Physcally-informed artificial neural networks for atomistic modeling of materials

J. Hickman[1], G. P. Purja Pun[2], V. I. Yamakov[3] and Y. Mishin[2]

[1]MML, National Institute of Standards and Technology, Gaithersburg, MD
[2]Department of Physics and Astronomy, George Mason University, Fairfax, VA
[3]National Institute of Aerospace, Hampton, VA

Sponsored by the Office of Naval Research

USACM Workshop:
Recent Advances in the Modeling and Simulation of the Mechanics of Nanoscale Materials

Philadelphia, PA, August 21-23, 2019

CONTENT

- Atomistic simulations with classical interatomic potentials
  - Traditional potentials: strengths and shortcomings
- Machine-learning potentials
  - NN potentials: strengths and shortcomings
- Physically-informed neural network (**PINN**) potentials
  - General idea. Formalism
  - PINN Al and Si potentials
- Conclusions and future work

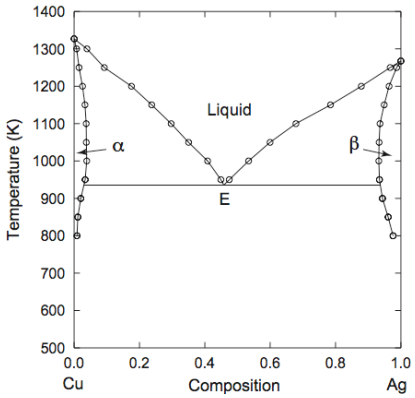
Atomistic simulations of materials

- Explicit treatment of atoms/molecules
  - Molecular dynamics
  - Monte Carlo
- Atomistic (interatomic) potentials (a.k.a. classical force fields)
  - Parameterize configuration space. Express energy and classical forces as functions of atomic coordinates
  - Contain adjustable parameters that are optimized
  - Energy and force calculation is very fast and scales as ~N. Access to large systems (~10^6 atoms, ~10-10^2 ns)
  - Can be complex: angular-dependent, environmentally-dependent, with reactive functionals, etc.

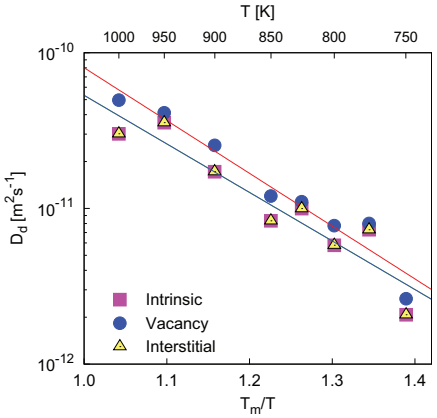
Traditional interatomic potentials

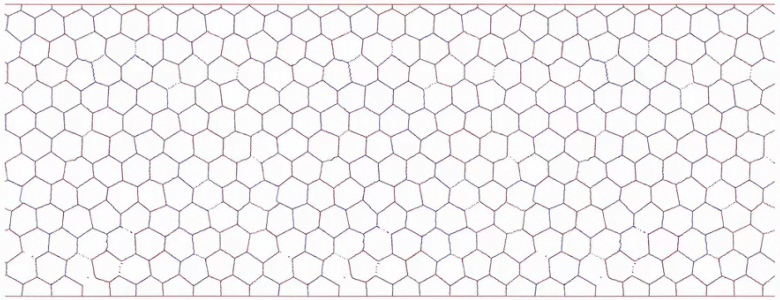
- Specific to material: metals (EAM, MEAM, ADP), covalent (Tersoff, SW), molecular systems and reactions (ReaxFF).
- General-purpose usage:
  - Thermodynamic properties (phase diagrams, interface free energies)
  - Mechanical properties (plastic deformation, fracture)
  - Diffusion kinetics

Cu-Ag phase diagram



Dislocation pipe diffusion in Al

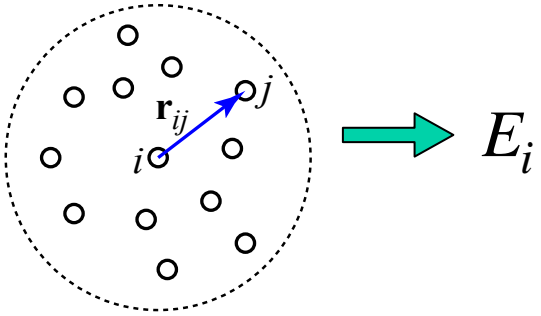




Deformation of nanocrystalline Al

Traditional interatomic potentials

- Partition the total energy: $E = \sum_i E_i$
- Local mapping: $(\mathbf{r}_1, \ldots, \mathbf{r}_n)_i \longmapsto E_i$



- Function:

$$E_i = E_i(\mathbf{r}_1, \ldots, \mathbf{r}_n, \mathbf{p})$$

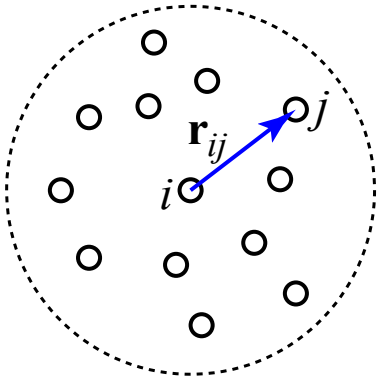
with adjustable parameters $\mathbf{p} = (p_1, p_2, \ldots, p_m)$ (usually, 10-20)

- Fit parameters to a small database of experimental and DFT data
- Direct fit to properties (not just energies): E_0, a_0, c_{ij}, E_v,…, T_m
- Important: the functional form is motivated by physical/chemical intuition

Traditional interatomic potentials: EAM

Embedded atom method (EAM)

- Atomic energy $E_i = \frac{1}{2}\sum_j \phi(r_{ij}) + F(\bar{\rho}_i)$



  - $\phi(r)$ - pair interaction function
  - $F(\bar{\rho})$ - embedding energy
  - $\rho(r)$ - electron density
  - Host electron density: $\bar{\rho}_i = \sum_{k \neq i} \rho(r_{ik})$

- Based on physical assumptions specific to metals (many-body central-force interactions, etc.).
- Derives from DFT or TB.
- The functions are parameterized by analytical expressions of cubic splines. Expected to have physical shapes.

