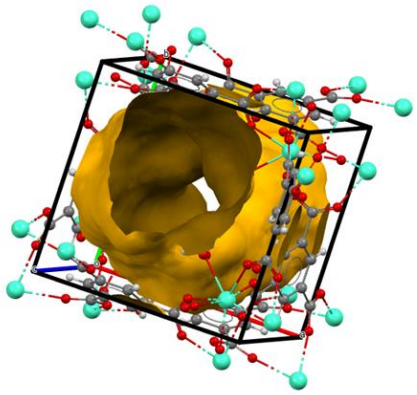
Exploring Metal-Organic Frameworks and Coordination Polymers in Mercury (MER-007)

Developed using
2025.2 CSD Release
(CSD 6.00 + 1 data update)



CCDC
advancing structural science

Table of Contents

Introduction 2
- Learning Outcomes 2
- Pre-required Skills 2
- Materials 2

Example 1. Comparing Coordination Polymers in Mercury 3
- Part 1: Exploring Coordination Geometry 3
- Part 2: Structure Comparison with the Structure Overlay tool 6
- Conclusion 8

Example 2. Examining Porosity in a Metal-Organic Framework 9
- Conclusion 13

Example 3. Batch Calculation of Pore Properties using the CSD Python API 14
- Conclusion 16

Summary 18
- Next Steps 18

Feedback 18

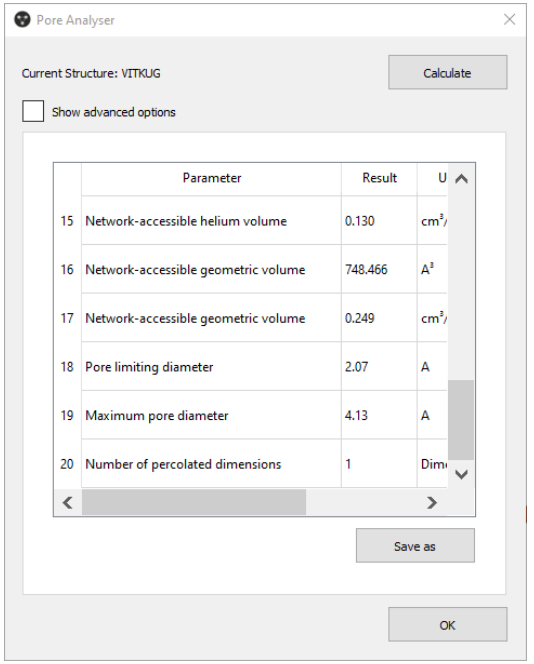
Glossary 19

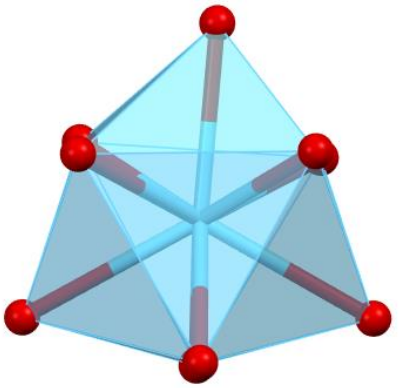
Basics of Mercury Visualization 22

Introduction

Mercury is the visualisation and analysis software of the Cambridge Structural Database (CSD). Mercury not only allows for flexible structure visualisation and structure comparison but provides an interface to a wealth of tools for calculating structure properties. One such tool is the Pore Analyser, which facilitates calculation of pore properties, including geometric and network-accessible volumes, maximum pore and pore-limiting diameter, and network dimensionality. Additionally, Pore Analyser is exposed via the CSD Python API, allowing large scale, programmatic processing of structures.

In this workshop, we will compare closely related coordination polymers using tools in Mercury and calculate the Pore Properties of a metal organic framework (MOF) using Pore Analyser.





Learning Outcomes

After completing this workshop, you will be able to:

- Visualise and control the display of coordination polymers
- Overlay structures in Mercury
- Run a Pore Analyser calculation, including on edited structures.
- Appreciate that calculation of pore properties can be done programmatically using the CSD Python API

This workshop will take approximately ***35 minutes*** to be completed. The words in *Blue Italic* in the text are reported in the Glossary at the end of this handout.

Pre-required Skills

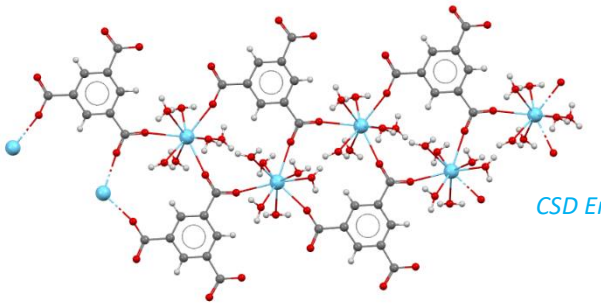
Basic familiarity with Mercury is required. You can find a brief summary guide at the end of the handout. If you wish to try Example 3, knowledge of Python is essential.

Materials

No additional materials are required for this workshop.

Example 1. Comparing Coordination Polymers in Mercury

In this example, we will use the polyhedral display style and the Structure Overlay tool in Mercury to compare related LnBTC (Ln = lanthanoid, BTC = 1,3,5-benzenetricarboxylate) compounds.



CSD Entry KEBSOA, $[La(BTC)(H_2O)_6]_n$

Part 1: Exploring Coordination Geometry

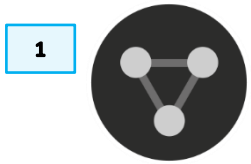
1. Begin by opening Mercury from the Start menu or by clicking on the desktop icon.

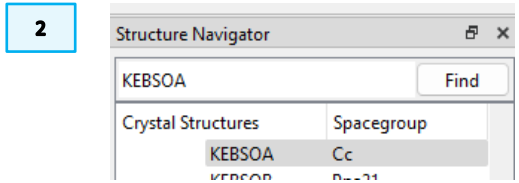
2. We need to select appropriate structures to compare. In the *Structure Navigator*, type refcode "KEBSOA". This is $[La(BTC)(H_2O)_6]_n$.

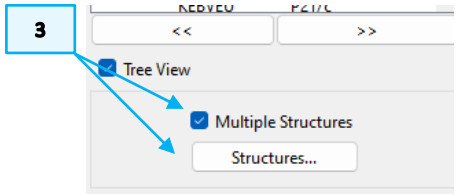
3. At the bottom of the *Structure Navigator*, tick the **Multiple Structures** box, then click on the **Structures…** button to open the *Multiple Structures* dialog box.

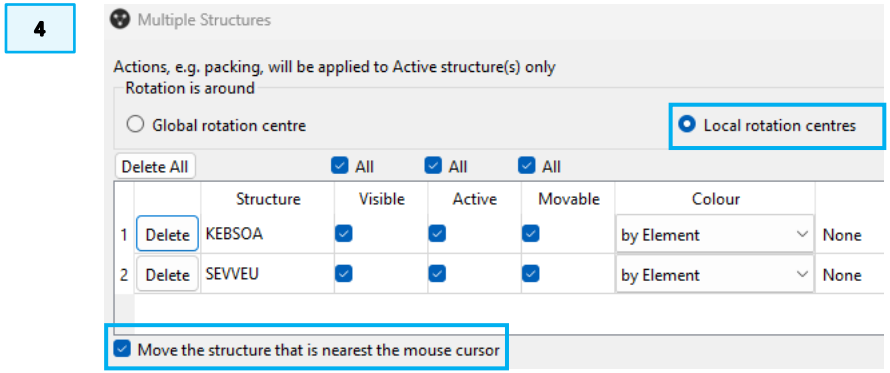
4. In the *Structure Navigator*, type "SEVVEU". The structure will appear in the visualiser area next to (or on top of) KEBSOA. To move the structures around more easily, tick **Move the structure than is nearest the mouse cursor** and **Local rotation centres**. (For a reminder of the mouse operations in Mercury, see here.)

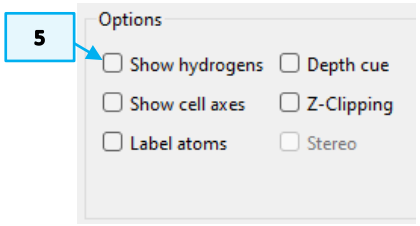
5. SEVVEU is $([Er(BTC)(H_2O)_5]H_2O)_n$ – the metal:BTC and metal:H_2O ratio is the same as in KEBSOA but one of the waters which is coordinated in KEBSOA is displaced from the coordination sphere of the metal in SEVVEU. In the case of SEVVEU, the positions of the hydrogen atoms were not determined. It may help to make the display of both structures consistent by unticking the **Show hydrogens box** in the display options.



