

Supplementary Material

Installation and getting started

THERM23 manual:

THERM23: A Thermodynamic Property Estimator for Gas Phase Species using the Group Additivity Method

Sergio Martinez, Manik Kumer Ghosh*, and Henry J. Curran*

Combustion Chemistry Centre, School of Biological and Chemical Sciences, Ryan Institute, MaREI, University of Galway, Galway, Ireland

manikghosh@gmail.com, henry.curran@universityofgalway.ie

Table of Contents

1	Setting up and Installation of Python	5
1.1	Python installation in Windows	5
1.1.1	Testing Python and installing modules	6
1.2	Python installation in Linux (distro: Fedora 30)	11
1.2.1	Installing Python in Fedora 30 and the corresponding modules.....	13
2	Operation Procedure of THERM23	16
2.1	THERM23 Code and Files.....	16
2.2	THERM23 options.....	18
2.2.1	Interactive.....	19
2.2.2	Automatic.....	26
2.2.3	Re-calculate.....	30
2.2.4	Output files content and format.....	35
2.2.5	Thermo plotter.....	38
2.2.6	Thermo fitter	46

Supplementary Material

List of Figures

Figure 1 THERM23 main folder (left), PythonSoftware subdirectory (middle), and Python executable for windows (right)	5
Figure 2 Python's wizard installation welcome screen.....	6
Figure 3 Python's wizard installation welcome screen with the addition of the Add Python to PATH option ticked.....	6
Figure 4 How to open a PowerShell terminal from the searching bar option. Searching bar indicator (left), and the PowerShell keyword option displayed (right).....	7
Figure 5 Python's version installed screen.....	7
Figure 6 Python's interactive mode and exit command from this mode.	8
Figure 7 Checking if module numpy is already installed in Python's software.	8
Figure 8 Example of the error thrown when a module is not installed in Python's software.....	9
Figure 9 Installation of numpy module in Python using pip manager command line.	10
Figure 10 Installation of multiple modules in Python using pip manager command line.	10
Figure 11 Installation of multiple modules in Python.....	11
Figure 12 How to open Linux-Fedora 30 terminal. Click on Activities, then click on the Terminal icon, and the terminal windows should show up.	12
Figure 13 How to open Linux-Fedora 30 terminal. Click on Type to search bar, then type the word terminal and press enter, terminal icon should show up. Finally, do left click on it with the mouse to open it.	12
Figure 14 Linux-Fedora 30 terminal screen showing the Python version installed (in green), and the error because there is not a module called numpy installed yet.....	12
Figure 15 Linux-Fedora 30 terminal screen showing that python3.7 is already installed.	13
Figure 16 Linux-Fedora 30 terminal screen showing python2 installation display.....	14
Figure 17 Linux-Fedora 30 terminal screen showing the common error throw by Python when a module that is not installed is imported, in this case numpy.	14
Figure 18 Linux-Fedora 30 terminal screen showing the pip manager installing the module numpy..	15
Figure 19 Linux-Fedora 30 terminal screen showing the pip manager installing multiple modules simultaneously for Python.	15
Figure 20 Linux-Fedora 30 terminal screen showing all modules have been imported properly and ready to work.	16
Figure 21 THERM23 main's folder and its content.....	16
Figure 22 GroupsDir folder and its content.	17
Figure 23 HC.grp file format content.....	18
Figure 24 InChIs.dat file format content.....	18
Figure 25 THERM23's welcome screen.	19

Supplementary Material

Figure 26 THERM23's welcome screen divided in sections. These sections are highlighted by a colour code in order to understand them better.....	19
Figure 27 Running THERM23 in 'Interactive' mode. In the red rectangle '1' was typed to choose the 'Interactive' mode, and in the green one, 'Example1' was typed to provide an output files name.	20
Figure 28 Files generated by Interactive mode when user provides as file name Example1. A whole collection of 5 different items is generated automatically when user provides the file name, however it is important to notice these files are still empty and open. User must wait until program is closed or stopped to open or visualise these files, in special the doc one.....	20
Figure 29 Left side: NUIGMech1.2.MECH showing that the specie's name for ethane is C ₂ H ₆ . On the right side: user entries for specie's name, chemical formula, number of C, and H atoms in chemical formula, and if this molecule is linear or not.....	21
Figure 30 Example of what happen if the user answered 'no' to the 'Is the data provided above, correct?'. User must answer again all the previous questions.	22
Figure 31 Example of providing a '2' as total GAVs to be used in pink rectangle. The GAV for carbon–carbon–3 × hydrogen bond is typed and ask the code to use it twice '2' pictured in the green rectangle of the picture.....	23
Figure 32 How to correct the GAV or GAVs if user mistyped them.....	24
Figure 33 Symmetry number and number of rotors in the green rectangle. Resume of the parameters provided and the proper calculations generated by THERM23.....	24
Figure 34 The top of the picture shows how the code will save the data calculated in the different files with extensions .dat, .doc, .LST, .rerun and .tmp, and is shown in the blue rectangle. Inside the yellow rectangle can be seen the parameters provided by the user for the calculation of the radical C ₂ H ₅	25
Figure 35 Skipping the calculation of a radical specie from its parent specie.	25
Figure 36 Calculation of radical specie based on parent molecule information.	26
Figure 37 Code display of the properties calculated for the radical specie.....	26
Figure 38 THERM23's automatic mode activation and output file name 'Example2' provided by user.	27
Figure 39 Output files generated by code in automatic mode; Example2.dat, Example2.doc, Example2.LST, Example2.rerun, and Example2.tmp.....	27
Figure 40 Input files (.INP) available in the main directory of THERM23.....	28
Figure 41 Content of THERM23.inp, which includes 14 different columns required for the calculation of thermodynamic properties.	28
Figure 42 Number of input files to run and its name.	29
Figure 43 Resume of the parameters provided for ethane (C ₂ H ₆) and the proper calculations generated by THERM23.....	30

Supplementary Material

Figure 44 Resume of the parameters provided for propane (C_3H_8) and the proper calculations generated by THERM23.....	30
--	----

Supplementary Material

1 Setting up and Installation of Python

THERM23 can be run in either a WINDOWS or LINUX environment. It is necessary for a user to have a suitable version of Python installed on their machine. An ideal version (any version equal to or superior to 3.7) containing the following modules: NumPy, Pandas, Matplotlib, SciPy, SymPy and Cython needs to be installed. If you already have a Python installation on your computer please move to Chapter 2, but if you do not know how to check the modules, please go to Section 1.2. Otherwise the instructions below are a step-by-step guide to the proper installation of, and correct functionality of the THERM23 software.

1.1 Python installation in Windows

To properly install Python's software, a user needs an executable copy of any Python version (equal to or superior to 3.7) available. The user can visit the official Python website [1] to download a suitable version. Inside the THERM23 folder, as shown in Fig. 1, the user will find the PythonSoftware folder, which contains the latest version of Python (June 2023), namely python-3.10.5-amd64.exe. However, the type of executable file should be chosen depending on the version of Windows installed on the computer and its architecture (32/64 bits). For example, in the case of Windows 10, python-3.10.5-amd64 is perfect, but if the Windows installation is older than Windows 8, it is advisable to use python-3.8.7-amd64 instead, which can be found on the official Python website [1].

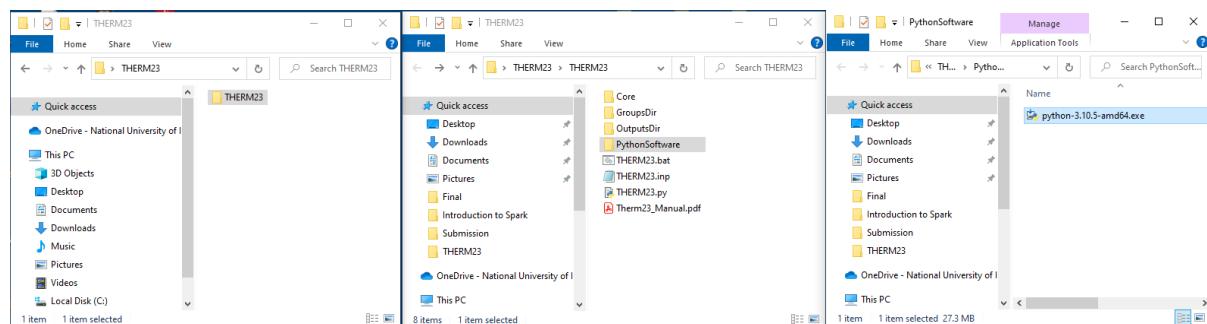


Figure 1. THERM23 main folder (left), PythonSoftware subdirectory (middle), and Python executable for windows (right).

Once a user has decided which version of Python they would like to install, then by doing double-click on the executable file on the left bottom of the mouse or select the file and press enter on the keyboard will be enough to start the installation wizard. Then, the installation wizard will open and present to the user the next screen, as shown in Fig. 2. For this example, we have used a Python 3.8 executable file. However, the procedure is the same for all software versions.

Supplementary Material



Figure 2. Python's wizard installation welcome screen.

IMPORTANT NOTE: Before pressing 'Install Now', the user must select the 'Add Python 3.8 to PATH' option. Please see the following snapshot for reference in Fig. 3. After this, the user can press 'Install now' and continue the instructions on the screen.



Figure 3. Python's wizard installation welcome screen with the addition of the Add Python to PATH option ticked.

1.1.1 Testing Python and installing modules

Once Python is installed on the user's computer, the next step is to test the installation to verify everything works perfectly. Then we will move to install the corresponding modules/dependencies, which are essential for a proper Python software performance. In

Supplementary Material

order to open a Windows PowerShell terminal the user can type in the search bar the word PowerShell, as shown in Fig. 4 (left). A window should appear with options related to this word as shown in Fig. 4 (right).

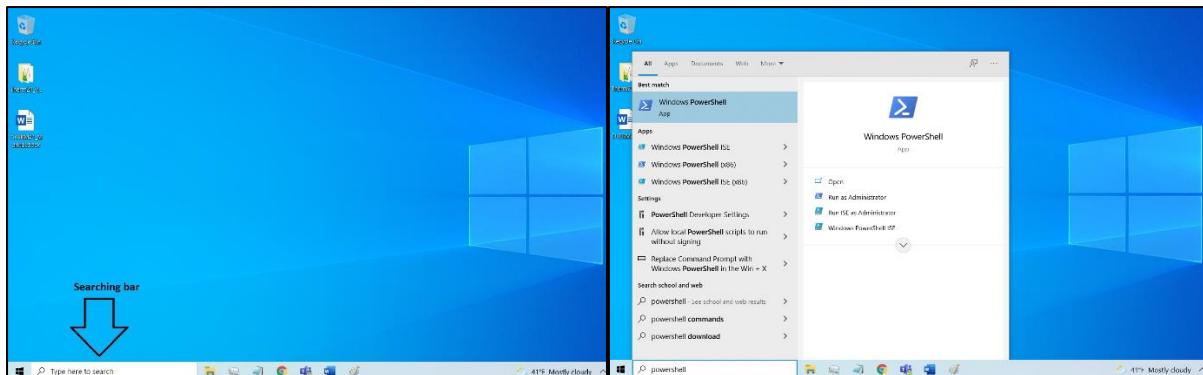


Figure 4. How to open a PowerShell terminal from the searching bar option. Searching bar indicator (left), and the PowerShell keyword option displayed (right).

Then, by selecting the first option “Windows PowerShell”, highlighted at the top of the Fig. 4 (right), a PowerShell terminal will open. PowerShell is a practical tool that can be used to check the installation in a simple and convenient way. To determine the Python installation version, type the command ‘python -V’ in the PowerShell window and then press ‘Enter’. The user is presented then with a screen as shown in Fig. 5.



```
Windows PowerShell

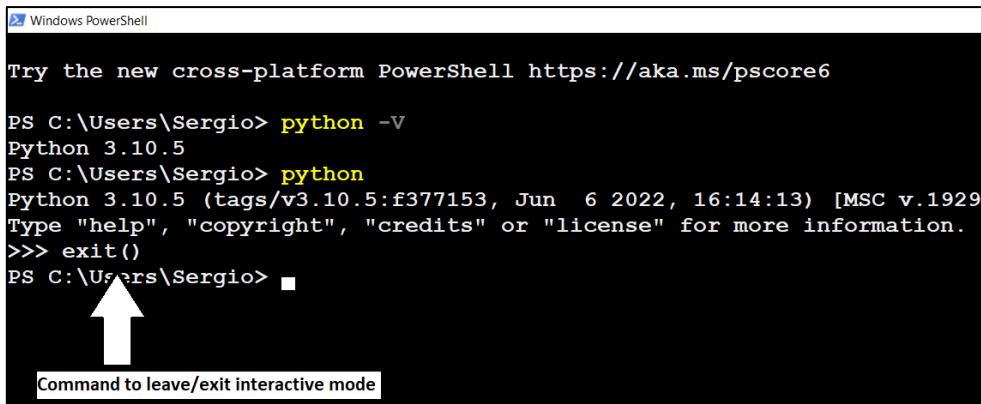
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Sergio> python -V
Python 3.10.5
PS C:\Users\Sergio>
```

Figure 5. Python's version installed screen.

Type the word ‘python’ in the PowerShell terminal window and press ‘Enter’. This will open Python’s interactive option wherein the user can check modules and even write some code if needed. However, it is not fully recommended to write a code in this mode. This will be used only to check the Python installation, and the module’s response. To leave this interactive mode one can type the word ‘exit()’ and press ‘Enter’, as shown in Fig. 6.

Supplementary Material



```
Windows PowerShell

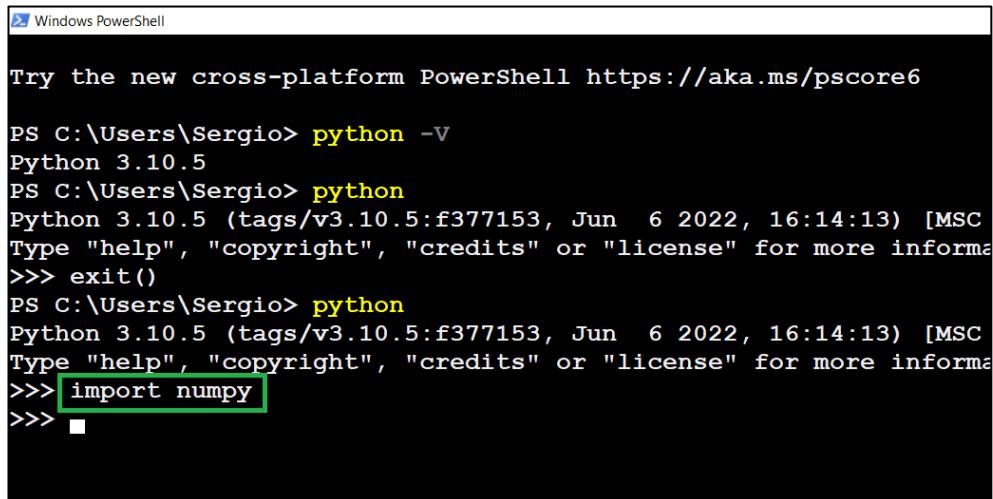
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Sergio> python -V
Python 3.10.5
PS C:\Users\Sergio> python
Python 3.10.5 (tags/v3.10.5:f377153, Jun  6 2022, 16:14:13) [MSC v.1929
Type "help", "copyright", "credits" or "license" for more information.
>>> exit()
PS C:\Users\Sergio> ■
```

Command to leave/exit interactive mode

Figure 6. Python's interactive mode and the command to exit from this mode.

To get back to Python's interactive mode a user can type 'python' again and press 'Enter' and check that the modules are already installed in the Python program. The modules needed to execute the THERM23 program are: numpy, pandas, matplotlib, scipy, sympy, and cython. To verify whether any of these modules are already installed in the Python software, the user needs to type in the next line for every module name listed: 'import numpy' followed by pressing 'Enter'. If the screen does not throw back any error that means the module has been imported properly and is working well, as highlighted in Fig. 7. An example of how the error looks is highlighted in blue in Fig. 8.



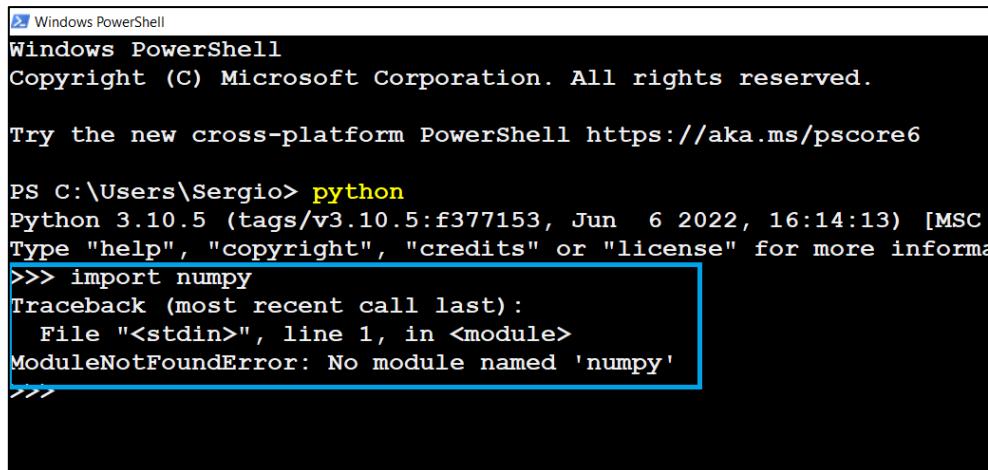
```
Windows PowerShell

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Sergio> python -V
Python 3.10.5
PS C:\Users\Sergio> python
Python 3.10.5 (tags/v3.10.5:f377153, Jun  6 2022, 16:14:13) [MSC
Type "help", "copyright", "credits" or "license" for more information.
>>> exit()
PS C:\Users\Sergio> python
Python 3.10.5 (tags/v3.10.5:f377153, Jun  6 2022, 16:14:13) [MSC
Type "help", "copyright", "credits" or "license" for more information.
>>> import numpy
>>> ■
```

Figure 7. Checking if module numpy is already installed in Python's software.

