Supplementary Material: Improving structure search with hyperspatial optimization and TETRIS seeding

Daviti Gochitashvili,[1] Maxwell Meyers,[1] Cindy Wang,[1] and Aleksey N. Kolmogorov[1]

[1] *Department of Physics, Applied Physics, and Astronomy, Binghamton University-SUNY, Binghamton, New York 13902, USA*

(Dated: September 26, 2025)

I Mapping

The original GOSH method relies on a random generation of clusters directly in the hyperspace that treats all coordinates on an equal footing. To make the method compatible with the cluster generation and evolution capabilities available in MAISE, we considered three different mapping protocols that expand structures given in the normal space with d_0 dimensions into an extra dimension. Here, we provide further details on the MAP2 scheme that produces nearly uniform atomic distributions in all d_0+1 dimensions starting with the d_0-dimensional clusters.

In order to ensure uniformity, the normal coordinates for each atom must be rescaled with a mapping function $r_{d_0} = f(r_{d_0}^{\text{old}})$ to account for the increase in the dimensionality of the hypersphere. For simplicity, we first consider unitless r_{d_0} and $r_{d_0}^{\text{old}}$ values to range between 0 and 1, and derive the analytic expression for $f(r_{d_0}^{\text{old}})$ in the special $d_0 = 2$ case (for convenience, we omit the d_0 and 'old' labels in the derivation). The atomic density uniform in the normal space will remain uniform in d_0+1 dimensions if the following integrals are proportional to each other and the A constant is chosen to enforce $f(1) = 1$ if $f(0) = 0$.

$$A\int_0^r d\rho 2\pi\rho = \int_0^{f(r)} df(\rho) 2\pi f(\rho)\sqrt{1-f^2(\rho)}$$

$$A\int_0^r d\rho\rho = \int_0^{f(r)} df f\sqrt{1-f^2}$$

$$\frac{A}{2}r^2 = \left[-\frac{1}{3}\left(1-f^2\right)^{3/2}\right]\Bigg|_0^{f(r)}$$

$$\frac{3A}{2}r^2 = 1-\left(1-f^2(r)\right)^{3/2}$$

With the $A = 2/3$ choice, the mapping function becomes

$$f(r) = \sqrt{1-\left(1-r^2\right)^{2/3}}$$

While we could not obtain analytical mapping functions for $d_0 \neq 2$, we found that the real uniform distributions of points generated directly in the d-dimensional hyperspace (Fig. S1) can be approximated accurately (Fig. S2) with

$$f(r_{d_0}^{\text{old}}) = \sqrt{1-\left(1-\left(r_{d_0}^{\text{old}}\right)^2\right)^p}$$

The exponent values of $p = 0.625$ for $d_0 = 1$ and $p = 0.694$ for $d_0 = 3$ were determined numerically by minimizing the root-mean-square deviation from the constant for histograms with 1-degree bins for angles formed by x_d and x_1 coordinates (see Fig. S2).

If the atomic distances to the origin in the normal space vary between 0 and R_0, the mapping function becomes

$$r_{d_0} = \tilde{R}_0\left[1-\left(1-\left(\frac{r_{d_0}^{\text{old}}}{R_0}\right)^2\right)^p\right]^{1/2} \tag{S1}$$

The $\tilde{R}_0$ scaling factor accounts for the reduction of the hypersphere radius to contain the same N atoms. It can be estimated by redistributing N cubes packed inside the original normal space sphere of radius R_0 as hypercubes with an additional side inside a hypersphere of radius $\tilde{R}_0$. The resulting $\tilde{R}_0/R_0$ ratios are $\frac{2}{\sqrt{\pi}}N^{-\frac{1}{2}} \approx 1.128\cdot N^{-\frac{1}{2}}$, $\left(\frac{3}{4}\right)^{\frac{1}{3}}\pi^{\frac{1}{6}}N^{-\frac{1}{6}} \approx 1.110\cdot N^{-\frac{1}{6}}$, and $2^{\frac{1}{4}}\left(\frac{4\pi}{3}\right)^{\frac{1}{3}}\pi^{-\frac{1}{2}}N^{-\frac{1}{12}} \approx 1.082\cdot N^{-\frac{1}{12}}$ for $d_0 = 1$, $d_0 = 2$, and $d_0 = 3$, respectively.

