

# porE: A code for deterministic and systematic analyses of porosities

Kai Trepte<sup>1, a)</sup> and Sebastian Schwalbe<sup>2</sup>

<sup>1)</sup>*SUNCAT Center for Interface Science and Catalysis, 2575 Sand Hill Road, Menlo Park, California 94025, USA*

<sup>2)</sup>*Institute of Theoretical Physics, TU Bergakademie Freiberg, Leipziger Str. 23, D-09596 Freiberg, Germany*

(Dated: 9 October 2020)

Accurate numerical calculations of porosities and related properties are of importance when analyzing metal-organic frameworks (MOFs). We present PORE, an open-source, general-purpose implementation to compute such properties and discuss all results regarding their sensitivity to numerical parameters. Our code combines the numerical efficiency of FORTRAN with the user-friendliness of PYTHON. Three different approaches to calculate porosities are implemented in PORE, and their advantages and drawbacks are discussed. In contrast to commonly used implementations, our approaches are entirely deterministic and do not require any stochastic averaging. In addition to the calculation of porosities, PORE can calculate pore size distributions and offers the possibility to analyze pore windows. The underlying approaches are outlined, and pore windows are discussed concerning their impact on the analyzed porosities. Comparisons with reference values aim for a clear differentiation between void and accessible porosities, which we provide for a small benchmark set consisting of 8 MOFs. In addition, our approaches are used for a bigger benchmark set containing 370 MOFs, where we determine linear relationships within our approaches as well as to reference values. We show how these relationships can be used to derive corrections to a give porosity approach, minimizing its mean error. As a highlight we show how complex workflows can be designed with a few lines of PYTHON code using PORE.

**Keywords:** porosity, pore size, MOF, porE

## INTRODUCTION

Metal-organic frameworks (MOFs) are a material class aiming for different possible applications<sup>1</sup>, such as gas absorbers<sup>2,3</sup>, catalysts<sup>4-6</sup>, optical sensors<sup>7,8</sup>, and post-synthetic modification (PSM) of MOFs for modulating reaction outcomes and biomedical applications<sup>9</sup>. The class of amorphous MOFs (aMOFs) has possible applications as liquids or melt quenched glasses<sup>10</sup>. Recently, it has been shown that the pore sizes in MOFs can be varied by enforcing an external pressure on a given MOF<sup>11</sup>. Many applications of MOFs are based on the porous nature of these materials, as MOFs typically exhibit several pores. These pores usually have different sizes. With that, an accurate determination of the porosity and the pore sizes is important<sup>12</sup>.

In general, the porosity  $\Phi$  is defined as the empty volume  $V_{\text{empty}}$  within a given total volume  $V_{\text{total}}$ , e.g., the unit cell of a MOF

$$\Phi = \frac{V_{\text{empty}}}{V_{\text{total}}} \cdot 100 \%. \quad (1)$$

While the total volume for crystal structures is always well defined, the empty volume misses this general definition. One major aim of this work is to define and clearly separate two different empty volumes, namely the void volume  $V_{\text{void}}$  and the accessible volume  $V_{\text{acc}}$ . The void volume is the space that is not occupied by any atom in the unit cell, which can easily be analyzed given the sizes

of the atoms, e.g., their respective van der Waals (vdW) radii<sup>13</sup>. From this volume the void porosity  $\Phi_{\text{void}}$  can be obtained, which serves as a first descriptor of a porous material. However, it has to be considered that a void porosity does not necessarily reflect the volume/porosity which can be assumed by adsorbed species. Such a porosity strictly depends on the size of that species<sup>12</sup>. With that another volume occurs, i.e., the accessible volume. Accordingly, the accessible porosity  $\Phi_{\text{acc}}$  can be defined. This porosity, in contrast to  $\Phi_{\text{void}}$ , depends on a probe radius  $r_{\text{probe}}$  which varies for different species. One has to be careful when analyzing the porosity in a material, as the porosity of interest is usually  $\Phi_{\text{acc}}$ . When reporting this quantity one needs to provide the respective probe radius, such as the vdW radius of H ( $r_{\text{probe}} = 1.20 \text{ \AA}$ ) or Xe ( $r_{\text{probe}} = 2.16 \text{ \AA}$ ).

Within this work it will be shown that choosing different probe radii significantly impacts the evaluated porosity. A systematic analysis of the probe radius dependence allows to evaluate the porosity for any adsorbed species, i.e., any atom or molecule with an effective probe radius.

This manuscript is structured as follows: in the next section, the theoretical background as well as detailed aspects of the implementation for the different approaches - a Helium approach (HEA), an overlapping spheres approach (OSA), and a grid point approach (GPA) - are presented, including discussions for the GPA about the grid size dependence as well as the importance of the probe radius dependence for the accessible porosity. Afterwards, an ansatz to calculate the pore size distribution is outlined, following by an approach to determine pore windows. A benchmark for a small test set containing 8 MOFs comes right before an analysis of a much larger test set of 370 MOFs, followed by the conclusion.

<sup>a)</sup>Electronic mail: [ktrepte@stanford.edu](mailto:ktrepte@stanford.edu) or [kai.trepte1987@gmail.com](mailto:kai.trepte1987@gmail.com)

## THEORETICAL BACKGROUND

We developed the PORE code to calculate porosities and related properties. While numerically demanding routines are written in FORTRAN, they are made available through a PYTHON user-interface. Thus, our PORE code combines numerical efficiency with user-friendliness. The open-source PORE code is available under Apache 2.0 licence at GITHUB, i.e., <https://github.com/kaitrepte/porE>, and can easily be installed through the PYTHON pip package manager. For calculations using PORE, one only needs the unit cell parameters and the coordinates of the atoms as input, similar to alternative implementations<sup>14–18</sup>.

Three different approaches to calculate porosities are implemented, namely a He approach (HEA), an overlapping spheres approach (OSA), and a grid point approach (GPA). The HEA is based on an cell list approach to predict He-fraction-like porosities, making it numerically efficient. While the OSA is also very efficient, it rather employs the overlap of atoms to calculate the (void) porosity. In contrast, the GPA uses a grid inside the unit cell, which can be adjusted to systematically approach the numerical limit of the porosity. It is also able to distinguish between void and accessible porosities. All approaches are summarized below.

In addition, PORE can calculate the pore size distribution (PSD) and, based on PSD results, allows for an evaluation of pore windows. These options are described below as well. For an initial verification and illustration of our porosity approaches we propose a small benchmark set called porE8, consisting of 8 MOFs. The porE8 benchmark contains UiO-66<sup>19–21</sup>, UiO-67<sup>20–22</sup>, DUT-8(Ni)<sub>open</sub><sup>23–25</sup>, DUT-8(Ni)<sub>closed</sub><sup>23</sup>, IRMOF-10<sup>26</sup>, MOF-5<sup>27</sup>, HKUST-1<sup>28–30</sup> and MOF-210<sup>31</sup>. For convenience, the two structures (open, closed) of DUT-8(Ni) are abbreviated with DUT-8(Ni)<sub>o</sub> and DUT-8(Ni)<sub>c</sub>. Pore centers determined using PORE are plotted for illustration as spheres within the periodic structures of the investigated MOFs, see Fig. 1 (MOF-210 is shown in the supplemental material).

For an extension of our tests, we use MOF structures optimized using density functional theory with DDEC charges from the CoRE database<sup>32,33</sup>. Of those 502 structures, redundancies and double-countings as found by Barthel et al.<sup>34</sup> were excluded. Our final count is slightly different from Barthel et al.<sup>34</sup>, as we found that some of their exclusion criteria double-counted certain MOFs. Thus instead of the 364 structures mentioned by Barthel et al.<sup>34</sup>, we used 370 which can be found at the PORE GITHUB repository as well. All structures were used as published at Zenodo<sup>32</sup>, and no further modifications were applied. This allows for an easy usage of the set as the structures are already published. We refer to this benchmark set as porE370.

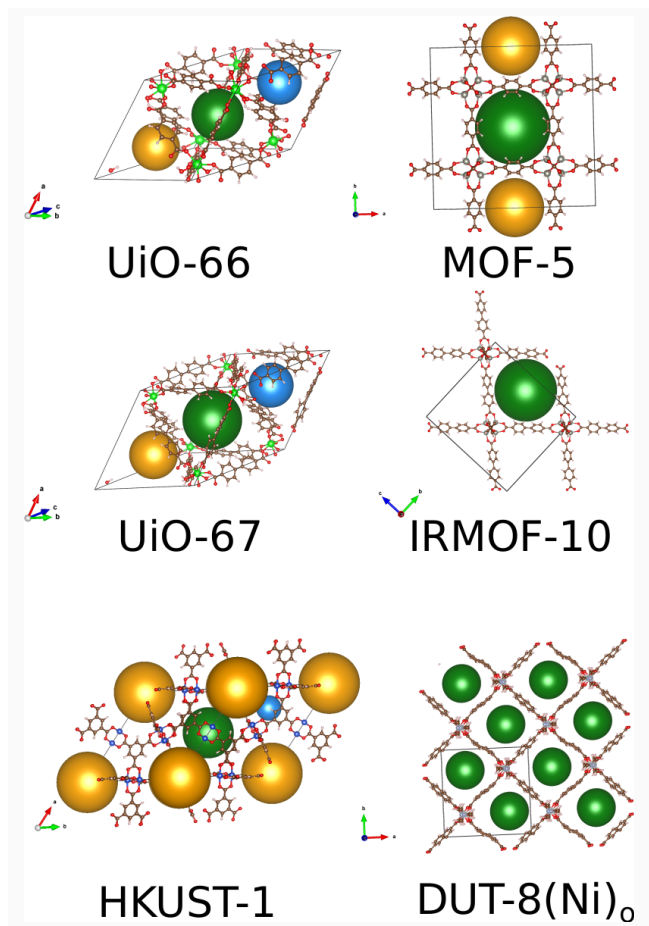


Figure 1. pore8: Pores visualized as spheres for the considered MOFs. HKUST-1, UiO-66, and UiO-67 have three distinct pores. MOF-5 has two characteristic pores, while DUT-8(Ni)<sub>o</sub> and IRMOF-10 have one pore, which is repeated within the periodic structure. The individual pictures are generated with VESTA using the calculated pore centers with additional species having an effective radius equal to the determined pore size. The entire picture was generated using the INKSCAPE program.

