Applications of Machine Learning in Molecular Dynamics

Yuan Hu

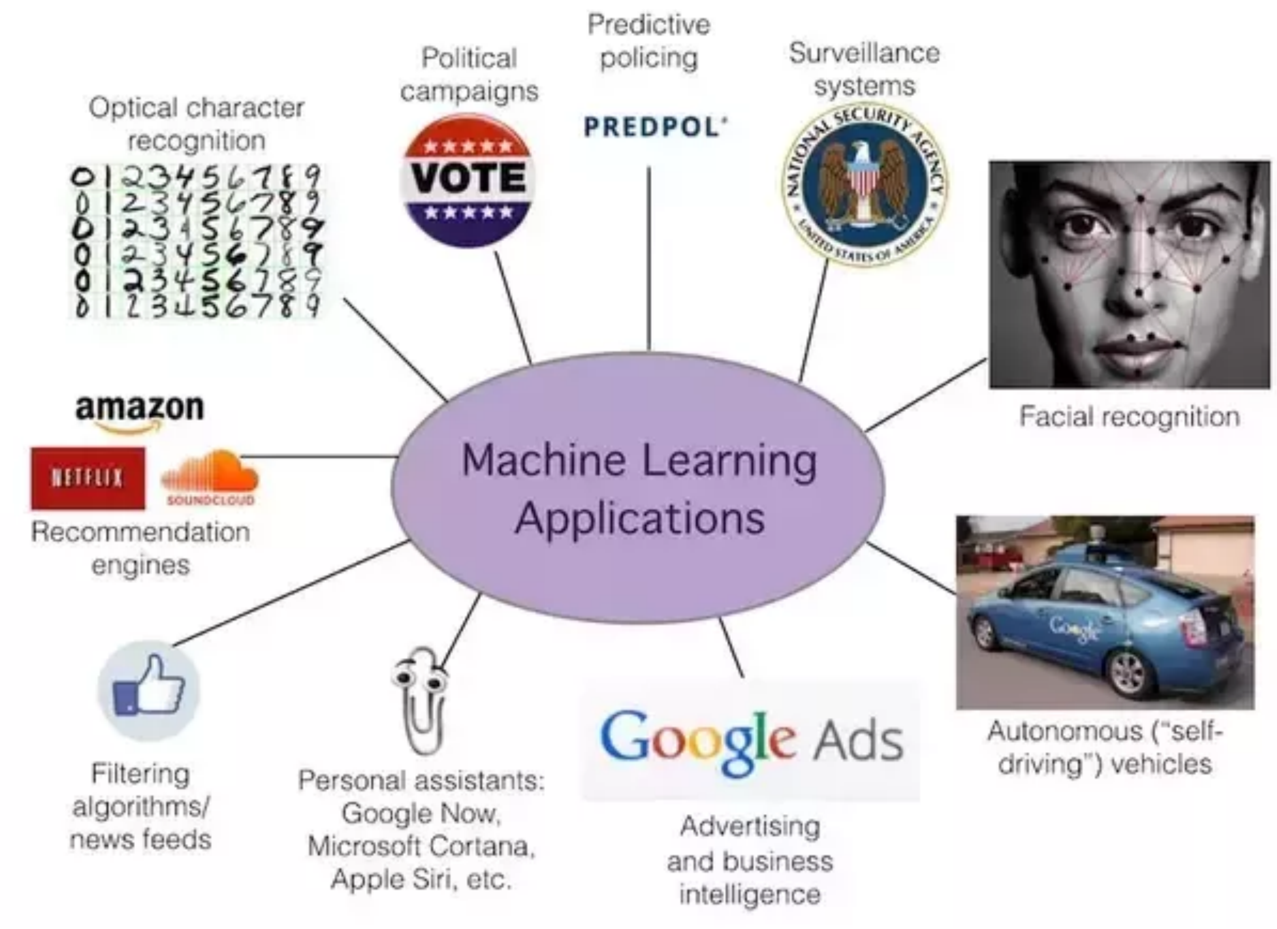
Shichun Wang

Size Zheng

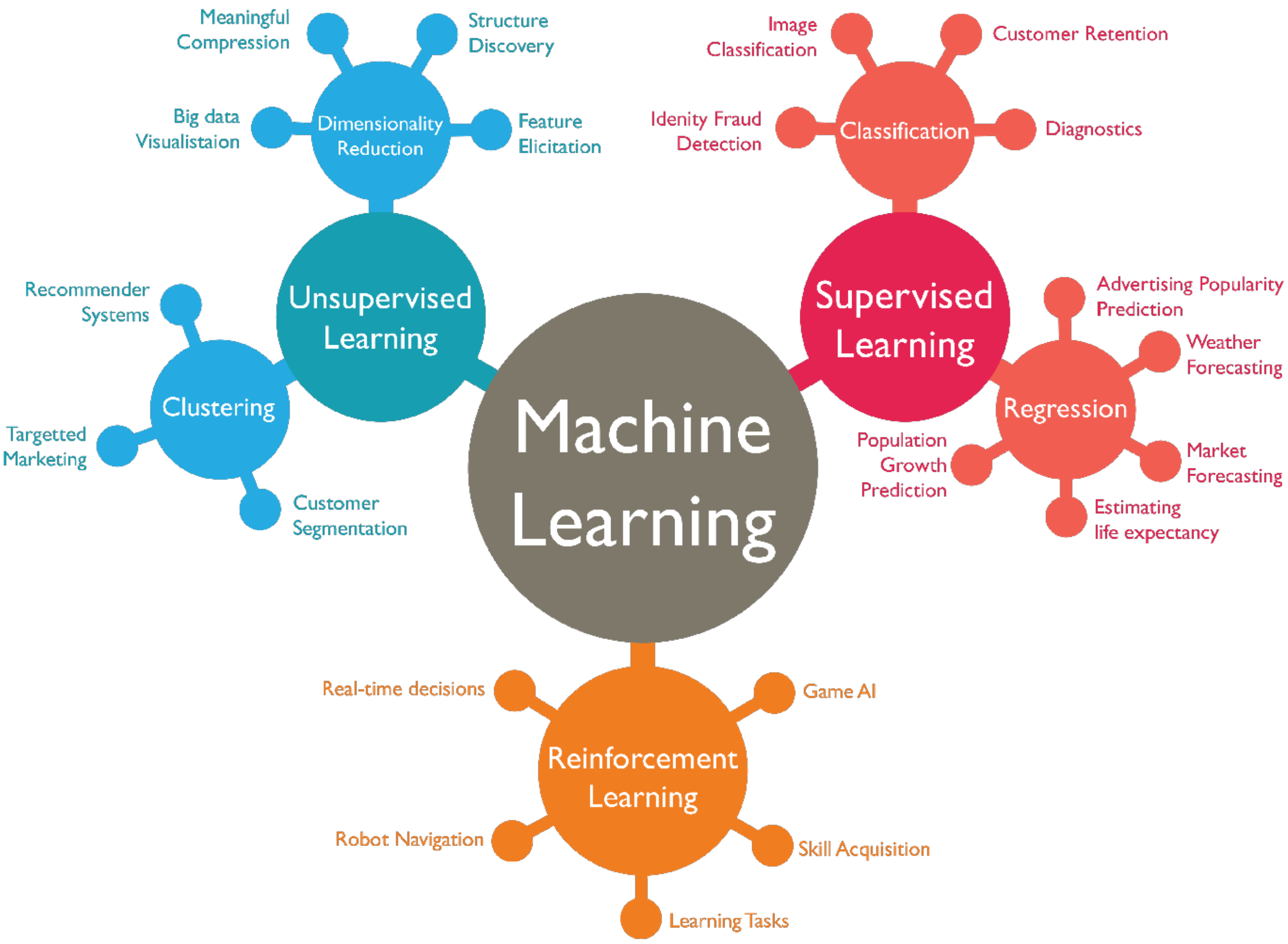
USC
School of Engineering

Mork Family Department of Chemical Engineering & Materials Science

University of Southern California


Machine Learning Applications
Optical character recognition
Political campaigns
VOTE
Predictive policing
PREDPOL
Surveillance systems
NATIONAL SECURITY AGENCY
UNITED STATES OF AMERICA
Facial recognition
amazon
NETFLIX
SOUNDCLOUD
Recommendation engines
Autonomous ("self-driving") vehicles
Google
Filtering algorithms/ news feeds
Personal assistants: Google Now, Microsoft Cortana, Apple Siri, etc.
Google Ads
Advertising and business intelligence

https://redshiftzero.github.io/assets/manip/ml_applications.png

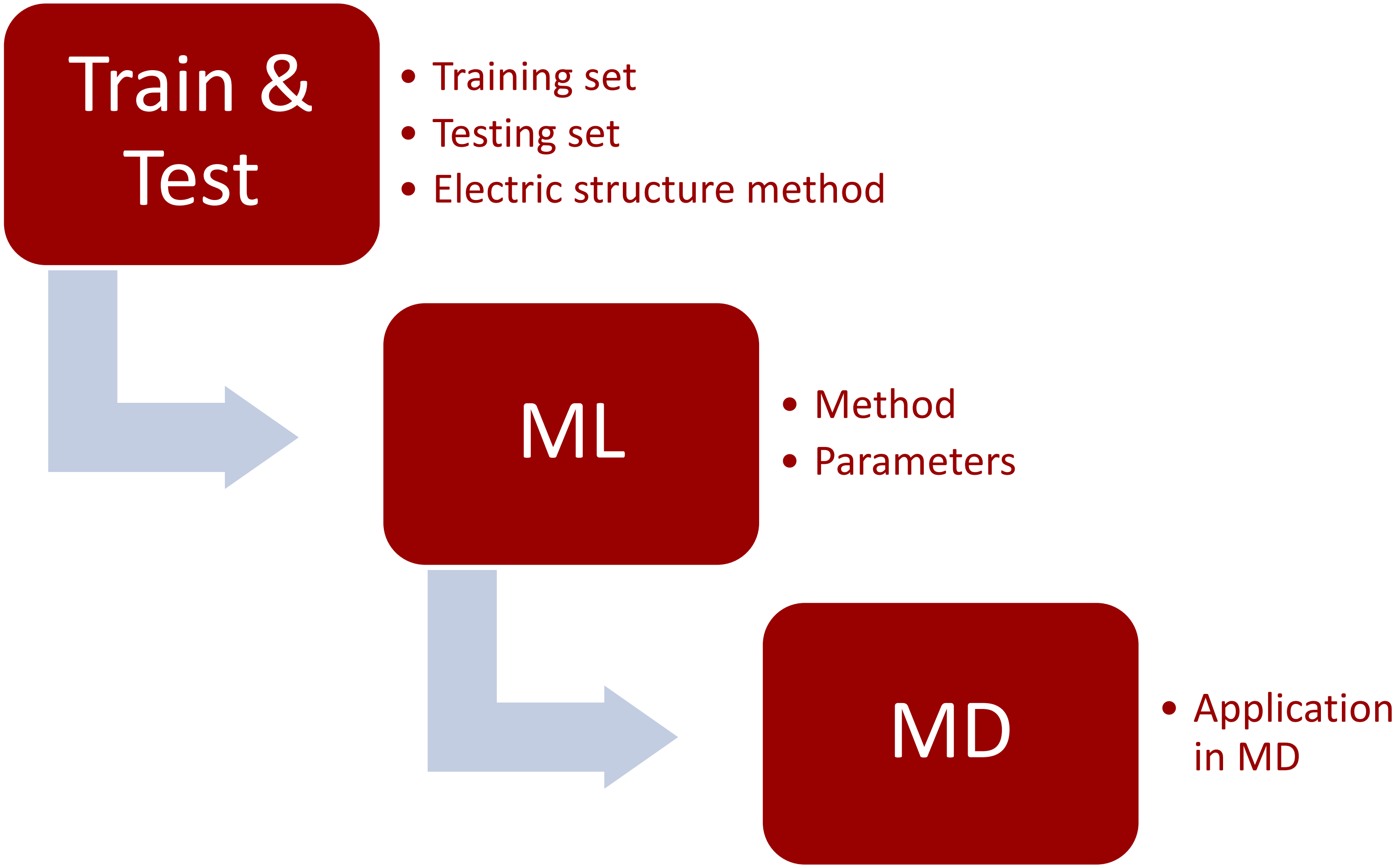


https://www.wordstream.com/images/machine-learning.png

In *ab initio* MD, the energies and forces need to be calculated “on-the-fly”: very expensive

