Rainfall Runoff Simulation of ASAN River Watershed using Agent Based Modelling

NAME	ENROLLMENT NUMBER	COURSE
ABHISEK MAITI	mgi17-10930	M.SC
SAYANTAN MAJUMDAR	mgi17-10933	M.SC
SHASHWAT SHUKLA	mgi17-10934	M.SC
SHIV NARAYAN YADAV	pgd17-10944	PGD
SAURABH GUPTA	pgd17-10945	PGD

FACULTY OF GEO-INFORMATION SCIENCE AND EARTH OBSERVATION
UNIVERSITY OF TWENTE, ENSCHEDE, THE NETHERLANDS

INDIAN INSTITUTE OF REMOTE SENSING,
INDIAN SPACE RESEARCH ORGANISATION, DEHRADUN, INDIA

Table of Contents

List of Figures

1. Introduction

1.1. Background

Flood hazard is one of the most frequent natural disasters in India. It is observed that, most of the floods occurs in northern part of India (Building Materials & Technology Promotion Council, 2009). Floods of northern India are generally riverine flood. Intense precipitation in monsoon, poor natural drainage, low percolation in soils and irregular sediment transportation are major contributing factors to these disasters (Pandey, Singh, & Nathawat, 2010). GIS plays an important role in predicting and mapping of potential flood areas. Appropriate agent based modelling plays an essential role for simulation of hydrological models (FEMA, 2004).

This study mainly focuses on the run off simulation of Asan river watershed of Uttarakhand. Equation based modelling requires all the system behaviours to be formalized mathematically. Many geophysical phenomena are either not possible to describe mathematically or extremely hard and complex to formalize. In such cases equation based modelling is not very suitable for simulation purposes (Helbing & Balietti, 2012). On the contrary, agent based modelling follows a simple approach of predicting macro level behaviour by accumulating the effects of numerous micro level phenomenon. This approach is based on the assumption that simple behavioural rules collectively generate complex behavioural pattern and the overall pattern may be quite different from simple sum of the micro level behaviours. Thus agent based modelling takes care of the non-linearity of the phenomenon to be simulated.

Agent based models are suitable for simulating very complex systems which are hard to characterize mathematically. River run off is a complex phenomenon which is hard to model mathematically. One of the study suggests that, agent based model is suitable to simulate such phenomenon (Khare, Patra, Mondal, & Kundu, 2014).

1.2. Study Area

Asan Barrage is situated at the confluence of Eastern Yamuna Canal and Asan River near Uttarakhand-Himachal Pradesh border region in Doon Valley, (Dehradun District). It has been selected as the study area. It is located between latitudes 30°14'14"N to 30°29'54"N and longitudes 77°39'42"E to 78°05'30"E, Dehradun district, Uttarakhand State, India as shown in Figure 1. It covers approximately 724 Km^2 area. The elevation of the study varies between 352 m and 2230 m with respect to mean sea level. The average annual temperature ranges from 21°C in summers to 5°C in winters. Most of the annual rainfall in the study area is received during the months from June to September, July and August being the rainiest. The mean annual rainfall in the watershed is around 1,917 mm.

