Supplementary Information to:

Machine-Learned Inter-Atomic Potentials by Active Learning: Amorphous and Liquid Hafnium Dioxide

Ganesh Sivaraman[1*], Anand Narayanan Krishnamoorthy[2,3], Matthias Baur[2], Christian Holm[2], Marius Stan[4], Gábor Csányi[5], Chris Benmore[6], and Álvaro Vázquez-Mayagoitia[7*]

[1]Leadership Computing Facility, Argonne National Laboratory, Lemont, 60439, IL, USA

[2]Institute for Computational Physics, Universität Stuttgart, Allmandring 3, 70569 Stuttgart, Germany

[3]Helmholtz-Institute Münster: Ionics in Energy Storage (IEK-12), Forschungszentrum Jülich GmbH, Corrensstrasse 46, 48149 Münster, Germany

[4]Applied Materials Division, Argonne National Laboratory, Lemont, IL 60439, USA

[5]Department of Engineering, University of Cambridge, Trumpington Street, Cambridge, CB2 1PZ, United Kingdom

[6]X-ray Science Division, Argonne National Laboratory, Argonne, IL 60439, USA

[7]Computational Science Division, Argonne National Laboratory, Argonne, IL 60439, USA

* Author for correspondence (E-mail: vama@alcf.anl.gov)

Supplementary Table

Gaussian Approximation Potential (GAP) Training Parameters

Cutoff radius [Å]	4.0
Smooth cutoff transition [Å]	1.0
Energy regularization [eV per atom]	0.001
Force regularization [eV Å^{-1} per atom]	0.1
Stress regularization [eV]	0.05
Kernel exponent	4
Sparse jitter	10^{-8}
(n_{max}, l_{max})	(6,6)
Sparse points	1800
GAP version	1548461341

Table S1. Final set of parameters used to train liquid and amorphous HfO_2 GAP model.

