Silico

Computational Chemistry Management

Quick Reference Guide

Version 0.17.1

14 January 2021

Table of Contents

How to Use This Document

This document is designed to act as rapid introduction and quick-reference to the Silico Computational Chemistry Management system.

Images are included for comparison and reference.

Supplementary information and helpful tips are displayed separately, as follows:

> **NOTE:** This is a tip.

Commands that should be typed by the user are displayed in the following format:

```
> echo Hello world
```

The arrow character (>) should not be typed; it is used to differentiate commands that should be typed from the resulting output (which will be displayed without the arrow character):

```
> echo Hello world
Hello world
```

In the above example, the user is being instructed to type: echo Hello world. The computer, in response, gives the output: Hello world.

Ellipses (…) indicate that the real, full output has been truncated:

```
> cat /etc/fstab
# /etc/fstab: static file system information.
#
...
```

About Silico

Silico is a set of programs designed to automate and simplify all aspects of computational chemistry, without which can be: confusing, frustrating, time-consuming, bizarre and/or upsetting.

Silico is under development and suggestions are welcome, please contact osl@st-andrews.ac.uk.

A non-exhaustive list of features is as follows:

- Submission to computational programs through a simple and unified interface.
- Simultaneous submission of multiple molecules/systems.
- Automatic in series submission of results from completed calculations to subsequent calculations.
- Automatic conversion of input files (including ChemDraw) to types appropriate for the selected computational program.
- Automatic and manual analysis of computation results, including tabulation to CSV format.
- Automatic and manual generation of PDF reports from computation results, including rendered 3D structures, orbital images and graphs.

Changes Since Previous Version (0.14.2)

The following major changes have been implemented in this version:

- **Spin-orbit coupling:** A modified version of PySOC[1] has been incorporated into Silico to permit calculation of singlet/triplet spin-orbit coupling (SOC). Currently, SOC can be calculated for TD and TDA calculations with both G09 and G16. See: Spin-Orbit Coupling (SOC).
- **Turbomole**: Support has been added for submitting calculations and extracting results with Turbomole. In addition to HF and DFT, higher-order calculations including MP2, MP3 and SCS-CC2 are supported, as well as molecular mechanics with the universal force field (UFF). Report generation is only partially supported, as orbital and molecular images cannot yet be generated. See: Turbomole SCS-CC2 and MR-TADF
- **Submission format:** A universal submission format (.si) has been implemented. In addition to containing atom coordinates, this format allows the charge and multiplicity to be set directly, useful for programs that do not support this with their own input formats (eg, Turbomole). Silico input files can now be generated from other formats using the ‘silico convert’ command. See: Universal Input Format (.si).
- **Charge and multiplicity**: Improved support for molecular charge and multiplicity has been added. These properties can be now set using the ‘-C’ and ‘-M’ options to ‘silico submit’.
- **Queue status**: The ‘silico status’ command has been added which prints how busy each submission queue is at the current moment. See: Checking The Queue.
- **Numerous bug fixes.**

Installation

To install Silico, connect to Kennedy using PuTTY and log-in to your account as normal. Once logged in, use the following command:

```
> /gpfs1/apps/EZC-tools/install-silico
```

NOTE: Installation only ever needs to be performed once (for each user). Updates to Silico will be become available automatically without the need to reinstall.

Installation should be near instantaneous. Once complete, a message will be printed:

```
> /gpfs1/apps/EZC-tools/install-silico
Installed successfully. Please log out and in again to complete
```

Use the 'exit' command to log out:

```
> /gpfs1/apps/EZC-tools/install-silico
Installed successfully. Please log out and in again to complete
> exit
logout
```

Once logged back in again, Silico will now be available for use. The installation can be tested by using the 'silico' command with the '-v' (**v**ersion) option. If successful, the version will be printed:

```
> silico -v
0.17.1
```

If the installation was not successfully, the following error message will be printed:

```
> silico -v
-bash: silico: command not found
```

If unsuccessful, seek help from another group member.

Submitting Calculations

Prepare Files

To submit a calculation, the system(s) of interest first need to be prepared by drawing the relevant structures electronically (on a program). Silico supports a wide variety of input file types, including both 2D and 3D formats. Notable entries include:

- G09 output (.log)
- GaussView (.com, .gjf, .gjc and .gau)
- ChemDraw (.cdx and .cdxml)
- Avogadro (.cml)
- MarvinSketch (.mrv)
- Crystallographic Information File (.cif)

NOTE: MarvinSketch (.cml) and Avogadro (.mrv) are free alternatives to ChemDraw and GaussView, respectively. You have probably already used MarvinSketch; it is the program used to draw structures on Reaxys.

Silico uses the OpenBabel library for file conversion. Please see https://open-babel.readthedocs.io/en/latest/FileFormats/Overview.html for a full list of supported formats.

NOTE: When using ChemDraw files (.cdx), remember to save each molecule individually to its own file. If a single file contains multiple structures, then all these structures will be submitted together to a single calculation.

In general, the closer to the final atom coordinates the input coordinates are, the fewer optimisation cycles will be required. As such, it is strongly recommended to favour 3D formats over 2D. If available, a Crystallographic Information File (.cif) or output file from a previous calculation (on the same or similar structure) should be favoured as input, for the same reason.

Upload Files

Once the structures have been prepared, transfer the input files to Kennedy using the WinSCP program. It is recommended that input files for related calculations be stored inside subfolders within your home directory on Kennedy.

Universal Input Format (.si)

Silico also supports a text-based, program-independent input format simply called the silico input format (.si). This format is useful because the input format native to certain computational programs, notably Turbomole, do not contain molecular charge or multiplicity information, but this information can be set in the .si format.

To create a .si file, use the ‘silico convert’ command to convert any file format supported by Silico:

```
> silico convert Benzene.com -O Benzene.si
```

The new file will be written to ‘Benzene.si’, and will have the following structure:

```
name: Benzene
charge: 0
multiplicity: 1
geometry: |-
  C          -1.20910        -0.69809         0.00000
  C          -1.20908         0.69806         0.00000
  C           0.00000        -1.39611         0.00000
  C           1.20909        -0.69804         0.00000
  C          -0.00000         1.39611         0.00000
  C           1.20910         0.69807         0.00000
  H          -2.14965        -1.24108         0.00000
  H          -0.00001        -2.48216         0.00000
  H          -2.14967         1.24098         0.00000
  H           0.00004         2.48216         0.00000
  H           2.14961         1.24116         0.00000
  H           2.14962        -1.24109         0.00000
```

This file can then be edited by the user, for example with the ‘nano’ program, to change the desired, name, charge and/or multiplicity. The input file can then be submitted as normal.

Submit Files

To submit files, use the ‘silico’ command, followed by the list of filenames to submit. For example, to submit a file in the current directory named ‘Benzene.com’:

```
> silico Benzene.com
```

NOTE: If you are used to submitting calculations on Kennedy, know that you do not need to write batch files with Silico; they are created automatically.

You can set an explicit charge and multiplicity using the ‘-C’ (**c**harge) and ‘-M’ (**m**ultiplicity) options. If given, these options will override any charge or multiplicity information in the input file.

```
> silico Benzene.com -M 3 -C 0
```

If the file name contains spaces, remember to use speech marks:

```
> silico "Methyl benzene.com"
```

If the file is not in your current directory, specify the full path to the input file (including directories):

```
> silico "Aromatic Folder/Methyl benzene.com"
```

NOTE: You can check what files are in the current directory by using the ‘ls’ command.

Alternatively, you can use the ‘cd’ command to change into the subdirectory before using silico:

```
> cd "Aromatic Folder"
> silico "Methyl benzene.com"
```

The directory in which the calculation is performed is created in the current directory, use the '-o' (**o**utput) option to change this default:

```
> silico "Aromatic Folder/Methyl benzene.com" -o "Aromatic Folder"
```

In the above example, the calculation would take place in the 'Aromatic Folder/Methyl benzene' folder.