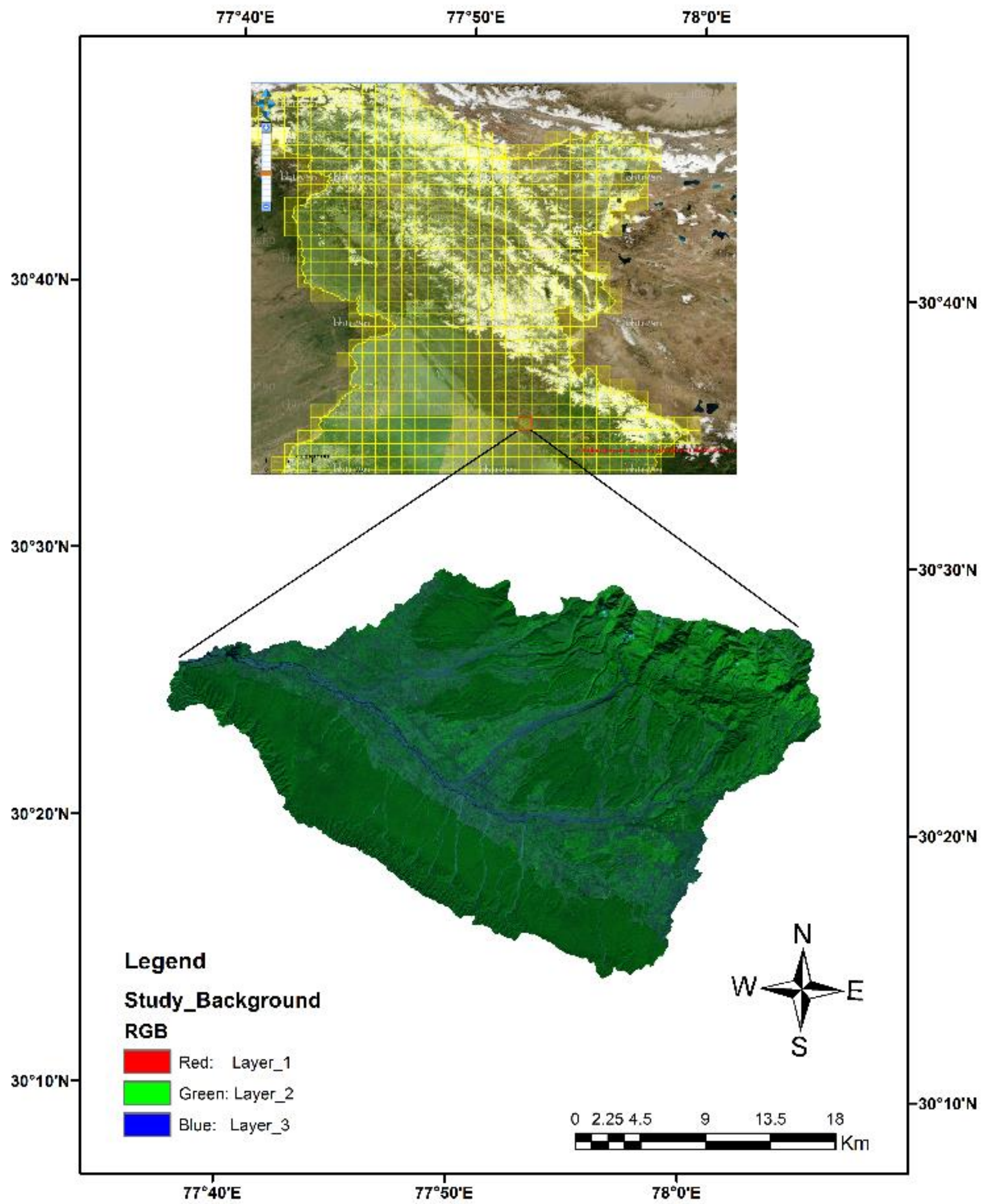

Figure 1.1: Location of Study Area

1.3. Objective

The main objective of the study is to simulate the rainfall runoff of Asan river watershed using agent based modelling technique. The following sub-objectives have been considered in the project to fulfil the main objective of the study:

a) *Data Preparation*: The database for the DEM (digital elevation model) and rainfall precipitation (2013) have been prepared during this phase.
b) *Building the model*: The Netlogo programming environment has been used to build the rainfall runoff model where the aforementioned datasets have been fed as inputs.
c) *Analysis*: The generated hydrograph using 'NetLogo' and 'R' has been compared with those of SWAT and HEC-HMS hydrology models in order to verify the agent based model correctness. An elevation change comparison has also been done to observe the effects of soil erosion due to the runoff.

2. Methodology

2.1. Model Selection

NetLogo is a programmable modelling environment for simulating natural and social phenomena. It is particularly well suited for modeling complex systems developing over time. Modelers can give instructions to hundreds or thousands of "agents" all operating independently. This makes it possible to explore the connection between the micro-level behavior of individuals and the macro-level patterns that emerge from their interaction ("NetLogo 6.0.2 User Manual," n.d.). The workflow below describes the methodology followed in performing the study.

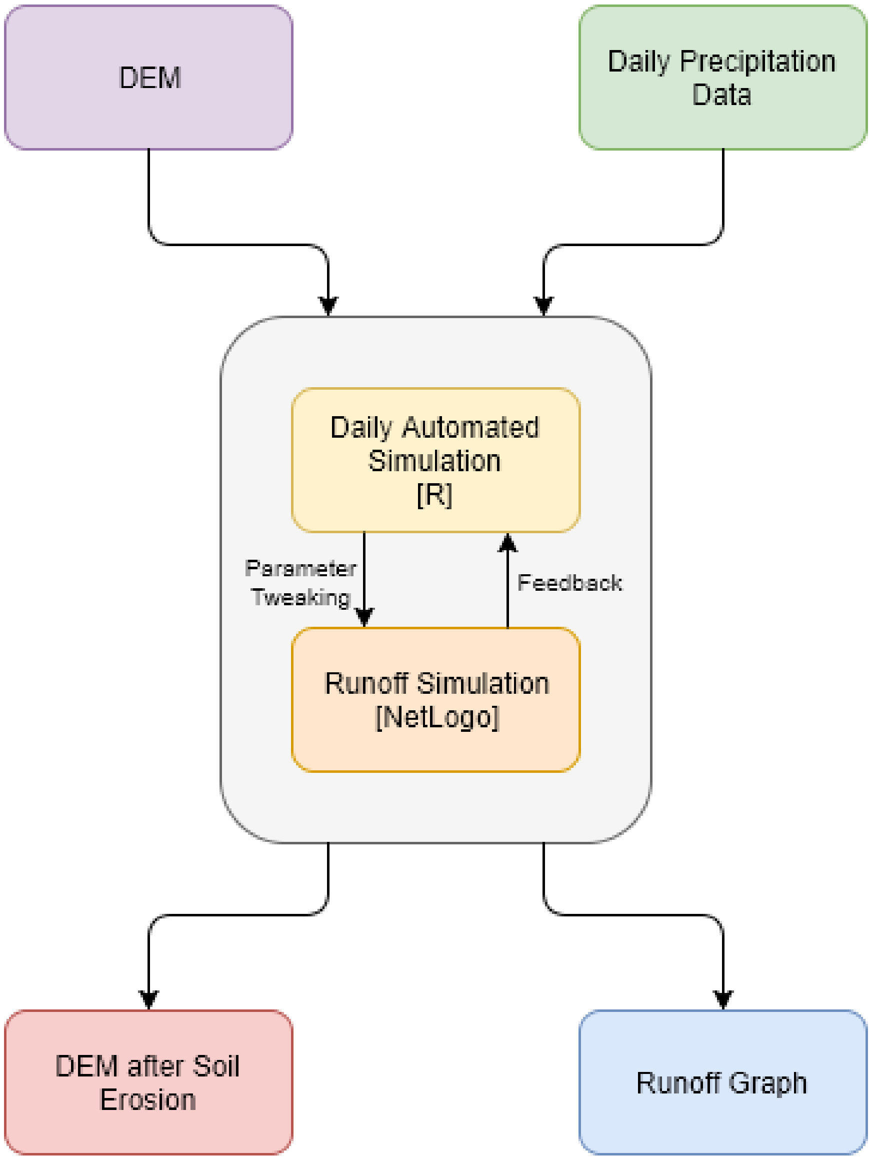

Figure 2.1 Workflow Diagram

2.2. Datasets

The following datasets are used in this study:

