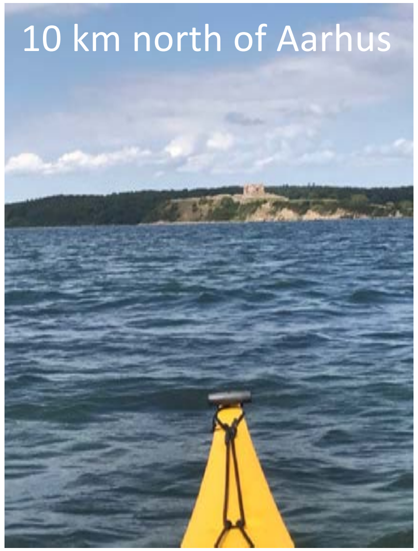
Machine learning – acceleration of global optimization

Bjørk Hammer

IKZ & FAIRMAT winter school – Jan, 2023



Hiring Postdocs ! Check out mldft.com

AARHUS UNIVERSITY

VILLUM FONDEN

INTERCAT
Center for Interstellar Catalysis

INSTITUT FOR FYSIK OG ASTRONOMI
IFA

Outline

Motivation

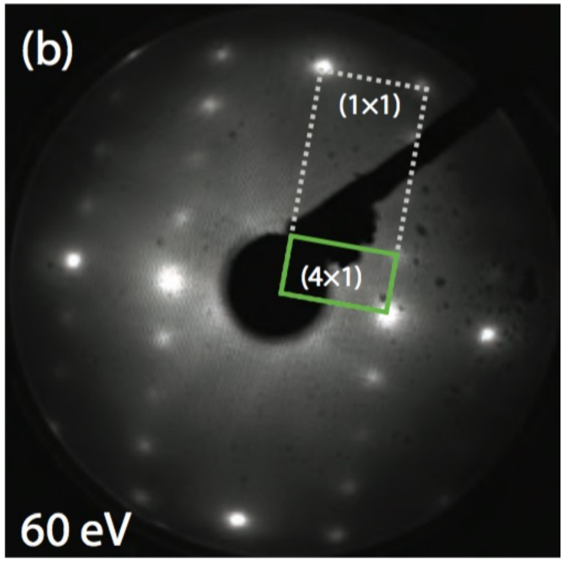
Machine learning learning of interatomic potentials

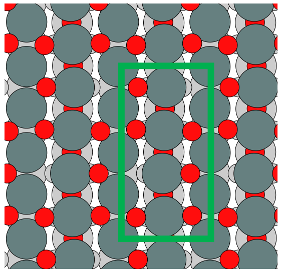
Learning chemistry? Artificial intelligence…

Why global optimization is needed

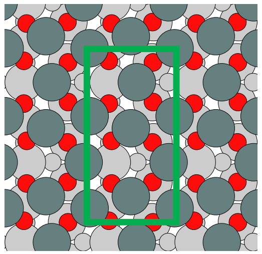
Determine structure

Reduced tin-oxide, SnO_2, surface

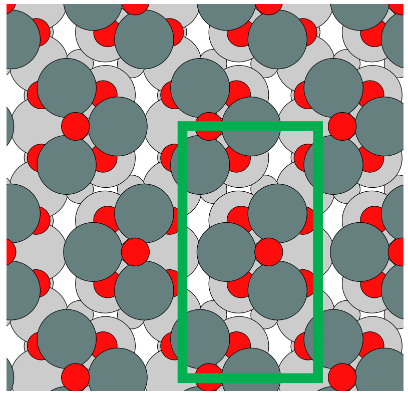




2001

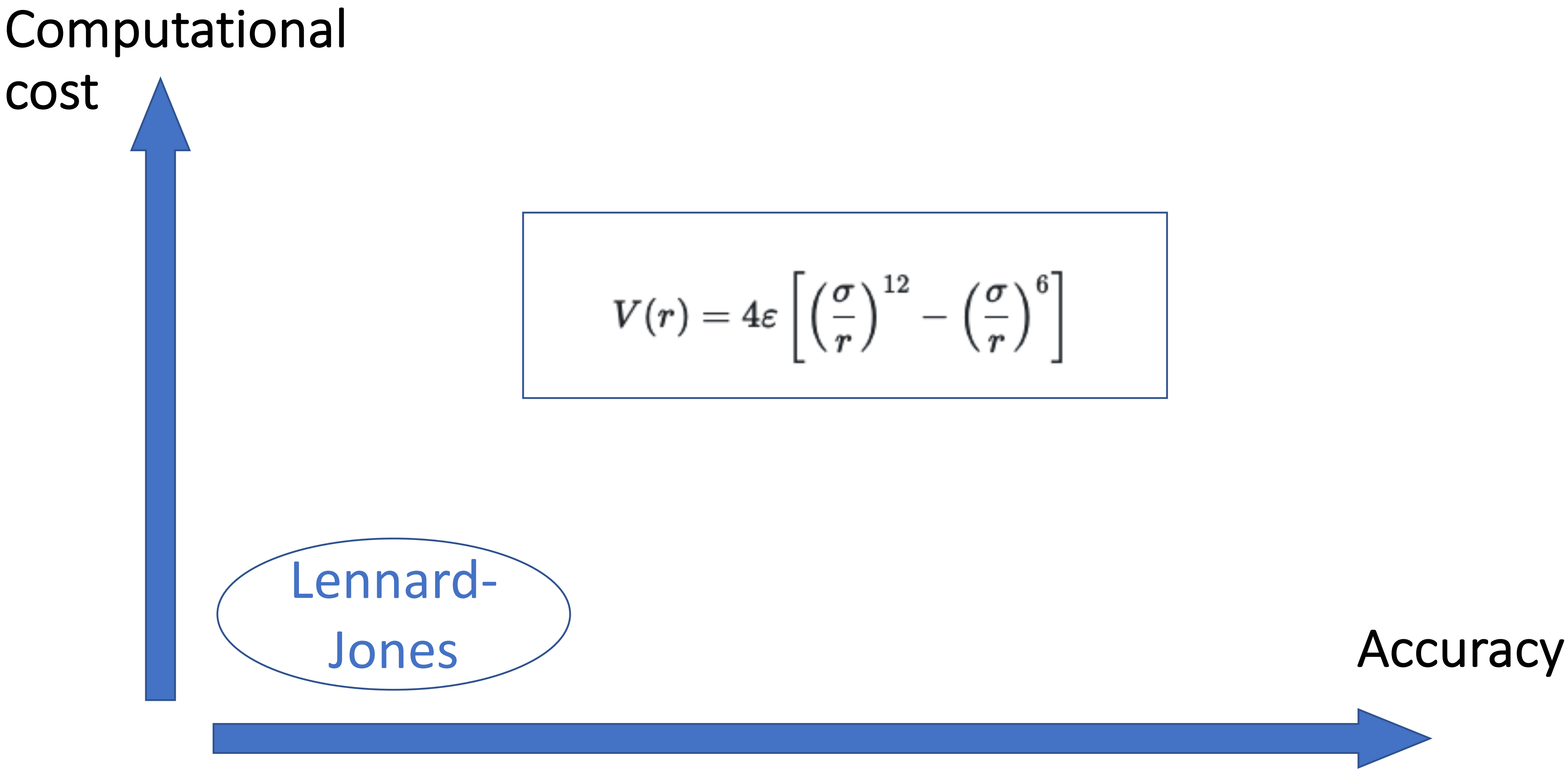


2003



2017

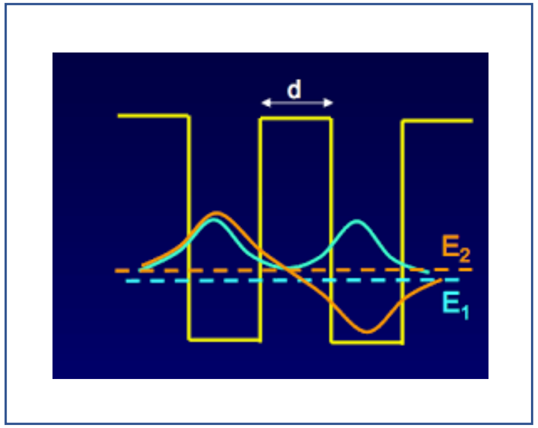
Total energy methods



$$V(r) = 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]$$

Total energy methods

Computational cost



Tight-binding

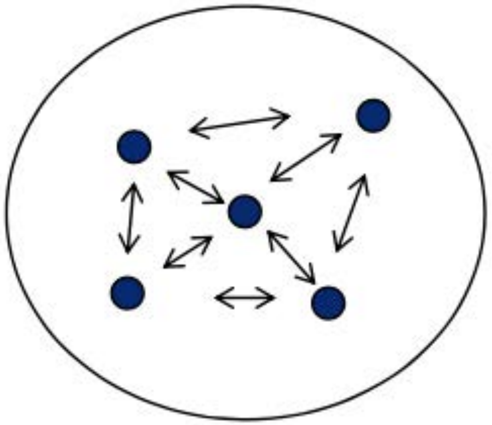
Lennard-Jones

Accuracy

Total energy methods

$$\hat{H}_{\text{tot}} = \sum_i \frac{\mathbf{p}_i^2}{2m} + \sum_i V_{\text{nucl}}(\mathbf{r}_i) + \frac{1}{2}\sum_{i \neq j} \frac{e^2}{|\mathbf{r}_i - \mathbf{r}_j|}$$

Many-body Hamiltonian



$$\Psi(\mathbf{r_1}, \mathbf{r_2}, \mathbf{r_3} \ldots \mathbf{r_N})$$

Many-body wave function!!!!

$$\Psi(\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_N) = \frac{1}{\sqrt{N!}} \begin{vmatrix} \phi_1(\mathbf{r}_1)\phi_2(\mathbf{r}_1) & \cdots & \phi_N(\mathbf{r}_1) \\ \phi_1(\mathbf{r}_2)\phi_2(\mathbf{r}_2) & \cdots & \phi_N(\mathbf{r}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(\mathbf{r}_N)\phi_2(\mathbf{r}_N) & \cdots & \phi_N(\mathbf{r}_N) \end{vmatrix}$$

Quantum Chemistry

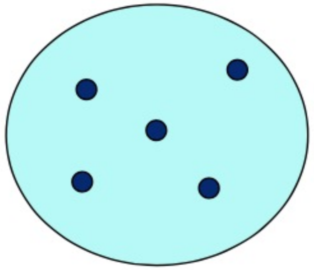
Accuracy

Density functional theory

Computational cost

$$\left(-\frac{\hbar^2}{2m}\nabla^2 + v_{\text{eff}}(\mathbf{r})\right)\varphi_i(\mathbf{r}) = \varepsilon_i\varphi_i(\mathbf{r}).$$

Single-particle Hamiltonian



$$\rho(\mathbf{r}) = \sum_i^N |\varphi_i(\mathbf{r})|^2.$$

$$\phi(\mathbf{r}_1)\phi(\mathbf{r}_2)\phi(\mathbf{r}_3)\ldots\phi(\mathbf{r}_N)$$

$$v_{\text{eff}}(\mathbf{r}) = v_{\text{ext}}(\mathbf{r}) + e^2\int \frac{\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}\,d\mathbf{r}' + \frac{\delta E_{\text{xc}}[\rho]}{\delta\rho(\mathbf{r})},$$

$$E[\rho] = \underbrace{T_s[\rho]}_{\text{kinetic}} + \underbrace{\int d\mathbf{r}\, v_{\text{ext}}(\mathbf{r})\rho(\mathbf{r}) + E_{\text{H}}[\rho]}_{\text{Coulomb}} + \underbrace{E_{\text{xc}}[\rho]}_{\text{many-body}},$$

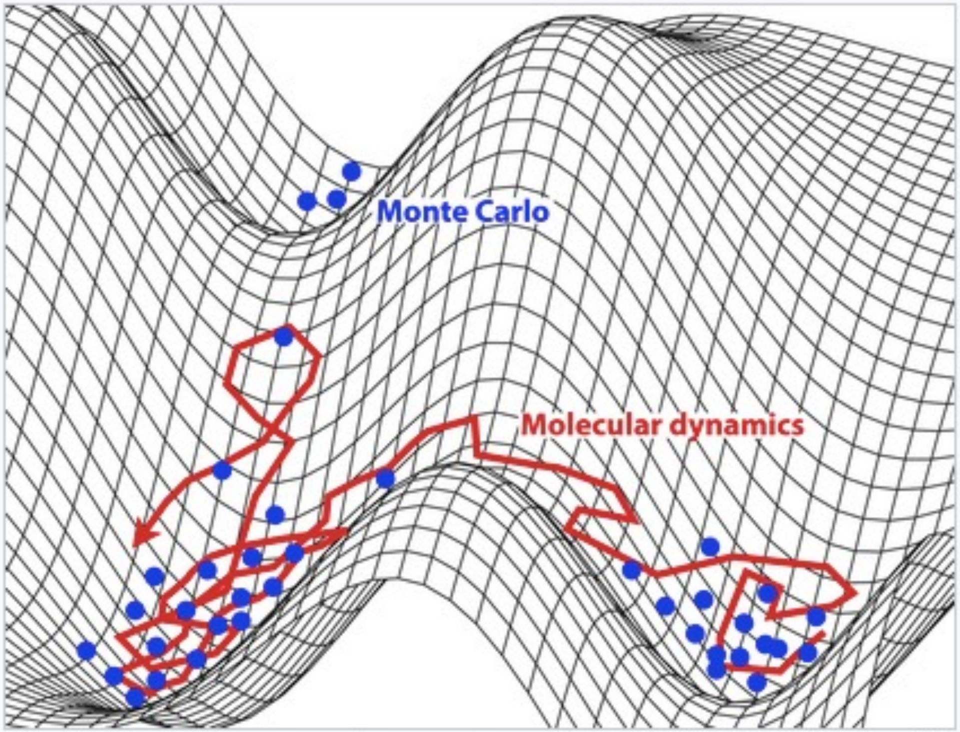
$$T_s[\rho] = \sum_{i=1}^N \int d\mathbf{r}\, \varphi_i^*(\mathbf{r})\left(-\frac{\hbar^2}{2m}\nabla^2\right)\varphi_i(\mathbf{r}), \quad E_{\text{H}}[\rho] = \frac{e^2}{2}\int d\mathbf{r}\int d\mathbf{r}'\,\frac{\rho(\mathbf{r})\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}$$

Quantum Chemistry

DFT

Accuracy

Simulation methods



Schematic representation of the sampling of the system's potential energy surface with molecular dynamics (in red) compared to Monte Carlo methods (in blue).

Figure credit: Wikipedia

Molecular dynamics:

$$F(X) = -\nabla U(X) = M\dot{V}(t)$$
$$V(t) = \dot{X}(t).$$

Metropolis Monte-Carlo:

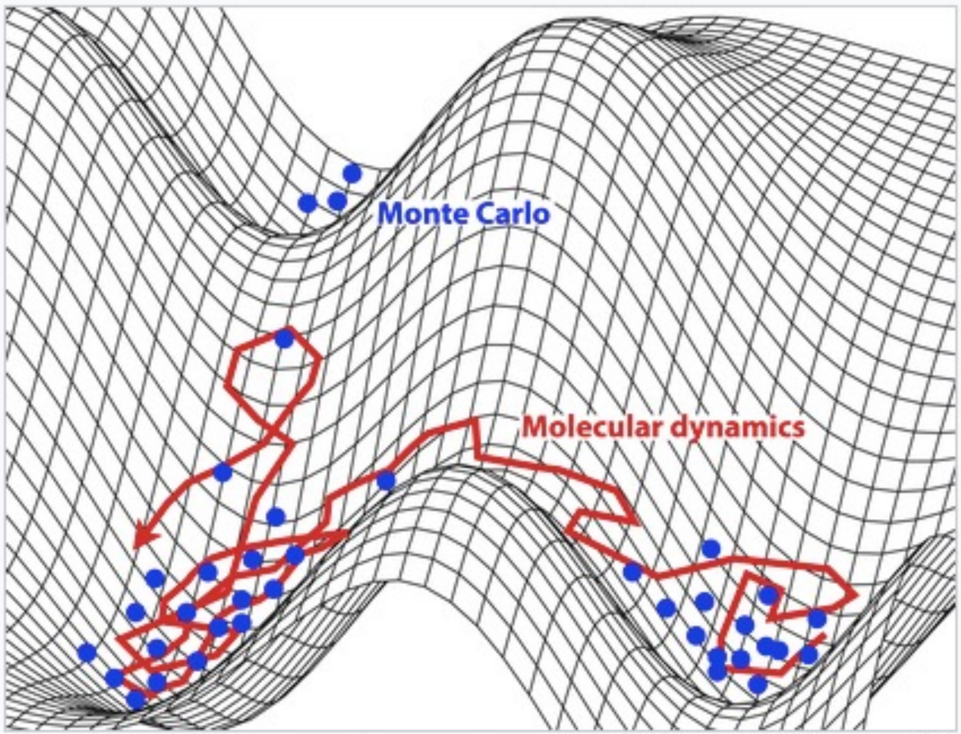
- Perturb the structure
- Accept if potential energy decreases,

$$\Delta U < 0$$

or

$$n < \exp(-\Delta U/\Delta U_0)$$

Simulation methods



Schematic representation of the sampling of the system's potential energy surface with molecular dynamics (in red) compared to Monte Carlo methods (in blue).

Figure credit: Wikipedia

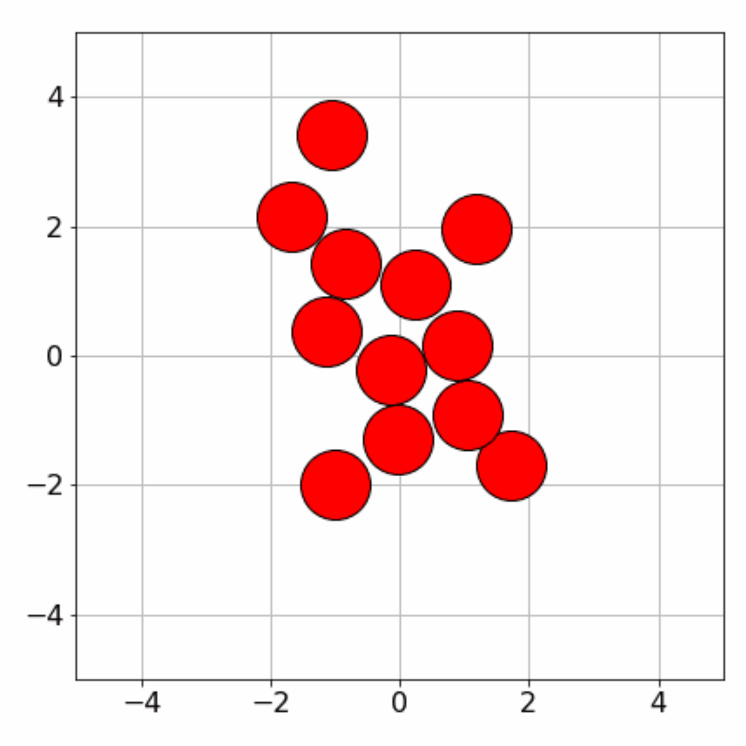


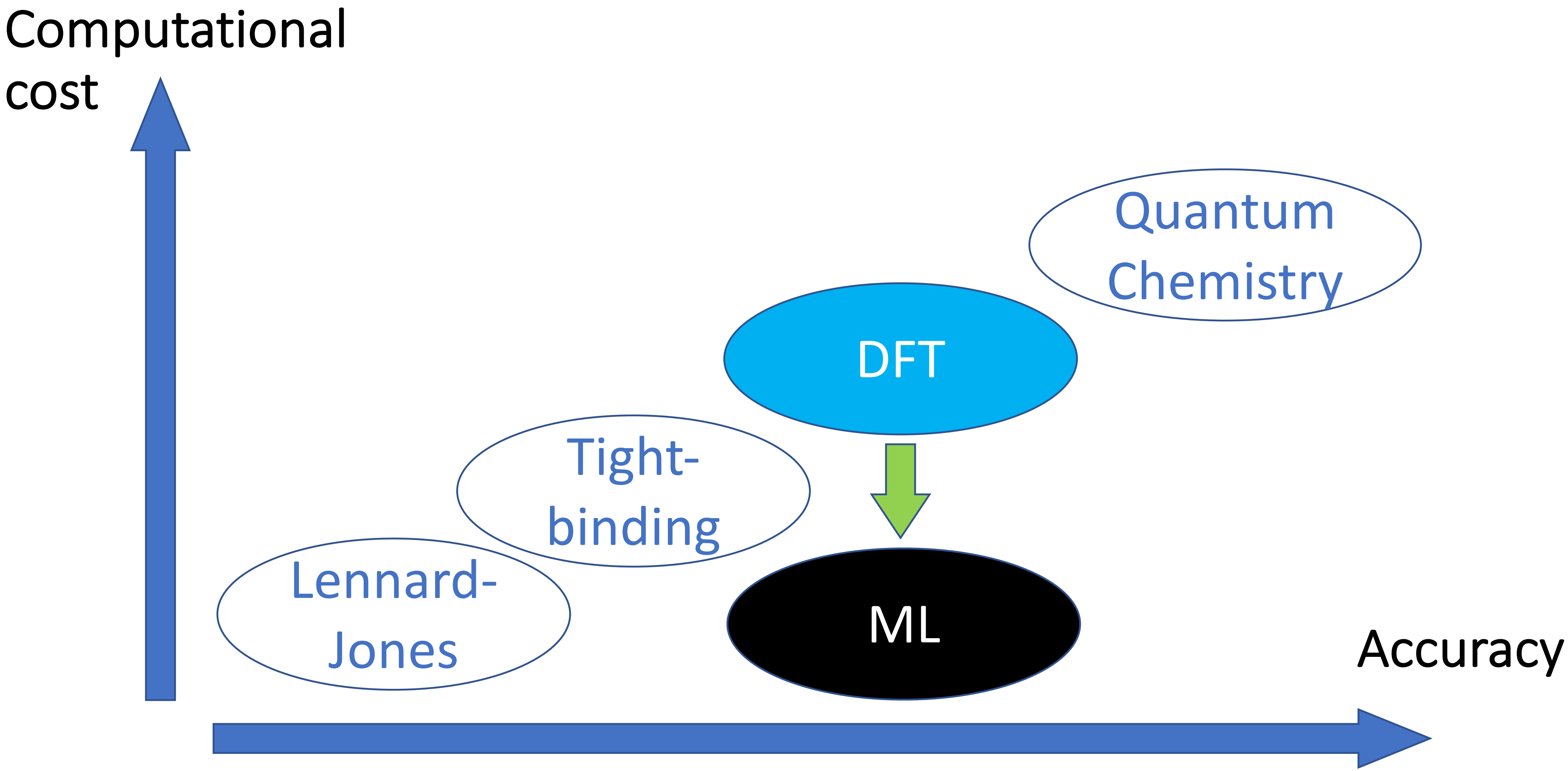
Figure credit: Python course at Aarhus University

Computational times for DFT simulations on a supercomputer

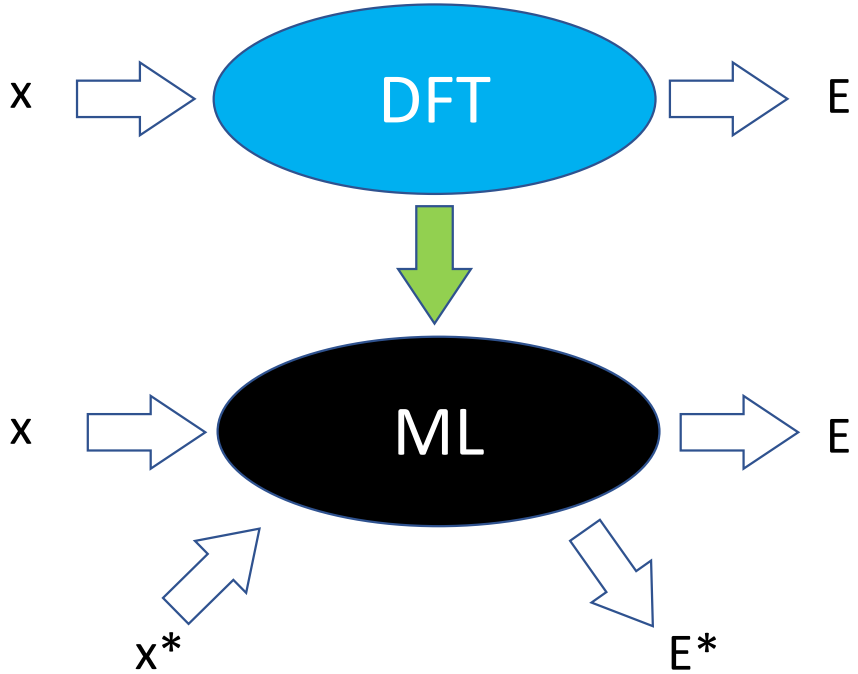
100 atoms on 8 CPU cores

Energy evaluation	1 hrs
Relaxation trajectory	100 x 1 hrs = 4 days
Molecular dynamics (10 ns, 0.1 fs step)	10,000 x 1 hrs = 1 year
Monte-Carlo search (100 perturbations)	100 x 100 x 1 hrs = 1 year

Machine learning based total energy



Machine learning based total energy

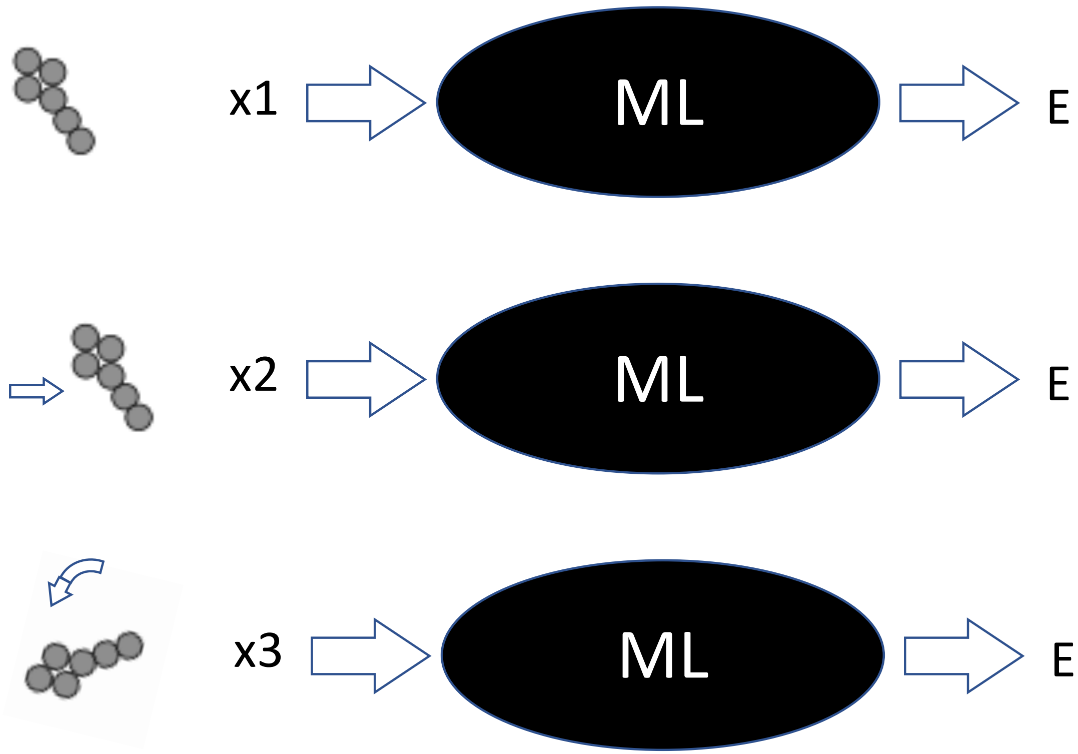


Collect structure-energy, (x,E), data at DFT level

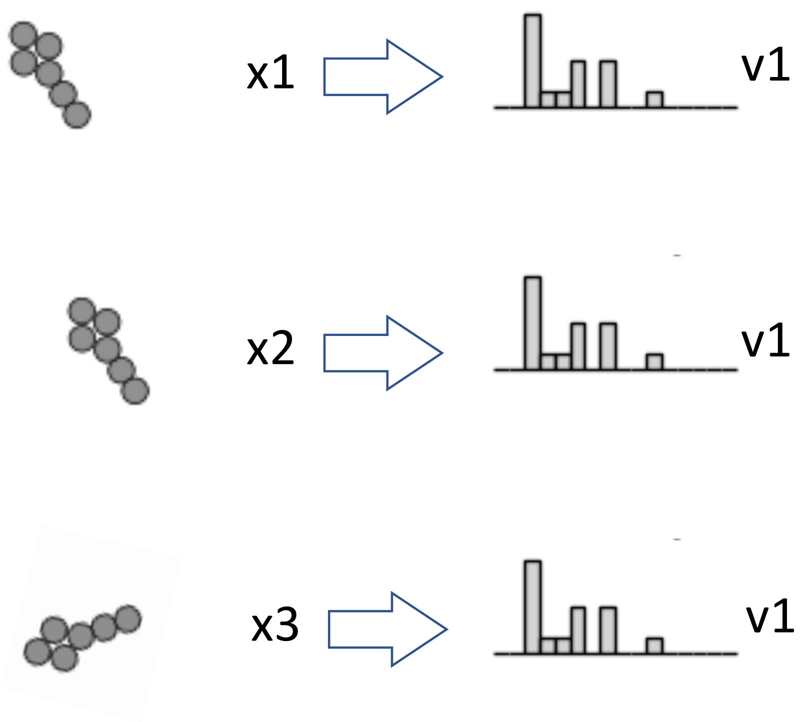
Train “black box” to reproduce E given x