To submit multiple files at once, separate each with a space:

```
> silico Benzene.com "Methyl benzene.com"
```

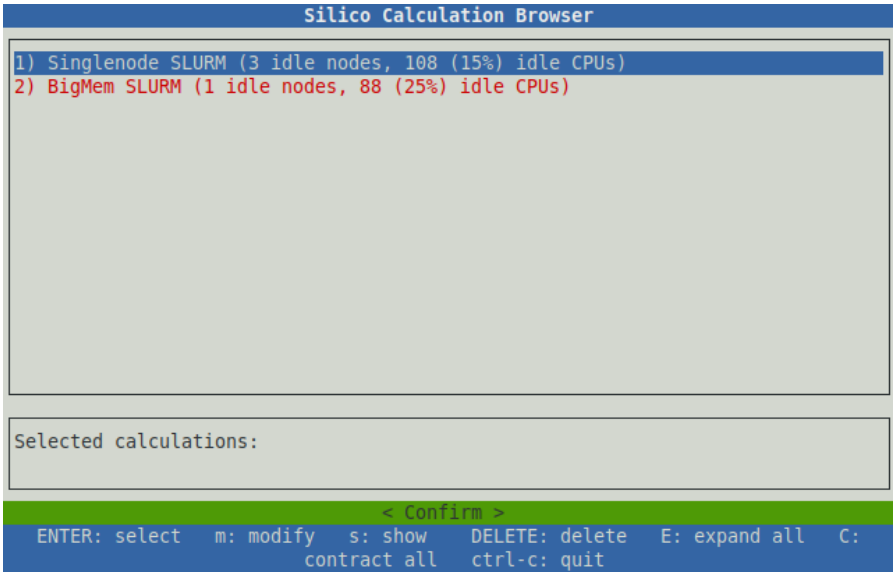
Finally, groups of files can be submitted by using the wildcard character (*). For example, to submit all files in the current directory ending in '.com':

```
> silico *.com
```

NOTE: All files submitted with the same silico command will be submitted to the same calculation. To submit to different calculations, use the silico command multiple times.

Calculation Browser

The Silico calculation browser will then open, from which a calculation to submit to can be chosen:



This browser is interactive; you can navigate up and down using the arrow keys. ‘Enter’ or ‘space’ can be used to expand or contract each item. ‘Esc’ or ctrl-c (the ‘ctrl’ key and the ‘c’ key pressed together) can be used to quit at any time.

Select a Method

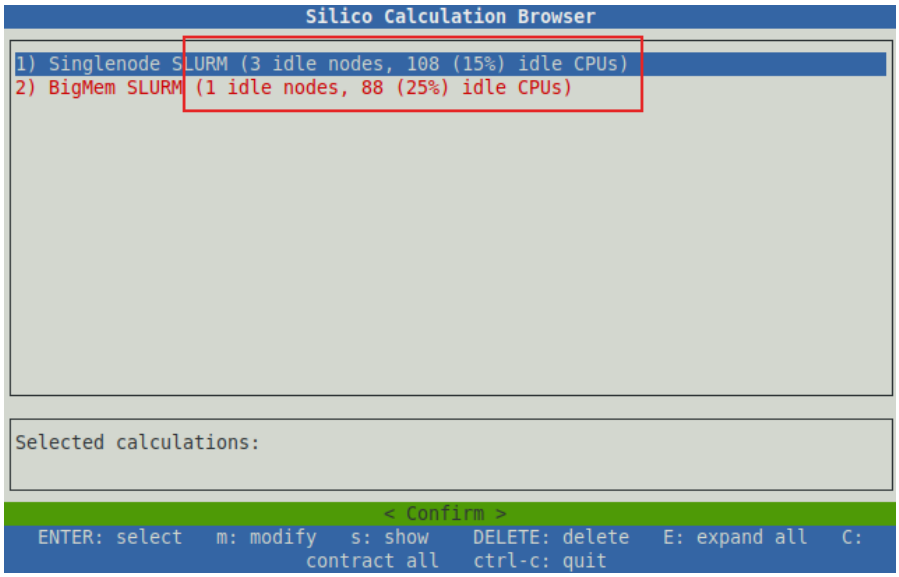
The first items to choose from are the calculation methods; these represent different groups (called partitions) of computing nodes (individual computers). Four are available at time of writing, they are:

- **Singlenode SLURM:** The default partition, **this should be chosen in nearly all cases**.
- **BigMem SLURM:** A partition containing nodes with greatly increased memory capacity. This method should only be used for very memory intensive calculations. **Please speak to a senior group member before selecting this partition.**
- **Debug SLURM:** A partition for testing purposes only. Debug has reduced resources and a time limit; it is not suitable for performing calculations.
- **Multi Node SLURM:** A partition intended for more intensive calculations that require multiple nodes. **Use of the multi-node partition is generally forbidden, you must speak to a senior group member before selecting this partition.**

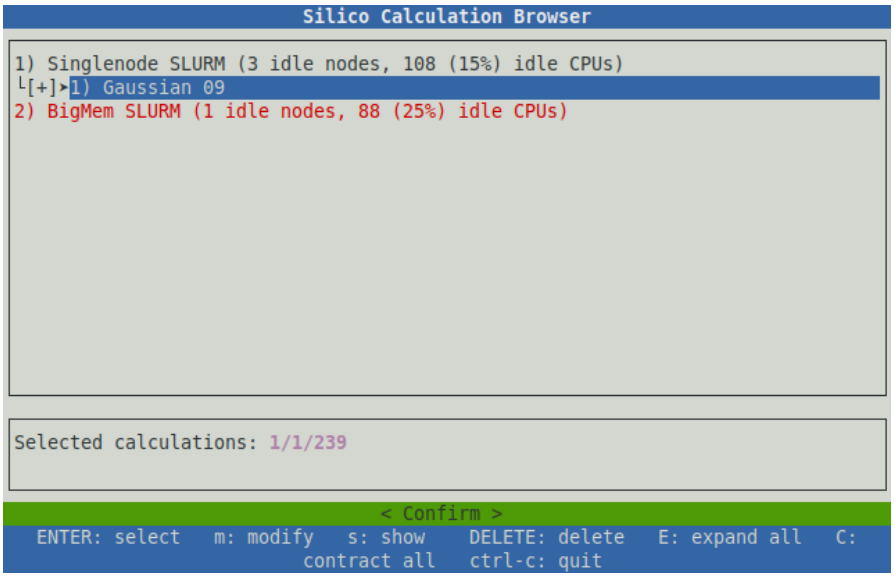
NOTE: Although BigMem SLURM does have more memory available than Singlenode SLURM, the latter already offers > 100 GB per node. This is likely to be more than sufficient for most cases.

NOTE: Debug SLURM and Multi Node SLURM are hidden by default, but they can be revealed by pressing the ‘s’ key. Please make sure you know what you are doing before using either.

Next to the name of each method, the status of the relevant partition is printed. This gives an indication of how busy each is at the current moment:



To select a method, navigate up and/or down using the arrow keys and use ‘enter’ to make your selection. For example, to select the first method, ‘Singlenode SLURM’:



Select a Program

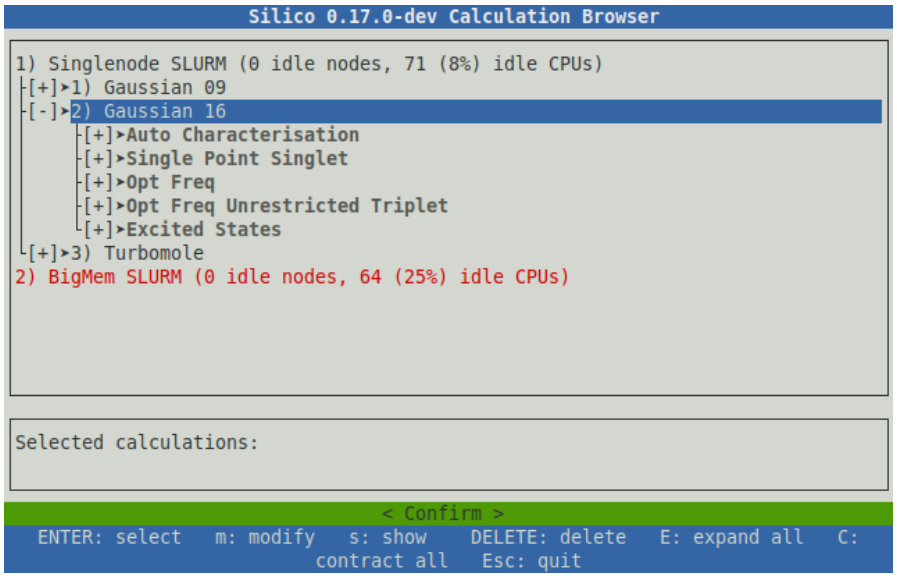
The next item to choose from the browser is the calculation program. Currently, three calculation programs are supported, they are: Gaussian 09, Gaussian 16 and Turbomole. In general, Gaussian 16 should be chosen for the characterisation of all organic and organometallic emitters and photocatalysts, as this later version offers a number of improvements over the older Gaussian 09. However, it is important to note that calculations with different programs (eg, Gaussian 09 and Gaussian 16) **are not comparable**, and so calculations on a series of molecules must be performed using the same program. Thus if some calculations on a series have already been

performed using G09, all further computations on that series should also use G09, or all calculations should be redone using G16.

