Input and output scripts

MATLAB HEC-RAS Interface Documentation, IMTLD

Last update : 9/7/2020 1:01:00 PM – version BETA 1_0_1

This paper contains useful information to code a custom hydraulic rule in the MATLAB HEC-RAS interface.

An example which employs a lot of concepts presented in this paper is shown in the video tutorial 1 – Simple automation example.

Table of contents

Common features between scripts

Variables of master function

You can obtain detailed documentation about every variable presented in the following paragraph, by writing in the MATLAB command window, after execution of a version of the interface, help <variable name>, and then by clicking on hyperlink, like in the screenshot below :

```
>> help Results
  Results class def
  to store and save results obtained at each HEC-RAS computation
  because HEC-RAS doesn't save previous results, HEC-RAS overwrite
  previous results

    Documentation for Results
```

List of available variables

Table 1 – master's variables

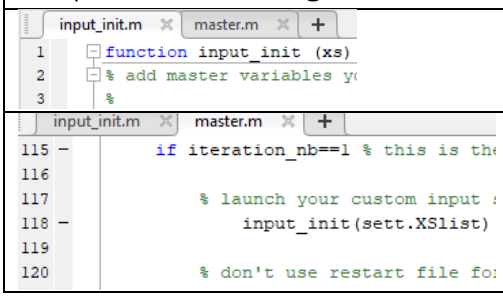
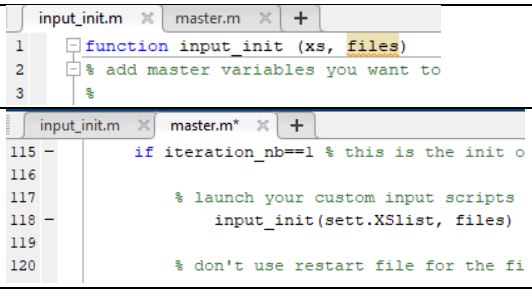
syntaxe to call it	description
sett	store every setting given in settings.ini
files	store every useful Ras filename e.g. unsteady flow file
results	Cell array of results objects, which gather results for a given XS
rp	RAS process, some functions need to talk to HEC-RAS directly
start	date of the beginning of the current simulation step
first	date of beginning of the first simulation step
last	date of ending of the last simulation step
end	Date of ending of the current simulation step
time_step	Duration of a simulation step
RASnewline	Specific newline character
iteration_nb	number of the current simulation step

Tableau 2 – useful subvariables

syntax to call it	description
sett.XSlist	List of XS choosed in settings.ini
results{<numéro de XS>}.Array	Results array for a given XS

To call a new variable in a input or output script, you have to change two code lines : in the call of the input or output function in the master function, and in the input or output function code file itself.

Example : add the files variable to your input_init script

Script before adding files variable	Script after adding files variable
	

Note : the sett.XSlist variable of the master script change name and is called XS in the input_init function. Because there is no need to have the same name for local variable and main variable. Cf. any programming language course or MATLAB internal documentation.

Input scripts

Input scripts are

- Input_before_init, which is a function called only before the first step (the simulation step during which a restart file is created for the next step, but which is not based on a restart file because it is the first)
- Input_before_step which is a function called before every step (except the first one)

Code structure

Steps that can follow your code in an input script :

Let variableX a variable that can be any variable from the master script, X an integer.

1. Get or instantiate variable1 with a useful value (e.g. water stage measured during previous step.
2. Logical rule, which change the value of variable2, which will be written in HEC-RAS files depending on the value of variable1.
3. Save into input_list MATLAB cell array, with parameters you want. To make MATLAB able to understand where in the Ras files, it must save variable2.

If you encounter any error, these errors can happen in a low level function, i.e. which are called by a lot of other functions. In this case, first check arguments given to input_list, before trying to edit a low level function. Keep in mind that they have been tested and they worked on other Ras projects. If, despite that, you think the function must be edited, please refer to Dev Doc paper.

Input_list object

This object is one of the main interfaces between your code and the MATLAB HEC-RAS interface code. It is very important to use parameters presented below, to avoid any issue.

Data is a variable and supports different types of values depending on every other parameters.

Additional information is an unused field for the time being. But it should be used if the MATLAB HEC-RAS interface supports editing of new types of parameters.

For the <empty> values, you can reduce the cell array size of input_list if there is a column or a row full of <empty>. You can too write what you want (ASCII) in this cell, its content will not be considered.

