Electronic Supplementary Material (ESI) for CrystEngComm.

Supplementary Information for:

Enabling efficient exploration of metal-organic frameworks in the Cambridge Structural Database

Aurelia Li,[a] *Rocio Bueno-Perez,*[a] *Seth Wiggin,*[b] *David Fairen-Jimenez*[a]

[a] Adsorption & Advanced Materials Laboratory (A^{2}ML), Department of Chemical Engineering & Biotechnology, University of Cambridge, Philippa Fawcett Drive, Cambridge CB3 0AS, UK

[b] The Cambridge Crystallographic Data Centre, 12 Union Road, Cambridge, UK

Contents

S1. Comparison between the Non-disordered CSD MOF subset and a 'no-disorder'-filtered search in the CSD MOF subset

The 'Non-disordered' CSD MOF subset does not correspond to a search with the 'no-disorder' filter in the CSD MOF subset, as Figure S1 shows. The readily available Non-disordered CSD MOF subset is tailored to avoid missing hydrogens in the frameworks of the entries. A 'no-disorder' filter will keep all entries with any hydrogen-related disorder. This search will thus include structures with missing hydrogens in the framework.

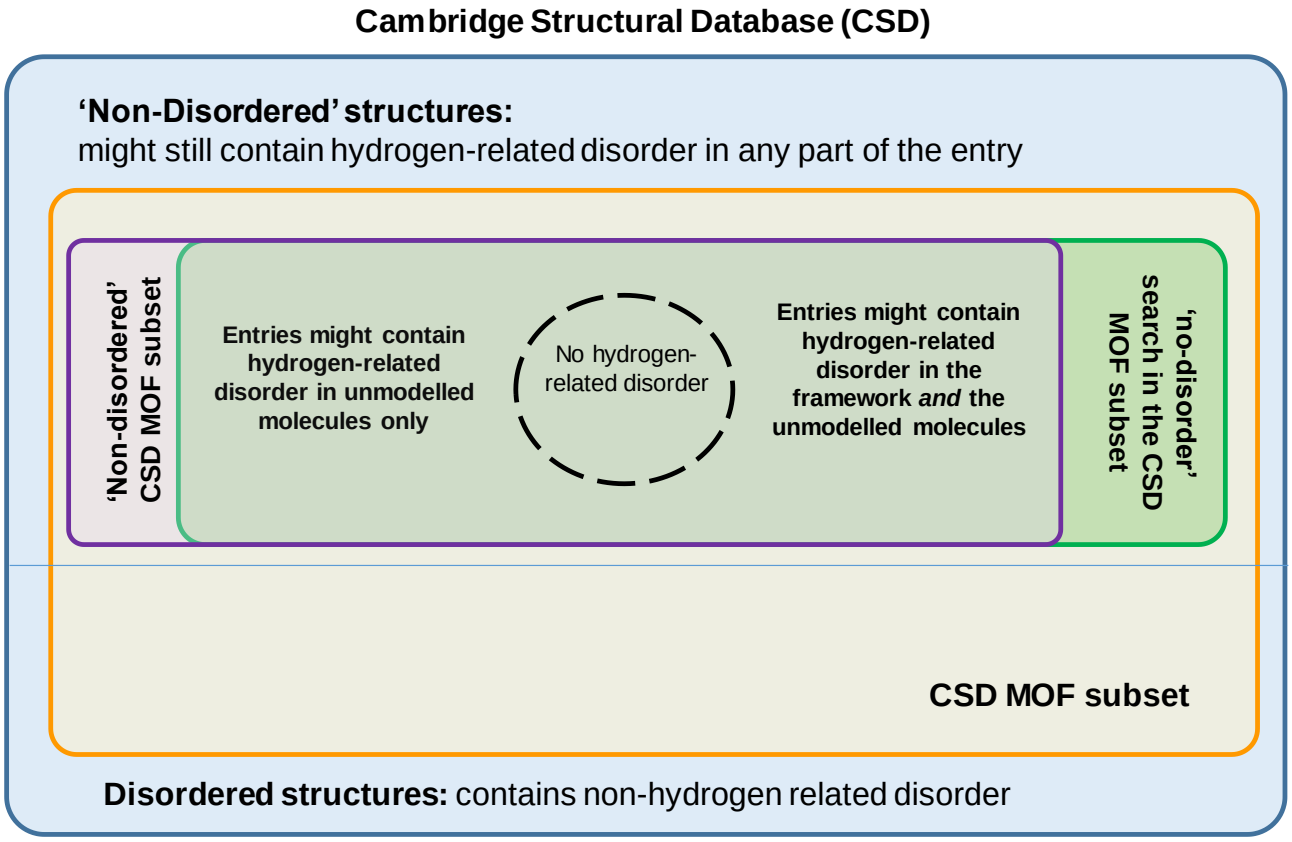

Figure S1 | Differences and overlaps between the Non-disordered CSD MOF subset and a 'no-disorder'-filtered search in the CSD MOF subset.

To carry out a 'no-disorder'-filtered search in ConQuest, create a *Draw* query and draw a single carbon atom (or *Any* atom), then click on *Search*. In the new window, under *Available Databases*, click on *Select Subset*. Then tick *Entries in a predefined hitlist* on the right pane and choose *MOF subset* in the list of subsets. Click *OK*. Make sure *Only Non-disordered* is ticked in the *Filter* pane on the right-hand side of the window. Then click *Search*. You can now combine the obtained hitlist with the Non-disordered MOF subset to see the differences.

S2. Examples of errors in the MOF subset

Below is a non-exhaustive list of example structures found in the *Non-disordered MOF subset*. These structures were found during high-throughput screenings, where calculated results seemed unusual. Readers should be aware that by the time this tutorial is published, these structures might have been removed from the *Non-disordered MOF subset*. We encourage users to report any kind of issue they find, in order to maintain a database of good quality.