Simulate with the “black box”

Expected invariant behavior of “black box”



Representation



Carthesian coordinates

”descriptor”,
”feature”, ”f
or *”fingerpri*

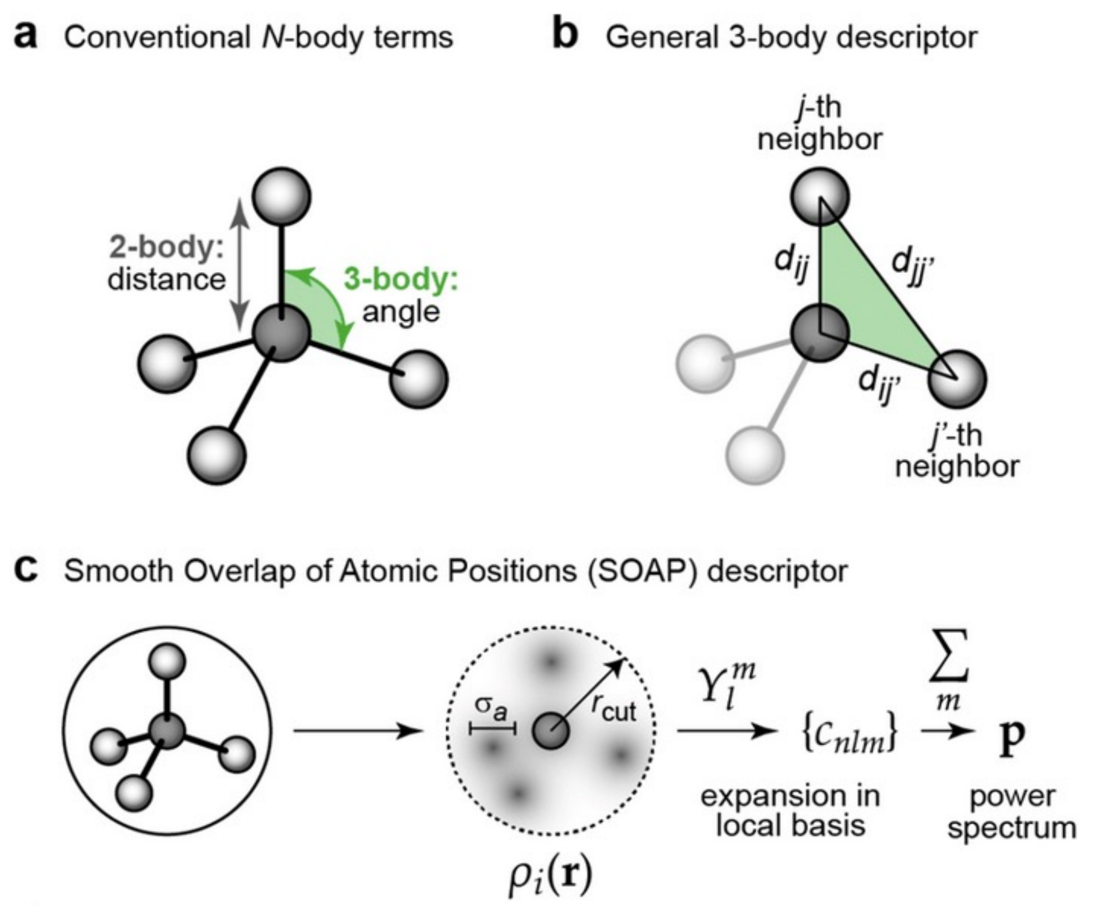
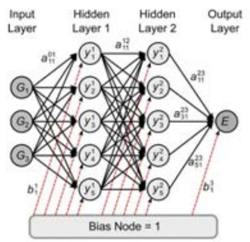


Figure credit: Volker Deringer

Two popular options for the "black box"

2007:
Artificial Neural Networks



PHYSICAL REVIEW LETTERS

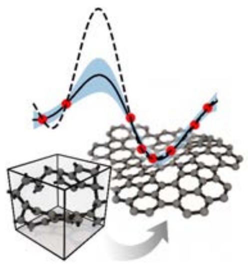
Highlights Recent Accepted Collections Authors Referees Search Press About

Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces

Jörg Behler and Michele Parrinello
Phys. Rev. Lett. **98**, 146401 – Published 2 April 2007

Citing Articles (1,783)

2010:
Gaussian Process Regression



PHYSICAL REVIEW LETTERS

Highlights Recent Accepted Collections Authors Referees Search Press About

Featured in Physics

Gaussian Approximation Potentials: The Accuracy of Quantum Mechanics, without the Electrons

Albert P. Bartók, Mike C. Payne, Risi Kondor, and Gábor Csányi
Phys. Rev. Lett. **104**, 136403 – Published 1 April 2010

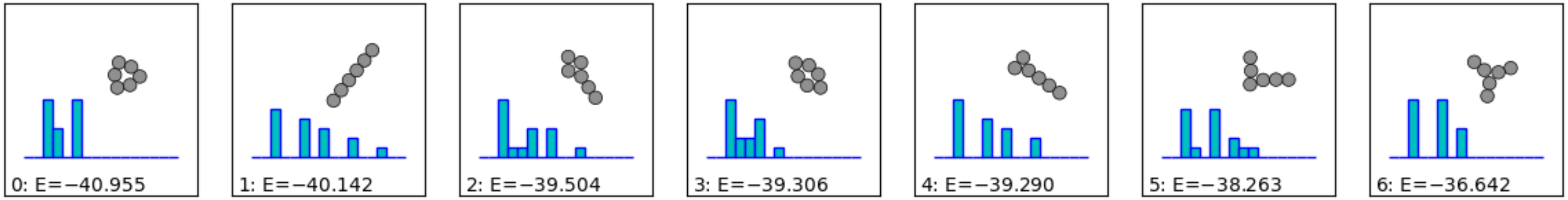
Citing Articles (1,227)

Gaussian Process Regression: Efficient with very little data

$$E_{sur}(\mathbf{x}_*) = \mathbf{k}_*^T \left(K + \sigma_n^2 I\right)^{-1} \mathbf{E}$$

$$k(\mathbf{x}_i, \mathbf{x}_j) = \theta_0 \mathrm{e}^{-||\mathbf{x}_i - \mathbf{x}_j||^2/(2\lambda)}$$

Gaussian Process Regression: Efficient with very little data



$\mathbf{x}_i$ describes the *i*'th structure

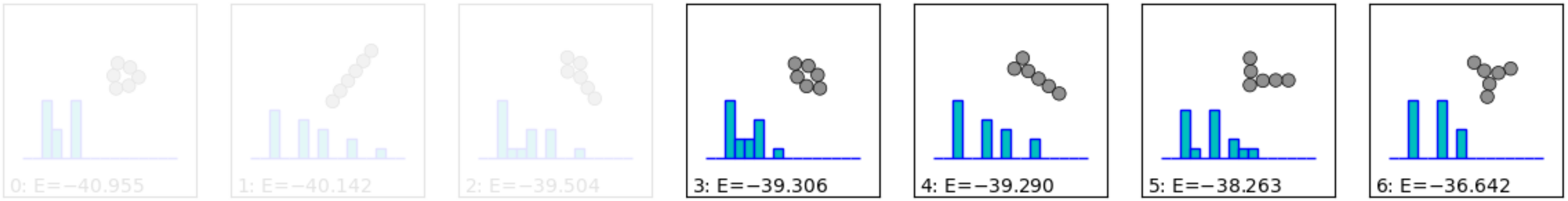
… is called: *"descriptor", "feature", "feature vector"*, or *"fingerprint"*

Kernel:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \theta_0 \mathrm{e}^{-||\mathbf{x}_i - \mathbf{x}_j||^2/(2\lambda)}$$

… measures how similar two structures are

Let’s say we already found structure 3, 4, 5, and 6: Construct K



$k(\mathbf{x}_3, \mathbf{x}_3) = 100$

$k(\mathbf{x}_3, \mathbf{x}_4) = 85$

$k(\mathbf{x}_5, \mathbf{x}_6) = 94$

$k(\mathbf{x}_3, \mathbf{x}_5) = 90$

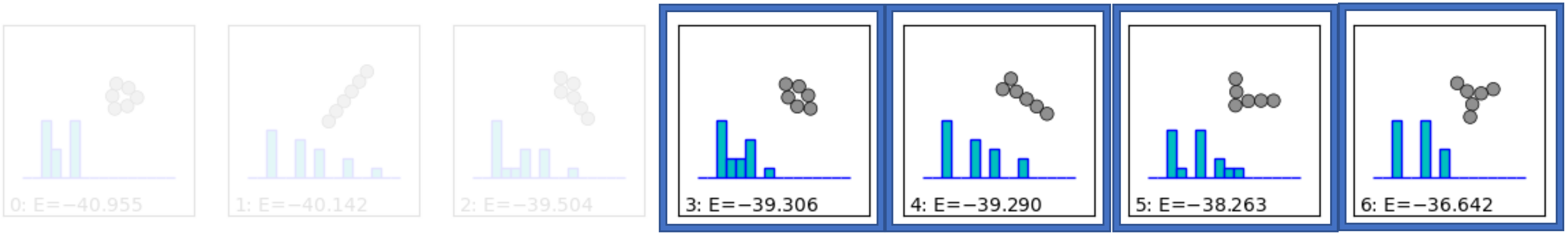
$k(\mathbf{x}_3, \mathbf{x}_6) = 85$

$$k(\mathbf{x}_i, \mathbf{x}_j) = \theta_0 \mathrm{e}^{-||\mathbf{x}_i - \mathbf{x}_j||^2/(2\lambda)}$$

$$\theta_0 = 100 \qquad \lambda = 50$$

$$K = \begin{bmatrix} 100 & 85 & 90 & 85 \\ 85 & 100 & 90 & 92 \\ 90 & 90 & 100 & 94 \\ 85 & 92 & 94 & 100 \end{bmatrix}$$

Solve for model parameters and check on training data:



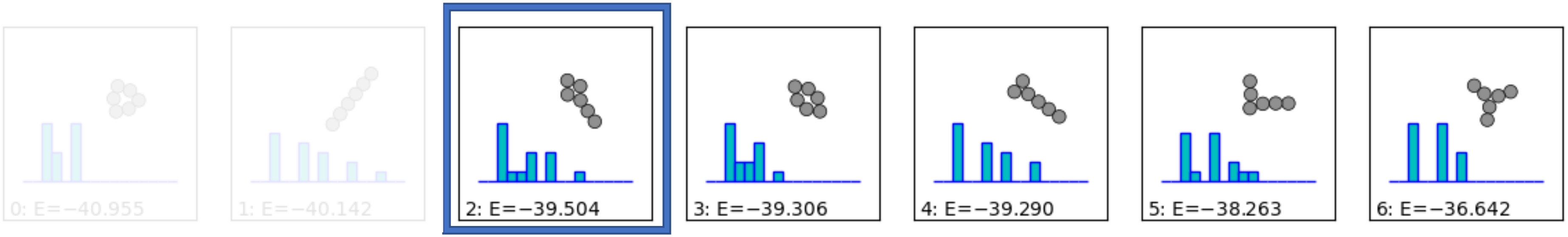
$$K\boldsymbol{\alpha} = \mathbf{E}$$

$$\begin{bmatrix} k(\mathbf{x}_3,\mathbf{x}_3) & k(\mathbf{x}_3,\mathbf{x}_4) & k(\mathbf{x}_3,\mathbf{x}_5) & k(\mathbf{x}_3,\mathbf{x}_6) \\ k(\mathbf{x}_4,\mathbf{x}_3) & k(\mathbf{x}_4,\mathbf{x}_4) & k(\mathbf{x}_4,\mathbf{x}_5) & k(\mathbf{x}_4,\mathbf{x}_6) \\ k(\mathbf{x}_5,\mathbf{x}_3) & k(\mathbf{x}_5,\mathbf{x}_4) & k(\mathbf{x}_5,\mathbf{x}_5) & k(\mathbf{x}_5,\mathbf{x}_6) \\ k(\mathbf{x}_6,\mathbf{x}_3) & k(\mathbf{x}_6,\mathbf{x}_4) & k(\mathbf{x}_6,\mathbf{x}_5) & k(\mathbf{x}_6,\mathbf{x}_6) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{bmatrix} = \begin{bmatrix} E_3 \\ E_4 \\ E_5 \\ E_6 \end{bmatrix}$$

$$K^{-1}K\boldsymbol{\alpha} = K^{-1}\mathbf{E}$$

$$\boldsymbol{\alpha} \approx \left(K + \sigma_n^2 I\right)^{-1}\mathbf{E} = \begin{bmatrix} -0.210 \\ -0.262 \\ -0.060 \\ 0.110 \end{bmatrix} \quad K\boldsymbol{\alpha} = \begin{bmatrix} -39.300 \\ -39.282 \\ -38.261 \\ -36.645 \end{bmatrix} \quad \mathbf{E} = \begin{bmatrix} -39.306 \\ -39.290 \\ -38.263 \\ -36.642 \end{bmatrix} \quad K = \begin{bmatrix} 100 & 85 & 90 & 85 \\ 85 & 100 & 90 & 92 \\ 90 & 90 & 100 & 94 \\ 85 & 92 & 94 & 100 \end{bmatrix}$$

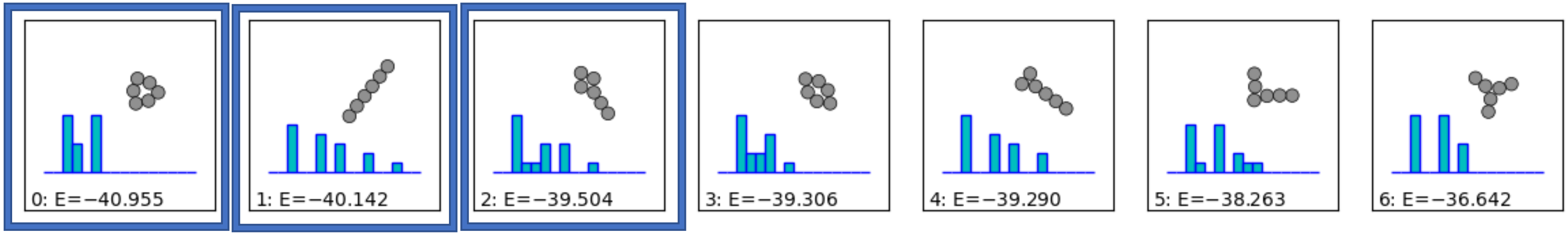
Then, for structure 2 we can construct the kernel vector:



$$\mathbf{k}_2 = \mathbf{k}(\mathbf{x}_2) = \begin{bmatrix} k(\mathbf{x}_3, \mathbf{x}_2) \\ k(\mathbf{x}_4, \mathbf{x}_2) \\ k(\mathbf{x}_5, \mathbf{x}_2) \\ k(\mathbf{x}_6, \mathbf{x}_2) \end{bmatrix} = \begin{bmatrix} 92 \\ 96 \\ 90 \\ 89 \end{bmatrix} \qquad E_{sur}(\mathbf{x}_2) = \mathbf{k}_2^T \boldsymbol{\alpha} = -40.1$$

$$E_{sur}(\mathbf{x}_*) = \mathbf{k}_*^T \left(K + \sigma_n^2 I\right)^{-1} \mathbf{E}$$

Uncertainty estimation



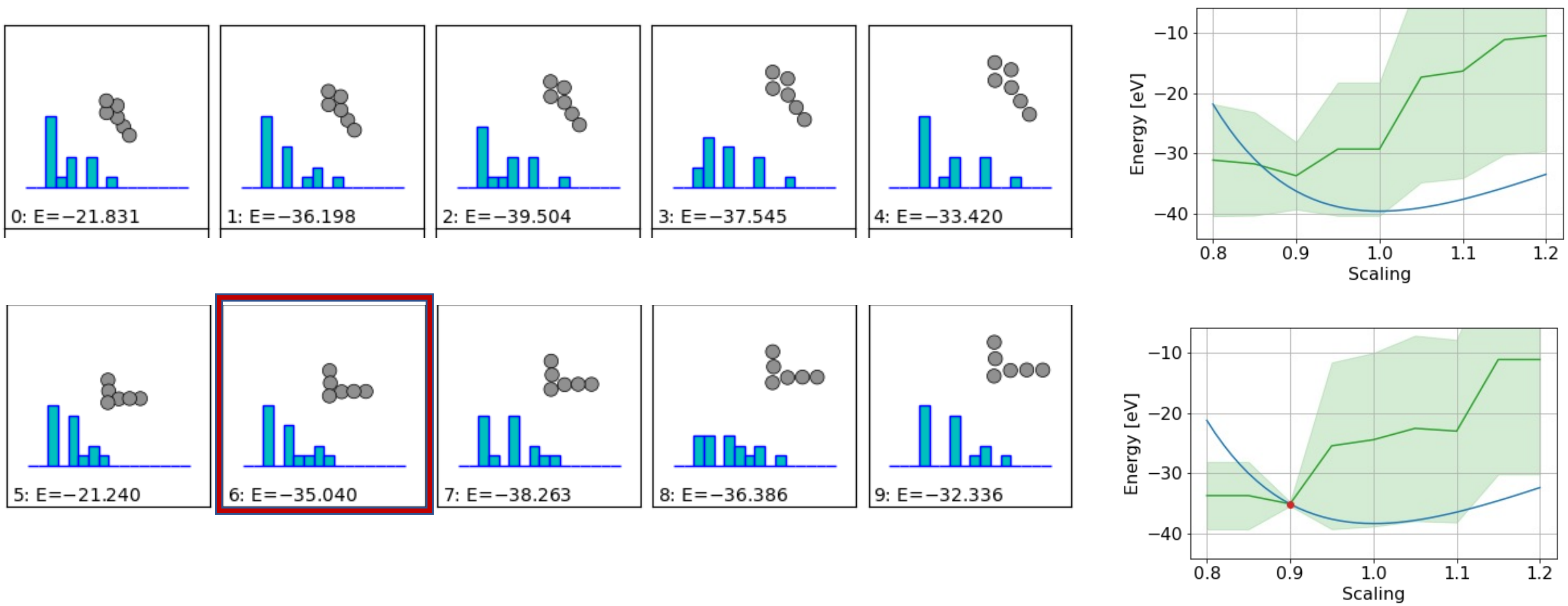
-35.3 ± 3.7 -38.6 ± 1.9 -40.1 ± 1.9

$$E_{sur}(\mathbf{x}_*) = \mathbf{k}_*^T \left(K + \sigma_n^2 I\right)^{-1} \mathbf{E}$$

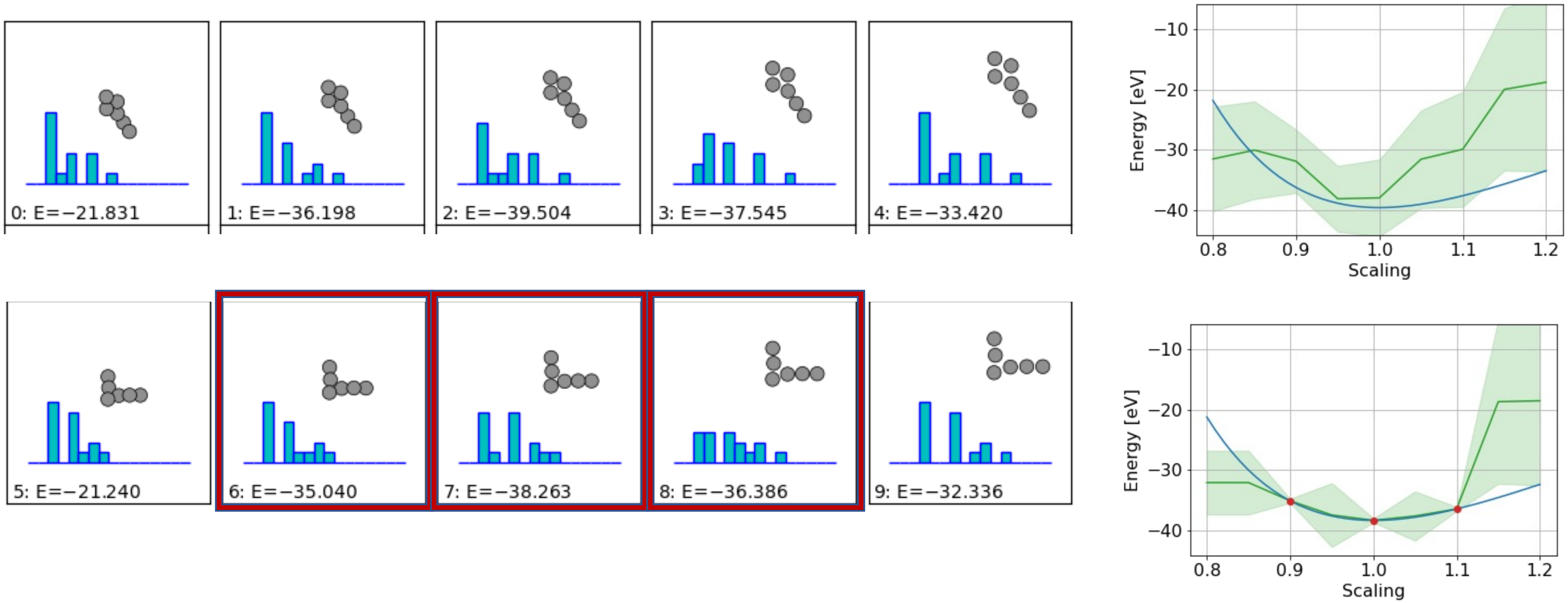
Gaussian Process Regression

$$\sigma_{sur}(\mathbf{x}_*)^2 = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^T (K + \sigma_n^2 I)^{-1} \mathbf{k}_*$$

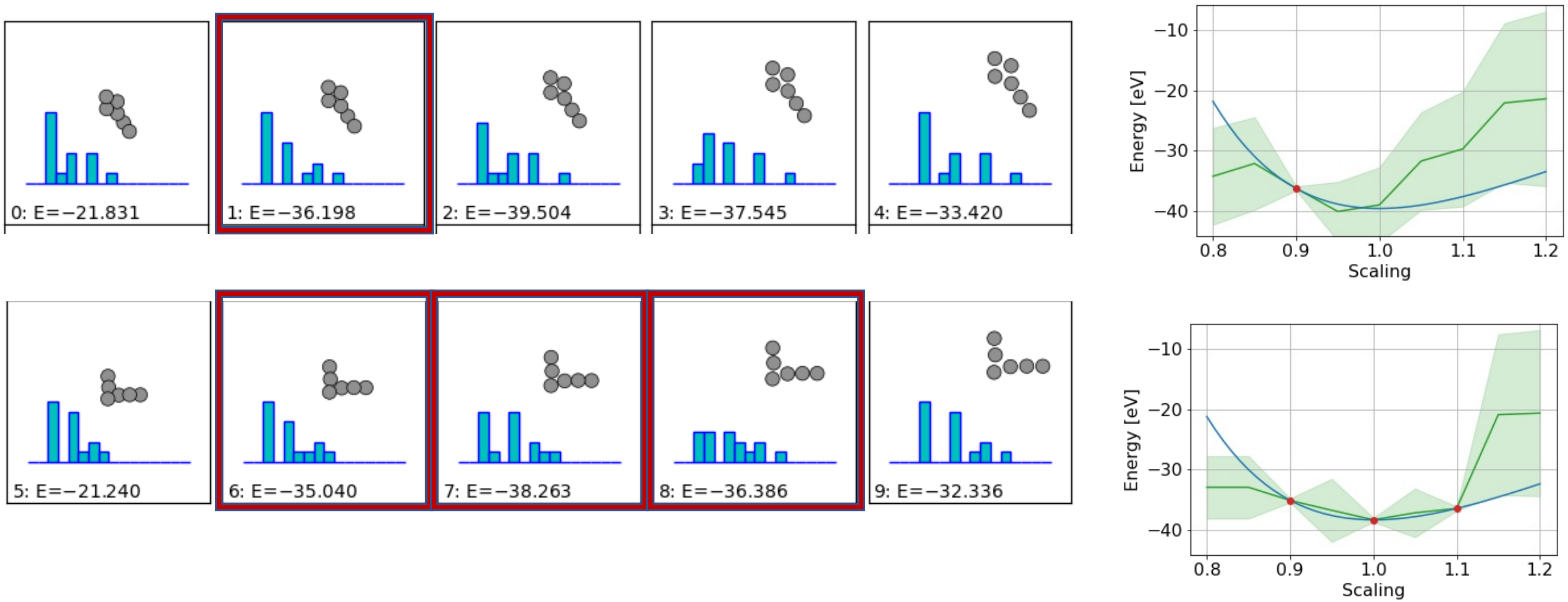
Gaussian Process Regression



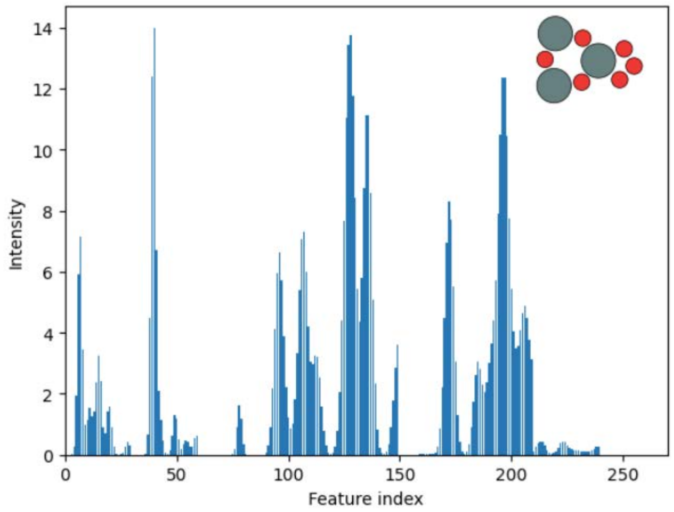
Gaussian Process Regression

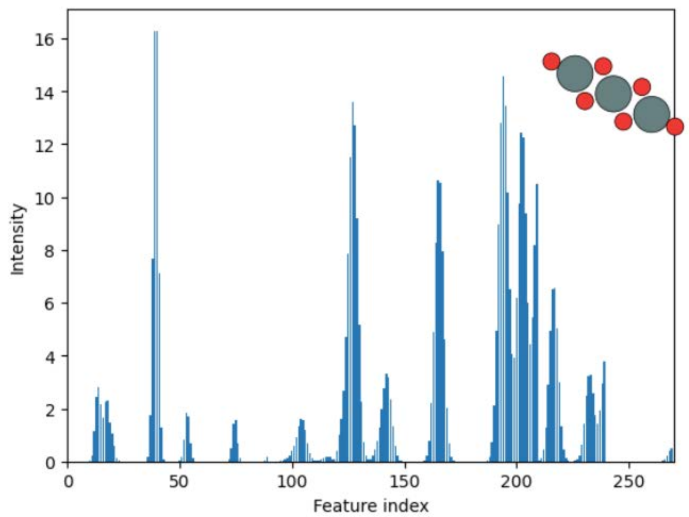


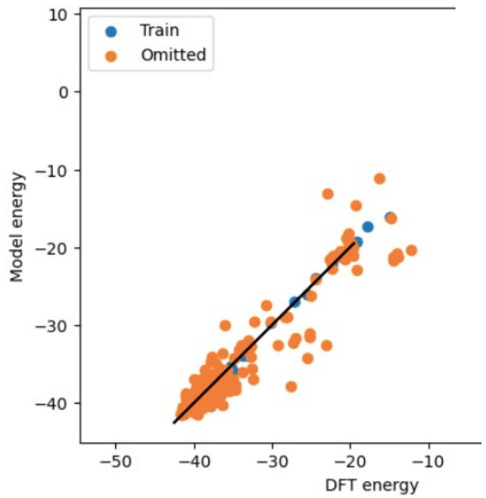
Gaussian Process Regression

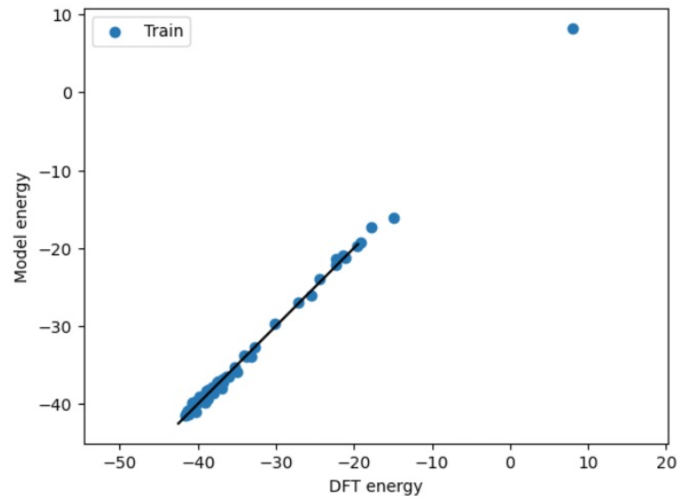


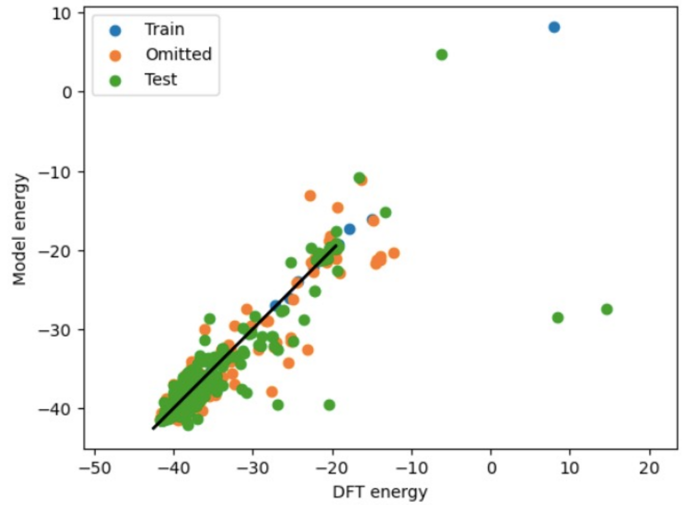
From the tutorial







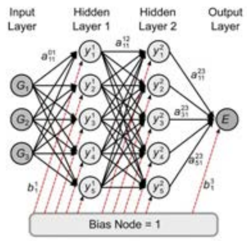





mldft.com
Hammer Group at Aarhus University on Machine Learning + DFT
Initial structure
$D_{mean} = 0.00$ Å
Rolemodel mutation
$D_{mean} = 1.20$ Å
Surrogate relaxation
$D_{mean} = 0.49$ Å
mldft.com
Sign in
mldft.com
2
Hammer group at Aarhus University
Choose a blog:
Hammer Group
3
mldft.com
Hammer Group
Login
News
1.1 Postdoc position
1.2 NOMAD workshop
Background
Methods
Papers
News
1.1 Postdoc position
A 2-year postition is available in the group.
1.2 NOMAD workshop
Checkout the exercises here.
4
1: Go to: mldft.com