Supplementary Material



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

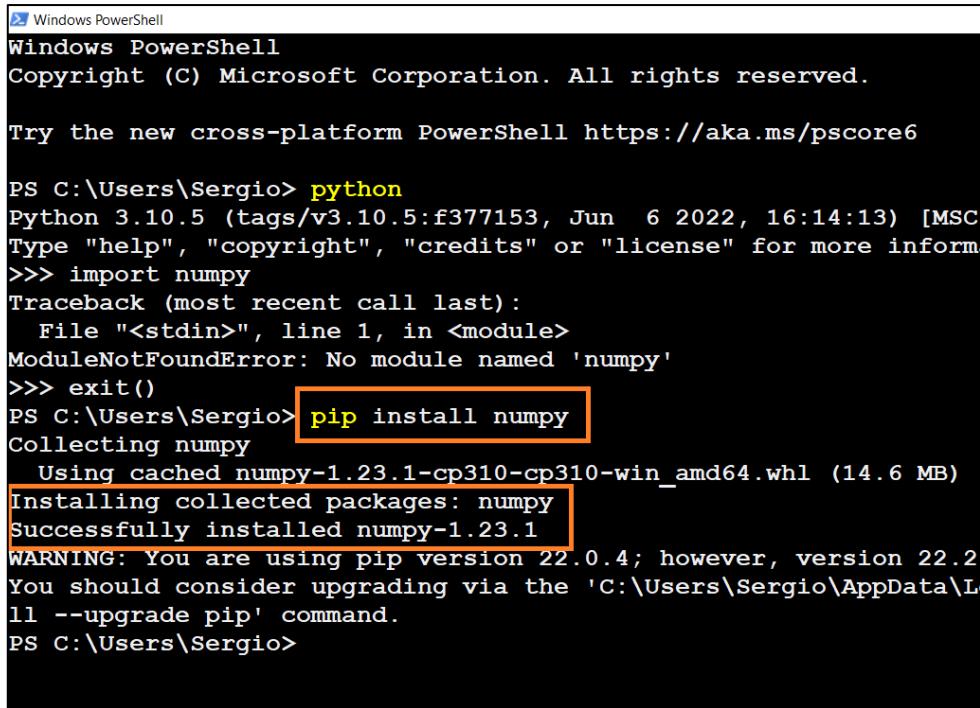
PS C:\Users\Sergio> python
Python 3.10.5 (tags/v3.10.5:f377153, Jun  6 2022, 16:14:13) [MSC
Type "help", "copyright", "credits" or "license" for more information
>>> import numpy
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'numpy'
```

Figure 8. Example of the error thrown when a module is not installed in Python's software.

If an error appears when we type any of the module's names in the list provided, as in Fig. 8, then we need to install that module manually. **IMPORTANT NOTE:** *Remember after checking whether the modules are working well or not, you need to type 'exit()' and press 'Enter' to leave the interactive mode before you continue with the installation of modules.*

Installing modules for Python's software is straightforward. If the user needs to install any of those modules just type in the PowerShell window the next line ‘pip install module name’, for example, ‘pip install numpy’, and press ‘Enter’ as highlighted in orange in Fig. 9. If a user wants to install all of the modules together, the user can type: ‘pip install numpy pandas matplotlib scipy sympy cython’ as highlighted in red in Fig. 10. It is worth noting that the computer needs access to the internet to use the pip manager and perform this action.

Supplementary Material

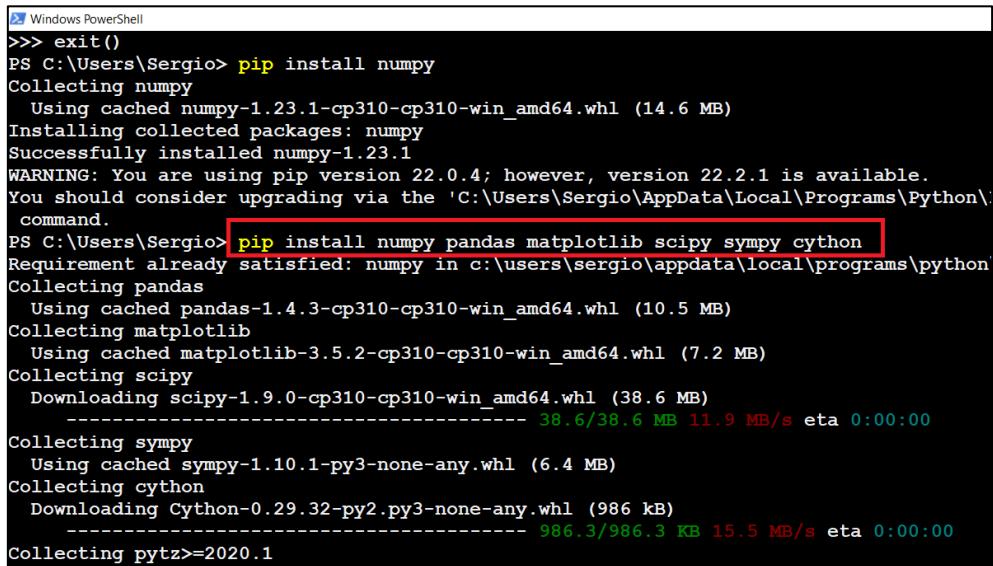


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\Sergio> python
Python 3.10.5 (tags/v3.10.5:f377153, Jun  6 2022, 16:14:13) [MSC
Type "help", "copyright", "credits" or "license" for more inform.
>>> import numpy
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'numpy'
>>> exit()
PS C:\Users\Sergio> pip install numpy
Collecting numpy
  Using cached numpy-1.23.1-cp310-cp310-win_amd64.whl (14.6 MB)
Installing collected packages: numpy
Successfully installed numpy-1.23.1
WARNING: You are using pip version 22.0.4; however, version 22.2
You should consider upgrading via the 'C:\Users\Sergio\AppData\Loc
11 --upgrade pip' command.
PS C:\Users\Sergio>
```

Figure 9. Installation of numpy module in Python using pip manager command line.



```
>>> exit()
PS C:\Users\Sergio> pip install numpy
Collecting numpy
  Using cached numpy-1.23.1-cp310-cp310-win_amd64.whl (14.6 MB)
Installing collected packages: numpy
Successfully installed numpy-1.23.1
WARNING: You are using pip version 22.0.4; however, version 22.2.1 is available.
You should consider upgrading via the 'C:\Users\Sergio\AppData\Local\Programs\Python\'
command.
PS C:\Users\Sergio> pip install numpy pandas matplotlib scipy sympy cython
Requirement already satisfied: numpy in c:\users\sergio\appdata\local\programs\python\
Collecting pandas
  Using cached pandas-1.4.3-cp310-cp310-win_amd64.whl (10.5 MB)
Collecting matplotlib
  Using cached matplotlib-3.5.2-cp310-cp310-win_amd64.whl (7.2 MB)
Collecting scipy
  Downloading scipy-1.9.0-cp310-cp310-win_amd64.whl (38.6 MB)
    38.6/38.6 MB 11.9 MB/s eta 0:00:00
Collecting sympy
  Using cached sympy-1.10.1-py3-none-any.whl (6.4 MB)
Collecting cython
  Downloading Cython-0.29.32-py2.py3-none-any.whl (986 kB)
    986.3/986.3 KB 15.5 MB/s eta 0:00:00
Collecting pytz>=2020.1
```

Figure 10. Installation of multiple modules in Python using pip manager command line.

After the package manager pip finishes installing all modules, the user can enter Python's interactive mode and type again 'import numpy' and press 'Enter' to check that no errors exist. Moreover, the user needs to do the same for the other module names, as indicated in Fig. 11. THERM23 is now ready to run. To become more familiar with the functionality of THERM23 and its capabilities please move to Section 2. If a user wants to learn how to install Python and its modules under Linux, please continue to Section 1.2.

Supplementary Material

```
Windows PowerShell
Requirement already satisfied: packaging>=20.0 in c:\users\sergi
otlib) (21.3)
Requirement already satisfied: mpmath>=0.19 in c:\users\sergio\appdata\local\Temp\pip-req-build-1jwvqz\mpmath (0.19.1)
Requirement already satisfied: six>=1.5 in c:\users\sergio\appdata\local\Temp\pip-req-build-1jwvqz\six (1.16.0)
PS C:\Users\Sergio> python
Python 3.10.5 (tags/v3.10.5:f377153, Jun  6 2022, 16:14:13) [MSC v.3.10.0.0+20220606.1]
Type "help", "copyright", "credits" or "license" for more information
>>> import numpy
>>> import pandas
>>> import matplotlib
>>> import scipy
>>> import sympy
>>> import cython
>>> exit()
PS C:\Users\Sergio> ■
```

Figure 11. Installation of multiple modules in Python.

1.2 Python installation under Linux (distribution: Fedora 30)

A copy of the THERM23 software is also available to run under the Linux environment with a directory name ‘LINUX_THERM23’. In principle, this should work in Unix systems on Apple computers or UBUNTU, but this is not guaranteed. All tests were performed on a Fedora 30 Linux operating system. To open the Fedora 30 Linux terminal window the user should click on the ‘Activities tag’ located at the top left edge of the screen. A vertical menu will pop up with the ‘Terminal’ icon and the user should click on it. Finally, a black Linux terminal window will appear in the centre of the screen, as shown highlighted by green arrows in Fig. 12. **IMPORTANT NOTICE:** *If the ‘Terminal’ icon does not appear, then the user should type the name terminal in the ‘Type to search’ bar and press ‘Enter’ as highlighted by orange arrows in Figs. 12 and 13. The ‘Terminal’ icon should pop up and the user should click on it to open it.*

Once the terminal window is open, one can check if there is a Python version installed. To do so type ‘python -V’, and press ‘Enter’. It should show the Python version already installed, similar the text highlighted in green in Fig. 14. The instructions to install Python in the Fedora 30 distribution are provided in Section 1.2.1, together with an explanation on how to check and install the corresponding modules required to run THERM23 properly in Linux.

Supplementary Material

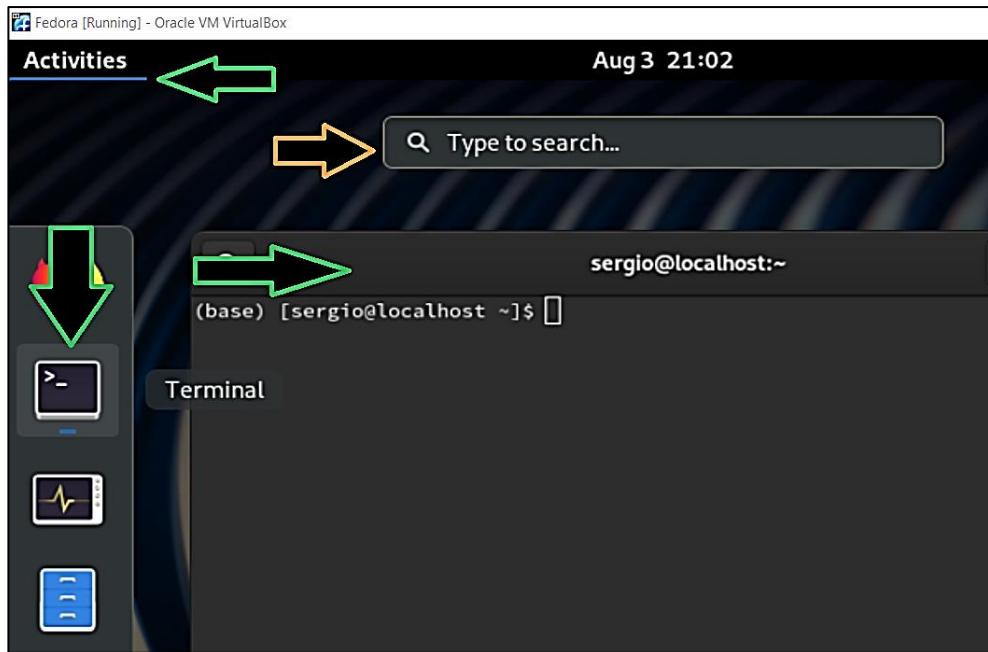


Figure 12. How to open Linux-Fedora 30 terminal. Click on Activities, then click on the Terminal icon, and the terminal windows should appear.

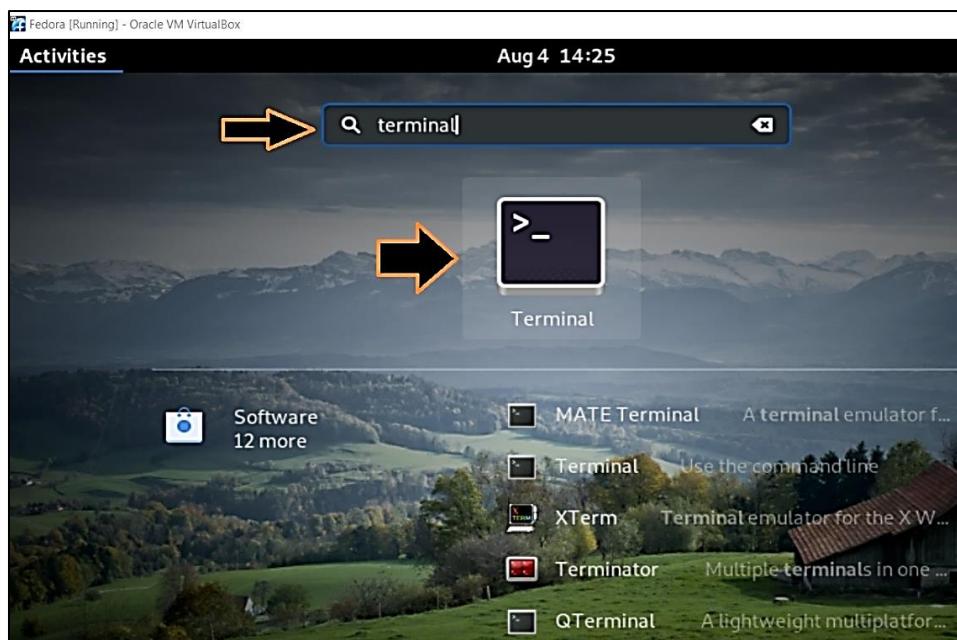


Figure 13. How to open Linux-Fedora 30 terminal. Click on Type to search bar, then type the word 'terminal' and press enter. The terminal icon should appear. Finally, click on it to open it.



Figure 14. Linux-Fedora 30 terminal screen showing the Python version installed (in green), and the error because there is no module called numpy installed yet.

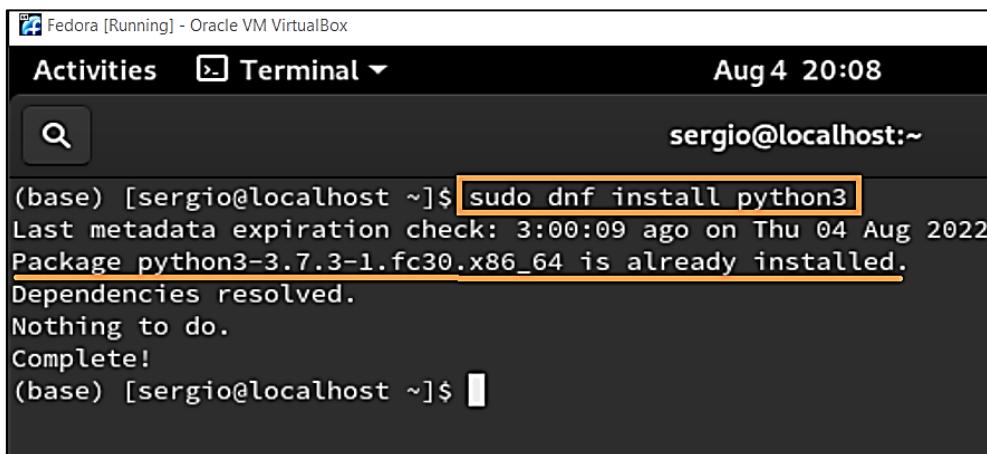
Supplementary Material

1.2.1 Installing Python in Fedora 30 and the corresponding modules

To install Python in Linux a user needs to know the proper commands for the corresponding distribution; in this tutorial we assume the user has Fedora 30 Linux Red Hat distribution. If we open a terminal window in Fedora and we have access to the internet, we can install Python straightforwardly by only typing the following:

```
sudo dnf install python3
```

sudo requires a password that the user should previously know. Two things can happen here, either a Python version is installed on the computer or it is not. If it is the user will be presented with a snapshot, highlighted in orange, similar to the one shown in Fig. 15. In this case, the system is reporting that Python 3.7 is already installed and that only the corresponding modules have to be checked.

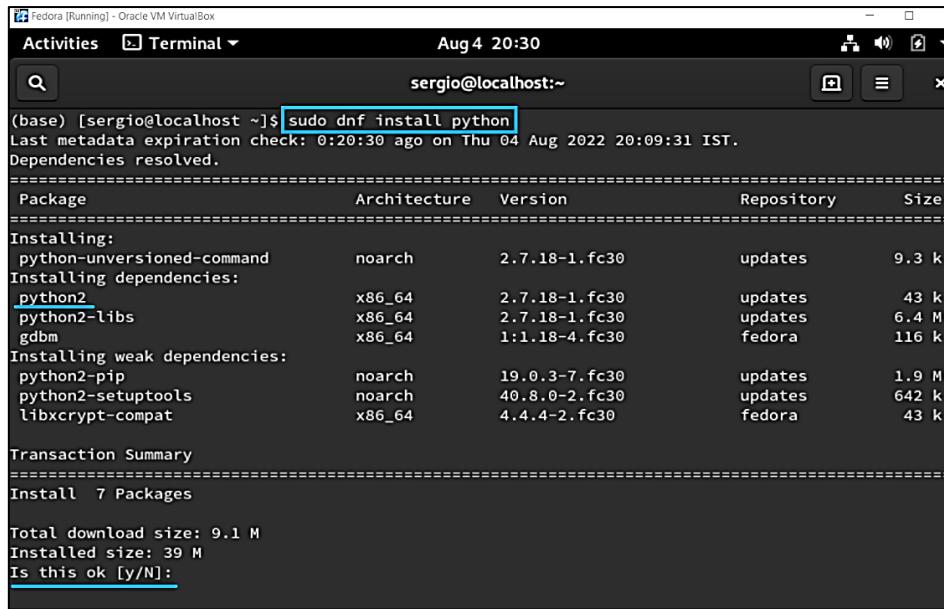


```
Fedora [Running] - Oracle VM VirtualBox
Activities Terminal Aug 4 20:08
sergio@localhost:~ (base) [sergio@localhost ~]$ sudo dnf install python3
Last metadata expiration check: 3:00:09 ago on Thu 04 Aug 2022
Package python3-3.7.3-1.fc30.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
(base) [sergio@localhost ~]$
```

Figure 15. Linux-Fedora 30 terminal screen showing that Python 3.7 is already installed.