### Helium approach (HEA)

A commonly used approach to determine the void volume is often referred to as the calculation of the He void fraction. While various codes differ in details of their respective implementations, all of them share that they insert He atoms in the framework. While in principle it is not known where empty/void positions reside in a given framework, a common approach is to use Monte-Carlo based methods to randomly sample positions inside a framework, determine the interaction energy  $E_{\text{tot}}$  of the He atom with the framework using force fields, and calculate the Boltzmann factor (BF) for each random position

$$\text{BF} = \exp\left(-\frac{E_{\text{tot}}}{k_{\text{B}}T}\right), \quad (2)$$

with  $k_B$  being the Boltzmann constant and  $T$  being the temperature. The BFs are then used to determine the He void fraction<sup>12,16</sup>

$$\Phi_{\text{He}} = \frac{1}{N} \sum_i^N \text{BF}_i \cdot 100\%, \quad (3)$$

with  $N$  being the number of accepted configurations from the Monte-Carlo procedure. In practice, such approaches may need a large number of Monte-Carlo steps to sample the entire empty space correctly, and also require a correspondingly large number of energy evaluations.

In contrast, our implementation is based on a cell list approach. A given unit cell is divided into smaller sub-cells. These sub-cells have the same orientation as the unit cell vectors of the original cell, and therefore conserve lattice symmetries. To determine the sub-cells, the lattice vectors lengths  $\mathbf{L} = (L_x, L_y, L_z)$  are divided by  $\mathbf{M} = (M_x, M_y, M_z)$ . The lattice vectors of the sub-cells are given by

$$(\mathbf{a}_{\text{sub-cell}}, \mathbf{b}_{\text{sub-cell}}, \mathbf{c}_{\text{sub-cell}}) = (\mathbf{a}_{\text{cell}}, \mathbf{b}_{\text{cell}}, \mathbf{c}_{\text{cell}}) / \mathbf{M}, \quad (4)$$

where  $\mathbf{a}_{\text{cell}}$ ,  $\mathbf{b}_{\text{cell}}$ , and  $\mathbf{c}_{\text{cell}}$  are the lattice vectors of the unit cell. The initial values,  $\mathbf{M}_0 = \mathbf{M}(d = 2r_{\text{He}})$ , are calculated to give the best He-He spacing based on the vdW radius of He and  $\mathbf{L}$

$$\mathbf{M} = \lceil \mathbf{L}/d \rceil, \quad (5)$$

with  $d = 2r_{\text{He}}$  and the ceiling function  $\lceil \cdot \rceil$ . Having this initial set,  $\mathbf{M}_0$ , the spacing  $d_{\text{opt}}$  of the He atoms in all directions is optimized using

$$d_{\text{opt}} = \frac{1}{3} \sum_i \frac{L_i}{M_{0,i}} \quad (6)$$

with  $i$  being the vector components  $x, y, z$ . For this cell-optimized He-He spacing, the final set  $\mathbf{M}$  is derived,  $\mathbf{M}_{\text{opt}} = \mathbf{M}(d = d_{\text{opt}})$ , which is used to determine the sub-cells.

Having this set of sub-cells, one can calculate the center of mass (COM) of each sub-cell and place a He atom at these positions. The overlap of all  $N_{\text{COM}}$  inserted He positions with the framework positions is then calculated; all He positions showing an overlap with the framework are excluded. For the overlap calculations, Eq. (8) of the supplemental material is applied. Further, for the He radius half of the optimal spacing  $r_{\text{opt}} = d_{\text{opt}}/2$  is used, because the He interactions are dominated by weak vdW interactions. All other species are described by their covalent radii, because the framework atoms are rather described by stronger covalent bonds.

This exclusion gives the reduced number of He positions  $N_{\text{He}}$ . One can calculate the total empty sub-cell volume

$$V_{\text{sub-cell}} = \mathbf{a}_{\text{sub-cell}} \cdot (\mathbf{b}_{\text{sub-cell}} \times \mathbf{c}_{\text{sub-cell}}) N_{\text{He}} \quad (7)$$

and the total He volume

$$V_{\text{He}} = \frac{4}{3} \pi r_{\text{opt}}^3 N_{\text{He}}, \quad (8)$$

while the total unit cell volume is

$$V_{\text{total}} = \mathbf{a}_{\text{cell}} \cdot (\mathbf{b}_{\text{cell}} \times \mathbf{c}_{\text{cell}}). \quad (9)$$

In cubic systems, our derived He positions are placed on a simple cubic (sc) lattice. One can calculate the atomic packing factor  $f$  to be  $f_{\text{sc}} = 0.52$ . Clearly, a given MOF does not always have a cubic crystal structure. We calculate the actual  $f$  via

$$f_{\text{HEA}} = \frac{V_{\text{He}}}{V_{\text{sub-cell}}}. \quad (10)$$

Note that for cubic frameworks  $f_{\text{HEA}} = f_{\text{sc}}$ . Finally, one can calculate the He void fraction within the HEA as

$$\Phi_{\text{HEA}} = N_{\text{He}} / N_{\text{COM}} \cdot 100 \% \quad (11)$$

$$= V_{\text{sub-cell}} / V_{\text{total}} \cdot 100 \% \quad (12)$$

$$= V_{\text{He}} / (V_{\text{total}} f_{\text{HEA}}) \cdot 100 \%, \quad (13)$$

As an illustrative example we present HEA applied to MOF-5 (see Fig. 2).

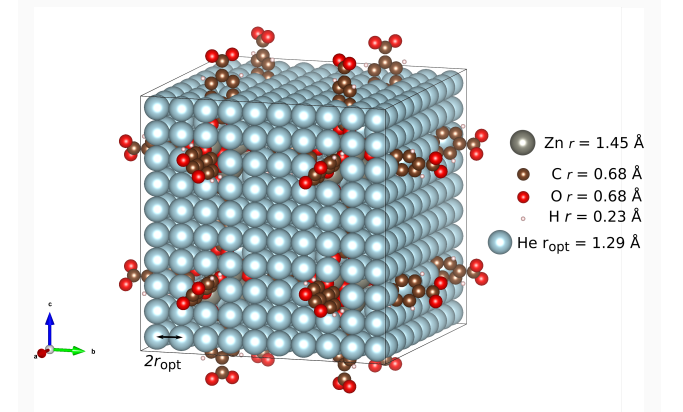


Figure 2. HEA applied to MOF-5 with a resulting He-void fraction of  $\Phi_{\text{HEA}} = 77\%$  and a packing factor of  $f_{\text{HEA}} = 0.52$ . The individual pictures are generated with VESTA with an effective radius equal to the radii in HEA (with  $r_{\text{opt}}$  for He). The entire picture was generated using the INKSCAPE program.

All results for the porE8 benchmark set are given in Tab. 1. We implemented the HEA completely in PYTHON as it has no computational bottlenecks and every property only needs to be computed once. In contrast to other He void fraction implementations our approach does not rely on Monte-Carlo, random sampling and energy evaluations within the unit cell. Our deterministic approach rather places He atoms according to the unit cell vectors, and obtains the He insertion/packing accordingly.

Table 1. porE8: All numbers  $N$  and volumes  $V$  (in  $\text{\AA}^3$ ) as well as the packing factor  $f_{\text{HEA}}$  and the optimal He radius  $r_{\text{opt}}$  (in  $\text{\AA}$ ) calculated using the HEA are given for each MOF in the porE8 benchmark set. The unit cell volume  $V_{\text{total}}$  is given in Tab. 2, and the HEA porosity  $\Phi_{\text{HEA}}$  is abbreviated with  $\eta$ .

MOF	$N_{\text{COM}}$	$N_{\text{He}}$	$V_{\text{sub-cell}}$	$V_{\text{He}}$	$f_{\text{HEA}}$	$r_{\text{opt}}$	$\eta$
DUT-8(Ni) <sub>o</sub>	256	178	2218	1524	0.69	1.27	70
DUT-8(Ni) <sub>c</sub>	60	15	162	122	0.75	1.25	25
UiO-66	216	136	1453	1076	0.74	1.24	63
UiO-67	343	233	3378	2501	0.74	1.37	68
IRMOF-10	770	656	8604	5511	0.64	1.26	85
MOF-5	1000	768	13290	6959	0.52	1.29	77
HKUST-1	392	284	3294	2787	0.85	1.33	72
MOF-210	17576	15308	125767	163714	1.30	1.37	87

## Overlapping sphere approach (OSA)

A different approach to evaluate the porosity within MOFs (or any porous material) is based on a hard sphere model. In this simple model, the overlap of atomic spheres is evaluated and consequently subtracted from the total volume of all spheres/atoms. The volume which is left,  $V_{\text{void}}$ , can be compared to the total volume of the unit cell, providing the (void) porosity  $\Phi_{\text{OSA}}$  as

$$\Phi_{\text{OSA}} = \frac{V_{\text{void}}}{V_{\text{total}}} \cdot 100 \%, \quad (14)$$

with the void, occupied, atomic and overlap volumes given by

$$V_{\text{void}} = V_{\text{total}} - V_{\text{occ}} \quad (15)$$

$$V_{\text{occ}} = V_{\text{atoms}} - V_{\text{overlap}} \quad (16)$$

$$V_{\text{atoms}} = \sum_i V_{i,\text{vdW}} \quad (17)$$

$$V_{\text{overlap}} = \sum_{i,j>i} V_{ij,\text{overlap}}. \quad (18)$$

Here,  $V_{i,\text{vdW}}$  is the volume of a sphere with a radius equal to the vdW radius of atom  $i$ . The sum of the volumes of all atoms  $V_{\text{atoms}}$  minus the overlap  $V_{\text{overlap}}$  between pairs of atoms defines the occupied volume  $V_{\text{occ}}$ . The overlap volume can be calculated analytically following the derivation in the supplemental material, and is evaluated if the sum of the covalent radii of two atoms  $i$  and  $j$  is smaller than their distance, i.e., if  $r_i^{\text{covalent}} + r_j^{\text{covalent}} \leq d_{ij}$ . This simple approach gives reasonable results (see Tab. 2, a comparison to literature values is given in Tab. 7) at essentially no computational cost.