- Energies can be fitted to high accuracy with very small remaining errors compared to the underlying reference data
- Energies can be calculated efficiently and require much less CPU time than electronic structure calculations.
- No knowledge about the functional form of the PES is required.
- The energy expression is unbiased, generally applicable to all types of bonding and does not require system-specific modification

- Neural Network Potential (NNP)
- Kernel Ridge Regression (KRR)
- Gaussian Approximation Potential (GAP)



Train & Test

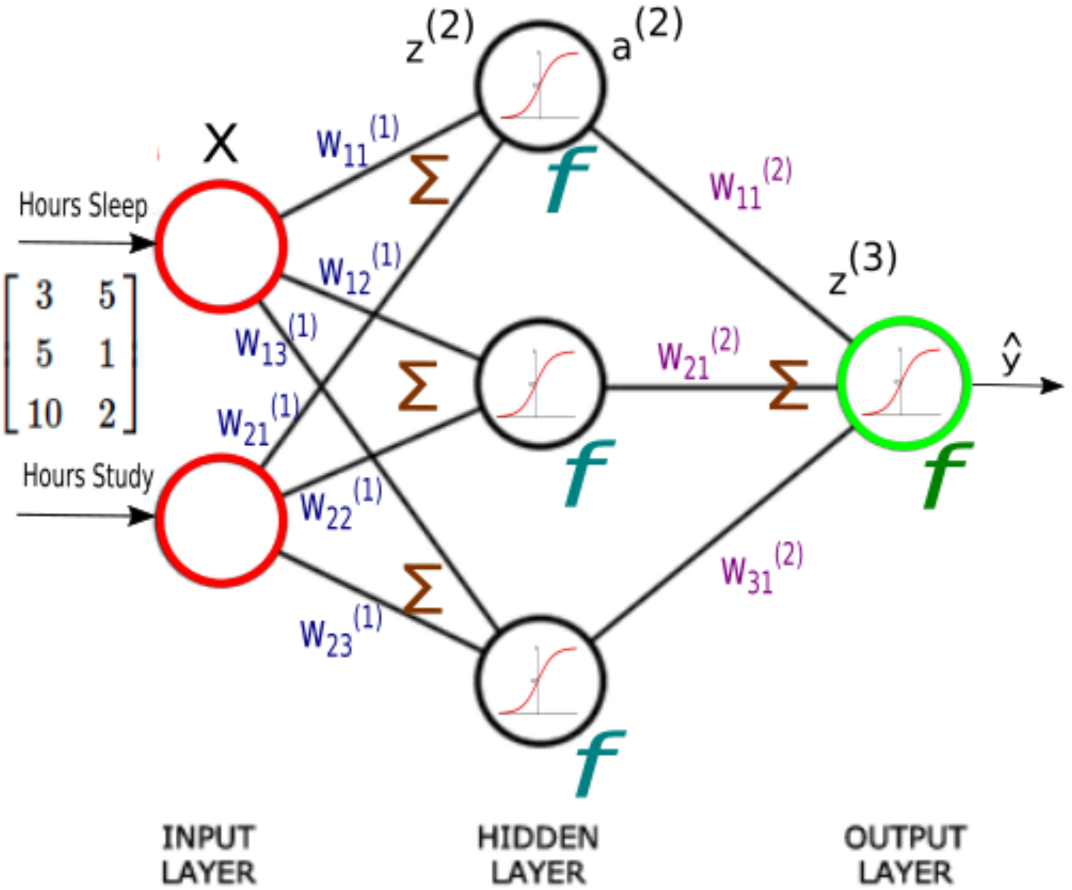
- Training set
- Testing set
- Electric structure method

ML

- Method
- Parameters

MD

- Application in MD



Forwarding

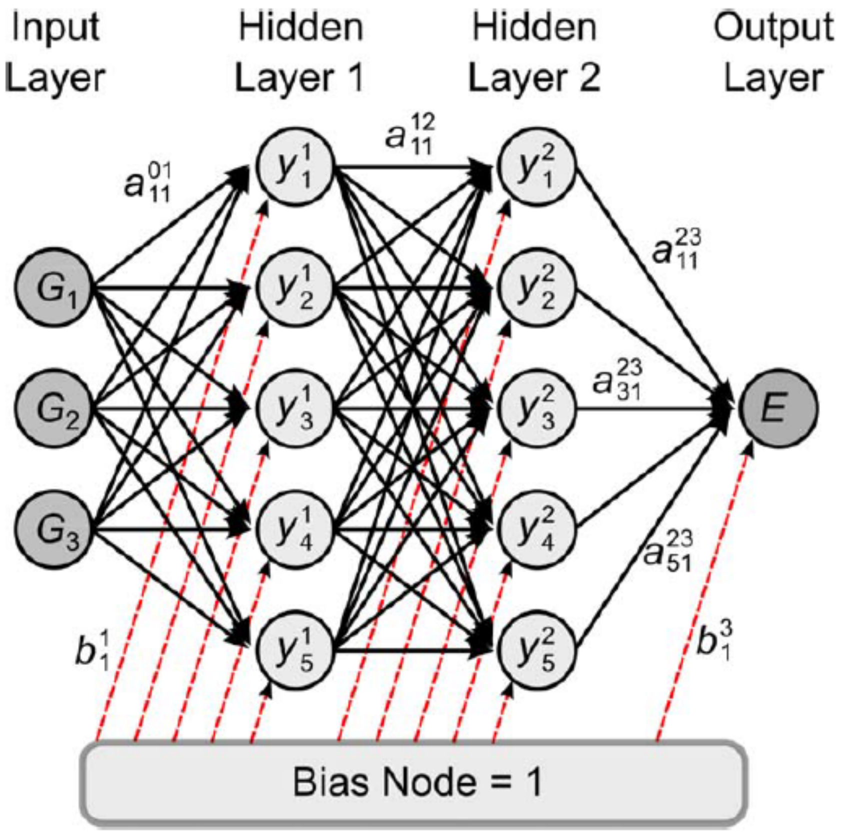
$$O = \text{activation function}(W \cdot I)$$

Backpropagating

$$\frac{\partial E}{\partial w_{jk}} = -2(t_k - o_k) \cdot \frac{\partial}{\partial w_{jk}} \text{activation function}\left(\sum_j w_{jk} \cdot o_j\right)$$

$$\text{new } w_{jk} = \text{old } w_{jk} - \alpha \frac{\partial E}{\partial w_{jk}}$$

http://www.bogotobogo.com/python/scikit-learn/images/NeuralNetwork1/NN-with-components-w11-etc.png

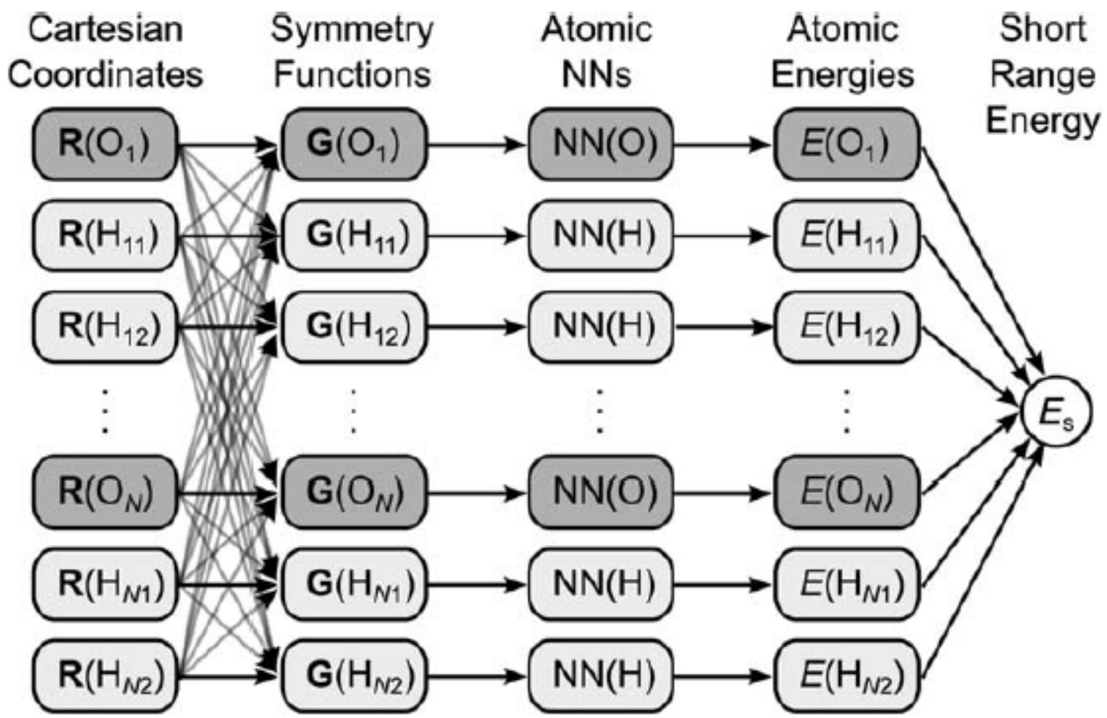


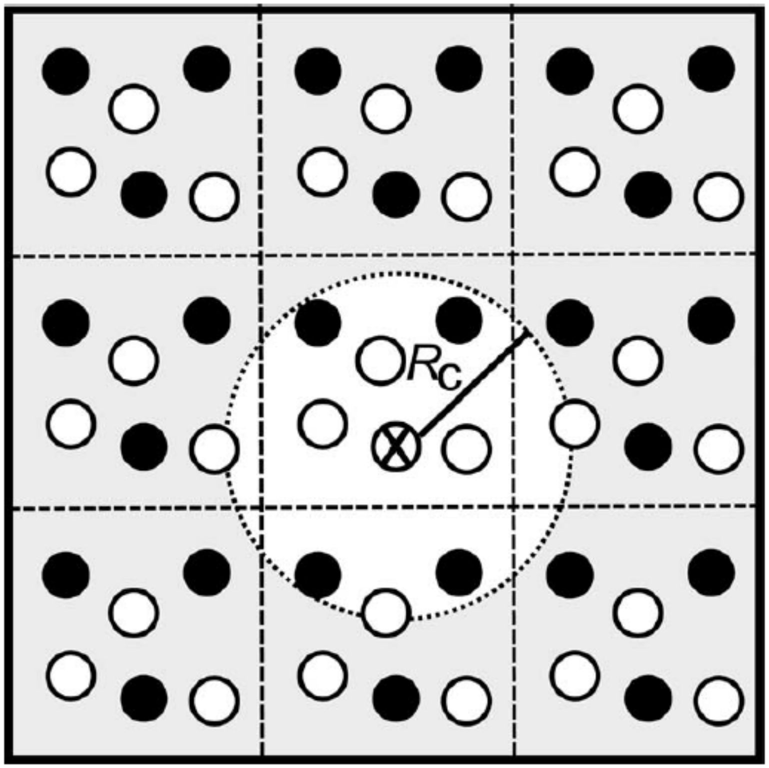
$$y_i^j = f_i^j \left(b_i^j + \sum_{k=1}^{N_{j-1}} a_{k,i}^{j-1,j} \cdot y_k^{j-1} \right)$$

Jorg Behler, *International Journal of Quantum Chemistry* 2015, *115*, 1032–1050

For High-dimensional NNPs, using a single NN for the construction of high dimensional NNPs is impossible.