II Behler-Parrinello symmetry functions

Behler–Parrinello symmetry functions map atomic environments into machine learning input of constant length:

$$G_i^1 = \sum_{\substack{j \neq i \\ \text{all}}} e^{-\eta(R_{ij}-R_s)^2} \cdot f_c(R_{ij}) \tag{1}$$

$$G_i^2 = 2^{1-\zeta} \sum_{\substack{j,k \neq i \\ \text{all}}} (1+\lambda\cos\theta_{ijk})^\zeta e^{-\eta(R_{ij}^2+R_{ik}^2+R_{jk}^2)} f_c(R_{ij}) f_c(R_{ik}) f_c(R_{jk}) \tag{2}$$

$$f_c(R_{ij}) = \begin{cases} 0.5 \times \left[\cos\left(\dfrac{\pi R_{ij}}{R_c}\right) + 1\right], & \text{for } R_{ij} \leqslant R_c, \\ 0, & \text{for } R_{ij} > R_c. \end{cases} \tag{3}$$

They can be naturally extended into higher-dimensional spaces with N coordinates by calculating distances and angles as:

$$R_{ij}^2 = \sum_{n=1}^{N} (x_{in} - x_{jn})^2 \tag{4}$$

$$\theta_{ijk} = \arccos\left(\frac{\sum_{n=1}^{N}(x_{jn}-x_{in})(x_{kn}-x_{in})}{R_{ij} \cdot R_{ik}}\right) \tag{5}$$

