Matlab/Simulink Benchmark Implementation

by

Ulf Jeppsson, IEA, LTH, Sweden

Note that this document may not be completely up to date but may still provide some hints on how to use the models. It is related to the original BSM1, which has now been somewhat updated. However, most of the principles still apply.

INTRODUCTION

The benchmarkfiles provided here are an exact implementation of what is described for the benchmark project at

http://www.benchmarkwwtp.org

Note there are some small differences compared to an earlier version of BSM1 which was used to generate the results described in the COST Benchmark book (Copp *et al.*, 2002). You should first study the documents on the web site very carefully so that you realise what this implementation is supposed to do. The provided models should work fine with Matlab R2019b and later versions. You do not need a separate C-compiler, instead just use the built-in Matlab C-compiler (MinGW compiler that can be downloaded from Matlab Add-Ons).

UNPACKING THE FILES

The files have been packed using *zip*. Just unzip the file using whatever software you normally use for this purpose.

FILE DESCRIPTIONS

When the files are unpacked you will find in the main directory (*benchmark*) the files that are associated with the closed loop approach, i.e. when we are simulation the WWT plant using active controllers. In the subdirectory *openloop* the files that are used for simulating the plant without active control are available. In the subdirectory *influent_files* the various influent data files are provided in ascii format in case you have to regenerate them. They are also available for downloading from the Benchmark home page.

Three simulink models are available:

- *openloop.mdl* – simulate the plant without control, i.e. in open loop;
- *benchmarkss.mdl* – simulate the plant with active controllers (closed loop) but with no noise and delay on the measurement signals. This model is used to quickly determine the quasi-steady state values before the actual benchmark is run);
- *benchmark.mdl* – simulate the plant with active control (closed loop) including noise and delays.

Five C-files are provided:

- *asm1.c* – C-file containing the AS Model no. 1;
- *settler1dv4.c* – C-file for a 10-layer one-dimensional settler model;
- *combiner.c* – adds two separate flows into one based on loads;
- *carboncombiner*.c – adds the external carbon flow to the rest of the wastewater;
- *hyddelayv3.c* – a special delay function (a very fast first-order exponential filter) to avoid algebraic loops.

These five files must be compiled on your local machine using the Matlab *mex* command before you can use the models (use the *mexall* script).

Four input data files are provided (+ a sensor noise file):

- *constinfluent.mat* – the constant value influent file;
- *dryinfluent.mat* – dry weather data file for two weeks (15 min samples);
- *storminfluent.mat* – storm weather data file for two weeks (15 min samples);
- *raininfluent.mat* – rain weather data file for two weeks (15 min samples).

These files must probably be recreated since the binary data format in Matlab normally differs somewhat between different platforms. In the subdirectory *influent_files* you may run the m-file *convert*. This will create new binary versions of the data files, and save them into the main directory. All data files contain 15 columns in the following order:
time, Si, Ss, Xi, Xs, Xbh, Xba, Xp, So, Sno, Snh, Snd, Xnd, Salk, flow

RUNNING THE BENCHMARK

When the C-files have been recompiled and the data files have been regenerated you are ready to run the benchmark. The idea is to put the path to the benchmark directory into the Matlab path and then change directory (Matlab command *cd*) into the openloop or influent_files subdirectories if you need to work here.

- start Matlab (it is a good idea to start it from the benchmark directory);
- command *benchmark* (the simulink window will appear);
- command *benchmarkinit* (initiates all variables and parameters, loads the data files etc. The file *asm1init.m* contains all variables and parameters with regard to the ASM1 model, *settlerinit.m* everything for the settlermodell, *reginit.m* everything for the controllers and so on). If you are running *openloop* from that subdirectory you instead initiate the system with the command *openloopinit*;
- write in the name of the input data file you want to use (CONSTINFLUENT, DRYINFLUENT, RAININFLUENT or STORMINFLUENT);
- in the simulink window parameters select simulation time, how often data should be stored etc. (if you are running *openloop* or *benchmarkss* with constant input use the ode15s solver but use ode45 if you use dynamic input data or active noise);
- start a simulation and let it finish. To simulate 14 days should take between 1 and 5 minutes depending on your computer performance (if you run *openloop* or *benchmarkss* with constant input it should only take a few seconds to simulate 100-150 days and determine a steady state);
- after a simulation all data are stored in the Matlab workspace and not to files. Use the *who* command to see what variables you have available.
- Some minor differences can occur when comparing with the values in the excel spreadsheet (Results folder). Follow the steps in the excel before comparing the results.

Some useful commands for analysis are also available:

- *statevalues* – this command will print on the screen a list of all values for all state variables for the last time sample of the latest simulation (*statevaluesol* for the openloop case);
- *stateset* – this command saves all the variables that are printed by *statevalues* into a file called *states.mat* and also sets them in the workspace so the benchmark is ready for a new simulation (this is very useful when you want to rerun simulations for different cases and want to be sure that you always start from the same initial values, then you just load the *states.mat* file and starts the simulation). Use *statesetol* for the openloop case;
- all variables can be easily plotted using the commands of the type *SHNplot*, *SSplot*, *sensorplot*, etc. (take a look at all m-files whose names end with plot.m). You provide the *starttime* and *stoptime* for the plots by setting the values of two variables with these names in the Matlab workspace;
- *perf_plant* – prints to the screen the values of all performance criteria related to the overall plant according to the definitions on the web site;

- *perf_controller* – the same as perf_plant but prints to the screen the performance of all the controllers and their control handles.

In principle all parameters and variables are defined in different m-files and the actual Simulink windows and the models therein never need to adjusted (unless you want to rebuild the plant layout or something similar). Within Simulink you only define parameters that are related to the solver and the name of which input data file should be used.

FINAL COMMENTS

Read the document about benchmarking from the web site carefully. Try to understand the structure of the different m-files to understand how they relate to each other. Try to run both *openloop*, *benchmarkss* and *benchmark* to make sure everything works. You can compare your results with the results provided on the web site and results provided in the directory Results.

When you feel confident that you understand this benchmark implementation you may start to add your own control strategies and try to improve the overall performance of the plant.

Good luck!