Traditional interatomic potentials: Pros and cons

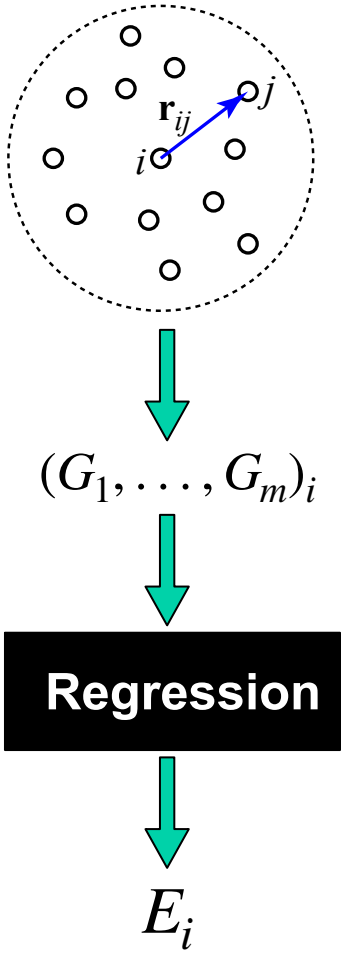
- **Pros:**
  - Very fast. Afford simulations of ~10^6 atoms for ~10^2 ns (or longer with accelerated MD)
  - Based on physics ⇒ reasonable transferability
  - Inaccurate but (usually) not crazy
- **Cons:**
  - Few parameters, small training dataset ⇒ inaccurate
  - Cannot be improved systematically
  - Specific to given class of materials
  - Development is painfully difficult and slow. Heavily relies on human experience. More art than science

Machine-learning interatomic potentials

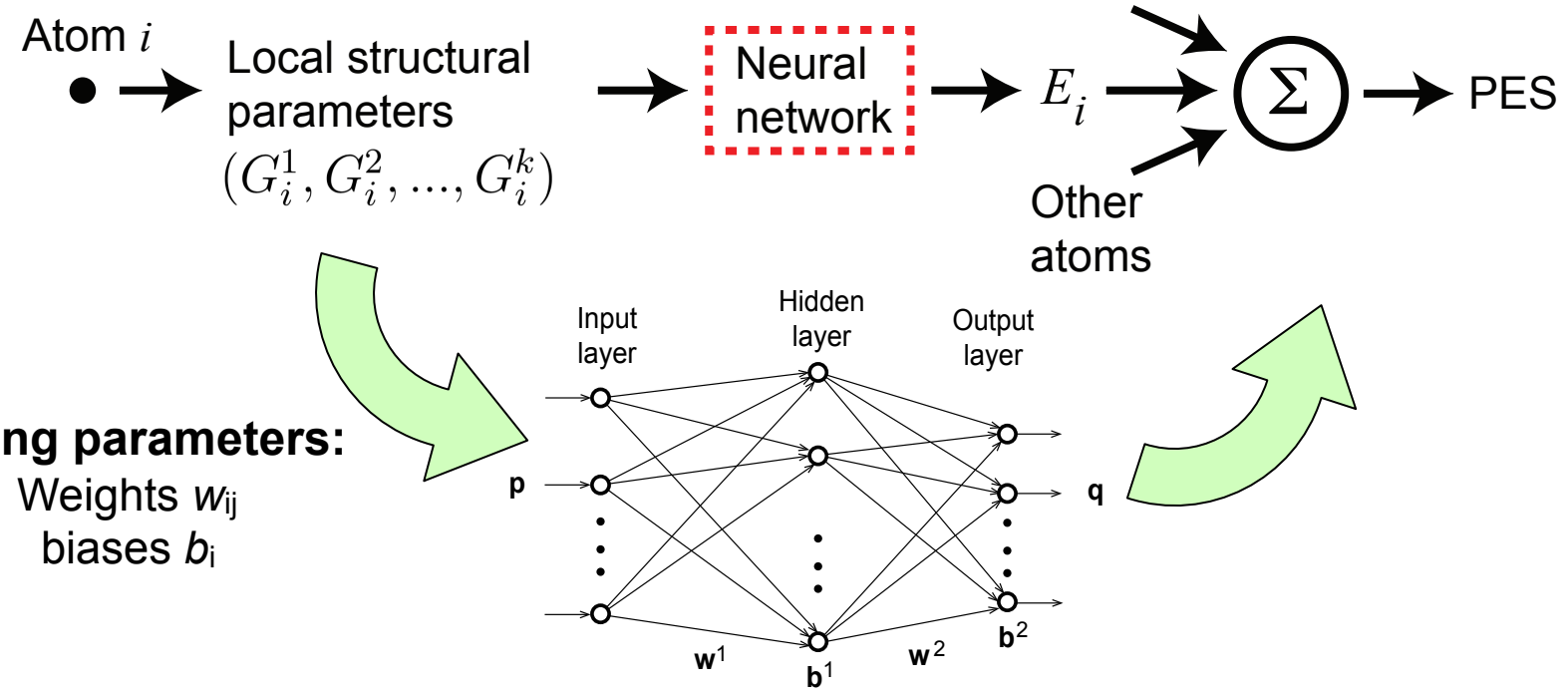
- First introduced by chemists in the 1990s
- Mapping of structure on potential energy surface (**PES**):

$$(\mathbf{r}_1, \ldots, \mathbf{r}_n)_i \longmapsto (G_1, \ldots, G_m)_i \longmapsto E_i$$

- "Fingerprints" $(G_1, \ldots, G_m)_i$ invariant under rotations and translations of coordinates
- Nonlinear regression with ~10^3 parameters to fit
- Training on a large DFT database with 10^3-10^4 supercells
- High accuracy of fit: ~ 1-5 meV/atom (DFT level)
- <u>Purely mathematical interpolation</u> between the training points. No guidance from physics or chemistry