- Two Aster Digital Elevation Tiles dated 28 October, 2008
- Weather Data acquired from Climate Forecast System Reanalysis (CFSR) online repository (National Centers for Environmental Prediction (NCEP), 2016).
- LULC map of Asan Watershed (used in HEC-HMS and SWAT models)
- Soil map of Asan Watershed (used in HEC-HMS and SWAT models)

2.3. Model Setup and Simulation

About 5 years (1 January 2008 – 31 September 2014) of meteorological data have been used as the precipitation parameter in NetLogo, SWAT and HEC-HMS models. The data have been collected from ("Global Weather Data for SWAT," n.d.). The following weather information was present in the data:

- Temperature (Unit: ° C)
- Precipitation (Unit: mm)
- Relative Humidity (Unit: fraction)
- Wind Speed (Unit: m/s)
- Solar Radiant Exposure (Unit: MJ/m^2)

2.4. Validation and Calibration

Initially the model was simulated using only one year (1 January 2013 – 31 December 2013) weather data which was available with the data set. Solar radiant exposure information was not available in the aforementioned weather data therefore it was simulated. Later, three years weather data was downloaded from Climate Forecast System Reanalysis (CFSR) online repository (National Centers for Environmental Prediction (NCEP), 2016) and used in the model. This yielded more refined result.

3. Results

3.1. Simulated Hydrographs for the year 2013

3.1.1. Using 'Netlogo' and 'R'

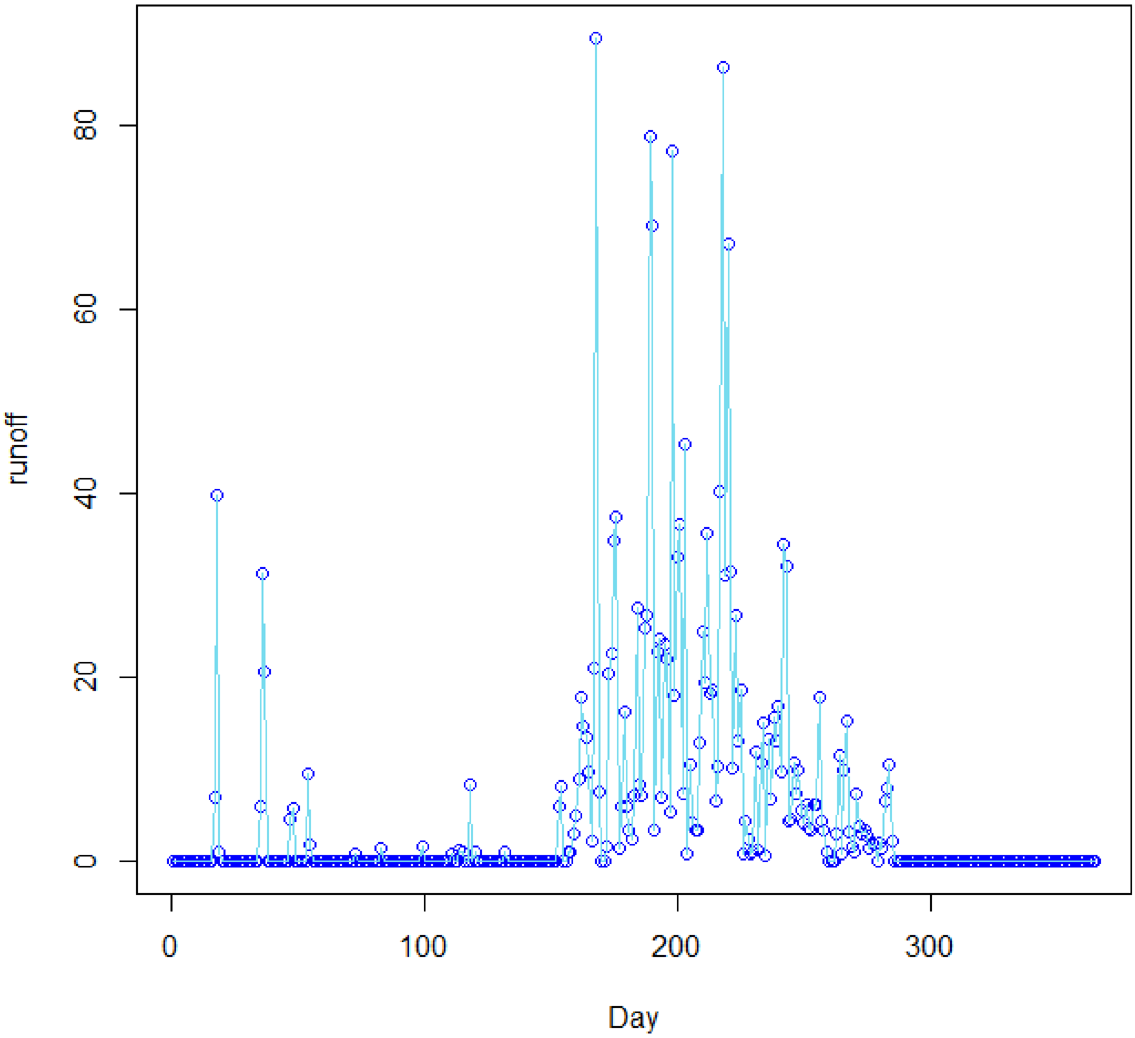

Figure 3.1 Hydrograph using NetLogo and R (2013)

The simulated hydrograph (Fig 3.1) is generated from the NetLogo model output. The maximum discharge of 89.364 m^3/s is observed in August.