Two popular options for the "black box"

2007:
Artificial Neural Networks



PHYSICAL REVIEW LETTERS

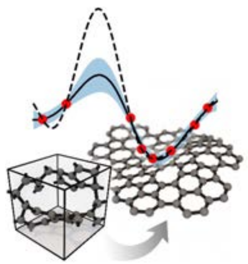
Highlights Recent Accepted Collections Authors Referees Search Press About

Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces

Jörg Behler and Michele Parrinello
Phys. Rev. Lett. **98**, 146401 – Published 2 April 2007

Citing Articles (1,783)

2010:
Gaussian Process Regression



PHYSICAL REVIEW LETTERS

Highlights Recent Accepted Collections Authors Referees Search Press About

Featured in Physics

Gaussian Approximation Potentials: The Accuracy of Quantum Mechanics, without the Electrons

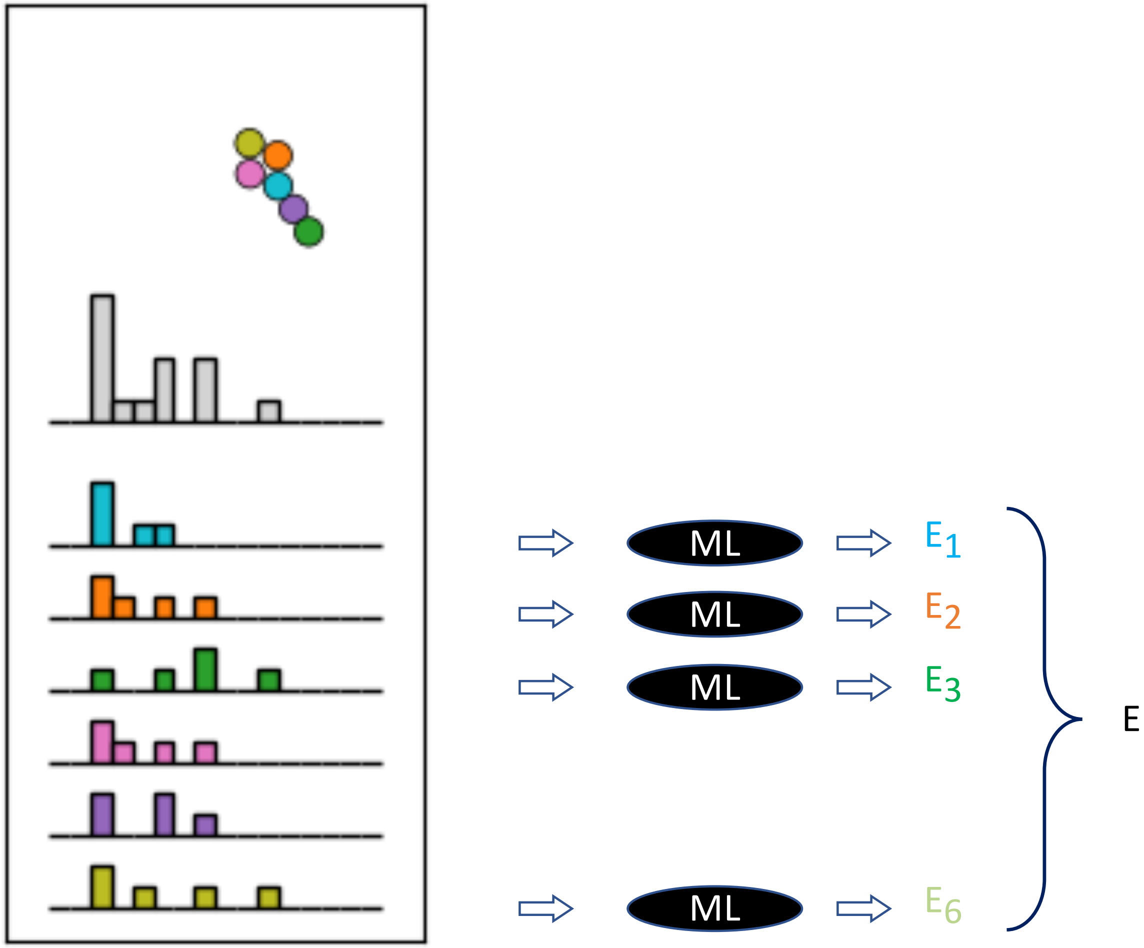
Albert P. Bartók, Mike C. Payne, Risi Kondor, and Gábor Csányi
Phys. Rev. Lett. **104**, 136403 – Published 1 April 2010

Citing Articles (1,227)

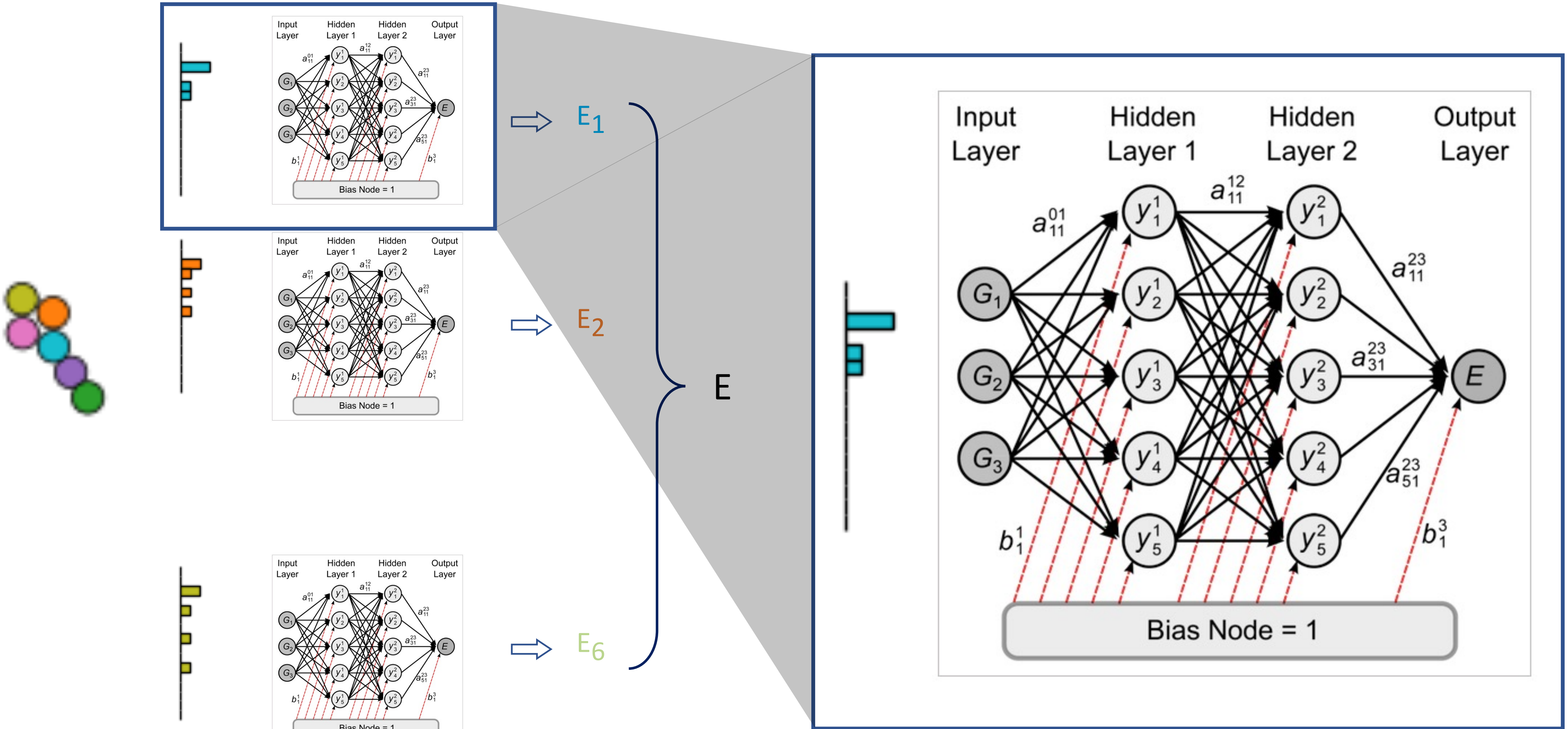
In fact, both methods use: a descriptor for <u>each atom</u>

Histogram of all interatomic distances

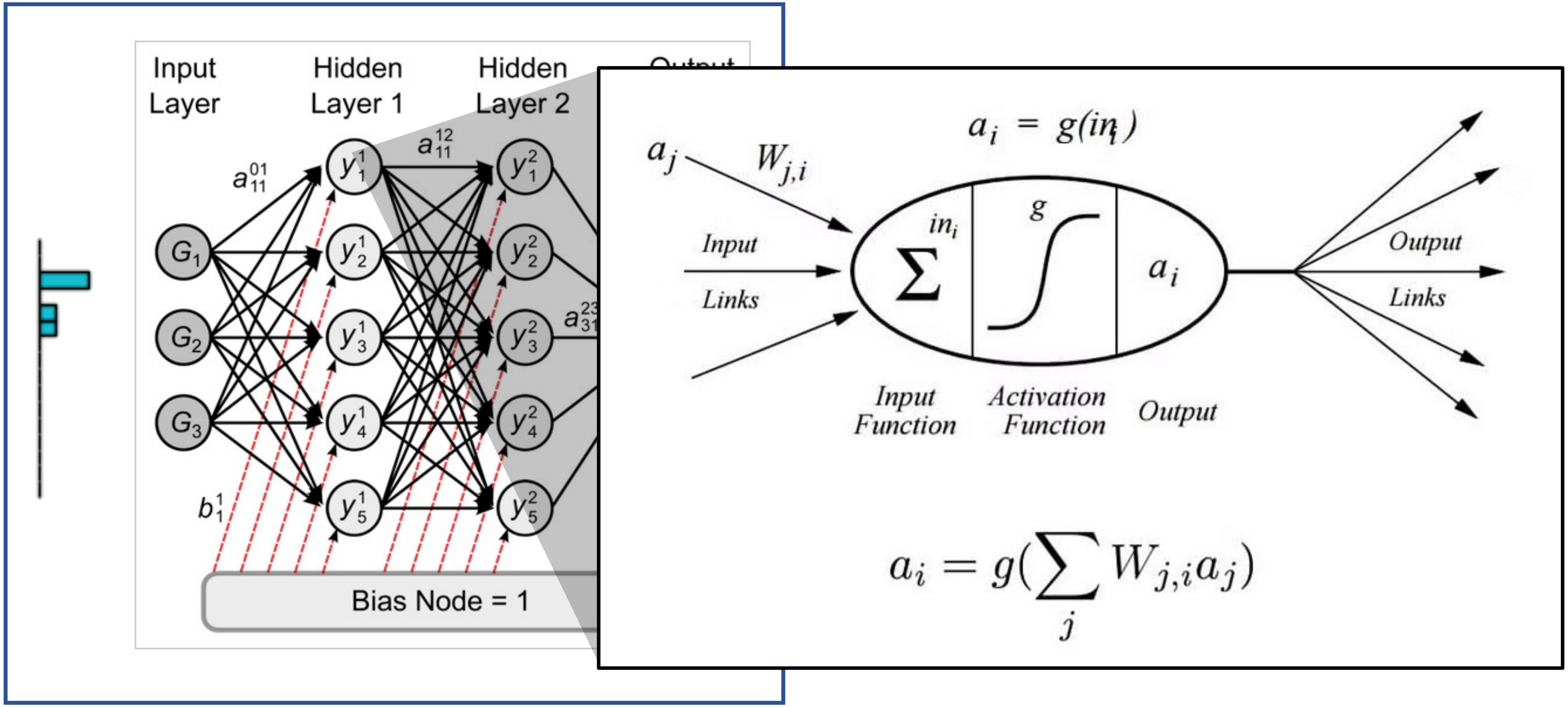
Histograms for each atom in molecule



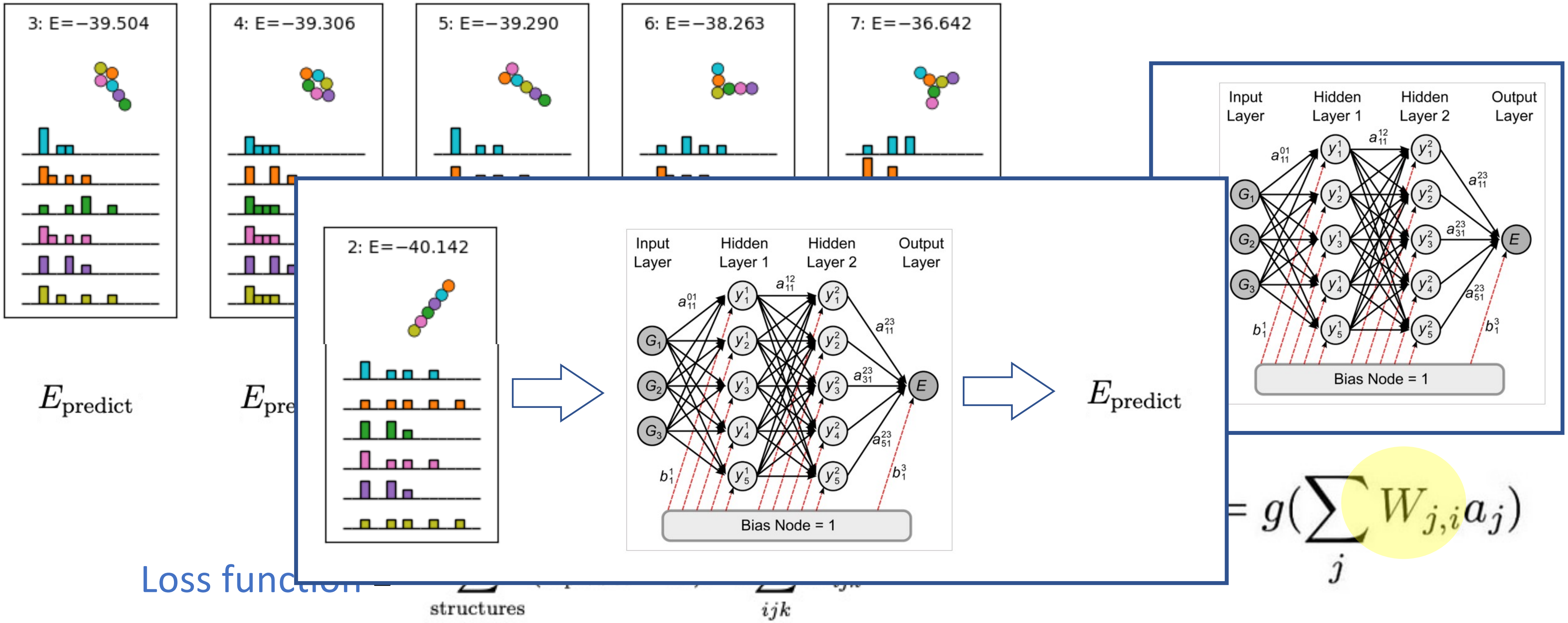
Artificial Neural Network approach



The neuron


Input Layer
Hidden Layer 1
Hidden Layer 2
Bias Node = 1
a_j
$W_{j,i}$
$a_i = g(in_i)$
Input Links
in_i
g
a_i
Output Links
Input Function
Activation Function
Output
$a_i = g(\sum_j W_{j,i} a_j)$

The training

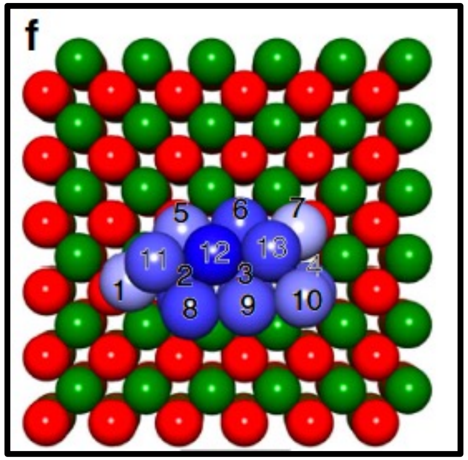


Loss function = ... structures ... ijk

Update W’s to minimize Loss function

Example: structural search:

2015:
Georgia Tech
Uzi Landman group
13 Pt atoms / MgO support
Relaxed 36 configurations
approx 3600 DFT calcs



Computational cost

DFT

Tight-binding

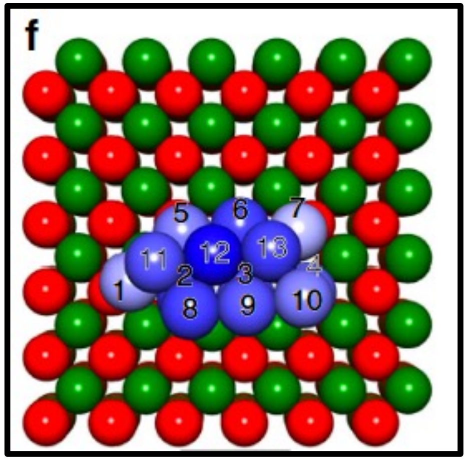
Lennard-Jones

Accuracy

Example: structural search:

Computational cost

2015:
Georgia Tech
Uzi Landman group
13 Pt atoms / MgO support
Relaxed 36 configurations
approx 3600 DFT calcs



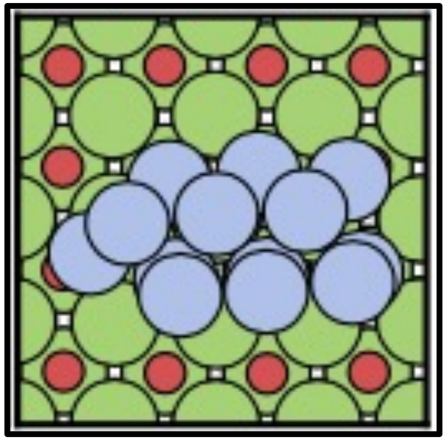
DFT

ML

Accuracy

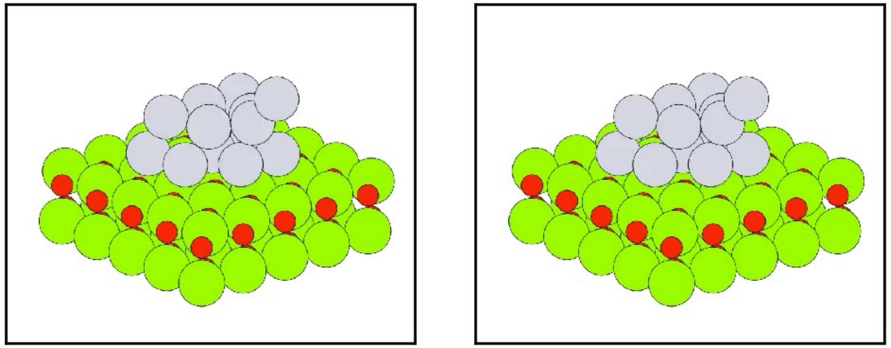
2018:
Aarhus University

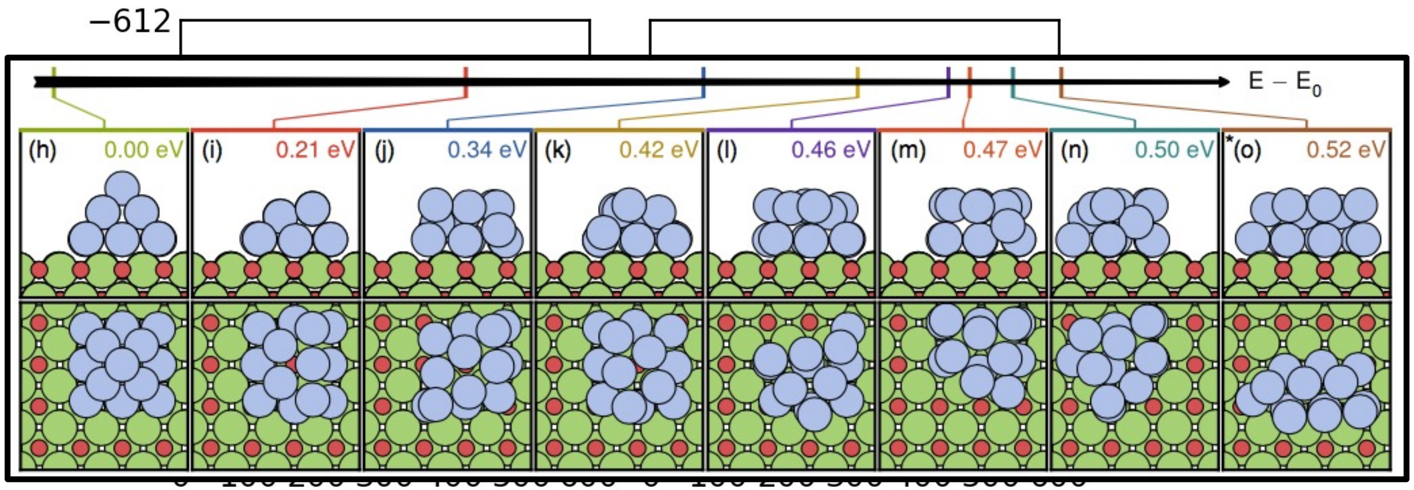
1800 DFT calcs
Relaxed 3 x 600 configurations
approx 180,000 ML calcs



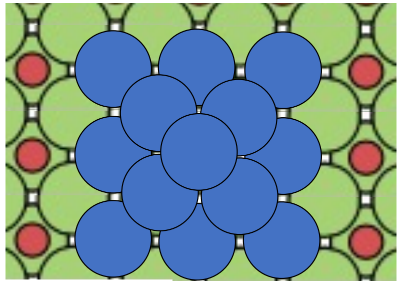
Monte-Carlo type search using ML: Pt_{13}/MgO

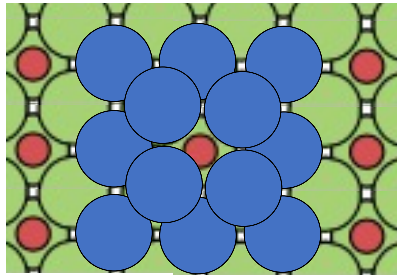
Esben Kolsbjerg, A A Peterson, BH Phys Rev B **97**, 195424 (2018)



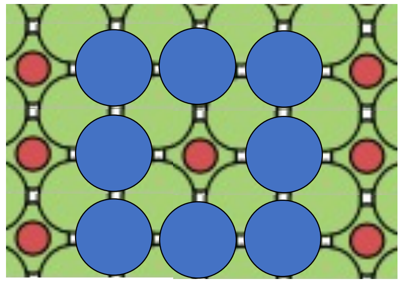


3rd layer: +1 atom





2nd layer:
+ 4 atoms

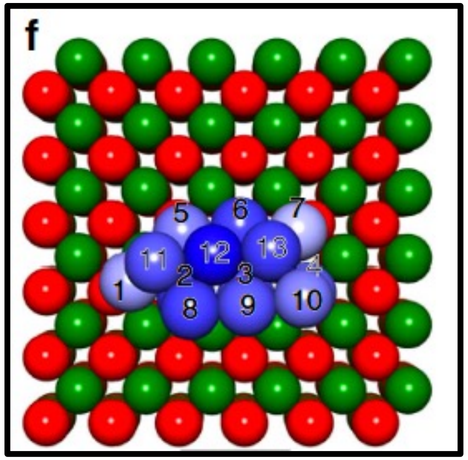


Bottom layer:
8 atoms

Example: structural search:

Computational cost

2015:
Georgia Tech
Uzi Landman group
13 Pt atoms / MgO support
Relaxed 36 configurations
approx 3600 DFT calcs