As higher-order terms are neglected, only the two-center overlap is calculated. However, the OSA recovers the total overlap (three-order and higher terms) almost entirely for the benzene molecule. The literature value<sup>35</sup> of the occupied volume in benzene, considering only the

Table 2. porE8: All volumes  $V$  (in  $\text{\AA}^3$ ) for the determination of the porosity  $\Phi$  (in %) of all considered MOFs based on the simple overlapping sphere approach (OSA). The OSA porosity  $\Phi_{\text{OSA}}$  is abbreviated with  $\alpha$ .

MOF	$V_{\text{total}}$	$V_{\text{atoms}}$	$V_{\text{overlap}}$	$V_{\text{occ}}$	$V_{\text{void}}$	$\alpha$
DUT-8(Ni) <sub>o</sub>	3190	1953	709	1244	1946	61
DUT-8(Ni) <sub>c</sub>	648	976	324	652	-4	-1
UiO-66	2308	1992	823	1169	1139	49
UiO-67	4972	2906	1308	1598	3374	68
IRMOF-10	10099	2549	1297	1252	8847	88
MOF-5	17305	6536	2592	3944	13361	77
HKUST-1	4546	2499	1062	1437	3109	68
MOF-210	144400	29385	14937	14448	129952	90

C-C overlap, is  $V_{\text{occ}}^{\text{ref}} = 114.8 \text{ \AA}^3$ . The proposed OSA delivers a value of  $V_{\text{occ}}^{\text{OSA}} = 115.2 \text{ \AA}^3$ . For comparison, the sum of the vdW spheres of all atoms is  $166.9 \text{ \AA}^3$ . The molecular geometry is taken from the CCCBDB<sup>36</sup> with  $d_{\text{C-C}} = 1.397 \text{ \AA}$ , in analogy to Gibson and Scheraga<sup>35</sup>. Given this result, it can be assumed that the results for the overlap should be accurate, even without the higher-order terms.

The main advantage of the OSA is the access of reasonable results with essentially no numerical effort. For example, the calculation for MOF-210 (1854 atoms per unit cell) takes about 1 s (see supplemental material). The main disadvantage is that technically only the void porosity is calculated and there is no information about accessible terms. The approach presented in the next section overcomes this shortcoming.

## Grid point approach (GPA)

A third approach to calculate the porosity is based on a numerical grid inside the unit cell. This procedure requires the explicit treatment of each grid point. Any grid point is either close to an atom (inside its vdW sphere) and can be considered occupied. If no such occupation is found, the grid point is unoccupied. In analogy to equation (12), the void porosity can be evaluated by the number of unoccupied points divided by the total number of grid points

$$\Phi_{\text{void}} = \frac{N_{\text{unoccupied}}}{N_{\text{total}}} \cdot 100 \%. \quad (19)$$

A suitable amount of grid points will provide accurate results. Using this ansatz one obtains an insight into the void volume and thus the void porosity. This is not equivalent to the accessible volume and porosity, which is often given in the literature. Thus, one needs to be careful when comparing, e.g., the values for DUT-8(Ni)<sub>c</sub> as explained in the supplemental material.

The accessible porosity can be obtained by modifications to the presented ansatz. Grid points need to be evaluated such that around each grid point, a sphere with

a probe radius  $r_{\text{probe}}$  is assumed. If this sphere has no contact with the vdW surface of the MOF, all points inside this sphere are considered to be unoccupied as well as accessible. With that, points can be occupied, unoccupied and not accessible or unoccupied and accessible (see Fig. 3).

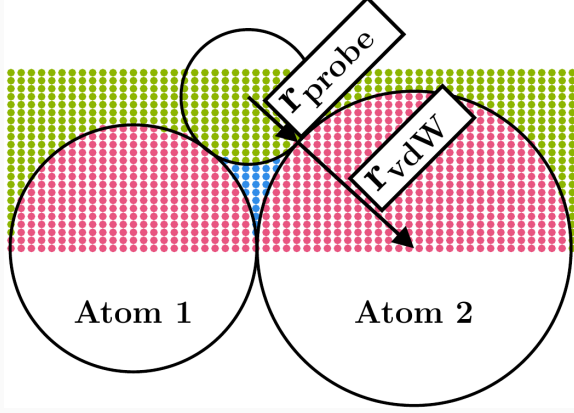


Figure 3. Visual explanation of different grid points in the GPA. If a grid point is inside the vdW sphere of an atom, it is occupied (dark pink). Otherwise, it is unoccupied. Then, two different cases can occur. If the point is in a region which can be accessed by a given probe radius  $r_{\text{probe}}$ , it is unoccupied and accessible (apple green). Otherwise, it is unoccupied and not accessible (bleu de france). The difference in the blue de france and the apple green points defines the difference between void and accessible porosities.

This ansatz gives rise to another quantity, the accessible porosity  $\Phi_{\text{acc}}$ , which depends on a probe radius  $\Phi_{\text{acc}} = \Phi_{\text{acc}}(r_{\text{probe}})$ . The relation  $\Phi_{\text{acc}}(r_{\text{probe}}) \leq \Phi_{\text{void}}$  is clearly fulfilled. A basic outline of the procedure is given in Fig. 4.

From a numerical point of view, the variable  $N_{\text{check\_acc}}$  is introduced (see Fig. 4). With this variable, there is no need to loop over all accessible points to determine which unoccupied points are also accessible. Only selected points have to be evaluated (see Fig. 5). This reduces the computational time while not changing the results. The variable  $N_{\text{check\_acc}}$  is defined as a subset of points chosen from all immediately accessible points, i.e., all points with a distance larger than  $r_{\text{vdW}} + r_{\text{probe}}$  for all atoms. This subset contains points which have a distance within  $r_{\text{vdW}} + \delta$ , with  $\delta = r_{\text{probe}} \cdot (1.0 + h)$ ,  $h = 1.0/\bar{n}$  and  $\bar{n}$  is the average grid point density per Å. Accordingly,  $\delta$  becomes smaller for larger grids. All points  $k$  within this subset are within a distance of  $r_{\text{vdW}} + r_{\text{probe}} \leq d_k \leq r_{\text{vdW}} + \delta$ . Visually, this subset forms a layer of thickness  $\delta - r_{\text{probe}} = r_{\text{probe}} \cdot h = r_{\text{probe}}/\bar{n}$  over the vdW surface of the MOF (indicated in rose quartz in Fig. 5).

The obtained subset ( $N_{\text{check\_acc}}$ ) is used to identify whether unoccupied points are also accessible. If the distance of any unoccupied point to any point in  $N_{\text{check\_acc}}$

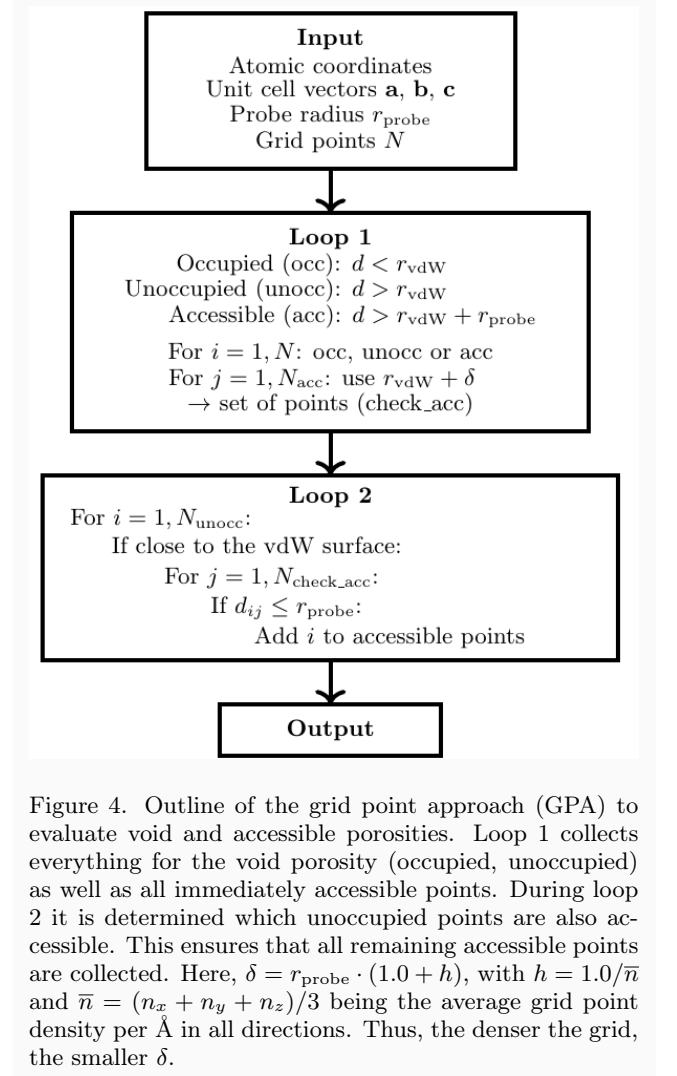


Figure 4. Outline of the grid point approach (GPA) to evaluate void and accessible porosities. Loop 1 collects everything for the void porosity (occupied, unoccupied) as well as all immediately accessible points. During loop 2 it is determined which unoccupied points are also accessible. This ensures that all remaining accessible points are collected. Here,  $\delta = r_{\text{probe}} \cdot (1.0 + h)$ , with  $h = 1.0/\bar{n}$  and  $\bar{n} = (n_x + n_y + n_z)/3$  being the average grid point density per Å in all directions. Thus, the denser the grid, the smaller  $\delta$ .

is smaller than  $r_{\text{probe}}$ , the unoccupied point is accessible (apple green point in Fig. 5). Otherwise, the point is not accessible (bleu de france point in Fig. 5). Additional modifications to this approach can be introduced by only evaluating  $N_{\text{check\_acc}}$  per atom. If an unoccupied point is close to atom  $i$ , only the points  $N_{\text{check\_acc},i}$  need to be evaluated. This is the basis of GPA<sub>sub-grid</sub> (see supplemental material), which gives an additional speed-up.