Mathematical NN potentials



Fitting parameters:
Weights w_{ij}
biases b_i

- Example: Behler and Parrinello (PRL 2007): NN potential for Si
- More NN potentials in recent years (metals, semiconductor, ionics)

Machine-learning potentials: Pros and cons

- **Advantages**
  - Extremely accurate (~meV/atom - DFT level)
  - Much faster than DFT. “Accelerated DFT”?
  - No physics ⇒ applicable to any type of bonding, including mixed metallic-covalent
  - Can be improved systematically with more DFT data

- **Drawbacks**
  - Much slower than traditional IPs
  - Require massive DFT calculations
  - Can only *interpolate* between the DFT energies. Cannot *extrapolate* outside the training dataset. Transferability poor and unpredictable. Can be physically meaningless

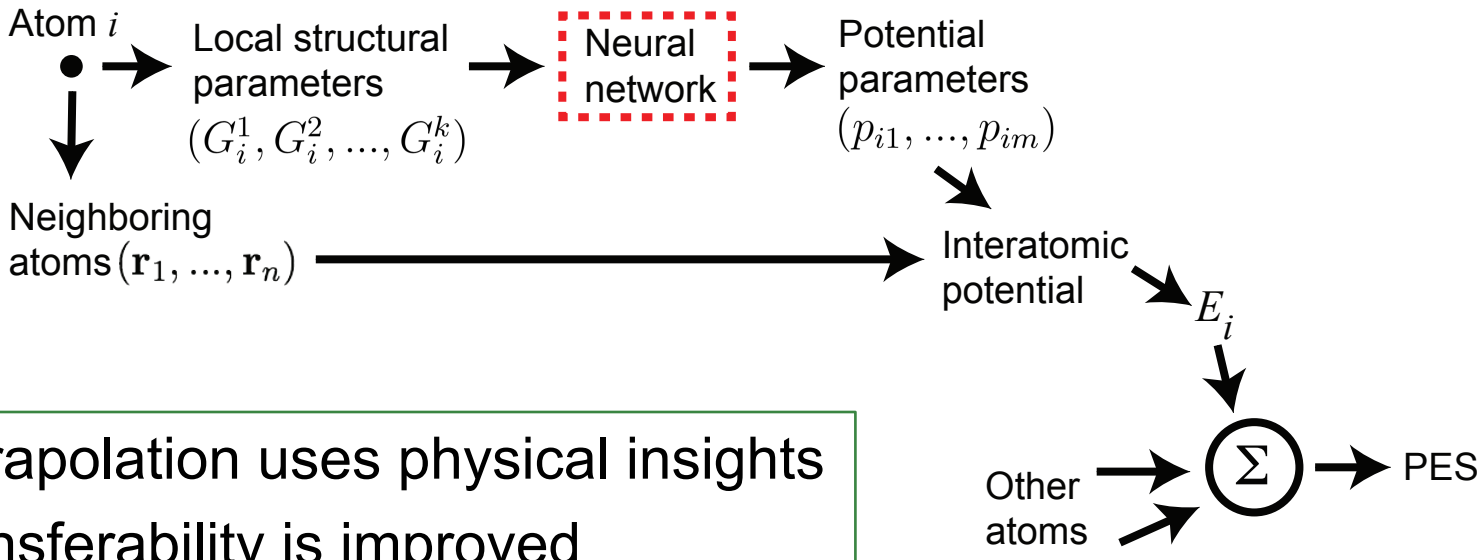
Physically-informed NN (PINN) potentials

Taking the best from both worlds

- Traditional potentials are locally very accurate. Each structure can be fit with different sets of parameters
- Idea: Make parameters functions of local environment

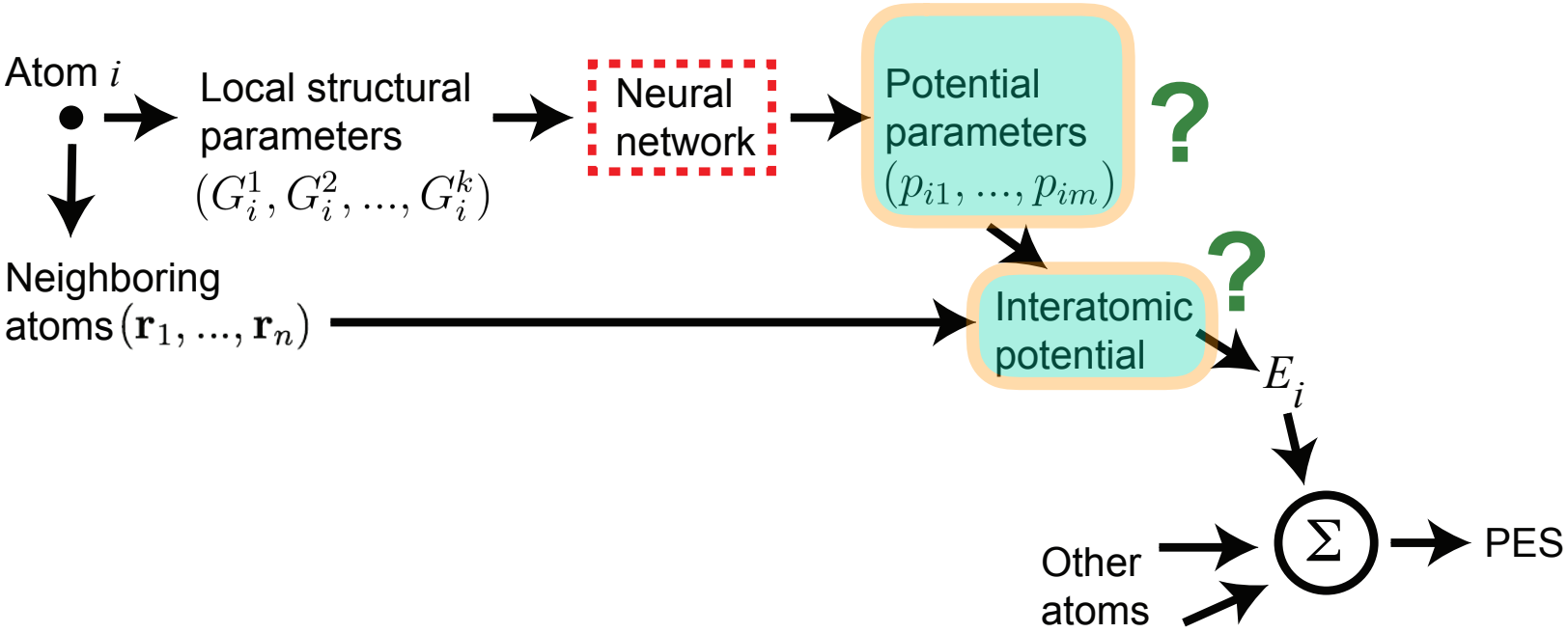
$$E_i = E_i(\mathbf{r}_1, ..., \mathbf{r}_n, \underbrace{p_1, p_2, ..., p_m}_{F_{NN}(\mathbf{r}_1,...,\mathbf{r}_n)})$$