Table S1 | Non-exhaustive list of erroneous structures found in the *Non-disordered MOF subset*

Refcode	Error
ADATAC	missing hydrogen in framework
BEHVUH	missing hydrogen in framework
BOQQAB	non-hydrogen related disorder in framework
EDUVII	non-hydrogen related disorder in framework
ERARUM	missing hydrogen in framework
KUFZES	non-hydrogen related disorder in framework
NEJJOE	two frameworks superimposed
NIHWIN	non-hydrogen related disorder in framework
NINVEM	non-hydrogen related disorder in framework
OFEHUE	non-hydrogen related disorder in framework
PEVQOY	missing hydrogen in framework
TURFIX	non-hydrogen related disorder in framework
XUPGIA	missing hydrogen in framework

In addition, below is a few examples of structures that shouldn't have been categorised as 'MOF', as their framework is not organo-metallic. These structures ended up in the MOF subset, because they are accompanied by molecules that are indeed organo-metallic. In order to avoid this, an additional check in the CSD will make sure only the polymeric part of an entry is organo-metallic and needs to satisfy the criteria previously developed.

Table S2 | A few examples of non-MOF structures found in the CSD MOF subset

Refcode	Error
BOJTAW	Framework not organo-metallic
EHAQIN	Framework not organo-metallic
FAVPUQ	Framework not organo-metallic
FAVQAX	Framework not organo-metallic
SONSOF	Framework not organo-metallic

S3. Accessing the newly added MOFs

As the CSD MOF subset is regularly updated, it is also useful for researchers to look at the recently added structures only. As the list of available datasets only shows the entire updated datasets, a few extra manipulations are necessary in order to obtain the latest structures only. For this, we can combine two lists in the *Manage Hitlists* pane under the top menu bar: the MOF subset and the new structures from the update. What is in common between the two lists are the new MOFs from the update. First, load the desired MOF subset as described in the main manuscript. The results are shown in the *View Results* pane. Go to *Manage Hitlists* where MOF_subset is listed. To load the updates, go to *View Databases* > *Entries in CSD version X updates* > *Update N [depending on what is available].* As you can notice, updates are not listed in the *Manage Hitlists* pane. Go back to *View Results* where the updates are loaded, export the list of structures as a GCD file, then import it again from *File* > *Open* > *Refcode list*. This time, the list should show in *Manage Hitlists*. On the left pane of the window, set List A and list B to be MOF_subset and the update list. Make sure the option *common to List A and List B* is ticked in *Generate a List of Entries*, then click OK. This will return a combination list that only includes the newly added MOFs.

S3. Using the CSD Python API to remove bound and unbound solvent

We published a Python script for the batch removal of bound and unbound solvent using the CSD Python API.[1] To use this script,

1. Make sure you have Python installed,
2. Navigate to the folder where you saved the Python script,
3. Have your GCD (extension .gcd) file of structures ready and if necessary, have your MOL2 (extension .mol2) file of solvents ready as well (if you do not provide a list of solvent, the script will use the default list of most common solvents in the CCDC),
4. Open a command-line window in the working folder (click on the navigation bar in your working folder, the path to your current location will be highlighted in dark blue, type cmd and press *Enter*),
5. Type in the command in the following format after the > symbol:
   `python name_of_the_script.py <filepath to the input gcd>`
   For general information about the script, type:
   `python name_of_the_script.py –h or`
   `python name_of_the_script.py –-help`

Table S3 summarizes the list of possible input options to be used with the script.

Table S3 | List of available options to the solvent removal script. The left column indicates the grammar of the input. The right column gives a description of the option. The sections in white give an example of how the command should look like with each input option.

Input options	Option description
-o OUTPUT_DIRECTORY, --output-directory OUTPUT_DIRECTORY	Directory into which to write stripped structures. By default, the output folders will be in the working folder.
python name_of_the _script.py –o <filepath to my output folder> <filepath to the input gcd>	
-m, --monodentate	Whether or not to strip unidentate (or monodentate) structures. This will strip *all* monodentate ligands, whether or not in the solvent list.
python name_of_the _script.py –m <filepath to the input gcd>	
-s SOLVENT_FILE, --solvent-file SOLVENT_FILE	Location of solvent file. By default, the script will use the files of common solvent molecules from the CCDC.
python name_of_the _script.py –s <filepath to the solvent list> <filepath to the input gcd>	

Should there be any installation issues, we advise our readers to check the CSD Python API documentation (https://downloads.ccdc.cam.ac.uk/documentation/API/) and the CSD Python API Forum (https://www.ccdc.cam.ac.uk/forum/csd_python_api/). A step-by-step case-study tutorial is also available at https://www.ccdc.cam.ac.uk/support-and-resources/ccdcresources/CSD-Python-API-workshop-material.pdf.

S4. Using Mercury to remove solvents in a single entry with Mercury

We also published a Python script for the removal of solvents from a selected entry in Mercury. Follow the steps below:

1. Open Mercury,
2. Save the Python script in a folder,
3. In *CSD Python API* > *Options*, click on *Add Location* and add the folder where the Python script is located. You can also change the *Output Directory*. Then click Save.
4. You should now see the script under the *CSD Python API* menu. Select the structure of interest in the database and click on the Python script in the *CSD Python API* menu. A window will show the progress of the script, and you can find the output in the directory you picked at step (iii).

S5. References

1. P. Z. Moghadam, A. Li, S. B. Wiggin, A. Tao, A. G. P. Maloney, P. A. Wood, S. C. Ward and D. Fairen-Jimenez, *Chemistry of Materials*, 2017, **29**, 2618-2625.