Supplementary Figures

Error Histogram

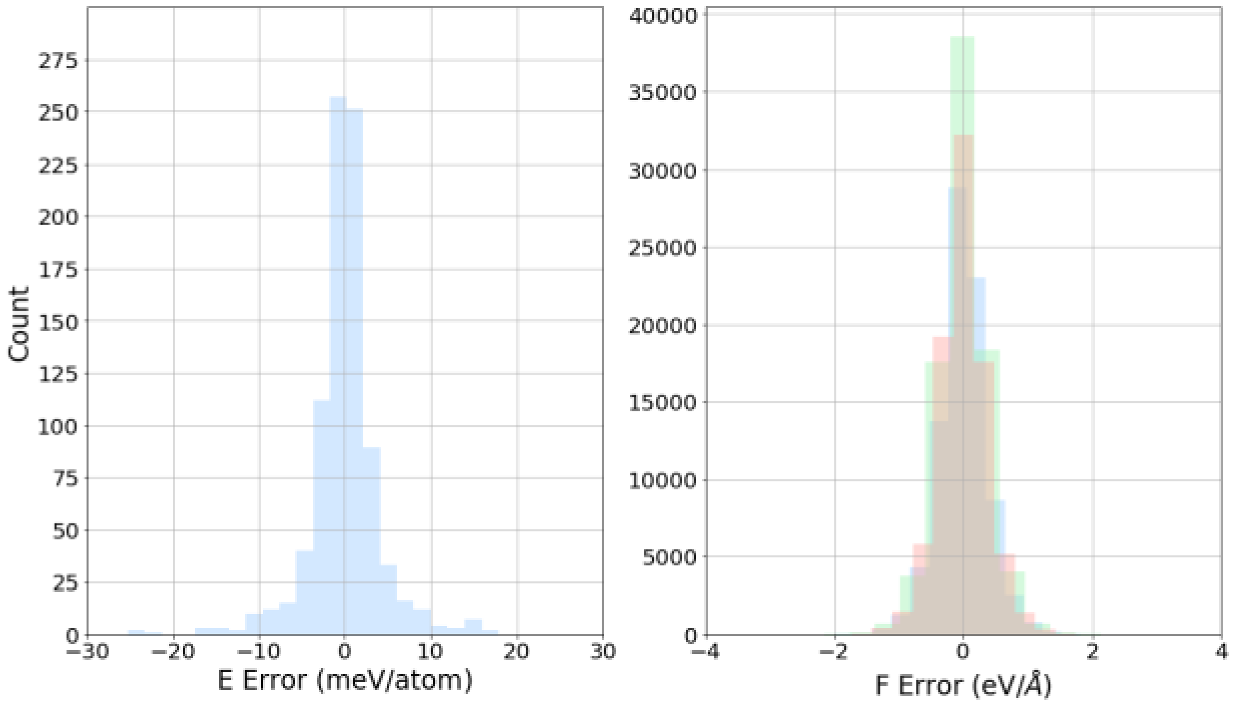

Figure S1: The error histogram prediction using GAP corresponding to the validation plot, Figure 1 from the article. (Left panel) energy, (Right panel) forces, the three different colors refer to XYZ components of the force vector.

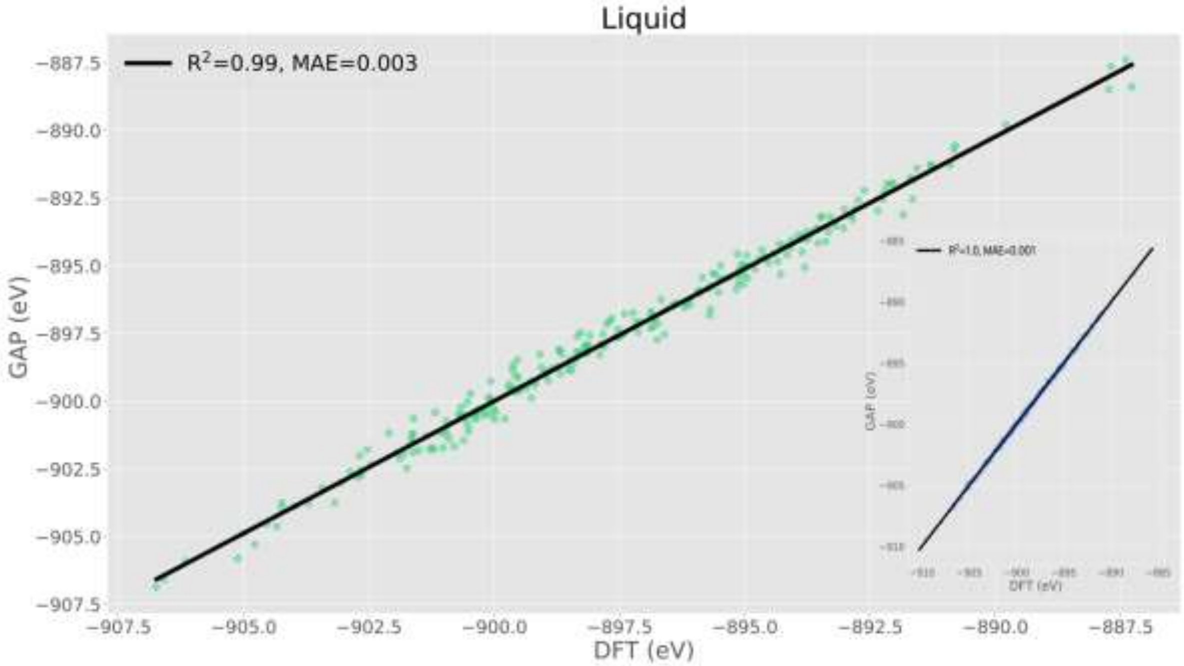

Figure S2: Validation plot for the manual sampled liquid dataset [N_{train} = 250, N_{iter} ~ 10 (i.e. manual iterations)] (Inset plot) Validation plot for the active learned dataset [N_{train} = 467, N_{iter} ~ 4].