As a note, the used grid is defined either as a total number of grid points along each cell vectors or as an approximate grid density per Å for all cell vectors. In either case, the grid points are placed along the cell vectors. With that, a unique grid for any system is generated. This grid can either be uniform, i.e., grid density along all cell vectors is the same, or non-uniform. In this work, we only investigate uniform grids.



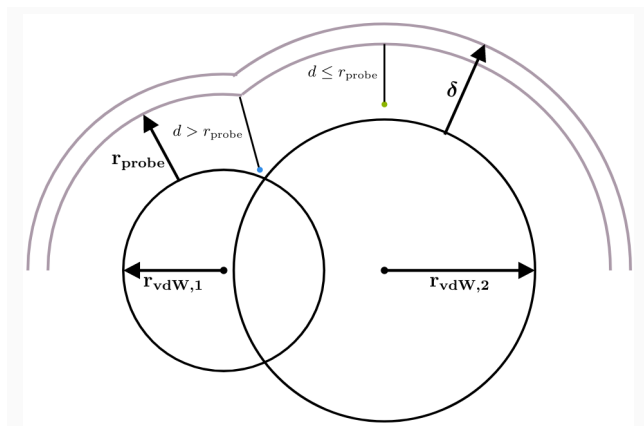


Figure 5. Visualization of  $N_{\text{check\_acc}}$ . Circles for Atom 1 and Atom 2 indicate their respective vdW radii  $r_{\text{vdW}}$ . The subset of points  $N_{\text{check\_acc}}$  is chosen from all immediately accessible points, i.e., all points with a distance larger than  $r_{\text{vdW}} + r_{\text{probe}}$  for all atoms. This subset (indicated in rose quartz) contains points which have a distance within  $r_{\text{vdW}} + \delta$ , where  $\delta = r_{\text{probe}} \cdot (1.0 + h)$ ,  $h = 1.0/\bar{n}$  and  $\bar{n}$  is the average grid point density per Å. Accordingly,  $\delta$  becomes smaller for larger grids. If the distance of an unoccupied point to any  $N_{\text{check\_acc}}$  is smaller than  $r_{\text{probe}}$ , this point is also accessible (apple green). Otherwise, it is not accessible (bleu de france, color code adopted from Fig. 3).

### Grid size dependence

As the GPA depends on the (uniform) distribution of grid points, it is important to determine how dense the grid needs to be to provide numerically reliable results. For the porE8 set, the grid was successively increased and the porosities (void and accessible) were calculated. A probe radius of 1.20 Å was used for all MOFs. In addition, a probe radius of 2.16 Å has been employed for DUT-8(Ni)<sub>o</sub>, UiO-66 and UiO-67. This was done to see whether the porosities converge differently using different probe radii. The results for the considered MOFs are summarized in the supplemental material. For UiO-66, the results are given in Fig. 6.

The void porosity converges fast with an increasing grid, and even a small number of grid points provides good results. For the presented example, a grid point density of 5 points/Å is sufficient for the void porosity. On the other hand, the accessible porosity converges much slower, and more grid points are needed to reach convergence. Using our example, at least 10 points/Å are needed to sufficiently converge the results. This is true for the other MOFs as well. Furthermore, using different probe radii influences the convergence as well. It should also be noted that the accessible porosity for the different probe radii is very different, which will be discussed in the next section in more detail.

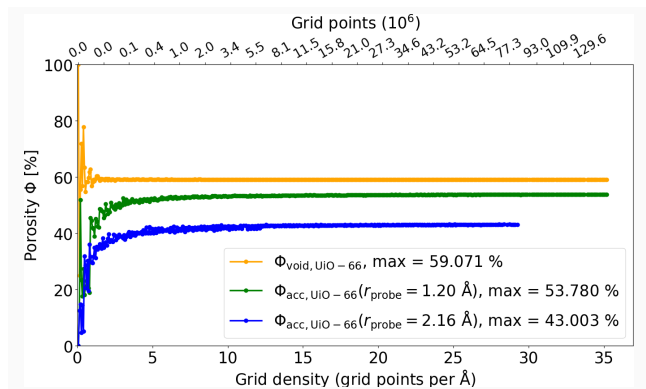


Figure 6. Grid size dependence for UiO-66. The void porosity as well as the accessible porosity for two different probe radii are shown. The x-axes denote the used grid.

### Probe radius dependence

After establishing that a grid size of ca. 10 points/Å is sufficient for an accurate description of the porosities, the next question is how the accessible porosity changes for different probe radii. This becomes especially important if the porosity is analyzed with respect to different adsorbed species, having different effective probe radii. Furthermore, it is important to analyze this behavior regarding the comparison with literature values. Usually, the accessible porosity is reported, while the probe radius is disregarded. Here we show that the accessible porosity strongly depends on the probe radius, which can be seen for the porE8 benchmark set in Fig. 7. Individual pictures and values for each MOF are given in the supplemental material.

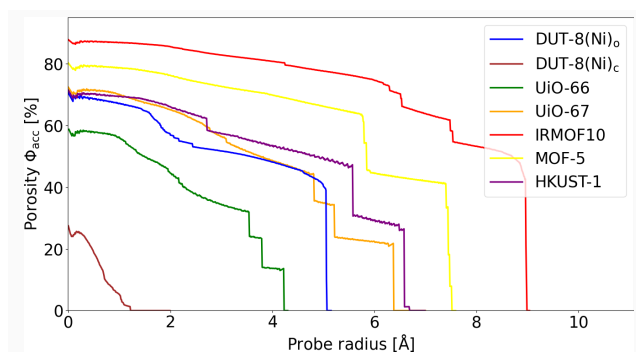


Figure 7. porE8: Probe radius dependence for MOFs in the porE8 benchmark set. The accessible porosity is given on the y-axis, while the probe radius is shown on the x-axis. The used grid is approximately 10 points/Å for all MOFs besides UiO-66 (12.5 points/Å) and DUT-8(Ni)<sub>c</sub> (20 points/Å). The drops in the accessible porosity correspond to different pore sizes/radii.

Given the results of Fig. 7, it is obvious that different species have different accessible porosities, because of their different probe radii. For example, one can compare

the effective probe radii of Xe (2.16 Å), CH<sub>4</sub> (≈ 2.29 Å), SO<sub>4</sub><sup>2-</sup> (≈ 3.01 Å) and C<sub>2</sub>H<sub>6</sub> (≈ 5.48 Å). These values were determined by using the bond distances and the vdW radii of the atoms. For C<sub>2</sub>H<sub>6</sub>, the distance between the most distant H atoms was used. It should be noted that all molecules are approximated as spheres having some effective probe radius, with the goal to make the comparison and the used probe radii more intuitive to understand. This shall not be interpreted as an accurate approximation for, e.g., adsorption investigations (clearly, C<sub>2</sub>H<sub>6</sub> is not spherical and shape effects become important), but only as a way to analyze the probe radius dependence.

The accessible porosity for the mentioned species will be smaller than for, e.g., H ( $r_{\text{probe}} = 1.20$  Å) or He ( $r_{\text{probe}} = 1.40$  Å). To illustrate this more quantitatively, the accessible porosities for the different probe radii are listed in Tab. 3. The same analysis was done with PLATON<sup>14</sup> and ZEO++<sup>18,37-40</sup> (see supplemental material), and the general trends are consistent between the used codes. As an example, the accessible porosity for UiO-66 behaves as follows: 59.1 % ( $r_{\text{probe}} = 0.00$  Å) to 53.3 % ( $r_{\text{probe}} = 1.20$  Å) to 50.8 % ( $r_{\text{probe}} = 1.40$  Å) to 42.9 % ( $r_{\text{probe}} = 2.16$  Å) to 39.9 % ( $r_{\text{probe}} = 2.29$  Å) to 34.2 % ( $r_{\text{probe}} = 3.01$  Å) to 0.0 % ( $r_{\text{probe}} = 5.48$  Å). It should be noted that the void porosity is recovered for  $r_{\text{probe}} = 0.00$  Å, and that the accessible porosity has to become zero for  $r_{\text{probe}} \rightarrow \infty$ .

Table 3. porE8: Accessible porosities (in %) for MOFs in the porE8 benchmark set depending on different probe radii (between 0.00 and 5.48 Å). The used grid contains ca. 10 points/Å, except for UiO-66 where the grid point density is 12.5 points/Å. Values with an \* indicate that the smallest pore window (see section Pore Windows) in the system is smaller than the probe radius. Thus, the accessible volumes become inaccessible.

MOF	0.00	1.20	1.40	2.16	2.29	3.01	5.48
DUT-8(Ni) <sub>o</sub>	70.5	66.3	65.6	55.7	54.5	51.7	0.0*
UiO-66	59.1	53.3	50.8	42.9*	39.9*	34.2*	0.0*
UiO-67	72.4	68.8	67.8	64.1	63.2	56.4*	23.2*
IRMOF-10	87.8	86.6	86.5	85.0	84.6	82.8	76.4
MOF-5	80.2	78.3	77.9	75.4	75.0	72.6	64.5*
HKUST-1	71.4	69.1	68.9	65.1*	64.6*	57.3*	47.0*

Besides the fundamental importance of the probe radius dependence, one can calculate the porosities for different probe radii and monitor the corresponding accessible porosity. Once the probe radius is larger than a specific pore, all grid points within this pore become inaccessible. Accordingly, the accessible porosity drops significantly. This gives an intuitive way to characterize the pore sizes (see Fig. 7). More information is given in the supplemental material. A more accurate approach to analyze the pore dimensions/ pore size distribution is discussed in the next section.