3.1.2. Using 'SWAT'

A recent comparative study of the major hydrological models suggests that SWAT is very good model for estimation of daily discharge (Golmohammadi, Prasher, Madani, & Rudra, 2014). SWAT is also proven to be more suitable for streamflow estimation in the Himalayan basins (Shrestha, Shrestha, & Shrestha, 2017). SWAT model put more stress on LULC and soil, water quality loading, flexibility of basin discretization and continuous temporal simulation (Arnold, Srinivasan, Muttiah, & Williams, 1998). These specification are very much aligned the requirements of this study. Also SWAT is computationally less intensive compared to other hydrological model (Malagò, Pagliero, Bouraoui, & Franchini, 2015).

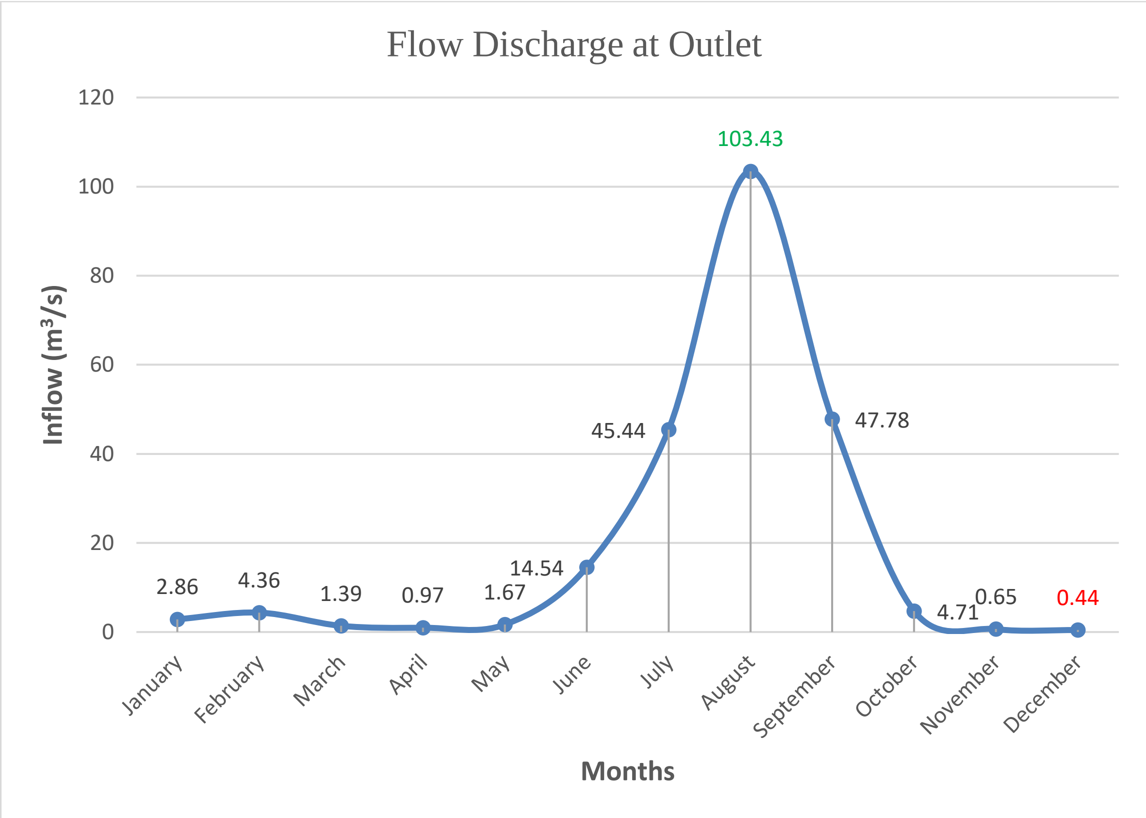

Figure 3.2 Hydrograph using SWAT (2013)

The simulated hydrograph (Fig 3.2) is generated from the SWAT model output. The minimum discharge of 0.44 m^3/s is observed in December and the maximum discharge of 103.43 m^3/s is observed in August.

3.1.3. Using 'HEC-HMS'

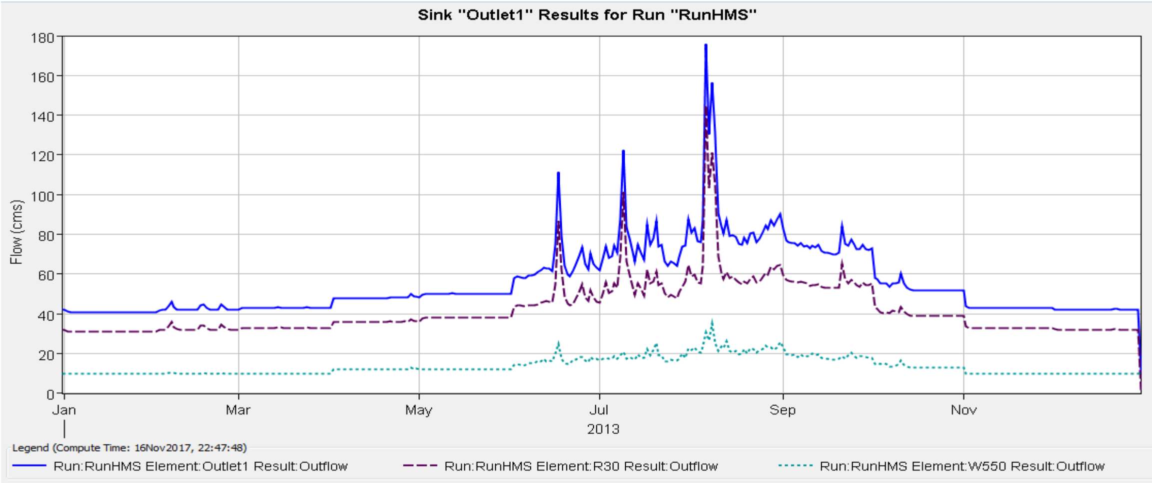

Figure 3.3 Hydrograph using HEC-HMS (2013)

The above hydrograph shows that the peak discharge of water through the outlet is during the monsoon period as well. To be specific, the specified outlet has the highest discharge of 175.3 m^3/s on Aug 6, 2013.