Turbomole is primarily used for the characterisation of multi-resonant thermally activated delayed fluorescence (MR-TADF) emitters, as the photophysics of these compounds require higher order methods, such as SCS-CC2, which are only supported by Turbomole (see Submitting Calculations/ Turbomole SCS-CC2 and MR-TADF) . If your molecule is not a MR-TADF type emitter, you likely to do not need to use Turbomole.

NOTE: Special permissions are required to perform calculations with Turbomole. If you do not have access to Turbomole and believe you require these permissions, please contact Oli (osl@st-andrews.ac.uk).

If in doubt, Gaussian 16 is likely the correct program to use. To choose a program, navigate down with the arrow keys and use ‘enter’ to make your selection:



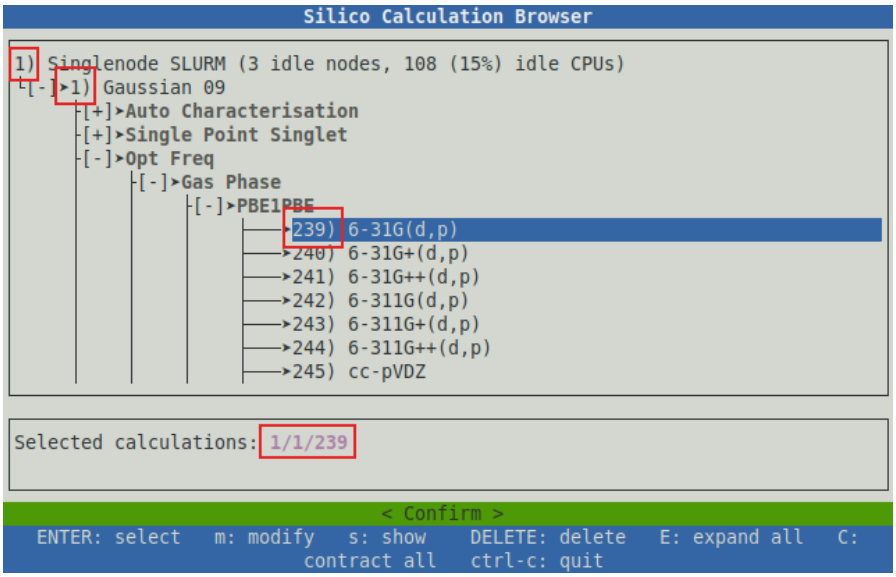
Select a Calculation

The final item to choose is the calculation itself. The available calculations are sorted into categories with self descriptive names, each category can be expanded in the same manner as before; with the arrow keys and enter key. The categories are as follows:

- Calculation type (Opt, Freq, SP, TD-DFT etc)
- Solvent (or gas phase)
- Functional
- Basis set

NOTE: There are 3 functionals available. PBE1PBE and B3LYP are popular and cheap (fast). M062X is more accurate but also more expensive (slow).

Once the desired calculation has been chosen, pressing enter will add it to the calculation list, represented by the box below the browser:



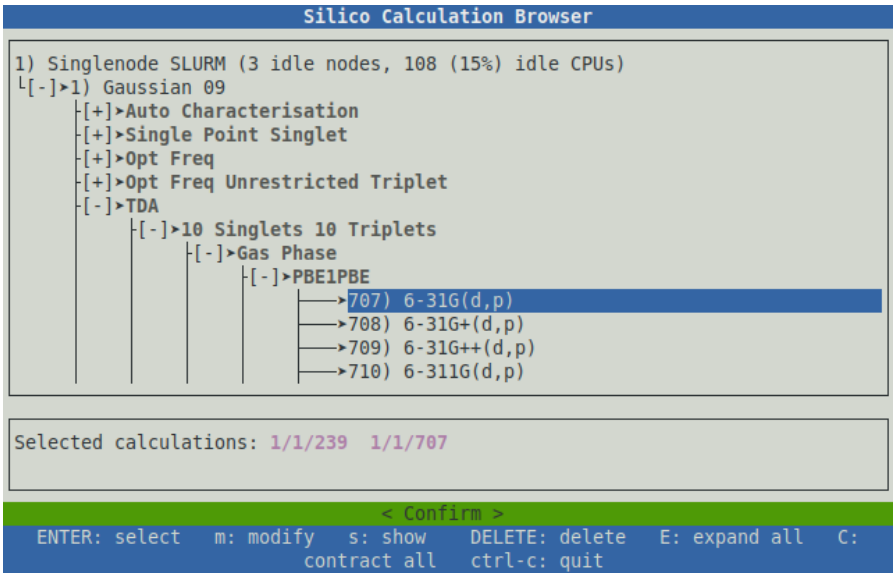
The calculation chosen is represented by 3 numbers corresponding to the selected method, program and calculation, which together make up a unique calculation code. In the above example, the choices were:

- Method 1 (Singlenode SLURM)
- Program 1 (Gaussian 09)
- Calculation 239 (Opt Freq PBE1PBE/6-31G(d,p) in the gas phase)

Resulting in the calculation code ‘1/1/239’.

NOTE: Pressing the ‘Delete’ key will delete the last calculation from the calculations list.

Multiple calculations can be chosen in the same way and will each be added to the calculations list:



Multiple calculations selected in this way will be run *in series*. That is, the first calculation (1/1/239) will be performed by itself. Once complete, the output of this calculation will be automatically submitted to the next in the list (1/1/707), and so on until all calculations requested have been performed.

NOTE: The calculations list is freely editable. Press the 'tab' key to first switch to the edit box before typing. Use 'shift'-'tab' to return to the browser.

NOTE: You can also queue up calculations to be performed with multiple programs, for example with Gaussian and Turbomole; the output from the previous calculation will automatically be converted to an appropriate input for the next program.

Confirm Selection

Once the desired calculations have been queued, press the 'tab' key to select the 'Confirm' button and then press 'enter' to confirm. If no error messages are printed, then the files have been submitted successfully.

If one of the files selected was a 2D format (ChemDraw for example), a warning similar to the following will be printed:

```
> silico Benzene.cdx
silico: WARNING: Generating 3D coordinates from file 'Benzene.cdx';
this will scramble atom coordinates
```

This warning indicates that the original 2D coordinates have been modified to produce 3D coordinates. In nearly all cases this is actually desirable and so this warning can be safely ignored.

Using Calculation Codes

The codes that identify each calculation (such as 1/1/239 and 1/1/707 used in the above example) are constant, unique and will always refer to the same calculation. This property allows computations to be submitted more quickly than by using the browser, if the calculation codes of interest are already known.

NOTE: Calculation codes are only constant for the same Silico version. Updates may add, remove or rearrange the available calculations, resulting in changes to the relevant codes. You will be notified if this occurs.