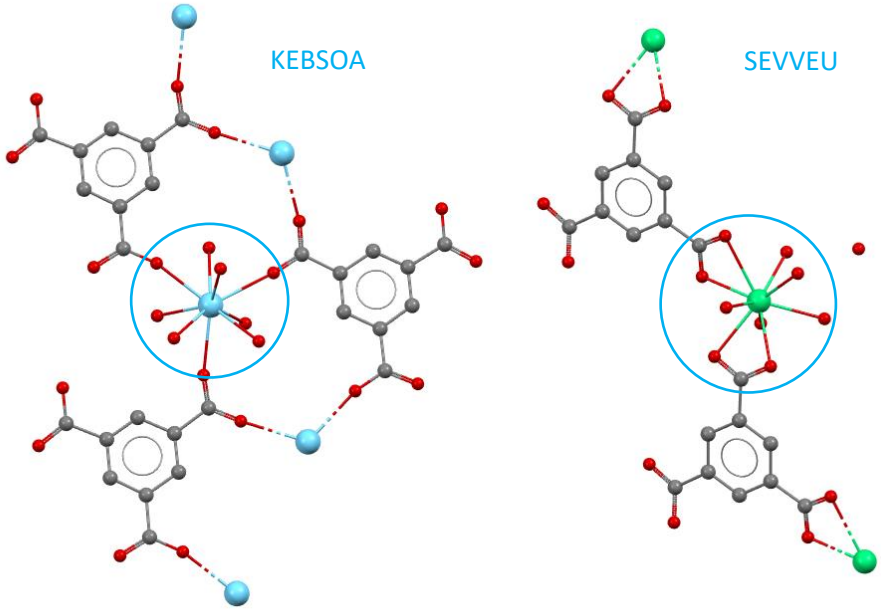




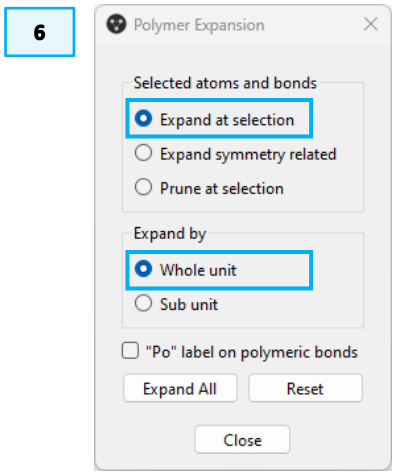


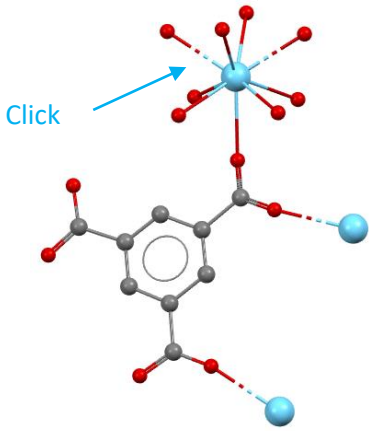


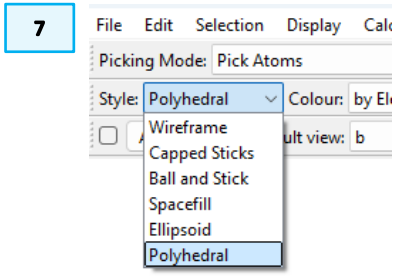
6. To explore the coordination and structure further, we will expand the portion of the structure we are viewing using the Polymer expansion tool. From the top menus go to Edit > Polymer Expansion…. Check that *Selected atoms and bonds* is set to **Expand at selection** and *Expand by* is set to **Whole unit**. Click on each of the two dashed bonds from the metal centres in the two structures. You should see the sections as shown below. Then Close the Polymer Expansion tool dialogue.

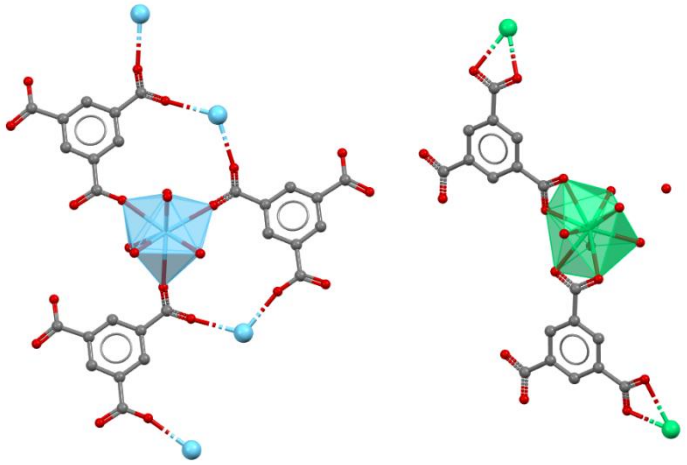


7. From this, you should see that each La centre in KEBSOA is bonded to three different BTC ligands via one of each of the two oxygen atoms of the carboxylate group. the remaining six ligands being water. In contrast, in SEVVEU, each of the Eu centres is bonded to two BTC ligands, each of which coordinates via bidentate carboxylate ligands. The remaining five ligands are water. To highlight the coordination geometry, change the display style to **Polyhedral** from the *Style* drop down menu.









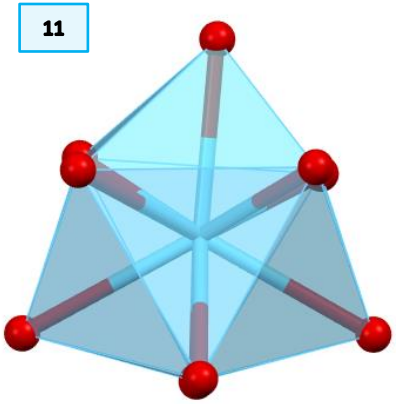
8. To make the coordination geometry even easier to see, first untick the box in the **Active** column of the *Multiple Structures* dialogue box of SEVVEU (this makes only KEBSOA active). Then use the lasso tool from the top toolbar to select the metal and coordinating oxygens.

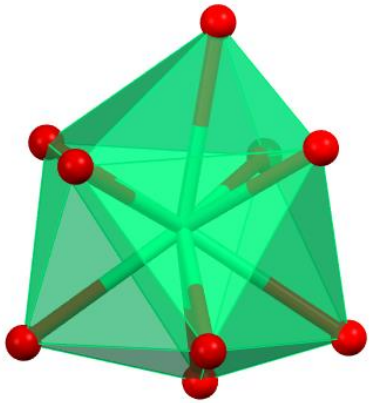
9. Right-click in the visualizer area and from the drop-down menu choose **Selection > Invert Selection**.

10. Right-click in the visualizer area and pick **Show/Hide > Hide** atoms.

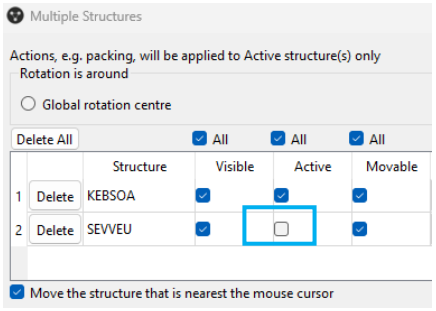
11. We shall want to do the same with SEVVEU. From the *Multiple Structures* dialogue, make KEBSOA inactive and make SEVVEU active again. Lasso the coordinating oxygen atoms and repeat steps 9 and 10. You can see that whilst the [LaO_9] units in KEBSOA have near ideal tricapped trigonal prismatic geometry, the [ErO_9] polyhedral in SEVVEU are considerably more distorted, presumably due to the small bite angle of the bidentate carboxylate ligand.

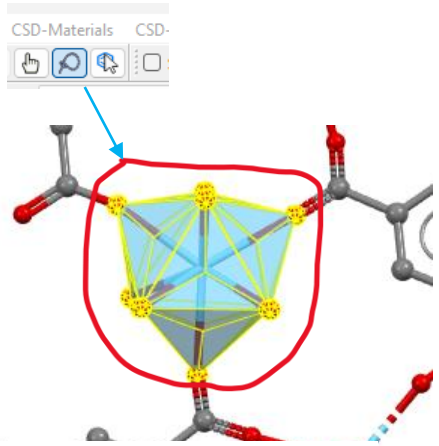
11



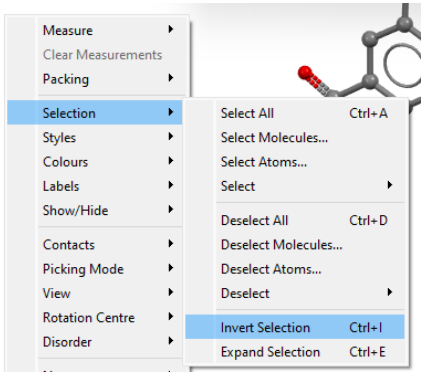


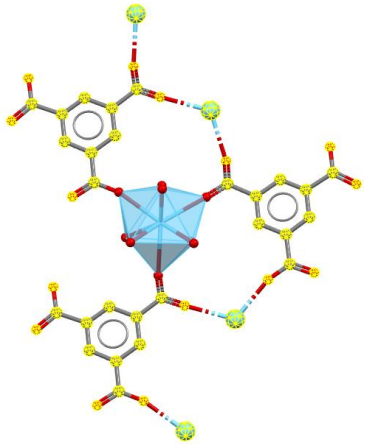
8



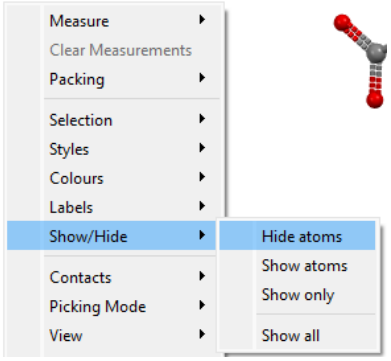


9





10



Part 2: Structure Comparison with the Structure Overlay tool

In Part 1, we could clearly see that the structures were fundamentally different in several ways. What about a scenario where the structures are more closely related.

12. To start, make sure both KEBSOA and SEVVEU are active via the *Multiple Structures* dialogue, then right-click in the visualizer area and select **Show/Hide > Show all**.

13. Set the display style back to Ball and Stick from the top toolbar drop-down menu.

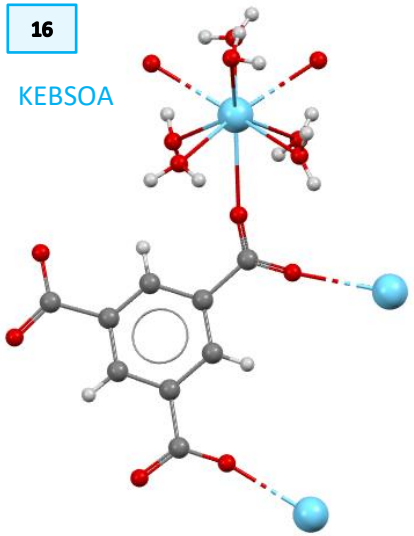
14. From the Multiple Structures dialogue, click **Delete** next to SEVVEU.