On the other hand, if there is no version of Python is installed on the computer, a fully descriptive screen relating to the installation requirements and questions for authorization to proceed will be shown. In Fig. 16 shows an example of how to install Python without mentioning which specific version to use. The dnf manager will try to install python2 and ask for authorization to proceed, as highlighted in blue in Fig. 16. **IMPORTANT NOTE: Please remember that THERM23 needs Python 3.7 as a minimum requirement, so in our case please change the 'python' word for 'python3' in the terminal command line to get the correct version.**

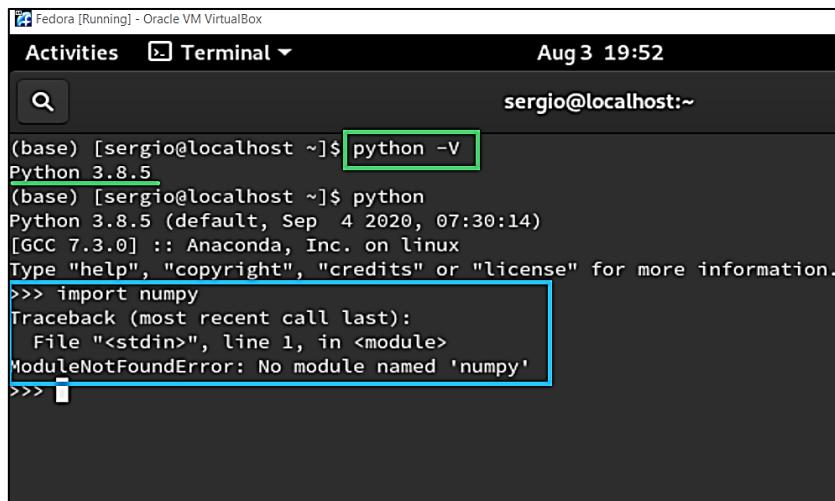
Supplementary Material



The screenshot shows a terminal window titled "Activities Terminal" running on a Fedora 30 system. The command `sudo dnf install python` is being run. The output shows the package manager resolving dependencies and installing several packages, including python-unversioned-command, python2, python2-libs, gdbm, python2-pip, python2-setuptools, and libxcrypt-compat. It also lists a transaction summary and asks if the user wants to proceed with the installation.

Figure 16. Linux-Fedora 30 terminal screen showing python2 installation display.

Finally, it is important to note that the procedure to install Python under Unix differs from that for Windows. However, the installation procedure for Python's modules in Linux is the same in Linux and Windows, see Section 1.1.1. Figure 17 presents the screen error reported by Fedora when a user tries to load/import a module that is not yet installed in Python.



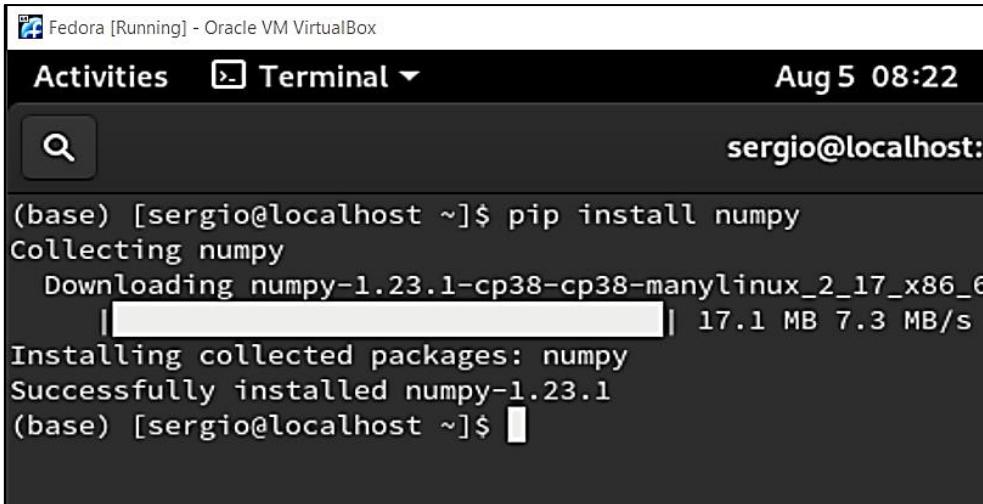
The screenshot shows a terminal window titled "Activities Terminal" running on a Fedora 30 system. The user runs `python -V` to check the Python version (3.8.5) and then attempts to import the numpy module. The import fails with a `ModuleNotFoundError: No module named 'numpy'` error message.

Figure 17. Linux-Fedora 30 terminal screen showing the common error throw by Python when a module that is not installed is imported, in this case numpy.

If this is the case, one should proceed to install the module with the next command line in our terminal window, as shown in Fig. 18.

```
pip install numpy
```

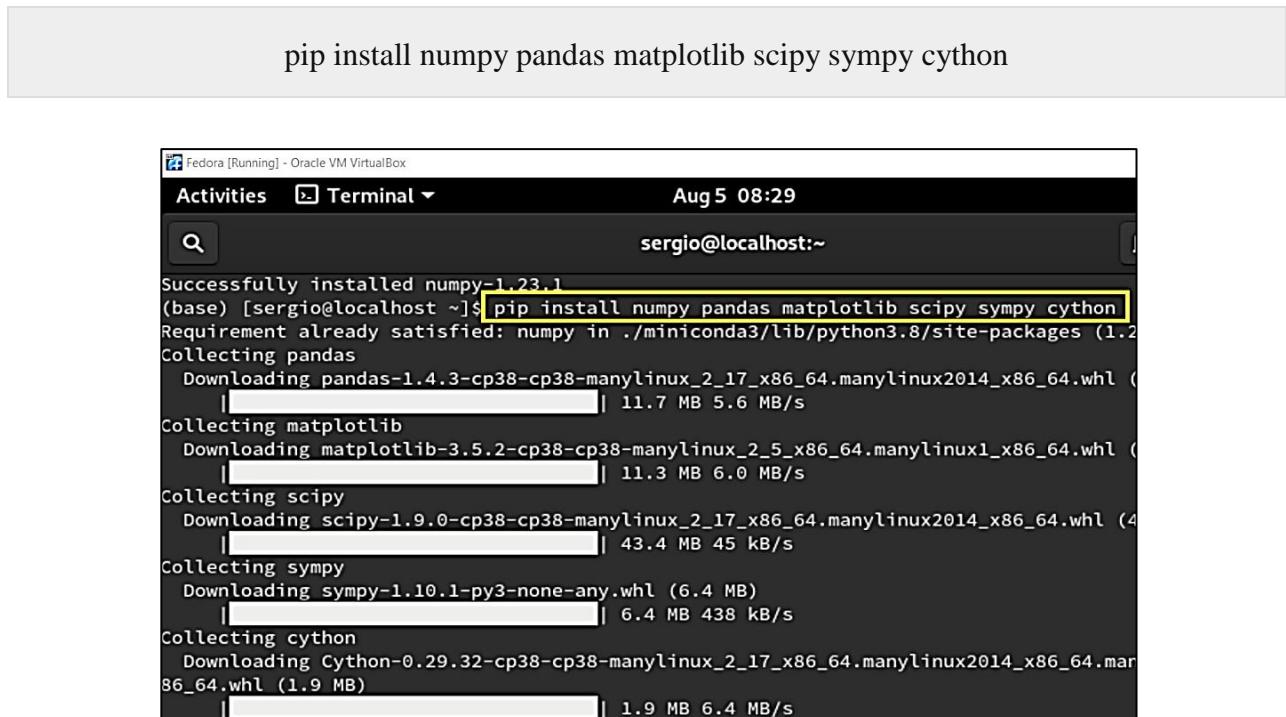
Supplementary Material



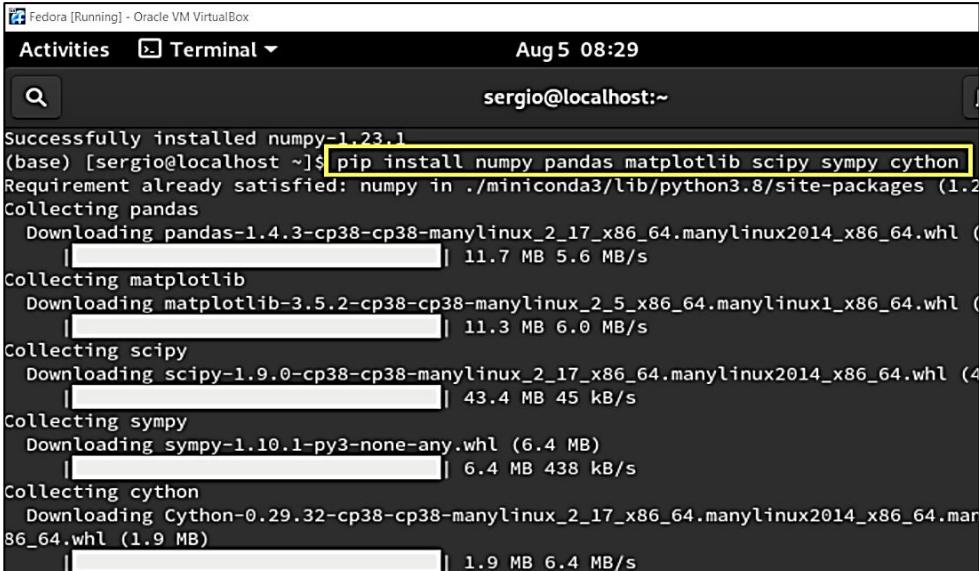
Fedora [Running] - Oracle VM VirtualBox
Activities Terminal ▾ Aug 5 08:22
sergio@localhost:
(base) [sergio@localhost ~]\$ pip install numpy
Collecting numpy
 Downloading numpy-1.23.1-cp38-cp38-manylinux_2_17_x86_64.whl
 |██████████| 17.1 MB 7.3 MB/s
Installing collected packages: numpy
Successfully installed numpy-1.23.1
(base) [sergio@localhost ~]\$

Figure 18. Linux-Fedora 30 terminal screen showing the pip manager installing the module numpy.

The user can always try to install all modules all at once by typing the command line highlighted in yellow in the terminal window in Fig. 19.



pip install numpy pandas matplotlib scipy sympy cython



Fedora [Running] - Oracle VM VirtualBox
Activities Terminal ▾ Aug 5 08:29
sergio@localhost:
Successfully installed numpy-1.23.1
(base) [sergio@localhost ~]\$ pip install numpy pandas matplotlib scipy sympy cython
Requirement already satisfied: numpy in ./miniconda3/lib/python3.8/site-packages (1.23.1)
Collecting pandas
 Downloading pandas-1.4.3-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl
 |██████████| 11.7 MB 5.6 MB/s
Collecting matplotlib
 Downloading matplotlib-3.5.2-cp38-cp38-manylinux_2_5_x86_64.manylinux1_x86_64.whl
 |██████████| 11.3 MB 6.0 MB/s
Collecting scipy
 Downloading scipy-1.9.0-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl
 |██████████| 43.4 MB 45 kB/s
Collecting sympy
 Downloading sympy-1.10.1-py3-none-any.whl (6.4 MB)
 |██████████| 6.4 MB 438 kB/s
Collecting cython
 Downloading Cython-0.29.32-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.manylinux1_x86_64.whl
 |██████████| 1.9 MB 6.4 MB/s

Figure 19. Linux-Fedora 30 terminal screen showing the pip manager installing multiple modules simultaneously for Python.

Once all of the modules are installed, they need to be checked in the interactive Python screen, as shown in Fig. 20. THERM23 is now ready to be used. To understand its capabilities and functionality, please continue to Section 2.

Supplementary Material

```
(base) [sergio@localhost ~]$ python3
Python 3.8.5 (default, Sep 4 2020, 07:30:14)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information
>>> import numpy
>>> import pandas
>>> import matplotlib
>>> import scipy
>>> import sympy
>>> import cython
>>> exit()
(base) [sergio@localhost ~]$
```

Figure 20. Linux-Fedora 30 terminal screen showing all modules have been imported properly and ready to work.

2 Operation Procedure of THERM23

2.1 THERM23 Code and Files

The main folder that contains the code is called ‘THERM23’, which is shown in Fig. 21 together with its content. Inside this directory, a list of sub-directories can be found, such as Core, GroupsDir, OutputsDir, and PythonSoftware. In addition, with those sub-directories, four files are included in this main directory, i.e., THERM23_Manual.pdf (manual of this code), THERM23.bat (executable for windows), THERM23.inp (input file for THERM23), and THERM23.py (THERM23 main Python code).

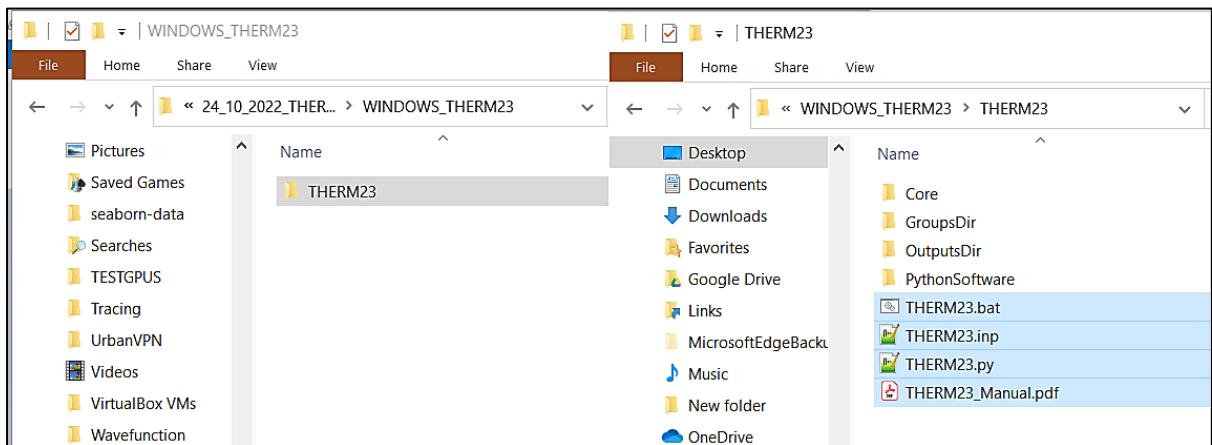


Figure 21. THERM23 main folder and its content.

The ‘Core’ directory is a folder that contains important files in Python format that THERM23 needs to work properly and efficiently (Fig.21). **IMPORTANT NOTE:** *It is strongly recommended that the user does not edit these files except if the intention is to update the classes and definitions contained therein. For this purpose, it is required a*

Supplementary Material

minimum understanding of programming in Python to understand these files and their structures. If these files undergo bad edits, it may considerably affect the functionality of THERM23.

In the second folder, ‘GroupsDir’, Fig. 22, there are eight files with the extension ‘.grp’ and one file with the extension ‘.dat’. The files with extensions ‘.grp’ contain different classes of group additivity values (GAVs) which allow the code to apply the group additivity method and calculate the desired thermochemistry. However, the content and the values of these files could be modified by a user. The user can always make new .grp files to add new GAVs or simply add them to any of the existing files. It is important to remember that to permit the code to find these GAVs it is mandatory that the extension of the file must be ‘.grp’. An example of the format of the content of one of these ‘.grp’ files is shown in Fig. 23. Here, the GAV for the C/C/H₃ group is highlighted as an example of the format, the user should realise that after every GAV name, there is a coma (,) symbol is essential and cannot be omitted. The second column presents the standard enthalpy of formation (ΔH_f^0) at 298.15 K in kcal mol⁻¹ and the third column the standard entropy (S^0) at 298.15 K in cal K⁻¹ mol⁻¹. The next seven columns refer to the heat capacity (C_p) values from 300 – 1500 K in cal K⁻¹ mol⁻¹. Any comment referring the author or source for this GAVs can be added at the end of the row, after the Cp values with five blanks separation as shown in Fig. 23.

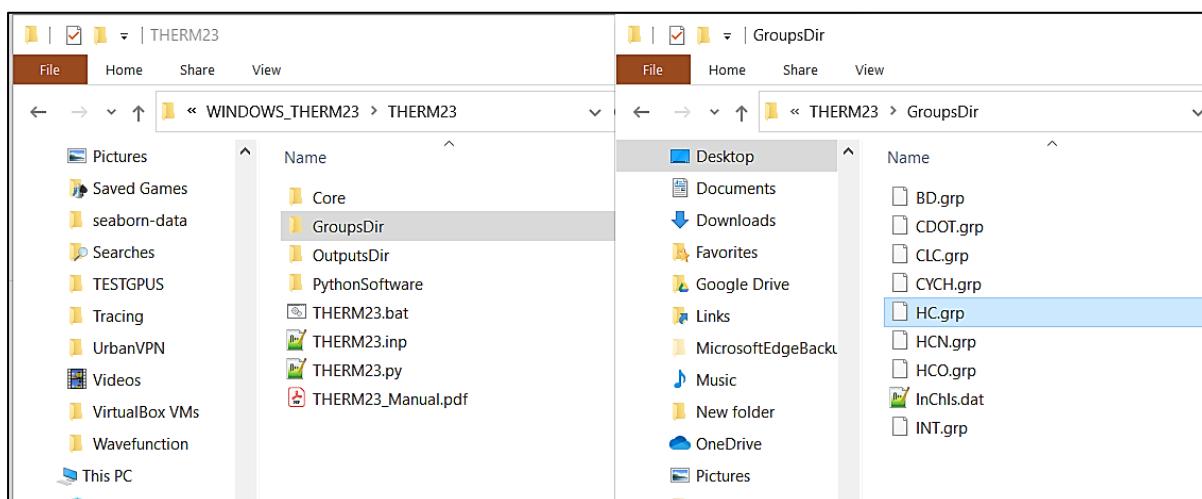


Figure 22. GroupsDir folder and its content.