To submit a calculation without using the browser, use the '-c' (**c**alculation) option to the silico command, followed by the relevant codes:

```
> silico Benzene.com –c 1/1/239
```

Calculations can be queued by separating each with a space, so to recreate the example above, the full command would be:

```
> silico Benzene.com –c 1/1/239 1/1/707
```

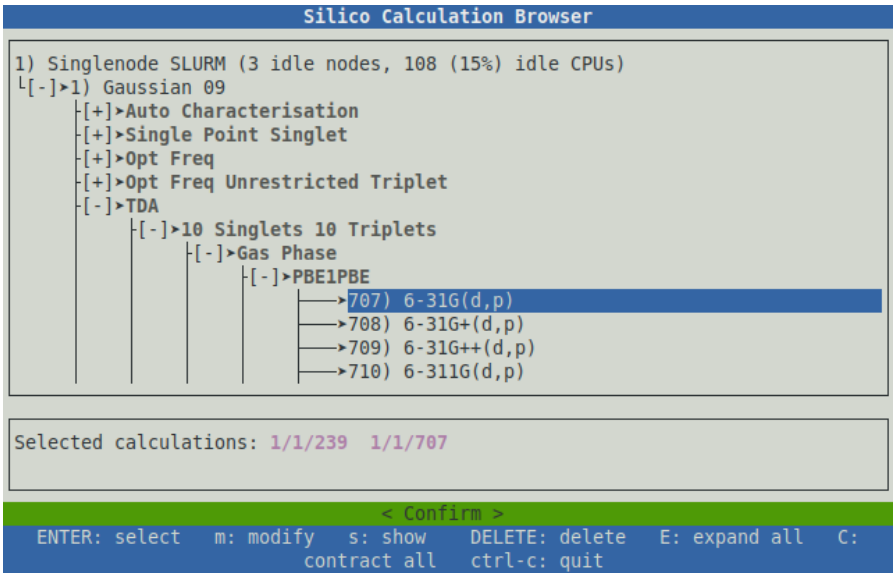
Common Codes

For reference, some of the more common calculation codes are printed here:

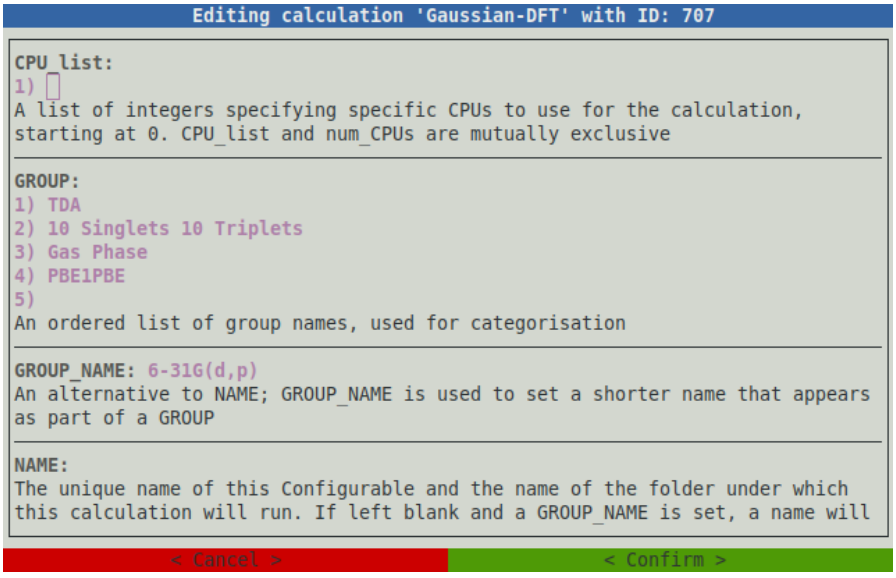
Code	Description	Comment
1/2/1	Performs the following at PBE1PBE/6-31G(d,p): • Ground optimisation (with frequency) • TDA excited states (50/50)	Used for characterising organic fluorescent and TADF emitters
1/2/4	Performs the following at B3LYP/6-31G(d,p), using SBKJC-VDZ-ECP for metals, in MeCN: • Ground optimisation (with frequency) • TD-DFT excited states (50/50) • Optimised triplet (with frequency) • Single point singlet at the same geometry	Used for characterising organometallic phosphorescent emitters
1/3/5	Performs the following at SCS-CC2/cc-pVDZ: • Ground optimisation • S_1 and S_2 excited states • T_1 and T_2 excited states	Used for characterising multiple-resonant TADF emitters. See: Turbomole SCS-CC2 and MR-TADF
1/2/275	Optimisation (with frequency) at PBE1PBE/6-31G(d,p)	EZC recommended for organic emitters
1/2/288	Optimisation (with frequency) at B3LYP/6-31G(d,p)	Popular method & basis set in the literature
1/2/338	Optimisation (with frequency) at B3LYP/6-31G(d,p), using SBKJC-VDZ-ECP for metals, in MeCN	EZC recommended for organometallic emitters
1/2/809	TDA excited states at PBE1PBE/6-31G(d,p)	EZC recommended for organic emitters
1/2/2132	TD-DFT excited states at B3LYP/6-31G(d,p), using SBKJC-VDZ-ECP for metals, in MeCN	EZC recommended for organometallic emitters
1/2/1277	TDA excited states (100 singlets) at PBE1PBE/6-31G(d,p)	Used for simulating absorption spectra of organic emitters
1/2/1340	TD-DFT excited states (100 singlets) at B3LYP/6-31G(d,p), using SBKJC-VDZ-ECP for metals, in MeCN	Useful for simulating absorption spectra of organometallic emitters

Advanced: Modifying Calculations

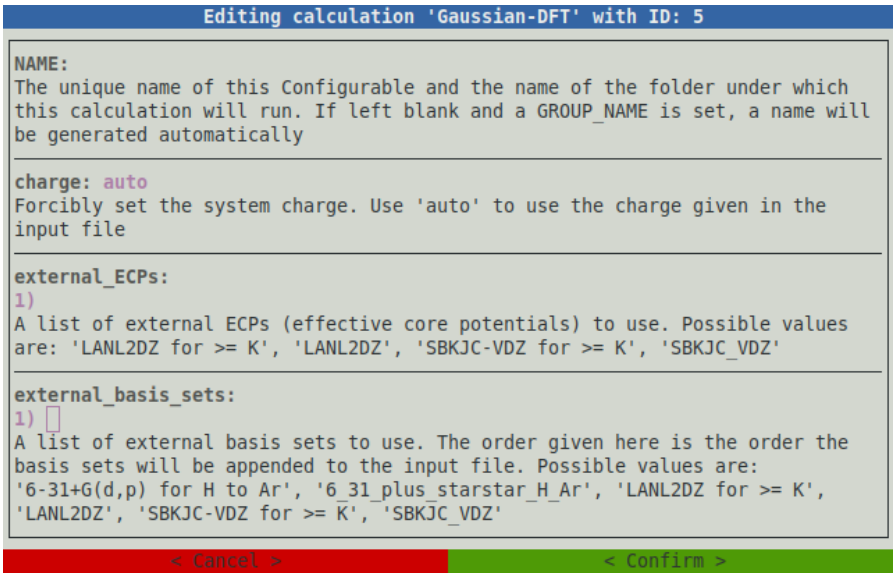
The calculations available in Silico can be modified prior to submission. This allows altering of such properties as: memory, CPUs, basis set, functional and more. To modify a calculation, follow the submission process until the calculation browser is displayed:



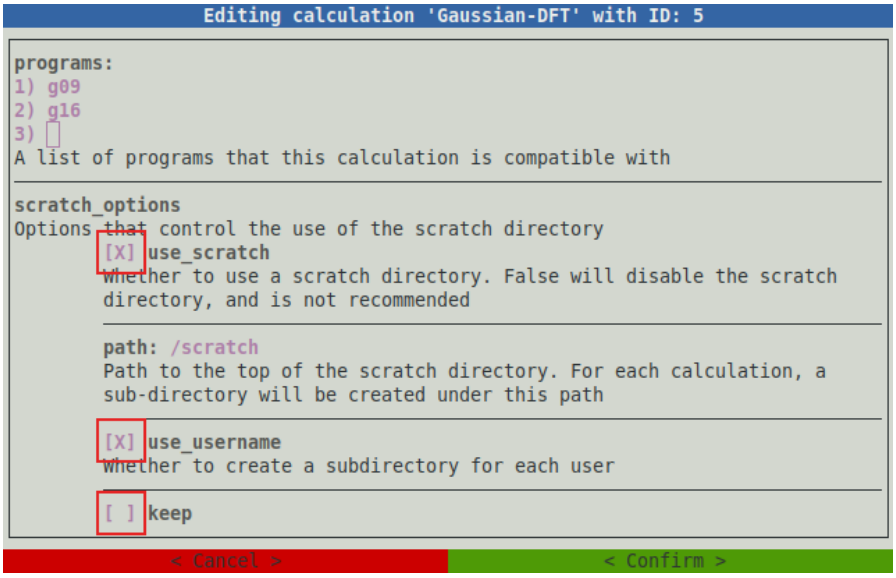
Once a calculation has been selected with the arrow keys, press ‘m’ (**m**odify) or ‘e’ (**e**dit) to open the edit window:



Here, numerous properties of the calculation can be changed. The up and down arrow keys are used to navigate between the different fields. Each comes with a help message, displayed below the field, explaining the property and how it impacts the calculation. There are a great many more fields than can be displayed at one time; navigating to the bottom of the screen will scroll down so more fields can be viewed:



Most fields are freely editable (can take any value). Some can only accept values from a list; in which case the list of possible values is given in the help. Some can only accept a ‘True’ or ‘False’ value, these fields appear as follows:



In such fields, the large ‘X’ indicates the option is set to ‘True’ or ‘Yes’, its absence indicates the opposite. ‘Space’ or ‘enter’ can be used to toggle the value.