Potential parameters adjusted on the fly



- Extrapolation uses physical insights
- Transferability is improved

PINN potentials: Choice of potential

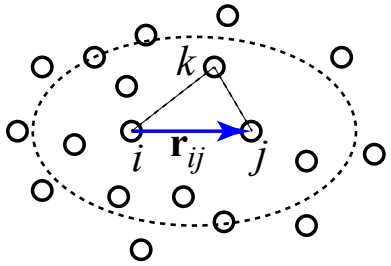


- Requirements for the potential model:
  - Reflect general properties of chemical bonding
  - General enough to include different classes of materials

Analytical bond order potential (BOP)

Total energy: $E = \sum_i E_i$

Atomic energy: $E_i = \frac{1}{2}\sum_{j\neq i}\left[e^{A_i-\alpha_i r_{ij}} - S_{ij}b_{ij}e^{B_i-\beta_i r_{ij}}\right]f_c(r_{ij}) + E_i^{(p)}$



Include 4-5 coordination shells!

Bond screening:

$$S_{ij} = \prod_{k\neq i,j} S_{ijk}$$

Bond order: $b_{ij} = (1 + z_{ij})^{-1/2}$

Coordination number:

$$S_{ijk} = 1 - f_c(r_{ik} + r_{jk} - r_{ij})e^{-\lambda_i(r_{ik}+r_{jk}-r_{ij})}$$

$$z_{ij} = a_i \sum_{k\neq i,j} S_{ik}(\cos\theta_{ijk} + h_i)^2 f_c(r_{ik})$$

Promotion energy: $E_i^{(p)} = -\sigma_i\left(\sum_{j\neq i} S_{ij}b_{ij}f_c(r_{ij})\right)^{1/2}$

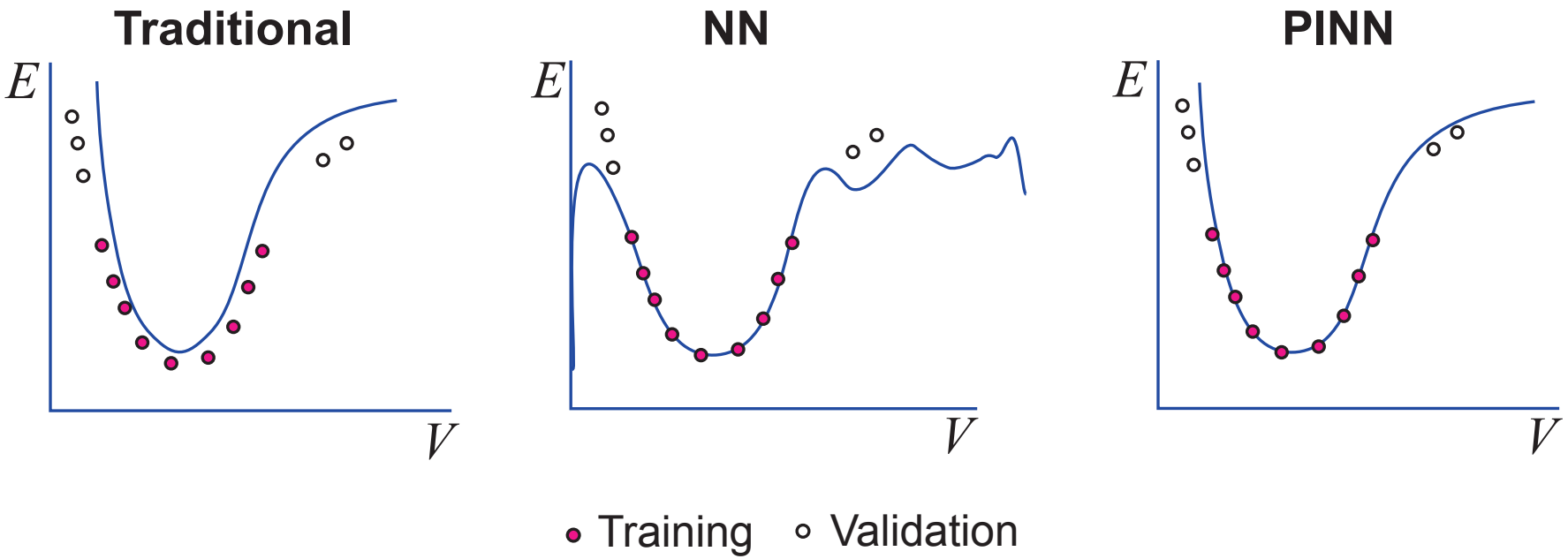
In metals $F(\bar{\rho}_i) = -\sigma_i\left(\bar{\rho}_i\right)^{1/2}$ $\qquad \bar{\rho}_i = \sum_{j\neq i} S_{ij}b_{ij}f_c(r_{ij})$

Analytical bond order potential (BOP)

- **Physical effects**
  - Short-range repulsion
  - Bond order (more neighbors - weaker bond)
  - Bond-angle dependence
  - Bond screening by neighbors

