Sparkle user guide

Koen van der Blom

January 28, 2020

1 Quick start

1.1 Installing Sparkle

1. Copy the Sparkle files to your desired directory

2. Install `epstopdf` as described in Section 3.1.1

1.2 Configuration

1. Make solver executable work on Grace

2. `sprakle_smac_wrapper.py`

3. Create and supply a PCS file

1.2.1 Making your algorithm run on Grace

Add helpful tips when available

Shell and Python scripts should work as is. If a compiled binary does not work, you may have to compile it on Grace and manually install packages on Grace that are needed by your algorithm.

1.2.2 Creating a wrapper for your algorithm

A template for the wrapper that connects your algorithm with Sparkle is available at `Examples/Resources/Solvers/template/sparkle_smac_wrapper.py`. Within this template a number of `TODO`s are indicated where you are likely to need to make changes for your specific algorithm. You can also compare the different example solvers to get an idea for what kind of changes are needed.

1.2.3 Parameter configuration space (PCS) file

The PCS (parameter configuration space) format[1] is used to pass the possible parameter ranges of an algorithm to Sparkle in a `.pcs` file. For an example see e.g. `Examples/Resources/Solvers/PbO-CCSAT-Generic/PbO-CCSAT-params_test.pcs`.

In this file you should enter all configurable parameters of your algorithm. Note that parameters such as the random seed used by the algorithm should not be configured and therefore should also not be included in the PCS file.

2 Settings

2.1 Slurm (focused on Grace)

Slurm settings can be specified in the `Settings/sparkle_slurm_settings.txt` file. Currently these settings are inserted as is in any **srun** or **sbatch** calls done by Sparkle. This means that any options exclusive to one or the other currently should not be used (see Section 2.1.2).

2.1.1 Tested options

Below a list of tested Slurm options for **srun** and **sbatch** is included. Most other options for these commands should also be safe to use (given they are valid), but have not been explicitly tested. Note that any options related to commands other than **srun** and **sbatch** should not be used with Sparkle, and should not be included in `Settings/sparkle_slurm_settings.txt`.

```
--partition / -p
--exclude
--nodelist
```

2.1.2 Disallowed options

The options below are exclusive to **sbatch** and are thus disallowed:

```
--array
--clusters
--wrap
```

The options below are exclusive to **srun** and are thus disallowed:

```
--label
```

---

[1]See: `http://aclib.net/cssc2014/pcs-format.pdf`

2.1.3 Nested `srun` calls

A number of Sparkle commands internally call the `srun` command, and for those commands the provided settings need to match the restrictions of your call to a Sparkle command. Take for instance the following command:

```
srun -N1 -n1 -p graceTST Commands/configure_solver.py
   -solver Solvers/Yahsp3 -instances-train Instances/
   Depots_train_few/
```

This call restricts itself to the `graceTST` partition (the `graceTST` partition only consists of node 22). So if the settings file contains the setting `--exclude=ethnode22`, all available nodes are excluded, and the command cannot execute any internal `srun` commands it may have.

Finally, Slurm ignores nested partition settings for `srun`, but not for `sbatch`. This means that if you specify the `graceTST` partition (as above) in your command, but the `graceADA` partition in the settings file, Slurm will still execute any nested `srun` commands on the `graceTST` partition only.

3 Required packages

3.1 Sparkle on Grace

Grace is the computing cluster of the ADA group[2] at LIACS, Leiden University. Since not all packages required by Sparkle are installed on the system, some have to be installed local to the user.

3.1.1 `epstopdf`

The `epstopdf` package is required for Sparkle's reporting component to work (e.g. `generate_report, generate_report_for_configuration`), it can be installed in your user directory as follows:

1. Download `epstopdf`

   `wget http://mirrors.ctan.org/support/epstopdf.zip`

2. Unzip the package

   `unzip epstopdf.zip`

3. Rename `epstopdf.pl` (inside the directory you just unzipped)

   `mv epstopdf.pl epstopdf`

4. Add this line to your `.bashrc` (open with e.g. `vim ~/.bashrc`)

   `export PATH="/<directory>/epstopdf:$PATH"`

   (replace "`<directory>`" with the path to the `epstopdf` directory)

---

[2]`http://ada.liacs.nl/`

3.2   Yahsp example

3.2.1   Compiling Yahsp

Check what of the below is needed when the yahsp binary is provided

1. Install gmp on Grace

   `wget https://gmplib.org/download/gmp/gmp-6.1.2.tar.xz`

   `tar -xf gmp-6.1.2.tar.xz`

   Inside the `gmp-6.1.2` directory:

   `./configure`

   `make`

   `make check`

2. Navigate to the `seq-agl-yahsp3` directory

3. Add the below after `-fpermissive` on line 24 of `cpt-yahsp/CMakeLists.txt`:

   `-I /home/blomkvander/lib/gmp-6.1.2/ -L /home/blomkvander/lib/gmp-6.1.2/.libs/`

   (replace `/home/blomkvander/lib/` with the path where you installed gmp)

4. Compile yahsp with:

   `./build`

5. In `yahsp/esegui.sh` the line `#!/bin/bash` was added to the start of the
   file to allow Grace nodes to find the 'time' utility.