15. Click **Reset** in the Display Options area. Note that this will re-enable the display of hydrogens. You can turn off the display of H-atoms again if you prefer.

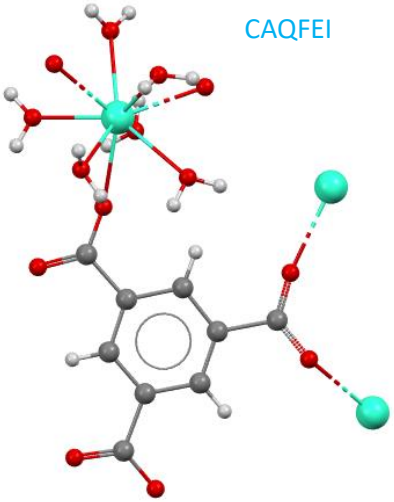
16. In the *Structure Navigator*, type "CAQFEI", which is $([Tb(BTC)(H_2O)_6])_n$. Comparing the repeat units, side-by-side, they look different.

16

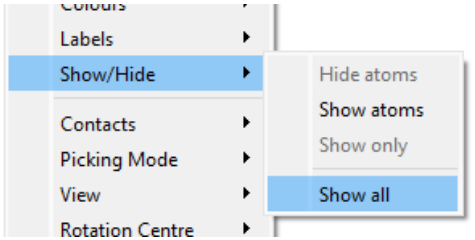
KEBSOA



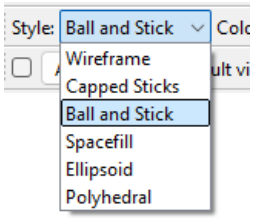
CAQFEI



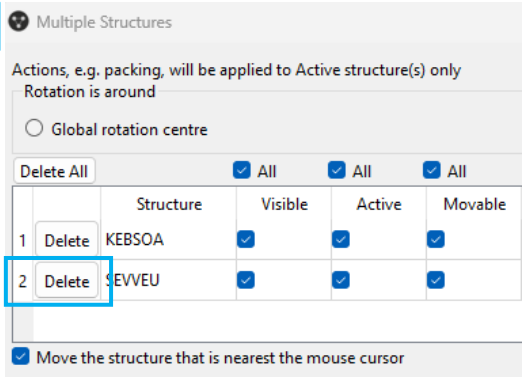
12



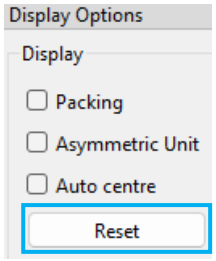
13



14



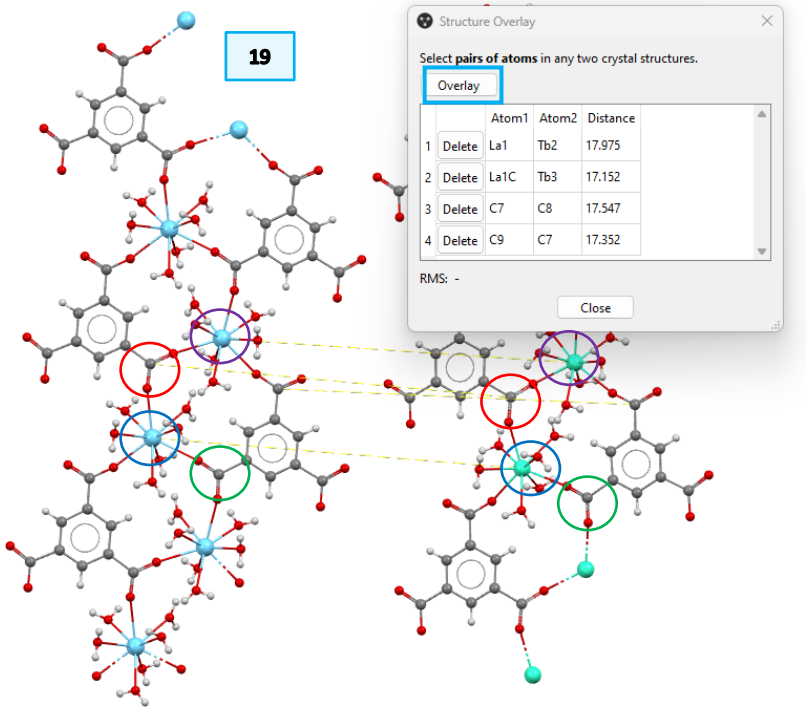
15

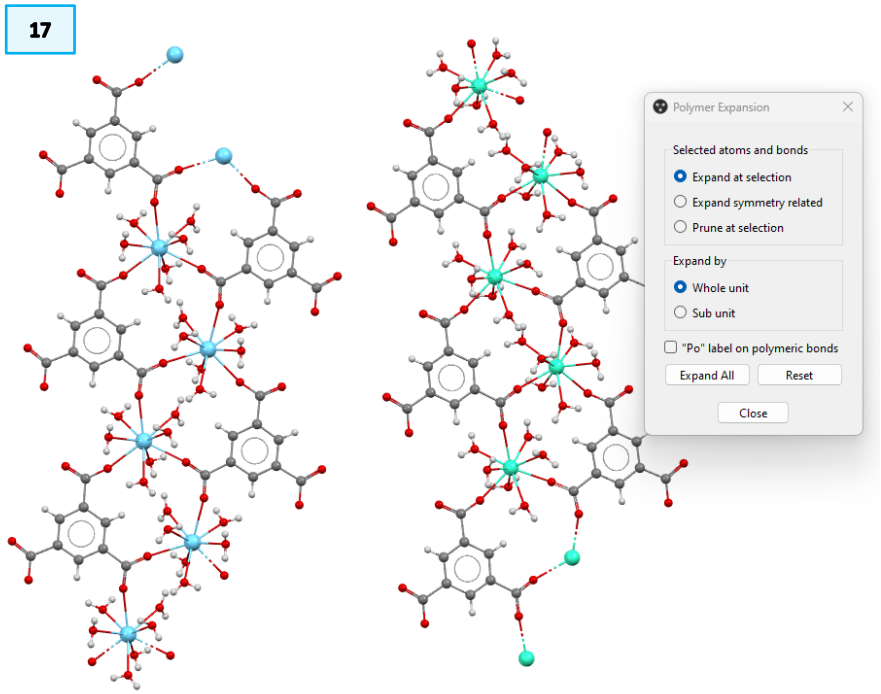


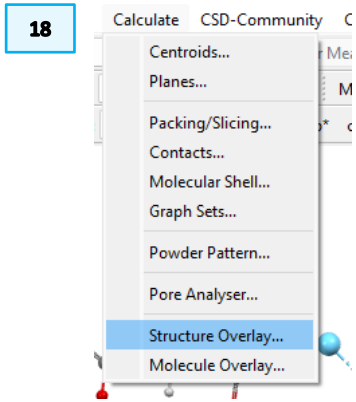
17. For a fair comparison, we need to expand the polymers, as in step 6. Once you have done this, we can see that the two structures do actually look very similar.

18. We will use the Structure Overlay tool to compare them in a more sophisticated way. From the top menus choose **Calculate > Structure Overlay**.

19. To overlay structures, you need to select pairs of atoms in the same environment. The overlay calculates a RMSD value to report how closely the structures match based on the selected pairs of atoms. The more pairs you select, the closer the RMSD will be to the value for the whole structure. We suggest you start by picking two pairs of metal atoms in the same environment and two a pairs of carbon atoms (e.g. of the carboxylate groups) in the same environment. As you add pairs, they will appear in the dialogue. Once you have finished picking, click **Overlay**.





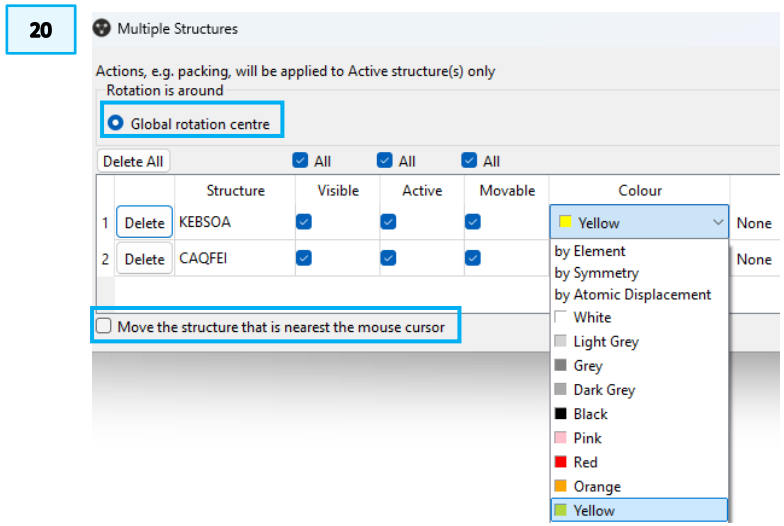


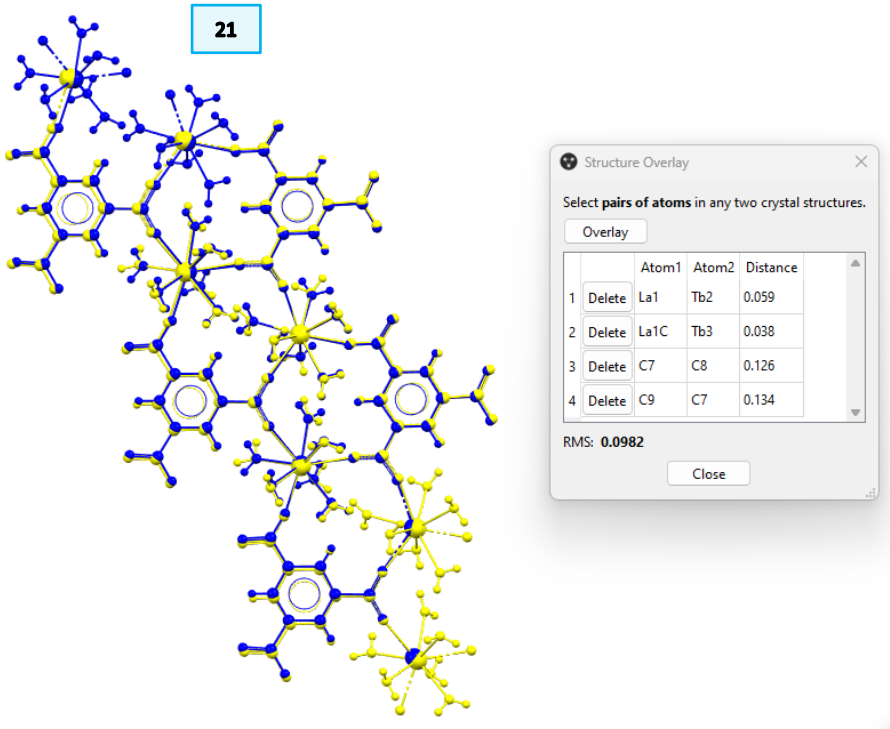
20. The structures will appear superimposed. With the atoms chosen as shown in step 19, the RMSD is 0.0982, which is an excellent score. To make it easier to inspect the structures, in the *Multiple Structures* dialogue, untick **Move the Structure that is closest to the mouse cursor** and tick Global rotation centres.