- Too many input nodes make the construction of the NNP inefficient
- Symmetry of the NN
- Not scalable





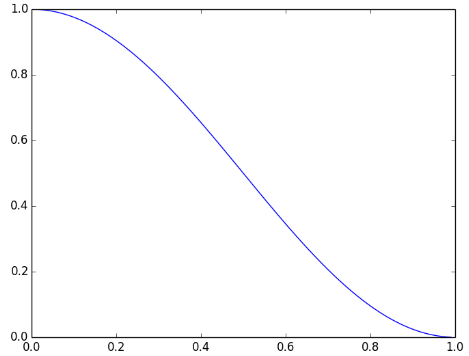
$$E = E_s + E_{\text{elec}} = \sum_{i=1}^{N_{\text{atom}}} E_i + E_{\text{elec}}.$$

Jorg Behler, *International Journal of Quantum Chemistry* 2015, *115*, 1032–1050

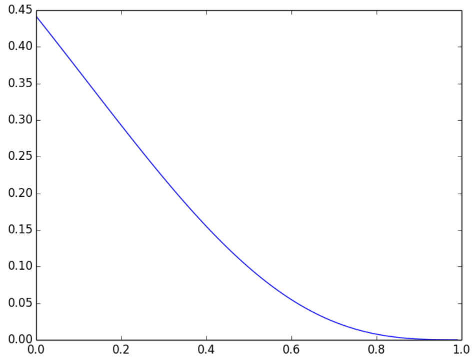
Symmetry Function

The cut-off function f_c

$$f_{c,1}(R_{ij}) = \begin{cases} 0.5 \cdot \left[\cos\left(\frac{\pi R_{ij}}{R_c}\right) + 1\right] & \text{for } R_{ij} \le R_c \\ 0.0 & \text{for } R_{ij} > R_c \end{cases}$$



$$f_{c,2}(R_{ij}) = \begin{cases} \tanh^3\left[1 - \frac{R_{ij}}{R_c}\right] & \text{for } R_{ij} \le R_c \\ 0.0 & \text{for } R_{ij} > R_c \end{cases}$$



Symmetry Function

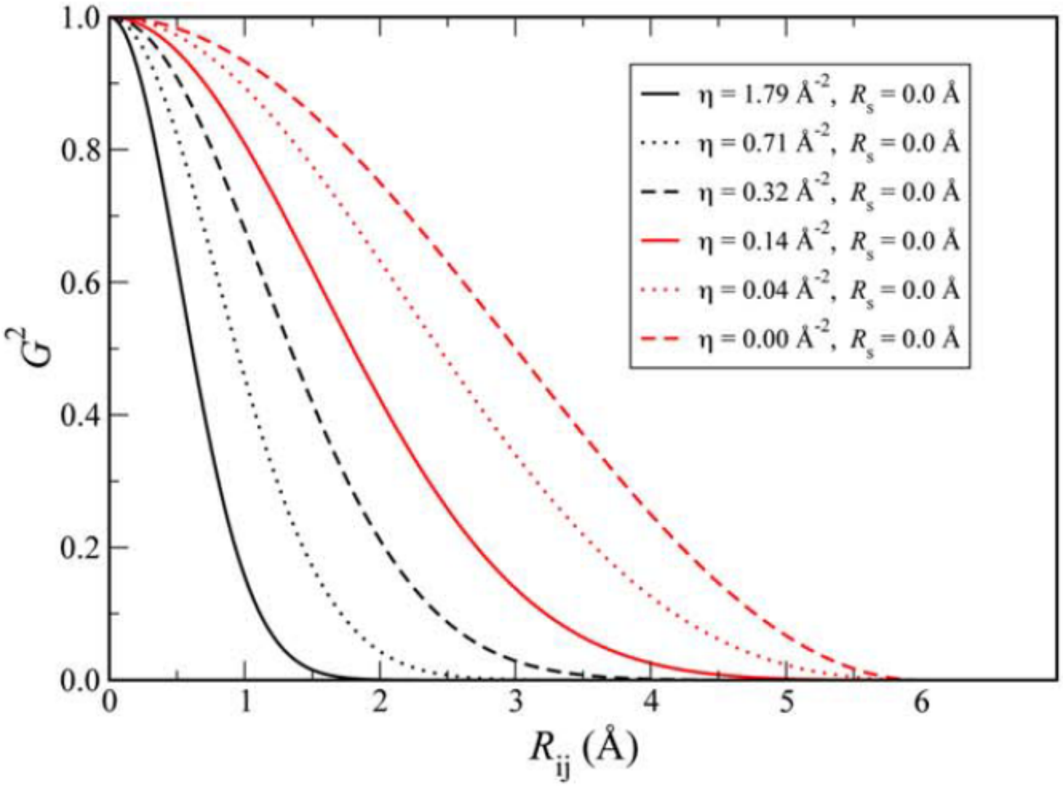
$$G_i^1 = \sum_{j=1}^{N_{\text{atom}}} f_c(R_{ij})$$

$$G_i^2 = \sum_{j=1}^{N_{\text{atom}}} e^{-\eta\left(R_{ij}-R_s\right)^2} \cdot f_c(R_{ij})$$

$$G_i^3 = 2^{1-\zeta} \sum_{j \neq i} \sum_{k \neq i,j} \left[\left(1 + \lambda \cdot \cos\theta_{ijk}\right)^{\zeta} \cdot e^{-\eta\left(R_{ij}^2+R_{ik}^2+R_{jk}^2\right)} \cdot f_c(R_{ij}) \cdot f_c(R_{ik}) \cdot f_c(R_{jk})\right]$$

$$G_i^4 = 2^{1-\zeta} \sum_{j \neq i} \sum_{k \neq i,j} \left[\left(1 + \lambda \cdot \cos\theta_{ijk}\right)^{\zeta} \cdot e^{-\eta\left(R_{ij}^2+R_{ik}^2\right)} \cdot f_c(R_{ij}) \cdot f_c(R_{ik})\right]$$

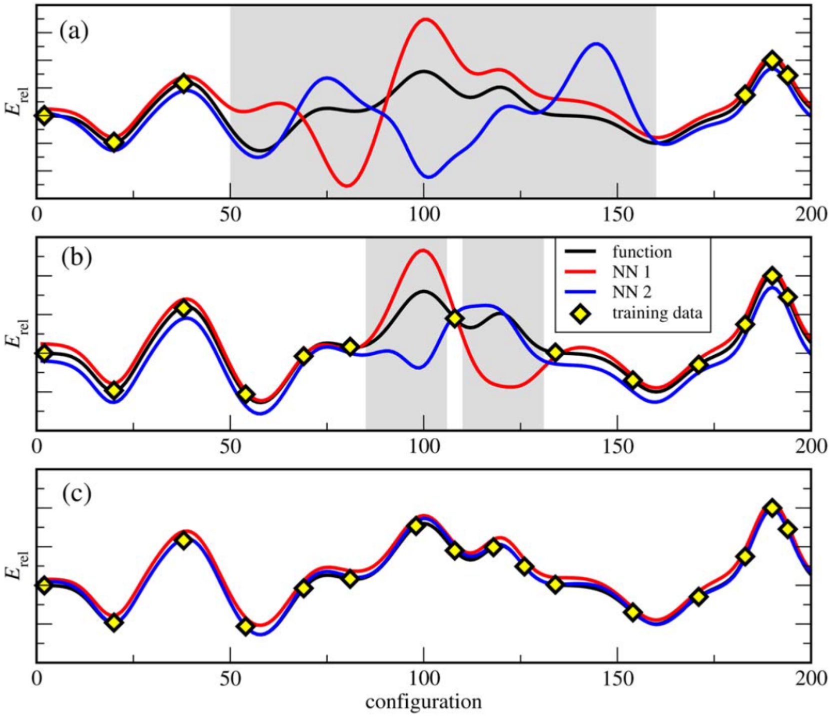
Symmetry Function - Example



$$G_i^2 = \sum_{j=1}^{N_{\text{atom}}} e^{-\eta\left(R_{ij}-R_s\right)^2} \cdot f_c(R_{ij})$$

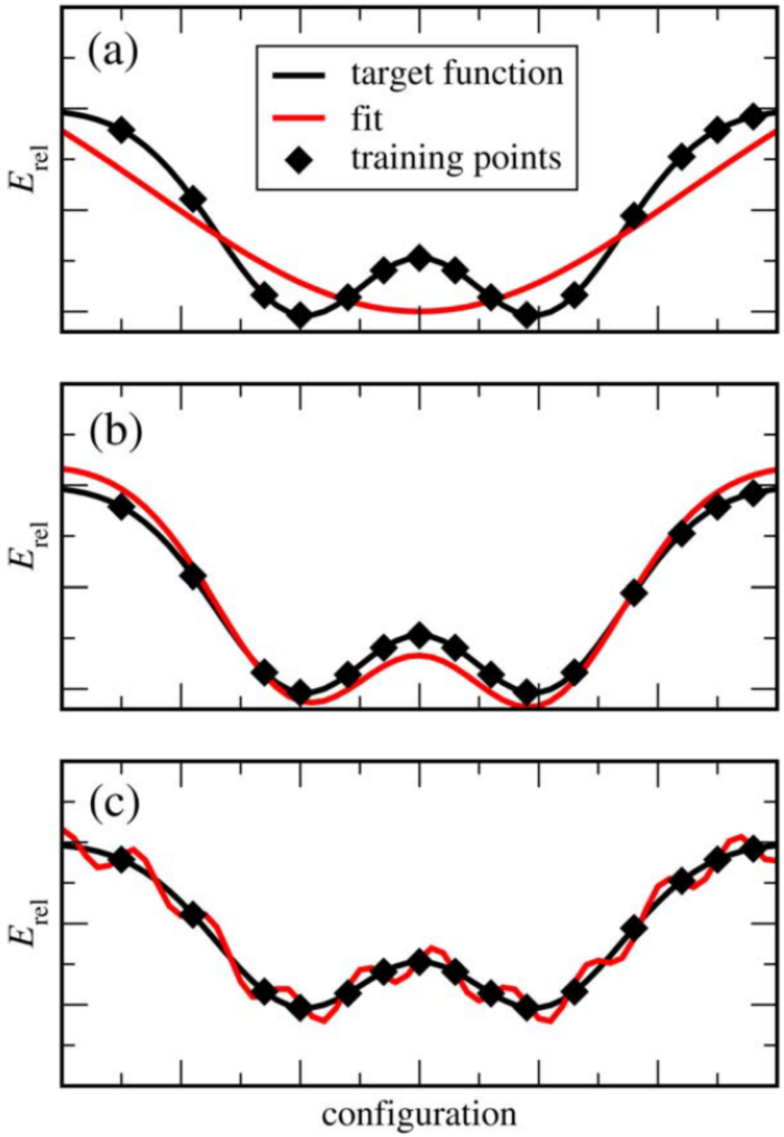
Jorg Behler, *International Journal of Quantum Chemistry* 2015, *115*, 1032–1050

Training NNs



1. Select an electronic structure method.
2. Define a first set of structures and determine the energies and forces.
3. Construct a first preliminary NNP
4. Carry out simulations using this NNP to find structures, give rise to extrapolation warnings or unphysical geometries.
5. Determine the electronic structure energies and forces of these structures, include them in the training set and improve the NNP.
6. Improve the NNP systematically and self-consistently by running NNP-based simulations using the multiple-NNP method.
7. As soon as no further warnings, the NNP is ready to use.