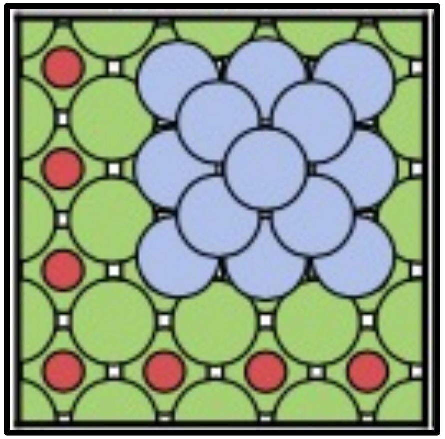


DFT

ML

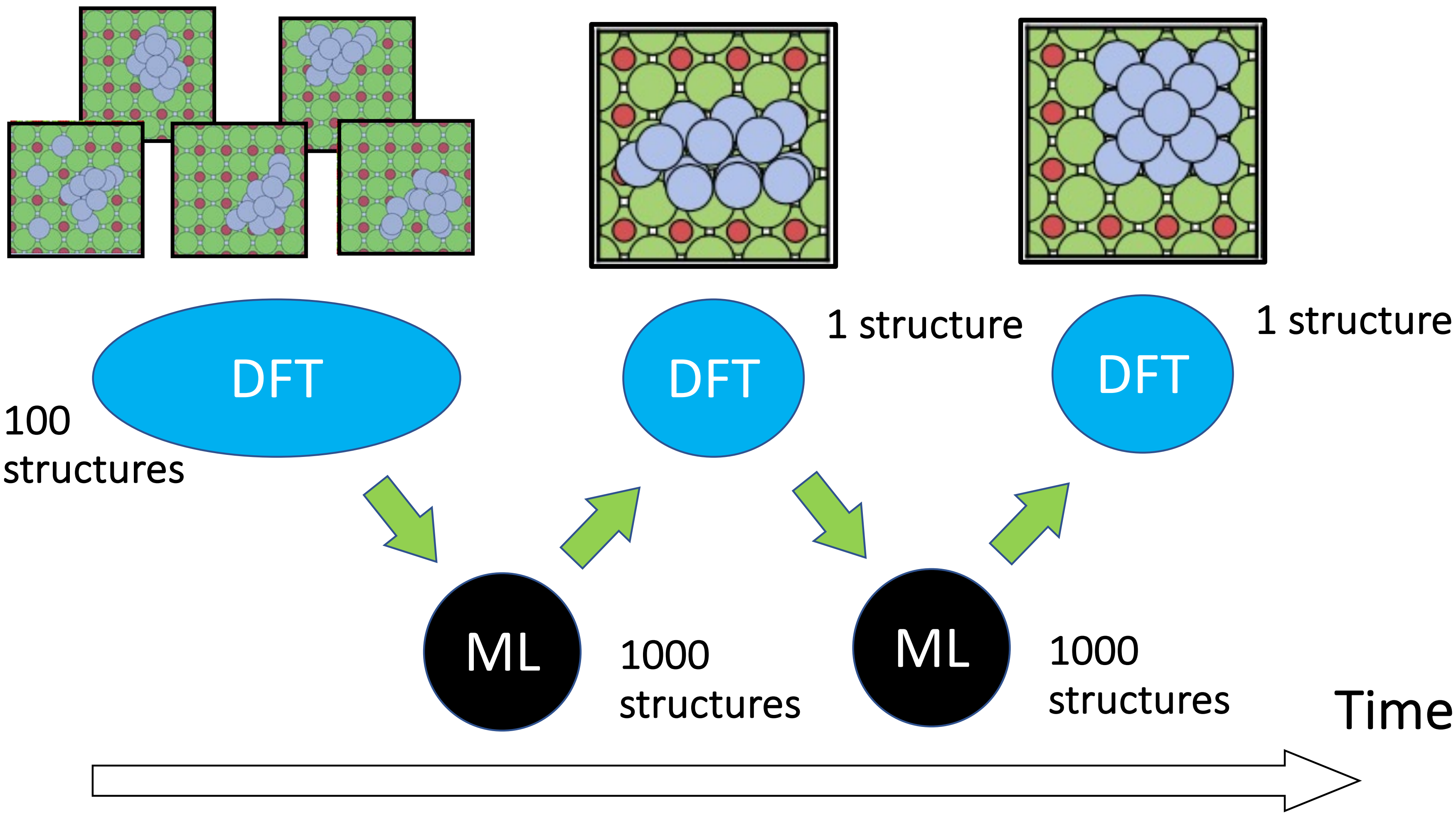
2018:
Aarhus University

1800 DFT calcs
Relaxed 3 x 600 configurations
approx 180,000 ML calcs



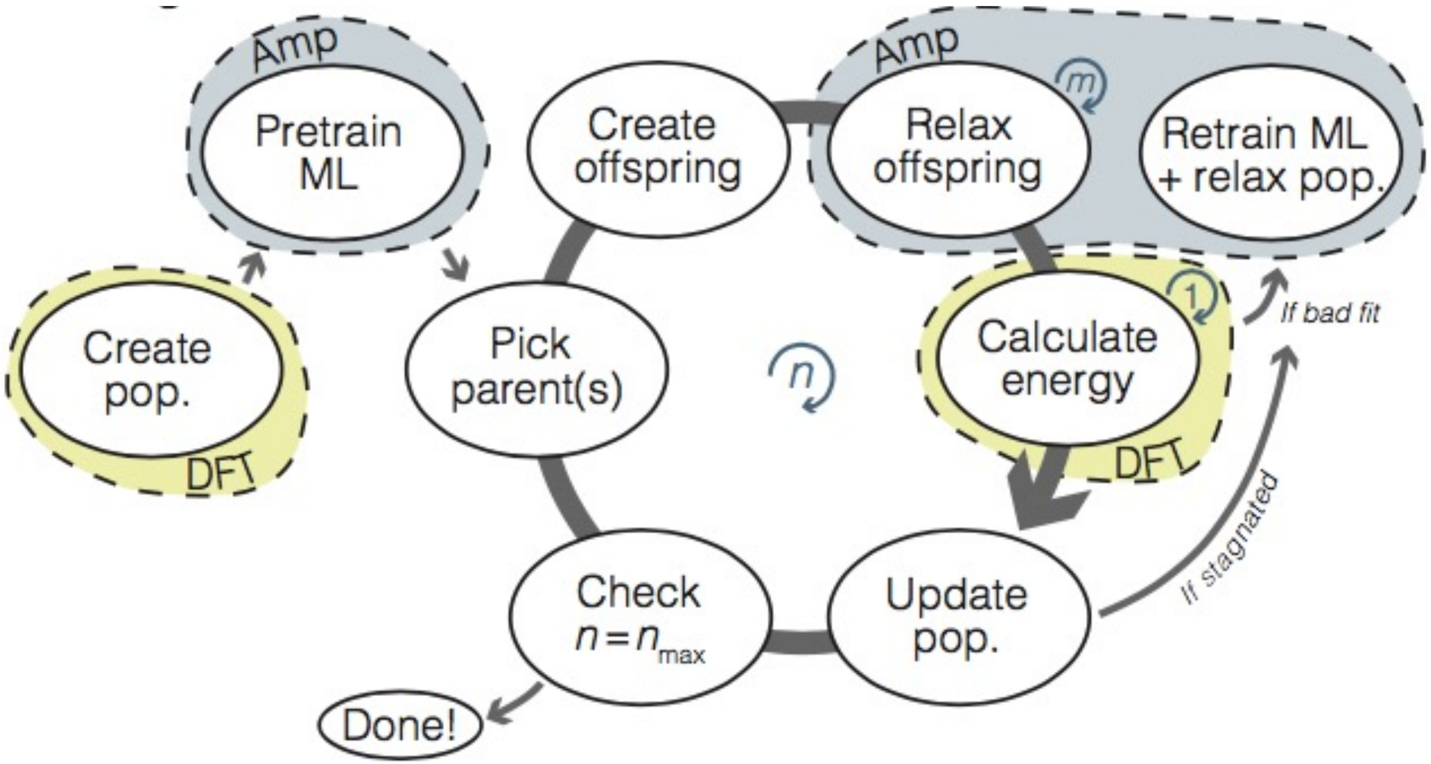
Accuracy

Active learning



Learning Evolutionary Algorithm: LEA

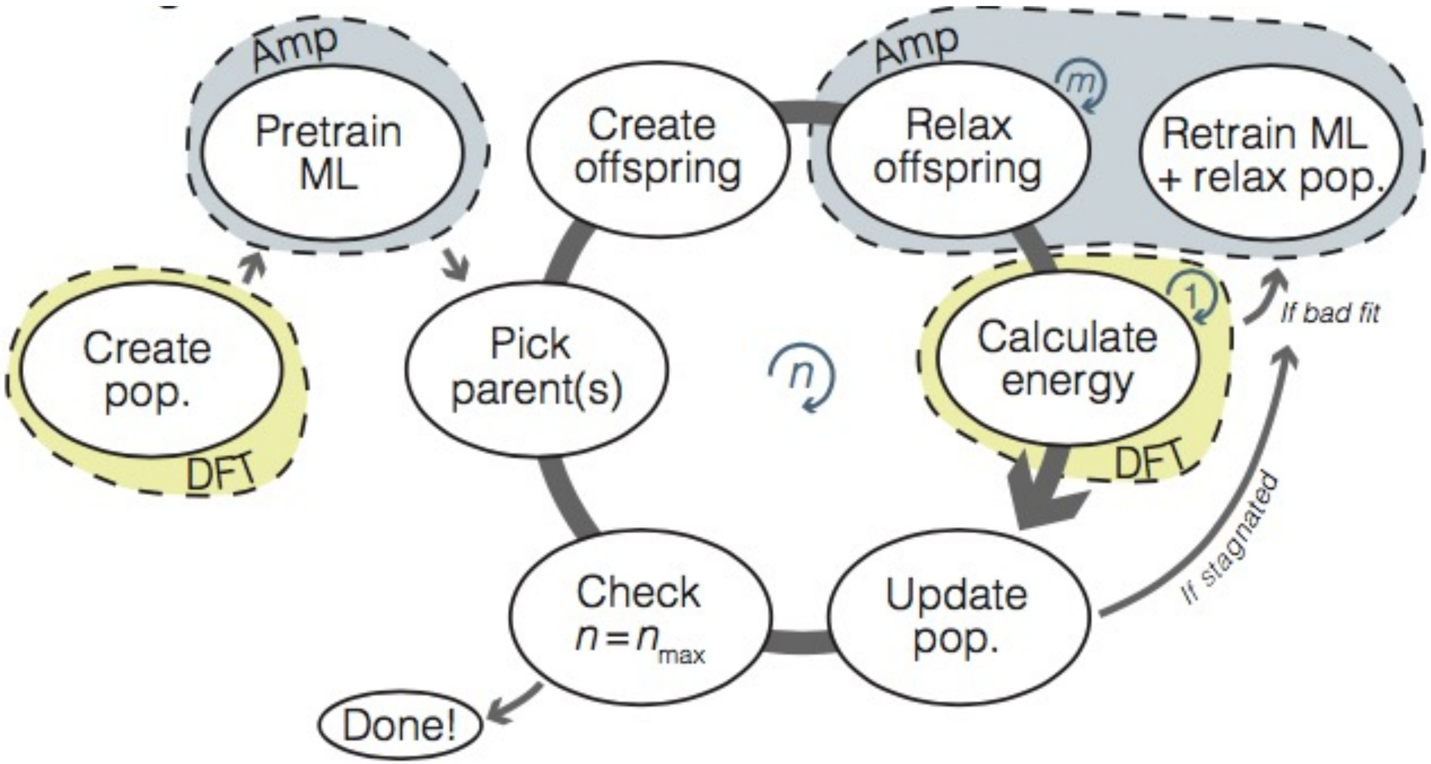
ML = Machine Learning

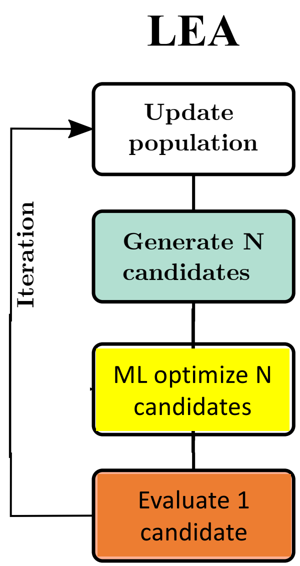


Esben Kolsbjerg, Andrew A Peterson, BH, Phys Rev B **97**, 195424 (2018)

Learning Evolutionary Algorithm: LEA

ML = Machine Learning

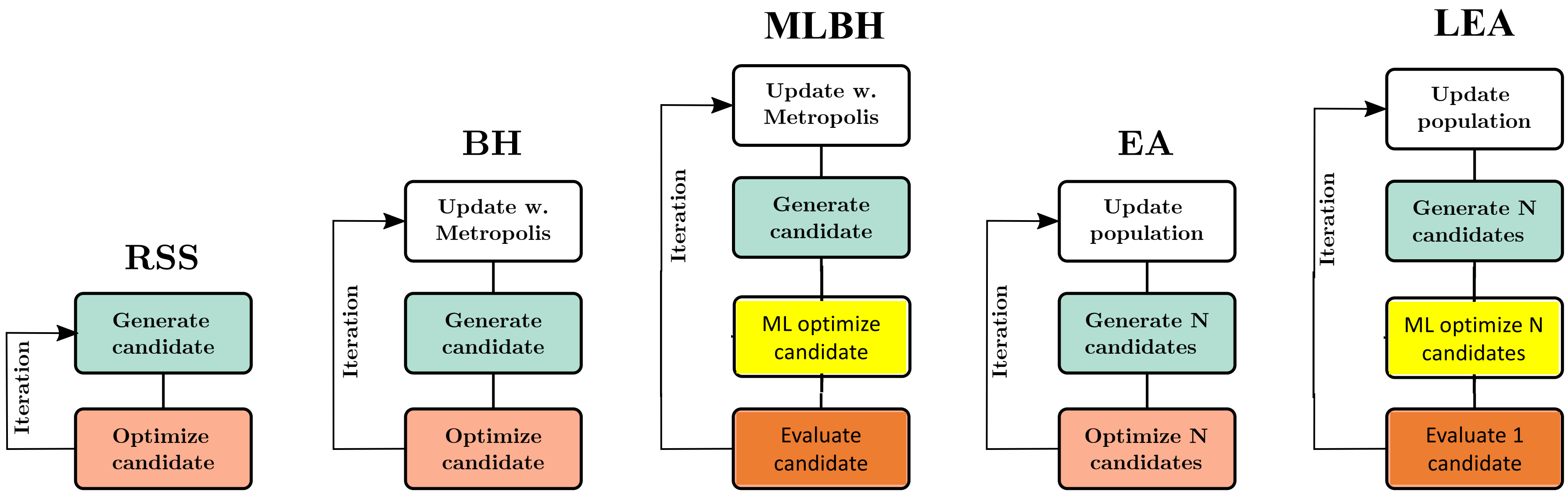




Esben Kolsbjerg, Andrew A Peterson, BH, Phys Rev B **97**, 195424 (2018)

Common elements in Atomistic Global Optimization: AGOX

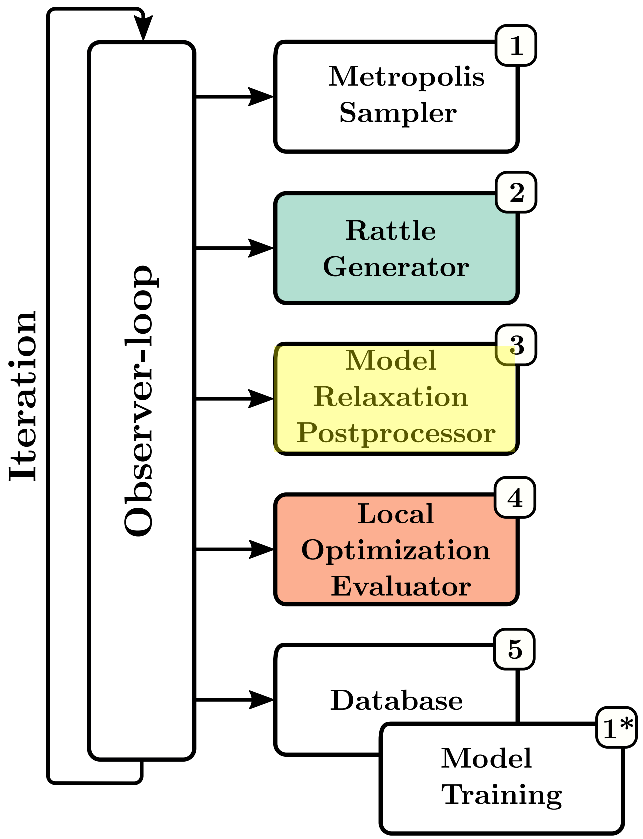
Christiansen et al, JCP **157**, 054701 (2022)



AGOX: ”Atomistic Global Optimization X”

Christiansen et al, JCP **157**, 054701 (2022)   https://gitlab.com/agox

ML-enhanced basin-hopping



Define a model:

```
model = ModelGPR.default(environment, database)
```

Define a module that uses this model:

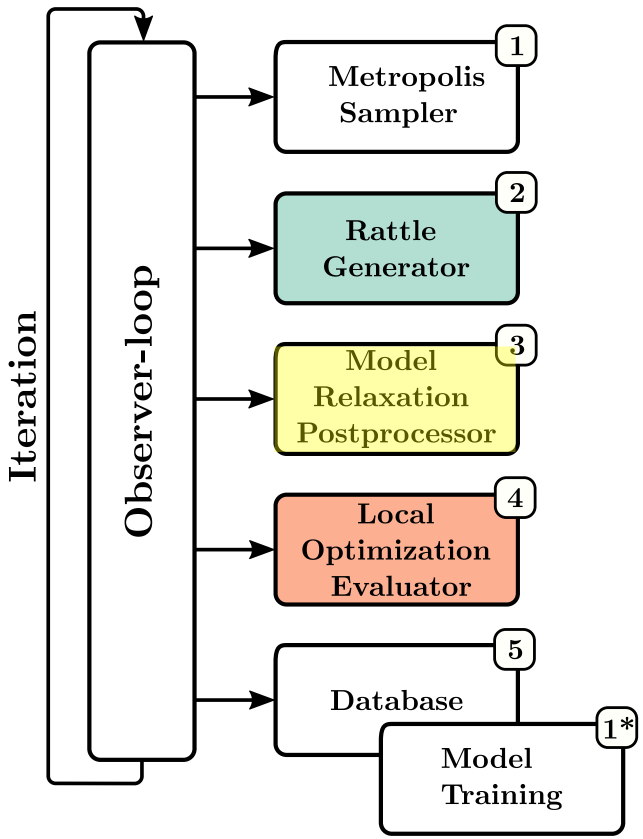
```
relaxer = RelaxPostprocess(model=model,
    start_relax=10,
    optimizer_run_kwargs={'fmax':0.05, 'steps':100},
    optimizer_kwargs ={'logfile':'-'},
    optimizer=BFGS, constraints=[BC], order=2)
```

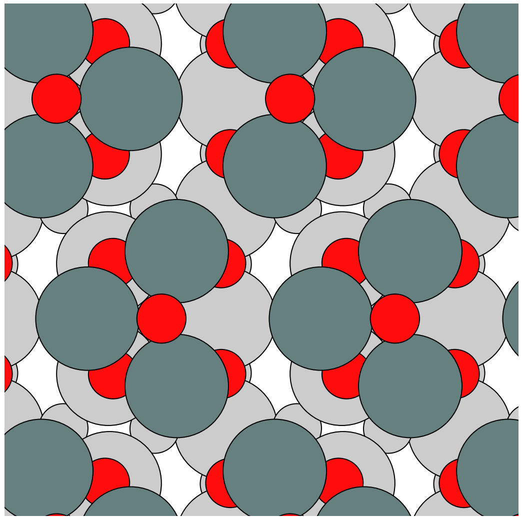
Give to AGOX

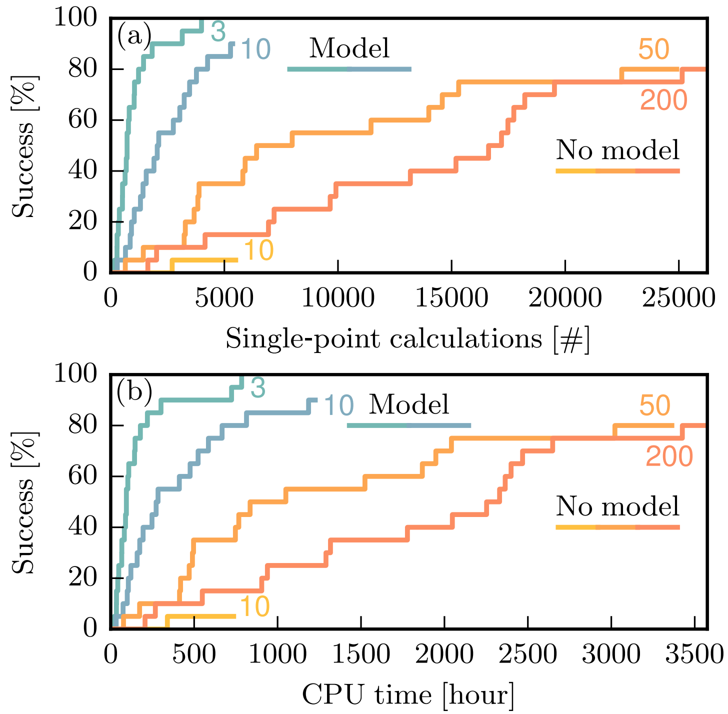
```
agox = AGOX(database, collector, sampler, evaluator,
            wrapper, relaxer, seed=run_idx)
```

Basin hopping search with local relaxation in GPR model

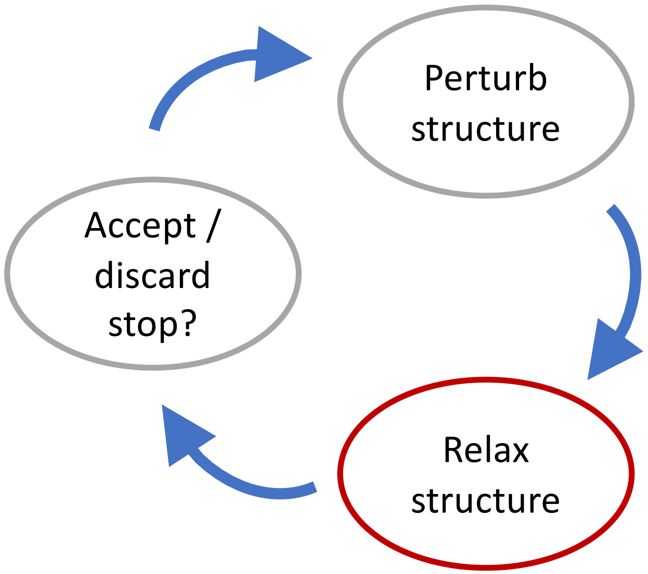
ML-enhanced basin-hopping

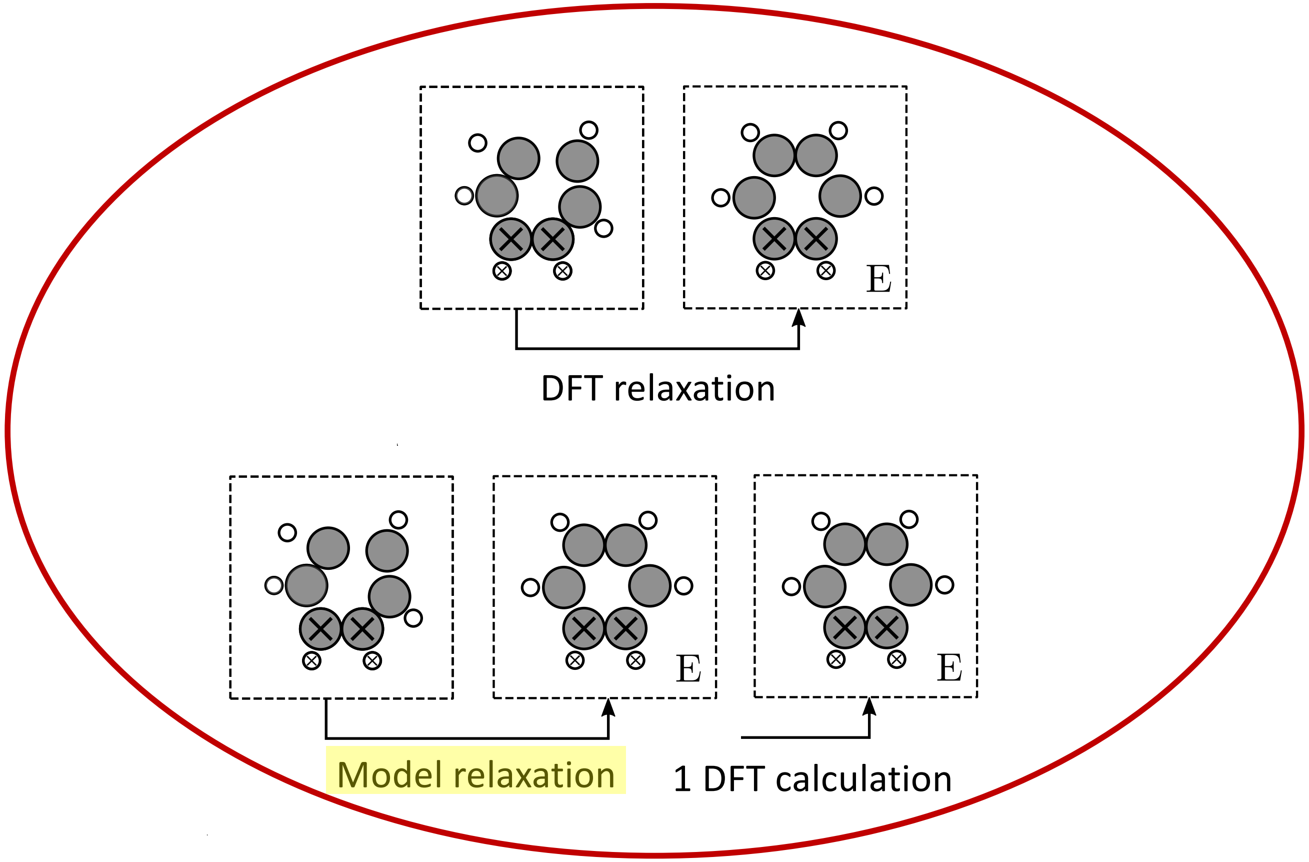






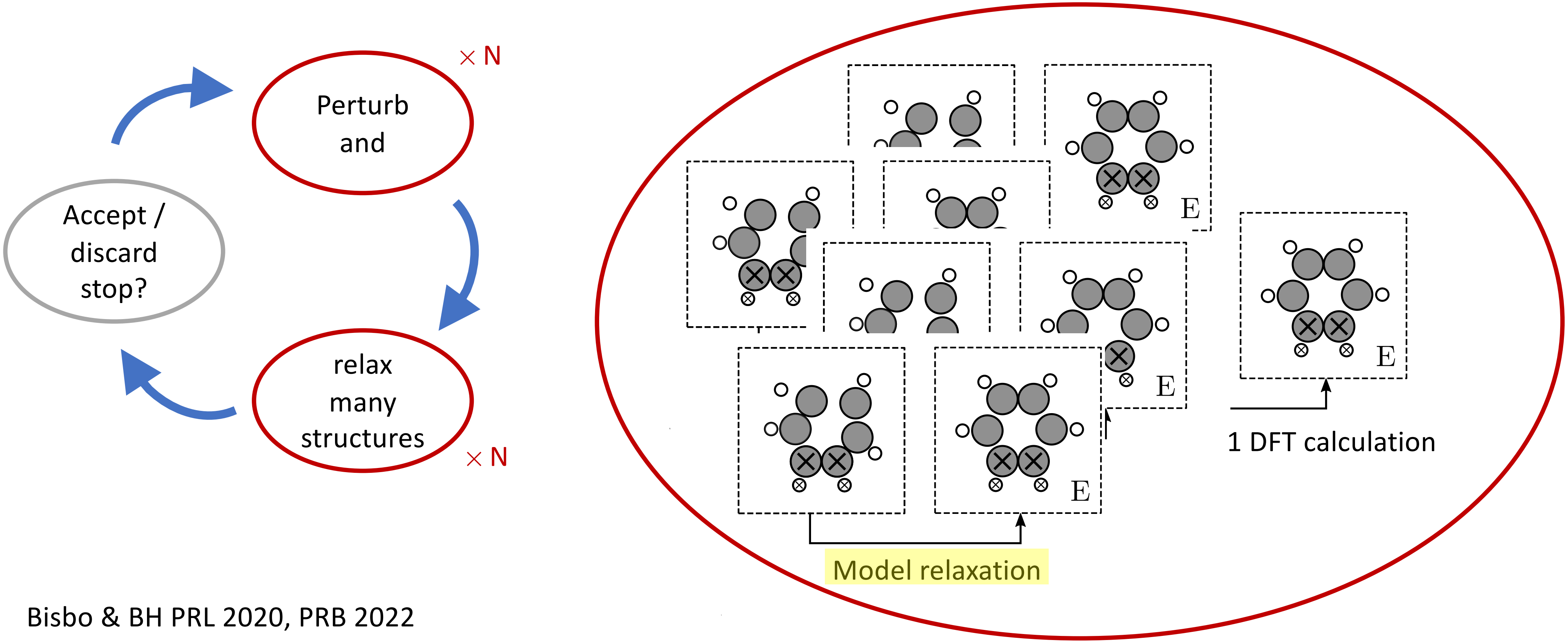
Structural search: Trick of relaxing in surrogate potential