21. For each structure in the *Colour* column in the *Multiple Structures* table, click the drop-down menu and select a colour. We suggest yellow and blue for high contrast.

Conclusion

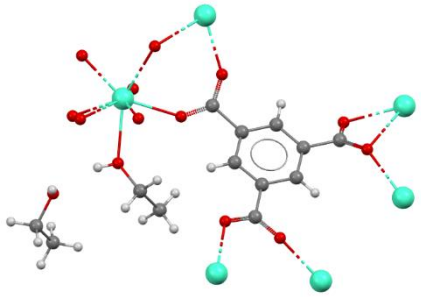
In this example, we have seen how we can compare metal-organic polymers in terms of their coordination environment using the tools available in Mercury, and where. We have seen three different 1D coordination polymers and determined that two of them, $([La(BTC)(H_2O)_6])_n$ and $([Tb(BTC)(H_2O)_6])_n$ are isostructural, whereas $([Er(BTC)(H_2O)_5]H_2O)_n$, in which one water is outside the metal coordination sphere, has a very different structure to either of the La or Tb cases. In the next example, we will explore a LnBTC MOF.





Example 2. Examining Porosity in a Metal-Organic Framework

MOF-76 is a family of compounds related to those studied in Example 1. These compounds have the general formula [Ln(BTC)(H_2O)] – a much lower level of hydration than in the previous examples. Though best known for the lanthanoid members of the family, other trivalent ions such as Y are known to form the structure-type. As with all MOFs they have potential porosity. In this example, we will explore the porosity of a MOF-76 type compound.



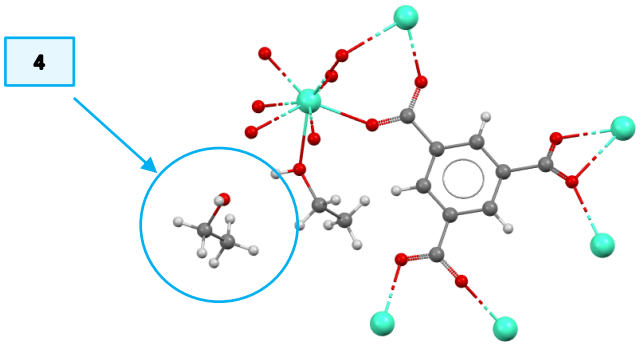
CSD Entry EBEXEQ01, ([Gd(BTC)(EtOH)] EtOH)$_n$

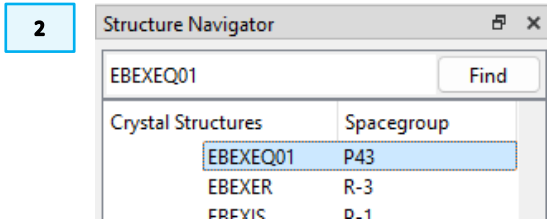
1. Open a new instance of Mercury from the Start menu or desktop icon.

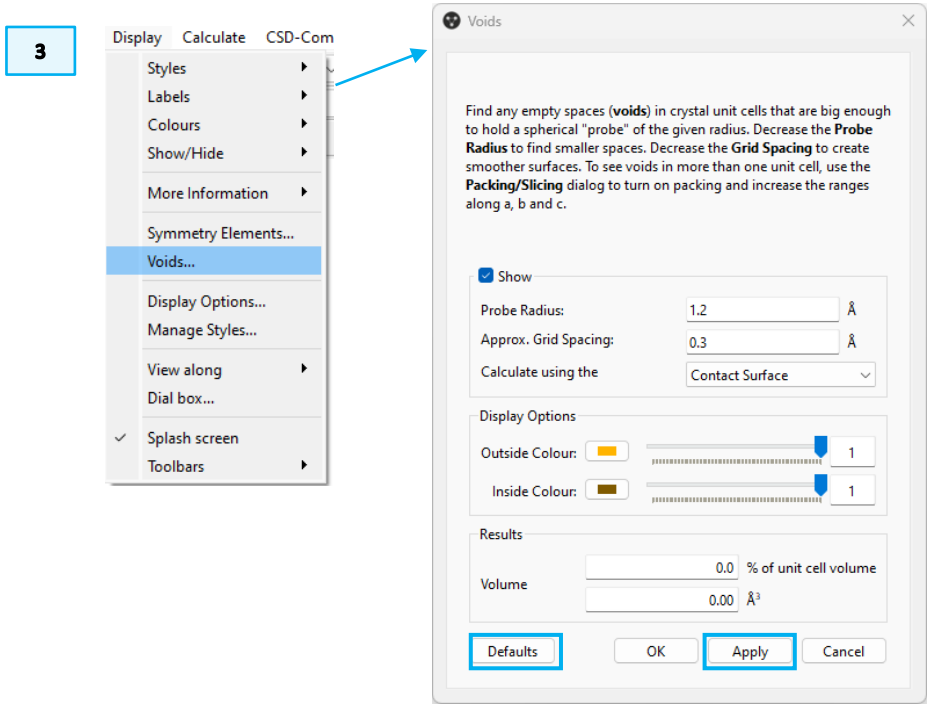
2. In the *Structure Navigator*, type “EBEXEQ01”. This will load the structure of ([Gd(BTC)(EtOH)]EtOH)$_n$. You will see that there is one coordinated ethanol molecule and one co-crystallised solvent ethanol molecule.

3. We will start our analysis by assessing if there is any void space in this structure. We can use the voids display as a quick check. From the top menus, choose **Display > Voids…**. In the resulting *Voids* dialogue, ensure the settings are default by clicking **Defaults**, then click **Apply**. You will see that the *Results* show 0.0 % of unit cell volume. This indicates that there is negligible pore space available in this structure when calculated with this particular value of the probe radius (1.2 Å). Bonus: you can try reducing the size of probe radius to see if any void space appears.

4. We noted that one molecule of ethanol is co-crystallised solvent. Therefore, we might wonder if pore space would be available if the solvent ethanol were not present. We can explore this by editing the structure.



