Simple CREST Lumped version

Coupled Routing and Excess Storage (CREST) Model

Reference:
Wang. J., Y. Hong, L. Li, J.J. Gourley, K. Yilmaz, S.I. Khan, F.S. Policelli, R.F. Adler, S. Habib, D. Irwn, S.A. Limaye, T. Korme, and L. Okello, 2011: The Coupled Routing and Excess STorage (CREST) distributed hydrological model. Hydrol. Sciences Journal, 56, 84-98.

CREST Lumped version for Matlab
Version 1.0 - December, 2012
Written by **Humberto Vergara** (humber@ou.edu)
Hydrometeorology and Remote-Sensing (HyDROS) Laboratory
Advanced Radar Research Center (ARRC)
The University of Oklahoma

```
%% ReadMe Block
% This is a template matlab script file to run a lumped version of
CREST.
% It includes settings for a sample dataset (included in package) that
can
% be modified and customized to accommodate alternative data. Read the
% comments preceding each line for instructions on how to use the
script
% file and how to define settings to run CREST.

%% -----------Run script Block-----------------------------------------
----
%Clear memory and screen (recommended)
clear; clc;

%% Step 1) Read input data

% Load input data into memory. This can be done in a variety of ways.
For
% the sample dataset, that has been stored in a single text file. Data
can
% be stored in multiple files. Data can be assigned to the appropriate
% variables directly if preferred (see "Assign input data to CREST
variables" below).

% Sample Dataset: Ft.Cobb basin
data = dlmread('ts.07325800.crest.csv', ',', 1,1);

% Assign input data to CREST variables (required): Here, input data is
assigned to
% appropriate CREST variables. All data is passed to the CREST function
as a single struct variable.
% In this example, the struct variable is named "rundata" but this can
be customized.
% However, the elements of the struct variable have names that must be
used
% accurately (i.e. not customizable).

% Input data must be time series of Potential Evapotranspiration (PET)
```

```
in
% mm/hr and Rainfall (PP) in mm/hr. All time series passed to the model
must
% match in the period the data is defined for:

%struct_var.PET - Potential evapotranspiration time series (mm/hr)
rundata.PET = data(:,4);
%struct_var.PP - Rainfall time series (mm/hr)
rundata.PP = data(:,3);

%% Step 2) Simulation settings
%Basin Area in km^2 - This value is used to convert runoff rate (mm/hr)
to
%Streamflow rate (m^3/s)
rundata.barea = 342;
%Simulation time step in hours
rundata.tstep = 1;
%Simulation time period - This is defined using Matlab's built-in
serial
%date numbers. This period must match the input data period and number
of
%elements of the time series (e.g. arrays).
rundata.tperiod =
datenum(2006,10,1,0,0,0):1/24:datenum(2007,10,01,0,0,0);

%% Step 3) Set CREST Parameters values
%PKE - Multiplier to convert PET
rundata.Parameters(1) = 0.5099;
%PIM - The impervious area ratio. This ratio is stored as a percentage
between 0 and 100.
rundata.Parameters(2) = 14.2402;
%PWM - The maximum soil water capacity (depth integrated pore space) of
the
%soil layer (mm). Range > 0
rundata.Parameters(3) = 108.1742;
%PFC - The soil saturated hydraulic conductivity (Ksat, mm/hr). Range >
0
rundata.Parameters(4) = 9.7051;
%LEAKO - The overland reservoir discharge multiplier. Range 0 - 1
rundata.Parameters(5) = 0.2977;
%LEAKI - The interflow reservoir discharge multiplier. Range 0 - 1
rundata.Parameters(6) = 0.060;
%PB - The exponent of the variable infiltration curve.
rundata.Parameters(7) = 1;
%Nqo - Number of overland reservoirs - Range >= 1
rundata.Parameters(8) = 3;

%% Step 4) Set CREST States values
rundata.States_0 = zeros(2+rundata.Parameters(8),1);
%IWU - The initial value of soil water. This is a percentage of the pWm
and should therefore vary between 0 and 100.
rundata.States_0(1) = 0.1;
%ISU - The initial value of interflow reservoir (mm)
rundata.States_0(2) = 0;
%ISO - The initial value of overland reservoir (mm). The total number
of
%overland reservoirs is defined with CREST parameter 8 (See above
```

```
parameter
%Nqo). Either a single initial value can be assigned to all reservoirs
or different initial values
%for each of them.
%For example: Nqo = 2:
%rundata.States_0(3) = 3.5;
%rundata.States_0(4) = 0.7;
rundata.States_0(3:end) = 0; %Same initial value for all overland
reservoirs

%% Step 5) Run the model
%Function CREST stores all output data in a struct variable. In this
%example, the variable "prediction" will take outputs. This name is
%customizable.
prediction = crest(rundata);

%% Plotting (optional)
%Load observed streamflow for validation (recommended)
rundata.obsQ = data(:,2);
rundata.obsQ_period =
datenum(2006,10,1,0,0,0):1/24:datenum(2007,10,01,0,0,0);

%Use Matlab's built in functions to plot simulation and observed
%hydrographs
plot(rundata.tperiod,data(:,2), 'Color', 'k', 'DisplayName',
'Observed', 'LineStyle', '-', 'LineWidth', 1);
hold all
plot(rundata.tperiod,prediction.Q, 'Color', 'b', 'DisplayName',
'Simulated', 'LineStyle', '-', 'LineWidth', 1)
set(gca, 'FontSize', 12);
datetick('x', 'mm/yy', 'keepticks', 'keeplimits');
legend('show');
xlabel('Time', 'FontSize', 12); ylabel('Streamflow (cms)', 'FontSize',
12);
```