

Supplementary information

Tianju Xue,^{*a} Alex Beatson,^b Maurizio Chiaramonte,^{*c} Geoffrey Roeder,^b Jordan T. Ash,^b
Yigit Menguc,^c Sigrid Adriaenssens,^a Ryan P. Adams^b and Sheng Mao^{*d,e}

^a Department of Civil and Environmental Engineering, Princeton University, Princeton, NJ 08544, USA.

^b Department of Computer Science, Princeton University, Princeton, NJ 08544, USA.

^c Facebook Reality Labs, Redmond, WA, 98052, USA. E-mail: mchiaram@fb.com

^d Department of Mechanics and Engineering Science, BIC-ESAT, College of Engineering, Peking University, Beijing 100871, PRC.

^e Department of Mechanical and Aerospace Engineering, Princeton University, Princeton, NJ 08544, USA.

* Corresponding authors: txue@princeton.edu, mchiaram@fb.com, maosheng@pku.edu.cn

1 Mesh refinement study

We conducted a mesh-refinement study here to maximize our simulation speed at a reasonable accuracy. First, we define the characteristic mesh size h as follows:

$$h = \sqrt{\frac{\phi_0 A}{N}},$$

in our problem, $L_0 = 0.5$ and $A = 4L_0^2 = 1.0$ since the RVE is composed of a 2×2 square array of unit cells. Porosity $\phi_0 = 0.5$ and N is the total number triangular elements.

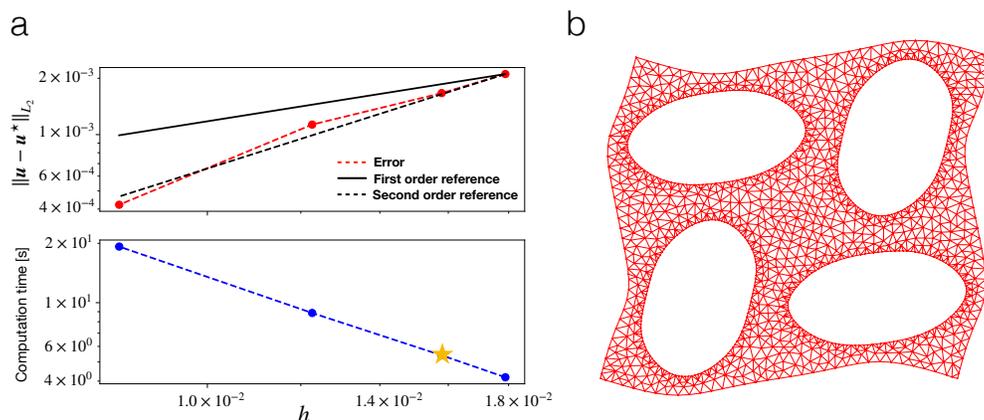


Figure S1: (a) Results of FEM mesh refinement study. Top: L_2 error measured with respect to u^* for of different mesh sizes. “First order reference” refers to an error proportional to h and “Second order reference” refers to an error proportional to h^2 . Bottom: the corresponding computation time. And the star in the figure indicates our choice of mesh size ($h = 0.0158$) used in data collection (also in the DNS). (b) Results of the example used in the refinement study with $h = 0.0158$.

We performed a FEM calculation on an RVE with pore A with $\overline{\mathbf{H}} = \begin{bmatrix} 0 & 0.1 \\ 0 & -0.1 \end{bmatrix}$). We then vary the mesh size and record the corresponding computational error and time. Since no analytical solutions are available for this problem, we first conducted FEM calculation on a fine mesh

size $h = 0.00473 \approx 0.0095L_0$ and denote the corresponding displacement field as \mathbf{u}^* . L_2 errors of coarser meshes are computed with respect to \mathbf{u}^* : $err = \|\mathbf{u} - \mathbf{u}^*\|_{L_2}$. Results are shown in Fig. S1a.

