ARCH 2017 Competition Falsification Benchmarks

In this folder, the powertrain model for the falsification benchmark is presented. The results of this experiment are presented in "Report.pdf".

Powertrain control benchmark provides a high complexity model of an automatic air-fuel control system. The system should satisfy different requirements. In this report, we tested the system over the following specification:

$$\phi_{PB} = \square_{(\tau_s,T)}((rise(a) \vee fall(a)) \rightarrow \square_{(\eta,\zeta)}\mu)$$

This Specification implies that whenever event rise or fall happens (this happens if the input value changes from 8.8 to 40) then the state μ should remain in the specified bound after the settling time η and before any other change is made to the input value. More information about the model, specification, and methods to find counter-examples is provided in the report.

1- System Requirements and Instructions

The reported results are based on a system with 64-bit Intel Xeon CPU (2.5GHz), 64-GB RAM and Windows Server 2012.

To run the experiments the following software should be installed:

1) Matlab 2015a (64-bit) (or newer[1])
2) (Optional in case you would like to check out S-Taliro from the SVN repository instead of downloading a zip file) SVN client such as TortoiseSVN (https://tortoisesvn.net/)
3) Microsoft Visual C++ 2013 (or any other C/C++ compiler compatible with Matlab mex)
4) Configure Matlab MEX to use the C/C++ compiler with the command:

```
>> mex -setup
```

   a. More information about the C/C++ compiler setup can be found in the following:

[1] S-TaLiRo works with any Matlab version later than 2010b. However, depending on the Matlab version and the versions of the other external packages, the results in the report may only be reproducible in the statistical sense. S-Taliro sets the pseudo-random number generators to the seed that was used to generate the results in the report. However, external packages may be using their own random number generators. Also, external tools like MPT may use different optimization solvers by default on different machine architectures, resulting again in divergence between the reported results and the actual results produced by S-TaLiRo.

https://www.mathworks.com/help/matlab/matlab_external/changing-default-compiler.html

To install S-TaLiRo toolbox on Matlab:

1) Check out S-TaLiRo SVN public repository using the following URL:
   https://subversion.assembla.com/svn/s-taliro_public/
   or download the zip file from the same link.
2) Locate the "trunk" folder of "s-taliro_public" folder in Matlab.
3) Install S-TaLiRo inside Matlab with the following command:

```
>> setup_staliro
```

In addition, the following Matlab packages should be installed:

1) Matlab BLG package is downloadable from the following:
   http://www.mathworks.com/matlabcentral/fileexchange/10922-matlabbgl
2) Matlab toolbox manager is available at (http://www.tbxmanager.com/).
   Using tbxmanager, the following packages should be installed:

```
cddmex, fourier, glpkmex, hysdel, lcp, mpt, mptdoc, sedumi, yalmip
```

2- List of files in the ARCH reproducibility package

1) `AbstractFuelControl_M1.mdl`

The Simulink model of the Powertrain Control system used to simulate system trajectories.

2) `BlackBoxAbstractFuelControl.m`

This script runs the Simulink model and extracts the necessary information for S-TaLiRo to search for appropriate parameters.

3) `load_specs_and_model.m`

This script initializes the environment for S-TaLiRo search by providing the STL formula, input bounds, and the other information.

4) `PulseInputSignal.m`

Given 3 level values and 2 timestamps for transitions, this script creates pulse signals. It is used for the experiment that corresponds to Table 1, Row 3 (P-SA).

5) `staliro_arch_competition_experiments.m`

This script reproduces the reported results. It asks which results (Table and Row) to reproduce and then uses either S-TaLiRo General Falsification approach (Table 1) or Vacuity Aware Falsification method (Table 2) by either Uniform Random Sampling (Row 1, UR) or Simulated Annealing (Row 2, SA) optimization method. The user also has the option to choose falsification using pulses (Table 1, Row 3, P-SA).

6) `seeds_VAF_SA.mat , seeds_VAF_UR.mat`

Seeds for reproducing the results of Table 2 in the Report file.

7) `FinalResults.m`

This script creates the report on the falsification and runtime after the simulation.

3- Instructions For Running the Experiments

The experiments produce the outputs that correspond to the results of each Table entries in the report. In this experiment, the user chooses one of the two reported Tables (from Report) and its corresponding row of the table. Then, the system starts the simulation. At the end of the simulation, the reported results for the entries will be reproduced and the runtime information will be provided.

To run the experiments in the package folder, run:

```
>> staliro_arch_competition_experiments

This is the set of benchmark problems as presented in
the ARCH 2017 competition benchmarks paper.

Press any key to continue
```

Then the interface asks the user to enter the number of the reported table:

```
Select a reported falsification method
 1. Table 1: General Falsification.
 2. Table 2: Vacuity Aware Falsification.
Select an option (1-2):
```

The user should choose Table 1 or 2 to reproduce its results in the Report.

