**Summary:**

To investigate watershed responses to various rainfall events, a framework for a synthetic rainfall-runoff database was built. Simple procedures are used in the framework to help with successful flood risk assessment and management. Rainfall generators, continuous rainfall runoff models, and database management systems are all included. In the upper Yarmouk River basin, the system was tested in two gauged river sections (Jordan). One of the key concerns was determining whether or not basic procedures could still provide high-quality outcomes. It was decided to simulate runoff by using a rainfall-runoff model that included a high number of rainfall episodes. Pre-simulated events are stored in the rainfall runoff database according to the quantity of rainfall, starting moisture conditions, and initial discharge. This analogue technique doesn't need any fresh model simulations for the real-time operational prediction. The predictions, on the other hand, are based on data from the rainfall-runoff database's simulations (for the specific class to which the forecast belongs). Consequently, the database may be a useful tool for estimating future streamflow situations based on a variety of rainfall amounts. The database's application to the research location confirms that the magnitudes of genuine flood occurrences were accurately documented. In order to make the suggested technique more useful, more work should be done to include a component that accounts for the actual temporal distribution of rainfall events into the stochastic rainfall generator and to employ other rainfall-runoff models.

The following files are distributed:

1) MATLAB codes

1.1) "model.m": model code

1.2) "model \_calibration.m": code for model calibration (requires optimization toolbox)

2) Secondary file:

2.1) "fixed\_par.txt": contains

a) the area of the basin (km^2)

b) the computational time step (for flood event simulation), 0.2 is OK

2.2) "IUH.txt": contains the coordinates of the dimensionless IUH (Instantaneous Unit Hydrograph)

2.3) "X\_opt\_ model.txt": parameter values of model

W\_p = PAR(1); % initial conditions, fraction of W\_max (0-1)

W\_max = PAR(2); % Field capacity

m2 = PAR(3); % exponent of drainage

Ks = PAR(4); % Ks parameter of infiltration and drainage

Nu = PAR(5); % fraction of drainage versus interflow

gamma1 = PAR(6); % coefficient lag-time relationship

Kc = PAR(7); % parameter of potential evapotranspiration

lambda = PAR(8); % initial abstraction coefficient

Sr\_coeff = PAR(9); % multiplicative coefficient for Sr

3) INPUT file :

3.1) "section.txt": example file for section (64 km^2)

It contains 4 columns

a) date (in numeric Matlab format)

b) rainfall depth (in mm)

c) air temperature (°C)

d) observed discharge data (m^3/s)

4) OUTPUT file:

4.1) "section.png": figure with the model output

Upper figure: relative soil moisture timeseries

Middle figure: rainfall timeseries

Lower figure: Observed vs simulated discharge

-------------------------------------------------------------------

To run the model for your own catchment the following steps have to be done:

1) creation of input file, e.g. "section.txt" and change the basin area in the "fixed\_par.txt" file

2) model calibration with " model \_calibration.m"

4) Run of the model with " model.m"

% ... few lines to run the model

name='section'; % name of the input file

model \_calibration.m (name) % model calibration

model (name,load('X\_opt\_ model.txt'),1) % RUN model