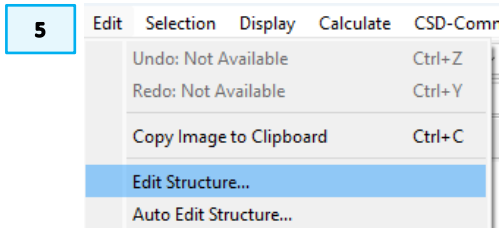


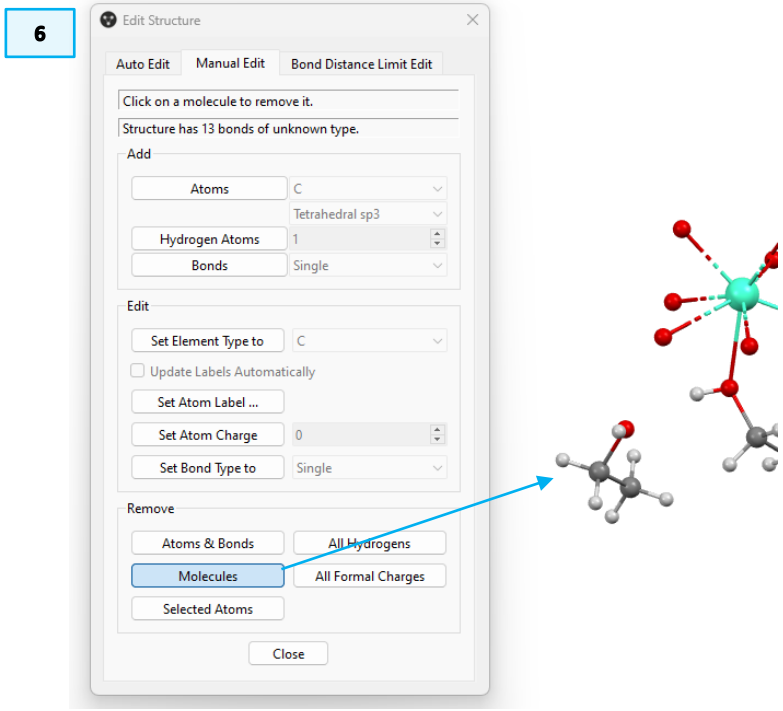


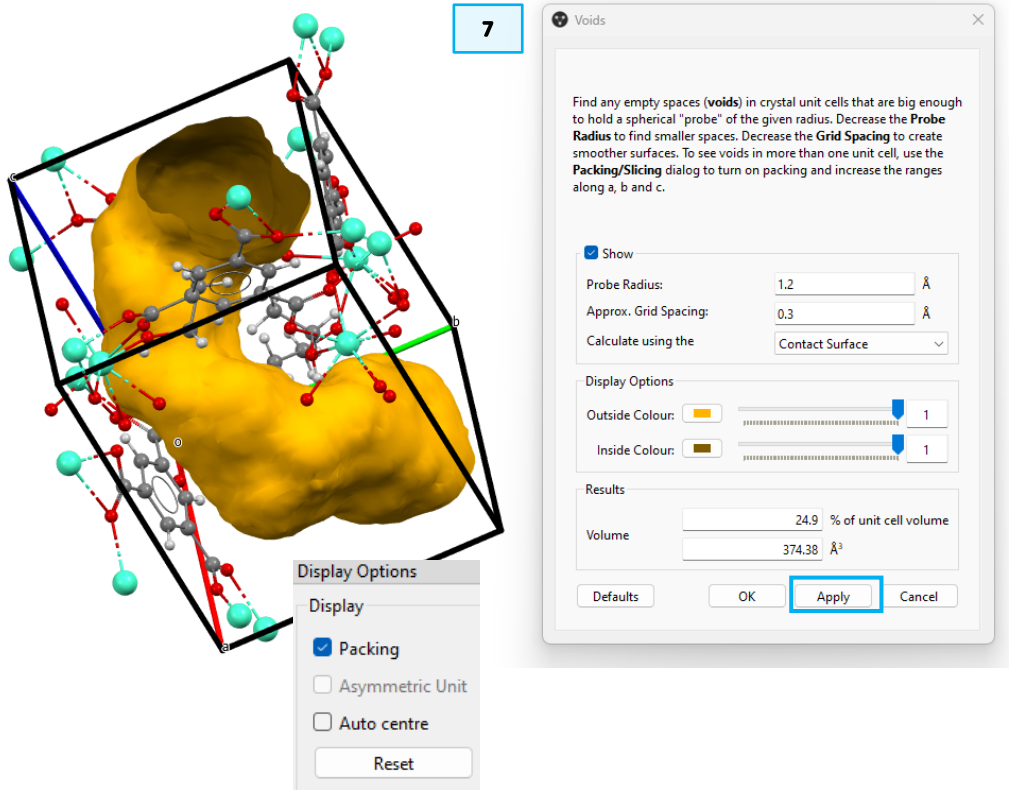
5. From the top menu, select **Edit > Edit Structure**.

6. In the *Manual Edit* tab of the *Edit Structure* dialogue, in the *Remove* section, click **Molecules** and then click on the ethanol molecule in the structure. It will be deleted. If you make a mistake when editing, press Ctrl + Z.

7. Return to the Voids dialogue and click **Apply** (if you have adjusted the settings in step 3, click **Default** first). You should find that there is significant void volume (24.9 % of unit cell volume). You can turn on **Packing** from the display option to see how the resulting pore space is distributed in the structure.



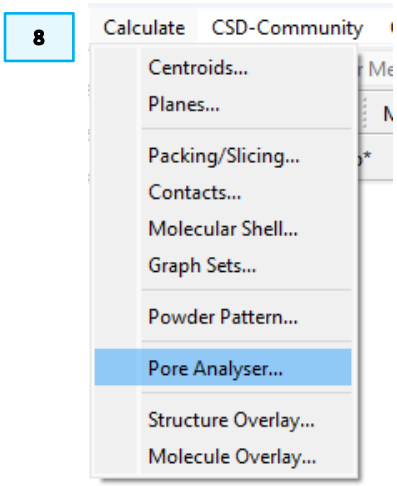


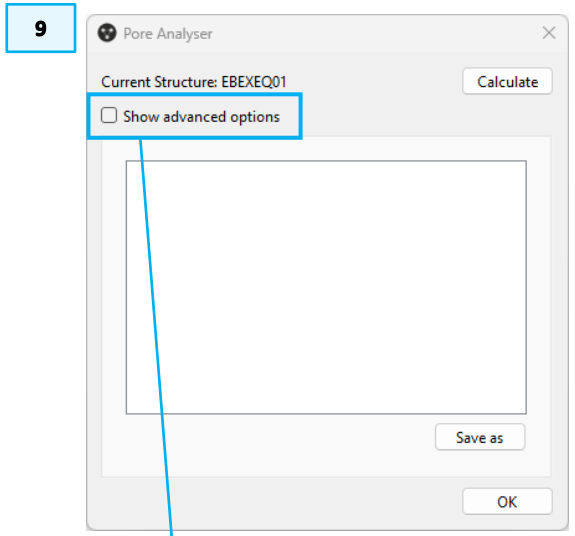


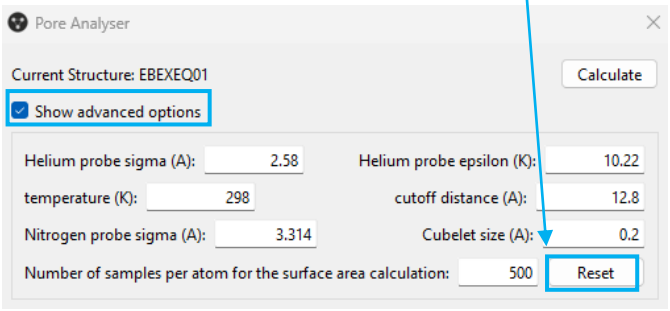
8. We can calculate more properties associated with the void space by using the Pore Analyser tool. From the top menus, choose **Calculate > Pore Analyser…**. This will open the *Pore Analyser* dialogue

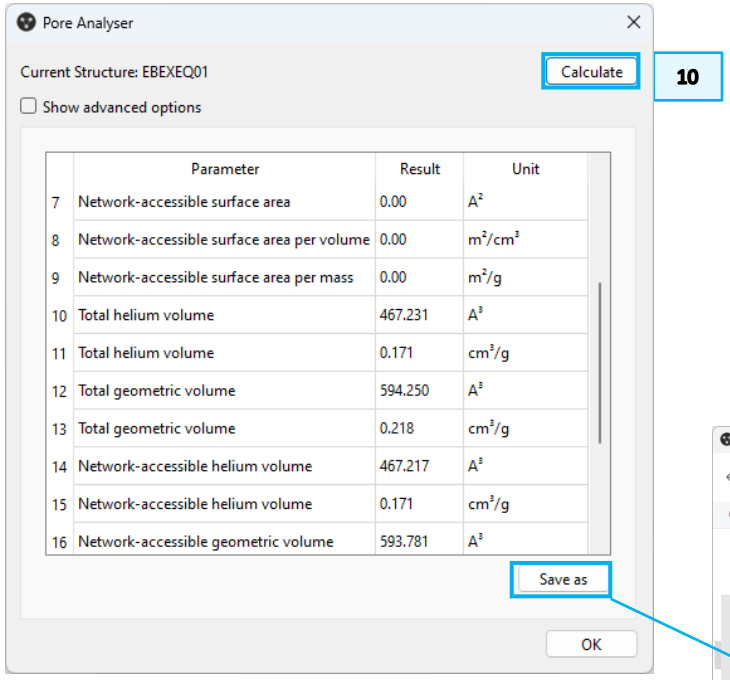
9. Click on Show advanced options. If you have changed any of these settings before, click **Reset**. Untick **Show advanced options**.

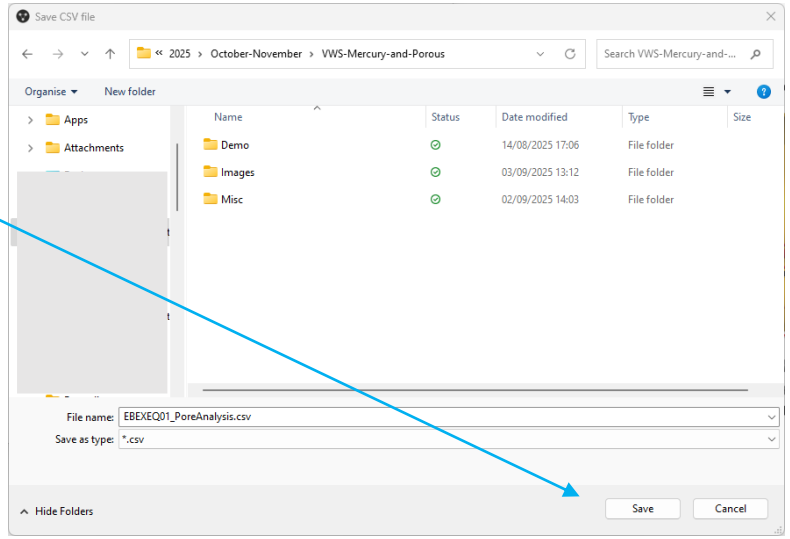
10. Click **Calculate**. The calculation will return various properties related porous structures, in several different units for each property. You can Save these results for future reference by clicking on **Save as** and choosing a suitable file name.