Christiansen et al, JCP **157**, 054701 (2022)

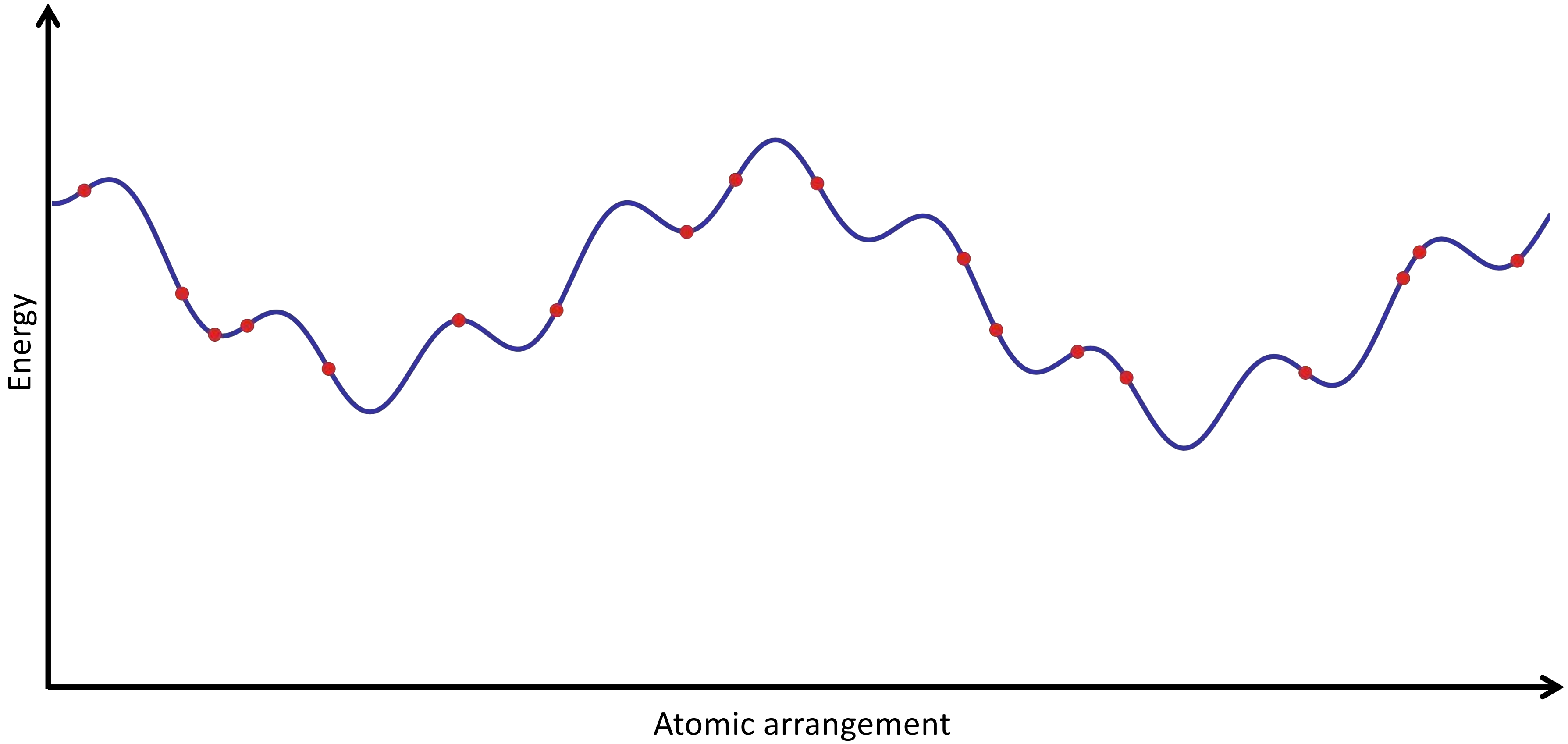
Global optimization of first-principles energy expressions (GOFEE)



Bisbo & BH PRL 2020, PRB 2022

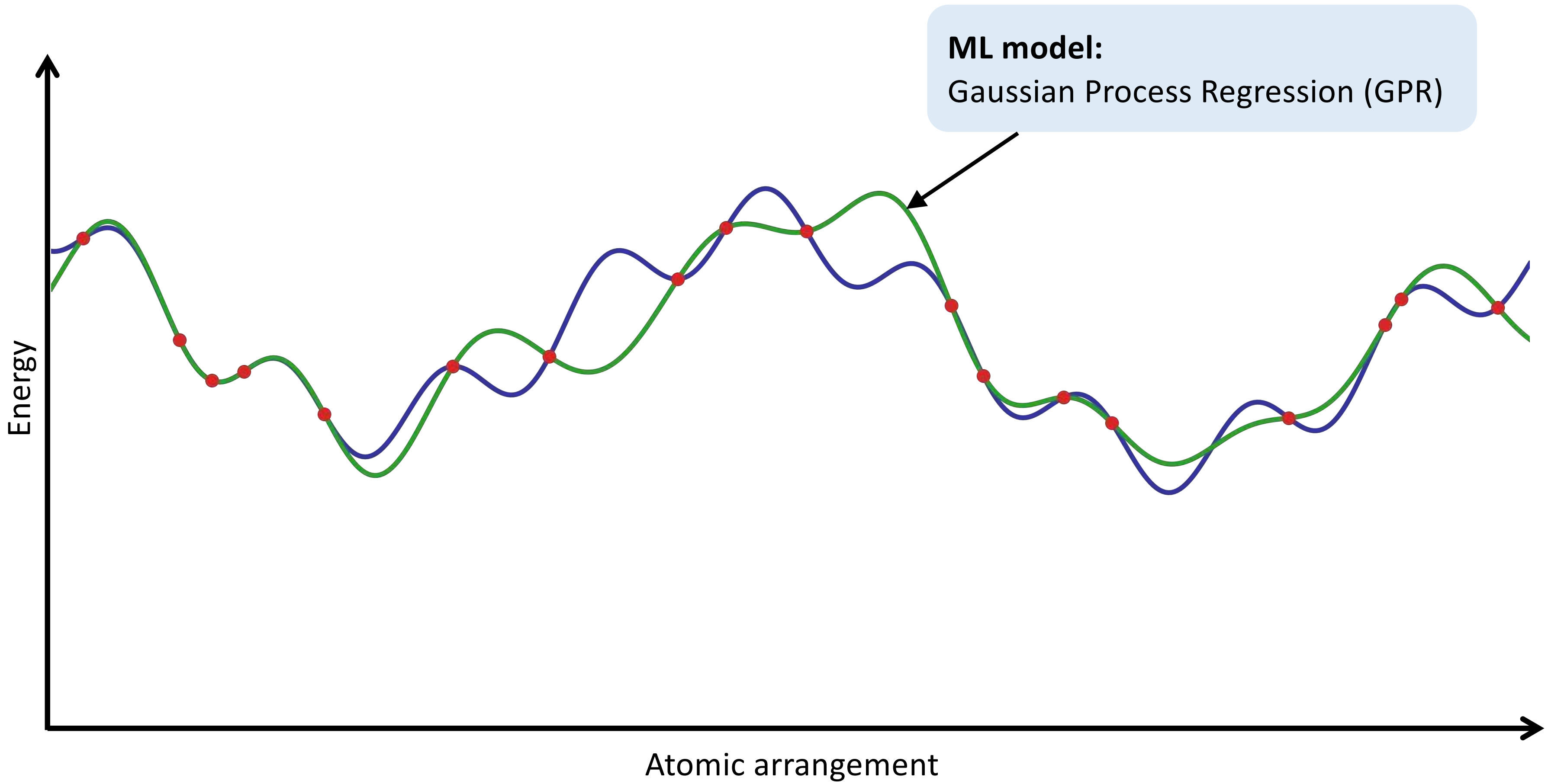
Global optimization of first-principles energy expressions (GOFEE)

(Bayesian statistics)



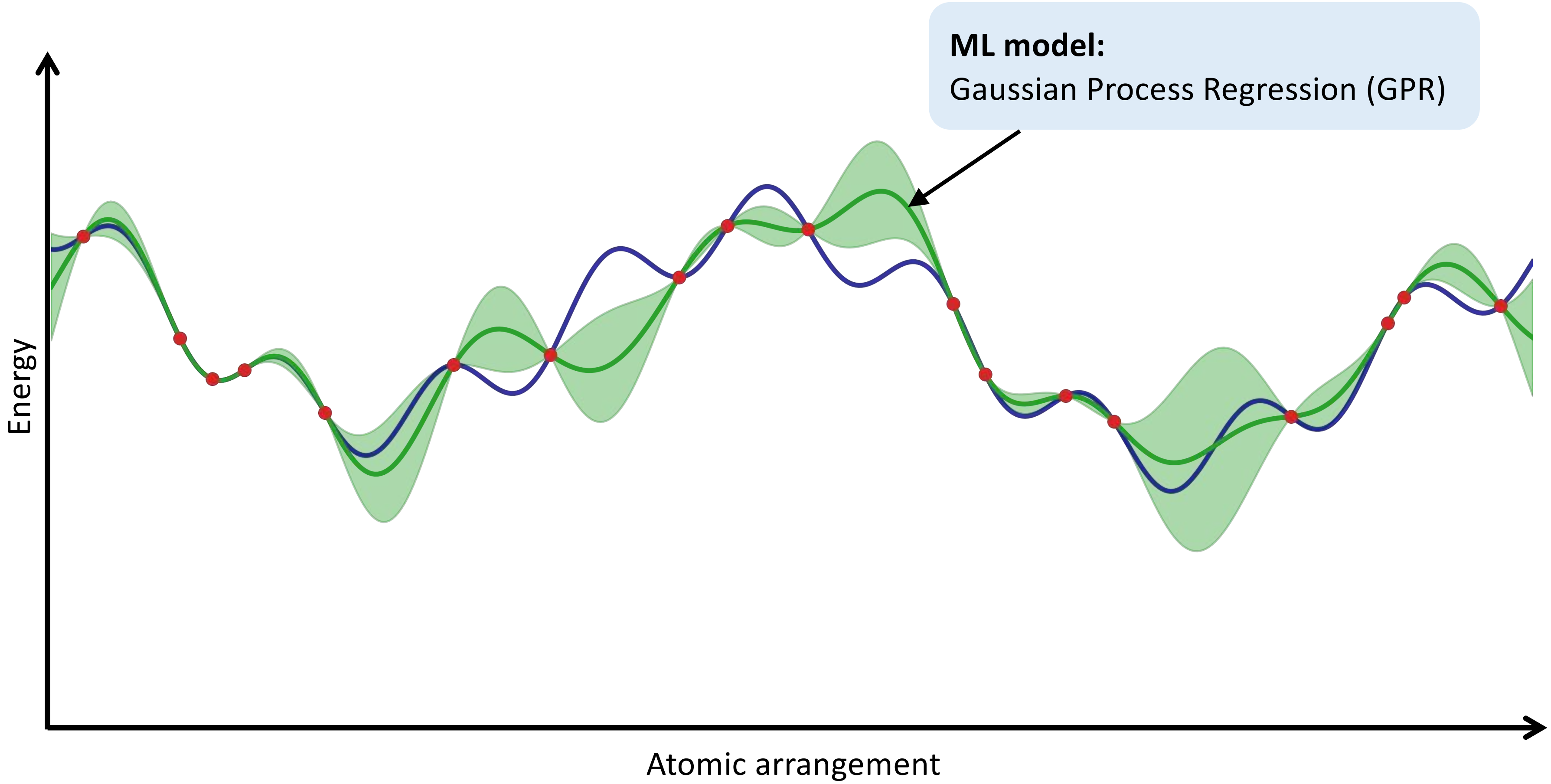
Global optimization of first-principles energy expressions (GOFEE)

(Bayesian statistics)



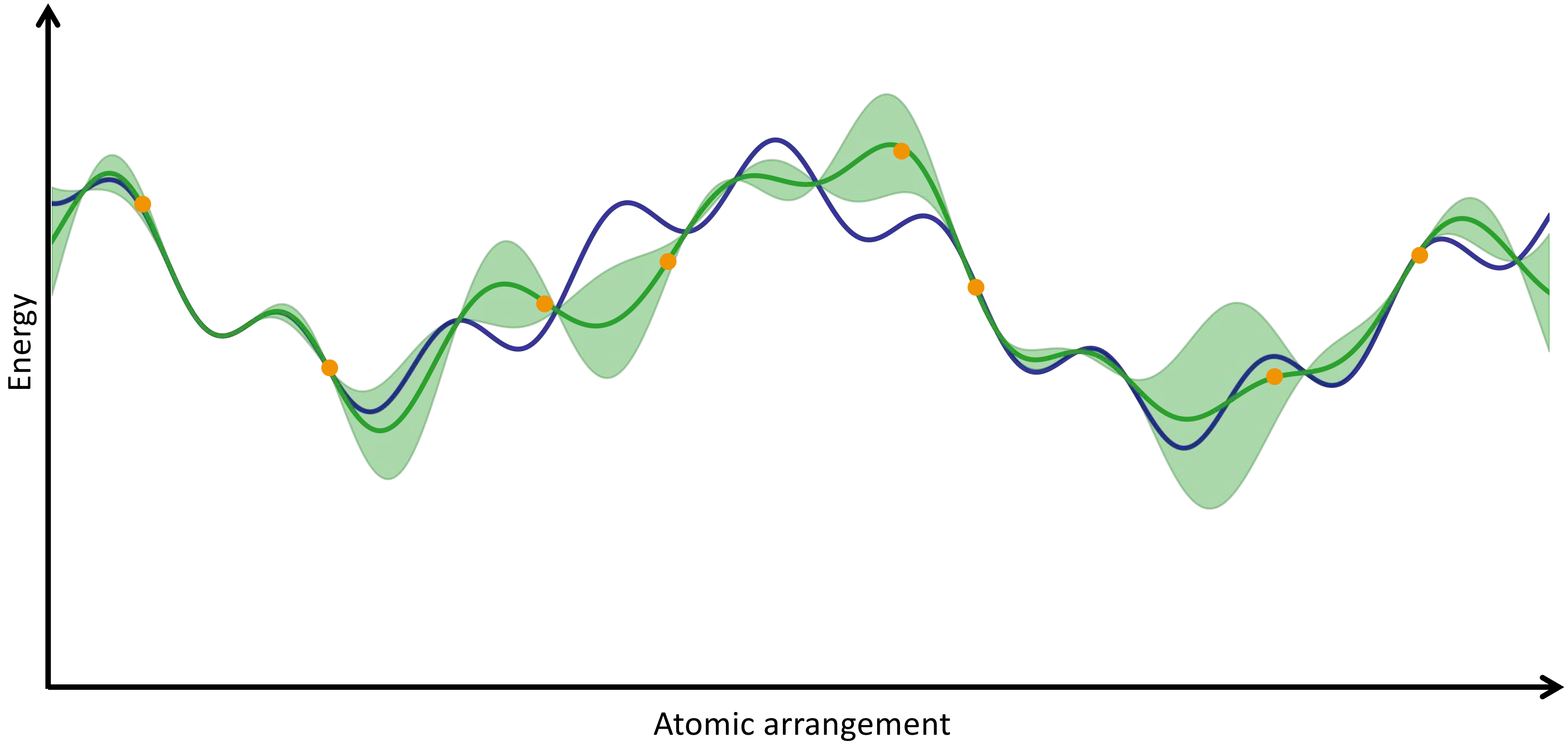
Global optimization of first-principles energy expressions (GOFEE)

Bayesian statistics



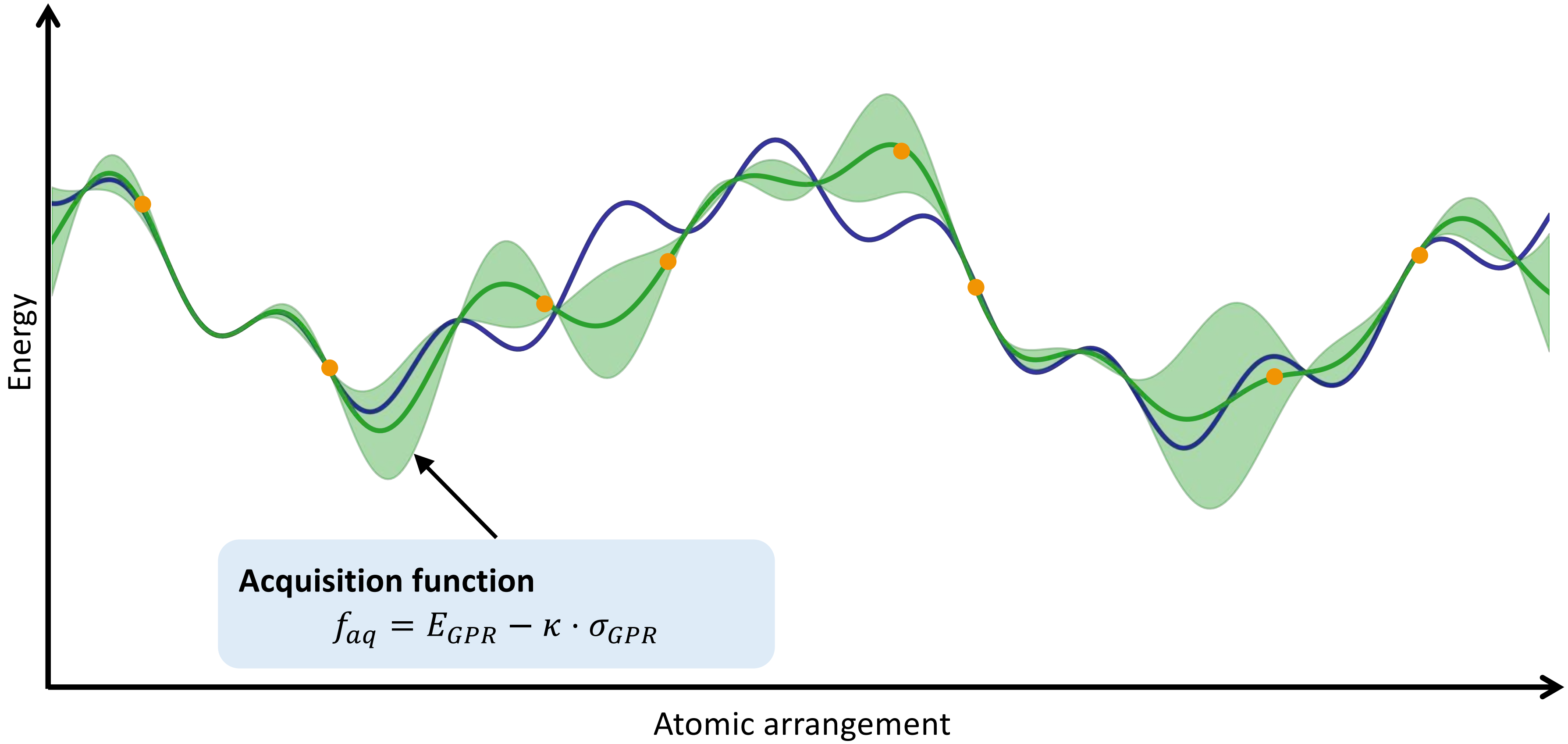
Global optimization of first-principles energy expressions (GOFEE)

Bayesian statistics



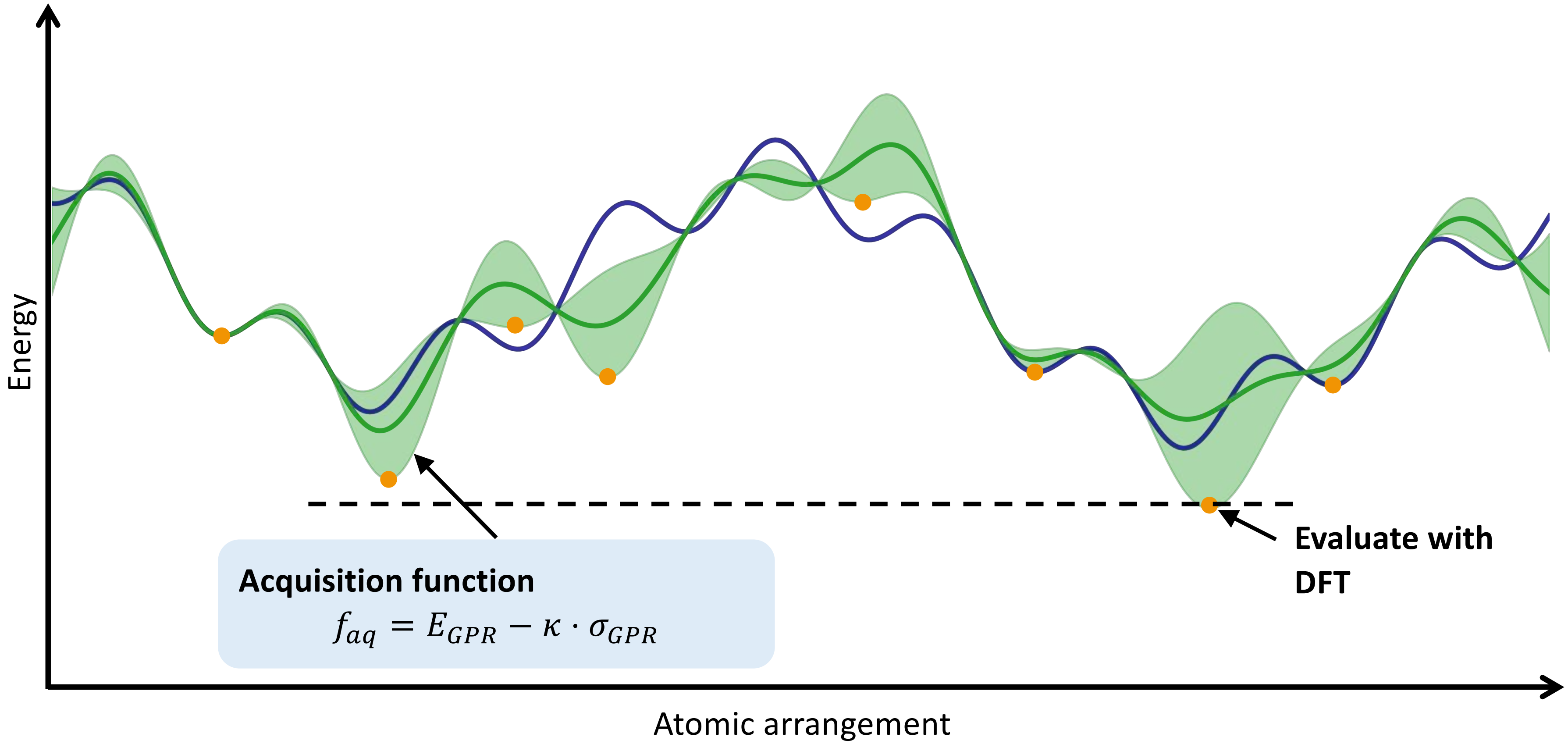
Global optimization of first-principles energy expressions (GOFEE)

Bayesian statistics



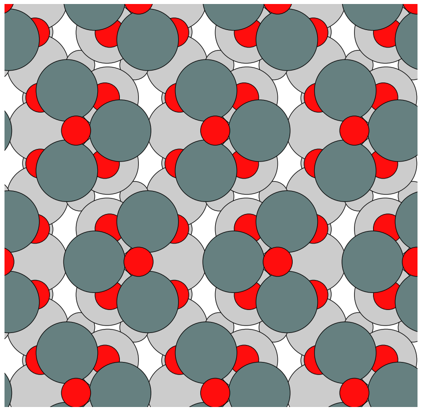
Global optimization of first-principles energy expressions (GOFEE)

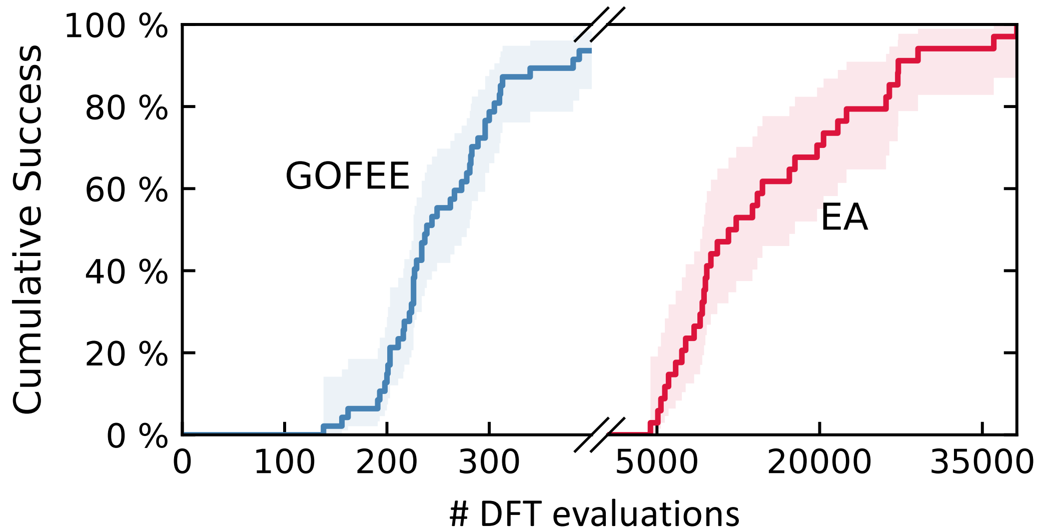
Bayesian statistics



Results – the reduced SnO_2 surface reconstruction

SnO_2(110)-(4x1)





GOFEE =
1) Adaptive learning
2) Relaxation in Acquisition function: $f_{aq} = E_{GPR} - \kappa \cdot \sigma_{GPR}$
3) Multiple candidates at model level, one at DFT level