- 8 parameters $A_i, B_i, \alpha_i, \beta_i, a_i, h_i, \sigma_i, \lambda_i$
- Applicable to covalent and metallic materials

Comparison of potentials



- **Inaccurate**
- **Physically meaningful**

- **Accurately trained**
- **Unpredictable extrapolation**

- **Accurately trained**
- **Physically meaningful extrapolation**
- **Interpolation can be also improved**

PINN and NN potentials for Al

- DFT database: 3649 supercells (127592 atoms)
  - EOS for 7 crystal structures under tension/compression
  - Surfaces (100), (110), (111), (311)
  - Vacancies, interstitials
  - Grain boundary, intrinsic stacking fault
  - Clusters with different sizes and shapes
  - AIMD at several pressures and temperatures

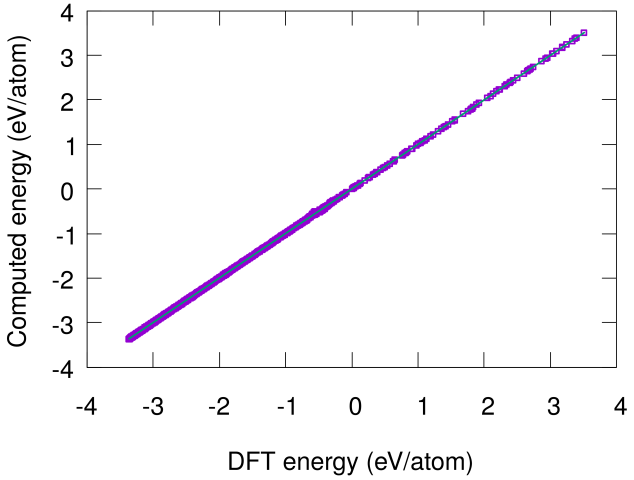
- Network architectures
  - PINN: 60 x 15 x 15 x 8 (1283 parameters)
  - NN: 60 x 16 x 16 x 1 (1265 parameters)

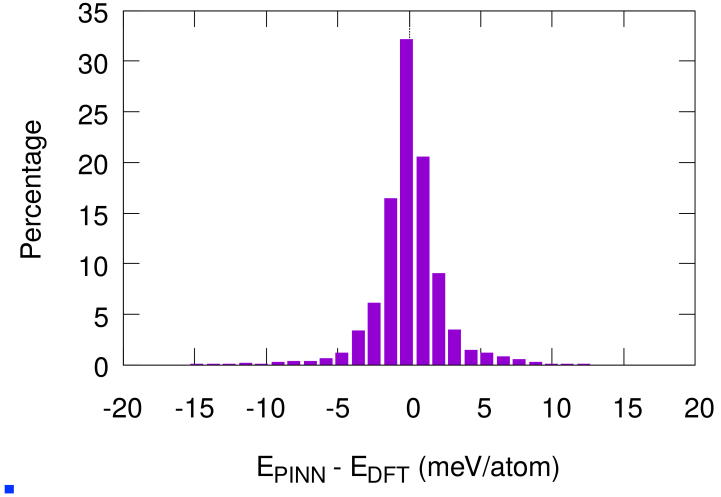
- RMSE of training and validation: ~3.4 meV/atom