## PORE SIZE DISTRIBUTION

An approach to determine the pore sizes based on a Monte-Carlo (MC) procedure has been implemented. The strategy is straightforward. At first,  $N_{\text{MC}}$  random starting points  $i$  are initialized within the unit cell, such that

$$\mathbf{r}_i = \alpha \cdot \mathbf{a} + \beta \cdot \mathbf{b} + \gamma \cdot \mathbf{c}. \quad (20)$$

Here,  $\alpha, \beta, \gamma \in [0.1, 0.9]$  are random numbers and  $\mathbf{a}$ ,  $\mathbf{b}$  and  $\mathbf{c}$  are the unit cell vectors. With that, the initial point is somewhere inside the unit cell (maybe even inside an atom). After this initialization,  $M_{\text{MC}}$  Monte-Carlo steps are carried out to move these initial points like

$$x_{i+1} = x_i + (2 \cdot \delta - 1) \cdot a_{\text{step}} \quad (21)$$

$$y_{i+1} = y_i + (2 \cdot \epsilon - 1) \cdot a_{\text{step}} \quad (22)$$

$$z_{i+1} = z_i + (2 \cdot \zeta - 1) \cdot a_{\text{step}} \quad (23)$$

for the  $x, y$  and  $z$  components. The random numbers  $\delta, \epsilon$  and  $\zeta$  are all  $\in [0, 1]$  and  $a_{\text{step}}$  is the step size. After moving a point, the minimal distance to the vdW surface

$$d_{i+1} = \min_A \left( \sqrt{\Delta_x + \Delta_y + \Delta_z} - r_{\text{vdW},A} \right), \quad (24)$$

$$\text{with } \Delta_x = (x_{i+1} - x_A)^2 \quad (25)$$

$$\Delta_y = (y_{i+1} - y_A)^2 \quad (26)$$

$$\Delta_z = (z_{i+1} - z_A)^2, \quad (27)$$

is computed. Here,  $r_{\text{vdW},A}$  is the vdW radius of atom  $A$ . Periodic boundary conditions are taken into account. If  $d_{i+1} < d_i$ , the new point is reset ( $\mathbf{r}_{i+1} = \mathbf{r}_i$ ). On the other hand, if  $d_{i+1} > d_i$ , the new point is kept ( $\mathbf{r}_i = \mathbf{r}_{i+1}$ ). Then, the next MC step is carried out. With that,  $d$  is maximized. This procedure is done for all  $N_{\text{MC}}$  initial points until the maximum number of MC steps  $M_{\text{MC}}$  is reached. Several initial points ensure that different pores inside a MOF can be analyzed. This gives information about the pore size distribution (PSD), simply by analyzing how many starting points end up in the same pore (i.e., bigger pores have more, smaller pores have less final points).

To increase efficiency, the step size  $a_{\text{step}}$  is changed throughout the MC cycle, and adapted to the structure under consideration. As a starting point, the length of the largest cell vector  $L_{\text{max}} = \max(L_x, L_y, L_z)$  is taken. The initial step size is defined as  $a_{\text{step}}^{\text{init}} = L_{\text{max}}/10$ , which is adjusted as shown in Tab. 4 and Fig. 8.

This adaptive way of generating the step size has several advantages over a fixed step size. Given that the step size in the beginning of the MC cycle is fairly large, the steps taken are large, too. With that, the points are moved much more quickly towards a nearby maximum, especially when the starting point is far away from any maximum (e.g., if it was initialized inside an atom). By reducing the step size over the course of the MC cycle, the maximum can be approached more and more accurately without the need for a very large number of MC

Table 4. Adaptive step size  $a_{\text{step}}$  with respect to the MC steps  $M_{\text{MC}}$ , given in intervals of fractions of the total number of MC steps.

Fraction of $M_{\text{MC}}$	$a_{\text{step}}/a_{\text{step}}^{\text{init}}$
[0.00,0.25]	1.000
[0.25,0.50]	0.100
[0.50,0.75]	0.010
[0.75,1.00]	0.001

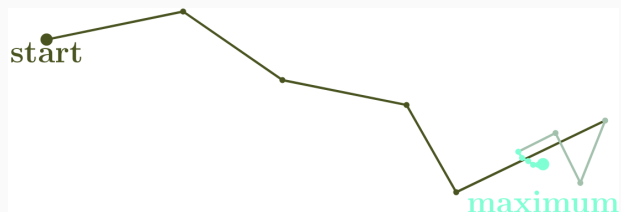


Figure 8. Illustration of the adaptive step size used in the calculation of the pore size distribution. The aquamarine dot is the maximum, while the Army green dot indicates the starting point. Three different step sizes are shown for simplicity. The Army green lines indicate the largest step size, the Cambridge blue ones refer to the next smaller step size and the aquamarine lines correspond to the smallest step size. With this scheme, accurate results are obtained using a small number of steps. This is not possible by only using one of these step sizes.

steps (as would be required for a fixed  $a_{\text{step}}$ ). For the last steps, the position of the pore centers as well as the pore diameters are determined very accurately (typically, the final values are within  $10^{-3}$  Å of the theoretical maximum. This can be checked for structures where the pore centers can be determined by symmetry arguments, like in the UiOs). With this ansatz, only a small number of MC steps (usually about 1000) is needed to properly converge the results. For bigger unit cells one should consider using larger  $M_{\text{MC}}$  to avoid having insufficient MC steps to reach a maximum. The PSDs for the considered MOFs are summarized in Tab. 5 and Fig. 9.

For MOF-210, the first two sizes correspond to distinguishable pores in the MOF structure. The other two share an elongated pore, which is why they are shown together in Tab. 5. Also, the larger one of those two pore sizes (27.98 Å) has two symmetry-equivalent positions around the smaller one (27.65 Å, which sits exactly at the center of the unit cell). With that, the distribution of the larger one is about double that of the smaller one. A figure of these pores is given in the supplemental material.

It should be noted that only symmetry-in-equivalent pore centers positions (as determined by the similarity of their pore sizes) are stored for the final evaluation of the PSD. These can be used to visualize the pores inside the frameworks (see Fig. 1).

Table 5. porE8: Pore size distributions for the porE8 benchmark set, using the pore diameters  $d_{\text{pore}}$  (in Å) and their distribution  $\Gamma$  (in %) evaluated from the Monte-Carlo procedure described in the text. For the distribution, every pore size with less than 5 % has been disregarded. Accordingly,  $\sum_i \Gamma_i$  might not add up to 100 %. For these values,  $N_{\text{MC}} = 200$  and  $M_{\text{MC}} = 2000$ , except for MOF-210 where  $M_{\text{MC}} = 10000$ .

MOF	$d_{\text{pore}}^1$	$\Gamma$	$d_{\text{pore}}^2$	$\Gamma$	$d_{\text{pore}}^3$	$\Gamma$
DUT-8(Ni) <sub>o</sub>	10.18	95.5				
UiO-66	7.19	23.0	7.71	26.5	8.44	50.5
UiO-67	9.63	16.0	10.42	17.5	12.75	66.5
IRMOF-10	18.02	99.5				
MOF-5	11.82	28.0	15.10	72.0		
HKUST-1	5.54	17.0	11.15	46.0	13.37	37.0
MOF-210	19.58	20.0	19.97	11.5	27.65	20.5
					27.98	40.0

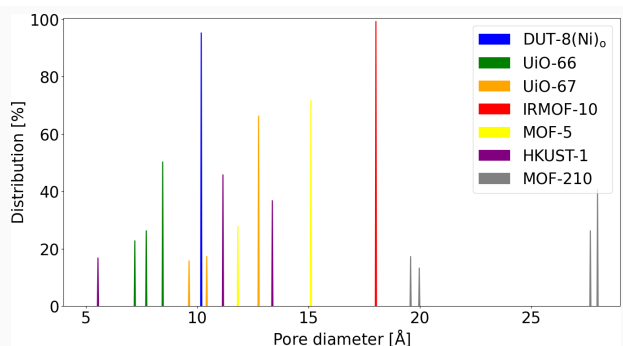


Figure 9. porE8: Pore size distributions (PSDs) for MOFs in the porE8 benchmark set. Using  $N_{\text{MC}} = 200$  and  $M_{\text{MC}} = 2000$ , except for MOF-210 where  $M_{\text{MC}} = 10000$ , see text for details.

## PORE WINDOWS

For an accurate determination of the porosity, it is not only important to analyze which regions in a MOF are occupied, void or accessible, and how large the pores are. It is also essential to analyze how large the windows between pores are. The pore window is the largest possible size a species can have to travel through the porous framework. If the species would theoretically fit into a pore but cannot reach it due to a small pore window, the pore itself has unoccupied volume which is, however, inaccessible. To analyze the pore window, the following ansatz is chosen. First, the PSD (see last section) is computed. With that, the centers of all pores within a MOF are known. Then, one draws a line in between all pores (taking periodic boundary conditions into account). The smallest distance of this line to the vdW surface is characterizing a minimum radius  $r_{\text{min}}$  between two pores. This approach is summarized in Fig. 10 using UiO-66.