Jorg Behler, *International Journal of Quantum Chemistry* 2015, *115*, 1032–1050



Weight Parameters

- Random numbers
- Different optimization algorithms to improve them

The architecture of the NNs

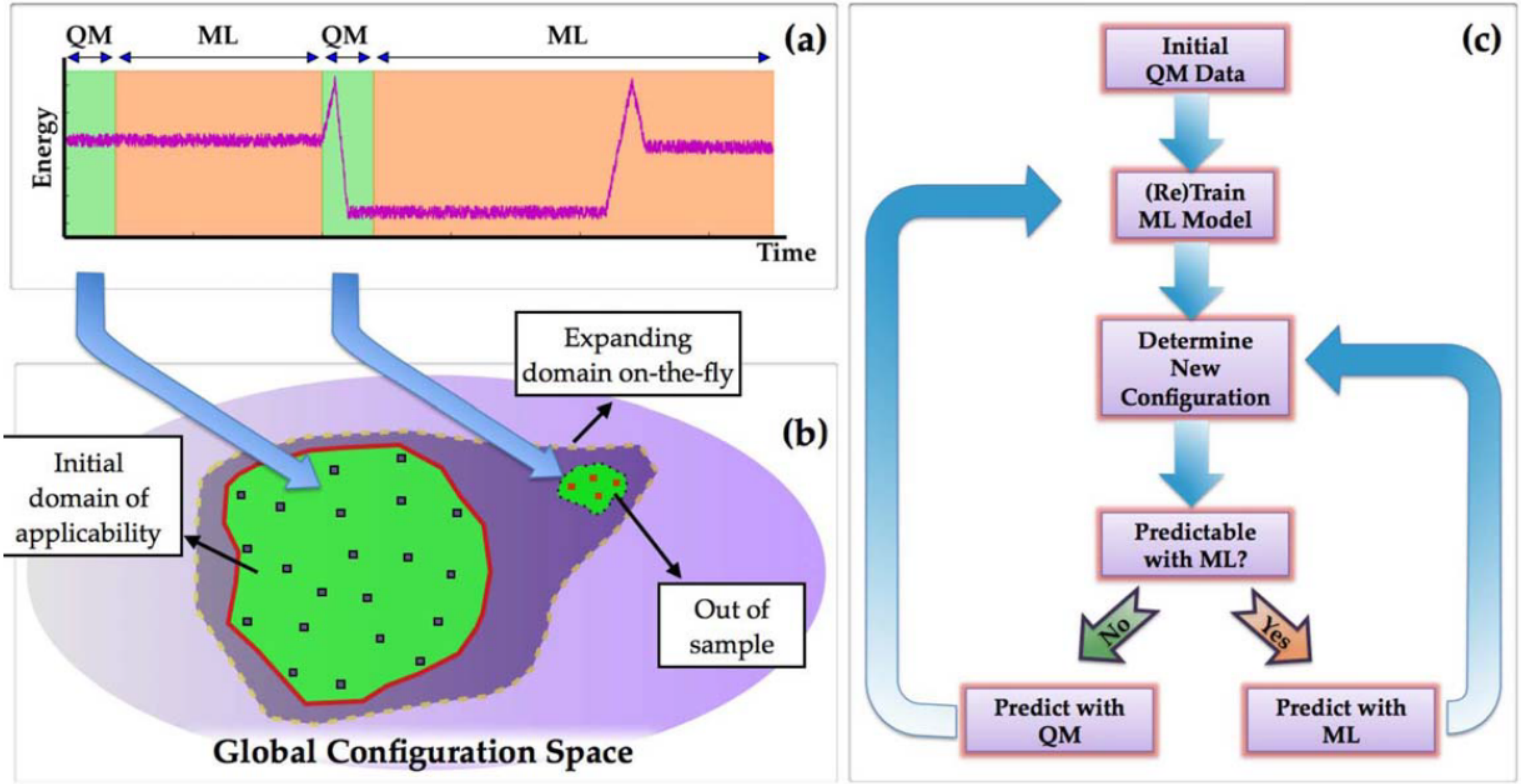
- Empirical
- Identify a suitable number of hidden layers and nodes per layer is simply to carry out a number of fits and to select the one with the lowest errors of the energies and forces in the test set.

Jorg Behler, *International Journal of Quantum Chemistry* 2015, *115*, 1032–1050

Disadvantages:

- The evaluation of NNPs is notably slower than the use of simple classical force fields.
- NNPs have no physical basis and only very limited extrapolation capabilities.
- The construction of NNPs requires substantial effort, and a large number of training points from electronic structure calculations is required.
- Currently, NNPs are limited to systems containing either only a few different chemical elements but many atoms or a small number of atoms with arbitrary nuclear charges.

Adaptive Machine Learning Framework



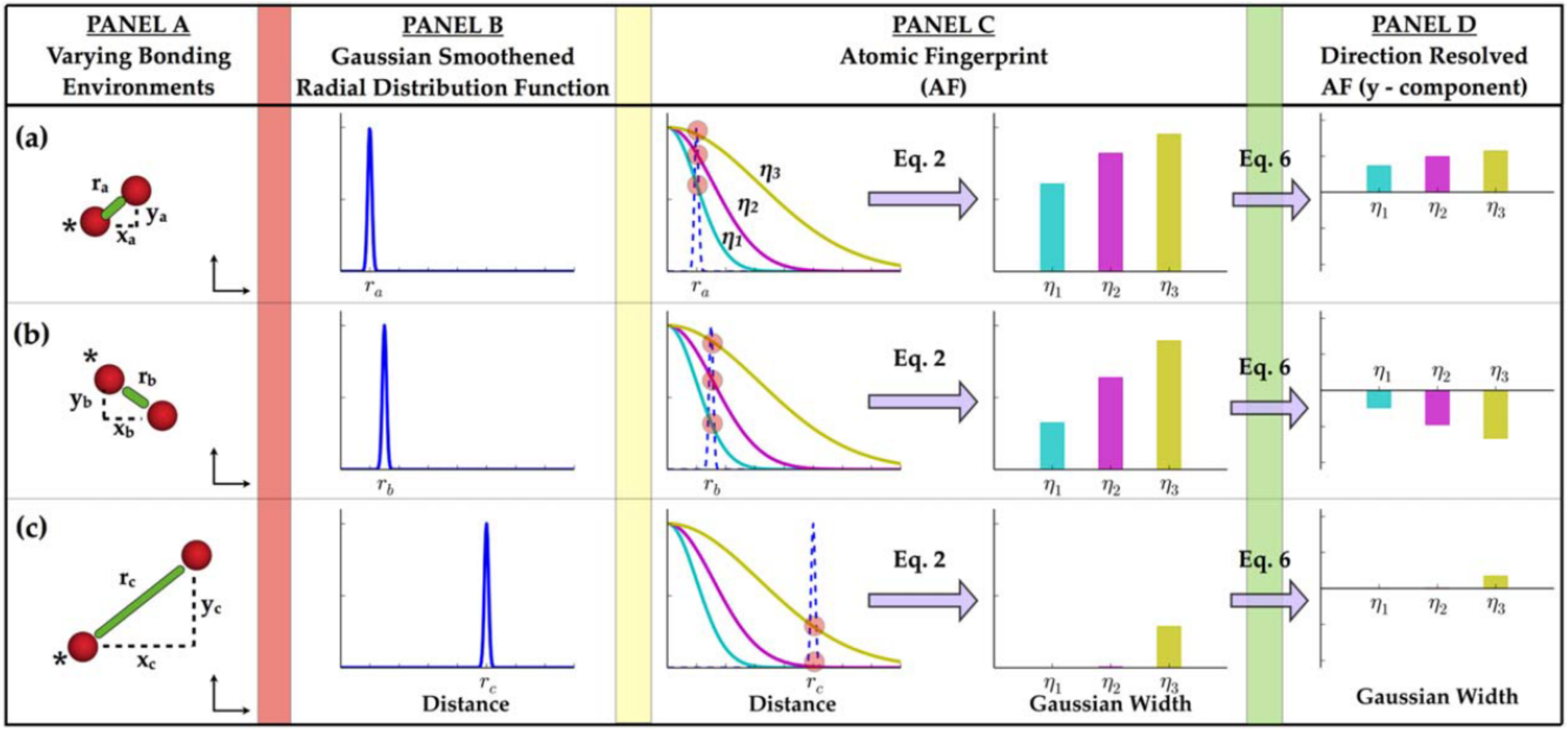
The significant redundancies implicit in conventional ab initio MD schemes can be systematically eliminated.

V. Botu, R. Ramprasad, *International Journal of Quantum Chemistry* 2015, *115*, 1074–1083

- To simulate 32 atom bulk Al, to use ML method, each prediction takes roughly a millisecond.
- For comparison, the same case with DFT will take about 45 min on a 16 core machine, a speed up on the order of 10^6!

V. Botu, R. Ramprasad, *International Journal of Quantum Chemistry* 2015, *115*, 1074–1083

Symmetry Function



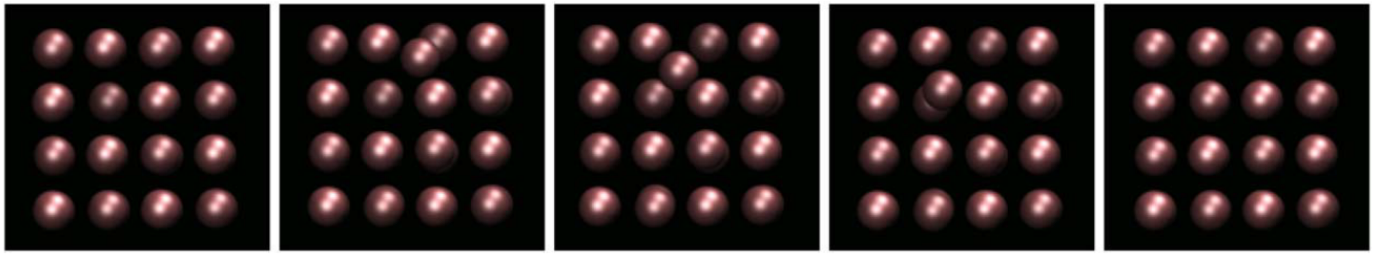
$$V_i^k(\eta) = \sum_{j\neq i}\left[\frac{r_{ij}^k}{r_{ij}} e^{-\eta(R_{ij}-R_s)^2} \cdot f_c(R_{ij})\right], k \in \{x, y, z\}$$

V. Botu, R. Ramprasad, *International Journal of Quantum Chemistry* 2015, *115*, 1074–1083

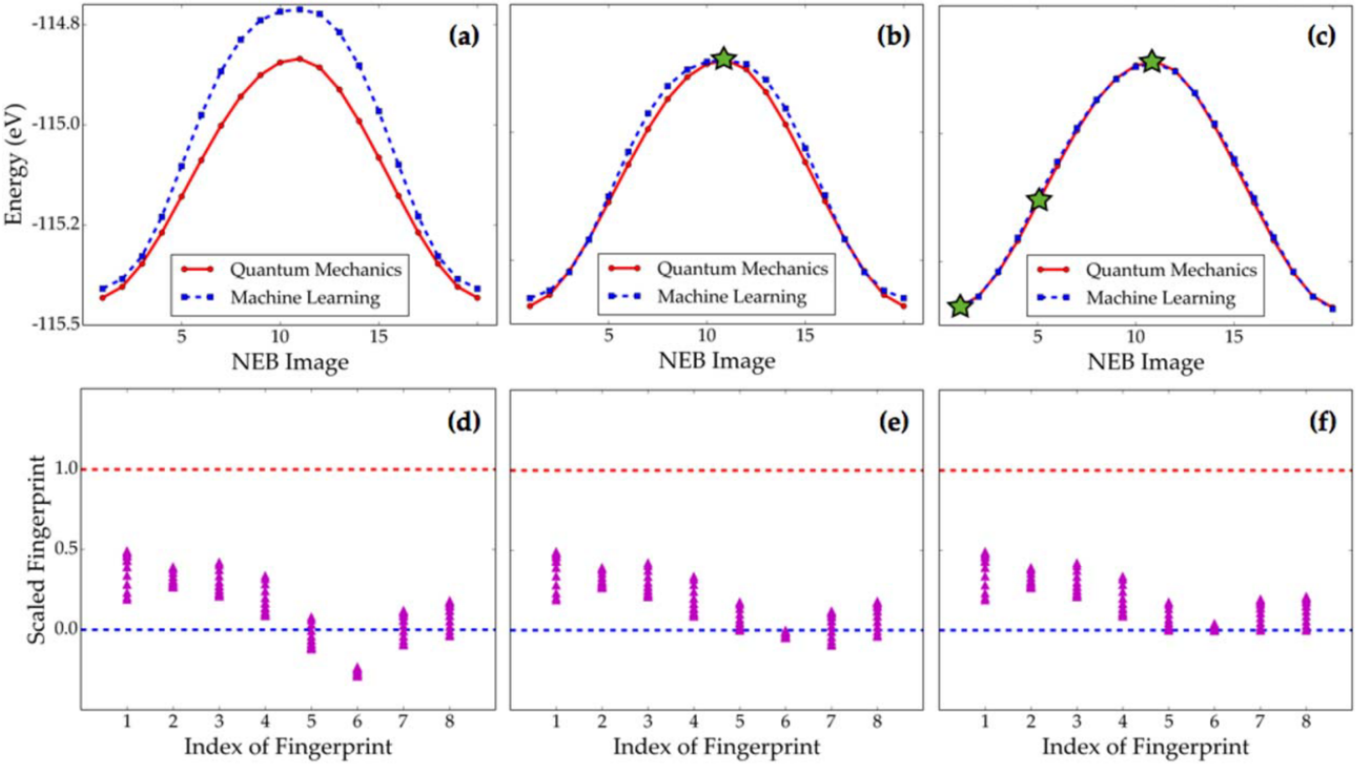
Machine Learning Method – Kernel Ridge Regression