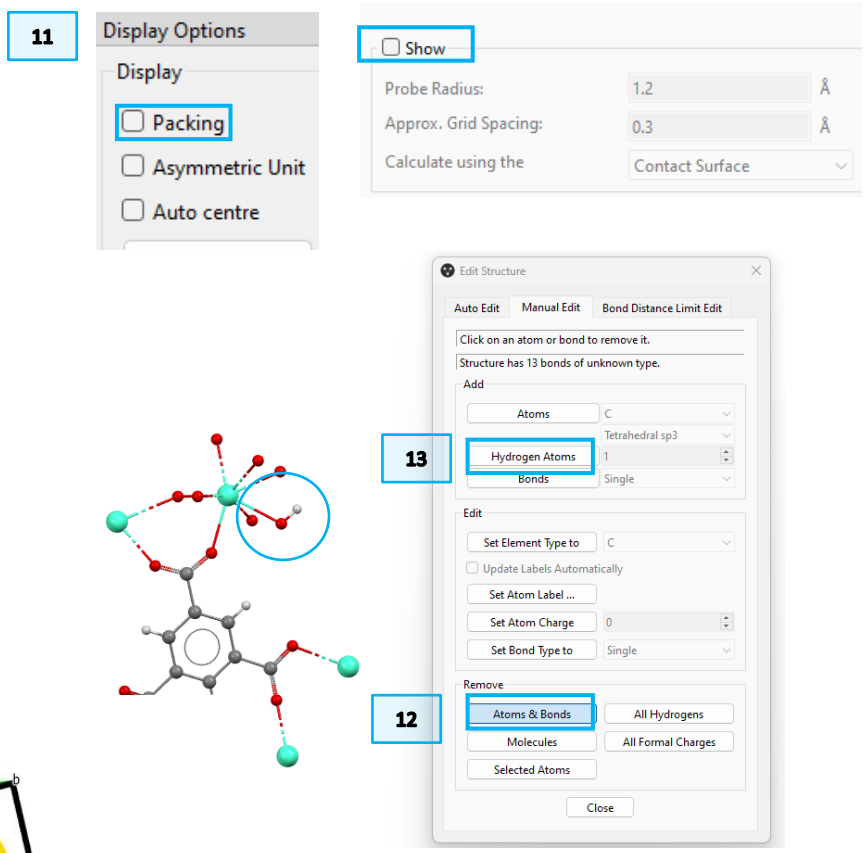


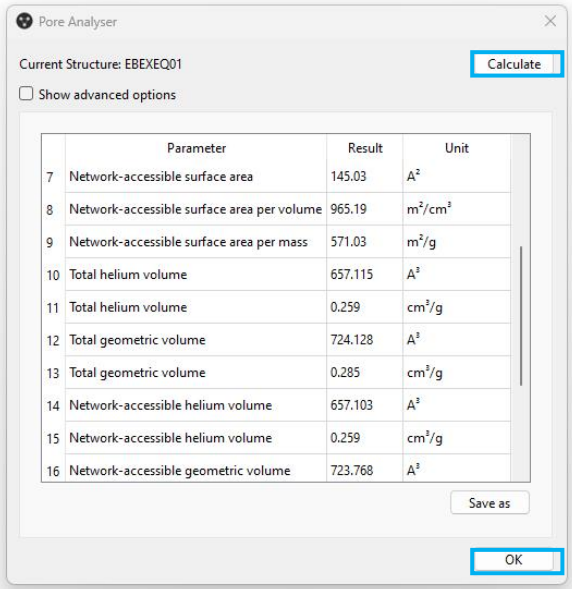
11. Untick **Packing** the *Display Options* toolbar and untick **Show** in the *Voids* dialogue.

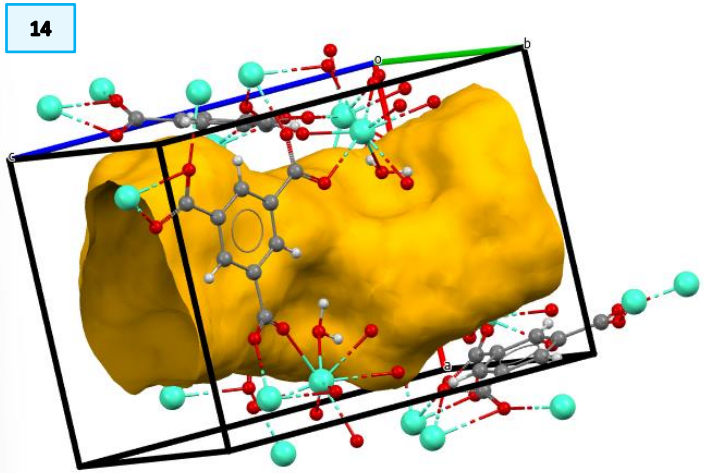
12. We shall further modify the structure by changing the coordinating ethanol molecule into a water molecule. In the *Edit* dialogue, in the *Remove* section of the *Manual Edit* tab, click **Atoms & Bonds**. The click on all the C and H atoms of the ethyl group in the visualizer area to remove the atoms.

13. In the *Add* section of the *Manual Edit* tab of the *Edit Structure* dialogue, click **Hydrogen Atoms**, then click on the remaining oxygen of the ethanol molecule that you just edited. You have now essentially edited the structure to be that expected for the MOF-76(Gd) structure.

14. Recalculate the pore properties in *Pore Analyser* and visualise the voids again by ticking **Show** in the *Voids* dialogue (and turn on Packing again from Display Options if you like; it is not essential). You will see that the pore space has increased again. Save the calculation results if you wish.



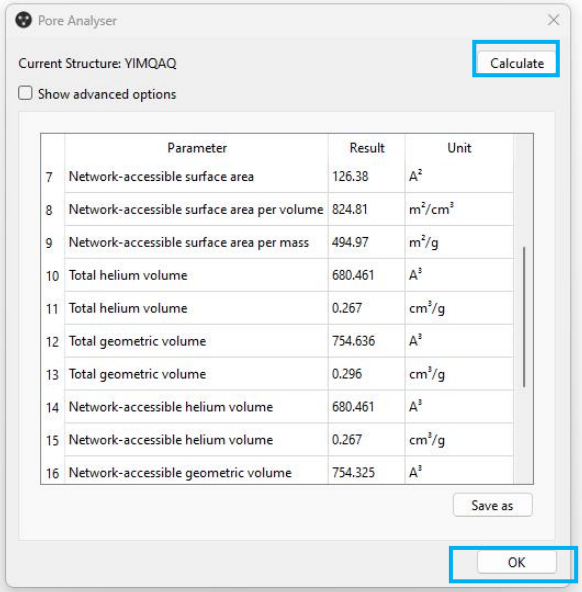


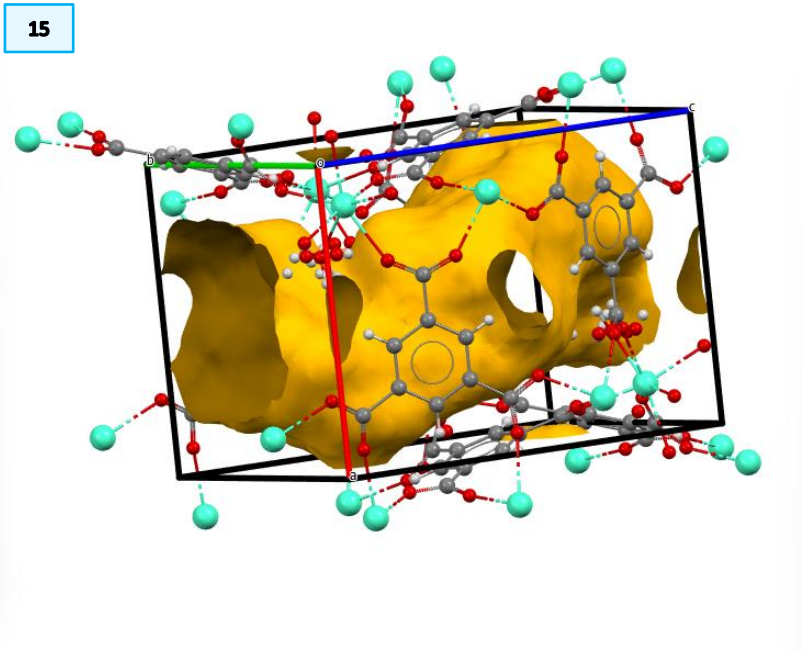


15. As it happens, the crystal structure of MOF-76(Gd), which has water in place of the coordinating ethanol, is known. It has CSD refcode YIMQAQ. Load it via the Structure Navigator and click Calculate in the *Pore Analyser* dialogue. Save the results.

Conclusion

This example has explored ways of predicting pore properties of closely related structures using structure Editing and Pore Analyser. The calculated values are quite close to those obtained by editing EBEXEQ01, indicating that the framework itself is probably fairly robust. See ***CrystEngComm***, 2011,**13**, 5849-5857 (https://doi.org/10.1039/C1CE05468F) for further discussion.



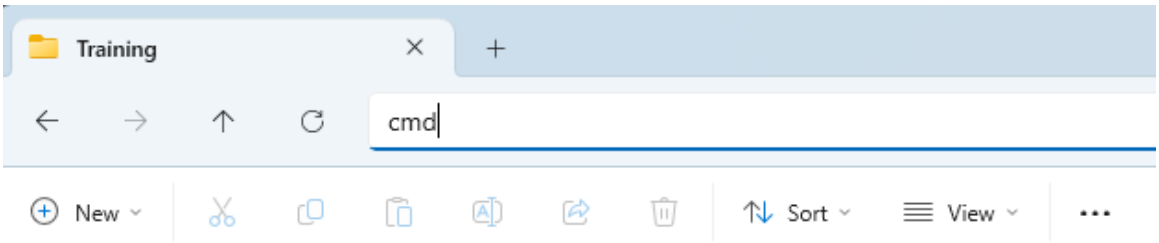


Example 3. Batch Calculation of Pore Properties using the CSD Python API

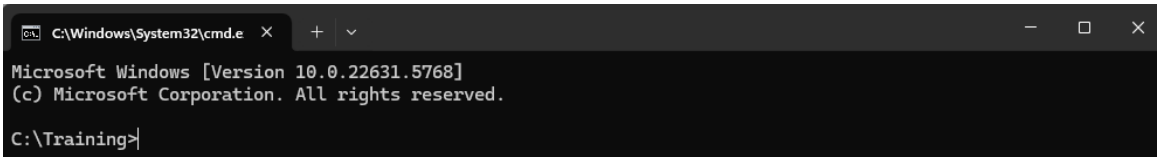
This section is suitable for those familiar with Python and the CSD Python API. If you are unfamiliar with the CSD Python API, you can review our introductory self-guided workshop first.

1. For this exercise we will be writing the script in a Python file that we can then run from a command prompt later. Start by creating a folder where you will save your Python files in a place where you have read and write access, for example C:\training\ for Windows, or something equivalent on macOS or Linux.

