Active Learning for Linearly Parameterized Interatomic Potentials

Khachik Sargsyan

Sandia National Laboratories, Livermore, CA

Data Science for Chemistry Reading Group
SNL-CA
May 21, 2019

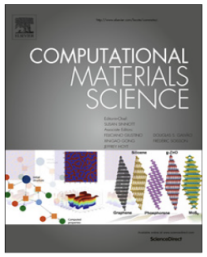
Focus Paper

Computational Materials Science 140 (2017) 171–180

Contents lists available at ScienceDirect

Computational Materials Science

journal homepage: www.elsevier.com/locate/commatsci



Active learning of linearly parametrized interatomic potentials

Evgeny V. Podryabinkin, Alexander V. Shapeev *

Skolkovo Institute of Science and Technology, Moscow, Russia

ARTICLE INFO

Article history:
Received 28 June 2017
Received in revised form 20 August 2017
Accepted 21 August 2017
Available online 15 September 2017

Keywords:
Interatomic potential
Active learning
Learning on the fly
Machine learning
Atomistic simulation
Moment tensor potentials

ABSTRACT

This paper introduces an active learning approach to the fitting of machine learning interatomic potentials. Our approach is based on the D-optimality criterion for selecting atomic configurations on which the potential is fitted. It is shown that the proposed active learning approach is highly efficient in training potentials on the fly, ensuring that no extrapolation is attempted and leading to a completely reliable atomistic simulation without any significant decrease in accuracy. We apply our approach to molecular dynamics and structure relaxation, and we argue that it can be applied, in principle, to any other type of atomistic simulation. The software, test cases, and examples of usage are published at http://gitlab.skoltech.ru/shapeev/mlip/.

Overview

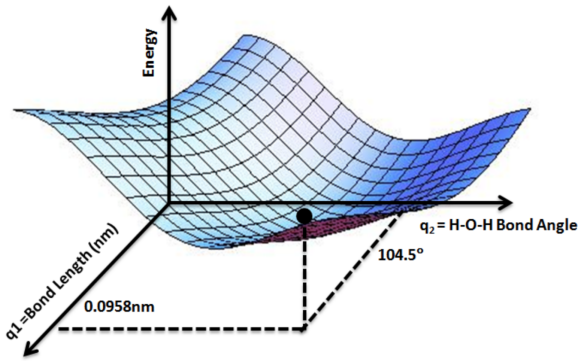
- Machine learning for interatomic potentials (MLIP)
- Active learning (AL) strategies
- Linear regression, moment tensor potentials (MTP)
- AL for MLIP

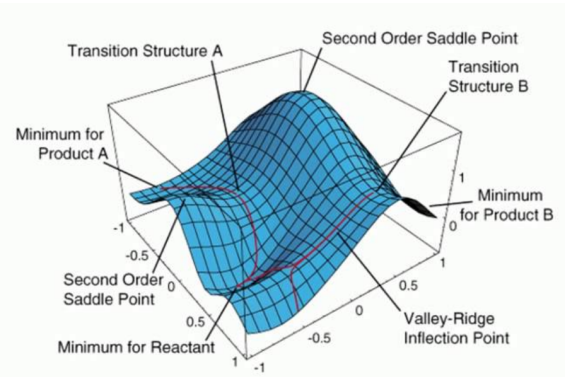
Main target: PES approximation

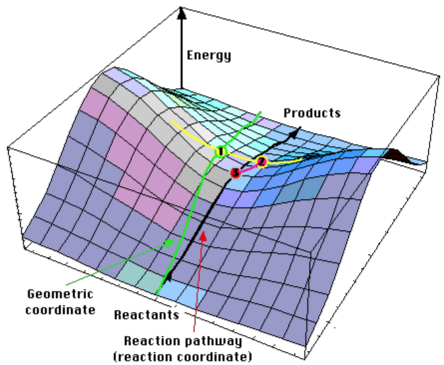
$$E = f(x)$$

x is coordinates/descriptors E is energy

- Accurate and fast surrogates for PES to replace QM computations for studies requiring many PES inquiries
  - saddle point search, transition paths, barrier heights
  - rapid assessment of reaction characteristics
  - automate the discovery of reactive pathways







ML Interatomic Potentials

- Partition the interatomic interaction energy into individual contributions of the atoms (and sometimes bonds, bond angles, etc.)
- Assume flexible functional forms of such contributions
  - Function of positions of the neighboring atoms
  - $O(100)$ parameters
- Require the energy, forces and/or stresses predicted by a MLIP to be close to those obtained by a QM model on some atomic configurations
  - Training set, training/fitting

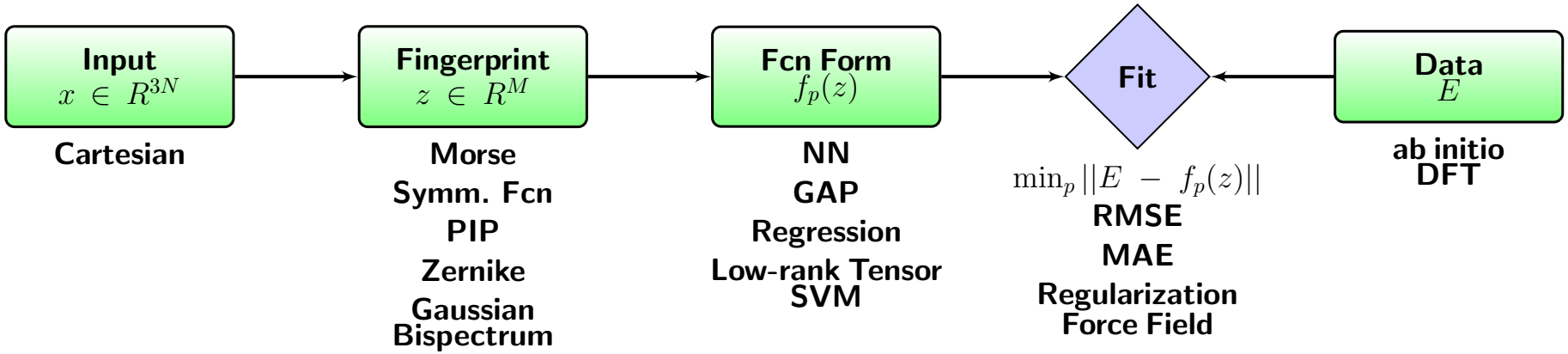
MLIP - desired features

- Good input descriptors
- Accurate, fast-to-evaluate, analytic derivatives
- High-dimensional, flexible functional form
- Transferable/generalizable to unseen atomic configurations
- Systematic improvement and tunability, e.g. reach arbitrary accuracy with more parameters and more training data
- Account for physics:
  - invariant with respect to translation,rotation, and reflection of the space, and also permutation of chemically equivalent atoms
- Locality (depend on surrounding atoms only within a finite cut-off radius), but remain smooth with respect to atoms entering and leaving the local neighborhood