Supplementary Material

UNITS:KCAL	Hf	S	Cp:	300	400	500	600	800	1000	1500		
91	-10.16	30.28	6.01	7.68	9.24	10.51	12.69	14.45	17.28	26/11/2021	MG optimisation	
C/C/H3,	-4.91	9.76	5.63	7.05	8.38	9.56	11.35	12.60	14.47	26/11/2021	MG optimisation	
C/C2/H2,	-1.68	-11.98	5.28	7.03	8.41	9.59	10.95	11.52	12.03	26/11/2021	MG optimisation	
C/C3/H,	0.14	-33.15	4.38	6.16	7.53	8.71	9.76	9.85	9.37	26/11/2021	MG optimisation	
C/C4,	-10.15	30.13	5.86	7.57	9.35	10.96	12.86	14.48	16.85	10/06/2022	Shan Zhu	
C/CT/H3,	-10.01	30.29	6.22	7.74	9.24	10.62	12.84	14.59	17.35			
CD/CD,	8.88	-14.60	4.40	5.37	5.93	6.18	6.50	6.62	6.72			
CD/CB/CD,	9.41	-14.09	3.86	4.63	5.36	5.99	6.17	6.43	6.61	10/06/2022	Shan Zhu	
CD/CB/H,	7.27	6.27	3.20	4.77	6.21	7.39	8.31	9.03	9.95	10/06/2022	Shan Zhu	
CD/C/CB,	10.07	-17.33	4.28	6.02	7.41	8.44	9.13	9.65	10.19	10/06/2022	Shan Zhu	
CD2/CT,	6.94	-16.50	4.70	6.13	6.87	7.10	7.20	7.16	7.06	D/CD/CT = CD/C/CD + { CD/C/CT - C		
CD/CD/CB,	8.00	-16.50	4.70	6.13	6.87	7.10	7.20	7.16	7.06	f est by Luo, S&Cp similar		
CD/CB/CT,	8.81	-14.22	2.69	3.77	5.42	5.18	5.64	6.07	6.37	10/06/2022	Shan Zhu	
CD/CD2,	4.60	-15.67	1.90	2.69	3.50	4.28	5.57	6.21	7.37	f=est, S, CP(T) = mopac nov99		
CD/CD/CT,	4.60	-13.92	2.07	2.64	3.93	3.54	4.65	5.38	6.45	f=est, S, CP(T) = mopac nov99		
CD/CT2,	6.46	-16.50	4.70	6.13	6.87	7.10	7.20	7.16	7.06	D/CT2 = CD/C/CT + { CD/C/CT - CD/		
CD/CB2,	8.00	-16.50	4.70	6.13	6.87	7.10	7.20	7.16	7.06	D/CB2 = CD/C/CB + { CD/C/CB - CD/		

Figure 23. HC.grp file format content.

Additionally, the file named ‘InChIs.dat’ contains a list of 192 molecular formulas, SMILE, and InChIs taken from the glossary of NUIGMech1.2 published in 2022 [2]. The content of this file is shown in Fig. 24. This is a feature that belongs to the automatic option of THERM23, which is discussed more detail in the corresponding Section 2.1.2. It is advised not to include other files in this directory except for those with extension ‘.grp’.

InChI	SMILES	Species	NameInMech	Formula
1 InChI=1S/C2H6/c1~2/h1~2H3	CC	C2H6	C2H6	C2H6
3 InChI=1S/C3H8/c1~3~2/h3H2,1~2H3	CCC	C3H8	C3H8	C3H8
4 InChI=1S/C4H10/c1~3~4~2/h3~4H2,1~2H3	CCCC	NC4H10	C4H10	C4H10
5 InChI=1S/C5H12/c1~3~5~4~2/h3~5H2,1~2H3	CCCCC	NC5H12	NC5H12	C5H12
6 InChI=1S/C6H14/c1~3~5~6~4~2/h3~6H2,1~2H3	CCCCCC	NC6H14	NC6H14	C6H14
7 InChI=1S/C7H16/c1~3~5~7~6~4~2/h3~7H2,1~2H3	CCCCCC	NC7H16	NC7H16	C7H16
8 InChI=1S/C4H10/c1~4~(2)~3/h4H,1~3H3	CC(C)	IC4H10	IC4H10	C4H10
9 InChI=1S/C5H12/c1~4~5~(2)~3/h5H,4H2,1~3H3	CCC(C)C	IC5H12	IC5H12	C5H12
10 InChI=1S/C5H12/c1~5~(2,3)~4/h1~4H3	CC(C)(C)C	NEOC5H12	NEOC5H12	C5H12

Figure 24. ‘InChIs.dat’ file format content.

The ‘OutputsDir’ folder contains all the files generated by THERM23 during its execution, along with those files that the user may use for re-calculating, thermo plotter and thermo fitter options, which are explained in detail in the next Sections 2.1.3, 2.1.4, and 2.1.5, respectively. The ‘PythonSoftware’ folder contains Python version python-3.10.5-amd64.exe, which is suitable for computers with 64-bit Windows 10 architecture. The main code is included in the file ‘THERM23.py’, and it can be run through the ‘.bat’ file for Windows users and directly from the terminal for Linux ones.

2.2 THERM23 options

The code has five main options for calculating the thermochemical properties and plotting. These are Interactive, Automatic, Re-calculate, Thermo plotter, and Thermo fitter. All of these options are discussed below.

Supplementary Material

The welcome screen is highlighted with different colours for the different sections in Fig. 25. The green rectangle shows the welcome section with information related to the developers and contact information is displayed. In the red rectangle section, the output and GAV files directories are shown labelled either as ‘already exists!’ or ‘successfully created!’. If the directory ‘OutputsDir’ does not exist in the main folder of THERM23, it will be created by THERM23 when it runs for the first time. The ‘GroupsDir’ folder must exist, or the code will give an error on the screen and stop functioning. The blue rectangle shows a resume of the ‘GroupsDir’ folder and the name of the GAV files. Finally, the yellow rectangle indicates the THERM23 program’s options or modes available. The following sub-sections will explain each option/mode in more detail. To access any of the modes listed on the welcome screen, the user needs to type the corresponding number and press ‘Enter’. The user can always end the program by either typing the word ‘END’ or ‘QUIT’ in capitals or lowercase. THERM23 is not case-sensitive for this command.

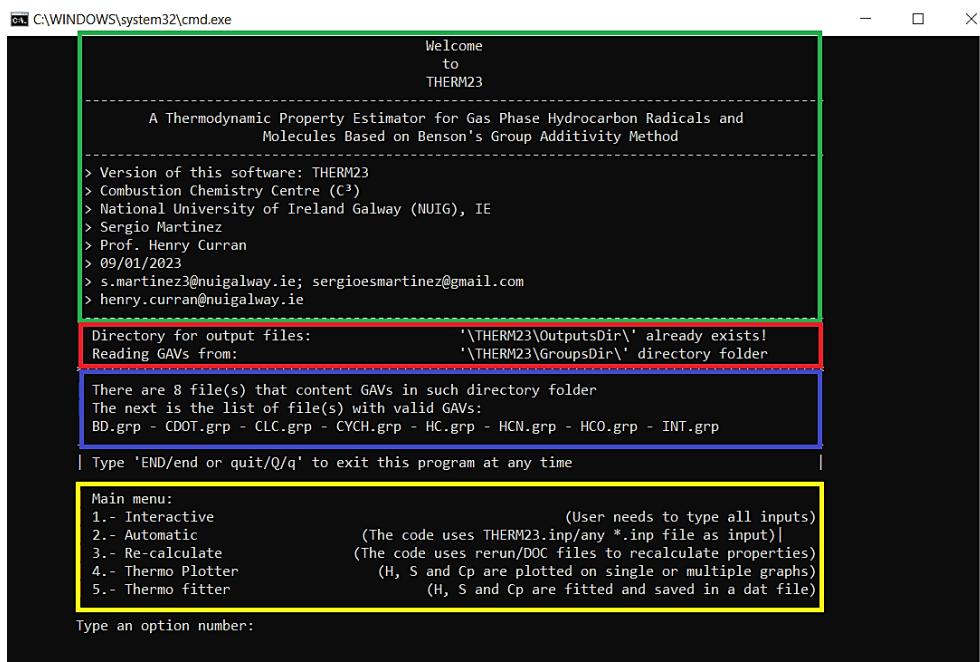


Figure 25. THERM23’s welcome screen divided in sections. These sections are highlighted by a colour code in order to understand them better.

2.2.1 Interactive mode

The first option or mode available in THERM23 is the ‘Interactive’ mode, as shown in Fig. 25. To calculate thermochemistry in ‘Interactive’ mode, the user should type ‘1’ and press ‘Enter’ as shown in Fig. 26. As a consequence, the code will print the line ‘Calculating new thermochemistry in Interactive mode...’, which indicates to the user that the ‘Interactive’

Supplementary Material

mode is successfully activated. Then, the user is prompted to provide a ‘name’ for the output file, for example, type ‘Example1’ and press ‘Enter’. This will generate a collection of five output files or items with different extensions including: ‘Example1.dat’, ‘Example1.doc’, ‘Example1.LST’, ‘Example1.rerun’, and ‘Example1.tmp’ as presented in Fig. 27. A proper explanation of these files, formats and features will be given at the end of Section 2.1.3. These files are going to be generated by the code either ‘Interactive’, ‘Automatic’, or ‘Recalculate’ modes. However, firstly we need to feed the code with information to generate the thermochemistry data and save it on those files.

The screenshot shows a terminal window with the following text:

```
| Type 'END/end or quit/Q/q' to exit this program at any time
| Main menu:
| 1.- Interactive          (User needs to type all inputs)
| 2.- Automatic           (The code uses THERM22_V2.0.inp/any *.inp file as input)
| 3.- Re-calculate         (The code uses rerun/DOC files to recalculate properties)
| 4.- Thermo Plotter       (H, S and Cp are plotted on single or multiple graphs)
| 5.- Thermo fitter        (H, S and Cp are fitted and saved in a dat file)

Type the option number you want to run: 1
Calculating new thermochemistry in Interactive mode...
Please provide the output file name:
Example1

|> Running in Interactive mode.

=====
Type the species' name (molecular name):
```

Figure 26. Running THERM23 in ‘Interactive’ mode. In the red rectangle ‘1’ was typed to choose the ‘Interactive’ mode, and in the green one, ‘Example1’ was typed to provide an output files name.

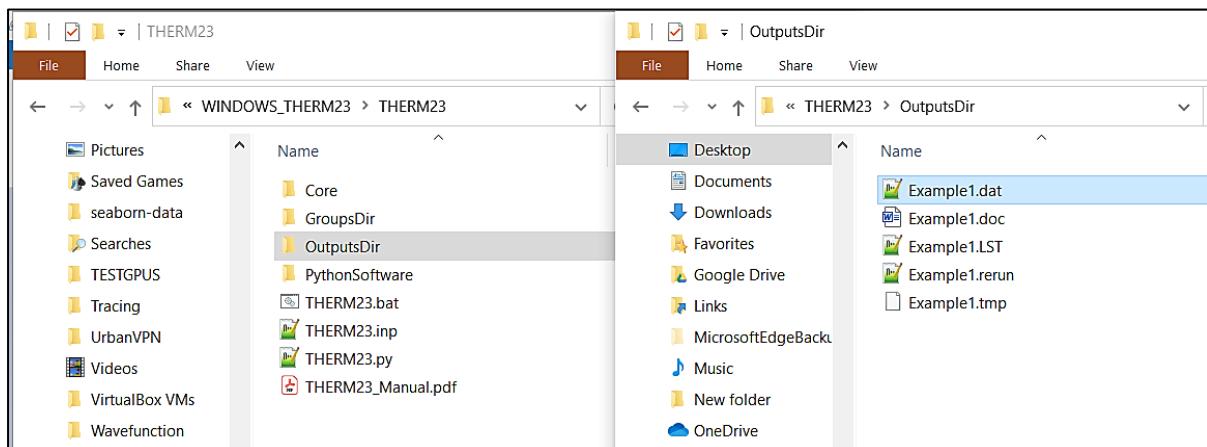


Figure 27. Files generated by Interactive mode when a user provides a file name Example1. A whole collection of 5 different items is generated automatically when a user provides the file name, however, it is important to notice these files are still empty and open. Users must wait until the program is closed or stopped to open or visualise these files, in special the doc one.

Once a user provides a file name for the output files, the user will be presented with a screen asking to type the species’ name with the next line ‘Type the species name (molecule mechanism name)’. This is quite important because it is not only a label or name to identify the species (molecule or radical) of interest, but it refers to the name of this specific species in

Supplementary Material

a mechanism. Therefore, it is better to put the name consistent with the name used in the mechanism so that the user does not need to make any alterations when using it in a mechanism. To the left of Fig. 28 and indicated by a black arrow the corresponding name of ethane is ‘C₂H₆’ in the NUIGMech1.3 [3] mechanism. Thus this name should be provided as the answer to the question posed by THERM23. **IMPORTANT NOTE:** *For ethane (C₂H₆), it is a coincidence that the species name in the mechanism is similar to its molecular formula, which is not true for all species, particularly for larger hydrocarbon and oxygenated species or more complex molecules. Therefore, a user needs to pay special attention to this detail in their mechanism.*

The screenshot shows two windows. On the left is a Notepad++ window titled 'NUIGMech1.2.MECH' containing the mechanism code. A black arrow points to the line '1\SUBMECH: C2H6'. On the right is a terminal window titled 'C:\WINDOWS\system32\cmd.exe' showing the output of the code execution. An orange rectangle highlights the command 'Type the species' name (molecular name):' where 'c2h6' is typed. A purple rectangle highlights the command 'Type the chemical formula of C2H6:' where 'C2H6' is typed again. Below these, the terminal asks for the number of carbon and hydrogen atoms, and whether the molecule is linear or non-linear.

```

C:\Users\Sergio\Downloads\NUIGMech1.2.MECH - Notepad++  C:\WINDOWS\system32\cmd.exe
File Edit Search View Encoding Language Settings Tools Mac
File Edit Search View Encoding Language Settings Tools Mac
NUIGMech1.2.MECH [1]
115 !-----+
116 !-----+
117 !END_SPECIES_MODULE: C1
118 !-----+
119 !-----+
120 !SPECIES_MODULE: C2
121 !-----+
122 !-----+
123 !SUBMECH: C2H6
124 !-----+
125 !\LTOXID
126 C2HSO2H
127 C2HSO2
128 !\PYROSPEC
129 C2H6
130 C2H5
131 !
132 !-----+
133 !\SUBMECH: C2H4
134 !-----+
135 !\LTOXID
136 CHOCHO
137 C2H3OOH
138 C2H3OO
139 CHCHO
140 !\PYROSPEC
141 C2H4

```

```

There are 8 file(s) that contain GAVs in such directory folder
The next is the list of file(s) with valid GAVs:
BD.grp CDT.grp CLC.grp CYCH.grp HC.grp HCO.grp INT.grp

| type 'END/end or quit/Q/q' to exit this program at any time

Main menu:
1.- Interactive (User needs to type all inputs)
2.- Automatic (The code uses THERM23_V2.0.inp/any *.inp file as input)
3.- Re-calculate (The code uses rereun/DOC files to recalculate properties)
4.- Thermo Plotter (H, S and Cp are plotted on single or multiple graphs)
5.- Thermo fitter (H, S and Cp are fitted and saved in a dat file)

Type the option number you want to run: 1
Calculating new thermochemistry in Interactive mode...
Please provide the output file name:
Example1
> Running in Interactive mode.

Type the species' name (molecular name): c2h6
Type the chemical formula of C2H6: C2H6
Number of carbon (C) atoms in C2H6? : 2
Number of hydrogen (H) atoms in C2H6? : 6
Is this molecule linear?: (yes/y or no/n) no

Is the data provided above correct?: (y/n)

```

Figure 28. Left side: NUIGMech1.2.MECH showing that the specie's name for ethane is C₂H₆. On the right side: user entries for specie's name, chemical formula, number of C, and H atoms in chemical formula, and if this molecule is linear or not.

On the right side of Fig. 28, indicated by an orange rectangle, the user can type ‘C₂H₆’ as an answer to the question and press ‘Enter’. Next, the code asks for the molecular formula, ‘Type the chemical formula on C₂H₆:’. At this stage, it does not matter if the user type in lowercase, capital or a mix of both, it would show the name of the species in capital letters. In this case, we can type the same C₂H₆ as follow in the violet rectangle of the same picture. Note that for a chemical formula, a user is only permitted to type formulas with carbon (C), hydrogen (H), oxygen (O) and nitrogen (N) atoms. Based on this chemical formula, the code will ask the user for the total number of C, H, O or N atoms in the species name. Once these questions are answered, the user is requested to let the code know if the species of interest is a linear or non-linear molecule. In the case of ethane, it is a non-linear molecule then we type ‘no’, and press ‘Enter’. **IMPORTANT NOTE:** *An example of a linear molecule is ‘CO₂’, then for this molecule user must answer ‘yes’ and press ‘Enter’ to enable proper corrections and accurate calculations.*

Supplementary Material

To the confirmation of all the information has been provided by the user: ‘Is the data provided above correct? (yes/y or no/n)’ will appear. If the user answer ‘yes’, then the code will move to the next question ‘Total number of groups in molecule? (Must be an integer)’. However, if the user answered ‘no’, the code would ask the user for all the previous questions one more time, as shown in Fig. 29 highlighted by the blue rectangle, before it moves automatically to the next question.

```
Type the option number you want to use: 1
Calculating new thermochemistry in Interactive mode...
Please give a name for the output files:
Example1
|> Running in Interactive mode.
=====
Type the specie's name (molecule mechanism name):      ethane
Type the chemical formula on ETHANE:                  c2h6
Number of carbons (C) atoms in ETHANE? :             2
Number of hydrogen (H) atoms in ETHANE? :            6
Is this molecule linear?: (yes/no)                   no
=====
Is the data provided above correct?: (yes/y or no/n)  no
Type the specie's name (any string or formula):       c2h6
Type the chemical formula on C2H6:                    c2h6
Number of carbons (C) atoms in C2H6? :              2
Number of hydrogen (H) atoms in C2H6? :             6
Is this molecule linear?: (yes/y or no/n)           no
```

Figure 29. Example of what happen if the user answered 'no' to the 'Is the data provided above, correct?'. User must answer again all the previous questions.