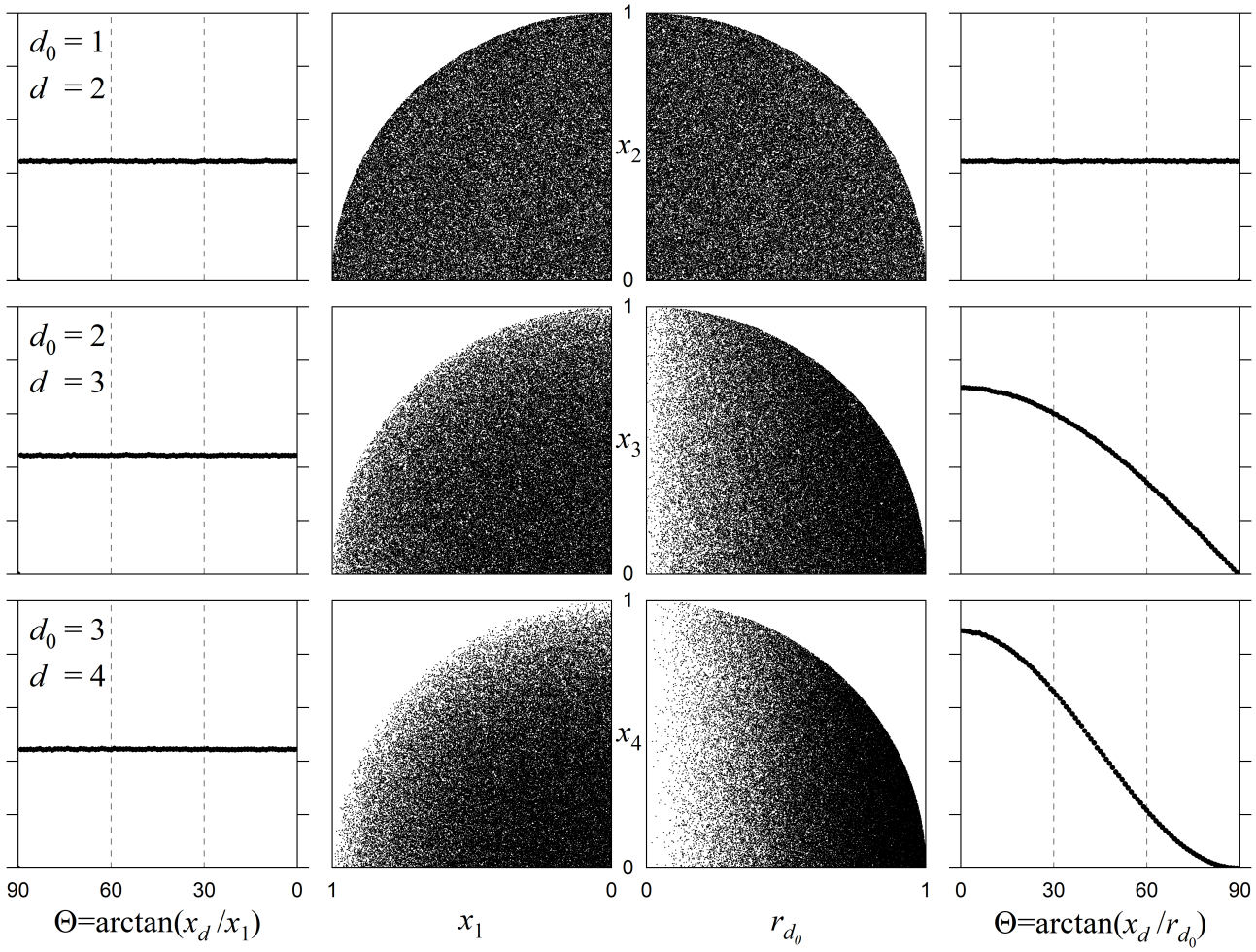

FIG. S1. Distribution of 10^7 random points generated directly in $d = d_0 + 1$ dimensions. Each row of panels corresponds to d_0 = 1, 2, or 3. The leftmost set shows histograms (in arbitrary units) of angles formed by the coordinates along the first and the extra dimensions; the independence on the number of points per bin on the angle signifies the equivalency of these dimensions. The middle left set displays point distributions in the (x_1,x_d) slice. The middle right set illustrates the location of points with the extra coordinate x_d and the normal distance radial vector $r_{d_0} = \sqrt{\sum_{n=1}^{d_0} x_n^2}$. The rightmost set shows histograms of angles formed by x_d and r_{d_0}; the decaying shape for $d_0 > 1$ reflects the lower density of points near the origin of the d_0-dimensional spheres.