ML for PES

- Weighted interpolation [Ischtwan 1994; Dowes, 2007-09; Maisuradze, 2009]
- Permutationally invariant polynomials [Xie, 2010]
- Gaussian processes [Bartok, Csanyi 2010-15; Mills, 2012; Rupp, 2013; Cui, 2016; Uteva, 2017; Guan, 2018; Schmitz, 2018]
- Low-rank tensor expansions [Jackle, 1996; Baranov, 2015; Rai, 2017, 2018]
- Support vector machines, kernel regression [Le, 2009; Balabin, 2011; Dral, 2017]
- Neural networks (NN) [Blank, 1995; Tai No, 1997; Prudente, 1998; Lorenz, 2004; Witkoskie, 2005; Manzhos, 2006-09; Malshe, 2008; Le, 2009] [Behler, 2010-16; Handley, 2010, 2014; Jiang, 2013; Li, 2013; Dolgirev, 2016; Khorshidi, 2016; Peterson, 2016; Carr, 2016; Kolb, 2016; Shao, 2016; Chmiela, 2017; Cubuk, 2017; McGibbon, 2017; Smith, 2017; Schutt, 2017; Yao, 2017; Hajinazar, 2017; Bereau, 2018; Lubbers, 2018; Unke, 2018; Wang, 2018; Natarajan, 2018; Zhang, 2018; Onat, 2018]

Main ingredients for supervised ML



- Training data $(\boldsymbol{x}_i, E_i)$ for $i = 1, \ldots, S$, and $\boldsymbol{x}_i \in R^{3N}$
  - *ab initio*, DFT
- Input representation, aka fingerprint, aka descriptor

$$(\boldsymbol{z}_i, E_i)$$

- Parameterized functional form of the approximation class

$$f_p(\boldsymbol{z})$$

- Loss function

$$\min_p \sum_{i=1}^{S} \left(E_i - f_p(\boldsymbol{z}_i(\boldsymbol{x}))\right)^2$$

Functional Forms

- Linear Regression $f(x) = \sum_{k=0}^{K} c_k \Psi_k(x)$
  - Polynomial basis
  - Radial basis functions
- Low-Rank Tensor Expansion $f(x) = \sum_{r=1}^{R} c_r \prod_{j=1}^{d} \Psi_{rj}(x_j)$
  - Canonical format
  - Tensor-train format
- Gaussian processes $P(f|\mathcal{D})$
  - Hierarchical correction
  - Flexible kernels
- Neural Networks $f(x) = ...W_3\sigma(W_2\sigma(W_1x + b_1) + b_2) + b_3$
  - Multilayer Feed-Forward NN
  - Convolutional NN
  - Recurrent NN
  - ...

Key target in this work: good training set

- Extrapolation (prediction outside the training domain) is dangerous always
  - e.g., predict the double vacancy formation energy if only single vacancies are present in the training set It is hardly expected that a MLIP can extrapolate beyond the training domain, but even developing a reliable problem-specific MLIP that would accurately interpolate within the training domain is nontrivial
- Naive idea: Sample the entire space of atomic environments within a constraint on the minimal interatomic distance. It is, however, not clear how to do this with sufficient accuracy due to high dimensionality of the space of atomic neighborhoods.
- Perturb from a ‘good’ set of configurations
- Sampling from an ab initio MD, or from a classical MD with empirical or ML potential

None of these are bulletproof, and leave gaps or are forced to extrapolate.

Active Learning

The key idea behind active learning is that a machine learning algorithm can achieve greater accuracy with fewer training samples if it is allowed to choose the data from which is learns.
(We call this OED, with a slight stretch of the meaning).