As mentioned previously, the next question is ‘Total number of groups in molecule? (must be an integer)’ this question refers to the number of GAVs that will be used to generate the thermochemistry. For ethane (C_2H_6), there are $2 \times C/C/H3$ and the user should type ‘2’ and press ‘Enter’. Now the code will sum the groups given by the user until the total sum is 2. The user should provide the GAV name as ‘ $C/C/H3$ ’ leave a blank space and type the number of times the GAV will be used, such as ‘ $C/C/H3 2$ ’ and press ‘Enter’. If the user provides only the GAV name without a number, the code will understand that the number of GAV is one, Fig. 30.

Supplementary Material

```
Please provide the output file name:  
Example1  
=====|> Running in Interactive mode.  
=====Type the species' name (molecular name): c2h6  
Type the chemical formula of C2H6: c2h6  
Number of carbon (C) atoms in C2H6? : 2  
Number of hydrogen (H) atoms in C2H6? : 6  
Is this molecule linear?: (yes/y or no/n) no  
=====Is the data provided above correct?: (y/n) y  
Keeping data  
=====Total number of groups in molecule?: (must be an integer) 2  
Give the group's name + Quantity  
Type either C/C/H3 1 or c/c/h3 1, or simply give the name and press enter; C/C/H3  
1 - C/C/H3 2  
=====Are the groups provided above correct?: (y/n)
```

Figure 30. Example of providing a '2' as total GAVs to be used in pink rectangle. The GAV for carbon–carbon–3 × hydrogen bond is typed and ask the code to use it twice '2' pictured in the green rectangle of the picture.

If the user makes a mistake by typing the total number of GAVs or providing a wrong GAV's name, the code can fix it once in the next step as shown in Fig. 31. By answering the question 'Are the groups provided above, correct?' with 'no', this would display the option to delete as many GAVs are mistyped by the user. To do this, the user needs to let the code know how many GAVs should delete and then provide the number in the list. For this example, in Fig. 31, inside the red rectangle, it was typed 'C/C/H3' which is wrong because it should be '2' after the GAV's name. For this purpose, the user should type 'no' to the question in the green rectangle. Now it can be seen the option for 'How many would you like to delete?' and type '1' and again '1', because the GAV is listed as the number '1 – C/C/H3', see the yellow rectangle in Fig. 31.

Then the program will throw on screen the expression 'New list is: List is empty now' which means it is successfully deleted the GAV and now the user can provide a new total number of GAVs to use. In this case, the question has changed by 'Give the number of additional groups: (must be an integer)' so type '2' which is right GAV's name and the number of GAV as 'C/C/H3 2' shown in the blue rectangle of Fig. 31.

Finally, the user needs to provide the code the symmetry number and the number of rotors of the species of interest. The symmetry number should consider both contributions internal and external. For the rotors, all non-linear species, like ethane, have $3N-6$ vibrational degrees of freedom. Therefore, in the case for ethane, there is only one vibrational mode corresponding to the rotation of the C–C bond and have one rotor. In Fig. 32 inside the green rectangle, a symmetry number of 18 and a rotor number equal to 1. In ethane, each methyl group (CH_3) has a symmetry number of 3 through 120° , making then $3 \times 3 = 9$ of symmetry plus an external plane of symmetry, through 180° , adding a factor of 2, leading to $9 \times 2 = 18$.

Supplementary Material

```

| Main menu:
| 1.- Interactive           (User needs to type all inputs)
| 2.- Automatic             (The code uses THERM22_V2.0.inp/any *.inp file as input)
| 3.- Re-calculate          (The code uses rerun/DOC files to recalculate properties)
| 4.- Thermo Plotter         (H, S and Cp are plotted on single or multiple graphs)
| 5.- Thermo fitter          (H, S and Cp are fitted and saved in a dat file)
-----
Type the option number you want to run:      1
Calculating new thermochemistry in Interactive mode...
Please provide the output file name:
Example1
=====
|> Running in Interactive mode.
=====
Type the species' name (molecular name):      C2H6
Type the chemical formula of C2H6:            C2H6
Number of carbon (C) atoms in C2H6? :        2
Number of hydrogen (H) atoms in C2H6? :       6
Is this molecule linear?: (yes/y or no/n)    NO
-----
Is the data provided above correct?: (y/n)     Y
Keeping data
=====
Total number of groups in molecule?: (must be an integer)   1
Give the group's name + Quantity
Type either C/C/H3 1  or  c/c/h3 1, or simply give the name and press enter; C/C/H3
1 - C/C/H3
=====
Are the groups provided above correct?: (y/n)  NO
How many would you like to delete?
1
# of the GAV to delete:
1
GAV removed
New list is:
List is empty now
=====
Give the number of additional groups: (must be an integer)  2
1 - C/C/H3 2
-----
Total symmetry number:

```

Figure 31. How to correct the GAV or GAVs if user mistyped them.

```

Type either C/C/H3 1  or  c/c/h3 1, or simply give the name and press enter; C/C/H3
1 - C/C/H3 2
-----
Are the GAVs provided above correct?: (yes/y or no/n)  YES
Keeping current GAVs
=====
Symmetry number (internal+external):  18
Number of rotors:  1
*****
Summarising:
SpecieName: C2H6
Formula: C2H6
Number of GAVs: 2
GAVs and Quantity:
C/C/H3 2
Number of rotors: 1
Symmetry number: 18
-----
Calculations:
Speciename  H(298K)  S(298K)  Cp300K  Cp400K  Cp500K  Cp600K  Cp800K  Cp1000K  Cp1500K  date
C2H6    -20.32 54.94 12.02 15.36 18.48 21.02 25.38 28.9 34.56 21/12/2021
Cp0     = 6.65
CpINF   = 42.75
-----
Do you want to save this data on file? (y/n)
-
```

Figure 32. Symmetry number and number of rotors in the green rectangle. Resume of the parameters provided and the proper calculations generated by THERM23.

Inside the blue rectangle in Fig. 32 is shown the code displays a resume of all the parameters the user has provided. It shows the proper calculations of the thermochemical properties listed on the screen; i.e. Species name, $H(298K)$, $S(298K)$, and Cp from 300 – 1500 K and finally the date. At the end of this screen, the final question is shown, ‘Do you want to save this data on file? (y/n)’. If the user says ‘yes’, the code will fill with the data generated by the corresponding files in the ‘OutputsDir’ folder otherwise the code will

Supplementary Material

continue its execution without saving any of the previously generated data, as highlighted in the blue rectangle in Fig. 33.

```
C2H6   -20.32 54.94 12.02 15.36 18.48 21.02 25.38 28.9 34.56 21/12/2021
Cp0    =  6.65
CpINF  = 42.75
.....
Do you want to save this data on file? (y/n)
Y
.....
C:\Users\Sergio\Desktop\WINDOWS_THERM21\Therm21_V4.0\OutputsDir\Example1_21_12_2021.doc updated.
C:\Users\Sergio\Desktop\WINDOWS_THERM21\Therm21_V4.0\OutputsDir\Example1_21_12_2021.rerun updated.
C:\Users\Sergio\Desktop\WINDOWS_THERM21\Therm21_V4.0\OutputsDir\Example1_21_12_2021.LST updated.
C:\Users\Sergio\Desktop\WINDOWS_THERM21\Therm21_V4.0\OutputsDir\Example1_21_12_2021.dat updated.
Please find your data in '\Therm21\OutputsDir' directory

Do you want to calculate a radical from this specie? (y/n)
Y
How many radicals do you want to calculate? (must be an integer)
1
Type the radical's name (radical mechanism name):      C2H5
Type the chemical formula on C2H5:      C2H5
Number of carbons (C) atoms in C2H5? :      2
Number of hydrogen (H) atoms in C2H5? :      5
Is this molecule linear?: (yes/no)      N
.....
Is the data provided above correct?: (yes/y or no/n)      Y
```

Figure 33. The picture shows how the code will save the data calculated in the different files with extensions, i.e. .doc, .rerun, .LST, and .dat are shown in the blue rectangle. Inside the yellow rectangle can be seen the parameters provided by the user for the calculation of the radical C₂H₅.

Figure 33 presents the option to calculate a specific number of radical species highlighted in the yellow rectangle. First, the user should answer the question (y/n) ‘Do you want to calculate a radical from this species?’ If the answer is ‘yes’, the code will skip this step and continue with the next species, as indicated in Fig. 34. If the user decides to calculate the radical species based on the corresponding parent molecule, the code will prompt questions similar to those for the parent molecule, see Figs. 35 and 36.

```
......
Data was not saved.
Do you want to calculate a radical from this species? (y/n)
n
......
Next species...

Type the species' name (molecular name):
```

Figure 34. Skipping the calculation of a radical specie from its parent specie.

Supplementary Material

```
WS\system32\cmd.exe
Number of hydrogen (H) atoms in C2H5? :      5
Is this molecule linear?:      (yes/no)      N
-----
Is the data provided above correct?: (yes/y or no/n)      Y
Keeping data
-----
These are the parent molecule (C2H6) GAVs:
1 - C/C/H3

Keep these groups? (yes/no) - press Enter for default 'yes'
if yes, you will be asked to give only the radical's GAVs to sum to the parent ones
if you answered no, you will be asked for the number of GAV to delete

Keeping parent's GAVs
Moving to the radicals
-----
The next group(s) is/are going to be added to the parent's ones
Give the number of additional groups: (must be an integer)      1
Type either C2H5 1 or c2h5 1, or simply give the name and press enter; c2h5
C2H5
Symmetry number (internal+external):      6
Number of rotors:      1

Summarising:
RadicalName: C2H5
Formula: C2H5
```

Figure 35. Calculation of radical specie based on parent molecule information.

```
Number of rotors:      1
*****
Summarising
RadicalName:      C2H5
Formula:      C2H5
Number of GAVs:      3
GAVs and Quantity:
1 - C/C/H3      - 2
2 - C2H5      - 1
Number of rotors:      1
Symmetry number:      6

.....
Calculations:
Speciename      H(298K)  S(298K)  Cp300K  400K   500K   600K   800K   1000K  1500K
C2H5          28.68    59.16   11.85   14.79   17.21   19.36   22.96   25.76   30.27
Cp0      =  6.65
CpINF     =  37.05

Do you want to save this data on file? (y/n)
y
-----
File Example1.doc updated.
File Example1.rerun updated.
File Example1.LST updated.
File Example1.dat updated.
Please find your data in '\THERM22\OutputsDir' directory

Next species...
```

Figure 36. Code display of the properties calculated for the radical specie.

The text highlighted in red in Fig. 35, and blue and purple in Fig. 36, depict the similar procedure to calculate the thermochemical properties of a radical species based on a parent molecule.

2.2.2 Automatic mode

The second option or mode available in THERM23 is ‘Automatic’ mode, as indicated in Fig. 25. To activate this mode, the user should type ‘2’ and press ‘Enter’, as shown inside

Supplementary Material

the green rectangle in Fig. 37. The code will print the line ‘Calculating new thermochemistry in automatic mode...’, which informs the user that ‘Automatic’ mode has been successfully activated. The user is required to provide a name for the output files, for example ‘Example2’, by typing ‘Example2’ and pressing ‘Enter’. This will generate five output files or items with different extensions including ‘Example2.dat’, ‘Example2.doc’, and ‘Example2.LST’, ‘Example2.rerun’, and ‘Example2.tmp’, as indicated in Fig. 38.

```
| Type 'END/end or quit/Q/q' to exit this program at any time |
| Main menu: |
| 1.- Interactive |
| 2.- Automatic | (User needs to type all inputs) |
| 3.- Re-calculate | (The code uses rerun/DOC files to recalculate properties) |
| 4.- Thermo Plotter | (H, S and Cp are plotted on single or multiple graphs) |
| 5.- Thermo fitter | (H, S and Cp are fitted and saved in a dat file) |

Type an option number: 2
Calculating new thermochemistry in automatic mode
Please provide the output file name:
Example2

List of input files available to run in automatic mode:
=====
DataWithInChIs.inp
THERM22_V2.0.inp
=====

How many input files would you like to use? (Provide an integer or if all use the keyword 'all')
If you don't provide a name and only press 'Enter' the code will run the first file in the list
```

Figure 37. THERM23’s automatic mode activation and output file name ‘Example2’ provided by user.

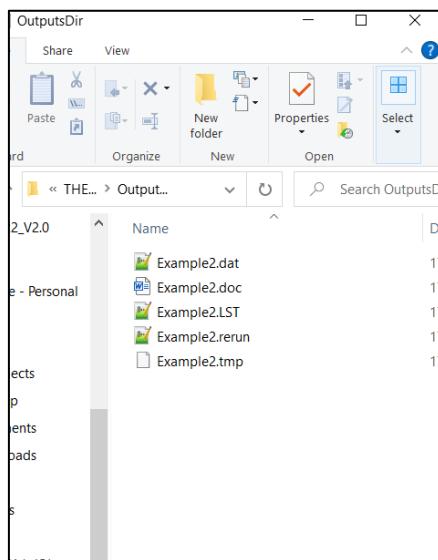


Figure 38. Output files generated by code in automatic mode; ‘Example2.dat’, ‘Example2.doc’, ‘Example2.LST’, ‘Example2.rerun’, and ‘Example2.tmp’.

When the user provides a file name for the output files, the user will see a list of input files available in the main directory, Fig. 39. This list includes only one file (at the time of

Supplementary Material

writing this manual), this file is provided together with THERM23 software with an example of every parameter needed to calculate the thermodynamic properties of a list of species. Every .INP file is required to contain 14 columns with headers as follows; ‘SpeciesName’, ‘Formula’, ‘NoGAVs’, ‘GAVsName’, ‘Quantity’, ‘Symmetry’, ‘Rotors’, ‘IfRad’, ‘ParentMol’, ‘C’, ‘H’, ‘O’, ‘N’, and ‘Linearity’. As shown in Fig. 40, these columns must be separated by a special character ‘Tab’. However, in between the columns and the ‘Tab’ character the user can insert ‘blank space’ as pleased.

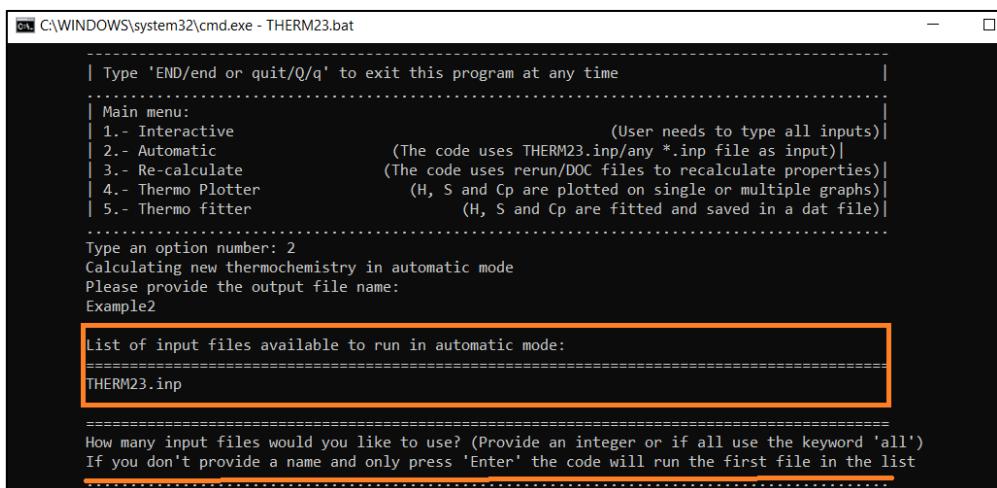


Figure 39. Input files (.INP) available in the main directory of THERM23.

SpeciesName	Formula	NoGAVs	GAVsName	Quantity	Symmetry	Rotors	IfRad	ParentMol	C	H	O	N	Linearity
C2H6	C2H6	2	C/C/H3	2	18	1	no	no	2	6	0	0	no
C2H5	C2H5	3	C/C/H3, C2H5	2, 1	6	1	yes	C2H6	2	5	0	0	no

Figure 40. Content of THERM23.inp, which includes 14 different columns required for the calculation of thermodynamic properties.

In this manual for this mode, as an example we will calculate the thermodynamic properties of ethane (C_2H_6) and its radical, ethyl radical (C_2H_5), Fig. 40. The ‘SpeciesName’ column contains any string to name the specie of interest, which should be similar with the species name in the mechanism to be used. In this case, ‘ C_2H_6 ’ and ‘ C_2H_5 ’. The ‘Formula’ column must contain the molecular formula of the listed species, which are ‘ C_2H_6 ’, and ‘ C_2H_5 ’, respectively. The third column is for the number of groups (NoGAVs). In this case ‘2’ and ‘3’ for C_2H_6 and C_2H_5 , respectively. The next column is the name of the groups used for each species (GAVsName). For C_2H_6 , only C/C/H3 and for C_2H_5 there are C/C/H3 and C2H5. After this, number of each group (Quantity) should be mentioned in the next column

Supplementary Material

as shown in Fig. 40 that is 2 and 2,1 for C₂H₆ and C₂H₅, respectively. In the ‘Symmetry’ column, the total symmetry number should be provided, in this ‘case ‘18’ and ‘9’ for C₂H₆ and C₂H₅, respectively. The user needs to provide the number of rotors (Rotors) and whether the species is a radical or not (IfRad). In the case of the ‘IfRad’ column, the user provides a string mentioning either ‘yes’ or ‘no’. If the user types ‘yes’, it is necessary to provide a parent’s molecule name in the next column named as ‘ParentMol’. The number of ‘C’, ‘H’, ‘O’ and ‘N’ should be provided in the next four columns. Finally, the last column called ‘Linearity’ in which the user should provide the information with strings either ‘yes’ or ‘no’.