3.2. Hydrology Cycle

Hydrology cycle (Fig 3.4) is generated using the SWAT Check tool.

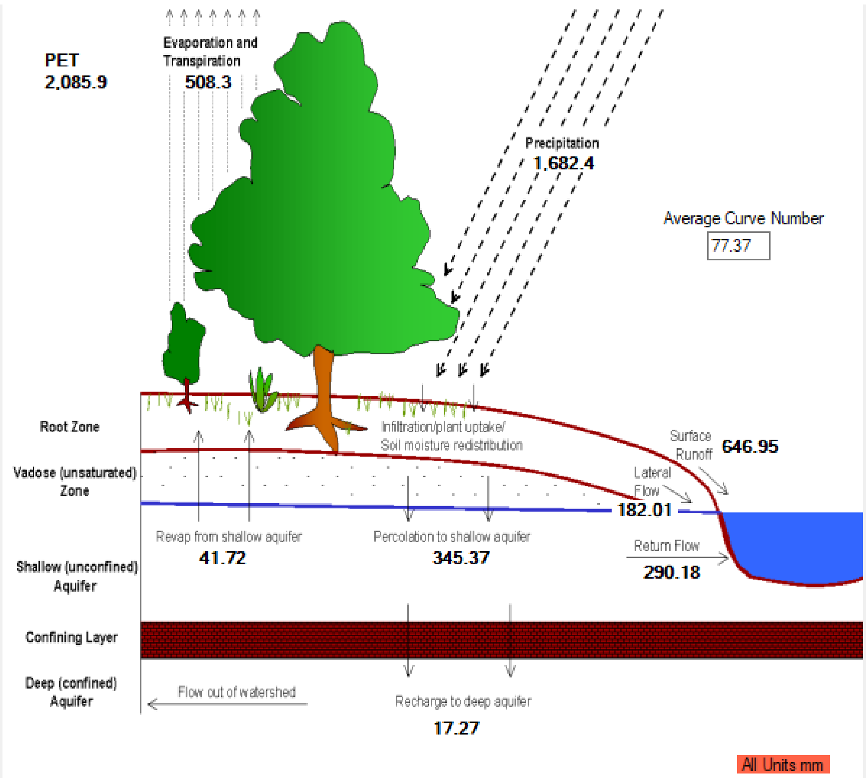

Figure 3.4 Simulated Hydrology Cycle using SWAT

3.3. Soil Erosion Results

Run off is directly related to precipitation intensities. Soil erosion is likely to be heavily affected by runoff and changes in rainfall and cover. According to Snyman, H. A. et al, terrain also plays an important role in soil erosion (Snyman, Van Rensburg, & Opperman, 1985). This study reveals the soil erosion pattern in the study area by simulating the expected soil erosion response depending upon the terrain and seasonal change in precipitation. The elevation change due to erosion is shown below.