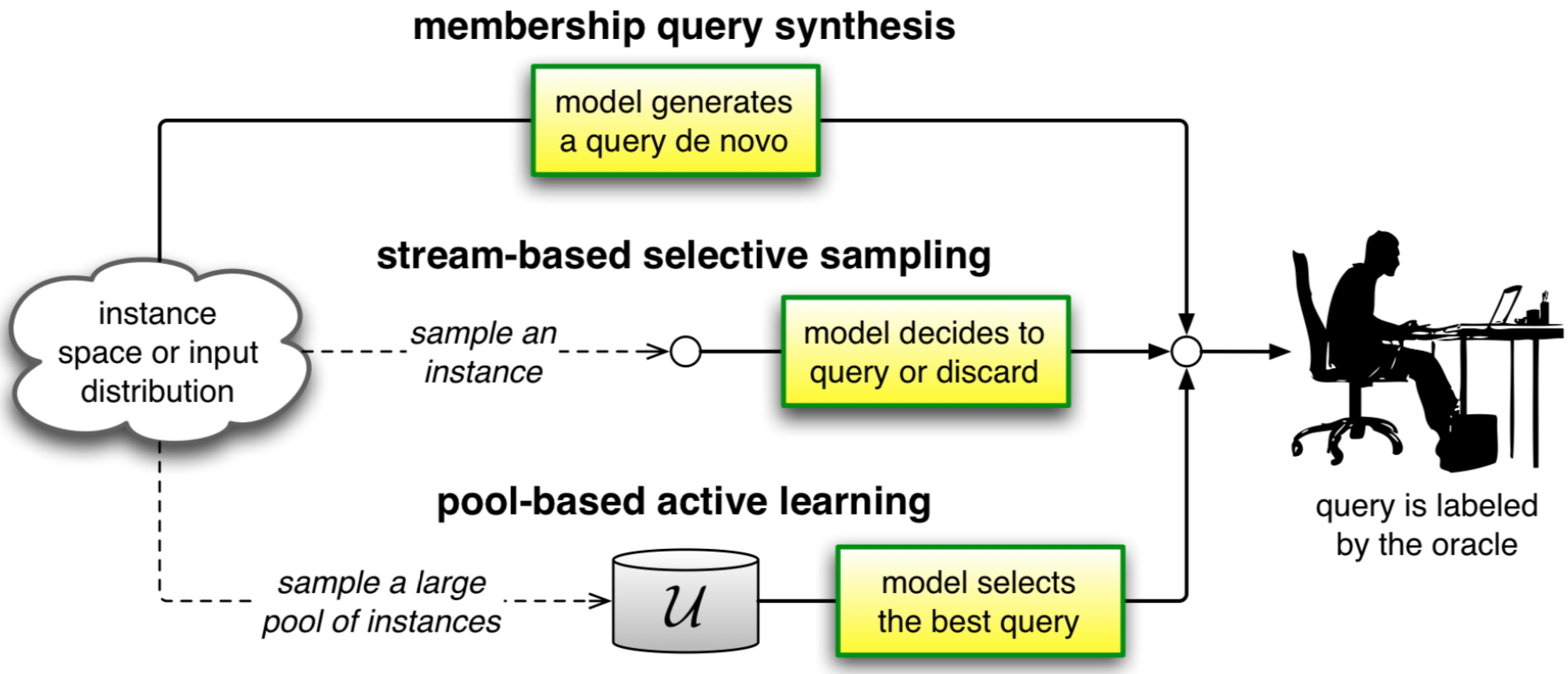
AL approaches

- Detect extrapolative configurations on-the-fly and get QM data for those
- Select a batch of extrapolative configurations offline, a priori

---

- Key: *query strategy*, whether to query QM or not. If such decision can be made reliably, then one does not need to start with a very good training set

Active Learning Scenarios



[B. Settles, "Active learning literature survey", Computer Sciences Technical Report 1648, University of Wisconsin-Madison, 2009]

Query Strategies

- **Uncertainty sampling:** an active learner queries the instances about which it is least certain how to label. Straightforward for probabilistic models.
- **Query-by-committee:** committee of competing models, that are consistent with the current training set. The most informative query is considered to be the instance about which they most disagree. Key is to have a meaningful set of models. Need a measure of disagreement. Again, Bayesian/probabilistic is the best bet, but there are also non-probabilistic methods such as query-by-boosting and query-by-bagging.
- **Expected model change:** which query would lead to greatest model change, e.g. largest gradient length.
- **Variance Reduction and Fisher Information Ratio:** in regression setting, minimizing the variance component of generalization error (usually some sort of approximation or via Fisher)
- **Estimated error reduction:** Estimate the expected future error that would result if some new instance x is labeled and added to training set, and then select the instance that minimizes that expectation. Naively retrain with all potential new points. Practical if incremental training is possible, e.g. GP, or linear MLIP such as in this paper.

AL: connections

- **Semi-supervised learning**, data augmentation: making the most out of unlabeled data. E.g. self-training in which the learner is first trained with a small amount of labeled data, and then used to classify the unlabeled data. Typically the most confident unlabeled instances, together with their predicted labels, are added to the training set, and the process repeats. AL uncertainty sampling is the opposite: the instances about which the model is least confident are selected for querying.
- **Reinforcement learning**: it is easy to converge on a policy of actions that have worked well in the past but are sub-optimal or inflexible. In order to improve, a reinforcement learner must take risks and try out actions for which it is uncertain about the outcome, just as an active learner requests labels for instances it is uncertain how to label. Exploration-exploitation trade-off in the reinforcement learning literature.
- **Optimal Experimental Design**

- Further possible enhancements to query strategies
  - incorporate weights to reduce outlier choice
  - incorporate cost of acquisition

AL for interatomic potentials

[S.L. Frederiksen, K.W. Jacobsen, K.S. Brown, J.P. Sethna, "Bayesian ensemble approach to error estimation of interatomic potentials", Phys. Rev. Lett. 93:16, 2004]: Bayesian query, shows that Bayesian error is a good placeholder for the true discrepancy (with empirical potentials)

[J. Behler, "Representing potential energy surfaces by high-dimensional neural network potentials", J. Phys. Condensed Matter 26:18, 2014] NNs with different architectures, the one point where they disagree the most, is the new selected point.

[V. Botu, R. Ramprasad, "Learning scheme to predict atomic forces and accelerate materials simulations", Phys. Rev. B 92:9, 2015] train a machine learning model predicting the force errors based on the distance between a given atomic configuration and the training set. In the kernel ridge regression context.

This work applies directly to linear MLIP only

$\mathbf{r}_i = (r_{i1}, \ldots, r_{in})$: neighborhood, a collection of vectors from atom i to its neighbors within a cut-off $R_{cut} > 0$.

Total energy is

$$E(x) = \sum_{i=1}^{N} V(\mathbf{r}_i)$$

- Linearity in parameters θ:

$$V(\mathbf{r}_i) = \sum_{j=1}^{m} \theta_j B_j(\mathbf{r}_i)$$

or

$$E(x) = \sum_{j=1}^{m} \theta_j b_j(x)$$

Bases are contractions of Moment Tensor Potentials

[A.V. Shapeev, “Moment tensor potentials”, Multiscale Model. Simul. 14:3, 2016]

$$V(\mathbf{r}_i) = \sum_{j=1}^{m} \theta_j B_j(\mathbf{r}_i)$$

MTP bases are

$$B_\alpha(\mathbf{r}_i) = \sum_{\gamma \in N^k} \left(\prod_{l<m}^{k} (r_{i,\gamma_m} \cdot r_{i,\gamma_l})^{\alpha_{m,l}} \right) \left(\prod_{l} f_{\alpha_{l,l}}(|r_{i,\gamma_l}|) \right)$$

In practice, B_α are tensor contractions of tensor-valued descriptors

$$M_{\mu,\nu}(\mathbf{r}_i) = \sum_j f_\mu(r_{ij}) \underbrace{r_{ij} \otimes r_{ij} \otimes \cdots \otimes r_{ij}}_{\nu \text{ times}}$$

where μ, ν depend on α. Interpreted as moments of inertia: if $f_\mu(r_{ij})$ is the weight of atom j in neighborhood of atom i, then $M_{\mu,0}$ is the neighborhood mass, $M_{\mu,1}$ is the vector of first moments of inertia, etc...
Provably approximate any regular function satisfying all the needed symmetries, since this is a basis for the set of all polynomials invariant with respect to **permutation, rotation, and reflection**.