Once the input files are ready, the user can run them either individually or all together. For example, in Fig. 39 underlined in orange the question ‘How many input files would you like to use?’ indicates that the user should provide the total number and name of those input files as shown in Fig. 41 and highlighted in a blue rectangle.

```
C:\WINDOWS\system32\cmd.exe - THERM23.bat
| Type 'END/end or quit/Q/q' to exit this program at any time |
| Main menu:
| 1.- Interactive
| 2.- Automatic
| 3.- Re-calculate
| 4.- Thermo Plotter
| 5.- Thermo fitter
| (User needs to type all inputs)
| (The code uses THERM23.inp/any *.inp file as input)
| (The code uses rerun/DOC files to recalculate properties)
| (H, S and Cp are plotted on single or multiple graphs)
| (H, S and Cp are fitted and saved in a dat file)

Type an option number: 2
Calculating new thermochemistry in automatic mode
Please provide the output file name:
Example2

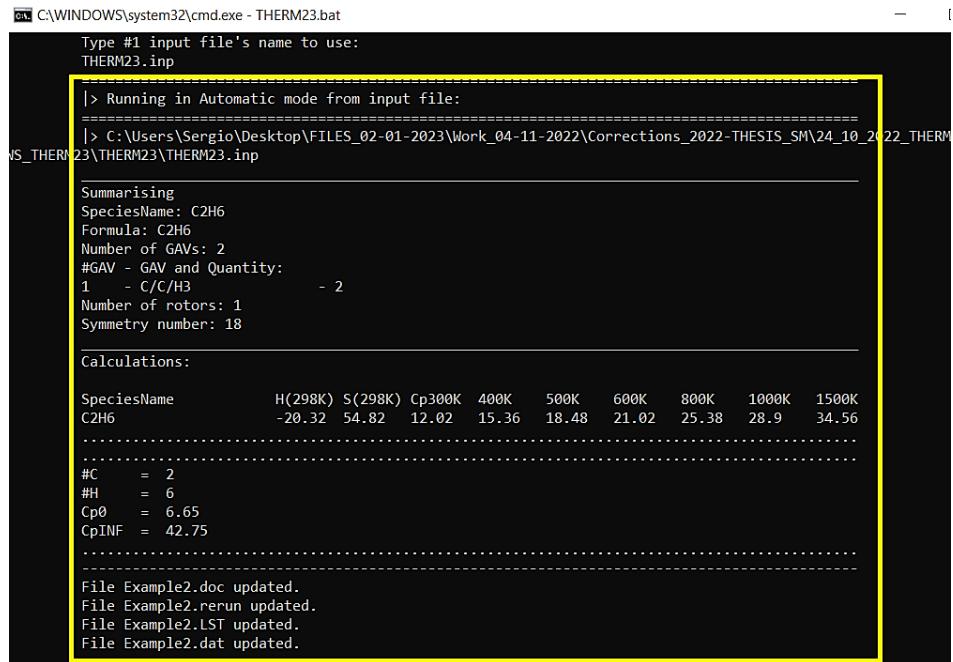
List of input files available to run in automatic mode:
=====
THERM23.inp
=====

How many input files would you like to use? (Provide an integer or if all use the keyword 'all')
If you don't provide a name and only press 'Enter' the code will run the first file in the list
1
Type #1 input file's name to use:
THERM23.inp
```

Figure 41. Number of input files to run and the input file name.

Figure 42 shows the output of the results, highlighted by a yellow rectangle, and the thermochemistry calculations generated using THERM23. As presented in Fig. 33 of Section 2.1.1, both figures show identical results for both different methods and the same species, ‘Interactive’ and ‘Automatic’ modes. The results for ethyl radical (C₂H₅) are highlighted by an orange rectangle. A synopsis of the parameters provided, and the thermochemistry calculations generated by THERM23, is shown in Fig. 43. Figures 42 and 43 show the instruction when the code updates the information of the output files with the new parameters calculated by every species, after the yellow and orange rectangles respectively.

Supplementary Material



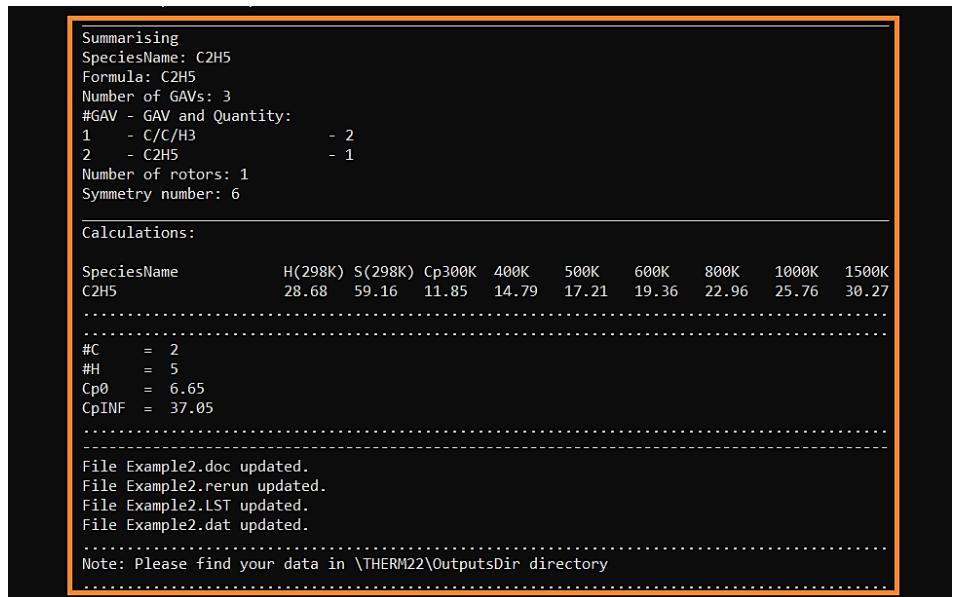
```
C:\WINDOWS\system32\cmd.exe - THERM23.bat
Type #1 input file's name to use:
THERM23.inp
|> Running in Automatic mode from input file:
=====
|> C:\Users\Sergio\Desktop\FILES_02-01-2023\Work_04-11-2022\Corrections_2022-THERSIS_SM\24_10_2022_THERM23\THERM23\THERM23.inp

Summarising
SpeciesName: C2H6
Formula: C2H6
Number of GAVs: 2
#GAV - GAV and Quantity:
1 - C/C/H3 - 2
Number of rotors: 1
Symmetry number: 18

Calculations:

SpeciesName      H(298K)  S(298K)  Cp300K  400K   500K   600K   800K   1000K  1500K
C2H6            -20.32   54.82    12.02   15.36   18.48   21.02   25.38   28.9    34.56
.....
#C      = 2
#H      = 6
Cp0     = 6.65
CpINF   = 42.75
.....
File Example2.doc updated.
File Example2.rerun updated.
File Example2.LST updated.
File Example2.dat updated.
```

Figure 42. Resume of the parameters provided for ethane (C_2H_6) and the proper calculations generated by THERM23.



```
Summarising
SpeciesName: C2H5
Formula: C2H5
Number of GAVs: 3
#GAV - GAV and Quantity:
1 - C/C/H3 - 2
2 - C2H5 - 1
Number of rotors: 1
Symmetry number: 6

Calculations:

SpeciesName      H(298K)  S(298K)  Cp300K  400K   500K   600K   800K   1000K  1500K
C2H5            28.68    59.16    11.85   14.79   17.21   19.36   22.96   25.76   30.27
.....
#C      = 2
#H      = 5
Cp0     = 6.65
CpINF   = 37.05
.....
File Example2.doc updated.
File Example2.rerun updated.
File Example2.LST updated.
File Example2.dat updated.
.....
Note: Please find your data in \THERM22\OutputsDir directory
```

Figure 43. Resume of the parameters provided for ethyl radical (C_2H_5) and the proper calculations generated by THERM23.

2.2.3 Re-calculation of thermodynamic properties

The third way to run THERM23 to calculate thermodynamic properties is in ‘Re-calculate’ mode. To activate this mode the user must type ‘3’ in the main menu screen highlighted in blue in Fig. 44. To calculate thermochemistry in this mode, the user needs to provide an input file, that has been previously created/calculated in either ‘Interactive’ or ‘Automatic’ mode to read the basic information needed. There are two types of input files

Supplementary Material

accepted by the ‘Re-calculate’ mode, ‘.rerun’ and ‘.doc’ extension files. These files, if available in the ‘OutputsDir’ directory, are listed in the software screen as highlighted in red in Fig. 45. Here we can see there are two files with extension ‘rerun’, and two files with extension ‘doc’, Example1.rerun, Example2.rerun, Example1.doc, and Example2.doc, respectively.

```
| Type 'END/end or quit/Q/q' to exit this program at any time |
| Main menu: |
| 1.- Interactive (User needs to type all inputs) |
| 2.- Automatic (The code uses THERM22_V2.0.inp/any *.inp file as input) |
| 3.- Re-calculate (The code uses rerun/DOC files to recalculate properties) |
| 4.- Thermo Plotter (H, S and Cp are plotted on single or multiple graphs) |
| 5.- Thermo fitter (H, S and Cp are fitted and saved in a dat file) |
|
Type an option number: 3
Re-calculating thermochemistry

List of files available to re-calculate thermochemistry:
=====
Example1.rerun
Example2.rerun
=====
Example1.doc
Example2.doc
~$HENOLS.doc
=====
```

Figure 44. Activation of ‘Re-calculate’ mode of THERM23.

```
| Type 'END/end or quit/Q/q' to exit this program at any time |
| Main menu: |
| 1.- Interactive (User needs to type all inputs) |
| 2.- Automatic (The code uses THERM22_V2.0.inp/any *.inp file as input) |
| 3.- Re-calculate (The code uses rerun/DOC files to recalculate properties) |
| 4.- Thermo Plotter (H, S and Cp are plotted on single or multiple graphs) |
| 5.- Thermo fitter (H, S and Cp are fitted and saved in a dat file) |
|
Type an option number: 3
Re-calculating thermochemistry

List of files available to re-calculate thermochemistry:
=====
Example1.rerun
Example2.rerun
=====
Example1.doc
Example2.doc
~$HENOLS.doc
=====
```

Figure 45. List of input files for the ‘Re-calculate’ mode of THERM23.

To illustrate how this mode works, we recalculate the data generated by the ‘Interactive’ mode by typing on the screen the file name ‘Example1.rerun’, and pressing ‘Enter’ highlighted in green in Fig. 46. Consequently, the code will run in a similar way to the ‘Automatic’ mode, see Fig. 47, and generates the corresponding output files as shown in Fig. 48.

Supplementary Material

```
C:\WINDOWS\system32\cmd.exe - THERM23.bat - THERM23.bat
| Type 'END/end or quit/Q/q' to exit this program at any time |
| Main menu:                                                 (User needs to type all inputs)|
| 1.- Interactive                                         (The code uses THERM23.inp/any *.inp file as input)|
| 2.- Automatic                                           (The code uses rerun/DOC files to recalculate properties)|
| 3.- Re-calculate                                         (H, S and Cp are plotted on single or multiple graphs)|
| 4.- Thermo Plotter                                       (H, S and Cp are fitted and saved in a dat file)|
| 5.- Thermo fitter                                        

Type an option number: 3
Re-calculating thermochemistry

List of files available to re-calculate thermochemistry:
=====
Example1.rerun
Example2.rerun
=====
Example1.doc
Example2.doc
~$HENOLS.doc
=====
Choose one and insert its name as requested. However, if your file is not listed
please copy and paste it into '\THERM22\OutputsDir\' directory
file should have an extension '.rerun' or '.doc', i.e. Test_25_12_2021.rerun

Please provide the file name: Example1.rerun
```

Figure 46. Recalculate the data generated by the ‘Interactive’ mode typing on the screen the file name ‘Example1.rerun’, and press ‘Enter’.

```
Re-calculating thermochemistry

List of files available to re-calculate thermochemistry:
=====
Example1.rerun
Example2.rerun
=====
Example1.doc
Example2.doc
~$HENOLS.doc
=====
Choose one and insert its name as requested. However, if your file is not listed
please copy and paste it into '\THERM22\OutputsDir\' directory
file should have an extension '.rerun' or '.doc', i.e. Test_25_12_2021.rerun

Please provide the file name: Example1.rerun
Reading... Example1.rerun
=====
> Running in Automatic mode.

=====
SpecieName: C2H6
Formula: C2H6
Number of GAVs: 2
GAVs and Quantity:
C/C/H3 2
Number of rotors: 1
Symmetry number: 18

Calculations:

Formula H(298K) S(298K) Cp300K Cp400K Cp500K Cp600K Cp800K Cp1000K Cp1500K date
C2H6 -20.32 54.82 12.02 15.36 18.48 21.02 25.38 28.9 34.56 25/10/2022
```

Figure 47. Running the ‘Re-calculate’ mode using the file ‘Example1.rerun’. This file extension is unique from THERM23, and it runs in a similar way as the ‘Automatic’ mode already described.

Supplementary Material

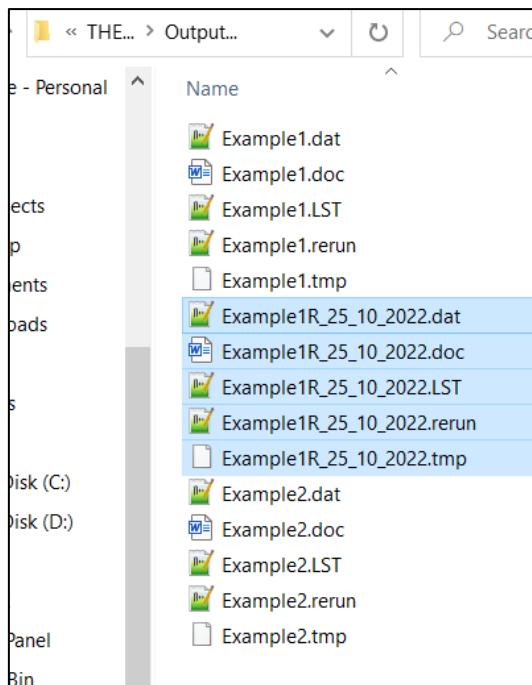


Figure 48. Output files generated by the 'Re-calculate' mode using the file 'Example1.rerun' as input file.

These output files are named adding 'R_' and the date when the calculations were performed. Thus, if the user executes again the code, and choose 'Option 3', a list with the corresponding files with extensions 'rerun' and 'doc' are shown. In this list, there should be two new files, 'Example1R_25_10_2022.rerun' and 'Example1R_25_10_2022.doc', respectively. These are the output files corresponding to the code running in 'Re-calculate' mode using the file named 'Example1.rerun', as shown in Fig. 49.

```
Main menu:
1.- Interactive
2.- Automatic
3.- Re-calculate
4.- Thermo Plotter
5.- Thermo fitter
.....  

Type an option number: 3
Re-calculating thermochemistry:  

List of files available to re-calculate thermochemistry:
=====
Example1.rerun
Example1R_25_10_2022.rerun
Example2.rerun
=====
Example1.doc
Example1R_25_10_2022.doc
Example2.doc
~$HENOLS.doc  

Choose one and insert its name as requested. However, if your file is not listed
please copy and paste it into '\THERM22\OutputsDir\' directory
file should have an extension '.rerun' or '.doc', i.e. Test_25_12_2021.rerun
```

Figure 49. Updated list of available input files for the 'Re-calculate' mode of THERM23 showed before in Fig. 46.

Supplementary Material

Similarly, using the procedure described previously, the user can run the ‘Re-calculate’ mode using the file ‘Example2.doc’. Once the code has finished, in the directory ‘OutputsDir’ the user will see that the list of files has increased as shown in Fig. 50. The new files added are those with names ‘Example2R_25_10_2022’.

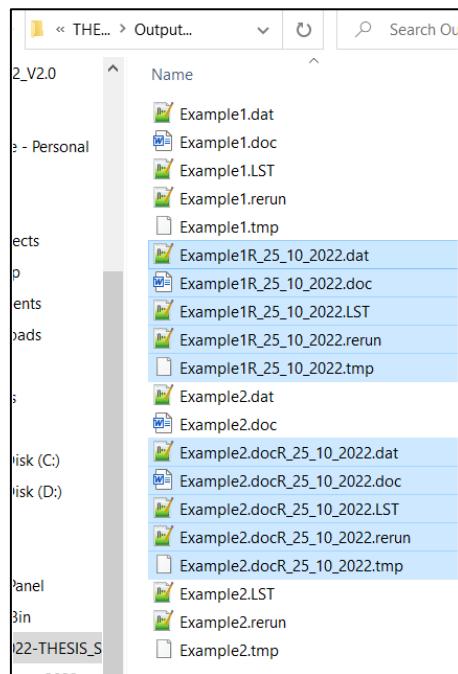


Figure 50. Output files generated by the 'Re-calculate' mode using the file 'Example2.doc' as input file.

Finally, for this Section, if the user runs the ‘Re-calculate’ mode using files with extensions either ‘rerun’ or ‘doc’ as input files, the content of the output files must have identical results, as can be seen in Fig. 51.

Supplementary Material

<pre> SPECIES C2H6 Thermo estimation for molecule C2H6 UNITS: K,kcal - NONLINEAR SPECIE GROUPS 1 Gr # Group ID Quantity 1 - C/C/H3 - 2 Hf S CP300 CP400 CP500 CP600 CP800 CP1000 CP1500 -20.32 54.82 12.02 15.36 18.48 21.02 25.38 28.90 34.56 CPINF = 42.75 Number of rotors:1 Symmetry: 18 Creation date: 26/10/2022 ENDSPECIES SPECIES C2H5 Thermo estimation for radical C2H5 RADICAL BASED UPON PARENT C2H6 PARENT FORMULA PARENT SYMMETRY UNITS: K,kcal - NONLINEAR SPECIE GROUPS 2 Gr # Group ID Quantity 1 - C/C/H3 - 2 2 - C2H5 - 1 Hf S CP300 CP400 CP500 CP600 CP800 CP1000 CP1500 28.68 59.16 11.85 14.79 17.21 19.36 22.96 25.76 30.27 CPINF = 37.05 Number of rotors:1 Symmetry: 6 R ln(2) has been added to S to account for unpaired electron BOND DISS Creation date: 26/10/2022 ENDSPECIES </pre>	<pre> SPECIES C2H6 Thermo estimation for molecule C2H6 UNITS: K,kcal - NONLINEAR SPECIE GROUPS 1 Gr # Group ID Quantity 1 - C/C/H3 - 2 Hf S CP300 CP400 CP500 CP600 CP800 CP1000 CP1500 -20.32 54.82 12.02 15.36 18.48 21.02 25.38 28.90 34.56 CPINF = 42.75 Number of rotors:1 Symmetry: 18 Creation date: 26/10/2022 ENDSPECIES SPECIES C2H5 Thermo estimation for radical C2H5 RADICAL BASED UPON PARENT C2H6 PARENT FORMULA PARENT SYMMETRY UNITS: K,kcal - NONLINEAR SPECIE GROUPS 2 Gr # Group ID Quantity 1 - C/C/H3 - 2 2 - C2H5 - 1 Hf S CP300 CP400 CP500 CP600 CP800 CP1000 CP1500 28.68 59.16 11.85 14.79 17.21 19.36 22.96 25.76 30.27 CPINF = 37.05 Number of rotors:1 Symmetry: 6 R ln(2) has been added to S to account for unpaired electron BOND DISS Creation date: 26/10/2022 ENDSPECIES </pre>
--	--

Figure 51. Content of the ‘Example1R_25_10_2022.doc’ in the left side, and the content of the file ‘Example2R_25_10_2022.doc’ in the right side. Both show identical results for both different ways to run the ‘Re-calculate’ mode.