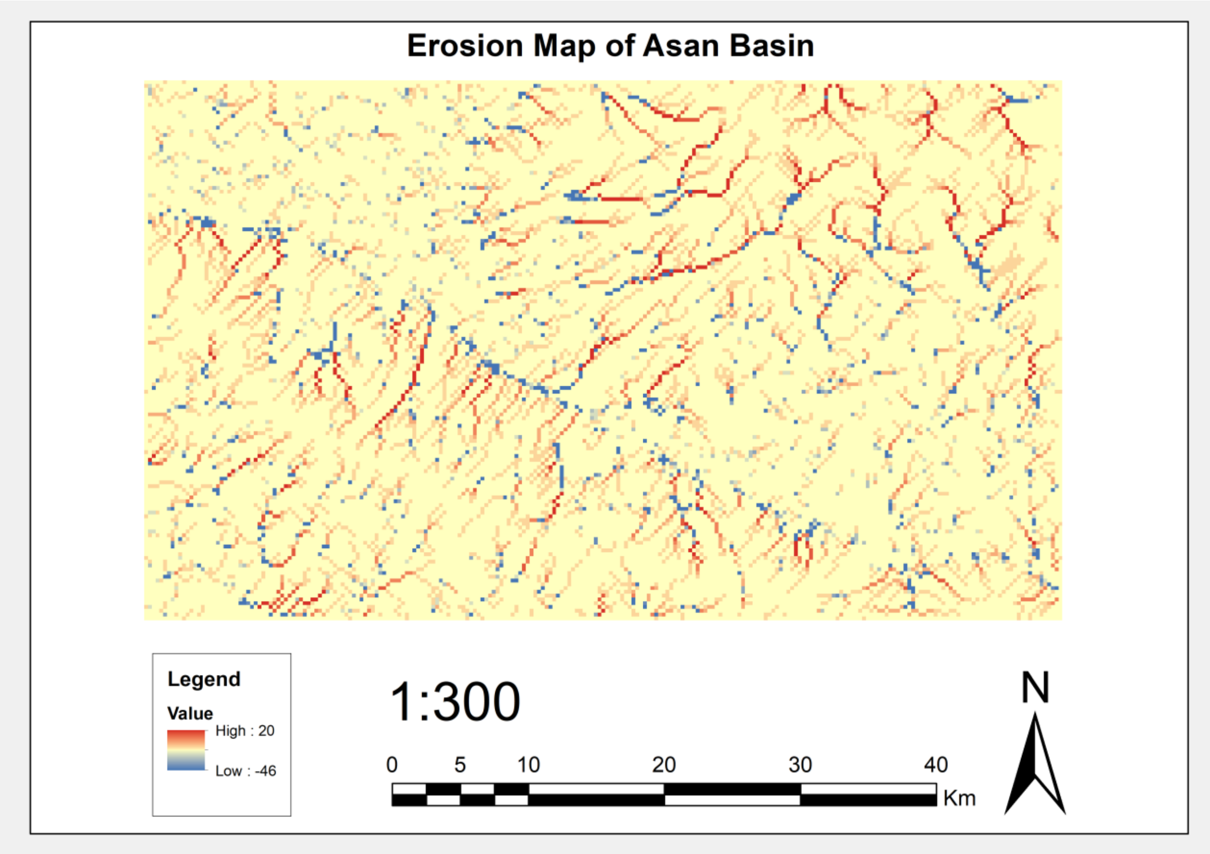

Figure 3.5 Soil Erosion Map

4. Conclusion

The obtained results simulate runoff and soil erosion pattern of the study area according to seasonal precipitation for the year 2013. This model is also verified on the previous five years (2008 - 2012) data. The run off graph shows high discharge during the months of June to October, which are aligned with the monsoon period of the area. Soil erosion increases with the increase in slope of the terrain and local precipitation.

The discharge sharply decreases after monsoon period. This also indicates the beginning of the winter season when the snow starts to get accumulated at the mountain peaks. Evaporation also increases due to dry winter season.

The simulation might improve with more weather data (Shamir, Imam, Gupta, & Sorooshian, 2005) and accurate soil parameters (Holman, 2012) of the study area. Base flow and absorption capability of soil also have significant roles in soil erosion. Although the overall result is acceptable, however it may not be sufficiently accurate. The ground verification of the simulation result has not been carried out to quantify the accuracy of the model.

References

Arnold, J. G., Srinivasan, R., Muttiah, R. S., & Williams, J. R. (1998). LARGE AREA HYDROLOGIC MODELING AND ASSESSMENT PART I: MODEL DEVELOPMENT. *Journal of the American Water Resources Association*, *34*(1), 73–89. https://doi.org/10.1111/j.1752-1688.1998.tb05961.x

Building Materials & Technology Promotion Council. (2009). Flood Hazard Map. *Vulnerability Atlas*, 92.

FEMA. (2004). Guidelines and Specifications for Flood Hazard Mapping Partners - Ch D.4.3 Flood Frequency Analysis Methods. *Volume 1: Flood Studies and Mapping*, (November), 1–20.

Global Weather Data for SWAT. (n.d.). Retrieved January 31, 2018, from https://globalweather.tamu.edu/

Golmohammadi, G., Prasher, S., Madani, A., & Rudra, R. (2014). Evaluating Three Hydrological Distributed Watershed Models: MIKE-SHE, APEX, SWAT. *Hydrology*. https://doi.org/10.3390/hydrology1010020

Helbing, D., & Balietti, S. (2012). *How to Do Agent-Based Simulations in the Future: From Modeling Social Mechanisms to Emergent Phenomena and Interactive Systems Design*. *Time*. https://doi.org/10.1007/978-3-642-24004-1

Holman, I. (2012). Soil Hydrology, Land Use and Agriculture: Measurement and Modelling. *Soil Use and Management*. https://doi.org/10.1111/j.1475-2743.2012.00392.x

Khare, D., Patra, D., Mondal, A., & Kundu, S. (2014). Impact of landuse/land cover change on run-off in a catchment of Narmada river in India. *Applied Geomatics*, *7*(1), 23–35. https://doi.org/10.1007/s12518-014-0148-6

Malagò, A., Pagliero, L., Bouraoui, F., & Franchini, M. (2015). Comparing calibrated parameter sets of the SWAT model for the Scandinavian and Iberian peninsulas. *Hydrological Sciences Journal*, 1–19. https://doi.org/10.1080/02626667.2014.978332

National Centers for Environmental Prediction (NCEP). (2016). Global Weather Data for SWAT.

NetLogo 6.0.2 User Manual. (n.d.). Retrieved January 31, 2018, from https://ccl.northwestern.edu/netlogo/docs/whatis.html

Pandey, A. C., Singh, S. K., & Nathawat, M. S. (2010). Waterlogging and flood hazards vulnerability and risk assessment in Indo Gangetic plain. *Natural Hazards*, *55*(2), 273–289. https://doi.org/10.1007/s11069-010-9525-6

Shamir, E., Imam, B., Gupta, H. V., & Sorooshian, S. (2005). Application of temporal streamflow descriptors in hydrologic model parameter estimation. *Water Resources Research*, *41*(6), 1–16. https://doi.org/10.1029/2004WR003409

Shrestha, S., Shrestha, M., & Shrestha, P. K. (2017). Evaluation of the SWAT model performance for simulating river discharge in the Himalayan and tropical basins of Asia. *Hydrology Research*.

Snyman, H. A., Van Rensburg, W. J. L., & Opperman, D. P. J. (1985). Soil loss and run-off measurements form natural veld with a rainfall simulator. *Journal of the Grassland Society of Southern Africa*, *2*(4), 35–40.