Linear regression

$$E(x_i) = y_i \approx \sum_{j=1}^{m} \theta_j b_j(x_i)$$

In matrix notation: $\boldsymbol{y} \approx \boldsymbol{A}\boldsymbol{\theta}$, with a design matrix $A_{ij} = b_j(x_i)$.
Least-squares solution is

$$\boldsymbol{\theta} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{y}$$

Covariance estimate of the solution $\Sigma_{\boldsymbol{\theta}} \propto (\boldsymbol{A}^T\boldsymbol{A})^{-1}$

Optimality options

Straight out of wiki...

- **A**-optimality ("**average**" or **trace**)
  - One criterion is **A-optimality**, which seeks to minimize the trace of the inverse of the information matrix. This criterion results in minimizing the average variance of the estimates of the regression coefficients.
- **C**-optimality
  - This criterion minimizes the variance of a best linear unbiased estimator of a predetermined linear combination of model parameters.
- **D**-optimality (**determinant**)
  - A popular criterion is **D-optimality**, which seeks to minimize $|(X'X)^{-1}|$, or equivalently maximize the determinant of the information matrix X'X of the design. This criterion results in maximizing the differential Shannon information content of the parameter estimates.
- **E**-optimality (**eigenvalue**)
  - Another design is **E-optimality**, which maximizes the minimum eigenvalue of the information matrix.
- **T**-optimality
  - This criterion maximizes the trace of the information matrix.

Other optimality-criteria are concerned with the variance of predictions:

- **G**-optimality
  - A popular criterion is **G-optimality**, which seeks to minimize the maximum entry in the diagonal of the hat matrix $X(X'X)^{-1}X'$. This has the effect of minimizing the maximum variance of the predicted values.
- **I**-optimality (**integrated**)
  - A second criterion on prediction variance is **I-optimality**, which seeks to minimize the average prediction variance *over the design space*.
- **V**-optimality (**variance**)
  - A third criterion on prediction variance is **V-optimality**, which seeks to minimize the average prediction variance over a set of m specific points.[9]

D-optimality, three strategies employed

- There is an efficient matrix algo for checking D-optimality (MaxVol, see below)
- Appealing mathematical interpretations, such as decreasing the uncertainty in determining the parameters or maximizing the volume spanned by the training set in the space of configurations, thus avoiding extrapolation.
- Can be applied to any linear potential, e.g. SNAP or GAP, not just MTP

QS_1: Only look at energy values. Maximizes the volume of the simplex in R^m formed by m descriptor vectors.
QS_2: Look at energies, forces and stresses.
QS_3: Also fit the neighborhood basis separately for each atom i.
"Catches" configurations with the most different atomic neighborhoods in the sense of the D-optimality criterion.

Details of the method, focus on QS_1

[E. Podryabinkin, E. Tikhonov, A. Shapeev, A. Oganov, "Accelerating crystal structure prediction by machine learning interatomic potentials with active learning", Phys. Rev. B 99, 2019]

$\boldsymbol{y} \approx \boldsymbol{A}\boldsymbol{\theta}$ is solved by $\theta = (\boldsymbol{A}^T\boldsymbol{A})^{-1}A^T\boldsymbol{y}$

Assume the number of training points is the same as basis size (not crucial), so $\theta = \boldsymbol{A}^{-1}\boldsymbol{y}$

Minimizing $|\boldsymbol{A}^{-1}|$ is maximizing $|A|$.

Form a row-vector $\boldsymbol{c} = (b_1(x^*), \ldots, b_m(x^*))\boldsymbol{A}^{-1}$.

Two parallel interpretations:

- Replacing k-th row of $\boldsymbol{A}$ with $(b_1(x^*), \ldots, b_m(x^*))$ increases its determinant by $|c_k|$.
- $E(x^*) = \sum_{i=1}^m c_i E(x_i)$, so $\max|c_i|$ is the degree of extrapolation! If all $|c_i| < 1$, we are interpolating. Kind of.

It remains to compare $\max|c_i|$ with a $\gamma_{thr} > 1$.

MaxVol algorithm, relevant for QS_2 and QS_3

[S. Goreinov, I. Oseledets, D. Savostyanov, E. Tyrtyshnikov, N. Zamarashkin, "How to find a good submatrix", in: Matrix Methods: Theory, Algorithms, Applications, Word Scientific, 2010.]

The query strategies QS_2 and QS_3 require finding the $m \times m$ submatrix A with maximal $|\det \mathsf{A}|$ (or, in other words, with the maximal volume) in an $k \times m$ matrix B. This is done by the maxvol algorithm [35]. This algorithm is based on greedy selection of rows from B. Each iteration of this algorithm has $O(mk)$ complexity. The algorithm is as follows.