Active Learned GAP vs Broglia (ref. 30)

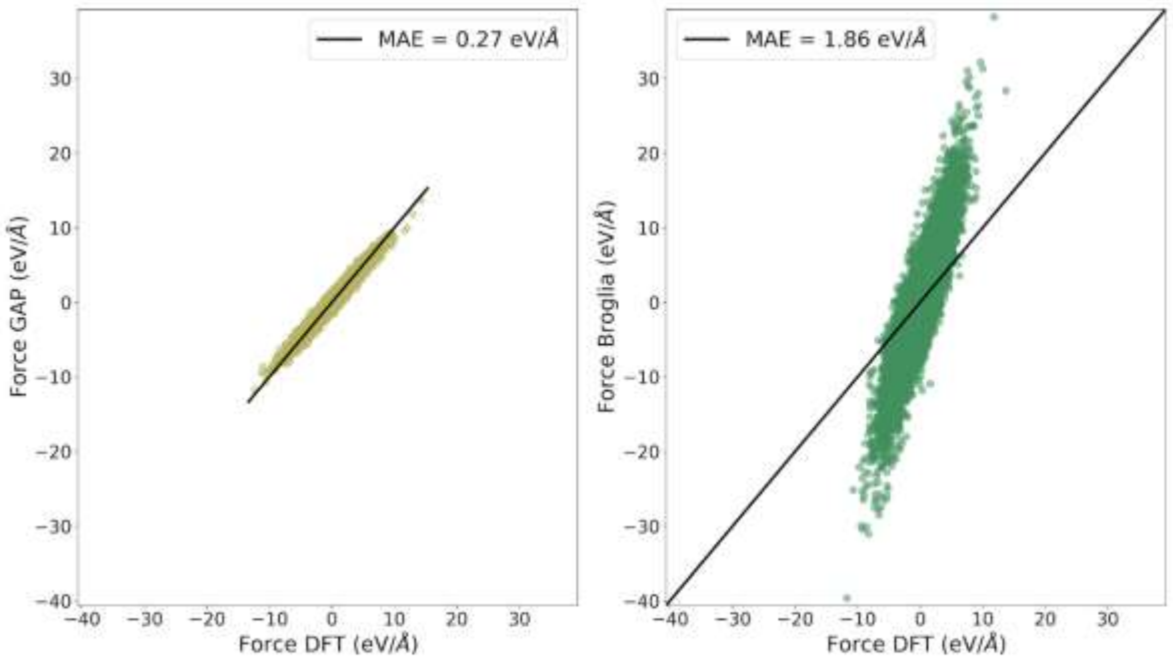

Figure S3: Comparison of (left panel) DFT force vs GAP force plot (right panel) DFT force vs Broglia force plot. Quench test dataset reported in Figure 1 has been used for this test.

Supplementary Discussion

Manual Configuration Selection

Generating the training and test configuration sets to which Machine Learning (ML) potentials are fitted is currently done by manual intuition of the physical systems that is under study. To fit GAP potentials, we need to optimize the desired number of training and test configurations that will achieve the required accuracy of Mean Absolute Error (MAE) < 5meV/atom to model liquid and amorphous hafnia.

For liquid hafnia the ab initio trajectory has 6000 snapshots, thus we split it into three equal-sized parts and choose training data from the 1st third (ids 0-1999), test data from the 3rd third (ids 4000-5999). By neglecting the middle part of the trajectory, we expect that training and test sets are uncorrelated. Training sets have sizes of 100, 200 samples which are either randomly chosen or with uniform index (and thus temporal) spacing (i.e. every ‘i’-th snapshot). From uniform spacing we expect less similarity between the training structures, so we tend to use these training sets. As explained in the text of the methods sections, the number of independent test configurations is chosen equal to the number of training configurations for consistency. Therefore, test sets have also 100, 200 samples chosen with uniform spacing (but from the above-mentioned test set regime). The training data selection is continued until the desired accuracy is reached. To improve the potential, we increase the number of training and test configurations to fit it in a much larger domain. The validation plot for the manual tuned model is shown in Figure S2. The corresponding plot for the active learned liquid dataset is reproduced in the inset. Note that the active learning has more control over the convergence as it stops once the desired E_{tol} is achieved with far less iterations. Similarly, for the quench dataset it took 40 manual iterations and more than 1000 data points to achieve required MAE (not shown). Thus, one can see that this laborious process can be overcome by our automated way of active learning-based fitting of GAP potentials to attain similar or much better accuracy without human intervention quickly.