NOTE: The ‘Cancel’ button will discard any changes and return to the calculation browser. The ‘Esc’ key also has the same effect.

Once the necessary changes have been made, press ‘tab’ to select the ‘Confirm’ button, then press ‘enter’ to confirm. The screen will then return to the calculation browser. You can now add the modified calculation to the calculation list as normal. It is important to note that any changes

made will effect all calculations with the same calculation code, regardless of whether the calculation was added to the list before or after changes were made.

NOTE: All modifications made in this way are temporary. The next time the calculation browser is opened, settings will revert to their default. If an edit is made in error, and the original value cannot be remembered, keep pressing 'Esc' until the browser is closed, then begin the submission process again. Future versions of Silico will offer to permanently save modifications.

Checking The Queue

The size of the server queue (the number of calculations waiting to be submitted) can be checked at any time using the 'silico status' command, which will display the following output:

```
> silico-dev status
  ID  Name              Status
----  ----------------  ----------------------------------
   1  Singlenode SLURM  0 idle nodes, 71 (8%) idle CPUs
   2  BigMem SLURM      0 idle nodes, 64 (25%) idle CPUs
   3  Debug SLURM       1 idle nodes, 32 (100%) idle CPUs
   4  Multi Node SLURM  1 idle nodes, 32 (1%) idle CPUs
   5  Local (series)    N/A (status not available)
   6  Local (parallel)  N/A (status not available)
```

This table lists one row for each available queue. Typically the top queue, Singlenode SLURM, is of most interest as this is where the majority of calculations are submitted to. The rightmost column describes the current size of the queue: the number of 'idle nodes' indicates the number of machines in the cluster that are currently 'empty' (doing nothing), while the number of 'idle CPUs' indicates the number of CPUs that are freely available. The smaller the number of idle CPUs, the busier the queue and the longer a calculation is likely to have to wait before starting.

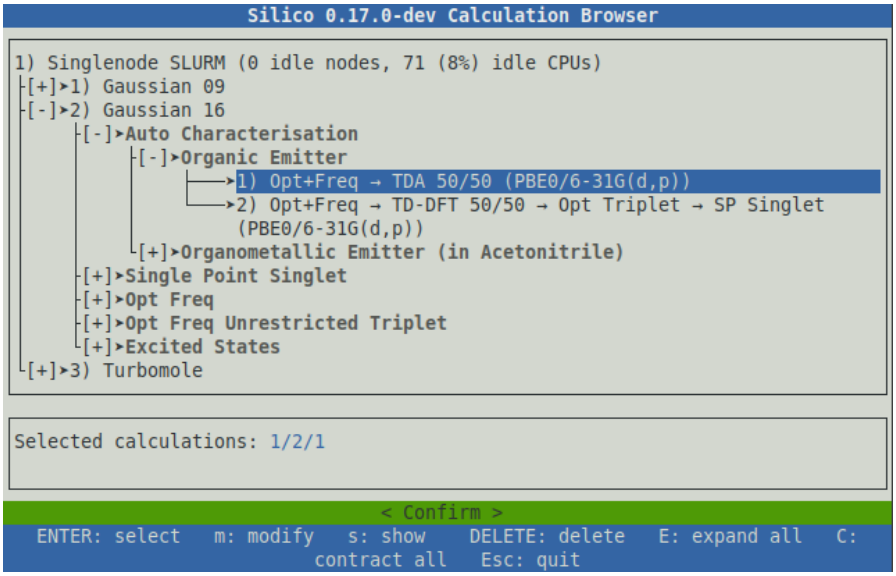
NOTE: Calculations can always be submitted regardless of the queue size; if insufficient CPUs are currently available then new calculations will be added to the back of the queue and will be started automatically once sufficient resources become available.

Turbomole SCS-CC2 and MR-TADF

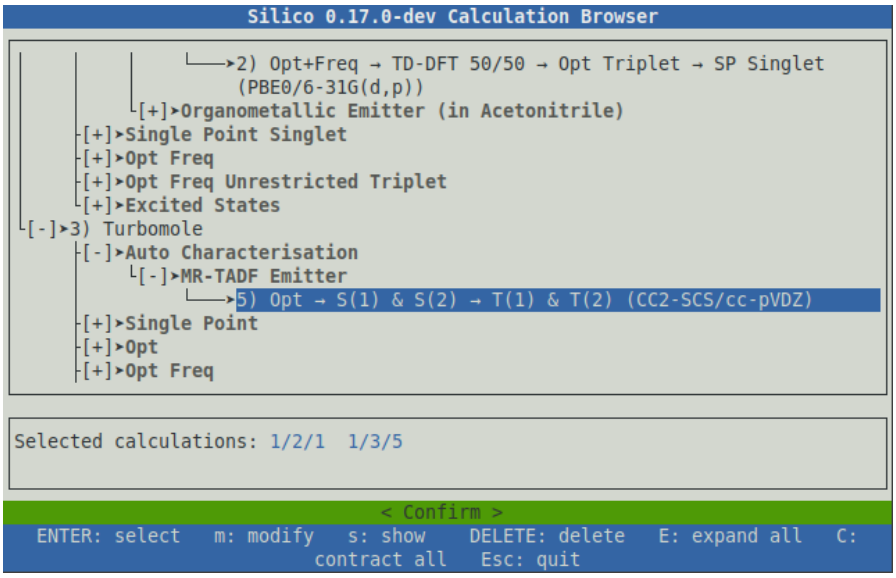
Multi-resonant TADF emitters cannot be accurately modelled using conventional DFT. Instead, these emitters require a class of ‘higher order’ methods called coupled cluster (CC) methods. Coupled cluster methods come in various flavours, some of which are implemented by both Gaussian and Turbomole, such as CCD, CCSD and CCSD(T), but the majority of these methods are extremely expensive (very slow). To overcome this, the approximate coupled-cluster singles and doubles model (CC2) uses a number of approximations to drastically increase the speed of computation, which makes it viable for (semi-) rapid characterisation. This CC2 method is only available in Turbomole, and so all MR-TADF type emitters must use Turbomole for the calculation of singlet and triplet energies.

Although faster than most CC methods, CC2 is still extremely slow compared to DFT. Therefore, it is recommended to first perform a DFT optimisation so the starting geometry can be brought closer to the final geometry, reducing the overall computation time. It is recommended that this DFT step be performed in Gaussian, so that calculated energies can be compared to other, non MR-TADF emitters.

To characterise an MR-TADF emitter, first queue up a full DFT auto characterisation with Gaussian 16 (code 1/2/1):



Next, select Turbomole from the program list, and queue up a full auto characterisation using SCS-CC2 (code 1/3/5):



You can now submit the calculations as normal. This will perform a total of five calculations in series:

1. **Gaussian (DFT):** Optimisation of ground state and calculation of IR frequencies.
2. **Gaussian (DFT):** Calculation of the first 10 singlet and triplet excited states.
3. **Turbomole (CC2):** Optimisation of ground state.
4. **Turbomole (CC2):** Calculation of the first two singlet excited states.
5. **Turbomole (CC2):** Calculation of the first two triplet excited states.

NOTE: Unlike Gaussian, Turbomole cannot calculate both singlet and triplet excited states in the same calculation, so these are instead performed in series.

NOTE: Even on relatively small structures, full characterisation with CC2 can take several days or weeks to complete.

Folder Structure

Silico will create a series of folders for each file and each calculation submitted. The topmost folder is named after the molecule submitted. Within this molecule directory, a separate folder is then created for each calculation program, within which is a separate folder each calculation, using the name of that calculation. If the same input file is submitted to multiple of the same calculation, a number will be appended to the calculation folder name to ensure each calculation is performed in a unique directory.