2. Open the command prompt from this folder. In Windows you can type 'cmd' in the File Explorer tab and press 'Enter'. In Linux you can right click on the folder and select Open in Terminal. In macOS, right click on the folder, select Services then click New Terminal at Folder.



The command prompt should now appear.

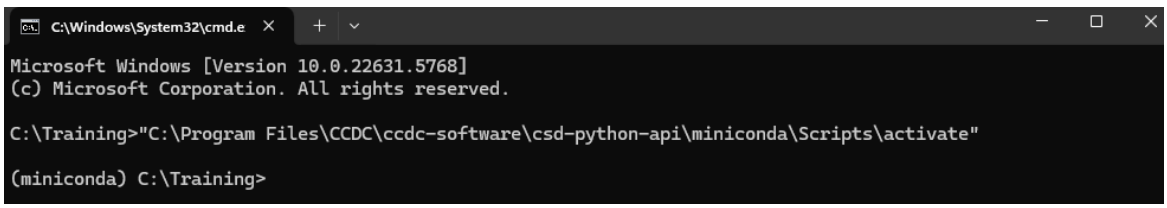


3. To run your Python scripts from the command prompt, you will first need to activate your environment. The activation method will vary depending on the platform:

   - **Windows**: Open a command prompt window and type (including the " marks):
     `"C:\path\to\CCDC\ccdc-software\csd-python-api\miniconda\Scripts\activate"`
   - **MacOS/Linux**: Open a terminal window and change directory to the CSD Python API bin folder:
     `cd /Applications/CCDC/ccdc-software/csd-python-api/miniconda/bin`

     Then activate the environment with:
     `source activate`

   Replace "path\to" with the exact path to your ccdc-software install. If the activation is successful, (miniconda) will appear at the beginning of your command prompt:



4. We can now start writing our script. In the folder you created, open your preferred text editor, and create a new Python file called *pore_analyser.py*. The following steps show the code that you should write in your Python file, along with explanations of what the code does.

5. The CSD Python API makes use of different modules to do different things. The `ccdc.io` module is used to read and write entries, molecules, and crystals. The `ccdc.descriptors` module contains classes concerned for calculating descriptors. To make use of modules, we first need to import them. We shall additionally import the Python packages `pandas` and `pathlib` to save the output later on. Add the following lines at the start of your script.

```
from ccdc.io import EntryReader
from ccdc.descriptors import CrystalDescriptors
from pathlib import Path
import pandas as pd
```

6. We next need to read the CSD entries. To do this we will use the EntryReader. The entries that we will read (refcodes in the list) have been selected due to their structural/compositional similarity. Add the following lines of code to your script:

```
# Read CSD entries
entry_reader = EntryReader('CSD')

# Selection of MOF-76 type structures M = lanthanoid, yttrium
refcodes = ['MARXEK', 'NADZID', 'QOTZEG', 'SADLIU',
            'SEHXIN', 'UPIBOM', 'YIMPAP', 'YIMPIX', 'YIMSAS']
```

7. We can now proceed to calculate the pore properties of these entries. The first part of the following code is just to give nicely formatted print output on the command prompt window. The remaining code gets the `Entry` object for each refcode and from each `Entry` object the `Crystal` object. We then calculate the pore properties using the `CrystalDescriptors.PoreAnalyser` object and its attributes and print the output to the console. We also collect the data in a list of dictionaries for use later. Add the following to your code:

```
# Headers for table
print(f"{'Refcode':<10} {'Formula':<20} {'He Volume (Å³)':>15} {'System
Vol (Å³)':>18}") Previous line continued.
print("-" * 65)

# Calculate pore properties
results = []
for refcode in refcodes:
    mof = entry_reader.entry(refcode)
    formula = mof.formula
    crys = mof.crystal  # Need crystal object to calculate descriptors
    pore_analyser = CrystalDescriptors.PoreAnalyser(crys)
    He_vol_tot = pore_analyser.total_helium_volume
    sys_vol = pore_analyser.system_volume
    results.append({'Refcode': refcode,
                    'Formula': formula,
                    'He Volume (Å³)': He_vol_tot,
                    'System Vol (Å³)': sys_vol})
    print(f"{refcode:<10} {formula:<20} {He_vol_tot:15.3f}
{sys_vol:18.3f}") Previous line continued.
```

8. We will now create a DataFrame to store the calculation results and use it to create a CSV file. To your script add the following:

```
# Create DataFrame
df = pd.DataFrame(results)
df = df.round(3)

# Write data to csv
workdir = Path.cwd()
df.to_csv(f'{workdir}/results.csv', index=False, encoding='utf-8-sig')
```

9. Run the script by typing `python pore_analyser.py` in the command prompt. You should see the following output:

Refcode	Formula	He Volume (Å³)	System Vol (Å³)
MARXEK	(C9 H5 Lu1 O7)n	713.613	1502.141
NADZID	(C9 H5 O7 Y1)n	467.468	1254.341
QOTZEG	(C9 H5 La1 O7)n	690.006	1509.310
SADLIU	(C18 H10 O14 Yb2)n	724.094	1510.517
SEHXIN	(C9 H5 Eu1 O7)n	719.660	1539.050
UPIBOM	(C9 H5 O7 Tb1)n	718.023	1550.838
YIMPAP	(C9 H5 Ce1 O7)n	648.294	1538.086
YIMPIX	(C9 H5 Nd1 O7)n	662.248	1523.750
YIMSAS	(C9 H5 O7 Tb1)n	662.465	1512.953

Additionally, you should find the file 'results.csv' in the directory where you are running your script e.g. `C:\Training\results.csv`.

The majority of these structures have total helium volumes approximately in the range 650—725 Å³, however, the yttrium MOF (NADZID) has lower porosity in terms of absolute volume per unit cell. Inspecting the structure in Mercury, you will see that it is not the tetragonal structure typical of MOF-76 but a monoclinic structure. (A MOF-76(Y) structure does exist, see for example YODDOO; we have just chosen a small selection of CSD structures for this exercise.) **Bonus:** use the `convert_a3_to_cm3_per_g(volume)` method to compare the structures. Does the picture change? If so, why?

e.g `cm3_per_g = pore_analyser.convert_a3_to_cm3_per_g(He_vol_tot)`

Conclusion

In this exercise, we have seen how Pore Analyser can be accessed programmatically via the CSD Python API for batch processing. We examined only a small number of structures and calculated properties; however you can see that combined with searching and other CSD tools exposed via the Python API, you can gain a very large amount of information in a relatively short time. This could be useful if you wish to screen structures with suitable porosity for a particular application.

You can download full script GitHub here.

```python
from ccdc.io import EntryReader
from ccdc.descriptors import CrystalDescriptors
from pathlib import Path
import pandas as pd


# Read CSD entries
entry_reader = EntryReader('CSD')

# Selection of MOF-76 type structures M = lanthanoid, yttrium
refcodes = ['MARXEK', 'NADZID', 'QOTZEG', 'SADLIU',
            'SEHXIN', 'UPIBOM', 'YIMPAP', 'YIMPIX', 'YIMSAS']

# Headers for table
print(f"{'Refcode':<10} {'Formula':<20} {'He Volume (Å³)':>15} {'System Vol (Å³)':>18}")
print("-" * 65)

# Calculate pore properties
results = []
for refcode in refcodes:
    mof = entry_reader.entry(refcode)
    formula = mof.formula
    crys = mof.crystal  # Need crystal object to calcuate descriptors
    pore_analyser = CrystalDescriptors.PoreAnalyser(crys)
    He_vol_tot = pore_analyser.total_helium_volume
    sys_vol = pore_analyser.system_volume
    results.append({'Refcode': refcode,
                    'Formula': formula,
                    'He Volume (Å³)': He_vol_tot,
                    'System Vol (Å³)': sys_vol})
    print(f"{refcode:<10} {formula:<20} {He_vol_tot:15.3f} {sys_vol:18.3f}")

# Create DataFrame
df = pd.DataFrame(results)
df = df.round(3)

# Write data to csv
workdir = Path.cwd()
df.to_csv(f'{workdir}/results.csv', index=False, encoding='utf-8-sig')

```

Summary

In this workshop, we have made use of a combination of tools available in Mercury to explore materials with potential porosity. We have probed fundamental coordination properties and seen how to compare materials at the structure level. We have then seen how the voids properties of MOFs results can be interrogated using the Pore Analyser tool in Mercury. You should now be able to:

- Visualize and manipulate polymeric structures in Mercury
- Overlay structures in Mercury
- Edit a structure in Mercury
- Run a Pore Analyser calculation in Mercury
- Write a Python script that uses Pore Analyser via the CSD Python API

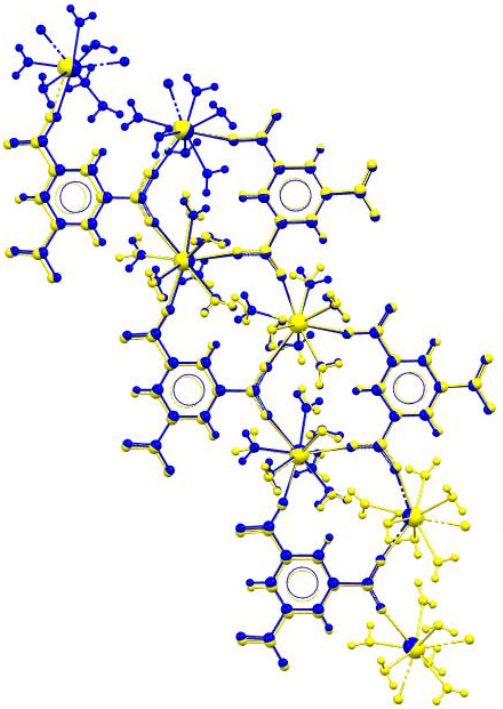
For your reference, you can consult the Mercury user guide and CSD Python API documentation.