Appendix 1. NetLogo Code

```
extensions[gis]
breed[raindrops raindrop]
breed[waters water]
globals [
  elevation-dataset
  border          ;; keep the patches around the edge in a global
                  ;; so we don't ever have to ask patches in go
  min-e           ;;minimum elevation
  max-e           ;;maximum elevation
  the-row         ;;used in export-data. it is the row being written
]
patches-own [
  elevation
  initial_elevation
  elevation_change
  amount_rain   ;;how many drops of rain here
]
turtles-own [
  soil        ;;how much soil a raindrop is carrying
  ]
to setup
  ca
  resize-world -121 121 -71 71
  set elevation-dataset gis:load-dataset "data/asan_dem.asc"
  gis:set-world-envelope gis:envelope-of elevation-dataset
  gis:apply-raster elevation-dataset elevation
  gis:apply-raster elevation-dataset initial_elevation
  show_elevation
  set-default-shape turtles "circle"
  set border patches with [ count neighbors != 8 ]
  reset-ticks
  end

to show_elevation
  set min-e gis:minimum-of elevation-dataset
  set max-e gis:maximum-of elevation-dataset
  ask patches [set pcolor scale-color black elevation min-e max-e]
  end

to go
  ;;this part uses codes from the library model Grand Cayon, with some modifications
  create-raindrops rain-rate
  [ ifelse show_water_amount? or show_elevation_change? [hide-turtle set color blue][set color blue]
    set size 2
    set soil 0
    move-to one-of patches
   ]
 ifelse draw?
    [ ask turtles [ pd ] ]
    [ clear-drawing
      ask turtles [ pu ] ]
  ask raindrops [ ifelse erosion? [flow_with_erosion][flow] ]
  ask border  [ask turtles-here [ die ]]
;;codes from Grand Cayon end here
  ifelse show_water_amount?
    [show_amount_of_water]
    [ifelse show_elevation_change? and erosion?[show_elevation_change ]
      [  ask turtles [show-turtle]
         show_elevation]]
  tick
```

```
  end
to flow
  ;;this part uses codes from the library model Grand Cayon, with some modifications
  let target min-one-of neighbors [ elevation + ( count turtles-here * water-height) ]
  ifelse [elevation + (count turtles-here * water-height)] of target
     < (elevation + ((count turtles-here - 1) * water-height))
    [ face target move-to target ]
    [ set breed waters ]
  ;;codes from Grand Cayon end here
end
to flow_with_erosion
  ;;this part uses codes from the library model Grand Cayon, with some modifications
  let target min-one-of neighbors [ elevation + ( count turtles-here * water-height) ]
  ifelse [elevation + (count turtles-here * water-height)] of target
     < (elevation + ((count turtles-here - 1) * water-height))
    [ ;;consider erosion effects
      ask patch-here [set elevation elevation - 1]
      set soil soil + 1
      face target
      move-to target
    ]
    [ set breed waters ask patch-here [set elevation elevation + [soil] of myself] set soil 0]
  ;;codes from Grand Cayon end here
end
to show_amount_of_water
  ;To show by scaled color. However, becuase the variation is small, it may be hard to see the difference.
  ask patches [set amount_rain count turtles-here  ]
  set max-e [amount_rain] of max-one-of patches [amount_rain]
  ask patches with [amount_rain > 0 ][set pcolor scale-color blue amount_rain (max-e + 1) 0 ]
  ask patches with [amount_rain = 0 ][set pcolor white]
  ask turtles [hide-turtle]
end
to show_elevation_change
  ask patches [set elevation_change elevation - initial_elevation]
  ask turtles [hide-turtle]
  ask patches with [elevation_change > 0][set pcolor green ];;increased
  ask patches with [elevation_change < 0][set pcolor red ];;decreased
  ask patches with [elevation_change = 0][set pcolor black]
end
to export_data
  ;file-delete "data/result.asc"
  file-open "data/result.asc"
  let i 71
  while [i >= -72]
    [ set the-row []
      set the-row patches with [pycor = i]
        foreach sort-on [pxcor] the-row [ ?1 -> ask ?1 [file-write  elevation ] ]
      file-write "nl"
      set i i - 1]
  file-close
end
```

