PDE-based analysis of void space of porous materials on multicore CPUs

Richard L. Martin, Maciej Haranczyk, Prabhat and James A. Sethian

Computational Research Division, Lawrence Berkeley National Laboratory, Berkeley, California, USA. Contact: mharanczyk@lbl.gov

Introduction

Porous materials such as zeolites and metal organic frameworks have been of growing importance as materials for energy-related applications such as CO_2 capture, hydrogen storage, and as catalysts for oil refinement. Very large databases of these structures are being developed, and so there is a requirement for methods to screen databases - which can contain millions of entries - to discover materials with certain properties important to these applications. However, without automated tools, for example for the analysis of void space within the materials, researchers are restricted to visual analysis and simulation of individual structures.

We have developed a partial differential equation (PDE) based tool for the automated analysis of void space, which utilizes the Fast Marching Method (FMM). Our tool can predict whether a guest molecule can traverse a structure, and determine the size and shape of accessible and inaccessible regions and calculate accessible volumes and surfaces. Our tool provides functionality which is important for characterization of porous materials, particularly in a database screening step, prior to molecular dynamics (MD) simulations.

It is also important to limit the space within each structure which must be explored in an MD simulation, such that regions of void space which are inaccessible (i.e. pockets) are not explored. It is common to exclude these regions using 'blocking spheres'; our method determines the size and position of these spheres by analyzing the periodic connectivity of distinct marches through the void space.

Characterizing porous material void space

Porous materials are characterized through analysis of their void space; this is achieved by attempting to move some guest molecule through the material. The disconnected regions of accessible space found are then examined to find pockets and channels. Unlike other tools of this kind, we can model the guest molecule not as a single sphere but as a probe built from multiple spheres (atoms, separated by some bond length). Then, by simulating the movement of this probe inside the structure we incorporate three translational degrees of freedom (as in the case of a single spherical probe), but also up to three rotational degrees of freedom (Figures 1 & 2).

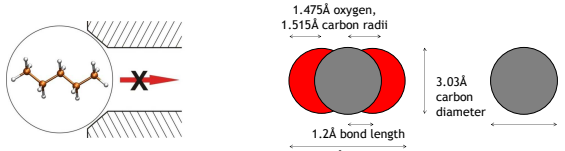
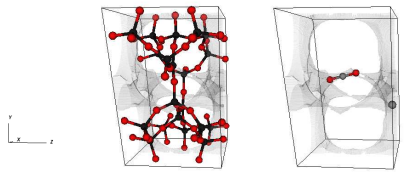


Figure 1. Representing a molecule by a spherical probe which encompasses it (left) leads to overestimation of its size and misjudgment of the ability of the molecule to enter or traverse a structure. Representing a molecule - e.g. CO_2 (center) - by a spherical probe of diameter equal to the minimum width of the molecule itself (right) gives an estimation of the molecule's ability to traverse the structure, but does not account for positions where the molecule would need to rotate to successfully navigate.

Figure 2. The zeolite WEN and its void space network (left). The CO_2 minimum sphere can traverse the void space in the *z* direction, but CO_2 itself gets stuck (right).



Overview of method

A probe is navigated through a structure using the Fast Marching Method. Fast Marching is an ordered upwind one-pass Dijkstra-like method to compute the optimal path through a region by solving an associated Eikonal equation for the cost function. We approximate the Eikonal equation:

$$|\nabla U| = F(x)$$

where $F(x)$ is the given cost function at point x in the domain. We replace the gradient by an upwind approximant of the form:

$$\begin{bmatrix} \max(D_{ij}^{-x}u, -D_{ij}^{+x}u, 0) + \\ \max(D_{ij}^{-y}u, -D_{ij}^{+y}u, 0) \end{bmatrix}^{1/2} = f_{ij}$$

where $\pm x/y$ represent neighboring points in each direction along the x and y axes; for higher dimensionality more such terms are used.

The process of analyzing a structure takes place in three main steps.

Step 1: Distance grid calculation

Prior to exploring void space, the distance to the nearest atom of the structure for each position of the probe in a discrete grid is required. If any distance is not smaller than the sum of the atomic radii in question, then that position is accessible. If the probe molecule consists of multiple atoms, as in the case of CO_2, then the minimum distance for any of the constituent atoms is selected. The distance grid alone does not directly convey any information about the navigability or otherwise of the system. Once the grid is determined however, Fast Marching can take place.

Step 2: Fast Marching

The Fast Marching loop begins by selecting some unvisited accessible point, and iteratively marches outwards, until all connected accessible points are visited (Figure 3). If unvisited accessible points remain, this loop repeats. The result is a set of disconnected marches (Figure 4).

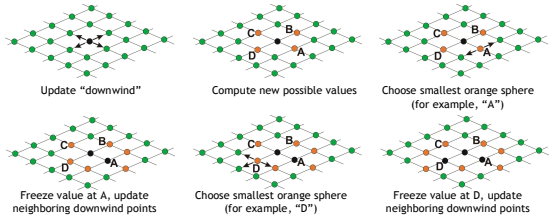


Figure 3. Fast Marching on an example two-dimensional grid.

