RW-Solver Manual

National Institute of Advanced Industrial Science and Technology (AIST)

8 February 2019

Contents

1   RW-Solver Command-Line Options

Option	Description
`-xs` ⟨x-scale⟩	specifies the scaling factor in the horizontal direction of the timetable. (0.1 ≤ ⟨x-scale⟩ ≤ 10.0)
`-ys` ⟨y-scale⟩	specifies the scaling factor in the vertical direction of the timetable. (0.1 ≤ ⟨y-scale⟩ ≤ 10.0)
`-t { pdf	svg }`	specifies the file format of the timetable (default: `pdf`).
`-o` ⟨filename⟩	specifies the file name of the timetable.
`-lz` ⟨basename⟩	saves the files for SMT-Solver by the basename ⟨basename⟩. (extensions `.z3in` and `.z3out` are automatically attached)
`-rz` ⟨filename⟩	reuses the output file (`.z3out`) for the generating timetable.
`-so`	is used for estimating the elapsed time in SMT-Solver. (no timetables is generated)
`-ptn`	prints track-IDs used between stations in the timetable.
`--sovler` ⟨solver-name⟩	specifies the SMT-Solver for solving constraints (default: z3).
`--help`	shows all the options.

2   Execution Environment for RW-Solver

It is explained how to construct the environment for executing the binary file `rwsolver` of RW-Solver in Ubuntu (Linux). The environment can be also constructed in "Windows Subsystem for Linux" on Microsoft Windows 10.

2.1   Required Software for RW-Solver

RW-Solver requires Z3 and Cairo, and software that they depend on.

Software	Description
Z3	SMT Solver
Cairo	Graphics library used for drawing timetables

2.2 Installing Required Software

Cairo has already been installed in Ubuntu. Z3 can be installed by the command `apt-get`

```
$ sudo apt-get install z3
```

3 Build Environment for RW-Solver

It is explained how to construct the environment for building the source files of RW-Solver .

3.1 Installing Software Packages

The following software is required.

- m4
- autoconf
- git
- aspcud
- libgtk2.0-dev
- opam
- z3

```
$ sudo apt-get install m4 autoconf git aspcud libgtk2.0-dev opam z3
```

3.2 Setting up OPAM

OPAM (a source-based package manager for OCaml) is set up as follows. See `https://opam.ocaml.org/` for the details of OPAM.

1. Initialize OPAM:

   ```
   $ opam init
   ```

2. Set environment variables:

   ```
   $ eval 'opam config env'
   ```

3. Change the version of OCaml to 4.03.0:

   ```
   $ opam switch 4.03.0
   ```

4. Set environment variables again:

```
$ eval 'opam config env'
```

5. Install the library Cairo for OCaml

```
$ opam install cairo
```

3.3   Building RW-Solver

Execute the command `make` in the directory including the source codes of RW-Solver :

```
$ make
```