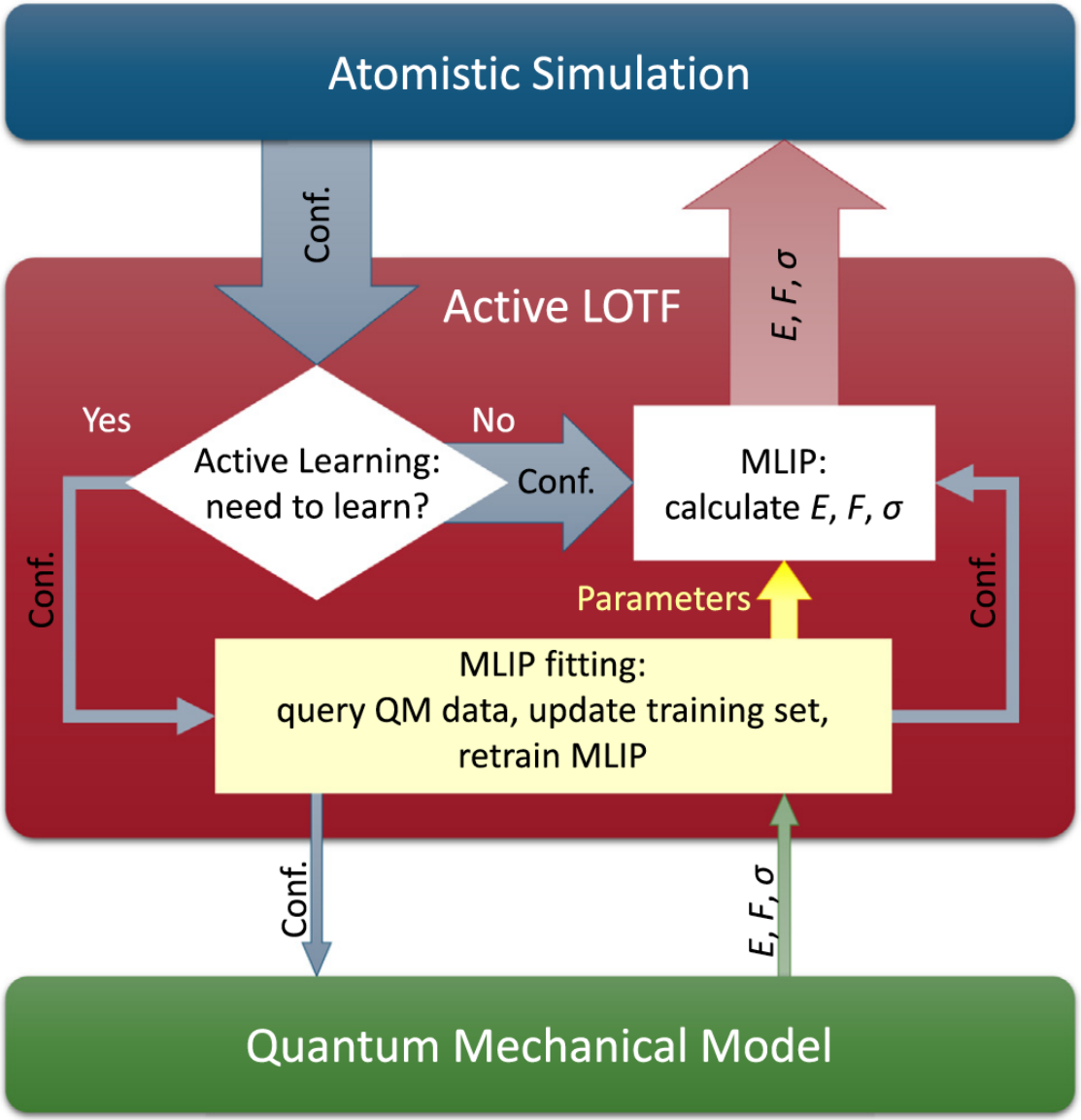
1. Start with an initial (e.g., randomly chosen) $m \times m$ submatrix A of B and calculate $\mathsf{C} = \mathsf{B}\mathsf{A}^{-1}$.
2. Find the maximal by absolute value element C_{ij} in this matrix.
3. If $|\mathsf{C}_{ij}| > \gamma_{\text{th}}$ then:
   3.1 swap the i-th row of A with the j-th row of B,
   3.2 update $\mathsf{C} = \mathsf{B}\mathsf{A}^{-1}$ using the Sherman-Morrison [38] rank-one update,
   3.3 go to step 2.

The smaller the threshold parameter $\gamma_{\text{th}} > 1$ is, the larger $|\det \mathsf{A}|$ will be, at the cost of making more iterations.

AL algorithm employed in this work

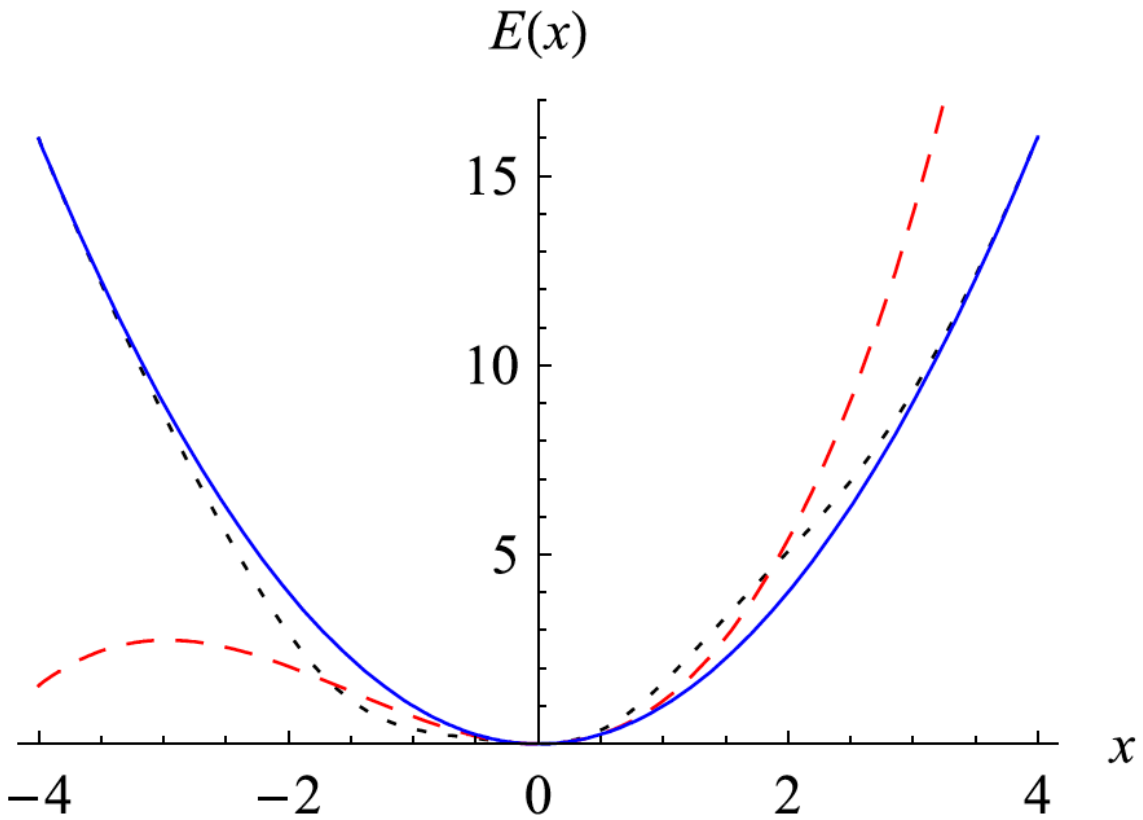
1. Calculate extrapolation grade, $\gamma(x^*)$. If $\gamma(x^*) \leqslant \gamma_{\text{th}}$ then go to step 5, else
    2. Calculate $E^{\text{qm}}(x^*), f^{\text{qm}}(x^*)$, and $\sigma^{\text{qm}}(x^*)$ with a quantum-mechanical model.
    3. Update X_{TS} (and hence A) with x^*.
    4. Re-fit the MLIP and obtain new $\theta_1, \ldots, \theta_m$.
5. Return $E(x^*), f(x^*), \sigma(x^*)$ according to the current values of $\theta_1, \ldots, \theta_m$.