KRR transforms the input fingerprint into a higher dimensional space whereby a linear relation between the transformed fingerprint and the property of interest can be established. To be precise, the mapping process between the fingerprint and property involves the “distances” between fingerprints rather than the fingerprints themselves. KRR may thus be viewed as a similarity-based learning method, that is, similar fingerprints will lead to similar properties.

$$P_u = \sum_v \alpha_v e^{-\frac{1}{2}\left(\frac{|d_{uv}|}{\sigma}\right)^2}$$



Vacancy migration within bulk Al correspond to steps 1, 5, 10, 15, and 20.



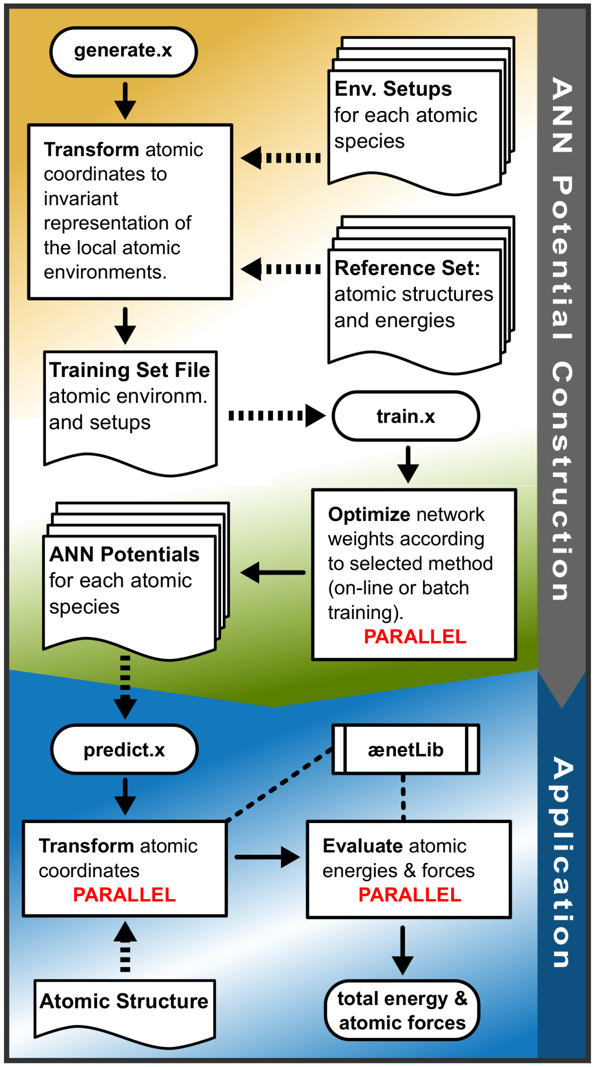
a) and d) with no retraining, b) and e) with the TS added to training and c) and f) with TS and image 1 and 5 added to the training.

V. Botu, R. Ramprasad, *International Journal of Quantum Chemistry* 2015, *115*, 1074–1083

Introduction to Atomic Energy Network (aenet) Package

Atomic Energy Network - aenet

- Open source http://ann.atomistic.net
- Potential construction tools
  - generate.x – preprocessing
  - train.x – train ANN force field
- Application tools
  - predict.x – get forces from atomic coordinates
  - aenetLib – interface to other program



ANN Potential Construction

generate.x

Transform atomic coordinates to invariant representation of the local atomic environments.

Env. Setups for each atomic species

Reference Set: atomic structures and energies

Training Set File atomic environm. and setups

train.x

Optimize network weights according to selected method (on-line or batch training). PARALLEL

ANN Potentials for each atomic species

Application

predict.x

ænetLib

Transform atomic coordinates PARALLEL

Evaluate atomic energies & forces PARALLEL

Atomic Structure

total energy & atomic forces

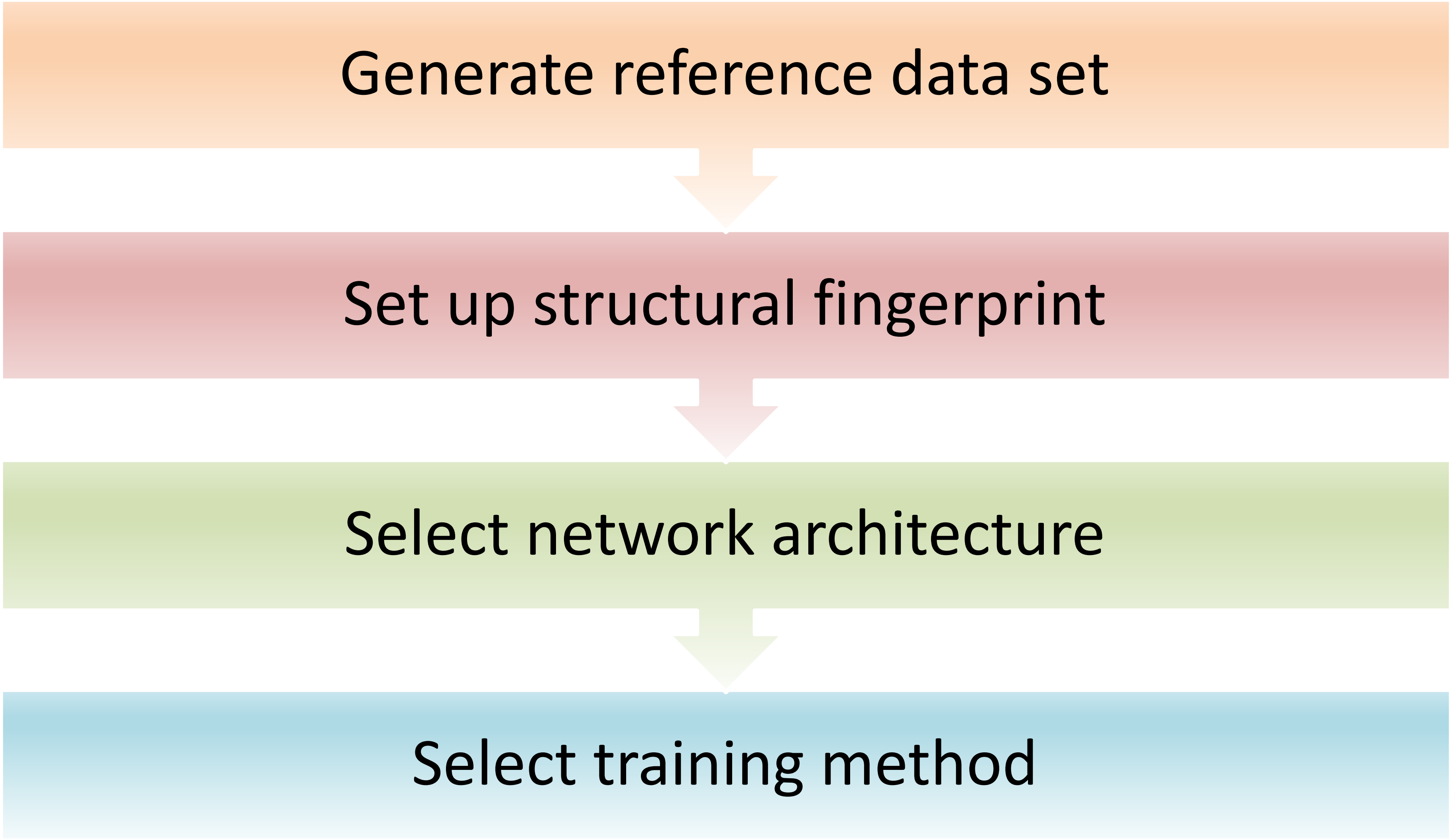
Train neural network force field

- Activation functions

linear function	$f_a^1(x) = x,$
hyperbolic tangent	$f_a^2(x) = \tanh(x) = \frac{1-e^{-2x}}{1+e^{-2x}},$
logistic function	$f_a^3(x) = \frac{1}{1+e^{-x}},$ and
tanh with linear twisting	$f_a^4(x) = 1.7159 \tanh\left(\frac{2}{3}x\right) + ax,$

- Learning methods – batch and online
- Optimization methods – gradient descent, limited-memory BFGS, Levenberg-Marquardt