An additional characteristic of a pore window is that it lies in between two pores. With that, if the distance between the coordinate of  $r_{\text{min}}$  and the pore centers,



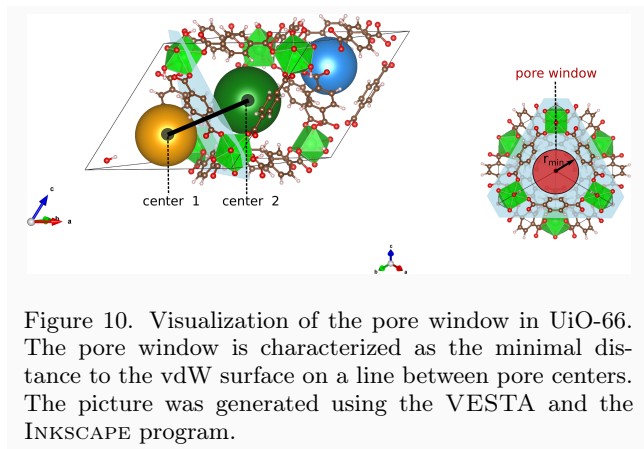


Figure 10. Visualization of the pore window in UiO-66. The pore window is characterized as the minimal distance to the vdW surface on a line between pore centers. The picture was generated using the VESTA and the INKSCAPE program.

$\min(d_{\text{center}})$ , is very different from the respective pore sizes  $d_{\text{pore}}$  (i.e.,  $|\min(d_{\text{center}}) - d_{\text{pore}}|/d_{\text{pore}} > 0.3$ ), the  $r_{\text{min}}$  does not represent a pore window. This corresponds to a case where the position of  $r_{\text{min}}$  lies almost completely within one pore, not between two pores. For example, in UiO-68 one initially finds four  $r_{\text{min}}$  with 1.89 Å, 2.06 Å, 4.22 Å and 4.69 Å. However, the first two do not characterize as pore windows and need to be excluded. Thus, the actual pore windows in UiO-68 are 4.22 Å and 4.69 Å.

By employing this approach, the pore windows for the considered MOFs were evaluated. The smallest one is the limiting pore window. If the probe radius is larger than this limiting pore window, the porosity/volume that is evaluated to be accessible becomes inaccessible. For more complex pore structures, this approach might not be sufficiently accurate. As this is the case for MOF-210, it is not shown in Tab. 6. An extension is planned for a future version of the code.

To justify our approach, the calculated pore windows are compared to ZEO++<sup>18,37-40</sup> and PYWINDOW<sup>41</sup>. We find consistent results, which are summarized in Tab. 6.

Table 6. porE8: Pore windows (in Å) for the given MOFs, including a comparison to ZEO++<sup>18,37-40</sup> and PYWINDOW<sup>41</sup>.

MOF	PORE: $r_1$	PORE: $r_2$	ZEO++	PYWINDOW
DUT-8(Ni) <sub>o</sub>	4.05		4.60	4.80
UiO-66	1.84	2.06	2.06	2.09
UiO-67	2.59	2.91	3.02	3.05
IRMOF-10	5.88		6.09	5.67
MOF-5	3.91		3.96	3.97
HKUST-1	1.84	3.21	3.32	3.33

While there is only one pore window for DUT-8(Ni)<sub>o</sub>, IRMOF-10 and MOF-5, there are two different pore windows in all UiOs and HKUST-1. The smaller one in each MOF represent the limiting pore window.

The presented ansatz is computationally efficient, and only requires an accurate PSD. The PSD only needs to be computed once per structure. Once the PSD is known

the pore windows, and accordingly the porosities, can be analyzed with respect to different grids.

## BENCHMARK SETS

### Verification benchmark: porE8

To further validate the implementations in PORE, several reference calculations were performed for the porE8 benchmark set. For this, we used the codes RASPA2<sup>16,17</sup>, POREBLAZER<sup>15</sup>, PLATON<sup>14</sup> and ZEO++<sup>18,37-40</sup> (see Tab. 7). The used unit cells contain the following amount of atoms: DUT-8(Ni)<sub>open</sub> (132), DUT-8(Ni)<sub>closed</sub> (66), UiO-66 (114), UiO-67 (174), HKUST-1 (156), IRMOF-10 (166), MOF-5 (424) and MOF-210 (1854). All structures are available at <https://github.com/kaitrepte/porE>.

Table 7. porE8: Comparison of calculated porosities between reference codes (RASPA2<sup>16,17</sup> ( $\pi$ ), POREBLAZER<sup>15</sup> ( $\rho$ ), PLATON<sup>14</sup> ( $\gamma$ ), ZEO++<sup>18,37-40</sup> ( $\xi$ )), literature values and the presented approaches (HEA ( $\eta$ ), OSA ( $\alpha$ ), GPA(void) ( $\omega_{\text{void}}$ ), and GPA(acc) ( $\omega_{\text{acc}}$ )). For the GPA,  $r_{\text{probe}} = 1.20$  Å using the largest possible grids; for MOF-210 we used  $\bar{n} = 5$  points/Å. All porosities are given in %. The literature values (REF) are: DUT-8(Ni)<sup>23</sup>, UiO-66 and UiO-67<sup>42</sup>, IRMOF-10<sup>26</sup>, MOF5 and MOF-210<sup>31</sup>, HKUST-1<sup>12,43</sup>.

MOF	$\pi$	$\rho$	$\gamma$	$\xi$	$\eta$	$\alpha$	$\omega_{\text{void}}$	$\omega_{\text{acc}}$	$\Phi_{\text{REF}}$
DUT-8(Ni) <sub>o</sub>	70	68	66	68	70	61	70	67	67
DUT-8(Ni) <sub>c</sub>	0	0	0	0	25	-1	27	1	0
UiO-66	52	51	53	55	63	49	59	54	53
UiO-67	72	72	69	70	68	68	72	69	68
IRMOF-10	91	90	86	87	85	88	88	87	87
MOF-5	81	81	76	79	77	77	80	79	79
HKUST-1	73	72	69	69	72	68	71	69	68
MOF-210	93	—	—	88	87	90	90	89	89

The RASPA2 code provides various features for the calculation of porous materials. The porosity can be accessed using the calculation mode for the determination of the helium void fraction. For that, the porous structure is probed with a helium atom at room temperature. The helium atom itself is described with a TraPPE force field<sup>16,17,44</sup>, while the porous structure can be described with various available force fields. For our reference calculations, we used the CrystalGenerator force field<sup>16,17</sup>, 2000 Monte Carlo cycles and the unit cell of the MOFs. The PLATON code offers several analysis techniques for crystal structures. It allows to analyze the accessible porosity using a grid based approach, similar to the GPA presented here. The POREBLAZER code can be used to analyze the surface areas, the pore size distribution and the porosity. In POREBLAZER, the porous system is described using the universal force field (UFF<sup>45</sup>) and the helium atoms are describe using a Lennard-Jones force field description. The (helium) void volume is calculated

using a cublet procedure. In ZEO++, the accessible volume is calculated by placing points randomly in the unit cell. Afterwards, each point is analyzed regarding its accessibility with respect to a given probe radius using a Voronoi decomposition scheme<sup>18</sup>.

In general, the void porosities using PORE with the GPA agree well with results of the RASPA2 and the POREBLAZER code (see Tab. 7). The accessible porosities applying PORE with the GPA is in excellent agreement with results calculated with the PLATON and the ZEO++ codes (see Tab. 7). Further, the comparison to literature values shows that accessible porosities are reported, while the corresponding probe radius is usually not provided. Furthermore, void porosities do not necessarily reflect the porous nature of the MOFs. This is especially true for systems with more complex pores, where the probe radius plays a crucial role (compare, e.g., UiO-66 and MOF-5). In MOFs with large, open pores the void porosity will already reflect the porous nature of the MOF fairly accurately. If a pore would be entirely spherical, the calculated porosity would be independent of the probe radius (unless  $r_{\text{probe}} \geq r_{\text{pore}}$ ). Using any probe smaller than the pore radius would sample the entire pore, as there are no areas/volumes which are inaccessible. With that, the void and accessible porosities coincide.

However, if the pores are different from the spherical symmetry, e.g., having tetrahedral or octahedral symmetries as in the UiOs, the probe radius dependence becomes significant. In such cases, a sphere is not a suitable approximation for the pores (see Fig. 1 for UiO-66 and UiO-67). Quantitatively, this can be seen for UiO-66 (see Tab. 7), where the void and the accessible porosities are quite different (59 % and 54 % for a probe radius of 1.20 Å). These differences become more pronounced the larger the probe radius becomes. This also explains the rather small dependence of the accessible porosity for different  $r_{\text{probe}}$  in IRMOF-10, MOF-5 and HKUST-1 (see Tab. 3 and the supplemental material), as spheres are already a better approximation to describe the respective pores for these three MOFs.

In summary, the accessible porosities from the GPA are in excellent agreement with literature values, reassuring a proper implementation. Given the geometric, deterministic approach of HEA, it is a very efficient way to compute void porosities. Considering its limitations, even the OSA provides reasonable void porosities.

## Extended benchmark: porE370

To obtain reference values for porE370, which has been introduced earlier, we performed calculations with RASPA2. We also calculated porosities using all presented approaches (HEA, OSA, GPA(void), and GPA(acc)). As a note, for the GPA calculation a grid density of 10 points/Å was used. The resulting porosities show a linear relationship to the RASPA2 porosities

(see Fig. 11). The quality of a linear regression is often compared with the  $R^2$  error

$$e_{\text{tot}} = \sum_i \left( y_{\text{ref},i} - 1/N \sum_j y_{\text{ref},j} \right)^2 \quad (28)$$

$$e_{\text{res}} = \left( \sum_i y_{\text{ref},i} - y_{\text{calc},i} \right)^2 \quad (29)$$

$$R^2 = 1 - e_{\text{res}}/e_{\text{tot}}, \quad (30)$$

where  $R^2 = 1$  indicates a perfect linear relationship, while smaller  $R^2$  indicate less linear relationships. In addition to the  $R^2$  we report mean errors

$$\text{ME} = 1/N \sum_i y_{\text{ref},i} - y_{\text{calc},i}. \quad (31)$$

Porosities from GPA(acc) show the highest degree of linearity with respect to the RASPA2 porosities,  $R^2 = 0.94$ . This is followed by GPA(void) with  $R^2 = 0.90$  and HEA with  $R^2 = 0.82$ . OSA shows the least degree of linearity with  $R^2 = 0.70$ . On the other hand, the order from lowest to largest mean error is ME(GPA(acc)) = 0.29 %, ME(HEA) = 3.26 %, ME(OSA) = 4.54 %, and ME(GPA(void)) = 13.59 %.