NOTE: The name of the 'molecule' directory is simply taken from the name of the input file; Silico does not try to automatically name structures.

Within each calculation directory, 5 sub folders are created. They are as follows:

Input

Contains the input file(s) for the calculation. The file(s) will be fully prepared for the calculation program, so they can be inspected to determine the specific parameters of the calculation. If an external basis set was requested (SBKJC-VDZ-ECP, for example) then this will be appended to the input file (if appropriate). If a non-native format was submitted (ChemDraw, for example), then this will have been converted appropriately.

NOTE: The file originally submitted is not stored in the Input directory; only the prepared input file is saved.

Output

Contains output files written by the calculation program.

NOTE: By default, Silico will automatically convert Gaussian .chk files to .fchk (before deleting the original .chk) in order to save file space. Occasionally, the original .chk file is required for post-analysis, in which case this option can be disabled (see **Submitting Calculations/Advanced: Modifying Calculations**).

Flags

Contains file flags, text files that convey information about the status of the calculation. See the section on **Monitoring Calculations/The Flags Folder** for more information about file flags.

Results

Contains text result files that are automatically written by Silico during post analysis.

Report

Contains a PDF report summary of the completed calculation, along with the rendered images used in the report. In addition, a smaller, mini report is also generated containing only atom coordinates, intended for easy inclusion into ESI.

Monitoring Calculations

The Flags Folder

Within each calculation directory, Silico manages a special folder with the name ‘Flags’. This folder contains a number of empty text files where the name of each file conveys status about the calculation. These files are created and destroyed at key points in the calculation submission process, so the calculation can be monitored by observing which files are present (or absent) from the ‘Flags’ folder at any given moment.

To check the file flags, use the ‘ls’ command, followed by the path to the relevant Flags folder. For example:

```
> ls "Benzene/Gaussian 16/Opt Freq Gas Phase PBE1PBE 6-31G(d,p)/Flags"
RUNNING  STARTED
```

Here, two flags are present: ‘RUNNING’ and ‘STARTED’, which indicate that the calculation is currently ongoing.

Typing the full path to the flags folder is time consuming, so it is recommended to make use of autocompletion by using the ‘tab’ key. Begin typing the path, for example:

```
> ls Benzene/Gaussian\ 16/Opt
```

Then press the ‘tab’ key. Linux will automatically guess the rest of the filename (based on the available files that match what has already been typed):

```
> ls Benzene/Gaussian\ 16/Opt
> ls Benzene/Gaussian\ 16/Opt\ Freq\ Gas\ Phase\ PBE1PBE\ 6-31G\
(d\,p\)/
```

Then type ‘Flags’ to complete the path:

```
> ls Benzene/Gaussian\ 16/Opt
> ls Benzene/Gaussian\ 16/Opt\ Freq\ Gas\ Phase\ PBE1PBE\ 6-31G\
(d\,p\)/
> ls Benzene/Gaussian\ 16/Opt\ Freq\ Gas\ Phase\ PBE1PBE\ 6-31G\
(d\,p\)/Flags
```

NOTE: Autocompletion is a feature of Linux, not Silico, meaning you can use it with any Linux command.

If the autocompletion function does not guess the filename, press ‘tab’ a second time to see a list of options:

```
> ls Benzene/Gaussian\ 16/Opt\ Freq\ Gas\ Phase\
Opt Freq Gas Phase B3LYP 6-31G(d,p)/   Opt Freq Gas Phase PBE1PBE 6-
31G(d,p)/
> ls Benzene/Gaussian\ 16/Opt\ Freq\ Gas\ Phase\
```

Choose from the options listed by typing:

```
> ls Benzene/Gaussian\ 16/Opt\ Freq\ Gas\ Phase\
Opt Freq Gas Phase B3LYP 6-31G(d,p)/   Opt Freq Gas Phase PBE1PBE 6-
31G(d,p)/
> ls Benzene/Gaussian\ 16/Opt\ Freq\ Gas\ Phase\ B
```

Then use ‘tab’ again to autocomplete the rest of the path:

```
> ls Benzene/Gaussian\ 16/Opt\ Freq\ Gas\ Phase\
Opt Freq Gas Phase B3LYP 6-31G(d,p)/   Opt Freq Gas Phase PBE1PBE 6-
31G(d,p)/
> ls Benzene/Gaussian\ 16/Opt\ Freq\ Gas\ Phase\ B3LYP\ 6-31G\(d\,p\)/
```

List of Flags

The currently available file flags are as follows:

Name	Description
PENDING	The calculation has been submitted but has not yet begun; most likely because it is waiting in the queue.
STARTED	The calculation has begun. This flag is never deleted, so it is useful for confirming that the calculation at least started, even if it did not finish.
RUNNING	The calculation is currently ongoing.
SUCCESS	The calculation finished successfully.
CONVERGED	The optimisation converged successfully. This flag is only used for optimisation calculations.
NOT_CONVERGED	The optimisation did not converge successfully. This flag is only used for optimisation calculations.
CLEANUP	The calculation has finished (successfully or otherwise) and Silico is currently cleaning up (saving files etc).
ERROR	The calculation has stopped because an error occurred.
POST	The calculation has finished and Silico is currently performing post analysis (writing result and report files).
DONE	All work (including post-analysis) has been completed; Silico will not make any changes after this flag. It is safe to move, download or delete the calculation folder.

Multiple Calculations

Multiple calculations can be checked simultaneously by specifying multiple directories to the 'ls' command:

```
> ls "Benzene/Gaussian 16/Opt Freq Gas Phase PBE1PBE 6-31G(d,p)/Flags"
"Benzene/Gaussian 16/TDA 10 Singlets 10 Triplets Gas Phase PBE1PBE 6-
31G(d,p)/Flags"
Benzene/Gaussian 16/Opt Freq Gas Phase PBE1PBE 6-31G(d,p)/Flags:
CONVERGED  DONE  STARTED  SUCCESS

Benzene/Gaussian 16/TDA 10 Singlets 10 Triplets Gas Phase PBE1PBE 6-
31G(d,p)/Flags:
DONE  STARTED  SUCCESS
```

This is more conveniently achieved using the Linux wildcard feature, using the asterisk character (*). For example, to check all calculations in the Benzene directory:

```
> ls Benzene/*/*/Flags
Benzene/Gaussian 16/Opt Freq Gas Phase B3LYP 6-31G(d,p)/Flags:
CONVERGED  DONE  STARTED  SUCCESS

Benzene/Gaussian 16/Opt Freq Gas Phase PBE1PBE 6-31G(d,p)/Flags:
CONVERGED  DONE  STARTED  SUCCESS

Benzene/Gaussian 16/Single Point Singlet Gas Phase PBE1PBE
6-31G(d,p)/Flags:
DONE  STARTED  SUCCESS

Benzene/Gaussian 16/TDA 10 Singlets 10 Triplets Gas Phase PBE1PBE 6-
31G(d,p)/Flags:
DONE  STARTED  SUCCESS
```

Similarly, to check the status of all calculations of all molecules in the current directory, three wildcards can be used:

```
> ls */*/*/Flags
246tCzPPC/Gaussian 16/TDA Optimised S(1) B3LYP 6-31G(d,p)/Flags:
PENDING

26tCzPPC/Gaussian 16/TDA Optimised S(1) B3LYP 6-31G(d,p)/Flags:
RUNNING  STARTED

2CzIPN/Gaussian 16/TDA Optimised S(1) B3LYP 6-31G(d,p)/Flags:
CONVERGED  DONE  STARTED  SUCCESS

2CzPN/Gaussian 16/TDA Optimised S(1) B3LYP 6-31G(d,p)/Flags:
CONVERGED  DONE  STARTED  SUCCESS

2CzTPN/Gaussian 16/TDA Optimised S(1) B3LYP 6-31G(d,p)/Flags:
CONVERGED  DONE  STARTED  SUCCESS
...
```

Post Analysis

Results

Once a calculation is complete (the DONE flag has been set), result files will be available for analysis in the 'Results' folder. Perhaps the most important of these is the 'summary' file, which will be named something like 'Benzene.summary'. This file can be viewed on the server using the 'less' or 'cat' commands:

```
> less "Benzene/Gaussian 16/Opt Freq Gas Phase B3LYP 6-31G(d,p)/Results
/Benzene.summary"
```

NOTE: When using the 'less' command, use the 'q' key to quit.