Train an ANN Force Field (TiO_2) via aenet



Generate reference data set

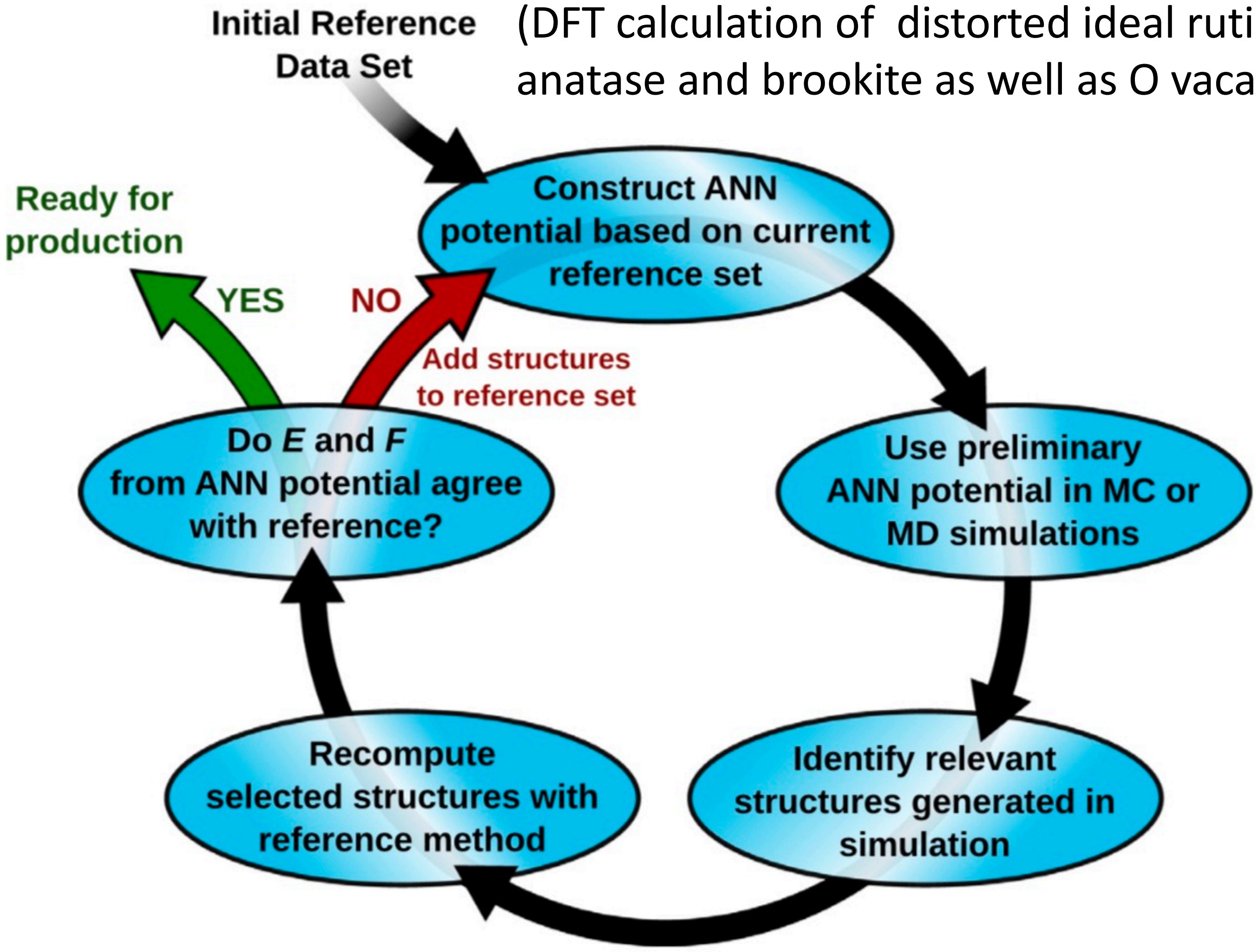
Set up structural fingerprint

Select network architecture

Select training method

Generate reference data set

(DFT calculation of distorted ideal rutile, anatase and brookite as well as O vacant structure)



Initial Reference Data Set
Construct ANN potential based on current reference set
Use preliminary ANN potential in MC or MD simulations
Identify relevant structures generated in simulation
Recompute selected structures with reference method
Do *E* and *F* from ANN potential agree with reference?
YES
Ready for production
NO
Add structures to reference set

Set up structural fingerprint

- 8 radial and 18 angular Behler-Parrinello basis functions for each combination of atomic species
- 8x2 (O, Ti) radial basis functions
- 18x3(O-O, Ti-Ti, O-Ti) angular basis functions
- Total 70 basis functions or nodes at the input layer

Radial function – G^2

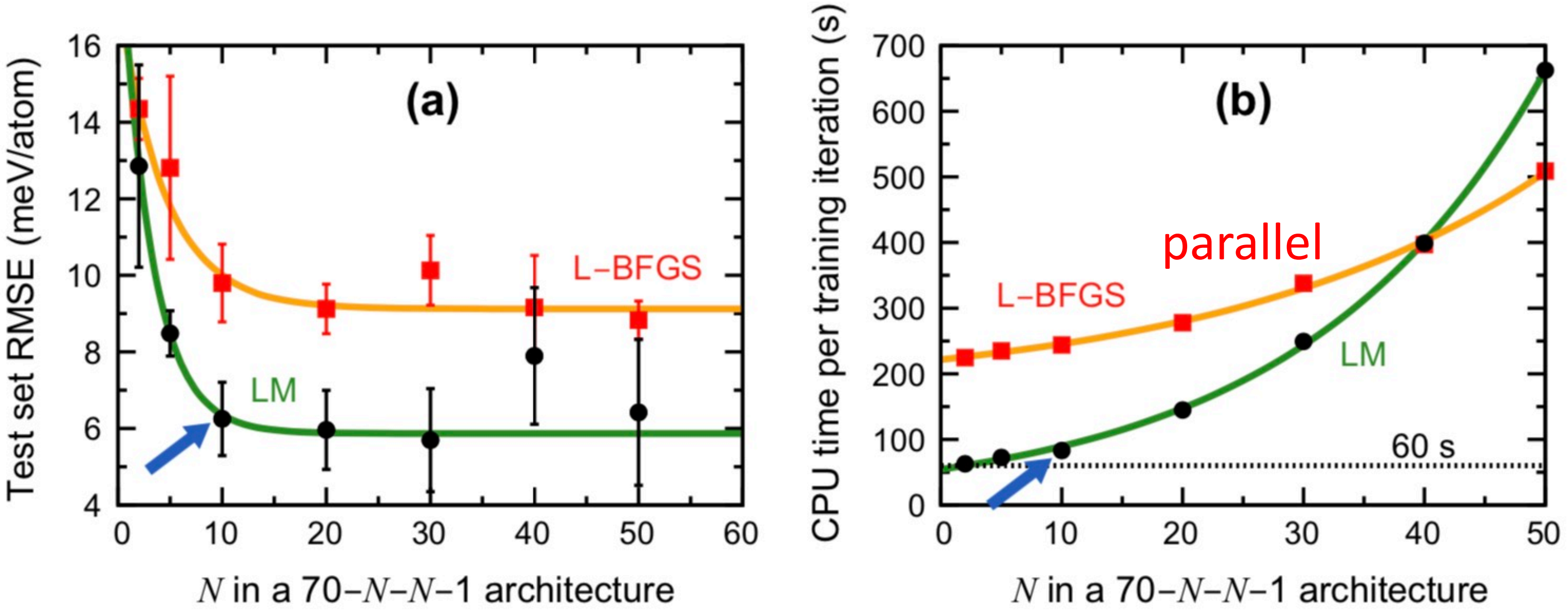
$$G_i^2 = \sum_{j \neq i} e^{-\eta (R_{ij} - R_s)^2} \cdot f_c(R_{ij}), \quad f_c(R_{ij}) = \begin{cases} 0.5 \left[\cos\left(\frac{\pi R_{ij}}{R_c} \right) + 1 \right] & \text{for } R_{ij} \leqslant R_c, \\ 0 & \text{for } R_{ij} > R_c. \end{cases}$$

Angular function – G^4

$$G_i^4 = 2^{1-\zeta} \sum_{j \neq i} \sum_{k \neq i,j} (1 + \lambda \cos \theta_{ijk})^{\zeta} \cdot e^{-\eta (R_{ij}^2 + R_{ik}^2 + R_{jk}^2)} \cdot f_c(R_{ij}) \cdot f_c(R_{ik}) \cdot f_c(R_{jk}),$$

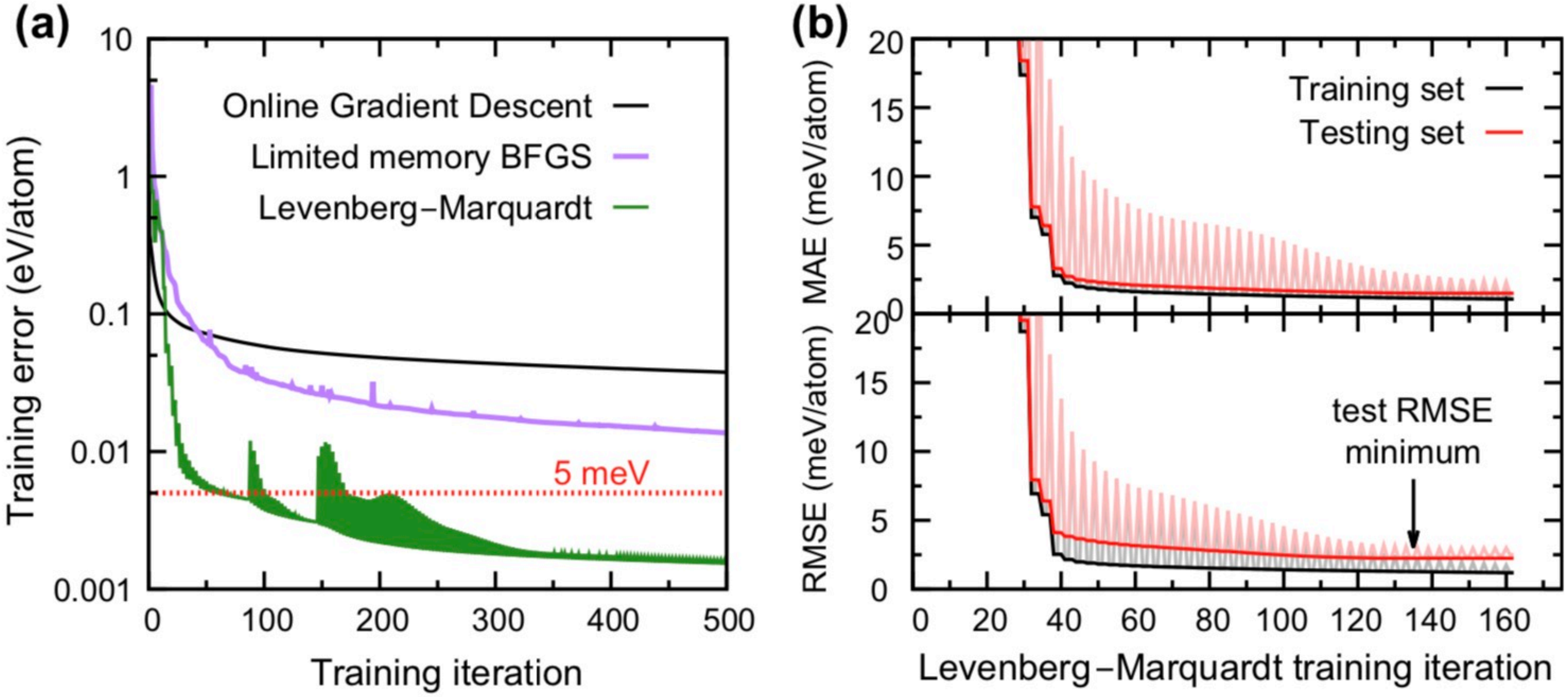
Select the best network architecture

- The balance between model complexity and transferability (predictive power)
- Network architecture: # of network layers + # of nodes per network layer
- Too simple → poor transferability
- Too complex → poor efficiency and possible overfitting (poor transferability)

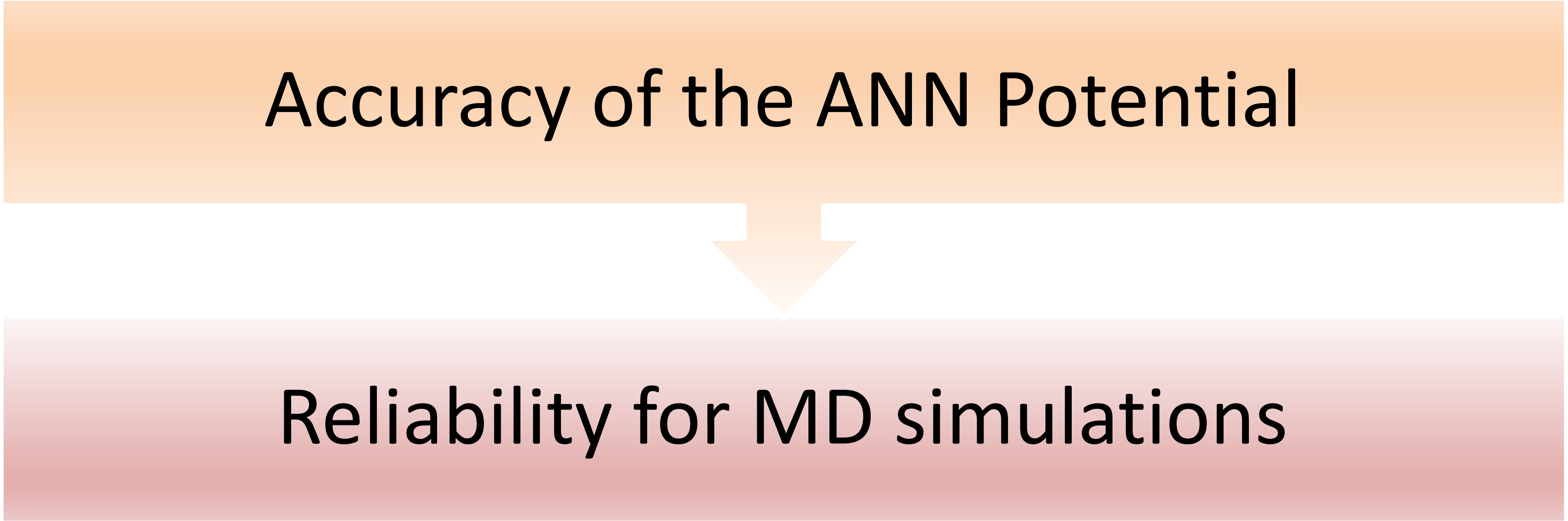


Select training/optimization method

- 3 available training methods: online GD, LBGFS, Levenberg-Marquardt (LM)
- Online GD – least computationally demanding; any ANN architectures; not well parallelized
- Batch LBGFS – well parallelized; scales linearly with reference structures
- Bacth LM – 2nd order convergence rate; needs to invert Hessian