Full Workflow



Toy example

- Black dotted line: true function
- Red dashed line: best fit minimizing L_2 distance
- Blue solid line: AL chooses the end-points as the optimal points



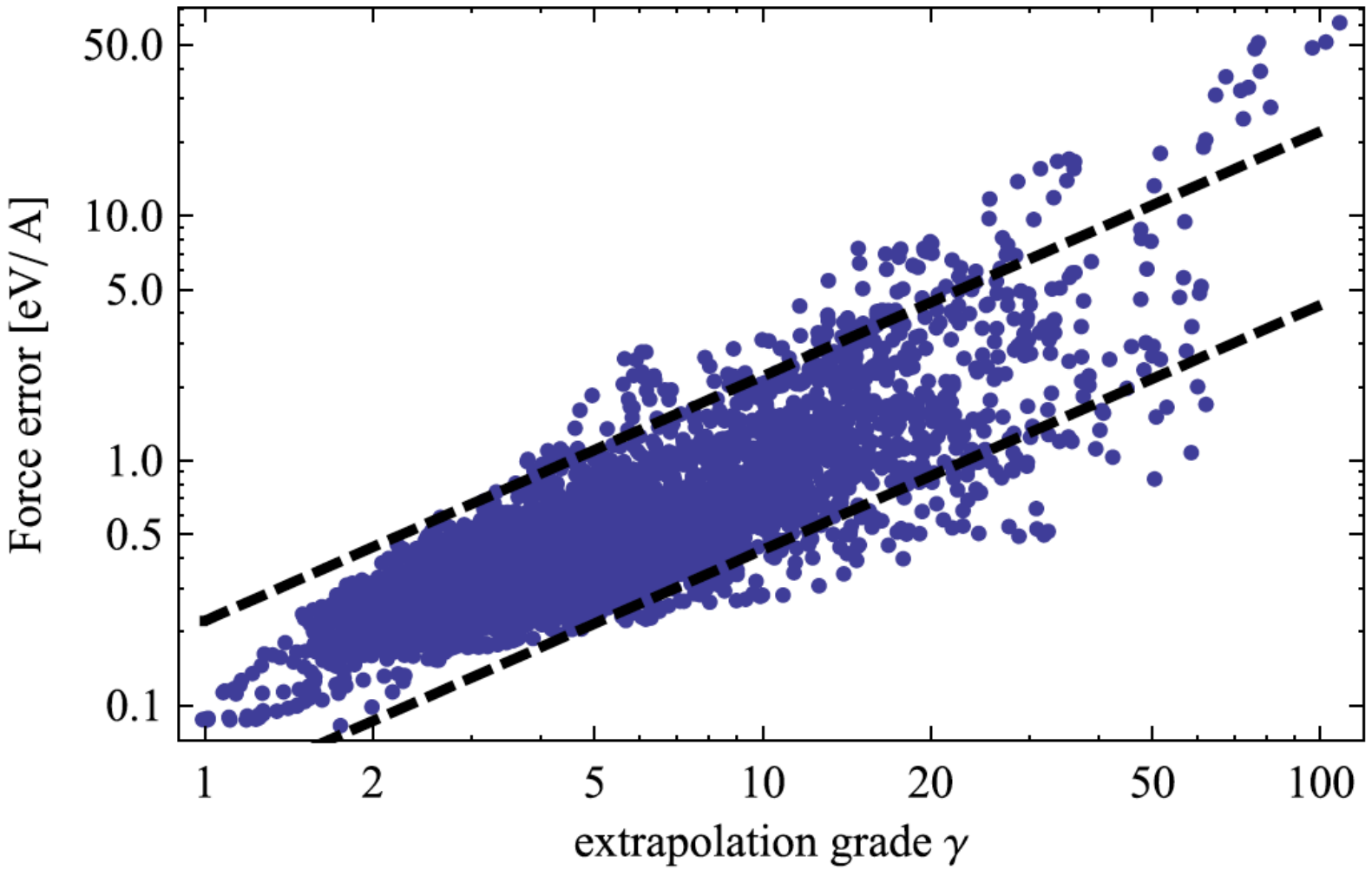
Errors of fitting of as basis set grows

- RMS fitting errors in energy, forces and stresses for MTPs with different number of basis functions.
- The root-mean-square (RMS) and the maximum errors are reported.

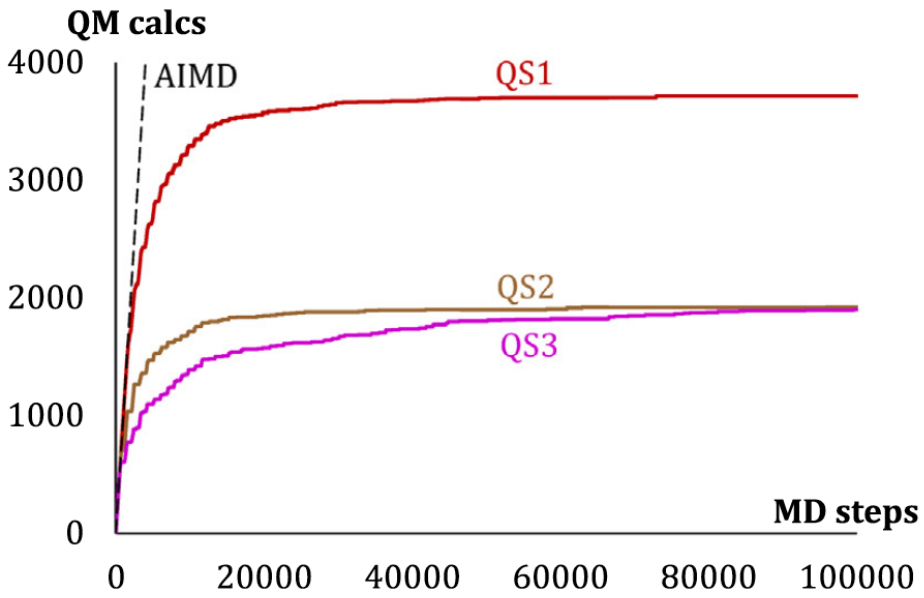
m	Energy error (meV/atom)	Force error		Stress error	
		(eV/Å)	(%)	(GPa)	(%)
10	0.35	0.023	7.4	0.073	7.0
30	0.22	0.018	5.8	0.060	5.8
100	0.19	0.016	5.1	0.052	5.2
300	0.17	0.015	4.9	0.045	4.4
1000	0.15	0.015	4.8	0.040	3.8