The choice of the “optimal” mesh size largely depends on the accuracy level that we choose. In principle, the finer the mesh size, the closer our results are to \mathbf{u}^* , but it can lead to huge computational costs. The accuracy level is quite heuristic and the choice is not unique, but at least it has to ensure that the results obtained will not miss the essential part of the mechanics, for example, the reorganization of the neighboring unit-cells; and also it should provide high-quality data for training the neural networks. In this work, we chose $h = 0.032L_0 = 0.0158$ as our optimal choice of mesh size, so that the error is well below 2×10^{-3} . Under such mesh size, typically, the differences in deformation field of RVE’s become small enough to capture the essence of the mechanical behaviors. The computation time for each RVE is around 6 seconds on a personal computer, with 3.2 GHz Intel Core i7 CPU and 16GB memory. Multiple increments (typically 10) are used to ensure the convergence of the RVE calculation. The resulting shape as well as the mesh is shown in Fig. S1b.

We maintain the same mesh size for our DNS as well, so that the essential mechanics of the CMM is accurately captured. In fact, for DNS, certain parts of the CMM’s requires a finer mesh, e.g. the traction-free boundaries for *pore A* as we will discuss in later sections. But in general, this mesh size can give us reasonable accuracy to compare with the neural network model.

2 Hyper-parameters tuning

Aside from the weights and biases, which are often called the parameters of neural networks, they are many other parameters associated with the neural network. For example, the architecture of the network, or the learning rate used for training. These parameters are often called the hyper-parameters and they need to be optimized before the training of the neural network.

nnhl=64	bs=32	bs=64	bs=128
lr=0.1	0.003023	0.001504	0.000686
lr=0.01	0.000204	0.000141	0.000134
lr=0.001	0.000204	0.000364	0.000853
nnhl=128	bs=32	bs=64	bs=128
lr=0.1	0.002193	0.001958	0.000382
lr=0.01	0.000298	0.000125	0.000233
lr=0.001	0.000297	0.000373	0.001226
nnhl=256	bs=32	bs=64	bs=128
lr=0.1	0.002479	0.000711	0.000259
lr=0.01	0.000163	0.000146	0.000180
lr=0.001	0.000366	0.000929	0.001479

Table S1: Cross-validation MSE for hyper-parameter tuning. The set with the smallest validation MSE is highlighted.

In this work, the hyper-parameters that we optimized include the number of neurons in the hidden layer (nnhl), batch size (bs), and learning rate (lr). We adopt a standard 5-fold cross-validation for this optimization [1]. First, we randomly divide the training set into five folds: one for validation in rotation and the other four folds for training. We compute the average MSE on validation sets as our cross-validation score and perform a grid search over different combinations

of the above-mentioned hyper-parameters. We report the cross-validation MSE truncated at 1000 epochs in Table S1. Based on the results, we chose the optimal hyper-parameter set as $\text{nnhl} = 128$, $\text{lr} = 0.01$ and $\text{bs} = 64$ according to the smallest cross-validation error reported in the table.

3 Other important results from DNS

3.1 Local orientation

As discussed in the main text, local orientation is an important issue for our proposed computational scheme. Aside from visual observation, here we define the local orientation of the pores by their major axis, i.e., the longest line segment between any two points along the contour of the pore, in the deformed configuration. The results are shown in Fig. S2.