2.2.4 Output files content and format

The content of the output files is presented in Figs. 52–54, with these files having extensions ‘.dat’, ‘.doc’, and ‘.lst’, respectively. The code generates these three types of files regardless of what mode it is used to run. Figure 52 shows the content of the output file called ‘Example1.dat’. The first line contains the word ‘THERMO’. The second line provides the range of temperatures valid for the file content, in this case ‘300.00 1000.00 5000.00’, highlighted in green in Fig 52. The next line is split into ten parts, each with a purpose and specific amount of space assigned. For example, the name of the species can be a maximum of 24 characters in length. The next 20 blank spaces are for the atom types and number of atoms, being ‘C 2H 6O 0’ ($2 \times$ carbon atoms, $6 \times$ hydrogen atoms and $0 \times$ oxygen atoms). Followed by a ‘G’ which represent the phase, gaseous in this case. The final part of the row indicates the temperature range for the NASA polynomials, and the number of the row for the specific species, ‘ 300.000 5000.000 983.000 1’. Anything after this point, ‘!’, is considered as comment by the software. Furthermore, this file also contains fourteen NASA polynomial coefficients allocated in the next three rows. These polynomials are valid from low-temperature (LT) \sim 300 K to high-temperature (HT) of \sim 5000 K. To achieve this, the polynomials are fitted using the Wilhoit functions in the range of temperature mentioned. The breaking point (BP) which separates LT from HT in the fits as shown in Fig. 52 by an orange box. These polynomials occupy the lines numbered as 2, 3, and 4 as mentioned in the last column.

Supplementary Material

Example1.dat										
1	THERMO									
2	300.00	1000.00	5000.00							
3	C2H6	C	2H	6	0	0G	300.000	5000.000	983.000	1 !
4	3.91337458E+00	1.50671576E-02	-5.28000268E-06	8.78580006E-10	-5.65578255E-14		2			
5	-1.24866927E+04	-2.12986410E-02	-9.37052998E-01	2.88181620E-02	-2.15147540E-05		3			
6	1.05879823E-08	-2.41480728E-12	-1.10557388E+04	2.52038805E+01			4			
7	C2H5	C	2H	5	0	0G	300.000	5000.000	983.000	1 !
8	4.56727055E+00	1.18845080E-02	-4.14861918E-06	6.93210885E-10	-4.48477375E-14		2			
9	1.21460167E+04	-5.79608278E-01	-7.31476956E-02	2.54575350E-02	-2.07385720E-05		3			
10	1.08717908E-08	-2.55039074E-12	1.34865033E+04	2.34308753E+01			4			
11	END									

Figure 52. Content of the 'Example1.dat' output file.

Figure 53 shows the out .doc file, 'Example1.doc' for our example. This file contains a full description of the input file information and the thermochemistry calculations. For C₂H₆, the following information is printed in this output file: the species name (C₂H₆); units for the estimations (K, Kcal); if the species is a linear molecule or not (nonlinear species); and a resume of # of groups used for the estimation with GAV's name and # of times that GAV is used (1 – C/C/H3 – 2). Thereafter, the thermochemistry calculations are shown with the numeric estimations for H_f, S, Cp 300 – 1500. Next, the heat capacity at infinite temperature is shown as CPINF, followed by the # of rotors, symmetry number, and finally, the creation date.

Supplementary Material

```

1  SPECIES
2  C2H6
3 Thermo estimation for molecule
4  C2H6
5 UNITS: K,kcal - NONLINEAR SPECIE
6 GROUPS 1
7   Gr # Group ID   Quantity
8     1 - C/C/H3 - 2
9   Hf   S   CP300 CP400 CP500 CP600 CP800 CP1000 CP1500
10  -20.32 54.82 12.02 15.36 18.48 21.02 25.38 28.90 34.56
11      CPINF = 42.75
12 Number of rotors: 1
13 Symmetry: 18
14 Creation date: 25/10/2022
15 ENDSPECIES
16
17
18 SPECIES
19 C2H5
20 Thermo estimation for radical
21 C2H5
22 RADICAL BASED UPON PARENT C2H6
23 PARENT FORMULA
24 PARENT SYMMETRY
25 UNITS: K,kcal - NONLINEAR SPECIE
26 GROUPS 2
27   Gr # Group ID   Quantity
28     1 - C/C/H3 - 2
29     2 - C2H5 - 1
30   Hf   S   CP300 CP400 CP500 CP600 CP800 CP1000 CP1500
31  28.68 59.16 11.85 14.79 17.21 19.36 22.96 25.76 30.27
32      CPINF = 37.05
33 Number of rotors: 1
34 Symmetry: 6
35 R ln(2) has been added to S to account
36      for unpaired electron
37 BOND DISS
38 Creation date: 25/10/2022
39 ENDSPECIES
40

```

Figure 53. ‘Example1.doc’ output file content.

Figure 54 shows the content of the ‘Example1.LST’ output file. This is the most important file, and the thermochemistry of the species is listed here. This file contains the species name, the enthalpy of formation, entropy, and the heat capacities in the temperature range 300 – 5000 K. In addition to the thermochemical information, it also stores the data regarding the number of different atoms and the number of rotors of the calculated species and finally the creation date of this file. It is important to note that the code can generate the NASA polynomials file (‘.dat’) from this .lst file and plot the results from this file. It is easy to see that the main reason for reporting the number of atoms and rotors in this file is that a proper limit is needed in the calculation of C_p s through CPINF shown in the previous ‘.doc’ file.

1	Units: K,calories	SPECIES	Hf	S	CP	300	400	500	600	800	1000	1500	DATE	ELEMENTS	C	H	O	N	Rotor
2																			
3		C2H6	-20.32	54.82		12.02	15.36	18.48	21.02	25.38	28.90	34.56				2	6	0	0
4		C2H5	28.68	59.16		11.85	14.79	17.21	19.36	22.96	25.76	30.27				2	5	0	0
5																			1

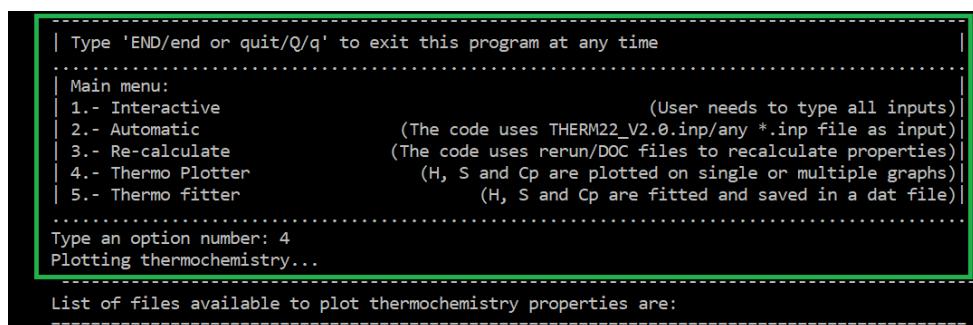
Figure 54. ‘Example1.doc’ output file content.

Supplementary Material

To finalise this section, it is important to note that, from these three different modes ('Interactive', 'Automatic', and 'Re-caulculate'), we have estimated the thermochemistry properties of ethane (C_2H_6) and ethyl radical (C_2H_5). All of the files listed in Fig. 50 must contain the same information and numerical estimations with the same uncertainties.

2.2.5 Thermo plotter

The 'Thermo plotter' mode is a graphical tool to easily visually interpret the NASA polynomial coefficients provided by the '.dat' file(s). In THERM23 there are two ways to plot any NASA polynomial data, generated using either the THERM23 or THERM codes. The user needs to access this mode by typing '4' in the welcome screen of THERM23, as shown in Fig. 55. This will activate the 'Thermo plotter' mode, and the legend 'Plotting thermochemistry...' will appear below the 'Type an option number:'.



The screenshot shows a terminal window with a green border. At the top, it says: 'Type 'END/end or quit/Q/q' to exit this program at any time'. Below that is a 'Main menu' with options: 1.- Interactive, 2.- Automatic, 3.- Re-calculate, 4.- Thermo Plotter, and 5.- Thermo fitter. To the right of the menu, there is explanatory text: '(User needs to type all inputs)', '(The code uses THERM22_V2.0.inp/any *.inp file as input)', '(The code uses rerun/DOC files to recalculate properties)', '(H, S and Cp are plotted on single or multiple graphs)', and '(H, S and Cp are fitted and saved in a dat file)'. Below the menu, it says 'Type an option number: 4' and 'Plotting thermochemistry...'. At the bottom, it says 'List of files available to plot thermochemistry properties are:' followed by a list of file names.

Figure 55. Activation of the 'Thermo plotter' mode in THERM23.

Once the user activates the 'Thermo plotter' mode the code will ask 'Would you like single plots or to compare thermo files?'. Here, the user will see two different options. The user needs to type either '1' or '2' for single or to compare thermodynamic parameters between two '.dat' file sources, respectively. For this example, we typed '1' as shown in Fig. 56 inside the yellow rectangle to plot from a single source file. **IMPORTANT NOTE:** at the top of this yellow rectangle shown in Fig. 56 the user can see there is a list of files to choose. It is important that the file intended to be plotted by the user has the '.dat' extension and that the content of the file should have the correct NASA polynomial format. Consequently, the software will ask for the 'Max temperature (K) to plot (integer, i.e. 3000 or 5000)'. This temperature value is a limit to perform the extrapolation for the plots with the corresponding properties such as heat capacity (C_p), enthalpy (H), and entropy (S). In this case, the maximum temperature recommended is 5000 K.

Supplementary Material

```
Type an option number: 4
Plotting thermochemistry...
List of files available to plot thermochemistry properties are:
=====
Example1.dat
Example1R_25_10_2022.dat
Example2.dat
Example2.docR_25_10_2022.dat
=====
Choose one and insert its name as requested. However, if your file is not listed
please copy and paste it into '\THERM22\OutputsDir\File2Plot.dat' directory
file should have an extension 'dat' and it must have 2 set of NASA polynomials format
-----
Would you like single plots or to compare thermo files?
For single plots type: 1           (or just press the 'Enter' key)
To compare plots type: 2
1
Max temperature to plot in K: (integer, i.e. 3000 or 5000)
5000
Running plotter in individual mode
-----
How many files would you like to plot?: -
```

Figure 56. The two options available in the 'Thermo plotter' of THERM23.

Furthermore, the code will ask 'How many files would you like to plot?'. In order to proceed, the user should put the number of files and the names of those files as shown in Fig. 58. However, the user can plot more than one file at the same time only by typing the number of desired files. For this example, we will use the file 'Example1.dat' that we have generated in the 'Interactive' mode previously. Currently, there are four '.dat' files in the 'OutputsDir' directory as can be seen in Fig. 57 inside the orange rectangle. Note that the user can plot the data from all of these four files.

```
Type an option number: 4
Plotting thermochemistry...
List of files available to plot thermochemistry properties are:
=====
Example1.dat
Example1R_25_10_2022.dat
Example2.dat
Example2.docR_25_10_2022.dat
=====
Choose one and insert its name as requested. However, if your file is not listed
please copy and paste it into '\THERM22\OutputsDir\File2Plot.dat' directory
file should have an extension 'dat' and it must have 2 set of NASA polynomials format
-----
Would you like single plots or to compare thermo files?
For single plots type: 1           (or just press the 'Enter' key)
To compare plots type: 2
1
Max temperature to plot in K: (integer, i.e. 3000 or 5000)
5000
Running plotter in individual mode
-----
How many files would you like to plot?: -
```

Figure 57. Number of files to use to generate graphs in the 'Thermo plotter' mode.

Figure 58, shows that the name of the file to be used 'Example1.dat' (highlighted within the pink rectangle), which is followed in the next line by the sentence 'The next is a list of

Supplementary Material

species found in the file'. This list may be very large depending on the size of the '.dat' file. However, for our example, this file only contains two species 'C2H5 and C2H6'.

```
.....  
Would you like single plots or to compare thermo files?  
For single plots type: 1 (or just press the 'Enter' key)  
To compare plots type: 2  
1  
Max temperature to plot in K: (integer, i.e. 3000 or 5000)  
5000  
.....  
Running plotter in individual mode  
  
How many files would you like to plot?: 1  
Please provide the file name(s)  
Please give file's name #1: Example1.dat  
Plotting:  
  
Example1.dat  
  
The next is a list of species found in the file  
C2H5 C2H6  
How many species do you want to plot? (Type 'all' to plot them all)
```

Figure 58. Number and name of file(s) to be used in the generation of graphs.

Here we have two options, either type a number such as '1' or '2' or type the keyword 'all'. If a user provides a different answer an error screen will appear, and the code will prompt for a proper answer, as shown in Fig. 59.

```
.....  
Running plotter in individual mode  
.....  
How many files would you like to plot?: 1  
Please provide the file name(s)  
Please give file's name #1: Example1.dat  
Plotting:  
  
Example1.dat  
  
The next is a list of species found in the file  
C2H6 C2H5  
How many species do you want to plot? (Type 'all' to plot them all)  
0  
0 species provided, no plots generated!.  
Please type a number or type 'all' to plot them all.  
If you press 'Enter' again code will end.  
-
```

Figure 59. Example of error thrown if the option typed is '0' species to plot in the 'Thermo plotter' mode.

After we have typed 'all' in our example, as shown in Fig. 60, we will be presented with the next sentence 'Total number of species in Thermo: 2' which summarise how many species are going to be plotted, this means 3×2 number of graphs, one for C_p , one for H , and a third for S .

Supplementary Material

```

The next is a list of species found in the file
C2H5   C2H6
How many species do you want to plot? (Type 'all' to plot them all)
all
Total number of species in Thermo: 2

Plotting NASA polynomials for: C2H5
.....
Polynomials extracted:
Breaking Point (BP) at : 868.0
a1 : -7.31476956E-02
a2 : 2.54575350E-02
a3 : -2.07385720E-05
a4 : 1.08717908E-08
a5 : -2.55039074E-12
a6 : 1.34865033E+04
a7 : 2.34308753E+01
a8 : 4.38980489E+00
a9 : 1.15975275E-02
a10: -3.64957326E-06
a11: 5.72833970E-10
a12: -3.55885783E-14
a13: 1.23223763E+04
a14: 7.14734251E-01
Breaking point...
CP at 868.0 1st set NASA: 23.968
CP at 868.0 2nd set NASA: 23.968
AbsError CPs: 0.000 %
H at 868.0 1st set NASA: 39.315
H at 868.0 2nd set NASA: 39.315
AbsError Hs: 0.000 %
S at 868.0 1st set NASA: 77.955
S at 868.0 2nd set NASA: 77.955
AbsError Ss: 0.000 %

Max. Cp1 value:
23.96797
Max. Cp2 value:
40.78175
C2H5_Example1.dat_CP.png saved.
C2H5_Example1.dat_H.png saved.
C2H5_Example1.dat_S.png saved.

```

Figure 60. Plotting resume of the generation of the graphs, species by species, including the Break Point (BP), and maximum values.

The name of the species, the values of the polynomials, and the BP for C_p , H , and S , are presented within the green rectangle in Fig. 60 for every species that is plotted. For this option, the plots generated will be individual ones and all are placed in a directory named as the ‘.dat’ file used as the source file, as shown in Fig. 61. For example, ‘OutputsDir/Plots/Single/Example1.dat’.

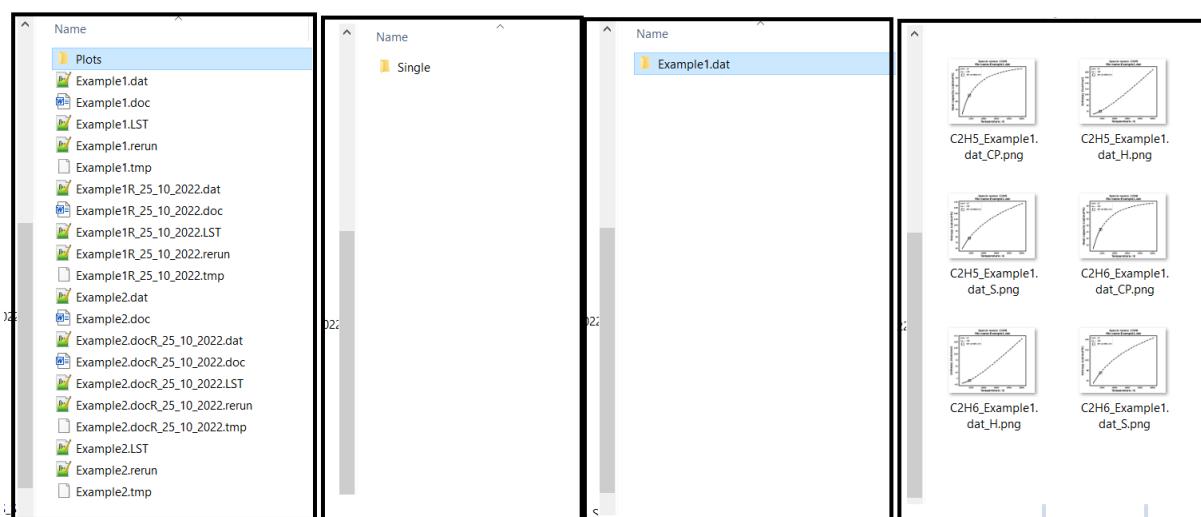


Figure 61. Directories created to storage the graphs generated by 'Thermo plotter' mode.

The graph for C_p is presented in Fig. 62, H in Fig. 63 and S in Fig. 62. The species name, the name of the source file, units of the property, LT, HT, and BP are displayed in the

Supplementary Material

plots. On the x-axis, the temperature (T) is plotted in K. On the y-axis, depending on the property, the units are kcal/mol for H and cal/mol*K for Cp and S .