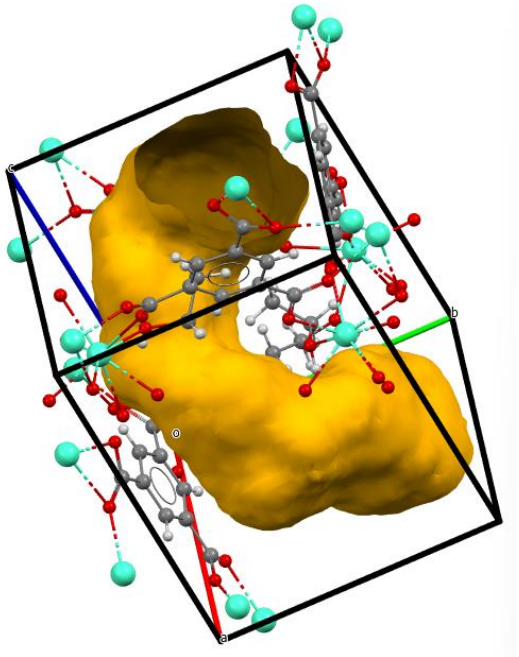
Next Steps

If you have enjoyed this workshop, you can find self-guided workshops on how you can combine searching in ConQuest and Mercury here for exploration of porous materials. You might also like to try our on-demand module on Visualization in Mercury or explore instructional videos on functionality such as Pore Analyser on our YouTube channel.

Feedback

We hope this workshop improved your understanding of Structure Overlay and Pore Analyser and you found it useful for your work. As we aim to continuously improve our training materials, we would love to hear your feedback. Follow the link on the workshop homepage and insert the workshop code, which for this self-guided workshop is MER-007. It will only take 5 minutes and your feedback is anonymous. Thank you!





Glossary

API

Application Programming Interface. This is a software intermediary that allows two applications to communicate.

Metal-Organic Frameworks (MOFs)

A metal-organic framework, abbreviated to a MOF, is a coordination network with organic ligands containing potential voids. In the CSD MOF subsets, each subset with a particular dimensionality (1D, 2D or 3D) also includes coordination polymers and coordination networks of the same dimensionality. In these subsets, the identified MOFs are not restricted to adsorption applications; therefore, 1D, 2D, and 3D MOF and MOF-like structures with all kinds of pore sizes as well as nonporous structures are included. Further information can be found in S. R. Batten, N. R. Champness, X-M. Chen, J. Garcia-Martinez, S. Kitagawa, L. Öhrström, M. O'Keeffe, M. P. Suh and J. Reedijk, *Pure Appl. Chem*., 2013, **85**, 1715-1724, and P. Z. Moghadam, A. Li, S. B. Wiggin, A. Tao, A. G. P. Maloney, P. A. Wood, S. C. Ward and D. Fairen-Jimenez, *Chem. Mater*., 2017, **29**, 2618-2625.

MOF-76

MOF-76 was first reported in *J. Am. Chem. Soc*. 2005, **127**, 5, 1504–1518, corresponding specifically to the compound Tb(BTC)$(H_2O)_{1.5}$(DMF), CSD refcode FIJFEK. The structure consists of helical rods with 7-coordinate Tb(III) centres coordinated by 6 carboxylate groups and one water ligand. The carboxylate groups bridge metal centres (μ-κ^1O:κ^1O'). Each rod is connected to four neighbours via the benzene ring of the BTC link. The rods pack in a tetragonal fashion resulting in square channels in the *c* direction. Many more MOFs of this type have since been reported. See for example *Chem. Mater.* 2010, **22**, 11, 3316–3322.

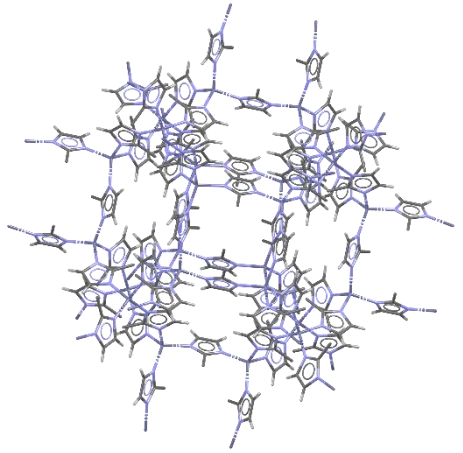
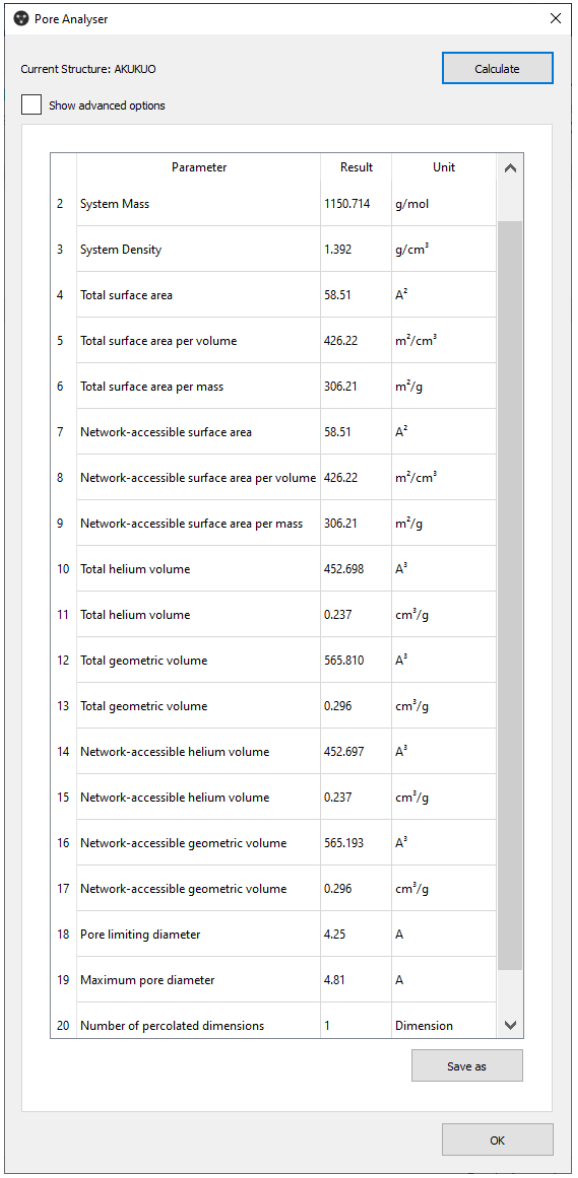


Figure of a MOF, FAWCEN03, ZIF-8.

Pore Analyser

Pore Analyser is the CSD tool for void space analysis. A pore analyser calculation reports several results including:

- System volume, mass, and density.
- Surface area, given as both total and network-accessible parameters.
    - The network accessible value considers if the probe can access the void space within a structure.
    - The total surface area will include isolated void space (closed porosity).
- Volume parameters: helium and geometric volumes differ by considering the size and interactions of the probe atom.
- Pore sizes:
    - The Pore limiting diameter value describes the largest sphere than can percolate through the void network.
    - The maximum pore diameter gives the diameter of the largest sphere that can fit in void space of the structure.
- The number of percolated dimensions:
    - Values range from 0 (isolated pores) to 3 (network of channels or layers).



	Parameter	Result	Unit
2	System Mass	1150.714	g/mol
3	System Density	1.392	g/cm³
4	Total surface area	58.51	A²
5	Total surface area per volume	426.22	m²/cm³
6	Total surface area per mass	306.21	m²/g
7	Network-accessible surface area	58.51	A²
8	Network-accessible surface area per volume	426.22	m²/cm³
9	Network-accessible surface area per mass	306.21	m²/g
10	Total helium volume	452.698	A³
11	Total helium volume	0.237	cm³/g
12	Total geometric volume	565.810	A³
13	Total geometric volume	0.296	cm³/g
14	Network-accessible helium volume	452.697	A³
15	Network-accessible helium volume	0.237	cm³/g
16	Network-accessible geometric volume	565.193	A³
17	Network-accessible geometric volume	0.296	cm³/g
18	Pore limiting diameter	4.25	A
19	Maximum pore diameter	4.81	A
20	Number of percolated dimensions	1	Dimension

The Pore Analyser results table.

Refcode
A refcode is a CSD entry identifier comprising six letters e.g. ABACOF. Two digits identifying additional structure determinations e.g. ABACOF03.

Root Mean Square Deviation (RMSD)
The root mean square deviation (RMSD) is a commonly used measure of the difference between two sets of values (usually comparing observed data to estimated data). The RMSD is defined as the square root of the mean squared error. In Mercury this is used to measure the geometric difference between packing features or packing patterns in crystal structures.

Voids
The overall volume of a crystal structure can be partitioned into occupied 'network' and unoccupied void space. The Voids feature in Mercury displays any empty space that is big enough to contain a spherical 'probe' of a given radius.

Basics of Mercury Visualization

Mercury is the CCDC's visualization software to view 3D structures of small molecules, generate images, and animations of molecules.

In the following we will see some of the basics of navigation and visualization in Mercury that you will find helpful to support your analysis.

In the **Mercury interface** we find:

- **At the top**: list of menus from which we can access visualization and analysis options, and other CSD components such as CSD-Materials.
- **On the right-hand side**: the **Structure Navigator**, with the database loaded (depending on your licence). The Structure Navigator allows you to select a refcode to visualize in the main Mercury window.
- Beneath the main display window: Display options toolbar. You can quickly view a packing diagram, display Hydrogen bonding and detailed information about the molecule using the More Info option.

Using the **mouse to enhance visualization**:

- Left mouse button and move – rotate molecules.
- Middle Mouse wheel – move molecules up and down.
- Right mouse button and move up and down – zoom in and out of molecules.
- Shift + Left mouse button and move - rotate in the plane molecules.
- Ctrl + Left mouse button and move - translate molecules.

Right click:
a) Near a molecule and
b) Away from a molecule