If the user chooses Table 1, three options will be provided for each row

```
Select the optimization method
 1. Table 1: Uniform Random Sampling (UR).
 2. Table 1: Simulated Annealing (SA).
 3. Table 1: Pulse Input Signal using Simulated Annealing (P-SA).
```

If the user chooses Table 2, two options will be provided for each row

```
Select the optimization method
 1. Table 2: Uniform Random Sampling (UR).
 2. Table 2: Simulated Annealing (SA).
```

Then, the experiments will start to falsify the following STL specification:

```
[]_(11,inf)(((low/\<>_(0,0.05)high) \/ (high/\<>_(0,0.05)low))-> []_[1,5](utr /\ utl))
```

Which is presented in the Report as

$$\phi_{PB} = \square_{(\tau_s,T)}((rise(a) \wedge fall(a)) \rightarrow \square_{(\eta,\zeta)}\mu)$$

It is expected that experiments for each table row would take several hours.

4- Output Results

The final results for each table row will be as follows. For example for Table 1 and Row 1 (UR) we have

```
************************************************************
 Reporting the Results of Table : 1 , Row 1
************************************************************
 Number of falsifications : 7/50
 Minimum Number of Tests : 18
 Maximum Number of Tests : 93
 Average Number of Tests : 52.1429
************************************************************
 The Robustness Results for the unfalsified experiments:
 Minimum Robustness : 1.694e-05
 Maximum Robustness : 0.0034946
 Average Robustness : 0.00088051
************************************************************
```

The above results reproduce the Table 1 Row 1 (UR) results.
In addition, we report the runtime execution (in Seconds) as follows:

```
 The Running Time of each Falsification Attempt:
 Minimum Runtime : 83.1781
 Maximum Runtime : 486.0754
 Average Runtime : 389.2304
************************************************************
```

It is expected that the runtime would be different.

The results of the other table entries are provided as follow:

Table 1 and Row 2 (SA):

```
************************************************************
 Reporting the Results of Table : 1 , Row 2
************************************************************
 Number of falsifications : 9/50
 Minimum Number of Tests : 13
 Maximum Number of Tests : 83
 Average Number of Tests : 50
************************************************************
 The Robustness Results for the unfalsified experiments:
 Minimum Robustness : 3.5382e-05
 Maximum Robustness : 0.0042019
 Average Robustness : 0.0012087
************************************************************
 The Running Time of each Falsification Attempt:
 Minimum Runtime : 54.385
 Maximum Runtime : 493.2527
 Average Runtime : 380.5529
************************************************************
```

Table 1 and Row 3 (P-SA):

```
************************************************************
 Reporting the Results of Table : 1 , Row 3
************************************************************
 Number of falsifications : 4/50
 Minimum Number of Tests : 34
 Maximum Number of Tests : 80
 Average Number of Tests : 55
************************************************************
 The Robustness Results for the unfalsified experiments:
 Minimum Robustness : 7.4114e-06
 Maximum Robustness : 0.0050803
 Average Robustness : 0.0015583
************************************************************
 The Running Time of each Falsification Attempt:
 Minimum Runtime : 141.6838
 Maximum Runtime : 452.1846
```

```
 Average Runtime : 384.7865
************************************************************
```

Table 2 and Row 1 (UR):

```
************************************************************
 Reporting the Results of Table : 2 , Row 1
************************************************************
 Number of falsifications : 9/50
 Minimum Number of Tests : 12
 Maximum Number of Tests : 96
 Average Number of Tests : 63
************************************************************
 The Robustness Results for the unfalsified experiments:
 Minimum Robustness : 3.4e-06
 Maximum Robustness : 0.0030185
 Average Robustness : 0.00085578
************************************************************
 The Running Time of each Falsification Attempt:
 Minimum Runtime : 35.8824
 Maximum Runtime : 556.7752
 Average Runtime : 433.9953
************************************************************
```

Table 2 and Row 2 (SA):

```
************************************************************
 Reporting the Results of Table : 2 , Row 2
************************************************************
 Number of falsifications : 29/50
 Minimum Number of Tests : 7
 Maximum Number of Tests : 95
 Average Number of Tests : 39
************************************************************
 The Robustness Results for the unfalsified experiments:
 Minimum Robustness : 2.3783e-06
 Maximum Robustness : 0.0043493
 Average Robustness : 0.0012694
************************************************************
 The Running Time of each Falsification Attempt:
 Minimum Runtime : 27.0165
 Maximum Runtime : 587.1818
 Average Runtime : 311.2613
************************************************************
```