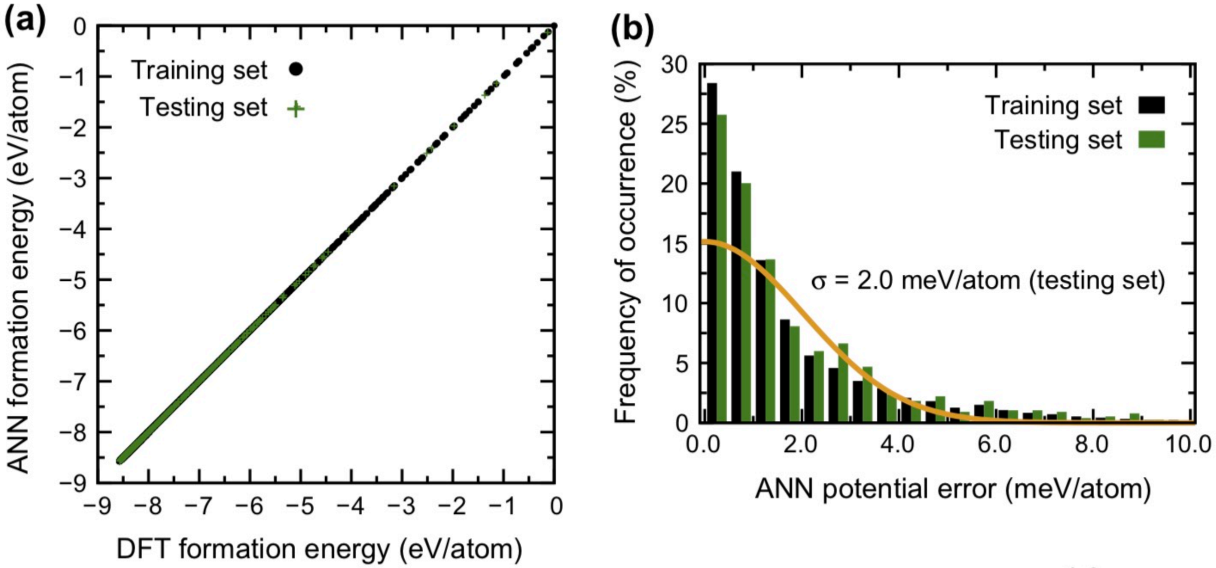
Evaluate the Trained ANN Force Field (TiO_2)



Accuracy of the ANN Potential

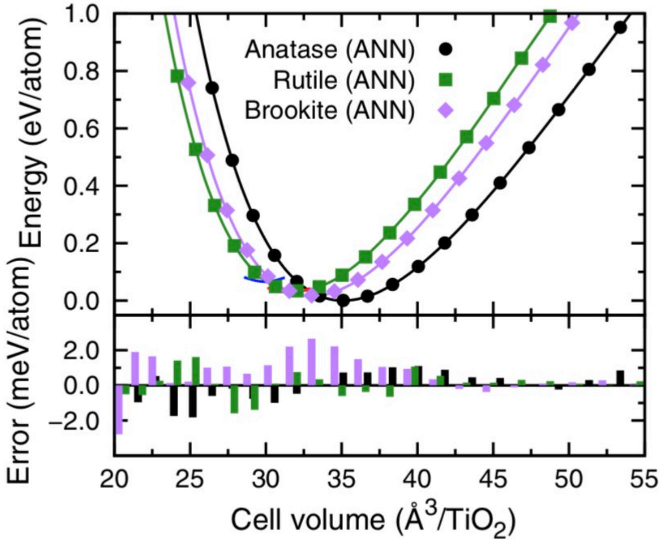
Reliability for MD simulations

Accuracy of the TiO_2 ANN Potential

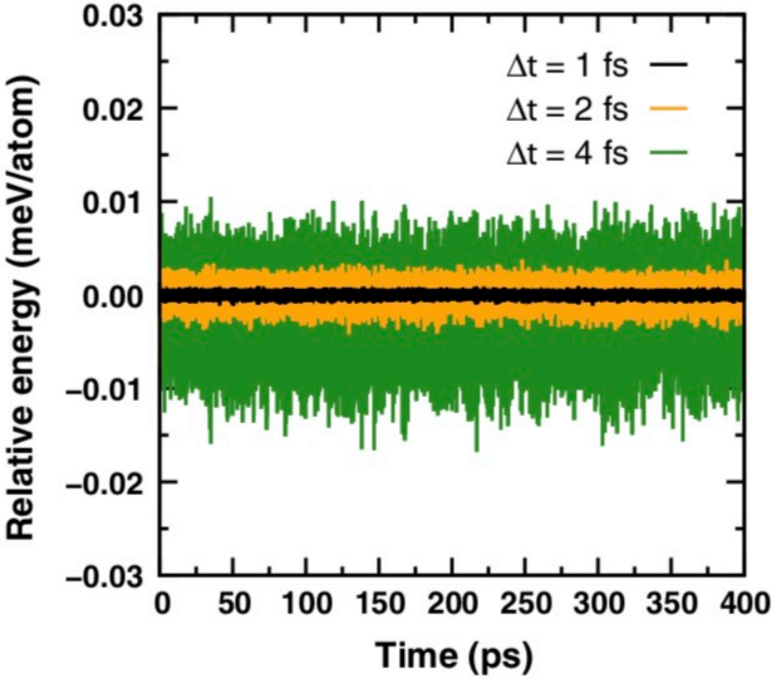


Error statistics

Smooth change of energy with volume

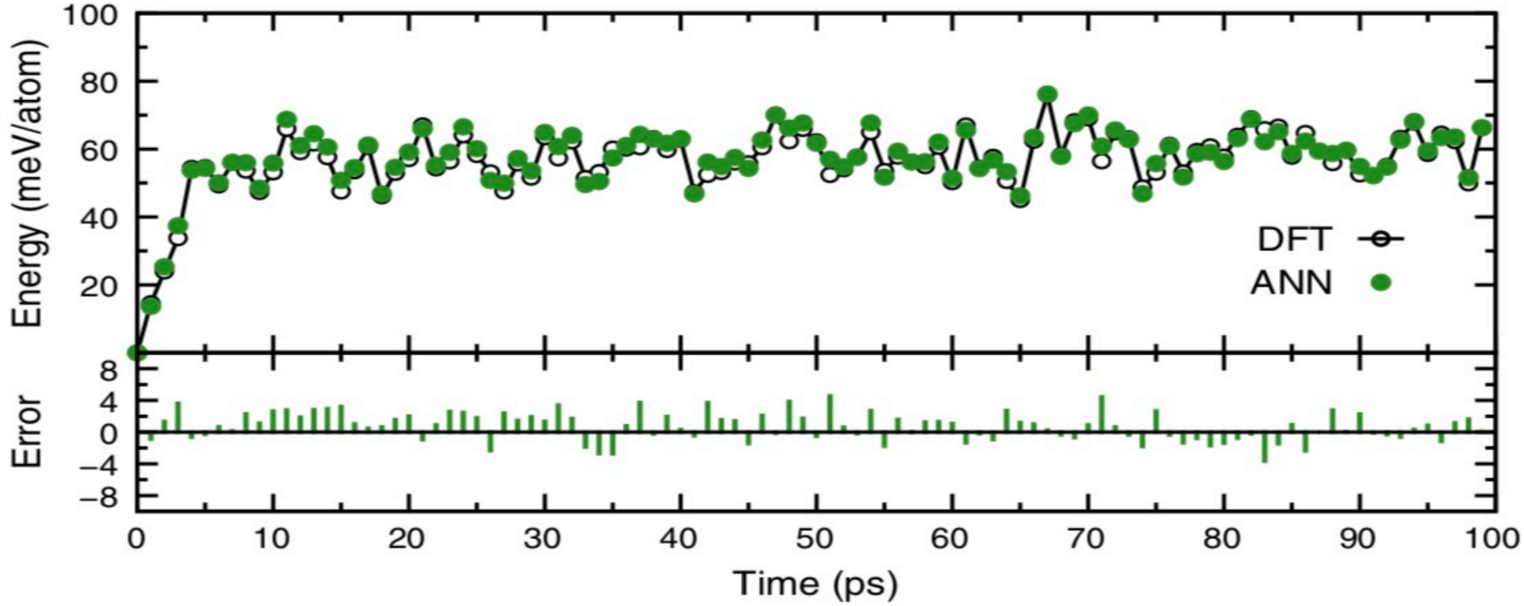


Reliability of the TiO_2 ANN Potential for MD Runs



NVE runs with
various timesteps

NVT runs at 500 K
for different structures



Machine learning based interatomic potential for amorphous carbon

Basic Ideas & Background

- Interatomic Potential: Mathematical functions for calculating the potential energy of a system of atoms with given positions in space
- Amorphous Carbon(a-C): free, reactive carbon that does not have any crystalline structure
- Tetrahedral Amorphous Carbon (ta-C): a new semiconductor which is able to accept dopants and shows photoconductivity
- GAP: Gaussian approximation potential

Basic Ideas & Background

- Fast potentials make large-scale molecular-dynamics (MD) simulations possible
- Efficient enough to perform thin-film deposition simulations
- Directly mirroring the atomic-scale processes in experiments

Shortcomings:

- prominent an underestimated concentration of sp^3-bonded atoms in *ta*-C and poor description of surfaces.
- density-functional theory(DFT)-based methods are restricted to quite small system sizes and they are limited in practice to a few hundred atoms.

Key Ideas of Machine Learning (ML)

- map a set of atomic environments directly onto numerical values for energies and forces
- these quantities are “trained” from a large and accurate quantum-mechanical reference database
- interpolated using the ML algorithm

Gaussian approximation potential (GAP):

- Determine the maximum accuracy that any finite-range potential can achieve in a carbon structures.
- Construct a GAP model that can indeed reach the target accuracy, by using a hierarchical set of two-, three-, and many-body structural descriptors.
- Show predictions for energies and structures of ta-C surfaces, which play a key role in wear and fracture mechanisms

Gaussian approximation potential (GAP)

Energy Function ε: energy function is expanded in a basis set adapted to the input database. Generated using a kernel function, or similarity measure of neighbor environments.

Extrapolation: a poor fit in regions of configuration space far away from any data points.