The summary file, as its name would suggest, contains a summary of the most important calculation results. An example file is display below:

```
> cat "Benzene/Gaussian 16/Opt Freq Gas Phase B3LYP 6-31G(d,p)/Results/
Benzene.summary"
--------------------
|     Metadata     |
--------------------
Name: Benzene/Opt Freq Gas Phase B3LYP 6-31G(d,p)/Output/Benzene.log
Date: 30/07/2020 at 10:27:17
Duration: 0 days, 0 hours, 1 minutes
Computational package: Gaussian (2009+D.01)
Calculations: Optimisation, Frequencies
Methods: DFT
Functional: B3LYP
Basis set: 6-31G(d,p)
Multiplicity: Singlet
Charge: 0
Orbital spin: restricted
Success: True
Converged: True
Calculation temperature /K: 298.15
Calculation pressure /atm: 1.00

---------------------------------
|     Vibrational Frequencies     |
---------------------------------
No. vibrations: 30
No. negative frequency: 0

--------------------
|     Geometry     |
--------------------
Formula: C6H6
Exact mass /gmol-1: 78.0469
Molar mass /gmol-1: 78.1118
No. atoms: 12
Alignment method: Minimal
```

```
X extension /Å: 5.00
Y extension /Å: 3.58
Z extension /Å: 2.85
Linearity ratio: 0.28
Planarity ratio: 0.20

----------------------
|     SCF Energy     |
----------------------
No. of steps: 31
Energy /eV: -6315.9237
Energy /kJmol-1: -609393.9974

--------------------
|     Orbitals     |
--------------------
HOMO eV: -5.73
LUMO eV: -1.88
HOMO/LUMO energy /eV: 3.85
No. virtual: 99
No. occupied: 21

---------------------------------
|     Permanent Dipole Moment     |
---------------------------------
Total /D: 1.23
Origin X /D: 0.00
Origin Y /D: 0.00
Origin Z /D: 0.00
Vector X /D: -0.25
Vector Y /D: 1.12
Vector Z /D: -0.43
X axis angle /°: 78.40
XY plane angle /°: 20.61
```

The remaining files in the ‘Results’ directory contain specific calculation results in CSV format. These too can be read in the same way as the summary file, but are more easily manipulated once downloaded to a personal computer.

NOTE: CSV format can be easily imported into a spreadsheet (such as Microsoft Excel, Libre Office or Google Sheets) and graphed, if desired.

Available Result Files

The currently available result files are listed below (*FILE* is used as a stand-in for the actual filename):

Name	Description
FILE.summary	Human readable summary of all calculation results
FILE.atoms.csv	Optimised (if applicable) atom coordinates
FILE.orbitals.csv	Orbital numbers, labels and energies (restricted calculations only)
FILE.alpha.csv	Alpha orbital numbers, labels and energies (unrestricted calculations only)

FILE.beta.csv	Beta orbital numbers, labels and energies (unrestricted calculations only)
FILE.SCF.csv	SCF (self consistent field) energy. For optimisations, includes the energy at each step which can be used to graph the calculation's convergence. The SCF energy is the HF (Hartree–Fock) or DFT energy.
FILE.MP.csv	Same as *FILE*.SCF.csv, but for Møller–Plesset energies. The energies are total (including HF and MP correction) and are from the highest MP level of the calculation.
FILE.CC.csv	Same as *FILE*.SCF.csv, but for coupled-cluster energies. The energies are total (including HF and CC correction, if applicable).
FILE.ES.csv	Excited state results (from TD-DFT, TDA etc). Includes energy, symmetry, multiplicity and orbital contributions.
FILE.transitions.csv	Excited state transitions. This information is also included in *FILE*.ES.csv, but the format here may be more convenient for directly comparing transitions.
FILE.TDM.csv	Excited state transition dipole moments.
FILE.UV-Vis.csv	A simulated UV-Vis spectrum in corrected units of wavelength (nm).
FILE.absorptions.csv	A simulated UV-Vis spectrum in uncorrected units of energy (eV).
FILE.vibrations.csv	Vibrational frequencies.
FILE.IR.csv	A simulated IR spectrum.
FILE.SOC.csv	Singlet/triplet spin-orbit coupling values.

Report

Silico will also generate a PDF report file at the end of each calculation, this is available under the 'Report' folder. The report contains much the same data as is available in the result files, but in a more convenient and easily transferable format. In addition, the PDF report contains graphical results that can not be represented in text, including simulated spectra, convergence graphs, energy diagrams and orbital density images.

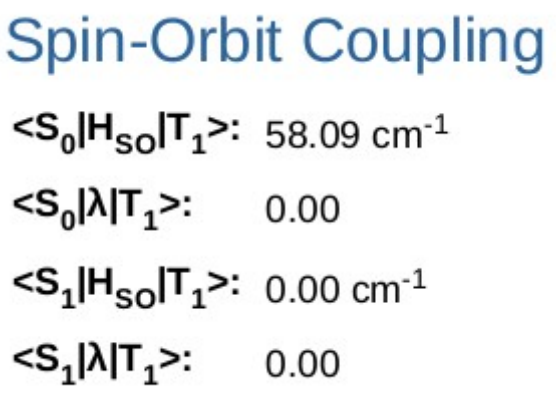
In addition to the report itself, all the rendered and generated image files are available under the 'Report/image' directory. For each 3D rendered structure image two images are available, a lower quality but smaller sized .jpg, and a higher quality, larger .png file.

NOTE: The PDF report itself uses the smaller .jpg files to ensure the file size is small enough to be sent by email. The higher quality .png files are included for publication purposes.

Finally, a smaller mini report is also available, named something like *FILE*.atoms.pdf, which only contains final atom coordinates. The intention of this mini-report is for it to be easily included in the ESI of journal papers.

Spin-Orbit Coupling (SOC)

Silico can compute spin-orbit coupling values using a modified version of the PySOC[1] program. SOC will be computed automatically for time-dependant DFT calculations (with or without the Tamm Dancoff approximation) performed using Gaussian (other program are not yet supported). SOC will be calculated between all singlet states, including the ground, and all triplet states. In the pdf report, a summary of the SOC between S_0/T_1 and S_1/T_1 is available on the summary page:



Here, H_{SO} is the SOC between the two given states, while λ is the first order mixing coefficient, given by:

$$\lambda = \frac{H_{SO}}{\Delta E_{ST}}$$

SOC values for all excited states are also available on the 'Table of Spin-Orbit Coupling' page later in the report:

Table of Spin-Orbit Coupling

Singlet	Triplet	SOC +1 /cm^{-1}	SOC 0 /cm^{-1}	SOC -1 /cm^{-1}	SOC Root Sum Square /cm^{-1}	H_{SO} /eV	ΔE /eV	First Order Mixing Coefficient
S_0	T_1	41.0762	0.0010	41.0762	58.0905	0.0072	3.3687	0.0021
S_0	T_2	0.0000	0.0002	0.0000	0.0002	0.0000	5.9293	0.0000
S_0	T_3	40.3854	0.0012	40.3854	57.1136	0.0071	8.0721	0.0009
S_0	T_4	0.0002	11.3528	0.0002	11.3528	0.0014	8.3342	0.0002
S_0	T_5	9.8311	0.0002	9.8311	13.9033	0.0017	9.7027	0.0002
S_0	T_6	0.0000	0.0004	0.0000	0.0004	0.0000	10.2885	0.0000
S_0	T_7	0.0004	21.0802	0.0004	21.0802	0.0026	11.1793	0.0002
S_0	T_8	3.4940	0.0001	3.4940	4.9412	0.0006	12.2589	0.0000

Here, 'SOC +1', 'SOC 0' and 'SOC -1' are the spin-orbit coupling values for quantum numbers +1, 0 and -1 respectively.

Analysis of Results

In addition to aiding calculation submission, Silico also includes a number of tools for analysing calculation result files. Analysis is performed using the 'silico result' sub-program.

Generating Summaries

Used without other arguments, result will generate a text summary of the given calculation result files. The content of this summary is the same as found in the *FILE*.summary file (see **Post Analysis/Results** section), and is useful for inspecting calculation files that were not submitted using silico.