Possible values for main parameters

In this array, parameters ref_param and xs are not listed, because they have to much different values possible.

The possible values of ref_param are available further in this paper, in the table 5.

It is not useful to give a list of possible values of xs, because cross section name possibilities are quite endless. To know how to instantiate a XS object, please refer to the integrated doc of this class (in the MATLAB code, or by writing help XS in the command window after an execution of launcher and any version).

Table 3 – possible values of main parameters

Full name	Type [1]	Ras file extension	Additional information
Name in script	Type	Fileext	other_info
Possible values	hydrograph	.u	<empty>
	param	.u	<empty>
	param	.p	<empty>

Tableau 4 - valeurs possibles de la variable data en fonction du paramètre ''type''

Type	supported data format
hydrograph	Cell_array with one row or column with datetime objects or date strings [2] and another one with numbers
hydrograph	Numeric array
param	string [3]

Types description

Hydrograph = array of numeric data and eventually dates

Param = a standalone param, this string must appear only one time in the target Ras file.

If you want to edit a subordinate param (called ref_param, or if it is a two times subordinate param, called sub_param). Please refer to Dev Doc, dependencies between parameters paragraph.

Possible combinations of parameters

In this table, list of combinations tries to be exhaustive, but it is possible that some combinations not tested work. If you find some, you can contact us to share your discovery.

[1] Type is a MATLAB HEC-RAS interface param, defined to help the input_writing function recognize data types
[2] Date strings must be in a format immediately recognized by the datetime function, like the date syntax used in settings.ini
[3] This string will be written as is in Ras files, MATLAB does not check anything about this string. Do not include a newline character in this string, a RASnewline character will be automatically added.

Table 5 – Possible combinations of parameters

Full name	Type (for the interface)	Ras file extension	Name of the reference parameter	Cross section where this parameter applies	Additional information
Name in script	Type	Fileext	Ref_param	XS	other_info
Possible values	hydrograph	.u	Flow hydrograph	XS class object, not on a structure	<empty>
	hydrograph	.u	Stage hydrograph	XS class object, not on a structure	<empty>
	hydrograph	.u	Stage and Flow Hydrograph	XS class object, not on a structure	<empty>
	hydrograph	.u	Gate Name=Gate #1	XS class object, **on** a structure	<empty>
	hydrograph	.u	Uniform Lateral Inflow Hydrograph	XS class object, not on a structure	<empty>
	rating curve	.u	Rating Curve	XS class object, only on a boundary	<empty>
	param	.u	Any string which appears only once in the target file	<empty>	<empty>
	param	.p	Any string which appears only once in the target file	<empty>	<empty>

Output scripts

Output scripts are :

- output_after_step called after each Ras simulation step
- output_after_all called only one time, at the very end of the master function

Code structure

Here is an example of simple code structure, but you do not have to follow this pattern in every case. There is a lot of computations, that you can perform after a simulation time step.

1. Global variables declaration, which will be used in the input script
2. Eventually, computation on data extracted from results MATLAB object
3. Store data in variables for later use, e.g. in input scripts

results object

results is a cell array of Results_ objects. So, results{1} give you access to a Results_ object, the number 1 is related to the XS number chosen in settings.ini. There, results{1} contains results at every simulation step of the XS saved in the XS1 variable of the settings.ini file.

But during call to GetValue method of the settings object, you have to give the xs{1} argument, because the Results_ object is not aware of his cell number (there is potential improvement there).

xs object

xs is a cell array structured like results, xs contains XS objects given in settings.ini. Which is cross section location information.

To get the xs number 1, you just have to get the cell number 1 in the xs array, like this : xs{1}.

rp object

this is a RAS process

Warning : Ras must run in background when MATLAB writes in Ras files.

Else, Results_ methods will be unable to ask information to Ras. That is why rp is required.

How to use the output_after_all script – concrete example

This example is based on an idea of Dr. Duviella, Professor at IMT Lille Douai

In the case of a real system automation, if you compute simulations for 24 hours, with a simulation time step of 10 minutes, it will produce a big amount of data. It can be handy, to restart a new global simulation, e.g. a master restart, based on the results you obtain during the last master execution. Because instead of keeping in RAM, data of previous days, they will be saved on hard drive.

You can, for instance, call an auto-launch script (not included in the interface code), which will save results in a spreadsheet and read thanks to input_before_init the content of this spreadsheet to obtains good initials conditions.