PINN errors of training and testing

Training

PINN vs DFT

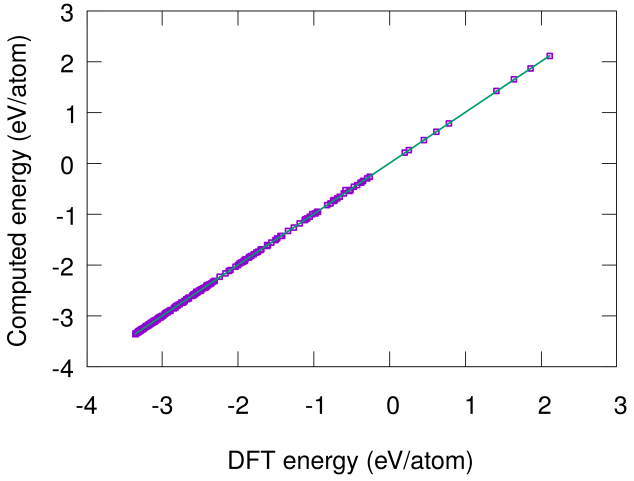


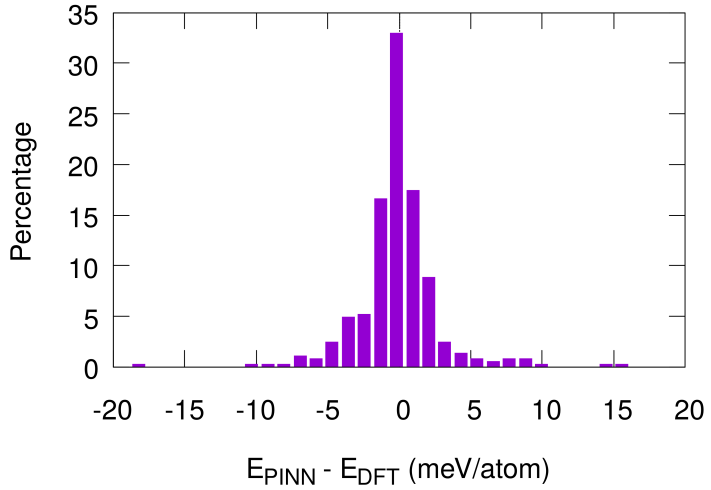


Validation

10-fold cross-validation

PINN vs DFT



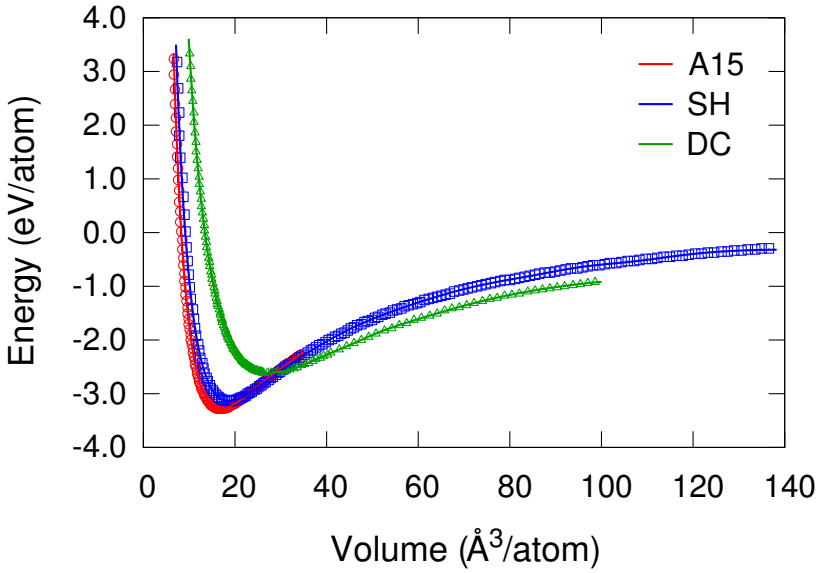


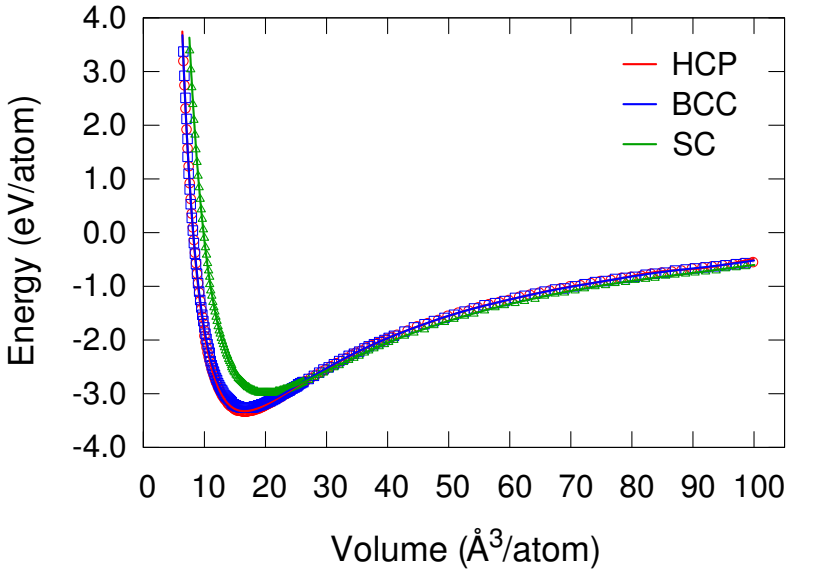
Al properties: PINN/NN versus DFT

Property	DFT	NN	PINN
E_0 (eV/atom)	-3.7480^a	-3.3606	-3.3609
a_0 (Å)	$4.039^{a,d}$; 3.9725–4.0676^c	4.0409	4.0396
B (GPa)	83^a; 81^f	80	79
c_{11} (GPa)	104^a; 103–106^d	108	117
c_{12} (GPa)	73^a; 57–66^d	66	60
c_{44} (GPa)	32^a; 28–33^d	25	32
$\gamma_s(100)$ (Jm^{-2})	0.92^b	0.897	0.899
$\gamma_s(110)$ (Jm^{-2})	0.98^b	0.986	0.952
$\gamma_s(111)$ (Jm^{-2})	0.80^b	0.837	0.819
E_v^f (eV)	0.665–1.346^c; 0.7^e	0.640	0.678
E_v^f (eV) unrelaxed	0.78^e	0.71	0.77
E_v^m (eV)	$0.304 - 0.621^c$	0.627	0.495
E_I^f (T_d) (eV)	2.200–3.294^c	2.683	2.840
E_I^f (O_h) (eV)	2.531–2.948^c	1.600	2.367
E_I^f $\langle 100 \rangle$ (eV)	2.295–2.607^c	1.529	2.246
E_I^f $\langle 110 \rangle$ (eV)	2.543–2.981^c	1.529^*	2.713
E_I^f $\langle 111 \rangle$ (eV)	2.679–3.182^c	2.631	2.815
γ_{SF} ($\mathrm{mJ/m}^2$)	134^i ; 146^g; 158^h	128	121
γ_{us} ($\mathrm{mJ/m}^2$)	162^j ; 169^i ; 175^h	143	132

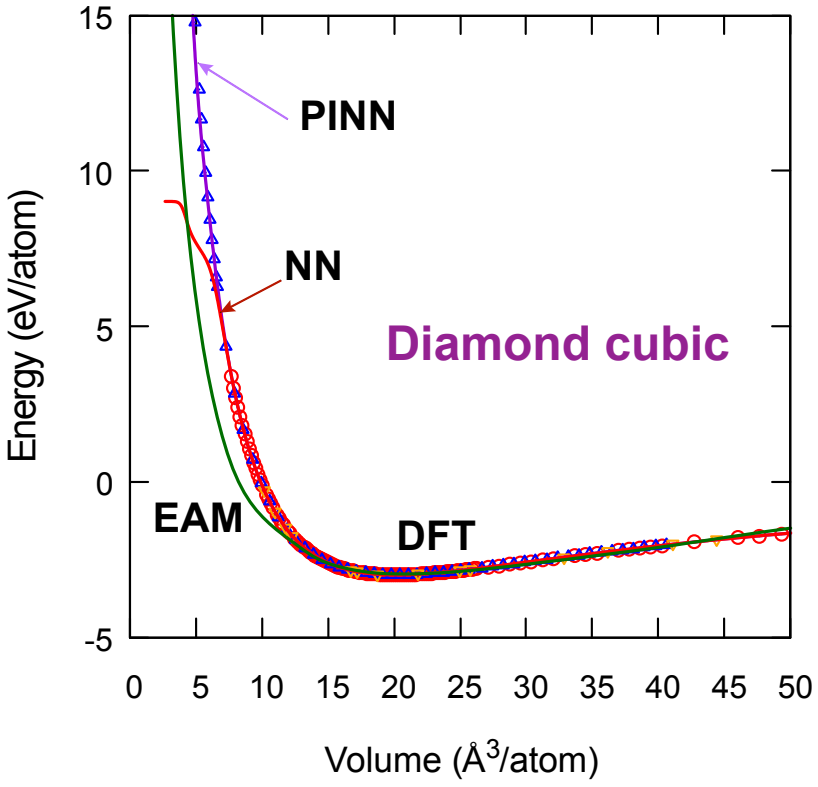
- The properties were not fitted to (only the PES was)
- Agreement with DFT
- PINN performs better than NN

