

Distributed Automatic Parameter Testing

Ben Duggan

February 8, 2019

1 Setup

1.1 Python dependencies

The python code uses gspread, oauth2client, flask and boxsdk. You can install them using ‘pip install gspread oauth2client flask boxsdk boxsdk[jwt]==2.0.0‘.

1.2 Google Sheet

The Google Sheet can be have any name you wish. The Google Sheet is used like a database with different sheets (tabs) acting as table, the first row of a sheet being the header and the following rows being the entries. You are only required to have one column which contains “id” and is used to reference specific parameter sets. There are several reserved column names which can be used and will be referenced or populated automatically. They are shown below:

- id: the trial id and must be unique.
- status: the current status of the parameter set. It can be empty meaning not started, in progress or finished.
- startTime
- endTime
- performedBy: the computer that performed the computation as defined by config.txt
- comment: any comments either added by the person entering values or main.py
- computerStrength: if the computer strength value for a parameter set is greater than the computer strength setting in the config.txt then the set won’t be ran on that computer.

You can add any other columns you wish and can access them in the parameter set returned by `dap.param.Param.requestParameters()`.

1.3 sheet.py

Using the Google Sheets API Python Startup Guide and Google Spreadsheets and Python guide written by Greg Baugues.

Follow the following steps:

1. Ensure that the dependencies are installed.
2. Create the credentials (this is only needed to create a new set of parameters).
 - (a) Go to <https://console.developers.google.com/>.
 - (b) Create a new project.
 - (c) Click Enable API. Search for and enable the Google Drive API and the Google Sheets API.
 - (d) Create credentials for service account key. Create a service account and select a Key type of JSON.
 - (e) Name the service account and grant it a Project Role of Editor.
 - (f) Download the JSON file.
 - (g) Copy the JSON file into DistParam.
3. Go to the spreadsheet you wish to use, copy spreadsheet ID (The ID in <https://docs.google.com/spreadsheets/d/17QJFFXto0MbOX5dH9GFP3NevNHluxj7eZf7Pevcg96U/edit#gid=0> is **17QJFFXto0MbOX5dH9GFP3NevNHluxj7eZf7Pevcg96U**) and set it equal to spreadsheetID in sheet.py

1.4 config.txt

This is the configuration file for the main file. It consists of key value pairs separated by a colon. There can't be a space between the key or value and the colon. It consists of the following fields:

- `userName` this is the user name of the system. It is not required but gets saved in the Google Sheet and allows for runs to be traced back to a particular system. Its default value is "default".
- `resetTime` is the time in seconds at which a parameter set will be added as available if the status hasn't changed from "in progress" to "finished". This is to ensure that if the computer shuts down or an error occurs the parameter set will still be run. The default value is "172800" or 2 days.
- `numOfRuns` is the number of trials you want main to run. If you set it to -1 it will run until there are no more parameters to run. The default value is "-1".
- `lastTest` is used by main.py to keep track of if a test didn't finish correctly. If it did not then it will try this set of parameters the next time. Its default value is "None".

2 Run Test

To run you must simply type `python main.py`

3 How it works

When you run `sheet.py` you enter a while loop that breaks when either there are no more parameters or you have reached the maximum number of parameters to run as defined in `config.txt`. At the beginning of the while loop block code checks to see if the DB has errors. Currently this is just checking to see if the code has been running for more than the time given in `config.txt`. Next the code fetches the next set of parameters to use.