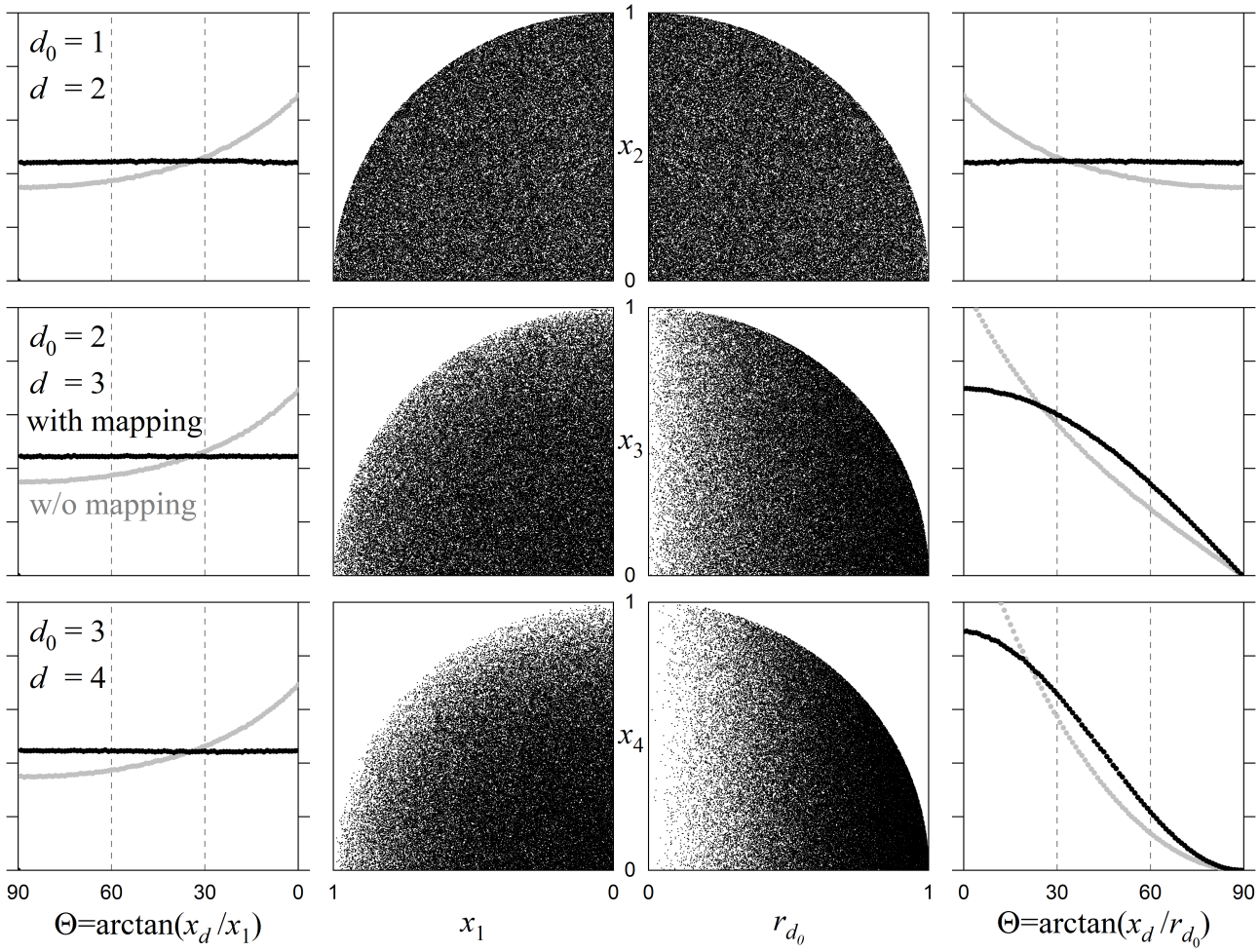

FIG. S2. Distribution of 10^7 random points generated in d_0 dimensions and extended into an extra dimension by randomizing x_d within $\pm\sqrt{\tilde{R}_0^2 - r_{d_0}^2}$ and using the mapping function for the normal coordinates introduced in Eqn. S1. The displayed quantities are described in the caption of Fig. S1, and the only additional data shown in gray correspond to histograms of point distributions without the mapping. The results demonstrate a close match between distributions of points generated directly in d dimensions and points extended into the extra dimension with the proper mapping of the normal coordinates.