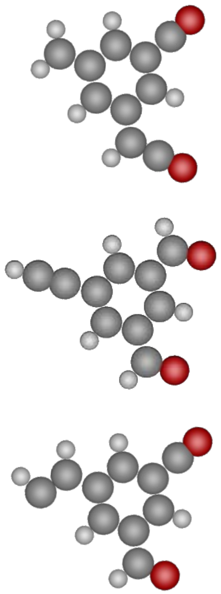
Bisbo & BH
PRL 2020, PRB 2022

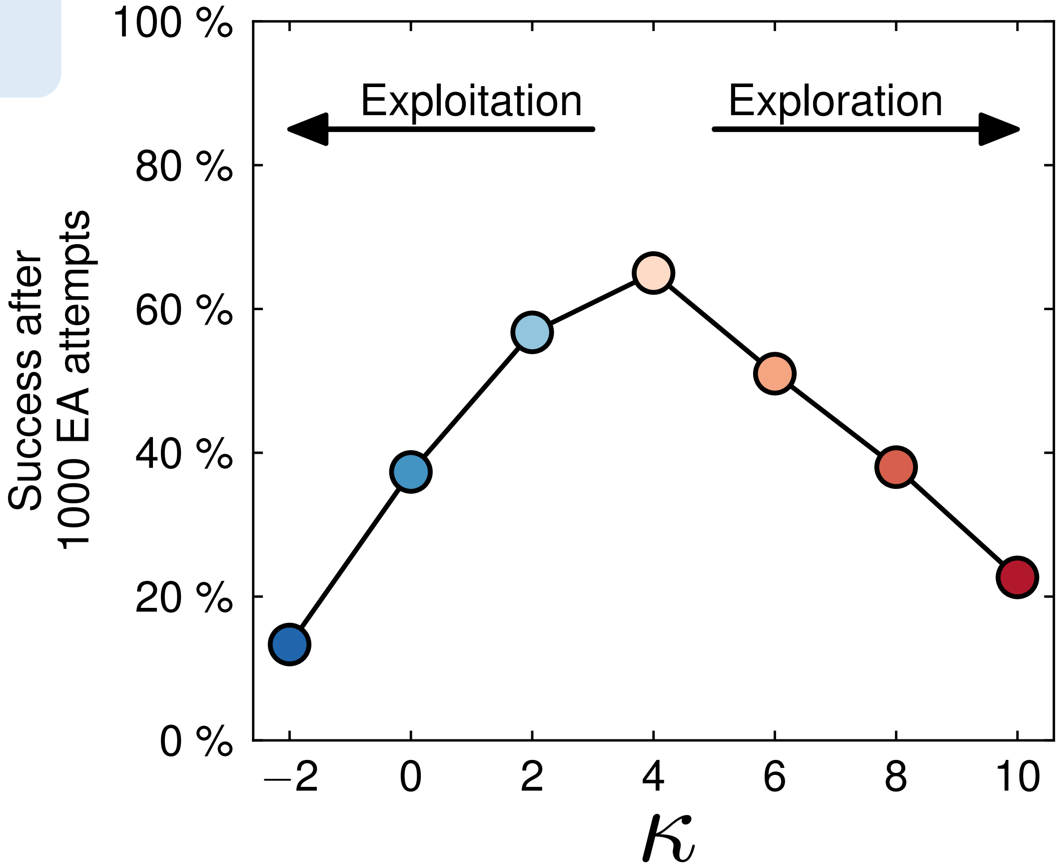
Favoring low energy or high uncertainty

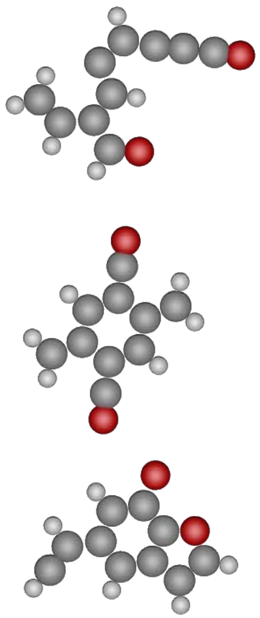
M S Jørgensen *et al.*, J Phys Chem A, **122**, 1504 (2018)

Acquisition function

$$f_{aq} = E_{GPR} - \kappa \cdot \sigma_{GPR}$$

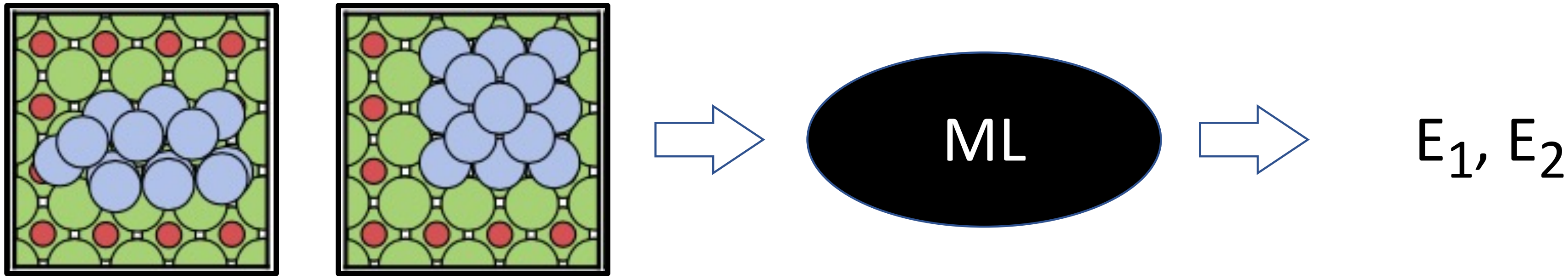




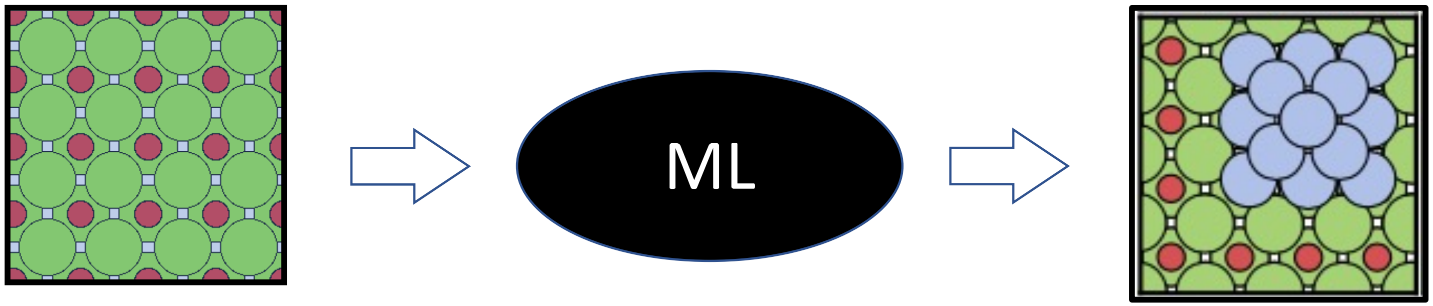


AARHUS
UNIVERSITY

Machine learned potential

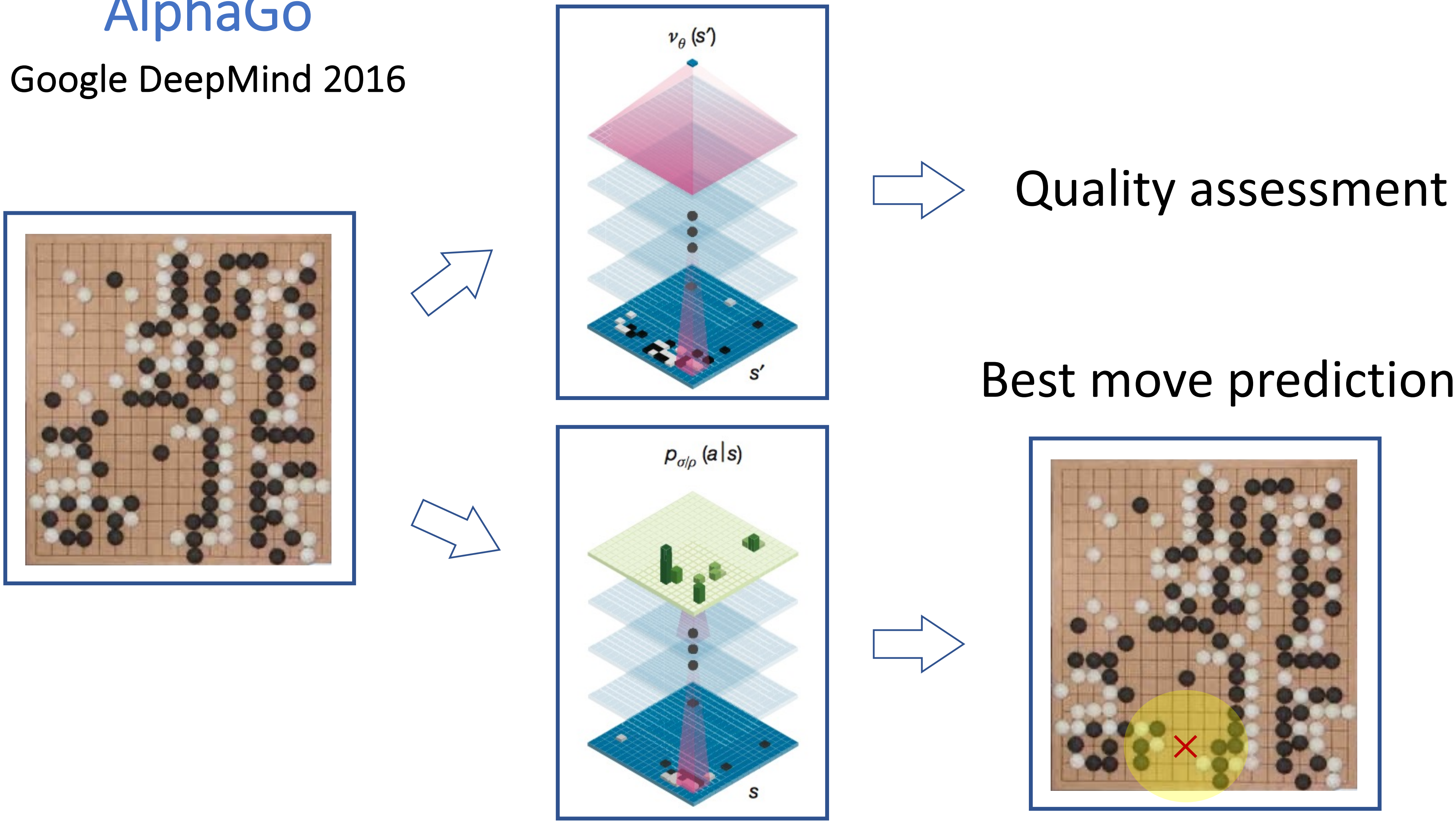


Artificial intelligence



AlphaGo

Google DeepMind 2016

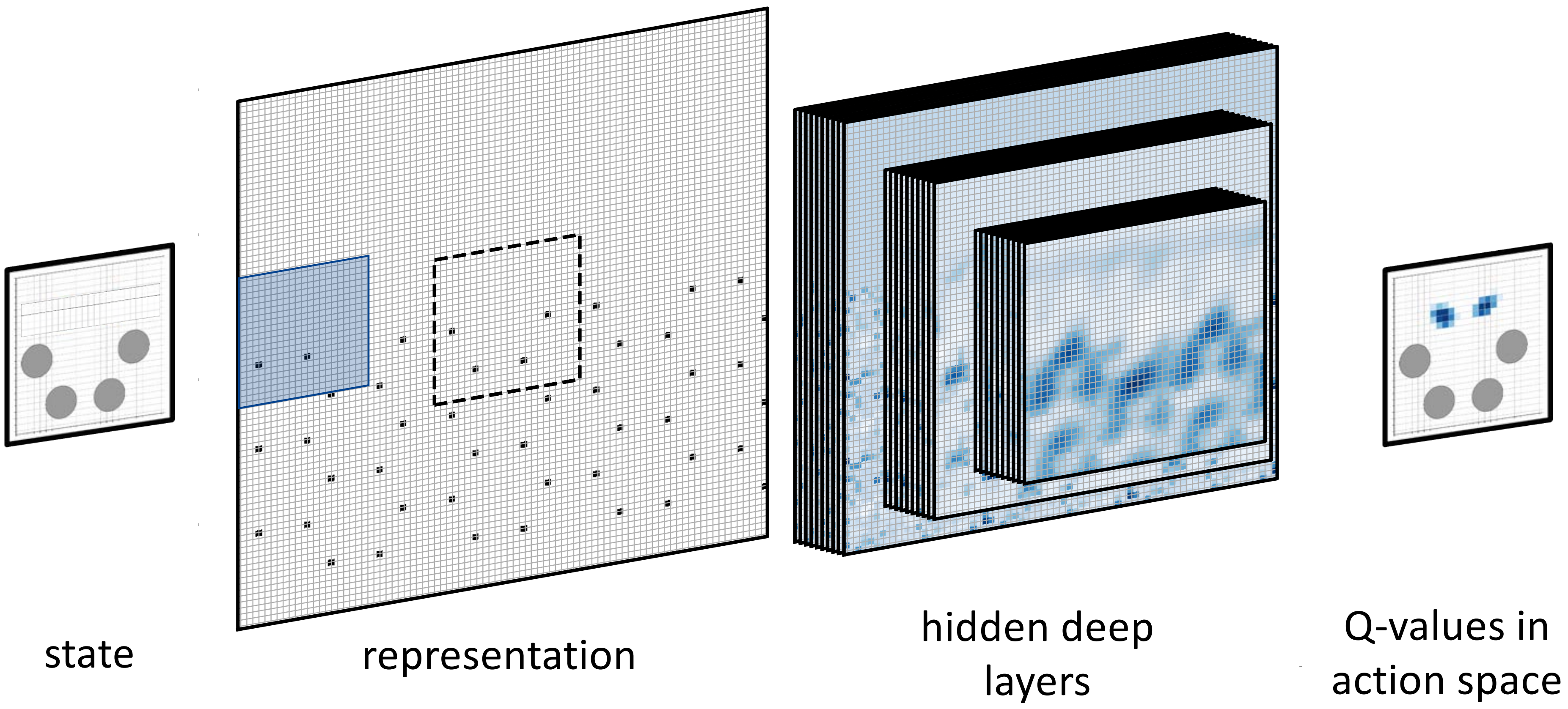


AlphaGo

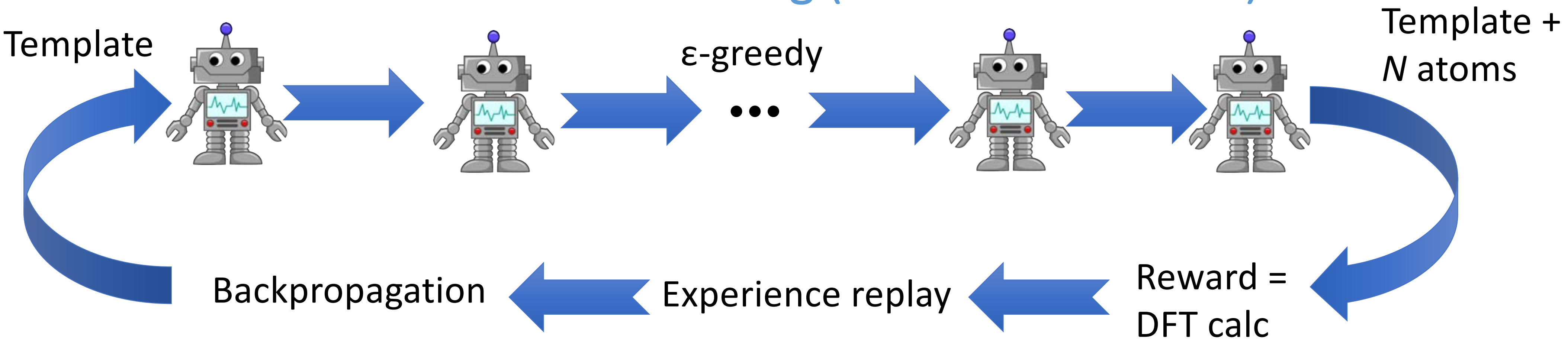
Google DeepMind 2016

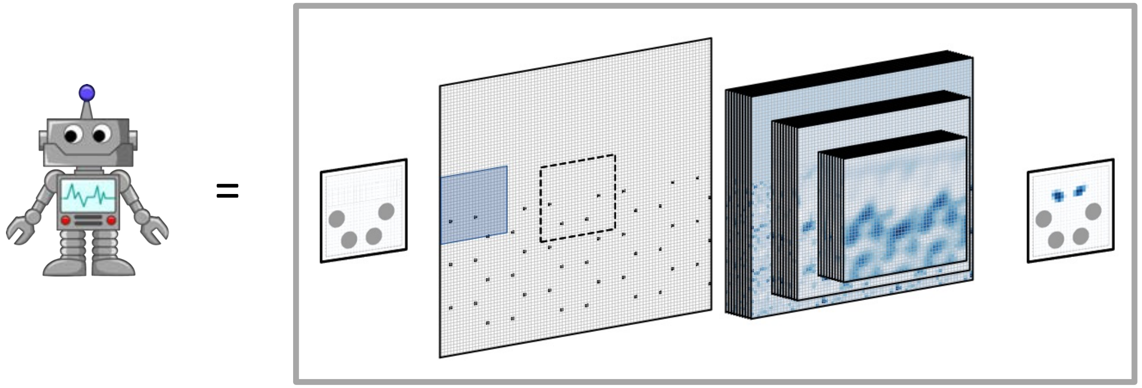
Atomistic structure learning algorithm (ASLA)

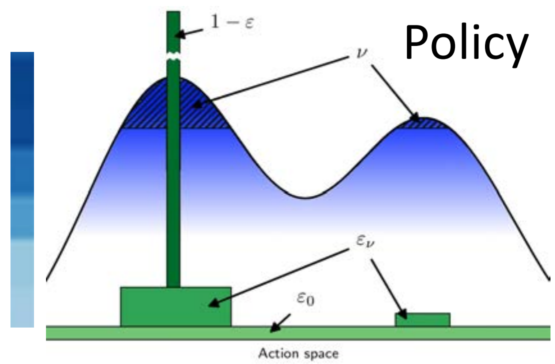
Image recognition: Convolutional Neural Network



Reinforcement learning (self-build + reward)

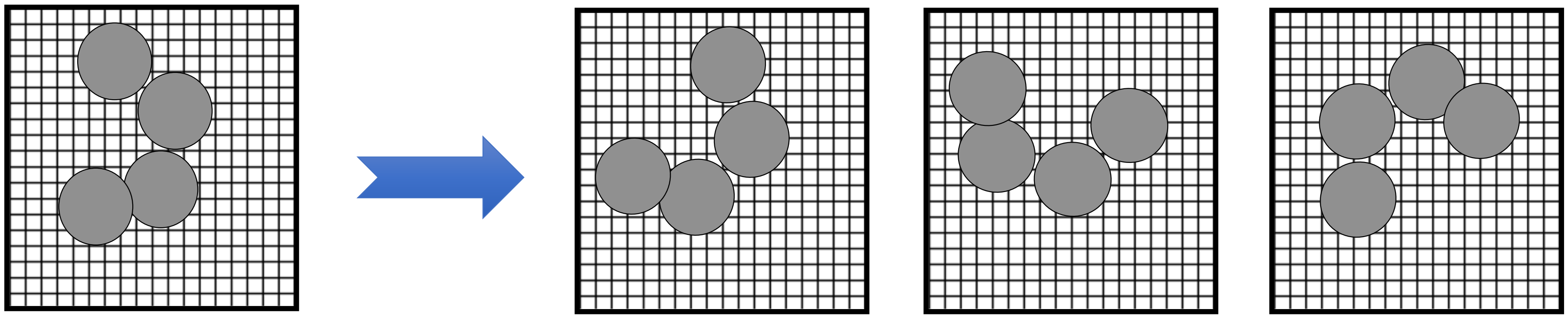






MS Jørgensen, HL Mortensen, SA Meldgaard, EL Kolsbjerg, ... J. Chem. Phys. **151**, 054111 (2019)

Obtain rotational invariance via data augmentation

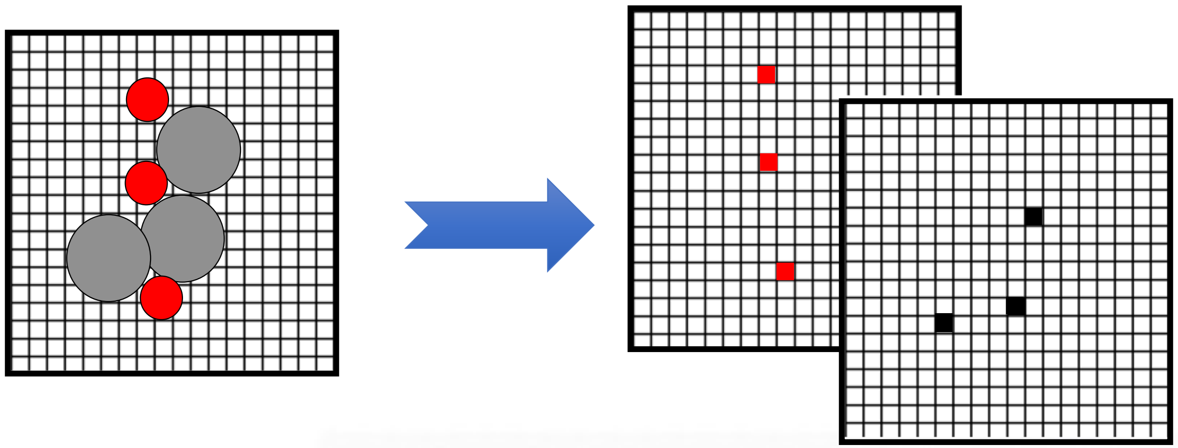


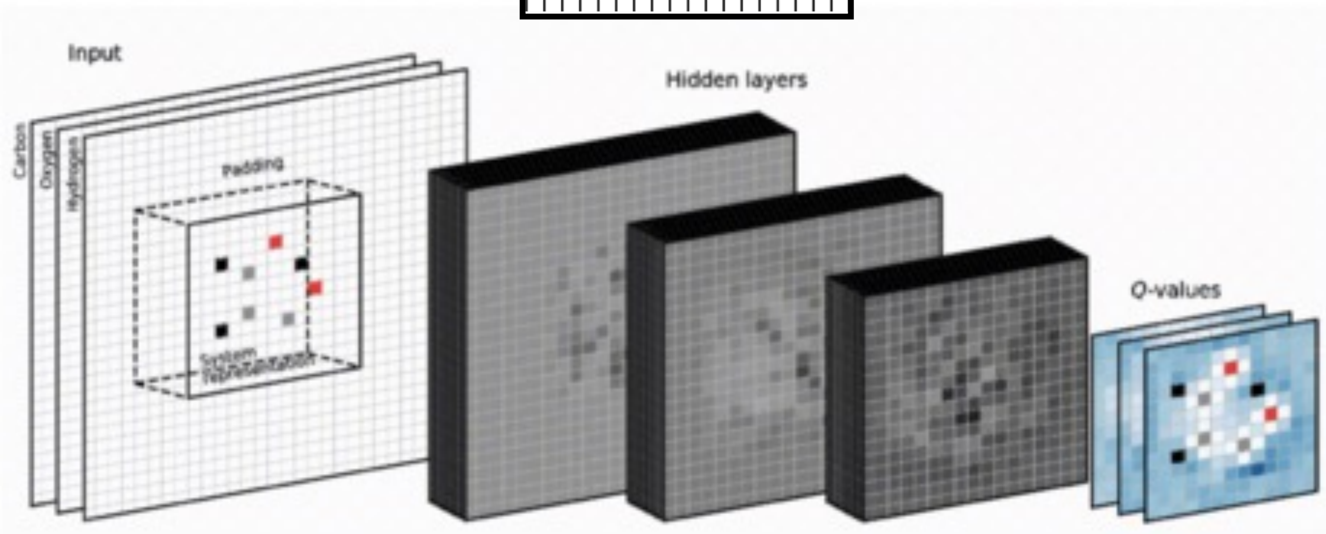
Reward evaluated based on actual DFT calculation

Given same reward

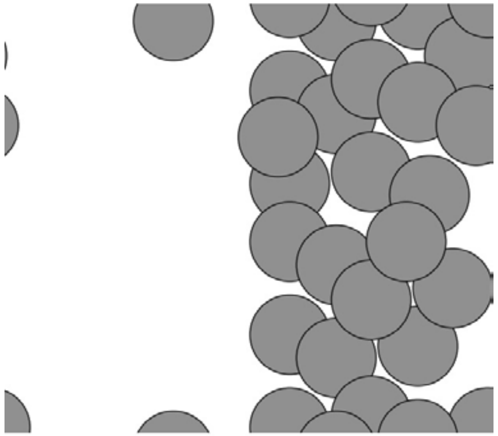
Not strictly true under periodic boundary conditions, but appears to act as regularization

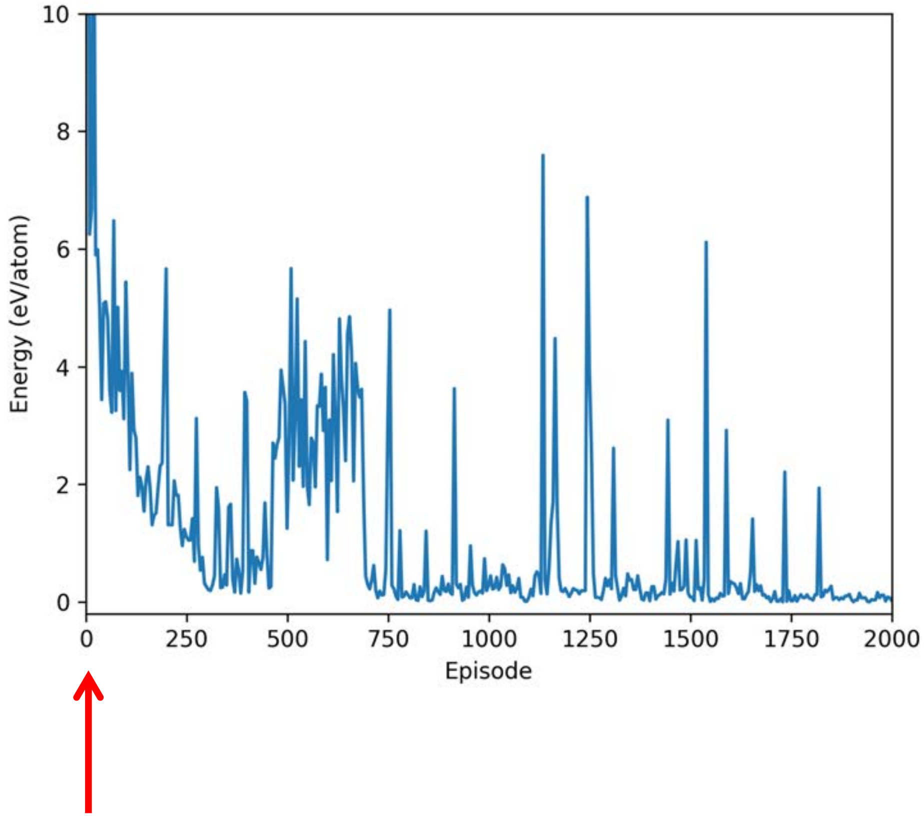
Several atom types: just add image layers – like RGB color images





Search for graphene: untrained agent

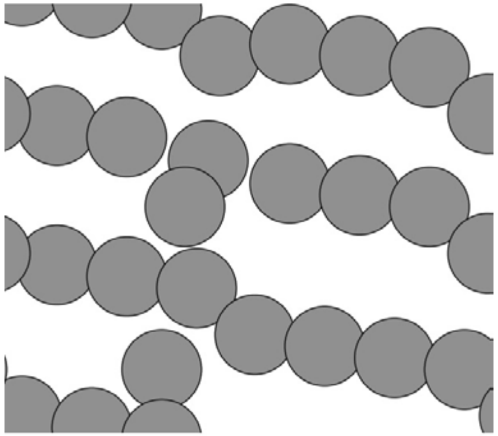


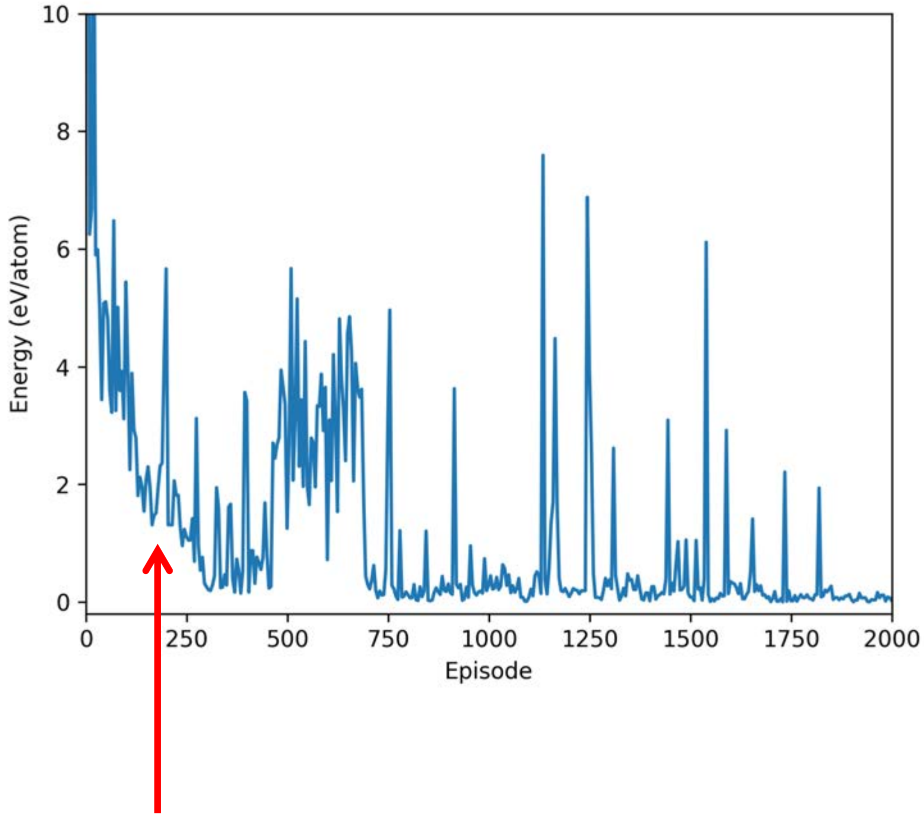


MS Jørgensen, HL Mortensen, SA Meldgaard, EL Kolsbjerg, ...
J. Chem. Phys. **151**, 054111 (2019)