For large porosities, i.e.,  $\Phi > 60$  %, all of our proposed approaches work similarly well. As explained above, this is given by the fact that under these circumstances the void porosities become a more sufficiently accurate approximation for the accessible porosities. In contrast, for smaller porosities we recommend more accurate approaches like GPA(acc).

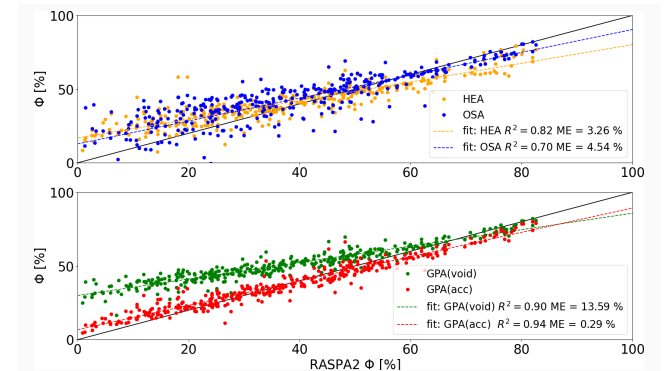


Figure 11. porE370: Results for HEA, OSA, GPA(void), and GPA(acc) given on the y-axis compared to RASPA2 He-void fraction results along the x-axis. For the linear regression fits the PYTHON module scikit-learn (sklearn)<sup>46</sup> was used.

Given the linear relationships to RASPA2 porosities, we compared and fitted OSA, GPA(void), and GPA(acc) against HEA and also found linear relationships for the porE370 benchmark set. HEA was chosen as origin because it presents the best approach combining numerical

efficiency and accuracy. Given the linear relationships between HEA and the other approaches, one can derive corrections to HEA resulting in OSA, GPA(void) and GPA(acc) accuracy

$$\Phi_{\text{HEA@OSA}} = 1.0879\Phi_{\text{HEA}} - 2.2688 \% \quad (32)$$

$$\Phi_{\text{HEA@GPA(void)}} = 0.7769\Phi_{\text{HEA}} + 19.3323 \% \quad (33)$$

$$\Phi_{\text{HEA@GPA(acc)}} = 1.1388\Phi_{\text{HEA}} - 8.5709 \%. \quad (34)$$

We compare our HEA corrections to RASPA2 results (see Fig. 12). While the  $R^2$  stay the same,  $R^2 = 0.82$ , the ME is now the same as the respective approach for which the corrections was derived.

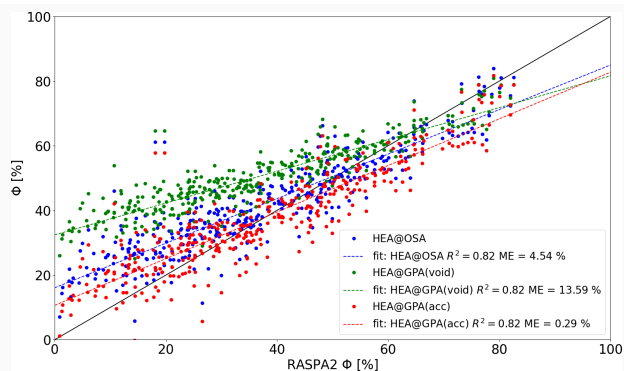


Figure 12. porE370: Results for  $\Phi_{\text{HEA@OSA}}$ ,  $\Phi_{\text{HEA@GPA(void)}}$ , and  $\Phi_{\text{HEA@GPA(acc)}}$  given on the y-axis compared to RASPA2 He-void fraction results along the x-axis. For the linear regression the PYTHON module scikit-learn (sklearn)<sup>46</sup> was used.

For the analysis of larger databases, these cheap and simple corrections may be helpful to efficiently determine trends. In the future, more advanced models may be constructed using, e.g., TENSORFLOW<sup>47</sup>. The PORE PYTHON file for the benchmark sets shown below has less than 30 lines of code (see Fig. 13). This illustrates the simplicity of building complex workflows in PORE with a few lines of PYTHON code.

In addition to calculating the porosities, the PSD has been calculated for all MOFs in the porE370 test set. Here, we used  $N_{\text{MC}} = 200$  and  $M_{\text{MC}} = 10000$ , see Fig. 14. Clearly, MOFs with one and two different pore sizes are most common, followed by 3 pore sizes. This is also true for the porE8 test set. MOFs with more pores are less common. The maximum pore sizes vary between 2 and 16 Å, but the most common maximum pore sizes are between 3.5 and 5.5 Å.

## CONCLUSION

We present three approaches to analyze the porosity in porous materials, in specific for metal-organic frameworks (MOFs). Various MOFs - with detailed analyses for DUT-8(Ni)<sub>open</sub>, DUT-8(Ni)<sub>closed</sub>, UiO-66, UiO-67, IRMOF-10, MOF-5, HKUST-1, and MOF-210, as

```
from pore import porosity as p
from porE.heal.HEA import HEA
from porE.io.ase2pore import *
import glob
# Collect cifs
cifs = glob.glob('*.cif')
cifs.sort()
f_out = 'benchmark.out'
with open(f_out, 'w', buffering=10) as o:
    o.write('f HEA OSA GPAvoid GPAacc\n')
    for cif in cifs:
        # Perform HEA
        r1 = HEA(cif)
        # Transform cif into porE format
        ase2pore(cif)
        xyz = 'pypore.xyz'
        # Perform OSA
        r2 = p.osa(xyz)
        # Perform GPA
        probeR = 1.2
        n = 10.0
        r3 = p.gpa_gridpera(xyz, probeR, n)
        # Collect results
        r = [r1[0], r2[0], r3[0], r3[1]]
        s = ' '.join([str(ri) for ri in r])
        o.write(s)
```

Figure 13. PYTHON file for the PORE porosity calculations for the porE8 and porE370 benchmark sets. As a note, HEA and OSA only depend on the structure, while the GPA also depends on a probe radius and the used grid.

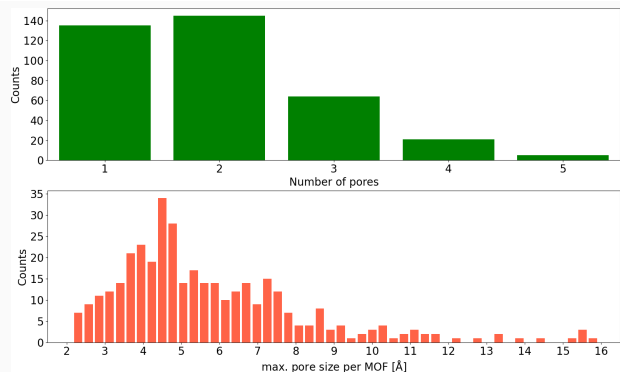


Figure 14. porE370: PSD evaluation for the upper plot shows occurrences of the number of pores, whereas in the lower plot we show the occurrence of the maximum pore size per MOFs within the porE370 benchmark. For this evaluation we used  $N_{\text{MC}} = 200$  and  $M_{\text{MC}} = 10000$ . Clearly, one and two pores have the highest counts within the test set, and the maximum pore sizes are centered around 4.5 Å.

well as additional analyses for 370 other MOFs - have been studied using these approaches.

The first approach, HEA, is based on a cell list approach to efficiently compute the He void fraction. Another approach employs the overlap of atomic spheres (OSA) to compute the void porosity, which is also very

efficient. The last approach uses a grid inside the unit cell (GPA), allowing for a clear differentiation between void and accessible porosities. It was shown that it is essential to converge the results with respect to the used grid. In contrast to commonly used implementations, all approaches are deterministic and do not require any stochastic methodologies, making them fully reproducible. A comparison to reference calculations and literature values confirms that all approaches work as intended, and that the GPA gives very accurate results.

Further, the crucial importance of a clear differentiation between accessible and void porosity for the correct description of porosities in MOFs was demonstrated. For the accessible porosity, it was shown that a correct treatment of the probe radius is essential for reliable results. Thus, this dependence must be considered in any case. In addition, the pore size distribution and the pore windows are analyzed. This is another critical aspect for an accurate determination of the porosities.

The presented PORE code offers many functionalities, like a graphical user interface (GUI), the ability to read .cif and other structure information files via ASE, the general user-friendliness of the Python programming language, and the ability to change any parameter that influences the results right away. Due to the PYTHON interface complex workflows are easily implemented, including reading and writing structures with ASE, the calculation of porosities and design of a useful output format for larger benchmark sets like poreE370 with PORE, data analysis with NumPy, visualization with matplotlib and building of simple machine learning models using scikit-learn (sklearn)<sup>46</sup> (see Fig. 15)

## DATA AVAILABILITY

The PORE code, presented and used within this study, is openly available on GITHUB (<https://github.com/kaitrepte/pore>), and can be referenced via <http://doi.org/10.5281/zenodo.3973264>.

## ACKNOWLEDGMENTS

The authors thank the ZIH in Dresden and the HPCC at MSU for computational time and support.

- <sup>1</sup>S. Yuan, L. Feng, K. Wang, J. Pang, M. Bosch, C. Lollar, Y. Sun, J. Qin, X. Yang, P. Zhang, et al., *Advanced Materials* **30**, 1704303 (2018).
- <sup>2</sup>J.-R. Li, R. J. Kuppler, and H.-C. Zhou, *Chemical Society Reviews* **38**, 1477 (2009).
- <sup>3</sup>H. Li, K. Wang, Y. Sun, C. T. Lollar, J. Li, and H.-C. Zhou, *Materials Today* **21**, 108 (2018).
- <sup>4</sup>F. X. L. i Xamena, A. Abad, A. Corma, and H. Garcia, *Journal of Catalysis* **250**, 294 (2007), ISSN 0021-9517.
- <sup>5</sup>L. Jiao, Y. Wang, H.-L. Jiang, and Q. Xu, *Advanced Materials* **30**, 1703663 (2018).
- <sup>6</sup>F.-L. Li, Q. Shao, X. Huang, and J.-P. Lang, *Angewandte Chemie International Edition* **57**, 1888 (2018).

