Cheat-Sheet for `tools-for-g16.bash` (0.1.0, 2019-02-04)

Martin C Schwarzer, February 4, 2019

Introduction

This accompanies the repository polyluxus/tools-for-g16.bash.

Various bash scripts to aid the use of the quantum chemistry software package Gaussian 16.

Preliminary notes

The notation in brackets `[ ]` indicate optional arguments/inputs; arguments in angles `< >` require human input; a bar `|` indicates alternatives.

The following abbreviations will be used:

`opt` Short for option(s)
`ARG` String type argument
`INT` Positive integer (including zero)
`NUM` Whole number (including zero)
`FLT` Floating point number
`DUR` Duration in format `[[HH:]MM:]SS`

Installation & Configuration

General settings for the scripts can be found in the file `g16.tools.rc`. Alternatively, settings can be stored in `.g16.toolsrc`, which always has precedence. Every script will check three different directories in the order 1. installation directory 2. user's home 3. parent working directory. It will load the last configuration file it finds.

Setting files can be generated with the `configure/configure.sh` script.

g16.prepare.sh

This tool reads in a file containing a set of cartesian coordinates and writes a Gaussian inputfile with predefined keywords. The script interfaces to Xmol format, Turbomole/ GFN-xTB `coord` format, too.

Usage: `g16.prepare.sh [opt] <file>`

`-T <FLT>` Temperature (kelvin)
`-P <FLT>` Pressure (atmosphere)
`-r <ARG>` Add `ARG` to route section
`-R <ARG>` Specific route section `ARG`
`-l <INT>` Load predefined route section
`-l list` Show all predefined route sections
`-t <ARG>` Adds `ARG` to end of file
`-C <ARG>` Specify caption/title of job; Replacements: `%F` input filename; `%f` input filename without `.xyz`; `%s` like `%f`, also filtering `start`; `%j` jobname; `%c` charge (with indicator `chrg`); `%M` multiplicity (with indicator `mult`); `%U` unpaired electrons (with indicator `uhf`).
`-j <ARG>` Jobname
`-j %f` Jobname is filename filtering `.xyz`
`-j %s` Jobname is filename filtering `start.xyz`
`-f <ARG>` Filename of generated input
`-c <NUM>` Charge
`-M <INT>` Multiplicity (≥ 1)
`-U <INT>` Unpaired electrons (≥ 0)
`-m <INT>` Memory (megabyte)
`-p <INT>` Processors
`-d <INT>` disksize via `MaxDisk` (megabyte)
`-s` Silence script (incremental)
`-h` Help file

g16.testroute.sh

This tool parses a Gaussian 16 inputfile and tests the route section for syntax errors with the Gaussian 16 utility `testrt`.

`-s` Silence script (incremental)
`-h` Help file

g16.dissolve.sh

This tool reads in a Gaussian 16 inputfile and adds relevant keywords for solvent corrections.

Usage: `g16.dissolve.sh [opt] <file>`

`-o <ARG>` Adds option `ARG` to the `scrf` keyword.
`-S <ARG>` Adds option `solvent=ARG` to the `scrf` option list.
`-O` Runs an optimisation (preserves or adds `OPT`)
`-r <ARG>` Add `ARG` to route section
`-t <ARG>` Adds `ARG` to end of file
`-m <INT>` Memory (megabyte)
`-p <INT>` Processors
`-d <INT>` disksize via `MaxDisk` (megabyte)
`-s` Silence script (incremental)
`-h` Help file

g16.freqinput.sh

This tool reads in a Gaussian 16 inputfile and adds relevant keywords for a frequency calculation.

Usage: `g16.freqinput.sh [opt] <file>`

`-o <ARG>` Adds option `ARG` to the `freq` keyword.
`-R` Adds option `ReadFC` to the `freq` option list.
`-T <FLT>` Temperature (kelvin)
`-P <FLT>` Pressure (atmosphere)
`-r <ARG>` Add `ARG` to route section
`-t <ARG>` Adds `ARG` to end of file
`-m <INT>` Memory (megabyte)
`-p <INT>` Processors
`-d <INT>` disksize via `MaxDisk` (megabyte)
`-s` Silence script (incremental)
`-h` Help file

g16.ircinput.sh

This tool reads in a Gaussian 16 inputfile from a frequency run and adds relevant keywords for two separate irc calculations.