AARHUS UNIVERSITY

Search for graphene: agent trained for 200 episodes

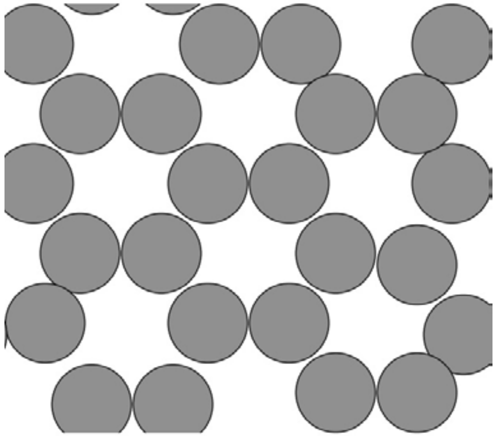


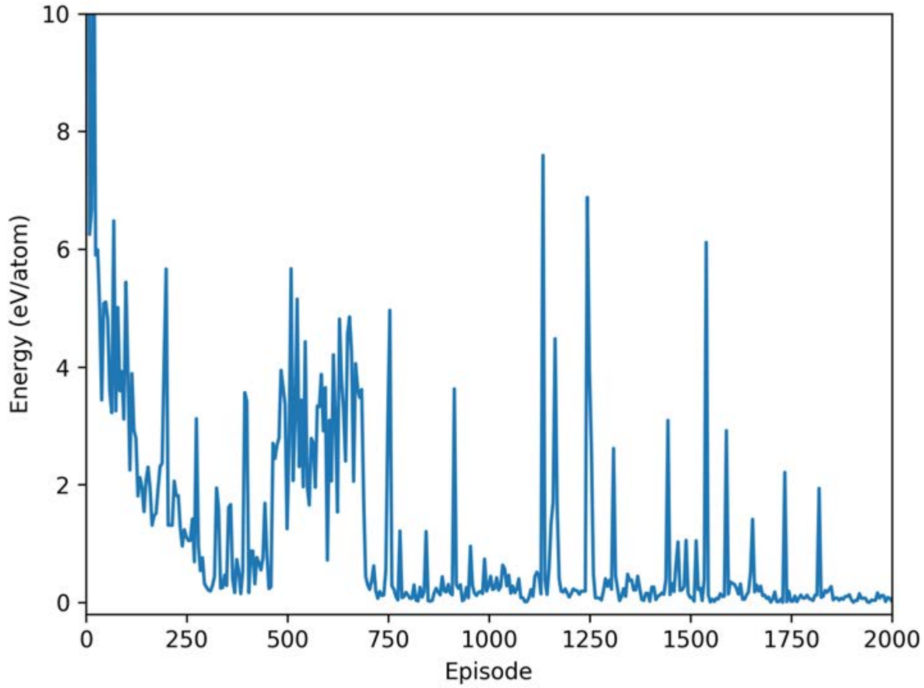


MS Jørgensen, HL Mortensen, SA Meldgaard, EL Kolsbjerg, ...
J. Chem. Phys. **151**, 054111 (2019)

AARHUS UNIVERSITY

Search for graphene: agent trained for 1000 episodes





MS Jørgensen, HL Mortensen, SA Meldgaard, EL Kolsbjerg, ...
J. Chem. Phys. **151**, 054111 (2019)

AARHUS UNIVERSITY

Structure of oxidized Pd(100)

ELSEVIER

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Surface Science 541 (2003) 101–112

SURFACE SCIENCE

www.elsevier.com/locate/susc

The Pd(1 0 0)–($\sqrt{5} \times \sqrt{5}$)R27°-O surface oxide revisited

M. Todorova [a], E. Lundgren [b,*], V. Blum [c], A. Mikkelsen [b], S. Gray [b], J. Gustafson [b], M. Borg [b], J. Rogal [a], K. Reuter [a], J.N. Andersen [b], M. Scheffler [a]

[a] *Fritz-Haber-Institut der Max-Planck-Gesellschaft, Faradayweg 4-6, D-14195 Berlin, Germany*
[b] *Department of Synchrotron Radiation Research, Institute of Physics, Lund University, Box 118, S-221 00 Lund, Sweden*
[c] *National Renewable Energy Laboratory, Golden, CO 80401, USA*

Received 26 March 2003; accepted for publication 23 June 2003

Abstract

Combining high-resolution core-level spectroscopy, scanning tunneling microscopy and density-functional theory calculations we reanalyze the Pd(1 0 0)–($\sqrt{5} \times \sqrt{5}$)R27°-O surface oxide phase. We find that the prevalent structural model, a rumpled PdO(0 0 1) film suggested by previous low energy electron diffraction (LEED) work [Surf. Sci. 494 (2001) L799], is incompatible with all three employed methods. Instead, we suggest the two-dimensional film to consist of a strained PdO(1 0 1) layer on top of Pd(1 0 0). LEED intensity calculations show that this model is compatible with the experimental data of Saidy et al.

Keywords: Scanning tunneling microscopy; Photoemission (total yield); Photoelectron spectroscopy; Low energy electron diffraction (LEED); Density functional calculations; Palladium; Oxygen

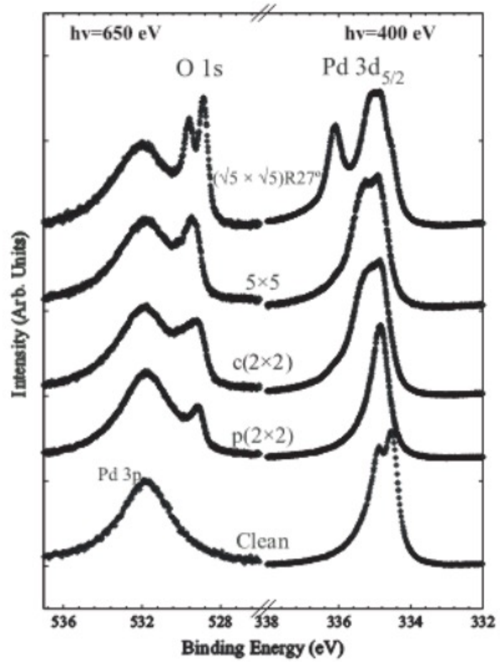
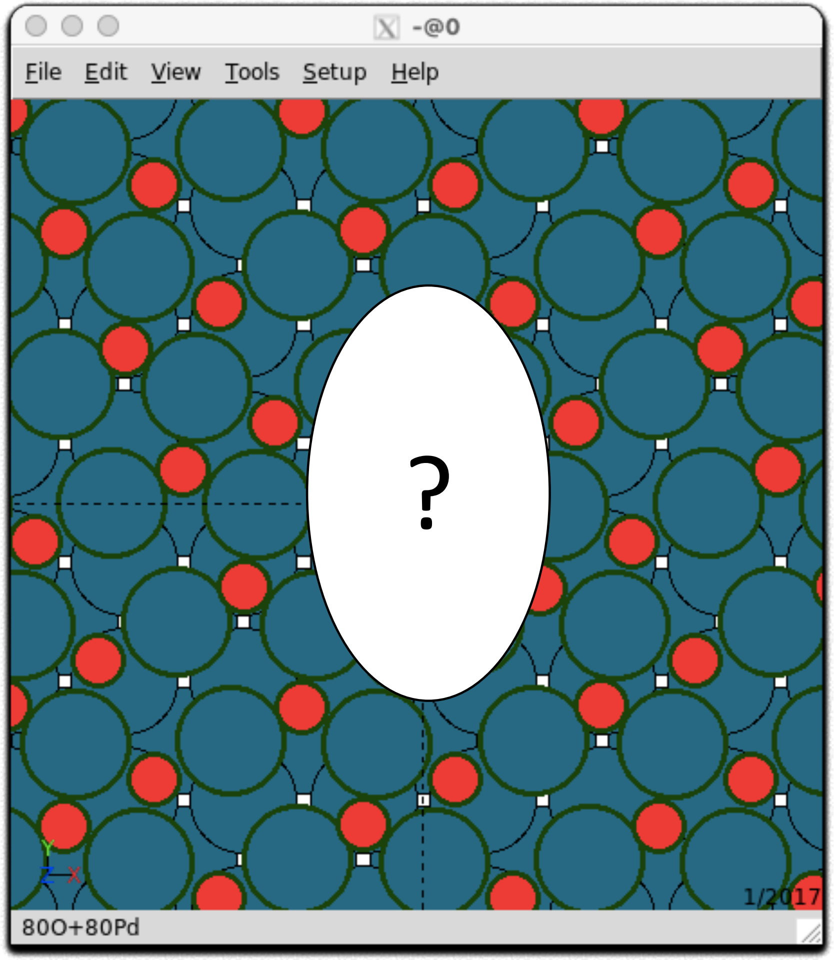


Fig. 1. Experimental HRCL spectra of the O 1s and the Pd $3d_{5/2}$ levels for the sequence of ordered structures that form on Pd(1 0 0) with increasing oxygen coverage. The photon energies were 650 and 400 eV respectively.

Structure of oxidized Pd(100)

Monte-Carlo search with cheap ML energy predictions.



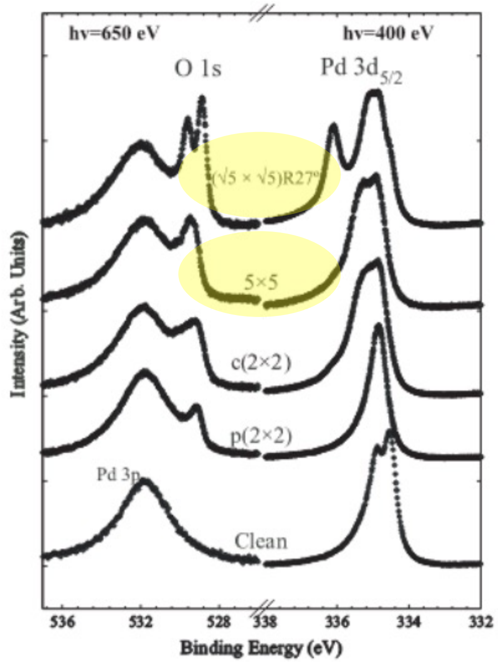
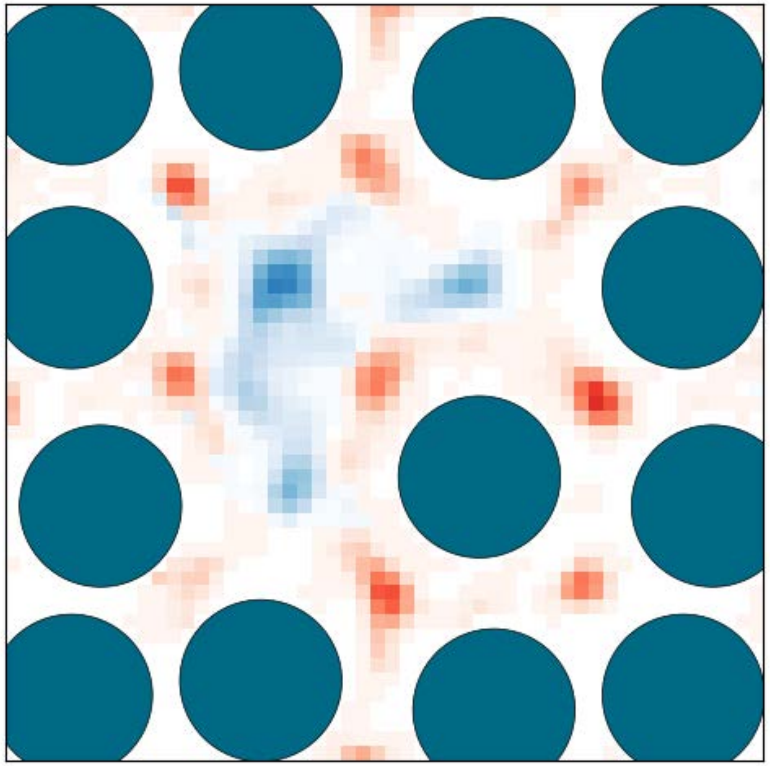
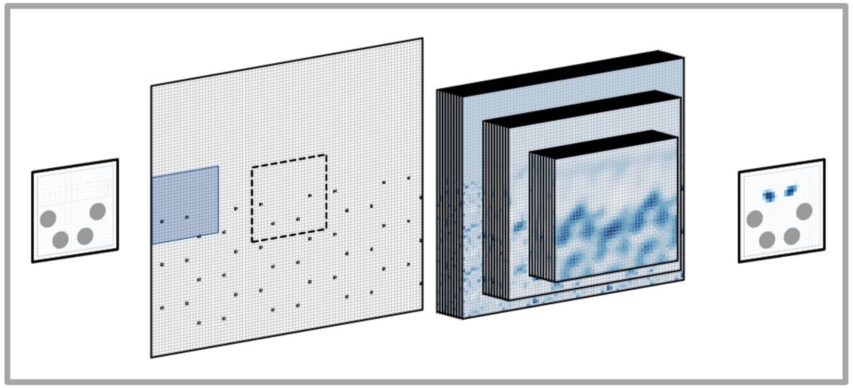


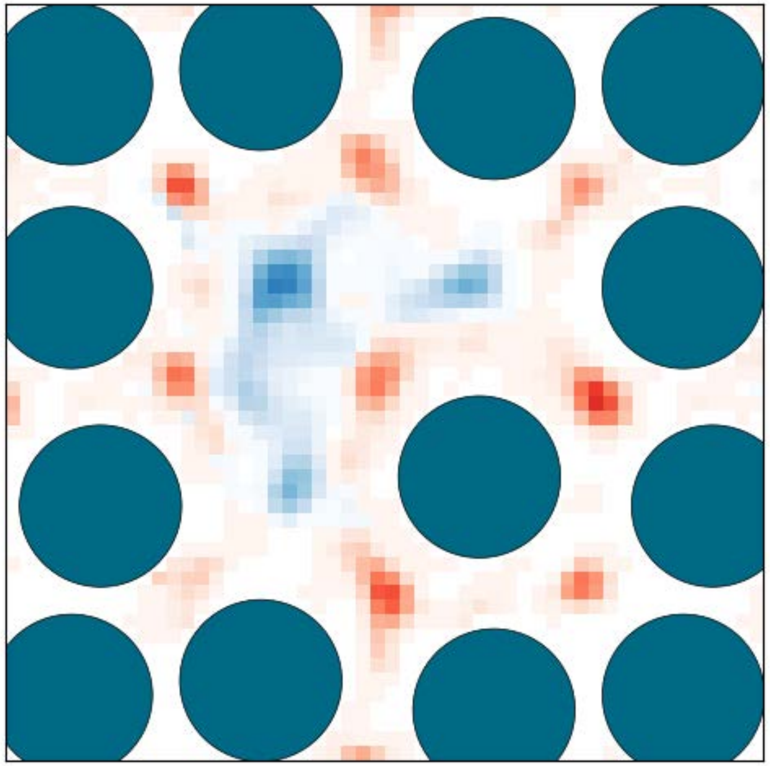
Fig. 1. Experimental HRCL spectra of the O 1s and the Pd $3d_{5/2}$ levels for the sequence of ordered structures that form on Pd(1 0 0) with increasing oxygen coverage. The photon energies were 650 and 400 eV respectively.

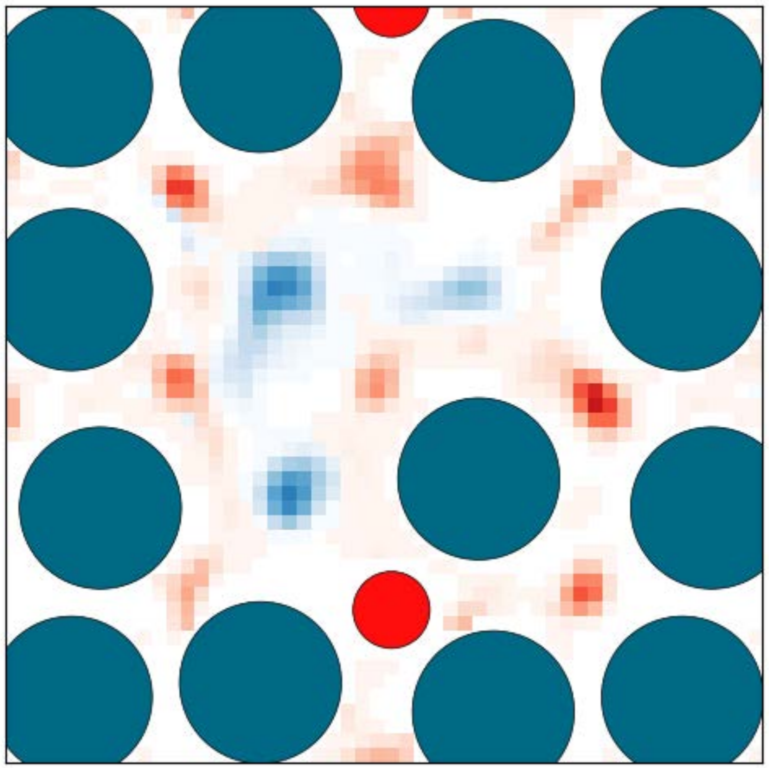
Train the ASLA agent for a (3x3) Pd_9O_6 film



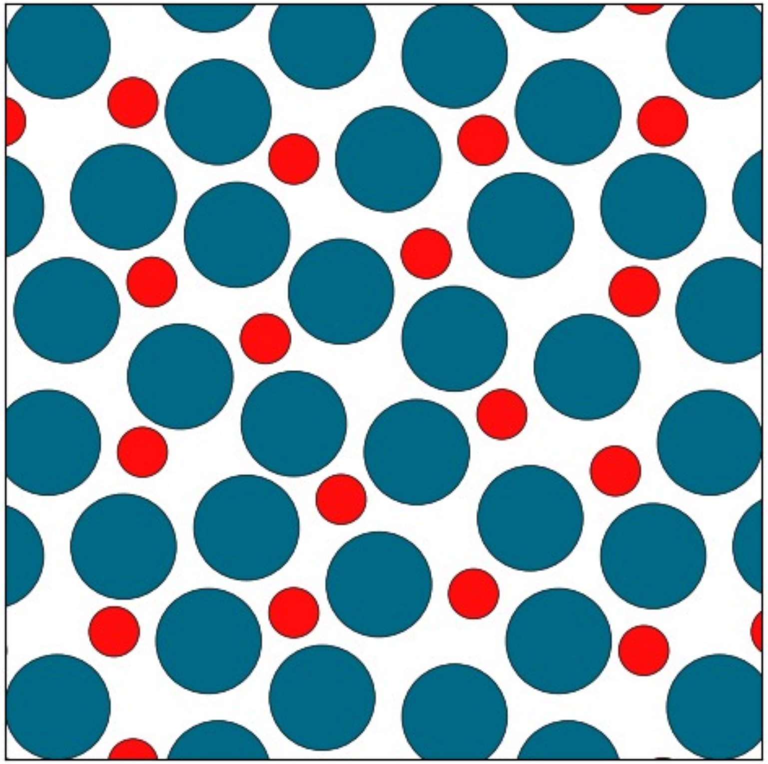


Train the ASLA agent for a (3x3) Pd_9O_6 film





Apply the ASLA agent to the (5x5) problem

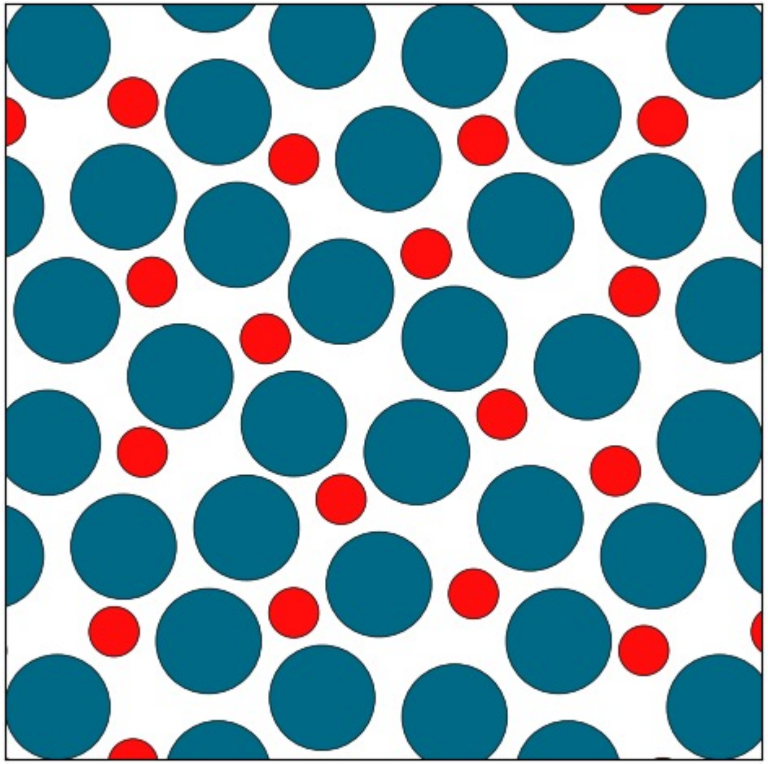


After 1000 supervised episodes on (3x3) problem and 2450 reinforcement episodes on (5x5)

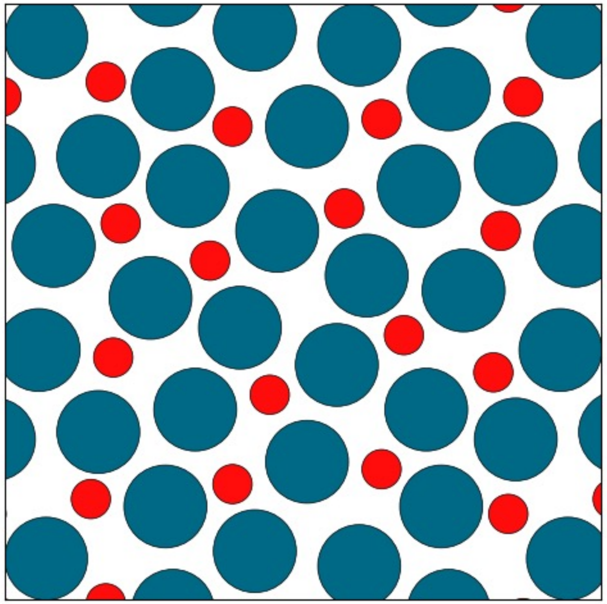
Self-taught:

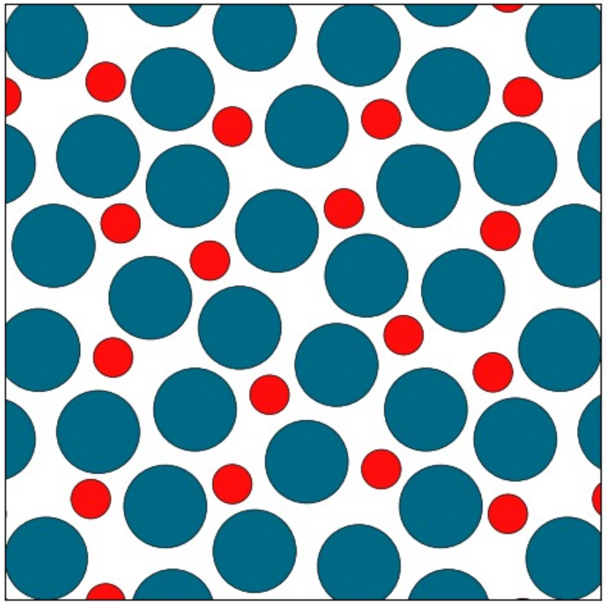
- positions
- atomic types
- sequence

Apply the ASLA agent to the (5x5) problem



Forcing the build with two other sequences:



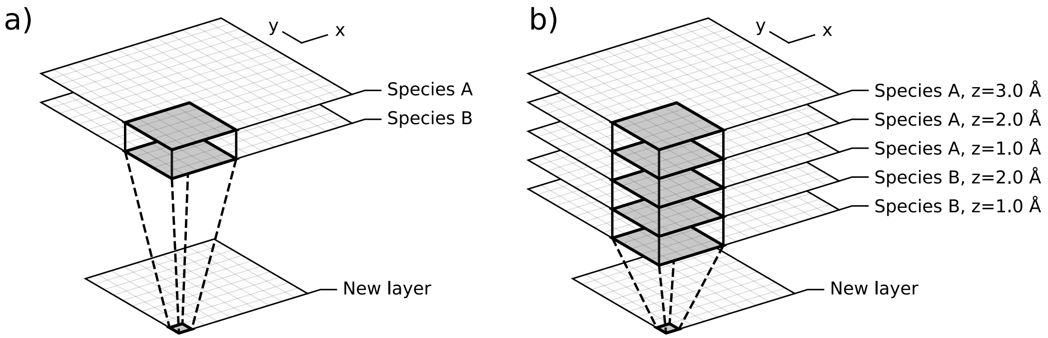


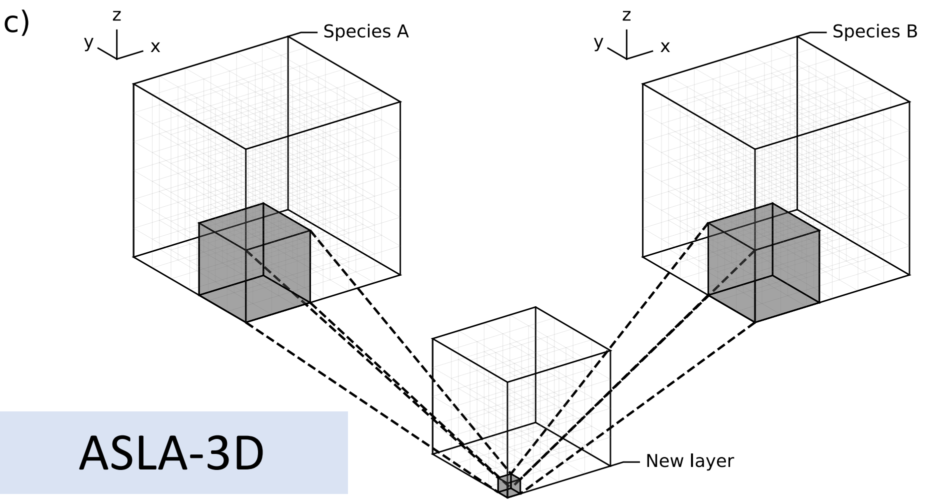
- the agent has obtained general knowledge !