Starting point for the total energy

$$E = (\delta^{(2b)})^2 \sum_{i \in \text{pairs}} \varepsilon^{(2b)}\left(\mathbf{q}_i^{(2b)}\right) + (\delta^{(3b)})^2 \sum_{j \in \text{triplets}} \varepsilon^{(3b)}\left(\mathbf{q}_j^{(3b)}\right) + (\delta^{(MB)})^2 \sum_{a \in \text{atoms}} \varepsilon^{(MB)}\left(\mathbf{q}_a^{(MB)}\right), \tag{1}$$

Local energy corresponding to each descriptor d $\in$ {2b,3b,MB}

$$\varepsilon^{(d)}(\mathbf{q}^{(d)}) = \sum_{t=1}^{N_t^{(d)}} \alpha_t^{(d)} K^{(d)}(\mathbf{q}^{(d)}, \mathbf{q}_t^{(d)}),$$

$$K^{(d)}(\mathbf{q}_i^{(d)}, \mathbf{q}_t^{(d)}) = \exp\left[-\frac{1}{2}\sum_{\xi} \frac{(q_{\xi,i}^{(d)} - q_{\xi,t}^{(d)})^2}{\theta_{\xi}^2}\right],$$

ξ is an index running over the components of the descriptor vector q(d)

$$q^{(2b)} = |\mathbf{r}_2 - \mathbf{r}_1| \equiv r_{12} \;;$$

the case of pairs

$$\mathbf{q}^{(3b)} = \begin{pmatrix} r_{12} + r_{13} \\ (r_{12} - r_{13})^2 \\ r_{23} \end{pmatrix}.$$

the case of triplets

Smooth overlap of atomic positions (SOAP) descriptor

$$\rho_a(\mathbf{r}) = \sum_b \exp\left[-\frac{(\mathbf{r}-\mathbf{r}_{ab})^2}{2\sigma_{\text{at}}^2}\right] \times f_{\text{cut}}(r_{ab}),$$

the sum is over neighboring atoms

$$\rho_a(\mathbf{r}) = \sum_{nlm} c_{nlm}^{(a)}\, g_n(r) Y_{lm}(\hat{\mathbf{r}}),$$

g_n: orthogonal radial basis functions

Y_{lm} : spherical harmonics

$$p_{nn'l}^{(a)} = \sqrt{\frac{8\pi^2}{2l+1}} \sum_m \left(c_{nlm}^{(a)}\right)^* c_{n'lm}^{(a)},$$

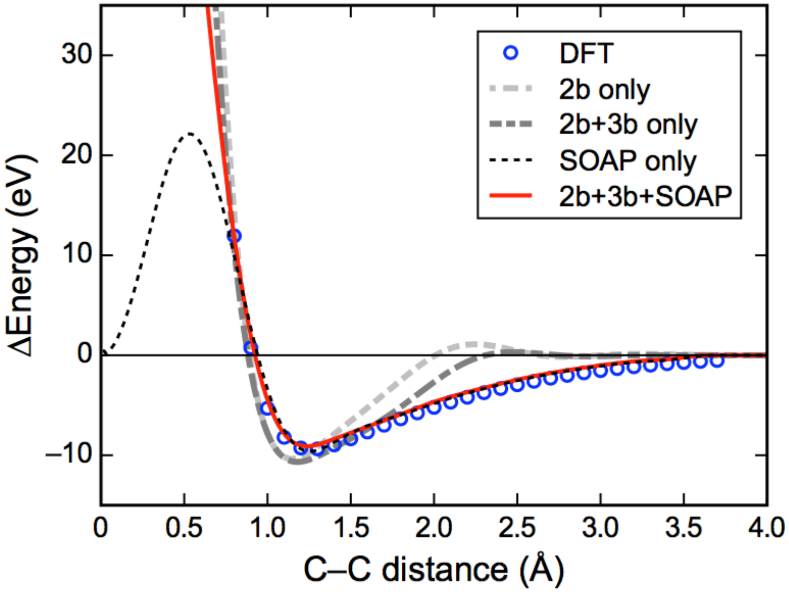
spherical power spectrum

The kernel function for the SOAP term

$$k\left(\mathbf{q}_a^{(\mathrm{MB})},\mathbf{q}_t^{(\mathrm{MB})}\right) = \sum_{nn'l} p_{nn'l}^{(a)}\, p_{nn'l}^{(t)} = \mathbf{q}_a^{(\mathrm{MB})}\cdot\mathbf{q}_t^{(\mathrm{MB})}$$

raise it to a small integer power for a sharper distinction between different environments

$$K^{(\mathrm{MB})}\left(\mathbf{q}_a^{(\mathrm{MB})},\mathbf{q}_t^{(\mathrm{MB})}\right) = \left|\mathbf{q}_a^{(\mathrm{MB})}\cdot\mathbf{q}_t^{(\mathrm{MB})}\right|^{\zeta}.$$



$$\begin{aligned}E = {} & (\delta^{(2b)})^2 \sum_i \sum_t \alpha_t^{(2b)} K^{(2b)}\left(\mathbf{q}_i^{(2b)},\mathbf{q}_t^{(2b)}\right)\\ & + (\delta^{(3b)})^2 \sum_j \sum_t \alpha_t^{(3b)} K^{(3b)}\left(\mathbf{q}_j^{(3b)},\mathbf{q}_t^{(3b)}\right)\\ & + (\delta^{(\mathrm{MB})})^2 \sum_a \sum_t \alpha_t^{(\mathrm{MB})} K^{(\mathrm{MB})}\left(\mathbf{q}_a^{(\mathrm{MB})},\mathbf{q}_t^{(\mathrm{MB})}\right),\end{aligned}$$

The expression for the total energy in GAP

All fitting coefficients α enter linearly

COMPUTATIONAL METHODS

- General protocol for melt-quench simulations : Structural data were obtained from melt-quench MD, following protocols that are well established for a-C
- DFT-based (*“ab initio”*) molecular dynamics :Structures for initial training, as well as benchmarks for a-C properties, were generated using DFT-based ab initio MD, using the QUICKSTEP scheme and a stochastic Langevin thermostat as implemented in CP2K
- Construction of the training database :contains structural snapshots from ab initio MD and also, as it is iteratively extended, from GAP- driven simulations.

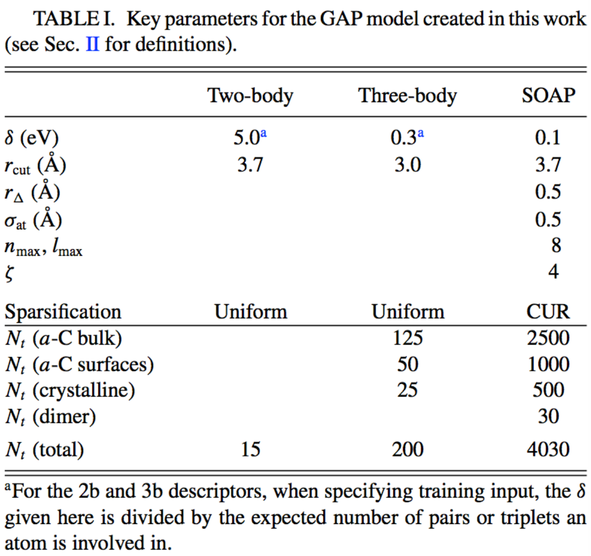
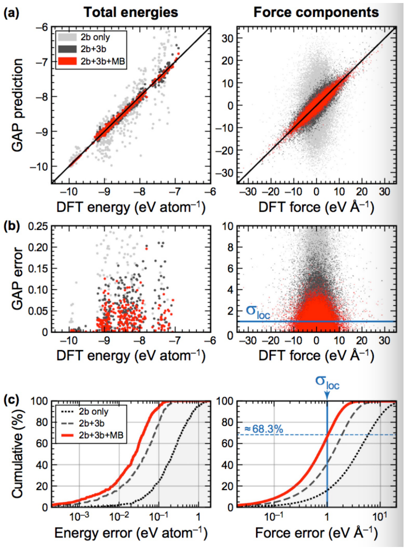


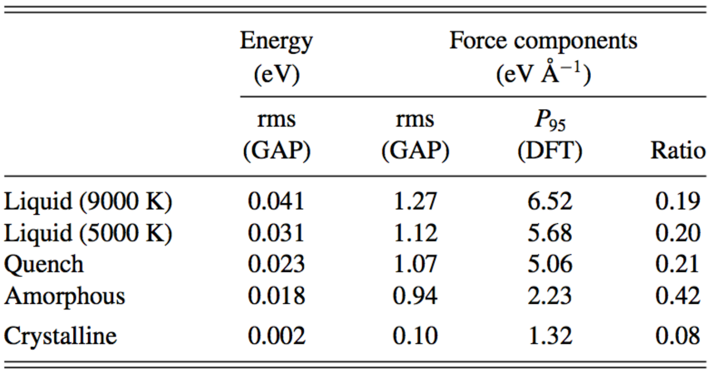
TABLE I. Key parameters for the GAP model created in this work (see Sec. II for definitions).

	Two-body	Three-body	SOAP
δ (eV)	5.0[a]	0.3[a]	0.1
r_{cut} (Å)	3.7	3.0	3.7
r_{Δ} (Å)			0.5
σ_{at} (Å)			0.5
$n_{\max}$, $l_{\max}$			8
ζ			4
Sparsification	Uniform	Uniform	CUR
N_t (*a*-C bulk)		125	2500
N_t (*a*-C surfaces)		50	1000
N_t (crystalline)		25	500
N_t (dimer)			30
N_t (total)	15	200	4030

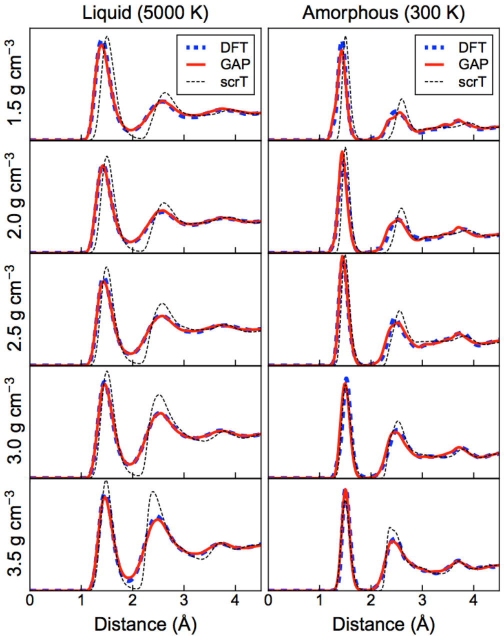
[a] For the 2b and 3b descriptors, when specifying training input, the δ given here is divided by the expected number of pairs or triplets an atom is involved in.

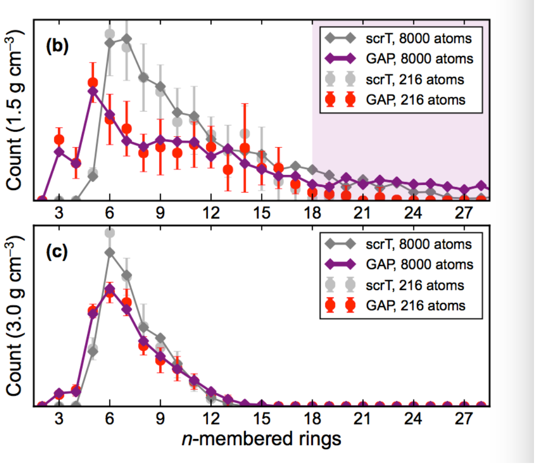
Results and Discussion





	Energy (eV)	Force components (eV Å^{-1})		
	rms (GAP)	rms (GAP)	P_{95} (DFT)	Ratio
Liquid (9000 K)	0.041	1.27	6.52	0.19
Liquid (5000 K)	0.031	1.12	5.68	0.20
Quench	0.023	1.07	5.06	0.21
Amorphous	0.018	0.94	2.23	0.42
Crystalline	0.002	0.10	1.32	0.08


Liquid (5000 K)
Amorphous (300 K)
DFT
GAP
scrT
1.5 g cm^{-3}
2.0 g cm^{-3}
2.5 g cm^{-3}
3.0 g cm^{-3}
3.5 g cm^{-3}
Distance (Å)


(b)
Count (1.5 g cm^{-3})
scrT, 8000 atoms
GAP, 8000 atoms
scrT, 216 atoms
GAP, 216 atoms
(c)
Count (3.0 g cm^{-3})
n-membered rings