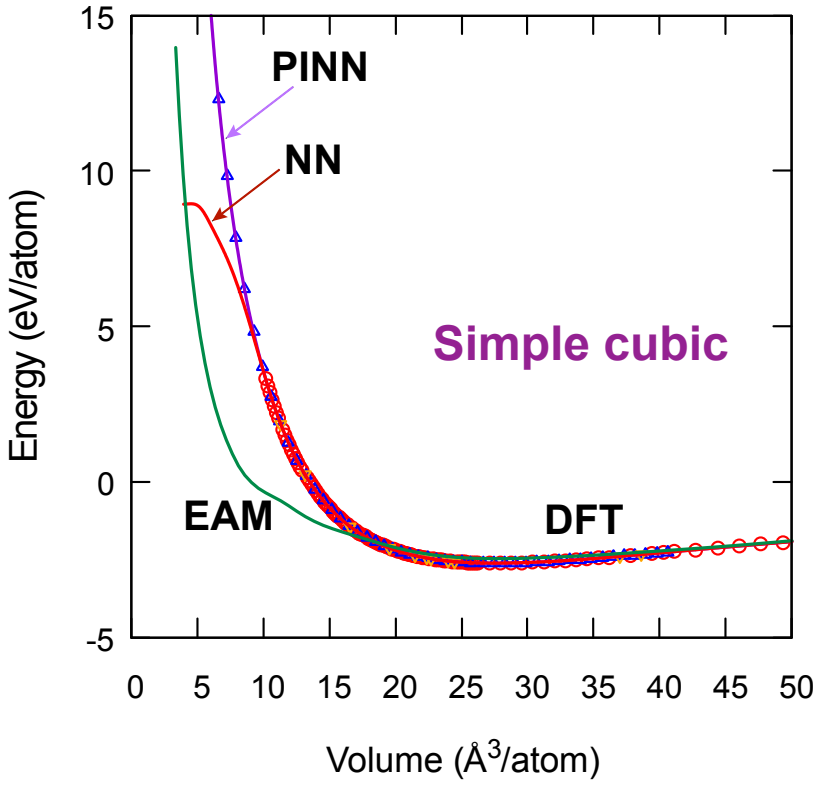
EOS of alternate crystal structures





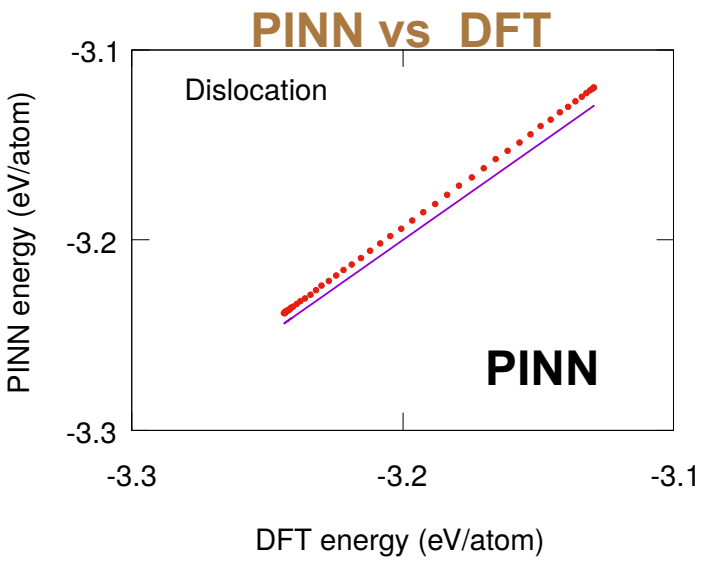
EOS of alternate crystal structures



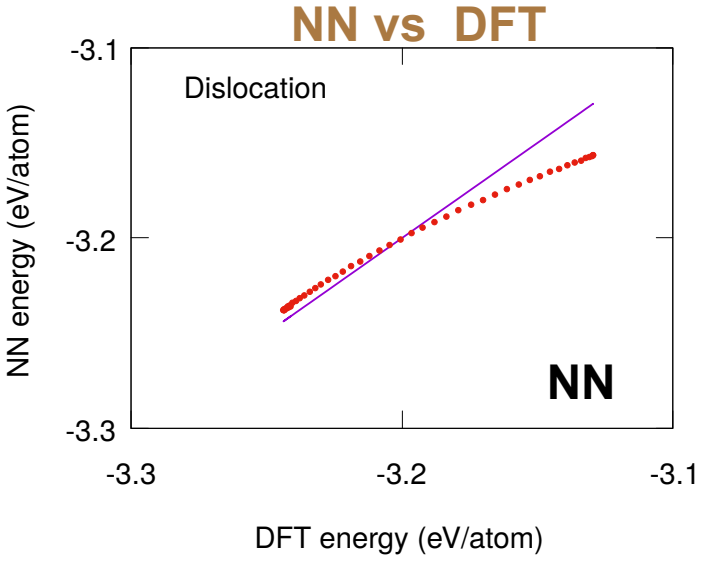


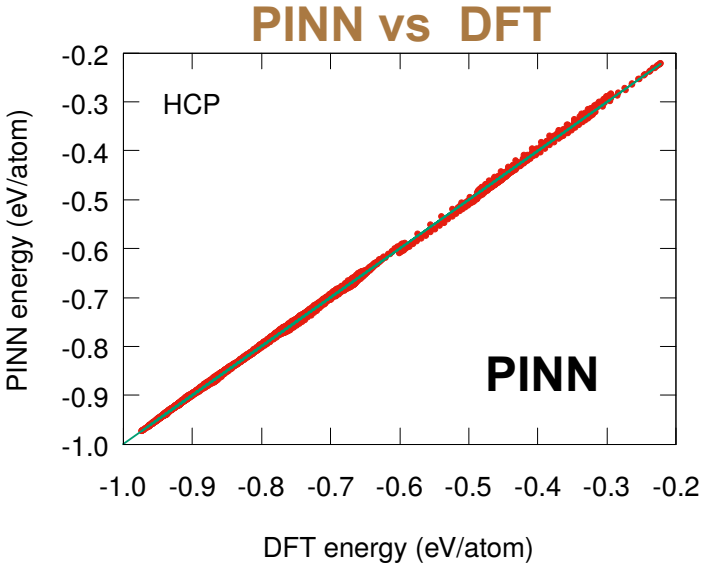
- **NN and PINN are accurately trained to DFT energies**
- **PINN remains accurate in the extrapolation domain**
