Thank you for your valuable feedback, which is very helpful in improving the paper. We're encouraged by the broadly positive feedback, and greatly appreciate the critical and constructive suggestions.

**Reviewer 1:** *"Put this in the context of other work on computational homogenization / multi-scale finite element methods"* We will squeeze this into the paper. Our method is related to these and the boundary element method (BEM). It can be seen as a way to learn a neural BEM for problems with no analytic BEM, or a way to do multi-scale FEM / direct macro-micro homogenization (Schroeder 2014) but replacing the fine-resolution FEM with a NN and doing the fine-to-coarse averaging somewhat differently.

*"Limitation associated with micro-scale buckling... the coarse-grain behavior might exhibit hysteretic effects":* Good observation: our model will not capture hysteretic effects in the "true" coarse-grained behavior which occur due to "latent" micro-scale displacements. No problem if you just want to find an energy minimizer, but a problem if you want to understand particular loading/solution paths. We will mention this.

**Reviewer 2:** *"How sensitive is the outer optimization to the accuracy of the surrogate gradients?"* Qualitatively, the surrogate gradients' accuracy is very important - the outer optimization produced poor results without Sobolev training and DAGGER. Quantitatively, measuring the sensitivity of the outer optimization to accuracy of the surrogate, and how to design surrogates which are more "robust" to outer simulation, are important questions for future work.

*"Do you know how the CES method scales with system size in terms of accuracy and evaluation time":* In terms of evaluation time, CES scale linearly with the number of cells in the composed solid. In terms of accuracy of the obtained solution, this will depend on the particular macro task, and measuring this scaling we leave to future work.

*"the method to solve the outer optimization over BCs to find minimum energy solutions to the composed surrogates is not clearly discussed":* Thanks for the feedback - we will try clarify this. We impose Dirichlet BCs by fixing the corresponding DoFs (DoFs = blue points in Fig 2). The energy in each cell is computed using the DoFs on the cell boundaries. Free DoFs are optimized to minimize total predicted energy using LBFGS.

*"The discuss of the surrogate and i.i.d. assumptions (starting line 266) is not clear, and as a result it is not clear what the role of DAGGER is":* Thanks for the feedback - we will try clarify this. The surrogate will only be accurate on cell BCs similar to training data. Given an arbitrary training set, the cell BCs encountered while solving the composed solid might be dissimilar to the training set. We use DAGGER to adjust the training set to match states encountered while solving, for a distribution over "reasonable" macro-problems.

*"Are the BCs shared when a boundary is common between two cells":* Yes. We have 1 DoF for each blue point in Fig 2.

*"Its not clear how the HMC and PDE solver are used together":* HMC is used to generate training BCs, preferring larger strains with lower energies (to explore pore-collapse behavior, avoiding non-physical / exploding-energy displacements). The PDE solver is used to compute the gradient of the pdf (which depends on E) w.r.t. the BC. This grad is needed each leapfrog integration step. Given BCs, we run the solver to determine the internal u and E. We compute dE/dBC with the adjoint method. Then we use this to compute the gradient of the pdf w.r.t. the BCs, needed for the leapfrog step.

*"does the HMC require a significant burn-in time before producing reasonable samples":* No. Note: we don't truly care about drawing samples from any given distribution: we just want to obtain a variety of data which explores buckling modes while avoiding non-physical BCs. This initial data is not that important as we use DAGGER to adapt the dataset to the task at hand. Per appendix, HMC took between 3 and 100 leapfrog steps per sample. We save the resulting BC regardless of the outcome of the accept/reject step (which determines the starting point for the next leapfrog integration).

**Reviewer 4:** *The process of using the surrogates to solve the original problem can be explained in more detail. Are the variables corresponding to the path boundary fixed, and the remaining variables optimized?* Yes. The variables corresponding to the BC are fixed, corresponding to a Dirichlet BC on the composed solid. The remaining variables are optimized to minimize the (surrogate-predicted) energy.

*Newton method is neither the fast nor the most stable... a comparison with more sophisticated methods would be helpful:* Thanks, we weren't aware of the Liu et al paper. From a brief look it looks like Liu et al's method is tailored for time-stepping problems where you want an explicit result at many densely/evenly-spaced frames. This is very different to statics (or timestepping with adaptive or larger step-sizes) where it may have less advantage vs Newton's. When we initially / informally tried SNES' quasi-Newton method, it was slower solving the static PDE than Newton's method. However we will make sure to read Liu et al carefully and to discuss and benchmark vs both quasi-Newton methods and dynamic relaxation with kinetic damping in the revision.

**Reviewer 5**: *"There is one outlier in L2 compression that was quite bad":* We will discuss this in the main paper.

*"A comment might help the reader situate this work within the more usual (less idyllic) context of approximating potential energies.":* This is a good suggestion: we will relate to other work in learning energies.

*"I wondered what the dimensions of subspaces were, and how many NN params were used":* The MLP had 3 layers of 128 units. The reduced space was 72d for a 2x2 cell and 690d for a 8x8 composite. Data was gathered with FEM of around 5000d for 2x2 cells (dep. on pore shape). Sizes of each composite FEM baseline are in the appendix.

*"There are enough tricky code pieces that it would be non-trivial for me to reproduce their results":* Yes, there are quite a few moving parts here (largely due to using both Fenics and Pytorch). We provide our code to help with reproducibility.