Active Learning Workflow Example Results

The hyperparameters and results of active learning workflow applied to the example dataset[10] are discussed here. The user defined hyperparameters ’minimum number of clusters’ and ’number of samples’ are both set to value of 10. A total of 10 GAP models are initially generated to fit the Gaussian process regression over the error metric. This is followed by 20 GAP models for the optimization run. Hence, a total of 30 GAP models per data iteration.

A total of 178 clusters are identified by the HDBSCAN. Number of data points in the smallest, largest cluster are 10, 195 respectively.

Self-Diffusion Coefficient

The self-diffusion coefficient (D) of atomic species ‘A’ is obtained from the mean square deviation (MSD) of all particles through Einstein relation[i]:

$$\lim_{t \to \infty} < ||r_i(t) - r_i(0)||^2 > i\epsilon A = 6\, D_A t \qquad (1)$$

Where $r_i(t)$ denotes the position of atomic species at time t. This can be obtained from MD results by fitting the MSD in linear regime.

Correlated-Diffusion Coefficient

To account for correlations, we calculated distinct diffusion constant[4,5,11,12] for Hf and O atoms.

$$\lim_{t \to \infty} < || \sum_{i=1}^{n} r_i(t) - r_i(0)\, ||^2 > = 6\, D_\sigma n t \qquad (2)$$

Here n represents the of atoms of distinct species.

Root Mean Square Deviation of Atomic Positions

Given two sets of atomic positions, u and v with ‘n’ points each. The RMSD[6] is defined as:

$$RMSD(u, v) = \sqrt[2]{\frac{\sum_{i=1}^{n}[(u_{ix} - v_{ix})^2 + \left(u_{iy} - v_{iy}\right)^2 + (u_{iz} - v_{iz})^2]}{n}} \qquad (3)$$

A Short Survey on Clustering

Clustering analysis[7] is the process of organizing unlabeled data in to groups. What constitute a true cluster is highly context and application dependent. Consequently, there are a variety of clustering algorithms each of which approaches differently on how the data is grouped into clusters. In particular, many of the clustering algorithm exploit the notion of distance similarity to group data in to clusters. DBSCAN is a popular algorithm which separate clusters into region of high density from low density based on a distance similarity. It does not require *apriori* setting the number of clusters. The DBSCAN algorithm[8] takes in two hyperparameters namely, ε, a distance scale and k, a density threshold expressed in terms of a minimum number of points. The only drawback of this algorithm is the difficulty in tuning these hyperparameters. The HDBSCAN algorithm[9] improves up on the DBSCAN algorithm by converting in to a single linkage clustering algorithm by defining a new mutual distance reachability distance metric. Thus, avoiding exhaustive search for ε and k. For a given fixed k, the mutual reachability metric can be derived from the distance metric d as follows:

$$d_{mreach}\left(X_i,X_j\right)= \begin{cases} \max\{\kappa(X_i),\kappa(X_j),\ \mathrm{d}(X_i,X_j)\} & \text{if } X_i \neq X_j \\ 0 & \text{if } X_i{=}X_j \end{cases} \qquad (4)$$

For any given point X_i, $\kappa(X_i)$ is the distance to its κ^{th} nearest neighbor.

Bayesian Optimization

The Bayesian optimization objective is to find the optimal hyper paramters (x_{opt}) so as to minimize the error metric (y_{opt}) for the ML model. The pseudo code for the Bayesian Optimization algorithm is provided below[2]:

```
Sample an initial 'p0' random points from the spaces of hyper parameter ($x_i$'s) and compute the corresponding error metrics ($y_i$'s)
Fit a Gaussian process prior on $\{(x_i,y_i)\}_{i=1:p0}$
for t in p0+1 to p0+P do
        • Estimate the next sample for hyper parameter, $x_t$ to be the maximizer of the acquisition function   over x, where the acquisition function is computed using the GP.
        • Compute the new $y_t$ from the $x_t$
        • Update GP with the augmented $\{(x_i,y_i)\}_{i=1:t}$
        • Increment t
end for
Return the best estimate of the solution: $\{x_{opt},y_{opt}\}$
```

In our workflow, Gaussian process regression is used as the surrogate model

$$y = f(x) \sim GP\big(\mu(x), K(x, x')\big) \qquad (5)$$

where $\mu(x)$ is the mean and $K(x,x')$ is the squared exponential covariance.