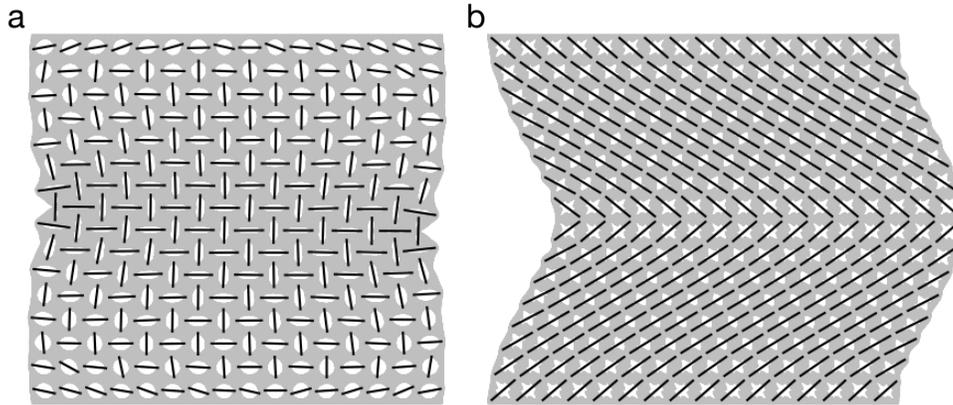


Figure S2: Local orientation of the pores according to the major axis for the CMM with (a) *pore A* and (b) *pore B*. The CMMs are both composed of a 16×16 square array of unit-cells and the result is shown for a compression of $\varepsilon = -0.1$.

As shown in Fig. S2b, for the CMM made of *pore B*, it mainly bifurcates into two different domains, each with a preferred local orientation. And the orientation varies smoothly within each domain—even at the boundaries, no drastic change of local orientation is observed.

Whereas for CMMs made of *pore A*, the local orientation varies drastically over the scale of a unit-cell, as in Fig. S2a. From the figure we also observed that the boundary also strongly affects the local orientation of the pores. On the top and bottom boundary, the relative organization of the pores are very different from those in the center; for those cells along the left and right boundary, severe deformation can happen for certain cells, which often leads the contact of the pores and even material failure at those places. Only in the very center of the CMMs do we observe similar behaviors of the RVEs.

3.2 Boundary effect

In a previous work [2], the experimental stress-strain curve agree well with the unit-cell calculation. The reason is that the stress and strain are obtained via averaging over a block at the center of the CMM. Based on their experimental images, the boundary effect also strongly affects the local orientation and the deformation of the pores. In a recent work [3], it is pointed out that such boundary effects will take place even at small deformation. When the distance of the pore to the boundary is on the same scale of pore size, the influence of boundary over the deformation of the pores can be prominent.

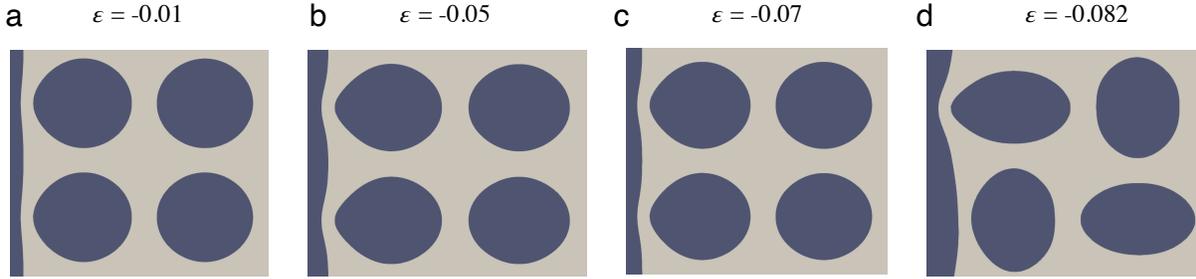


Figure S3: The deformation of the pores (type A) at the boundary under a compression of (a) 1%, (b) 5%, (c) 7% and (d) 8.2%. Note that the deformation field of (a) and (b) are exaggerated for 5 and 2 times respectively for better visual presentation.

As shown in Fig. S3, even from a 1% compression, the boundary start to strongly influence the deformation of the pores nearby. At a larger compression, we can clearly observe that the pores start to develop a sharp tip towards the boundary. This effect persists after the instability happens. As in Fig. S3d, although the local orientation is similar to the RVE results, the top-left pore is clearly affected by the boundary. Strong stress concentration can happen at these tips, leading to possible contact and material failure at these places. We emphasize here that such pore-boundary interaction is present for CMMs with *pore A* under different sizes and since such interaction strongly perturbs the local fields, the boundary effect will not vanish as with larger CMM size.