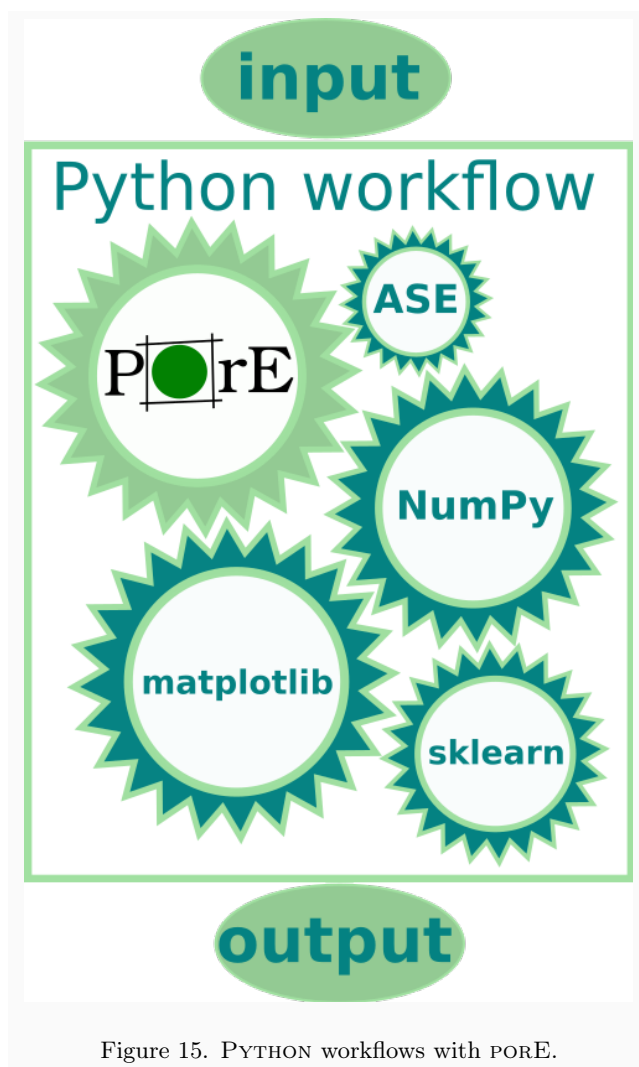


Figure 15. PYTHON workflows with PORE.

- <sup>7</sup>J. Lei, R. Qian, P. Ling, L. Cui, and H. Ju, *TrAC Trends in Analytical Chemistry* **58**, 71 (2014), ISSN 0165-9936, new Nanobiosensing Techniques and Bioanalysis.
- <sup>8</sup>Y. Zhang, S. Yuan, G. Day, X. Wang, X. Yang, and H.-C. Zhou, *Coordination Chemistry Reviews* **354**, 28 (2018).
- <sup>9</sup>S. M. Cohen, *Chemical Science* **1**, 32 (2010).
- <sup>10</sup>T. D. Bennett and A. K. Cheetham, *Accounts of Chemical Research* **47**, 1555 (2014).
- <sup>11</sup>S. A. Moggach, T. D. Bennett, and A. K. Cheetham, *Angewandte Chemie International Edition* **48**, 7087 (2009).
- <sup>12</sup>D. Ongari, P. G. Boyd, S. Barthel, M. Witman, M. Haranczyk, and B. Smit, *Langmuir* **33**, 14529 (2017).
- <sup>13</sup>A. Bondi, *The Journal of Physical Chemistry* **68**, 441 (1964).
- <sup>14</sup>A. L. Spek, *Acta Crystallographica Section D* **65**, 148 (2009).
- <sup>15</sup>L. Sarkisov and A. Harrison, *Molecular Simulation* **37**, 1248 (2011).
- <sup>16</sup>D. Dubbeldam, A. Torres-Knoop, and K. S. Walton, *Molecular Simulation* **39**, 1253 (2013).
- <sup>17</sup>D. Dubbeldam, S. Calero, D. E. Ellis, and R. Q. Snurr, *Molecular Simulation* **42**, 81 (2016).
- <sup>18</sup>T. F. Willems, C. H. Rycroft, M. Kazi, J. C. Meza, and M. Haranczyk, *Microporous and Mesoporous Materials* **149**, 134 (2012).
- <sup>19</sup>S. Chavan, J. G. Vitillo, D. Gianolio, O. Zavorotynska, B. Civaleri, S. Jakobsen, M. H. Nilsen, L. Valenzano, C. Lamberti, K. P. Lillerud, et al., *Physical Chemistry Chemical Physics* **14**, 1614 (2012).

- (2012).
- <sup>20</sup>K. Treppe, J. Schaber, S. Schwalbe, F. Drache, I. Senkovska, S. Kaskel, J. Kortus, E. Brunner, and G. Seifert, *Physical Chemistry Chemical Physics* **19**, 10020 (2017).
  - <sup>21</sup>K. Treppe, S. Schwalbe, J. Schaber, S. Krause, I. Senkovska, S. Kaskel, E. Brunner, J. Kortus, and G. Seifert, *Physical Chemistry Chemical Physics* **20**, 25039 (2018).
  - <sup>22</sup>Q. Yang, V. Guillermin, F. Ragon, A. D. Wiersum, P. L. Llewellyn, C. Zhong, T. Devic, C. Serre, and G. Maurin, *Chemical Communications* **48**, 9831 (2012).
  - <sup>23</sup>V. Bon, N. Klein, I. Senkovska, A. Heerwig, J. Getzschmann, D. Wallacher, I. Zizak, M. Brzhezinskaya, U. Mueller, and S. Kaskel, *Physical Chemistry Chemical Physics* **17**, 17471 (2015).
  - <sup>24</sup>K. Treppe, S. Schwalbe, and G. Seifert, *Physical Chemistry Chemical Physics* **17**, 17122 (2015).
  - <sup>25</sup>S. Schwalbe, K. Treppe, G. Seifert, and J. Kortus, *Physical Chemistry Chemical Physics* **18**, 8075 (2016).
  - <sup>26</sup>M. Eddaoudi, J. Kim, N. Rosi, D. Vodak, J. Wachter, M. O’Keeffe, and O. M. Yaghi, *Science* **295**, 469 (2002).
  - <sup>27</sup>N. L. Rosi, J. Eckert, M. Eddaoudi, D. T. Vodak, J. Kim, M. O’Keeffe, and O. M. Yaghi, *Science* **300**, 1127 (2003).
  - <sup>28</sup>S. S.-Y. Chui, S. M.-F. Lo, J. P. H. Charmant, A. G. Orpen, and I. D. Williams, *Science* **283**, 1148 (1999).
  - <sup>29</sup>C. H. Hendon and A. Walsh, *Chemical Science* **6**, 3674 (2015).
  - <sup>30</sup>K. T. Butler, C. H. Hendon, and A. Walsh, *Journal of the American Chemical Society* **136**, 2703 (2014).
  - <sup>31</sup>H. Furukawa, N. Ko, Y. B. Go, N. Aratani, S. B. Choi, E. Choi, A. Ö. Yazaydin, R. Q. Snurr, M. O’Keeffe, J. Kim, et al., *Science* **329**, 424 (2010).
  - <sup>32</sup>DOI: 10.5281/zenodo.3986569 (2016).
  - <sup>33</sup>D. Nazarian, J. S. Camp, Y. G. Chung, R. Q. Snurr, and D. S. Sholl, *Chemistry of Materials* **29**, 2521 (2017).
  - <sup>34</sup>S. Barthel, E. V. Alexandrov, D. M. Proserpio, and B. Smit, *Crystal Growth & Design* **18**, 1738 (2018).
  - <sup>35</sup>K. D. Gibson and H. A. Scheraga, *Molecular Physics* **62**, 1247 (1987).
  - <sup>36</sup>R. D. Johnson, *NIST Computational Chemistry Comparison and Benchmark Database* (NIST Standard Reference Database Number 101, 19–04–2018), URL <http://cccbdb.nist.gov/>.
  - <sup>37</sup>R. L. Martin, B. Smit, and M. Haranczyk, *Journal of Chemical Information and Modeling* **52**, 308 (2012).
  - <sup>38</sup>M. Pinheiro, R. L. Martin, C. H. Rycroft, A. Jones, E. Iglesia, and M. Haranczyk, *Journal of Molecular Graphics and Modelling* **44**, 208 (2013).
  - <sup>39</sup>M. Pinheiro, R. L. Martin, C. H. Rycroft, and M. Haranczyk, *CrystEngComm* **15**, 7531 (2013).
  - <sup>40</sup>R. L. Martin and M. Haranczyk, *Crystal Growth & Design* **14**, 2431 (2014).
  - <sup>41</sup>M. Miklitz and K. E. Jelfs, *Journal of Chemical Information and Modeling* **58**, 2387 (2018).
  - <sup>42</sup>H. Wu, T. Yildirim, and W. Zhou, *The Journal of Physical Chemistry Letters* **4**, 925 (2013).
  - <sup>43</sup>J. A. Mason, M. Veenstra, and J. R. Long, *Chemical Science* **5**, 32 (2014).
  - <sup>44</sup>M. G. Martin and J. I. Siepmann, *The Journal of Physical Chemistry B* **102**, 2569 (1998).
  - <sup>45</sup>A. K. Rappe, C. J. Casewit, K. S. Colwell, W. A. Goddard, and W. M. Skiff, *Journal of the American Chemical Society* **114**, 10024 (1992).
  - <sup>46</sup>F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., *Journal of Machine Learning Research* **12**, 2825 (2011).
  - <sup>47</sup>M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al., *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*, URL = <https://www.tensorflow.org/> (2015), software available from [tensorflow.org](https://www.tensorflow.org/), URL <https://www.tensorflow.org/>.