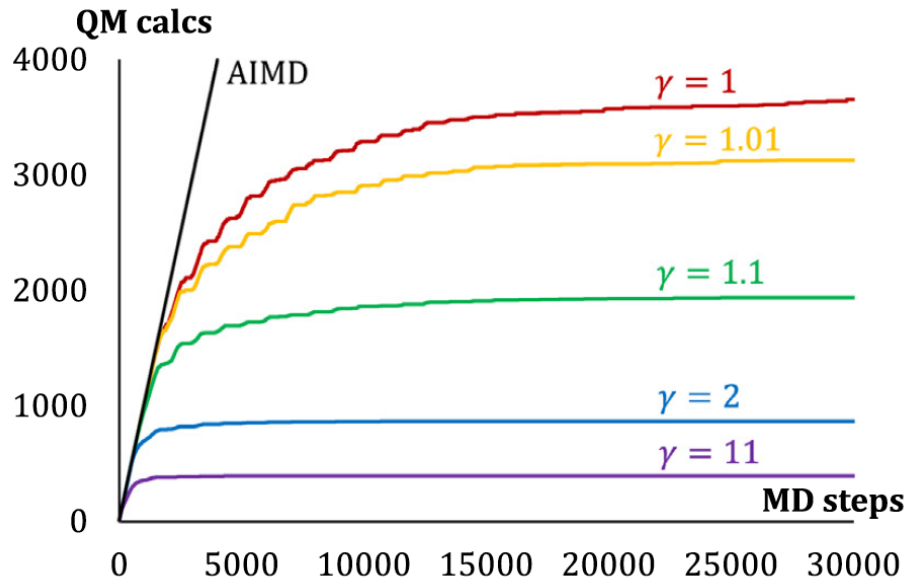
Extrapolation grade correlates with actual error

- Atomistic simulations of Lithium
- Each point is an MD time step



Comparison (of hyperparameters) studies

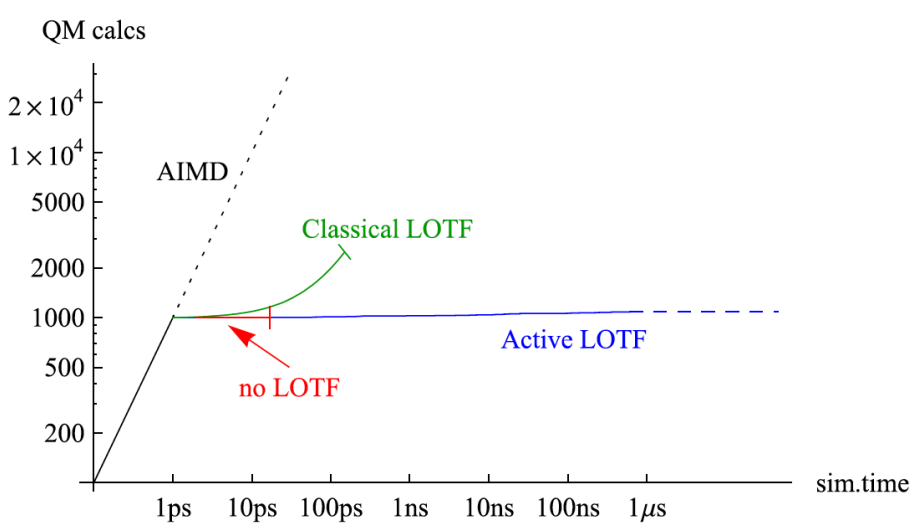


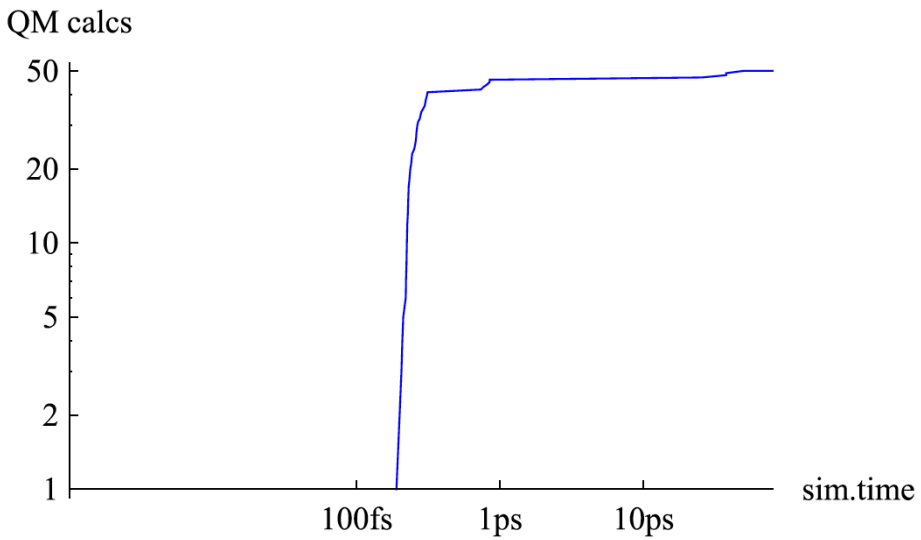


Based on our experience, we find that a value for γ_{thr} between 2 and 11 is a good choice in practice: it does not significantly reduce the accuracy, while the number of the QM calculations is just a few times higher than the theoretical minimum (which is equal to the number of undetermined parameters).

AL reduces the QM calculation count

If a potential is trained on a fixed database, it is observed that once in about 15ps the atomistic system escapes into an unphysical region characterized by very low (below 1 A) bond lengths. Therefore, to assess the reliability of a potential, we terminate the MD if after some simulation time the minimal distance between atoms becomes smaller than 1.5 A. We call the simulation time after which half the trajectories are terminated (i.e., the trajectory half-life), the failure time. From the transition state theory, we estimate that in an AIMD, the failure time is of the order of 10^{10} s, which is much larger than is accessible even with a classical MD.