To analyse a file named 'Benzene.log', the result command would be used as follows:

```
> silico result Benzene.log
--------------------
|     Metadata     |
--------------------
Name: Benzene/TDA 10 Singlets 10 Triplets Gas Phase PBE1PBE 6-31G(d,p)/
Output/Benzene.log
Computational package: Gaussian (2009+D.01)
Calculations: Excited States"
Methods: DFT
Functional: PBE1PBE
...
```

To save the summary to a file, Linux file redirection can be used with the '>' character. For example, to save the summary to a file named 'Benzene.summary':

```
> silico result Benzene.log > Benzene.summary
```

NOTE: The standard redirection character (>) will overwrite the file if it already exists, deleting any existing data. To avoid this behaviour, use the double redirection character instead (>>).

Analysis of Multiple Results

Although extracting and formatting results from a single calculation output file may be useful, the real strength of the result tool is being able to analyse and tabulate multiple calculation result files simultaneously.

This is achieved by specifying multiple files to 'silico result' with the '-a' (table) option. It is recommended that this command be paired with the 'less' command, which allows the considerable quantity of data to be scrolled, as follows:

```
> silico result Benzene.log "Methyl benzene.log" -a | less -S
```

NOTE: Silico loads calculation result files in parallel, meaning there is little difference in time between reading one file or two (up to the number of cores in the computer).

The table format is useful for 'quickly' comparing the results from a handful of calculations, but may become cumbersome when comparing/analysing hundreds of results. For extended post analysis, the '-c' (**C**SV) option is recommended, which produces files that can be easily imported into a spreadsheet of choice.

To write tabulated CSV result files, use result with the '-c' option and redirect the output using the standard Linux redirection character '>'. For example, to write to a file named 'Results.csv':

```
> silico result Benzene.log "Methyl benzene.log" -c > Results.csv
```

When analysing a great many result files, typing each file name individually is cumbersome. Instead, using the Linux wildcard character (*) is recommended. For example, to tabulate the results of all calculations in the current directory, the following command would be used:

```
> silico result */*/*/Output/*.log -c > Results.csv
```

The program may take some time to complete, depending on the number of calculations being analysed. The program will print the name of each file being read to keep track of progress:

```
> silico result */*/*/Output/*.log -c > Results.csv
silico: INFO: Parsing calculation result file 'Benzene/Gaussian 16/Opt
Freq Gas Phase B3LYP 6-31G(d,p)/Output/Benzene.log'
silico: INFO: Parsing calculation result file 'Benzene/Gaussian 16/Opt
Freq Gas Phase PBE1PBE 6-31G(d,p)/Output/Benzene.log'
silico: INFO: Parsing calculation result file 'Benzene/Gaussian
16/Single Point Singlet Gas Phase PBE1PBE 6-31G(d,p)/Output
/Benzene.log'
silico: INFO: Parsing calculation result file 'Benzene/Gaussian 16/TDA
10 Singlets 10 Triplets Gas Phase PBE1PBE 6-31G(d,p)/Output
/Benzene.log'
silico: INFO: Parsing calculation result file 'Benzene/Gaussian 16/TD-
DFT 10 Singlets 10 Triplets Gas Phase PBE1PBE 6-31G(d,p)/Output
/Benzene.log'
```

Once complete, the resulting file can be downloaded from the server and analysed at leisure.

Generating Reports

Normal Usage

The PDF reports automatically generated by Silico at the end of a calculation can also be generated manually by using the 'silico report' command. Reports require both the main output file (.log) and also the checkpoint file (either .chk or .fchk), as image data is only stored in the latter. However, the report command will automatically search for checkpoint files in the same folder as the .log file, so long as they have the same name, so often only the .log file needs to be specified. Silico report is used as follows:

```
> silico report Benzene.log
```

The program will search in the same folder as 'Benzene.log' for a file named either 'Benzene.fchk' or 'Benzene.chk'. The checkpoint file can also be given explicitly using:

```
> silico report Benzene.log C6H6.fchk
```

The report will be written, by default, to a folder name 'report' **in the same folder as the input (.log) file**. This can be changed using the -o (**o**utput) option:

```
> silico report Benzene.log C6H6.fchk –o "Benzene Report"
```

Generating the report can take some time, especially for calculations involving excited states (TDA and TD-DFT). The report program will print status messages periodically to keep track of progress:

```
> silico report Benzene.log
silico: INFO: Found 'Benzene.fchk' in input directory; using as
gaussian/formatted-checkpoint file
```

This message indicates that silico found the checkpoint file correctly. Other output will appear similar to the following:

```
> silico report Benzene.log
silico: INFO: Found 'Benzene.fchk' in input directory; using as
gaussian/formatted-checkpoint file
silico: INFO: Reading result file 'Benzene.log'
silico: INFO: Generating report 'report/Benzene.pdf'
silico: INFO: Converting gaussian/formatted-checkpoint file
'Benzene.fchk' to gaussian/cube file
'report/image/Structure/Benzene.structure.cube'
silico: INFO: Rendering report/image/Structure/Benzene.structure.jpg to
file(s)
silico: INFO: Rendering report/image/Benzene.SCF_graph.png to file(s)
silico: INFO: Rendering report/image/Dipole Moment/Benzene.dipole.jpg
to file(s)
```

```
silico: INFO: Rendering report/image/Orbital
Diagram/Benzene.HOMO_LUMO.png to file(s)
silico: INFO: Converting gaussian/formatted-checkpoint file
'Benzene.fchk' to gaussian/cube file
'report/image/HOMO/Benzene.HOMO.cube'
silico: INFO: Rendering report/image/HOMO/Benzene.HOMO.jpg to file(s)
silico: INFO: Converting gaussian/formatted-checkpoint file
'Benzene.fchk' to gaussian/cube file
'report/image/LUMO/Benzene.LUMO.cube'
silico: INFO: Rendering report/image/LUMO/Benzene.LUMO.jpg to file(s)
silico: INFO: Rendering report/image/HOMO LUMO/Benzene.HOMO_LUMO.jpg to
file(s)
silico: INFO: Rendering report/image/Orbital
Diagram/Benzene.orbitals.png to file(s)
silico: INFO: Rendering report/image/Benzene.simulated_frequencies.png
to file(s)
silico: INFO: Done generating report 'report/Benzene.pdf'
```

Including Emission Energy

Reports do not, by default, include results on emission energy. This is because these results require multiple calculation result files, and Silico cannot know where to find these automatically. To include emission data, use the '--emission' option with the path to the emission result file. For example, using Benzene with calculation code 1/2/4 (the standard for organometallic phosphorescent emitters) the full command would be:

```
> silico report "Opt Freq Acetonitrile (SCRF-PCM) B3LYP 6-
31+G(d,p)_SBKJC-VDZ (ECP)/Output/Benzene.log" --emission "Opt Freq
Unrestricted Triplet Acetonitrile (SCRF-PCM) uB3LYP 6-31+G(d,p)_SBKJC-
VDZ (ECP)/Output/Benzene.log" -o "Benzene emission"
```

NOTE: EZC SOP is to calculate emission data for phosphorescent emitters only (because these calculations are more accessible). The emission calculation is the 'Opt Freq Unrestricted Triplet' calculation. The difference in energy between this calculation and the normal, ground state optimisation gives the **adiabatic** (relaxed) emission energy. SOP is also to calculate the **vertical** (unrelaxed) emission energy, which is the difference between the ground state optimised and a single point calculation at the triplet geometry. These calculations are all performed by code 1/1/4.

This will calculate the **adiabatic** emission energy. To also include the **vertical** emission, use the '--vertical_ground' option:

```
> silico report "Opt Freq Acetonitrile (SCRF-PCM) B3LYP 6-
31+G(d,p)_SBKJC-VDZ (ECP)/Output/Benzene.log" --emission "Opt Freq
Unrestricted Triplet Acetonitrile (SCRF-PCM) uB3LYP 6-31+G(d,p)_SBKJC-
VDZ (ECP)/Output/Benzene.log" --vertical_ground "Single Point Singlet
Acetonitrile (SCRF-PCM) B3LYP 6-31+G(d,p)_SBKJC-VDZ
(ECP)/Output/Benzene.log" -o "Benzene emission"
```

In this case, the report would be rendered to the "Benzene emission" folder.

Bibliography

1 X. Gao, S. Bai, D. Fazzi, T. Niehaus, M. Barbatti and W. Thiel, *J. Chem. Theory Comput.*, 2017, **13**, 515–524.