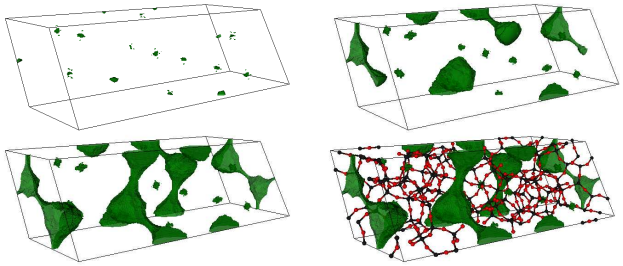


Figure 4. The result of Fast Marching on the zeolite DDR. Each distinct march begins at a different position in the material's void space, and expands as far as possible. The resolution may be such that some pockets are represented as many small pockets (upper left). The structure itself is also shown (lower right).

Step 3: Pocket blocking

Given some quantity of distinct marches, their periodic connectivity can be analyzed to determine which marches constitute pockets and which channels (Figure 5).

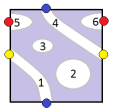


Figure 5. Abstract representation of the periodic connectivity of marches in an example two-dimensional system. Features are created at the centroids of connected regions on some face, represented here as colored circles. Marches 2 and 3 are pockets due to having no associated features. Marches 1 and 4 create a periodic loop, and so together form a channel. Marches 5 and 6 create a periodic pocket.

With knowledge of which marches are pockets, the pockets can be iteratively blocked with exclusion spheres. The optimal radii for spheres centered at the most dense point in each pocket are calculated, such that no point in a channel is excluded (Figure 6). Some pockets - due to their shape or the proximity of channels - require multiple spheres to block.

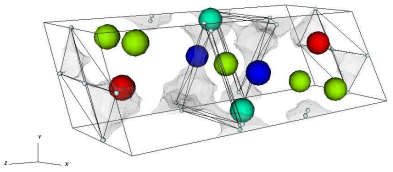


Figure 6. The zeolite DDR has its pockets excluded by spheres. The yellow spheres towards the left and right of the image are excluding pockets which cross periodically in the *x* direction.

Scaling on multicore CPUs

We found that the most time-consuming step in this process is calculating the distance grid to determine whether the guest molecule can access each position of discretized space in each orientation (Figure 7). Since the distance calculation at each grid point is also independent of any other, this part of the process is a natural choice for the application of a multicore method. A specified number of threads are created, and each is given a single coordinate in the *x* axis along which to iterate, calculating distances. Once a thread finishes it is assigned the next unassigned *x* coordinate, until the entire grid has been calculated.

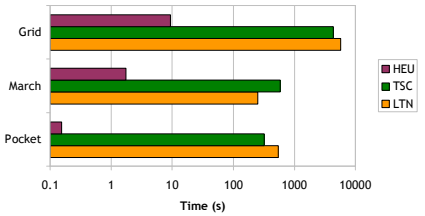


Figure 7. A logarithmic plot of the time in seconds required to calculate the distance grid, process all marches, and block all pockets, for three zeolites of increasing size from the IZA dataset. The times are averages over three runs. As the size of the examined structures increases, the difference in time requirements between the distance grid and any other step approaches one order of magnitude.

It was found that the distance grid computation time scaled almost linearly with respect to the number of threads, up to a total of twenty-four threads (Figure 8), on a quad six-core AMD Opteron (model 6172), 2.1GHz, 48Gb RAM. This is an encouraging result which demonstrates the applicability of the multicore approach to the most time-consuming step in the procedure.

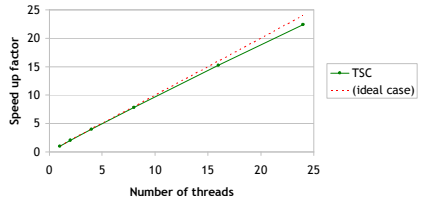


Figure 8. The speed up factor achieved by implementing a multicore distance calculation; only the speed up in the distance grid calculation step is shown. The times are averages over three runs, for a specified number of threads. The time is shown to scale almost linearly with the number of threads, up to twenty-four, the number of cores in the machine. TSC is one of the largest zeolites in the IZA dataset.

Other features of the method

Given the distance grid alone, certain features of the structure can be determined. For instance, the quantity of orientations of a non-spherical probe which are valid at each grid point (Figure 9).

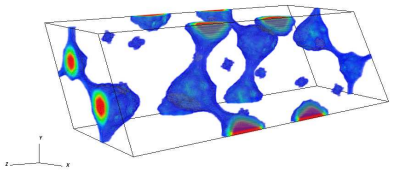


Figure 9. The quantity of discrete orientations of a CO_2 molecule which are valid at each point in a discrete grid for the DDR zeolite can be intuitively represented as a volume rendering. This value quantifies the degree of rotational freedom at each point, which approximates the entropy of the guest molecule.

Future work

The next objective for parallel implementation of this method is to determine the time scaling achievable through a GPU implementation. Additionally, since the pocket blocking step can be considered to be an independent process for each pocket, this step is also a natural choice for parallelization. A similar adaptation of the marching process is less straight forward, however one option is to march independently through distinct subgrids, before combining the marching information.

Acknowledgments

This work was supported by the Director, Office of Advanced Scientific Computing Research, Office of Science, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231 through the Scientific Discovery through Advanced Computing (SciDAC) program's Visualization and Analytics Center for Enabling Technologies (VACET).