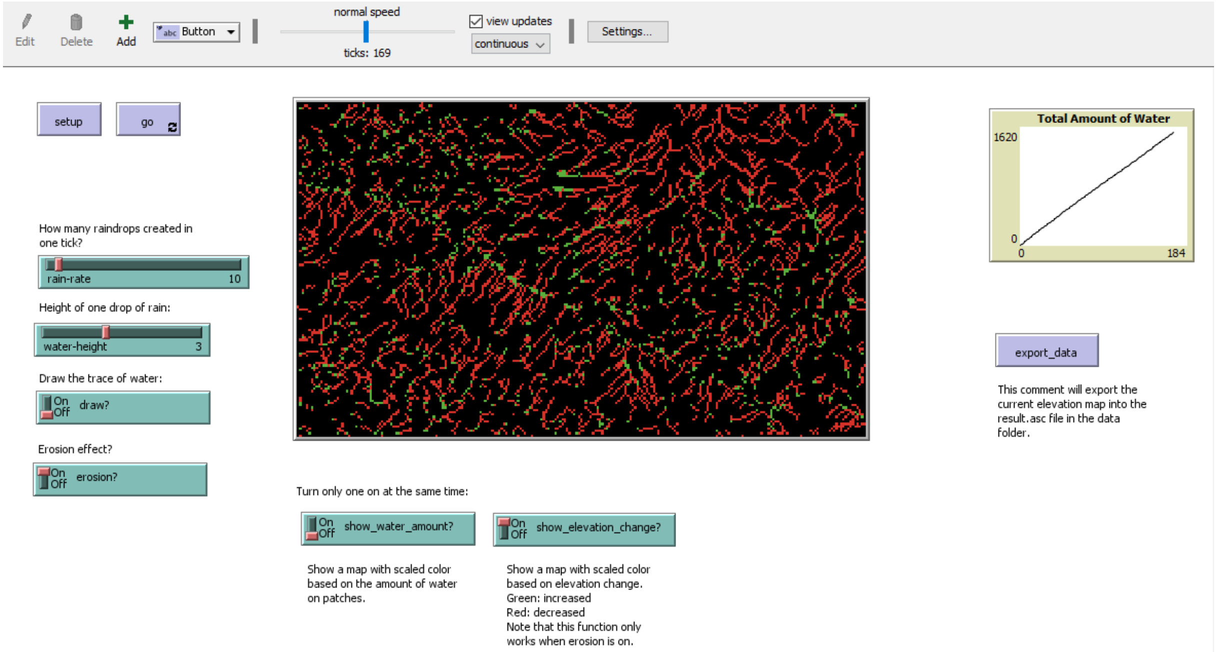

Figure Soil Erosion

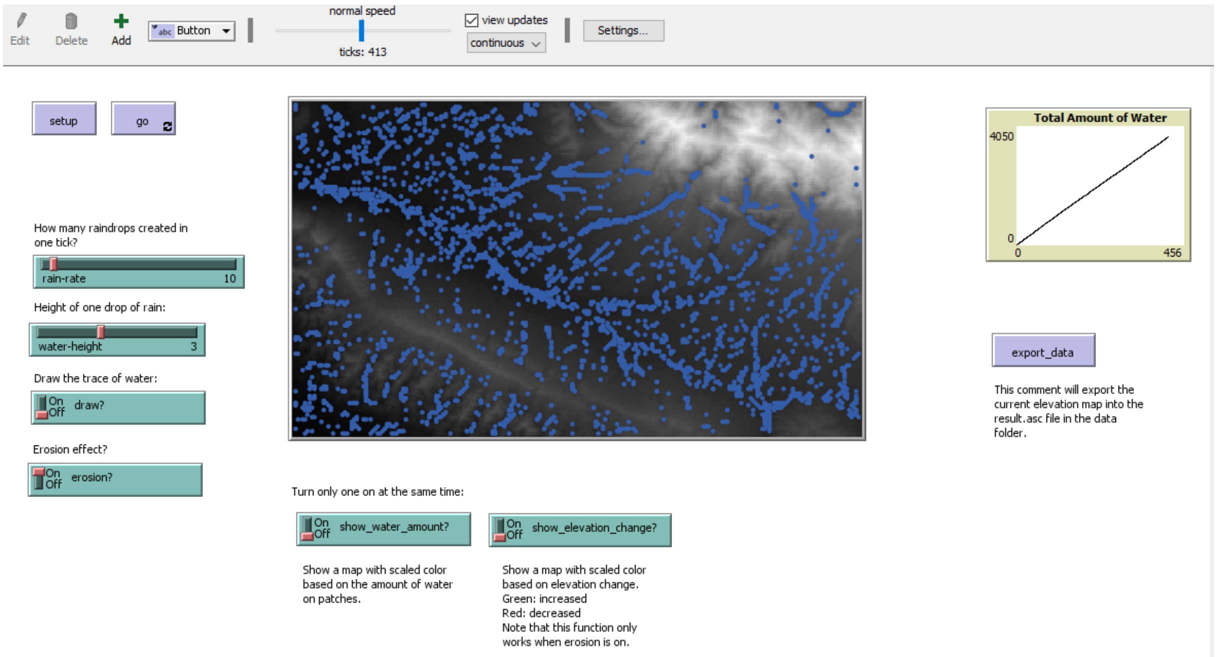

Figure Precipitation

Appendix 2. R Code

```
options(java.home = "C:\\Program Files\\Java\\jre1.8.0_162")
library(RNetLogo)
model.path < -"D:\\SDAM\\Project\\rainfall_ssa.nlogo"
nlDir < -"C:\\Program Files\\NetLogo 6.0.2\\app"
setwd(nlDir)
nl.path < -getwd()
NLStart(nl.path, gui = TRUE, nl.jar = 'netlogo-6.0.2.jar')
NLLoadModel(model.path)
NLCommand("set draw? false")
NLCommand("set erosion? true")
NLCommand("set show_water_amount? false")
NLCommand("set show_elevation_change? false")
runoff = c(1: 365)
rain_rate = read.csv(file = "D:\\SDAM\\Project\\data\\weather_2013.csv",
header = TRUE, sep = ",")
for (i in 1: 365) {
    rr = round(rain_rate$Precipitation[i])
    wh = runif(1, 1, 4)
    if (rr == 0) {
            wh = 0
    } else if (rr >= 70) {
        wh = runif(1, 4, 6)
    }
    wh = round(wh)
    NLCommand("set rain-rate", rr)
    NLCommand("set water-height", wh)
    NLCommand("setup")
    NLCommand("go") runoff[i] = NLReport("count turtles * ( 0.55 + random-float 0.45 )")
}
print(runoff)
plot(runoff, xlab = "Day", col = "blue")
lines(runoff, col = "#77dbee")
```

Appendix 3. Python Code

```
def clean_dem_file(input):
    fp = open(input, 'r')
    data = fp.read()
    lines = data.split('"nl"')
    fp.close()
    values = []
    new_lines = []
    numlines = len(lines) - 2
    for line in lines[: numlines]:
        line = line.strip()
        new_lines.append(line)
        values.append(len(line.split(' ')))
    return max(values), numlines, new_lines

def read_lines(fp, n):
    lines = []
    while n > 0:
        lines.append(fp.readline())
        n -= 1
    fp.close()
    return lines[2:]

def generate_new_dem_file(original_dem, output, data):
    params = read_lines(open(original_dem, 'r'), 6)
    fp2 = open(output, 'w')
    ncols = 'ncols         ' + str(data[0])
    nrows = 'nrows         ' + str(data[1])
    fp2.write(ncols + '\n' + nrows + '\n')
    for param in params:
        fp2.write(param)
    for d in data[2]:
        fp2.write(d + '\n')
    fp2.close()

ncols, nrows, new_lines = clean_dem_file("D:/SDAM/Project/data/result.asc")
generate_new_dem_file("D:/SDAM/Project/data/asan_dem.asc", "D:/SDAM/Project/data/out.asc", (ncols, nrows, new_lines))
```