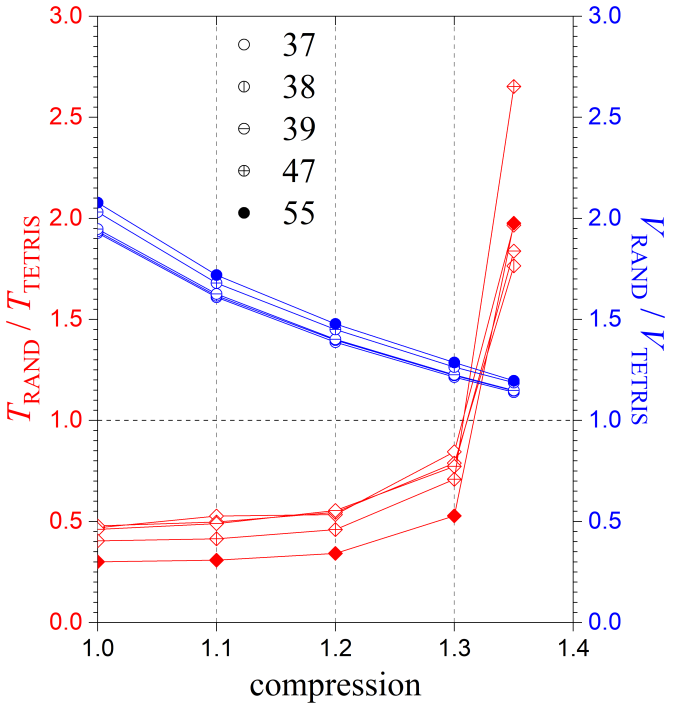

FIG. S3. Performance comparison between TETRIS and BLOB generation of 500 clusters for each size of 37, 38, 39, 47, and 55 atoms. The x axis corresponds to the compression factor c in the BLOB scheme that adds spring forces toward the origin for atoms outside radius $R_{\rm cl}/c$, where $R_{\rm cl}$ is a target radius of the cluster estimated as $1.5\left(\sum R_i^3\right)^{\frac{1}{3}}$ and R_i are hard-core radii of each species defined as 0.7 of the equilibrium bond length in their ground state bulk structures. The red diamonds show the average ratios of CPU times used to create clusters in the two approaches, while the blue circles show the average ratios of the resulting cluster volumes. The test illustrates that the TETRIS scheme creates significantly more compact clusters at a modestly increased computational cost. The sharp increase in the CPU time required to create clusters compressed by a factor of 1.35 reflects a large rate of failed attempts to satisfy the criteria for minimum interatomic distances even after 10 coordinate adjustment iterations with the repulsive and spring potentials.

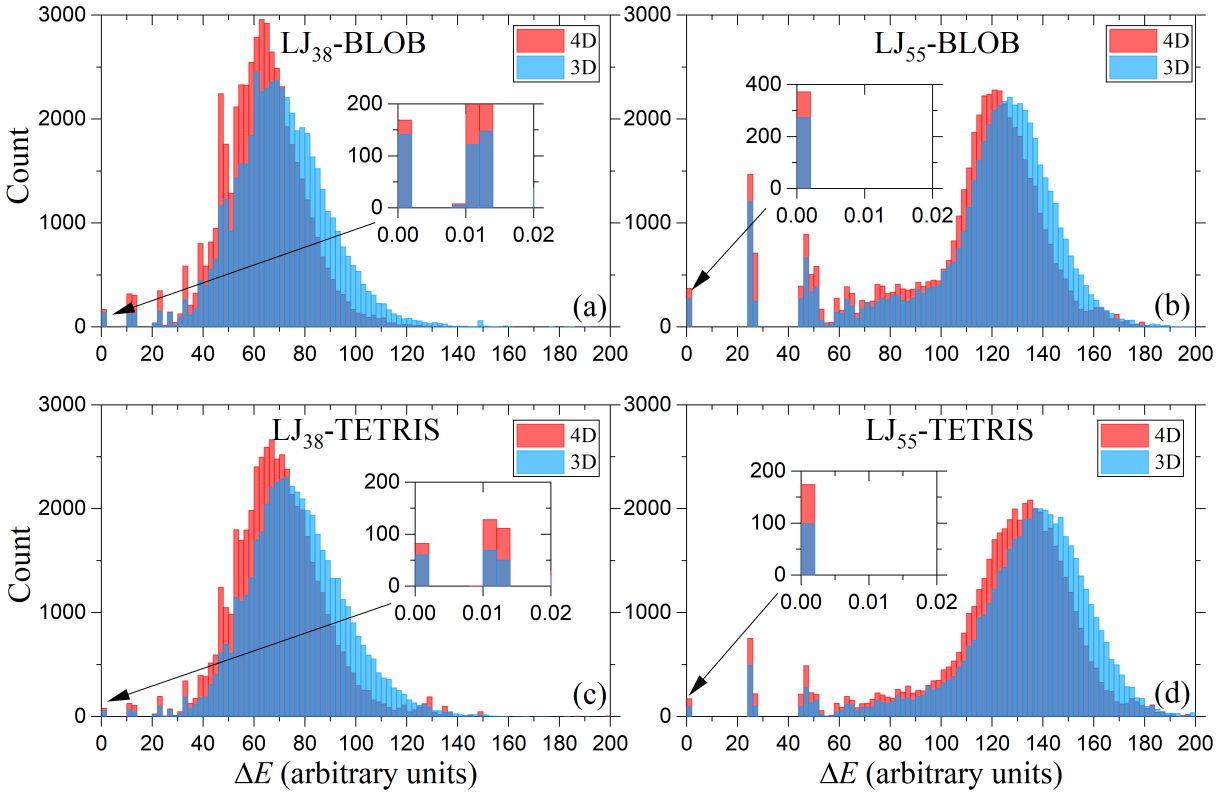

FIG. S4. Histograms of energy differences (ΔE) relative to the ground state for LJ_{38} and LJ_{55} clusters using two structure generation protocols (BLOB and TETRIS) and two relaxation methods (3D and 4D). Panels (a) and (b) show results for BLOB-initialized structures, while (c) and (d) correspond to TETRIS-initialized structures. Red and blue bars represent distributions for 4D and 3D optimization, respectively. Insets magnify the low-energy regions, where configurations close to the ground state appear.