3.3 Size of the CMM

In the main text, we presented the results for CMMs made of a 16×16 array of unit cells. We have conducted simulation for larger sizes as well (with the same mesh size). As more DoF's need to be solved, the computational time, will increase as with the size of the CMMs. This is confirmed in Table S2.

	pore A			pore B		
Size	16×16	20×20	32×32	16×16	20×20	32×32
Computation time per increment [s]	117	175	457	90	139	391

Table S2: Computation time per time increment for CMMs of different sizes.

Note that the above simulation is carried out using the same relaxation parameter of 0.2. Using a small relaxation parameter is a common practice to improve the convergence of Newton-Raphson solvers, however, as shown in Table S3, a smaller value of this parameter can significantly slow down the simulation. In the main text, we used the largest possible relaxation parameter to minimize the time cost for DNS. But for CMMs with larger sizes, we have to prescribe a small value to this parameter, otherwise the Newton-Raphson solver will easily diverge. Moreover, to reach the same level of compression, more increments might be needed as well. Therefore, with these three reasons mentioned above, DNS of CMMs with larger sizes can be extremely time consuming.

Other than the huge computational cost, according to our results, DNS of CMMs with larger size are more prone to possible contacts of the pores. As shown in Fig. S4, severe overlapping is observed for CMMs composed of a 32×32 array of unit cells, with both *pore A* and *pore B*. In reality, at those places, contacts and even material failure can take place. But since our FEM model do not take these effects into account, the stress-strain curve obtained via DNS on such CMMs are thus not accurate.

	pore A		pore B	
Size	16×16	16×16	16×16	16×16
Relaxation parameter	1.0	0.2	0.6	0.2
Computation time per increment [s]	9	117	26	90

Table S3: Computation time per increment when different relaxation parameters are used.

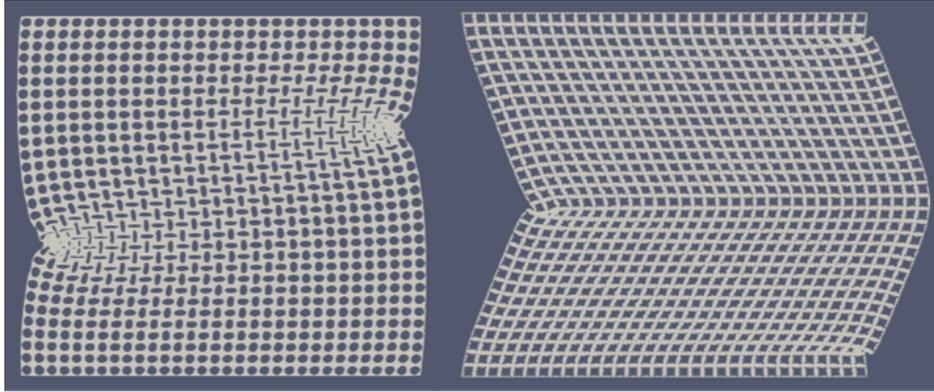


Figure S4: DNS results of CMMs made of a 32×32 array of unit cells, with *pore A* (left) and *pore B* (right), with $\varepsilon = -0.1$.

References

- [1] M. Stone, “Cross-validators choice and assessment of statistical predictions,” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 36, no. 2, pp. 111–133, 1974.
- [2] K. Bertoldi, P. M. Reis, S. Willshaw, and T. Mullin, “Negative poisson’s ratio behavior induced by an elastic instability,” *Advanced materials*, vol. 22, no. 3, pp. 361–366, 2010.
- [3] S. Sarkar, M. Cebron, M. Brojan, and A. Kosmrlj, “Image charges for deformed 2d solid structures with circular holes and inclusions,” *arXiv: 2004.01044*, 2020.