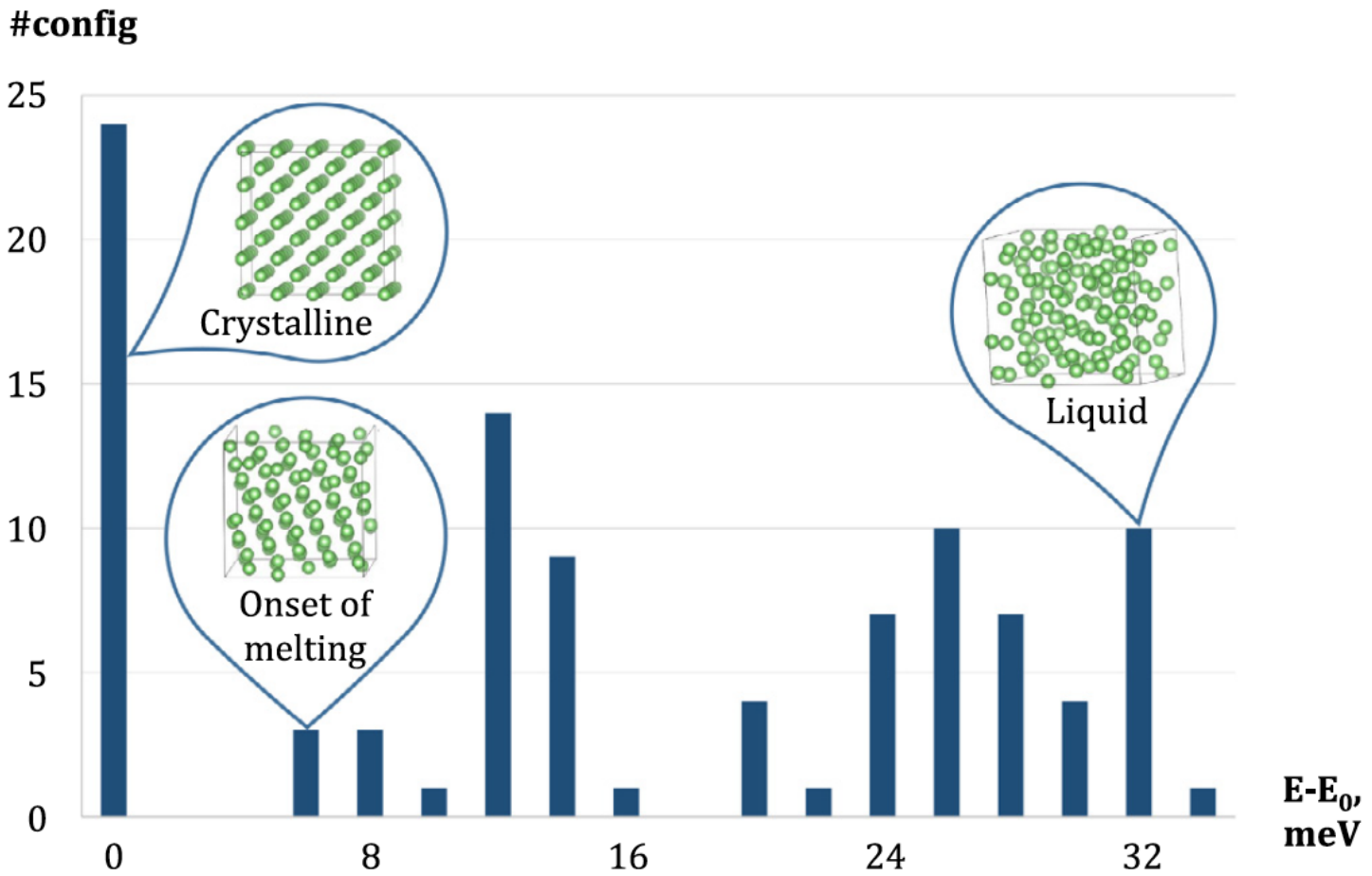


Train at one temperature and predict at another

Potential	Force error at:		
	300 K	450 K	900 K
MTP_{300}	0.016 (4.6%)	0.022 (5.7%)	12.1
MTP_{450}	0.017 (4.7%)	0.020 (5.2%)	11.7
MTP_{900}	0.030 (8.4%)	0.033 (8.5%)	0.062 (7.0%)

Relaxation to known configurations

Found all the benchmark configurations when relaxing the active-learned MTPs.



Errors of fitting of different AL approaches

Query strategy	$\#X_{TS}$	Energy error (meV/atom)		Force error (eV/Å)		Stress error (GPa)	
		RMS	Max	RMS	Max	RMS	Max
Passive	24000	0.19	0.82	0.016	0.13	0.052	0.13
Random	100	0.21	0.92	0.017	0.28	0.057	0.14
QS_1	100	0.21	0.78	0.016	0.10	0.053	0.13
QS_2	84	0.25	0.89	0.016	0.10	0.056	0.13
QS_3	92	0.23	0.79	0.016	0.09	0.057	0.13

Reliability/transferability

Average failure time, i.e., simulation time until some bond is compressed to 1.5 Å.

Query strategy	$\#X_{TS}$	Failure time (ns)	
		$T = 300$ K	$T = 450$ K
Passive	24000	0.18	0.06
Random	100	0.17	0.03
QS_1	100	0.66	0.04
QS_2	84	1.29	0.06
QS_3	92	3.84	0.16

Finally, we have performed a test of reliability when the potentials are trained offline. We use the potentials from the previous test fitted on MD trajectories at T=300 K and use them in MD at T=300 K and T=450 K. We measure the failure time, i.e., simulation time until the minimal interatomic distance becomes less then 1.5 A. We performed 100 MD runs and calculated the expected failure time for different MTPs.

Summary/Discussion

- Does not reduce the accuracy of interatomic potentials while always keeping the MD trajectory within the physical region
- Overhead of algorithm and retraining is small compared to QM calculation
- Is not MD-specific - can be applied to, e.g., structure relaxation, Monte-Carlo sampling, nudged elastic band etc..
- Detects when extrapolation is attempted and retrains the potential on those configurations
- Controls the degree of extrapolation

Summary/Discussion - cont'd

- One can apply AL to atomistic systems with any number of chemically different types of atoms, however, most linearly parametrized potentials developed to date are only applicable to systems with a single type of atoms.
- In addition, we show that even without learning on the fly, AL can optimize the training set, in the sense of extracting a significantly smaller subset, training on which reduces the maximal error and improves transferability.
- This query strategy is based on geometrical information (atomic positions and supercell vectors) of a configuration and does not use the QM data, thus a well-trained potential will trigger the QM calculations very rarely.
- Code is here: http://gitlab.skoltech.ru/shapeev/mlip/