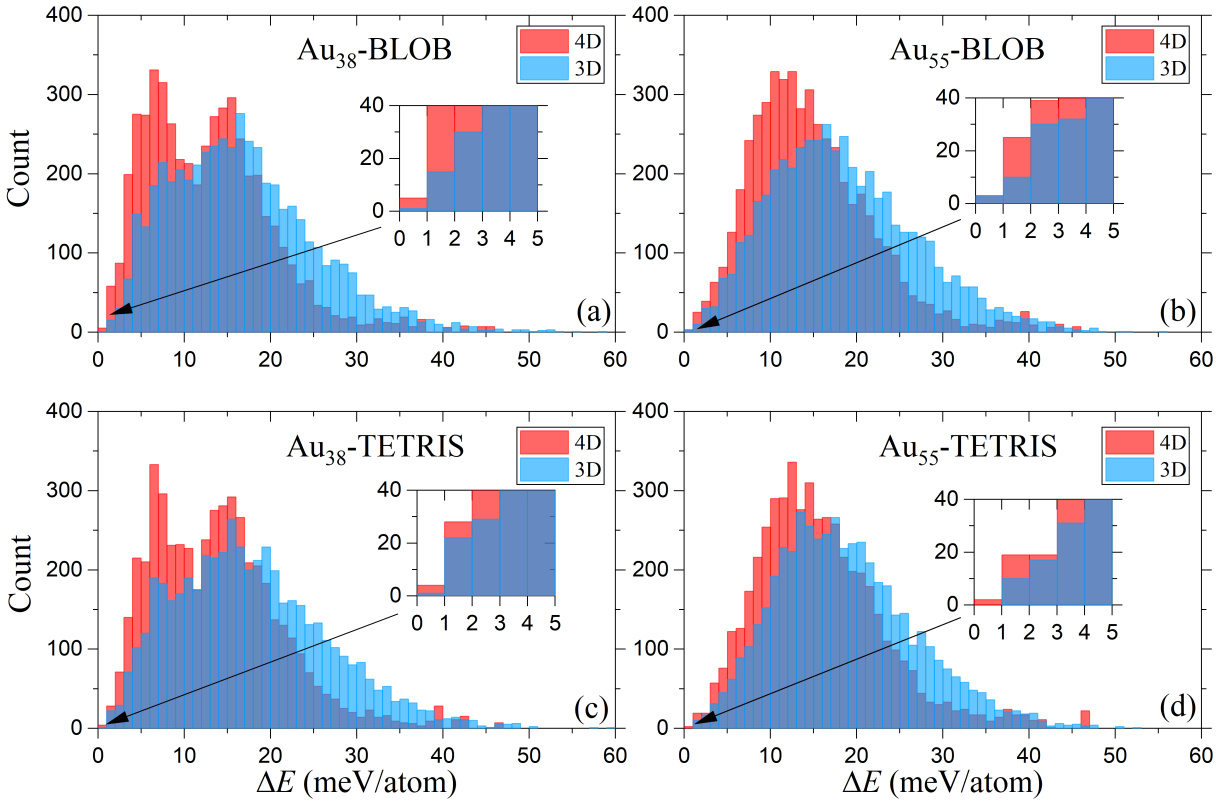

FIG. S5. Histograms of energy differences (ΔE) relative to the ground state for Au_{38} and Au_{55} clusters using two structure generation protocols (BLOB and TETRIS) and two relaxation methods (3D and 4D). Panels (a) and (b) show results for BLOB-initialized structures, while (c) and (d) correspond to TETRIS-initialized structures. Red and blue bars represent distributions for 4D and 3D optimization, respectively. Insets magnify the low-energy regions, where configurations close to the ground state appear.