For the acquisition function we have used the Expected Improvement (EI)[3]. This would mean in our pseudocode the next $\mathbf{x}_t$ which be chosen such that:

$$\boldsymbol{x}_{\mathrm{t}} = argmax_{\mathrm{x}}\, EI(\boldsymbol{x}) \qquad (6)$$

where

$$EI(x) := \quad \mathbb{E}[max(\, f(x^{+}) - \, f(x), 0)] \qquad (7)$$

where f($\mathbf{x}^{+}$) is the value of the lowest error metric observed so far and **x**+ is the location of hyperparameter corresponding to that. For the Gaussian Process Regression, EI can be estimated analytically as follows:

$$\mathrm{EI(x)}= \begin{cases} \big(\mathrm{f(x^{+})} - \mu(\mathrm{x})\big)\Phi(\mathrm{Z}) + \sigma(\mathrm{x})\phi(\mathrm{Z}) & \text{if } \sigma(\mathrm{x}) > 0 \\ 0 & \text{if } \sigma(\mathrm{x})=0 \end{cases} \qquad (8)$$

where

$$\mathrm{Z} = \begin{cases} \dfrac{(\ \mathrm{f(x^{+})} - \mu(\mathrm{x})\ )}{\sigma(\mathrm{x})} & \text{if } \sigma(\mathrm{x}) > 0 \\ 0 & \text{if } \sigma(\mathrm{x})=0 \end{cases} \qquad (9)$$

μ(**x**) and σ(**x**) represents the mean and the standard deviation respectively predicted from the Gaussian process regression at a given x. Φ and ϕ represents the probability density function (PDF) and cumulative density function (CDF) of the standard normal distribution, respectively. The two summation terms on equation 8, can be interpreted as the tradeoff between exploration and exploitations in selection of the next sample $\mathbf{x}_{\mathbf{t}}$.

Supplementary References

[1] Allen, M. P., & Tildesley, D. J. Computer simulation of liquids. *Oxford university press* (2017).
[2] Frazier, P. I. A tutorial on Bayesian optimization. *arXiv preprint arXiv:***1807.02811** (2018).
[3] Jones, D. R., Schonlau, M. & Welch., W. J. Efficient global optimization of expensive black-box functions. *Journal of Global optimization* **13**, 455-492 (1998).
[4] Helfand, E. Transport coefficients from dissipation in a canonical ensemble. *Physical Review* **119.1**, 1 (1960).
[5] Richards, W. D., et al. Design and synthesis of the superionic conductor $\mathrm{Na_{10}SnP_2S_{12}}$. *Nature communications* **7**, 11009 (2016).
[6] https://en.wikipedia.org/wiki/Root-mean-square_deviation_of_atomic_positions
[7] Hennig, C. What are the true clusters?. *Pattern Recognition Letters* **64**, 53-62 (2015).

[8] Schubert, E., et al. DBSCAN revisited, revisited: why and how you should (still) use DBSCAN. *ACM Transactions on Database Systems (TODS)* **42.3**, 1-21 (2017).

[9] McInnes, L. & Healy, J. Accelerated Hierarchical Density Based Clustering. *2017 IEEE International Conference on Data Mining Workshops (ICDMW),* 33-42 (2017).

[10] https://github.com/argonne-lcf/active-learning-md

[11] Uebing, C. & Gomer, R. Determination of surface diffusion coefficients by Monte Carlo methods: Comparison of fluctuation and Kubo–Green methods**.** *J. Chem. Phys.* **100**, 7759–7766 (1994).

[12] Shao, Y., Hellström, M., Yllö, A., Mindemark, J., Hermansson, K., Behler, J., & Zhang, C. Temperature effects on the ionic conductivity in concentrated alkaline electrolyte solutions. *Physical Chemistry Chemical Physics* (2020).