Usage: `g16.ircinput.sh [opt] <file>`

`-o <ARG>` Adds option `ARG` to the `irc` keyword.
`-r <ARG>` Add `ARG` to route section
`-t <ARG>` Adds `ARG` to end of file
`-m <INT>` Memory (megabyte)
`-p <INT>` Processors
`-d <INT>` disksize via `MaxDisk` (megabyte)
`-s` Silence script (incremental)
`-h` Help file

g16.optinput.sh

This tool reads in a Gaussian 16 inputfile preferably from an IRC run and writes and inputfile for a subsequent structure optimisation.

Usage: `g16.optinput.sh [opt] <file>`

`-o <ARG>` Adds option `ARG` to the `opt` keyword.
`-r <ARG>` Add `ARG` to route section
`-t <ARG>` Adds `ARG` to end of file
`-f <ARG>` Filename of generated input
`-m <INT>` Memory (megabyte)
`-p <INT>` Processors
`-d <INT>` disksize via `MaxDisk` (megabyte)
`-s` Silence script (incremental)
`-h` Help file

g16.spinput.sh

This tool reads in a Gaussian 16 inputfile and writes and inputfile for a subsequent calculation. It is possible to overwrite the existing route section, but still add the `geom`/`guess` directives to base it on.

Usage: `g16.spinput.sh [opt] <file>`

`-r <ARG>` Add `ARG` to route section
`-R <ARG>` Overwites route section with `ARG`
`-t <ARG>` Adds `ARG` to end of file
`-f <ARG>` Filename of generated input
`-m <INT>` Memory (megabyte)
`-p <INT>` Processors
`-d <INT>` disksize via `MaxDisk` (megabyte)
`-s` Silence script (incremental)
`-h` Help file

g16.submit.sh

This tool parses and then submits a Gaussian 16 inputfile to a queueing system.

Usage: `g16.submit.sh [opt] <file>`

`-m <INT>` Memory (megabyte)
`-p <INT>` Processors
`-d <INT>` disksize via `MaxDisk` (megabyte)
`-w <DUR>` Walltime limit
`-e <ARG>` Specify an environment variable `ARG` in format `<VAR=value>`
`-j <INT>` Wait for job with ID `INT`
`-H` Submit with status hold (PBS) or `PSUSP` (BSUB)
`-k` Only create (keep) the jobscript, do not submit it.
`-Q <ARG>` Queue for which job script should be created (`pbs-gen`/`bsub-rwth`)
`-P <ARG>` Account to project (BSUB); if `ARG` is `default`/`0`/`''` presets are overwritten.
`-M <ARG>` Specify a machine type (BSUB); if `ARG` is `default`/`0`/`''` presets are overwritten.
`-u <ARG>` set user email address (BSUB); if `ARG` is `default`/`0`/`''` presets are overwritten.
`-s` Silence script (incremental)
`-h` Help file

g16.getenergy.sh

This tool finds energy statements from Gaussian 16 calculations.

Usage: `g16.getenergy.sh [opt] [<file(s)>]`

If no files given, it finds energy statements from all log files in the current directory.

`-i <ARG>` Specify input suffix if processing directory
`-o <ARG>` Specify output suffix if processing directory
`-L` Print the full file and path name (seperated by newline)
`-s` Silence script (incremental)
`-h` Help file

g16.getfreq.sh

This tool summarises a frequency calculation and extracts the thermochemistry data.

Usage: `g16.getfreq.sh [opt] <file(s)>`

`-v` Incrementally increase verbosity
`-V <INT>` Set level of verbosity directly, (0-4)
`-c` Separate values by comma (`-V0` or `-V1`)
`-f <ARG>` Write summary to file instead of screen
`-s` Silence script (incremental)
`-h` Help file

g16.chk2xyz.sh

A tool to convert a checkpoint file to an `xyz` file. This formats the `chk` first to a `fchk`.

Usage: `g16.chk2xyz.sh [-s] -h | -a | <chk-file(s)>`

`-a` Formats all checkpointfiles that are found in the current directory
`-s` Silence script (incremental)
`-h` Help file

Author, Bugs, and the Rest

Martin C Schwarzer Martin-マーチン polyluxus