clear Qhbu3004
clear CID
%%HBU3007
Qhbu3007=netcdf.getVar(hbunc3007,1); %variable 1= timestep
CID=netcdf.getVar(hbunc3007,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu3007=Qhbu3007(CIDhbu,:); %stores vector of hourly discharge s
clear Qhbu3007
clear CID
%%HBU3010
Qhbu3010=netcdf.getVar(hbunc3010,1); %variable 1= timestep
CID=netcdf.getVar(hbunc3010,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu3010=Qhbu3010(CIDhbu,:); %stores vector of hourly discharge s
clear Qhbu3010
clear CID
%%HBU3013
Qhbu3013=netcdf.getVar(hbunc3013,1); %variable 1= timestep
```

```
CID=netcdf.getVar(hbunc3013,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu3013=Qhbu3013(CIDhbu,:); %stores vector of hourly discharge s
clear Qhbu3013
clear CID

%%HBU3016
Qhbu3016=netcdf.getVar(hbunc3016,1); %variable 1= timestep
CID=netcdf.getVar(hbunc3016,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu3016=Qhbu3016(CIDhbu,:); %stores vector of hourly discharge s
clear Qhbu3016
clear CID
%%HBU3019
Qhbu3019=netcdf.getVar(hbunc3019,1); %variable 1= timestep
CID=netcdf.getVar(hbunc3019,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu3019=Qhbu3019(CIDhbu,:); %stores vector of hourly discharge s
clear Qhbu3019
clear CID
%%HBU3022
Qhbu3022=netcdf.getVar(hbunc3022,1); %variable 1= timestep
CID=netcdf.getVar(hbunc3022,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu3022=Qhbu3022(CIDhbu,:); %stores vector of hourly discharge s
clear Qhbu3022
clear CID
%%HBU3101
Qhbu3101=netcdf.getVar(hbunc3101,1); %variable 1= timestep
CID=netcdf.getVar(hbunc3101,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu3101=Qhbu3101(CIDhbu,:); %stores vector of hourly discharge s
clear Qhbu3101
clear CID
%%HBU3104
Qhbu3104=netcdf.getVar(hbunc3104,1); %variable 1= timestep
CID=netcdf.getVar(hbunc3104,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu3104=Qhbu3104(CIDhbu,:); %stores vector of hourly discharge s
clear Qhbu3104
clear CID
%%HBU3107
Qhbu3107=netcdf.getVar(hbunc3107,1); %variable 1= timestep
CID=netcdf.getVar(hbunc3107,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923); %identifies row of desired COMID stream reach
Qpeakhbu3107=Qhbu3107(CIDhbu,:); %stores vector of hourly discharge s
clear Qhbu3107
clear CID


%%HBU2907
Qhbu2907=netcdf.getVar(wbpnc2907,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc2907,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923) %identifies row of desired COMID stream reach
Qpeakhbu2907=Qhbu2907(CIDhbu,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 2015
clear Qwbp2907
clear CID

%%HBU2922
Qhbu2922=netcdf.getVar(wbpnc2922,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc2922,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923) %identifies row of desired COMID stream reach
Qpeakhbu2922=Qhbu2922(CIDhbu,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 2015
clear Qwbp2922
clear CID

%%HBU2213
Qhbu2213=netcdf.getVar(wbpnc2213,1); %variable 1= timestep
CID=netcdf.getVar(wbpnc2213,0); % variable 0=COMIDS
CIDhbu=find(CID==5781923) %identifies row of desired COMID stream reach
Qpeakhbu2213=Qhbu2213(CIDhbu,:); %stores vector of hourly discharge starting from 22:00 Oct 24, 2015
clear Qwbp2213
clear CID
```