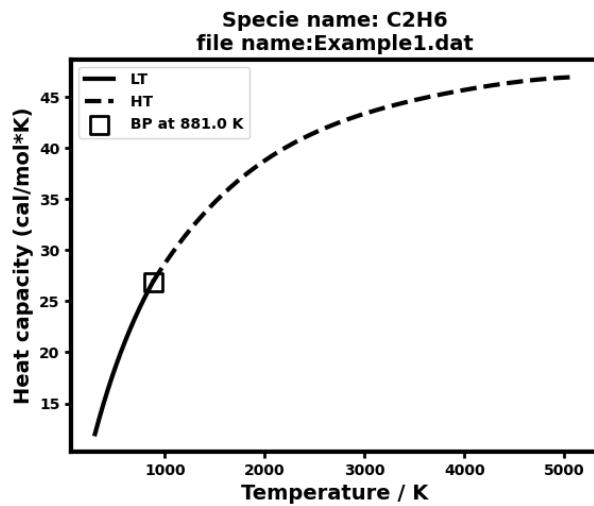


Figure 62. Heat capacity (C_p) in cal/mol*K of ethane (C₂H₆) versus T in K.

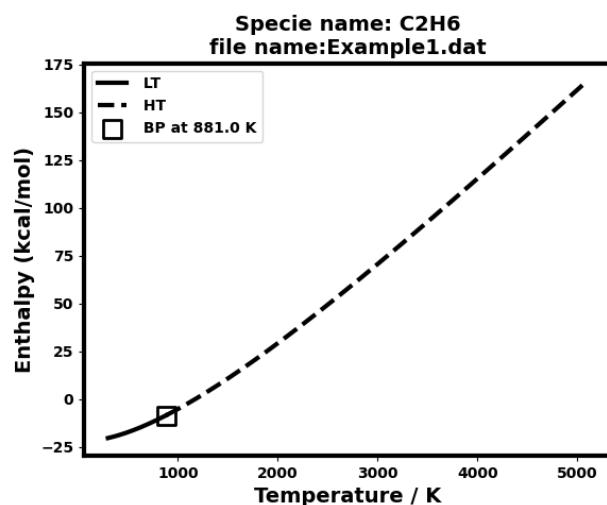


Figure 63. Enthalpy (H) in kcal/mol of ethane (C₂H₆) versus T in K.

Supplementary Material

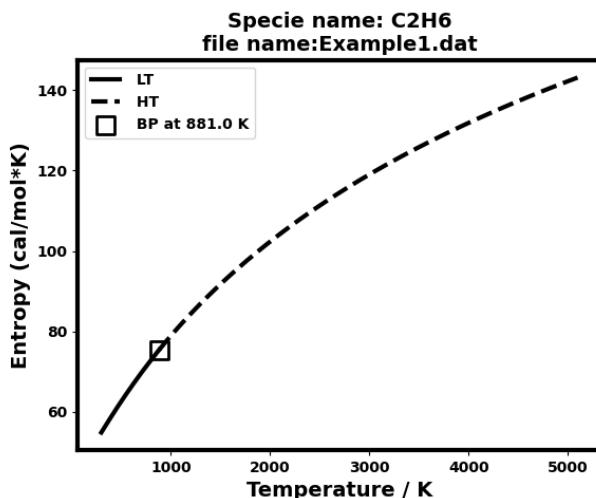


Figure 64. Entropy (S) in cal/mol*K for ethane (C₂H₆) versus T in K.

There is a second option to generate plots using THERM23. This is achieved by typing ‘2’ at the prompt as shown in Fig. 65. Similar to option ‘1’, the screen will ask for a limit temperature to plot the graphs for option ‘2’ as: ‘Max temperature to plot in K: (integer, i.e., 3000 or 5000)’ and we type 5000 here, shown in Fig. 65. Figure 66 shows the list of species in the ‘.dat’ file (inside the orange rectangle) that can be used to plot the comparison of thermochemistry properties. At the bottom of Fig. 66, inside the yellow rectangle, we see that we provided Example1.dat’ and ‘Example2.dat’ as files #1 and #2 . Thus, the thermodynamic properties contained in these two files will be compared. The corresponding graphs for C_p , H , and S can be found in a specific directory, shown in Fig. 67. The individual plots are presented in Figs. 68, 69, and 70, respectively. The species name, the name of the source files, property units, LT, HT, and BP for both trends are displayed in the plots. On the x-axis, the temperature (T) is plotted in K. The y-axis units are cal/mol*K, or kcal/mol, depending on the plotted property. **IMPORTANT NOTE:** comparison plots usually are used to show or see discrepancies in the estimation of C_p , H , and/or S , from different file sources. However, in this example, we have compared two different sources of the same calculations made using THERM23 in different modes, therefore these overlap with one another.

Supplementary Material

```
.....  
Type an option number: 4  
Plotting thermochemistry...  
-----  
List of files available to plot thermochemistry properties are:  
=====  
Example1.dat  
ExampleIR_25_10_2022.dat  
Example2.dat  
Example2.docR_25_10_2022.dat  
=====  
Choose one and insert its name as requested. However, if your file is not listed  
please copy and paste it into '\THERM22\OutputsDir\File2Plot.dat' directory  
file should have an extension 'dat' and it must have 2 set of NASA polynomials format  
-----  
Would you like single plots or to compare thermo files?  
For single plots type: 1 (or just press the 'Enter' key)  
To compare plots type: 2  
2  
Max temperature to plot in K: (integer, i.e. 3000 or 5000)  
5000  
-----  
Running plotter in Multi mode  
-----  
This mode allows the user to compare thermochemistry properties making graphs with them  
from two different thermo files.  
Please type the two file names to compare from the list above:  
Please give file's name #1:
```

Figure 65. Option number '2' of the 'Thermo plotter' of THERM23.

```
.....  
List of files available to plot thermochemistry properties are:  
=====  
Example1.dat  
ExampleIR_25_10_2022.dat  
Example2.dat  
Example2.docR_25_10_2022.dat  
=====  
Choose one and insert its name as requested. However, if your file is not listed  
please copy and paste it into '\THERM22\OutputsDir\File2Plot.dat' directory  
file should have an extension 'dat' and it must have 2 set of NASA polynomials format  
-----  
Would you like single plots or to compare thermo files?  
For single plots type: 1 (or just press the 'Enter' key)  
To compare plots type: 2  
2  
Max temperature to plot in K: (integer, i.e. 3000 or 5000)  
5000  
-----  
Running plotter in Multi mode  
-----  
This mode allows the user to compare thermochemistry properties making graphs with them  
from two different thermo files.  
Please type the two file names to compare from the list above:  
Please give file's name #1: Example1.dat  
Please give file's name #2: Example2.dat  
Plotting:  
-----  
Example1.dat vs Example2.dat  
The next is a list of species found in the file  
C2H5 C2H6  
How many species do you want to plot? (Type 'all' to plot them all)  
all
```

Figure 66. Top inside the orange rectangle, the list of available '.dat' files to plot is shown. Below inside the yellow rectangle, we typed 'Example1.dat', and 'Example2.dat', respectively.

Supplementary Material

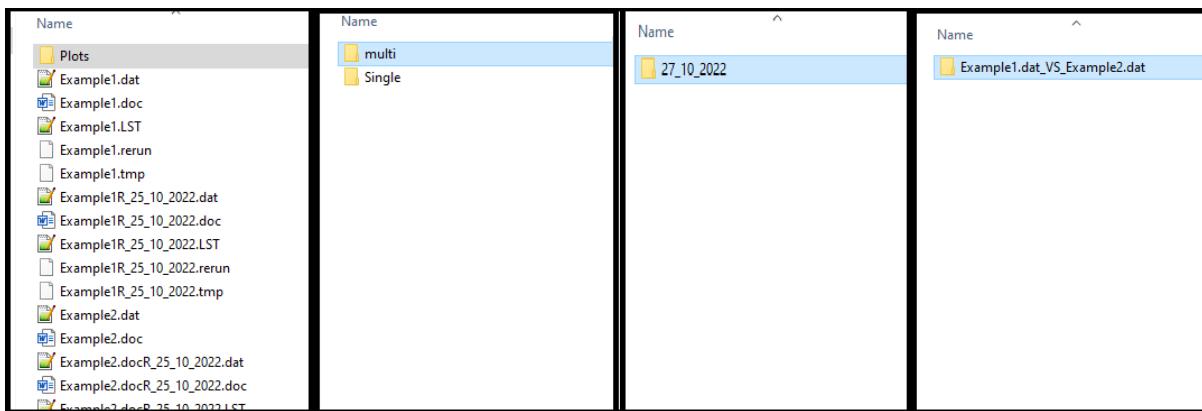


Figure 67. Output directory for the multiple plot (comparison) generated by the option number 2 in the 'Thermo plotter'.

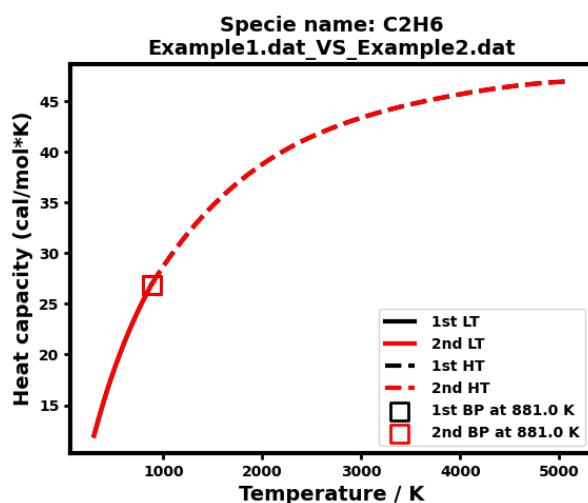


Figure 68. Comparison graph of heat capacity (C_p) of ethane (C₂H₆), from different file sources: 'Example1.dat' and 'Example2.dat'.

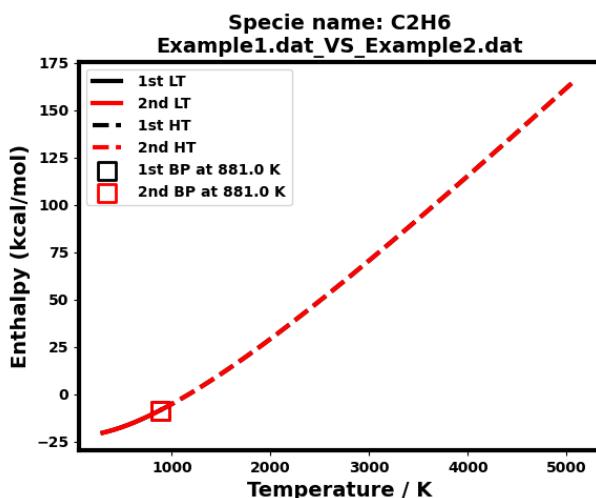


Figure 69. Comparison graph of enthalpy (H) of ethane (C₂H₆), from different file sources: 'Example1.dat' and 'Example2.dat'.

Supplementary Material

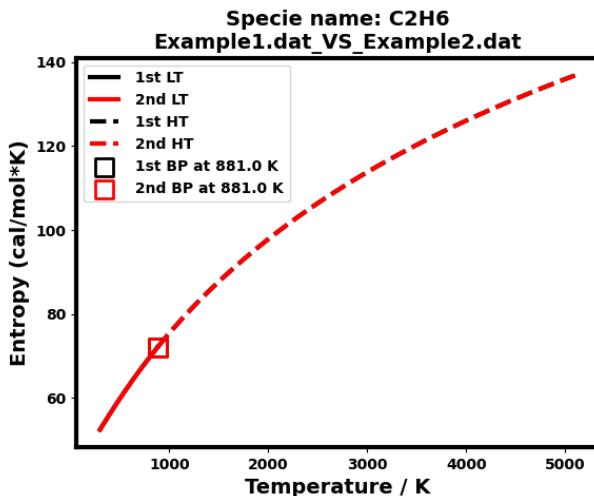


Figure 70. Comparison graph of entropy (S) of ethane (C_2H_6), from different file sources: ‘Example1.dat’ and ‘Example2.dat’.

2.2.6 Thermo fitting

The last mode of THERM23 is the ‘Thermo fitter’. This is a functionality, which allows users to fit any information from ‘.lst’ file to NASA polynomial coefficients, which can be subsequently plotted using the ‘Thermo plotter’ mode already explained in Section 2.1.4. Thus, to activate this mode we must type ‘5’ in the welcome screen, as shown in Fig. 71.

```
| Type 'END/end or quit/Q/q' to exit this program at any time |
| Main menu:                                         (User needs to type all inputs) |
| 1.- Interactive                                     (The code uses THERM22_V2.0.inp/any *.inp file as input)|
| 2.- Automatic                                       (The code uses rerun/DOC files to recalculate properties)|
| 3.- Re-calculate                                     (H, S and Cp are plotted on single or multiple graphs)|
| 4.- Thermo Plotter                                 (H, S and Cp are fitted and saved in a dat file)|
| 5.- Thermo fitter                                    |
-----|
Type an option number: 5

-----|
Running GAVs (CPs) fitter mode

-----|
List of LST files available:
=====
Example1.LST
Example1R_25_10_2022.LST
Example2.docR_25_10_2022.LST
Example2.LST
=====
How many files would you like to plot?:      1
Please give file's name #1:
```

Figure 71. Activation of the ‘Thermo fitter’ mode of THERM23.

Once we have activated this mode, we should provide an integer that represents the number of files we would like to use to generate the NASA polynomials. For this example, we typed ‘1’ and inserted the file’s name ‘Example1.LST’, as shown in Figs. 72 and 73.

Supplementary Material

```

C:\Users\Sergio\Desktop\27_10_2022-therm22\WINDOWS_THERM22\THERM22_V2.0\OutputsDir\Example1.LST - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
Example1.LST
1 Units: K.calories
2 SPECIES Hf S CP 300 400 500 600
3 C2H6 -20.32 54.82 12.02 15.36 18.48 21.02
4 C2H5 28.68 59.16 11.85 14.79 17.21 19.31
5
6
7
8
9
10
11 END

Example1.LST.dat
1 THERMO
2 300.00 1000.00 5000.00
3 C2H6 C 2H 6 0 OG 300.000 5000.000 881.000 1
4 3.97902504E+00 1.42909165E-02-4.40517679E-06 6.95979999E-10-4.27268880E-14 2
5 -1.23926496E+04-4.52122209E-02-9.37052998E-01 2.88181620E-02-2.15147540E-05 3
6 1.05879823E-08-2.41480728E-12-1.10557388E+04 2.52038805E+01 4
7 C2H5 C 2H 5 0 OG 300.000 5000.000 869.000 1
8 4.389804898E+00 1.15975275E-02-3.64957326E-06 5.72833970E-10-3.55885783E-14 2
9 1.23223763E+04 7.14734251E-01-7.31476956E-02 2.54575350E-02-2.07385720E-05 3
10 1.08717900E-08-2.55039074E-12 1.34065033E+04 2.34308753E+01 4
11 END

```

Figure 72. Content of file 'Example1.LST' in the left, and the result after running the 'Thermo fitter' mode in the right side.

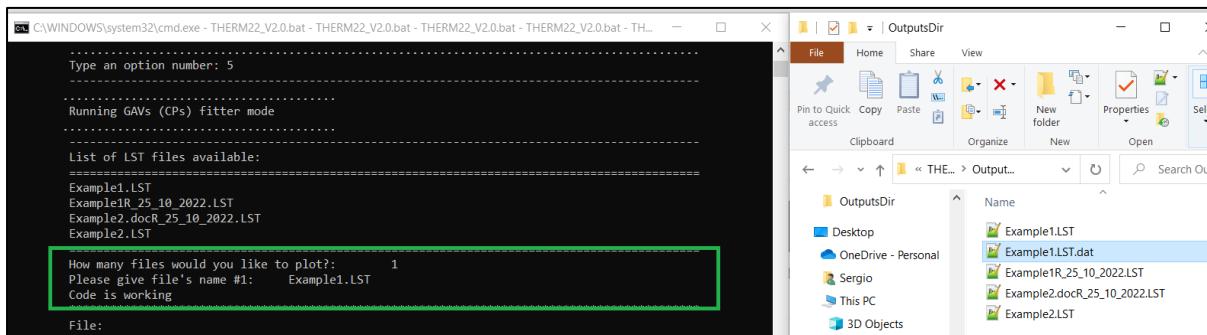


Figure 73. Running the 'Thermo fitter' using the file 'Example1.LST' in the left, and the output directory where the '.dat' file generated is storage at.

In this mode it is possible to use more than one file, and also to provide an alternative output file name. However, the code would name the output file using the same name of the last '.lst' file provided. If we ask for multiple '.lst' files to work with, then the code will put all the data into a unique file, see Fig. 74. To illustrate this, we typed 'all' instead of an integer value. This instruction asks the code to run all of the '.lst' files available in the 'OutputsDir' directory. Also, we only pressed 'Enter' without providing any string to name the output file, then the file 'Example2.LST.dat' will be stored in the 'OutputsDir' directory.

```

Type an option number: 5
-----
Running GAVs (CPs) fitter mode
-----
List of LST files available:
-----
Example1.LST
Example1R_25_10_2022.LST
Example2.docR_25_10_2022.LST
Example2.LST
-----
How many files would you like to plot?:      all
Code is working
Would you like to provide a name for the output 'dat' file?      if not only press 'Enter'
-----
File:
Example1.LST

```

Figure 74. Example of running multiple 'LST' files in one go.

Supplementary Material

References

- [1] G.V. Rossum, F.L. Drake, Python 3 Reference Manual. Scotts Valley, CA: CreateSpace., (2009).
- [2] M.K Ghosh, S.N Elliott, K.P Somers, S.K Klippenstein, H.J. Curran, Group additivity values for the heat of formation of C₂–C₈ alkanes, alkyl hydroperoxides, and their radicals, Combust. Flame doi:[https://doi.org/10.1016/j.combustflame.2022.112492\(2022\) 112492](https://doi.org/10.1016/j.combustflame.2022.112492(2022) 112492).
- [3] S. Martinez, M. Baigmohammadi, V. Patel, S. Panigrahy, A.B. Sahu, S. Nagaraja, A. Ramalingam, K.A. Heufer, A. Pekalski, H.J. Curran, A comprehensive experimental and modeling study of the ignition delay time characteristics of ternary and quaternary blends of methane, ethane, ethylene, and propane over a wide range of temperature, pressure, equivalence ratio, and dilution, Combust. Flame 234 (2021) 111626.