H REPEATABILITY EVALUATION

Contact: If any issues or inconsistencies arise, please do not hesitate contact Ivan Ruchkin at iruchkin@cis.upenn.edu.

The goal of this repeatability evaluation is to replicate the randomized analysis that produces the numbers in Tables 1 and 2. Each cell contains the mean (before ±) and the standard deviation (after ±) for each monitor. It will be replicated via 4 analysis scripts, one per each case study and per averages/standard deviations. These scripts run over the provided logs of confidence monitors collected from system executions.

This evaluation was originally performed on a Lenovo laptop with Ubuntu 18.04, Linux kernel 4.15.0-161-generic, Intel Core i7-6600U CPU, and 16 Gb RAM. It is recommended, although not necessary, that you use a Linux environment on your host machine. The Docker version should be at least 20.10. Stay connected to the Internet throughout the evaluation.

The evaluation consists of four steps: (i) set up a docker container, (ii) authenticate into Wolfram, (iii) run the scripts, and (iv) compare the script outputs to the numbers in Tables 1 and 2.

H.1 Setting up docker container

You are provided with the "coco-docker-image.tar" archive with a docker image "coco-docker:latest". You can load it to the docker with:

```
docker load --input coco-docker.tar
```

An alternative method of procuring the image is building it from the sources in "coco-docker-src.zip". To do that, unpack this archive, change into the directory with the Dockerfile, and execute:

```
docker build . -t coco-docker:latest
```

Once the image is loaded/built, start a container: [2]

```
docker run --rm -it coco-docker:latest
```

H.2 Logging into Wolfram

Upon the creation of the container using the command above, you will be asked to authenticate into the Wolfram platform. This is a necessary step in order to use our Wolfram language-based analysis. Follow these steps:

- Create a free Wolfram ID at https://account.wolfram.com/login/create.
- Accept the free license terms at https://www.wolfram.com/engine/free-license.
- Enter your Wolfram ID and password into the prompt that appears after starting a docker container.

Alternatively, you can use the following credentials that we created for this exercise:

- Wolfram ID: repeatability@protonmail.com
- Password: pasWxSeeNcRID0A#

These credentials may not work depending on anyone else using them at the same time. So this approach is *not guaranteed to always work*, but may prove to be a useful shortcut. It is always safer to create your own Wolfram ID an log in with it.

[2]Note that the '–rm' flag means the container will be deleted when you disconnect from it. Remove it if you want to create a persistent container.

H.3 Running the scripts

To reproduce the numbers reported in the paper, 4 scripts need to be run, one per left/right side of Table 1/2. Each script will produce a pair of CSVs files, one with the means and one with the standard deviations.

For the UUV case study (right side of the tables):

- Neutral calibration (Table 1, right side):

```
wolframscript uuv-analysis-neutral.wls
```

- Conservative calibration (Table 2, right side):

```
wolframscript uuv-analysis-conservative.wls
```

For the mountain car case study (left side of the tables):

- Neutral calibration (Table 1, left side):

```
wolframscript mountaincar-analysis-neutral.wls
```

- Conservative calibration (Table 2, left side):

```
wolframscript mountaincar-analysis-conservative.wls
```

The parameters for scripts (iteration counts, weights) can be set in the beginning of the .wls files. By default, they are set to 20 iterations as per the paper. If debugging is needed, it may be convenient to set the iteration count to 2.

During the analysis, you may get a notification "Loading from Wolfram Research server..." – it is normal for the Wolfram Engine to download its own libraries. Depending on your hardware, the analysis may take hours to finish. Please allow the time for that. This is especially true for the mountain car case study with a substantial number of traces.

In the process of the analysis, warnings about numeric precision and complex/symbolic arguments may appear. Ignore them: they are part of the normal workflow. However, if an overwhelming number of complex-looking errors shows up, then something is probably mis-configured.

H.4 Comparing the numbers

The scripts will produce a total of 8 CSV files: 4 with means and 4 with standard deviations. The files with "0.5" in their name refer to neutral calibration (Table 1) and the files with "0.8" in their name refer to conservative calibration (Table 2). As mentioned above, the means are compared to the numbers on the left of the ± sign, and the standard deviations are compared to the numbers on the right of the ± sign.

To view the CSV files conveniently, use the pretty_csv command, for example:

```
pretty_csv uuv_means_2_0.5.csv
```

This command can be exited with Ctrl+C.

Note that the scripts run a randomized procedure of cross-validation. So your exact numbers *will* differ from those in Tables 1 and 2. The replication is successful if the numbers only differ in the 2nd–3rd decimal digit, and the relative performance of the monitors is unchanged (e.g., the bold numbers remain the smallest/largest in their respective sub-column).