Full 3D ASLA

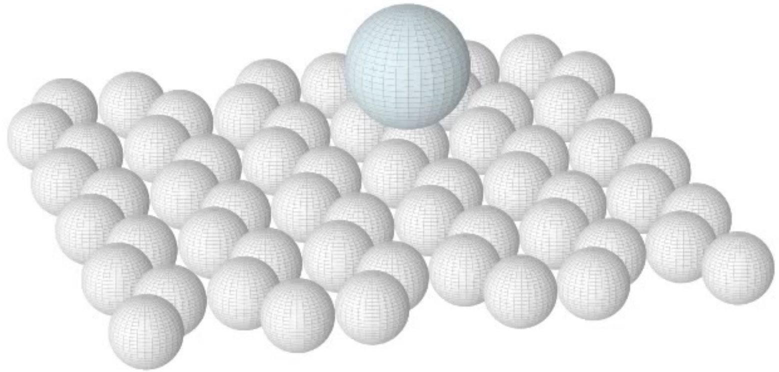
ASLA-2D

ASLA-pseudo-3D





ASLA-3D



Pt_8 on graphene

Conclusion

Machine learning takes:

1) Data
2) Clever representation
3) Model

Provides:

1) Tremendous speedup
2) Intelligent solutions

Collaborators:
Lindsay Merte, Malmø
Edvin Lundgren, Lund
Andrew Peterson, Brown Uni
Karsten W Jacobsen, DTU
Jun Li, Tsinghua
Funding:
Danish National Research Council
VILLUM Foundation

2019: Mortensen, Christiansen, Meldgaard, Bisbo

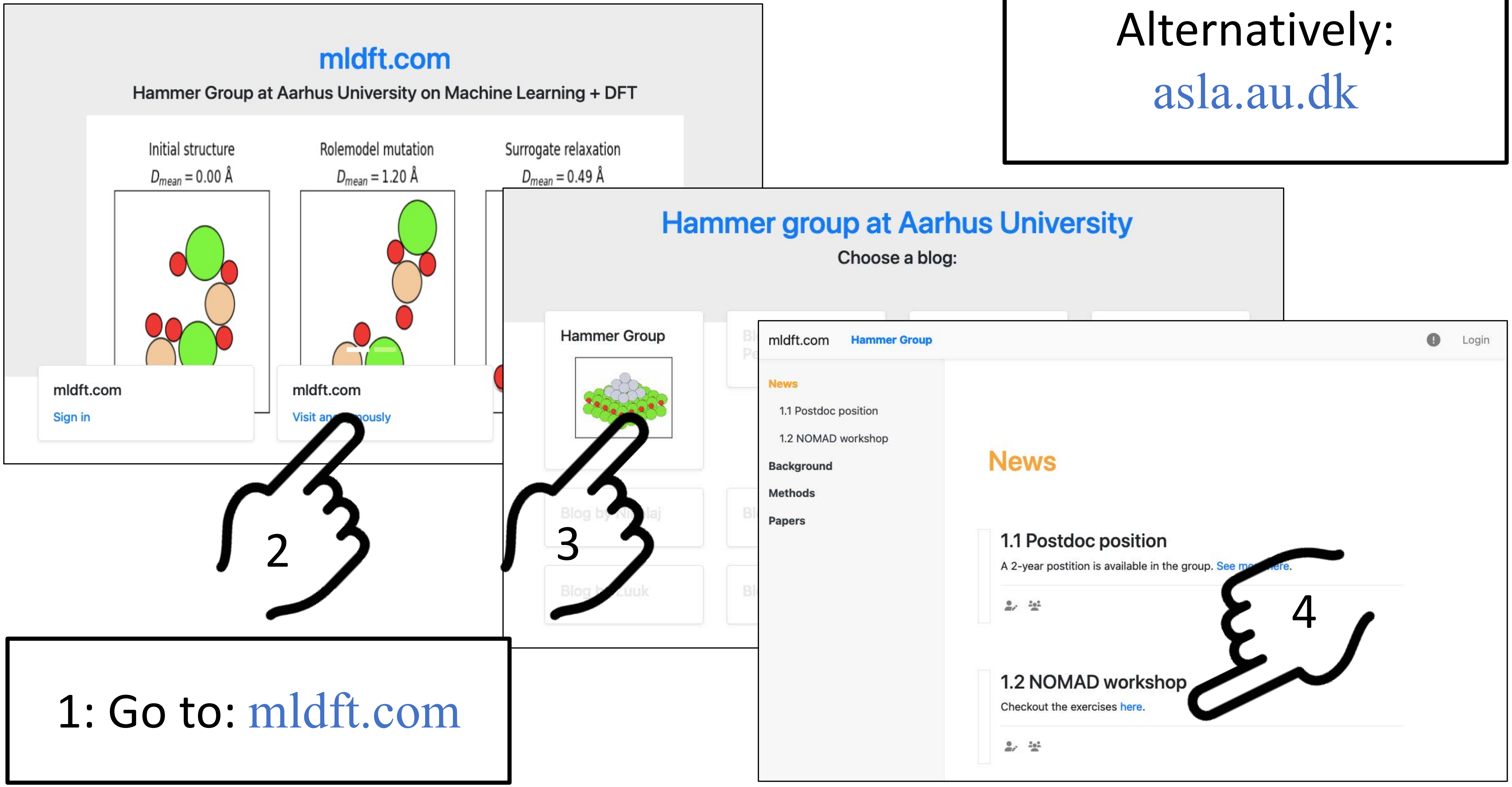
2022: Rønne, Christiansen, Brix

AARHUS UNIVERSITY

INTERCAT
Center for Interstellar Catalysis

VILLUM FONDEN

Alternatively:
asla.au.dk



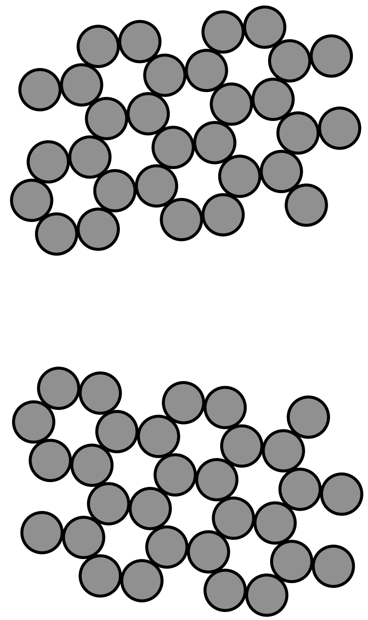
1: Go to: mldft.com

AGOX tutorial

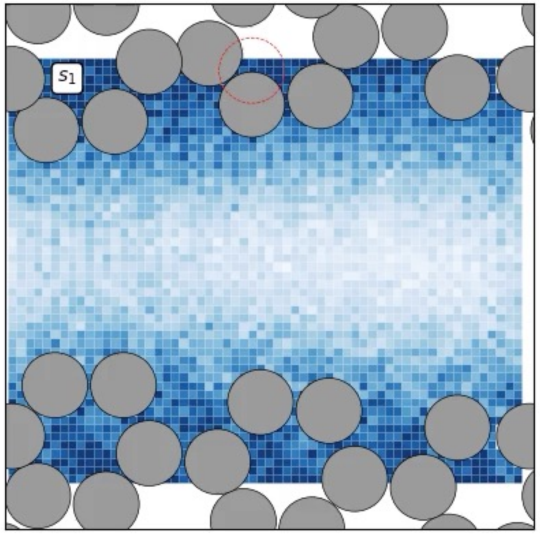
These are the notebooks for the AGOX tutorial / the Hammer presentation:

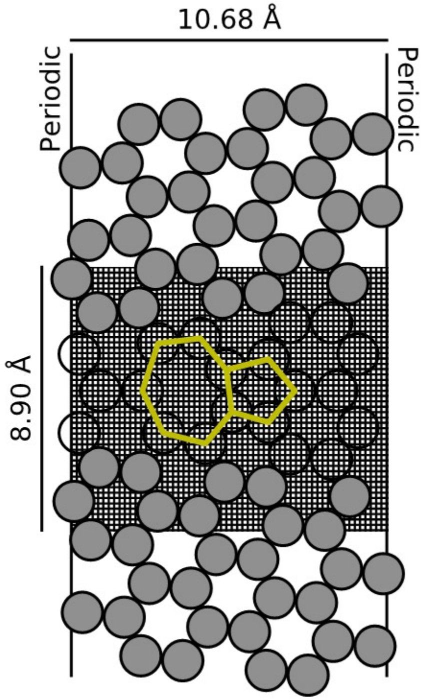
* 01_plot_training_data.ipynb
  - practise basic python operations on dataset
* 02_descriptor.ipynb
  - learn about descriptor
  - learn about k-means
  - learn about dimensionality reduction
* 03_parity_plot.ipynb
  - learn about amount of data needed for model
  - learn about hyperparameters
* 04_relaxation.ipynb
  - see consequences for relaxations for poor model
* 05_basin_hopping.ipynb
  - basin hopping with expensive target potential
  - basin hopping with static cheap model
  - basin hopping with adaptive learning, i.e. iteratively improving model
  - basin hopping with lower confidence bound landscape
* 06_gofee.ipynb
  - example of GOFEE method

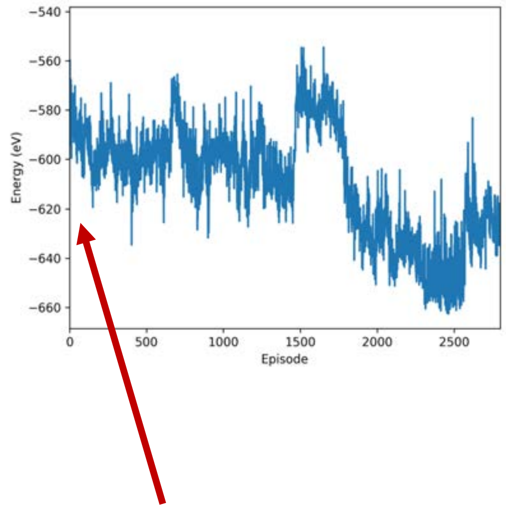
Application to grain boundary problems



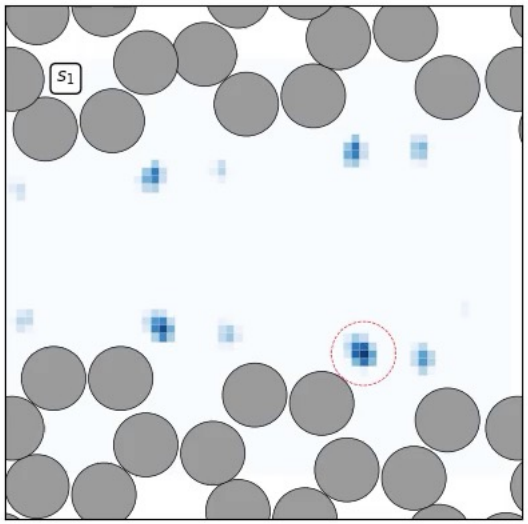
Application to grain boundary problems

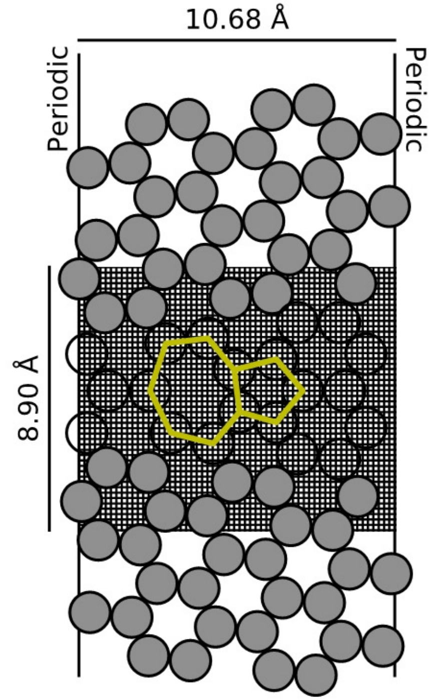


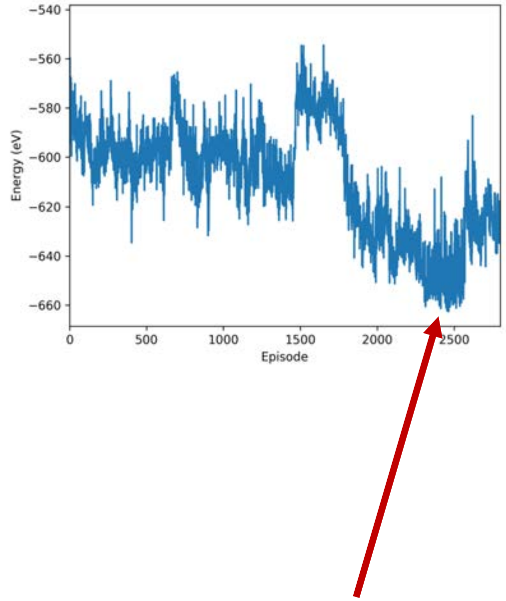




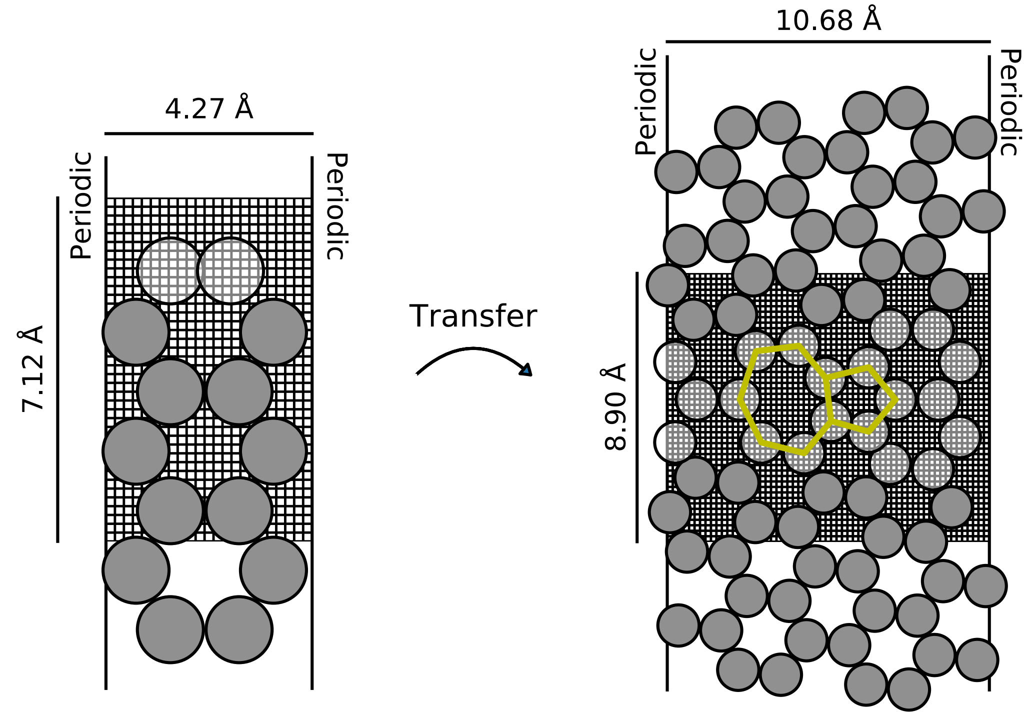
Eventually, the problem is solved



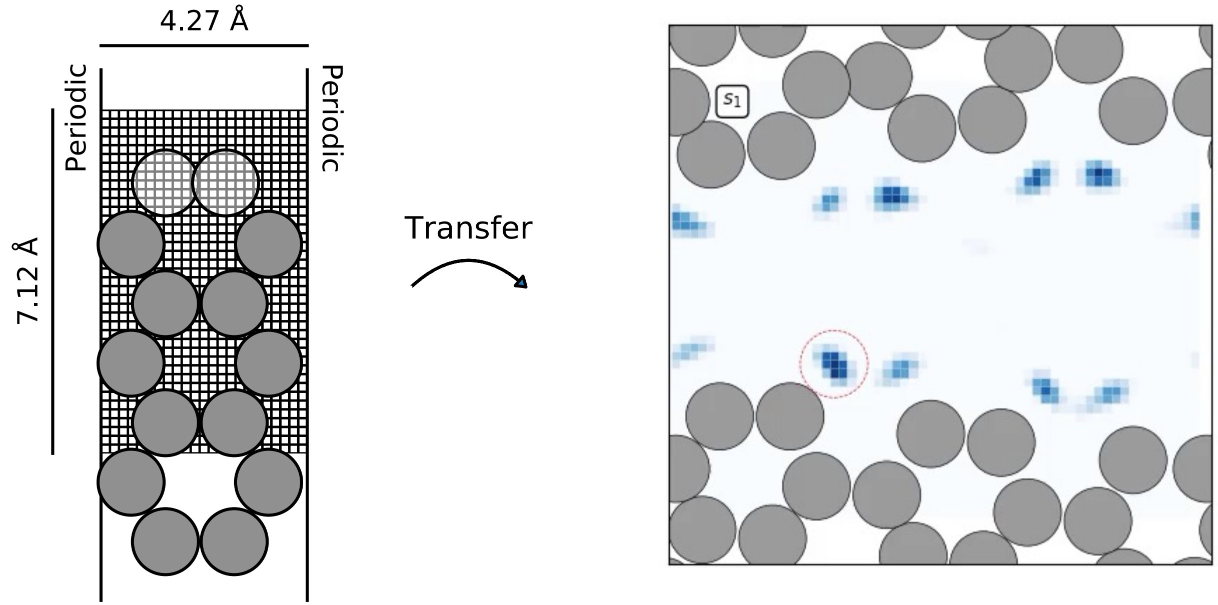




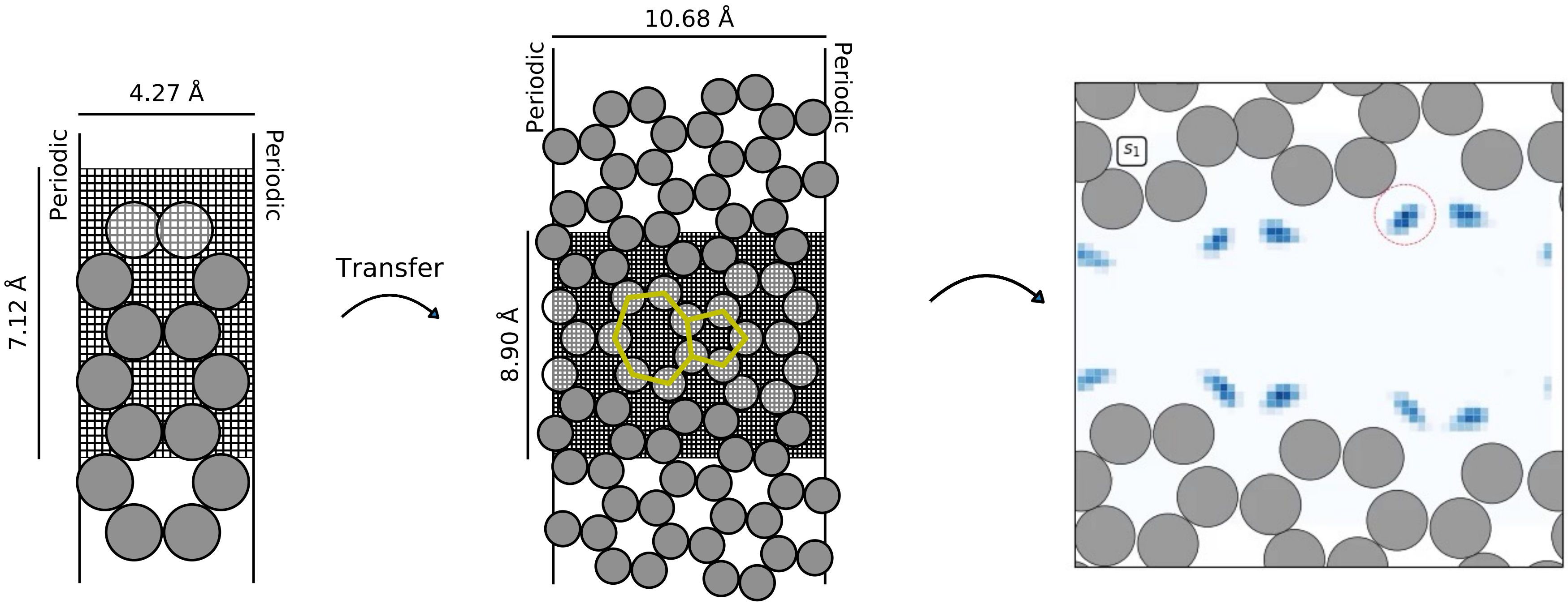
Learn elements from a simpler problem



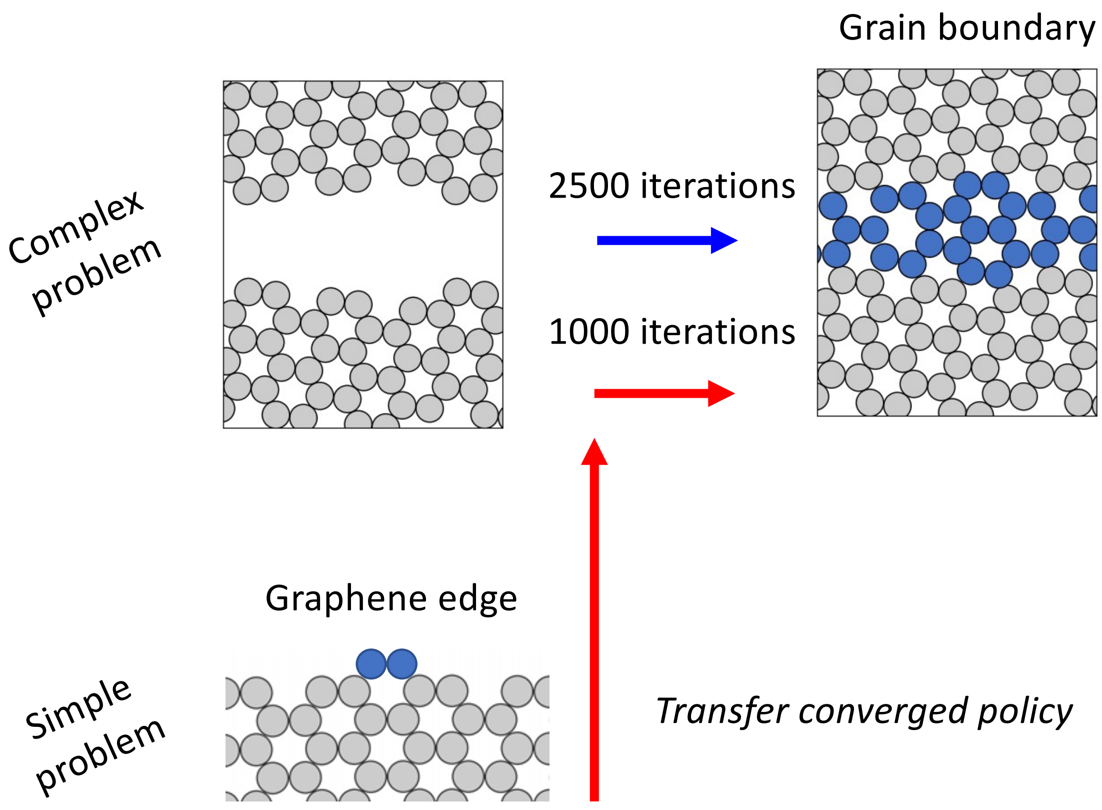
Right after transfer – something was learnt

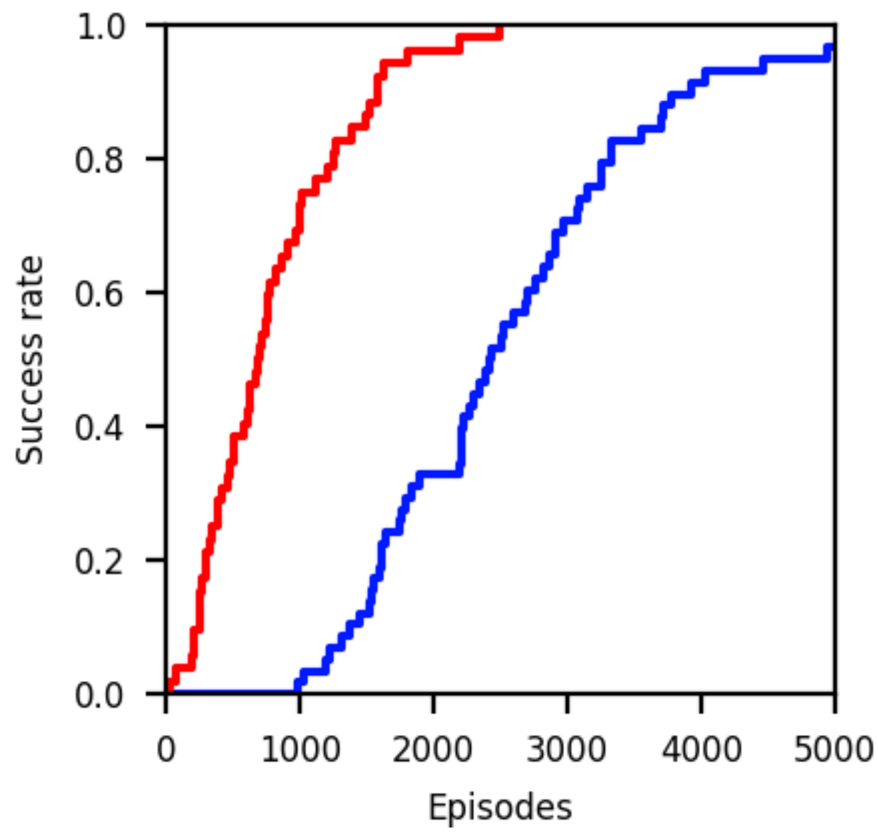


Eventually, the problem is solved

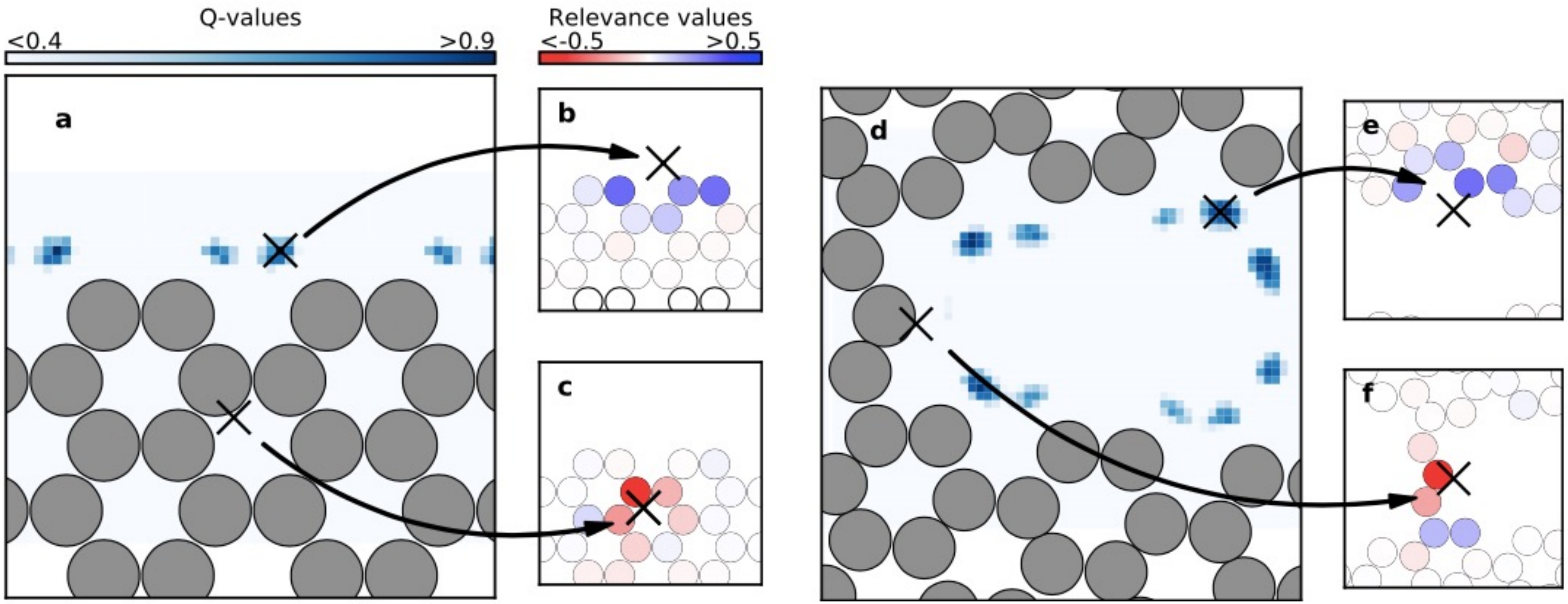


Starting from agent trained on simpler problem





Layer-wise relevance propagation



MS Jørgensen, HL Mortensen, SA Meldgaard, EL Kolsbjerg, ...
J. Chem. Phys. **151**, 054111 (2019)