- **NN extrapolation is unphysical**
- **EAM is less accurate but physically meaningful**

Tests of transferability

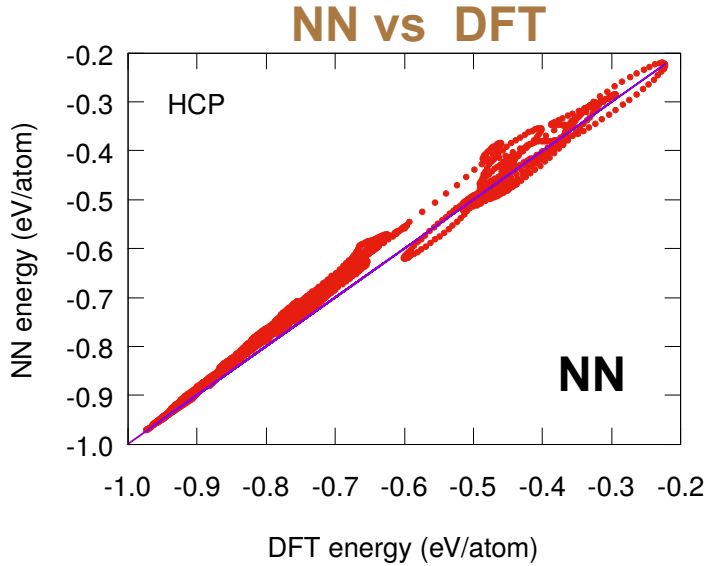


Al dislocation 700 K



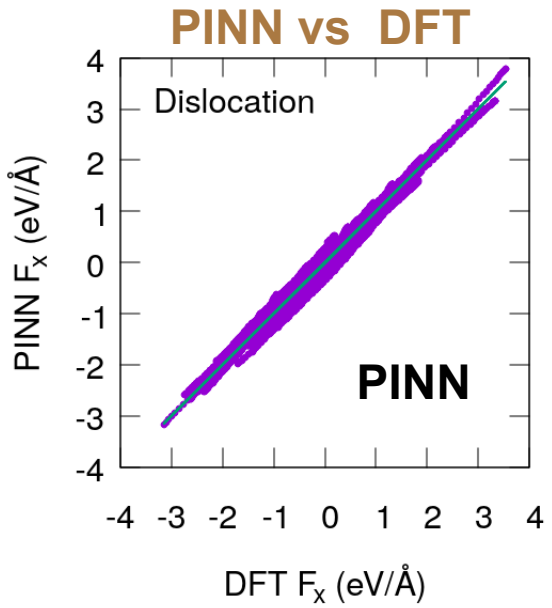


HCP Al 1000-4000 K

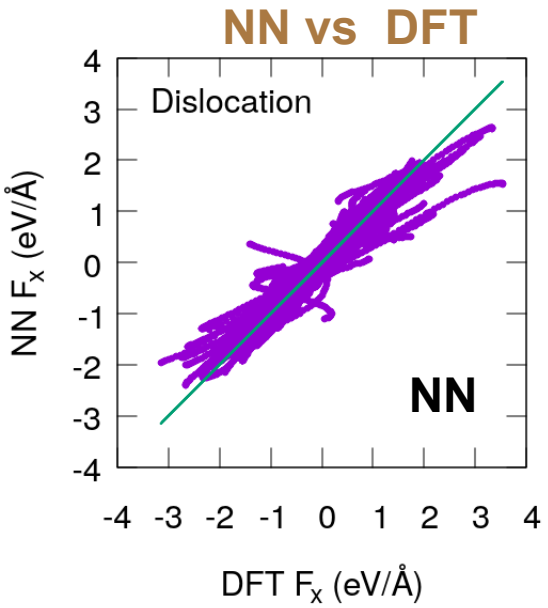


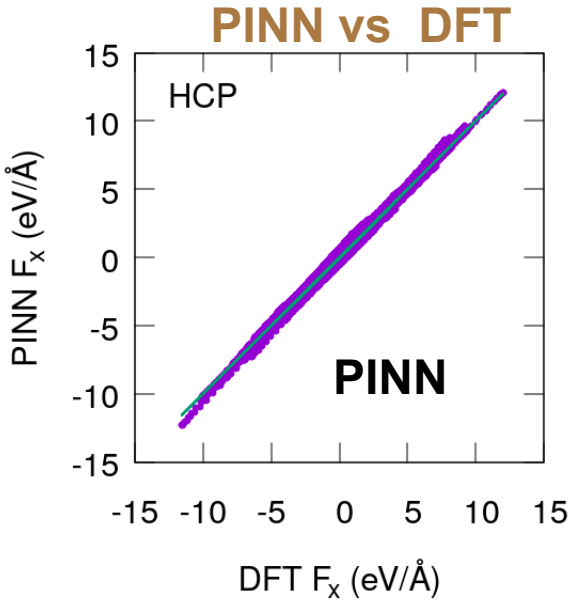
… and many similar tests

Tests of transferability of forces

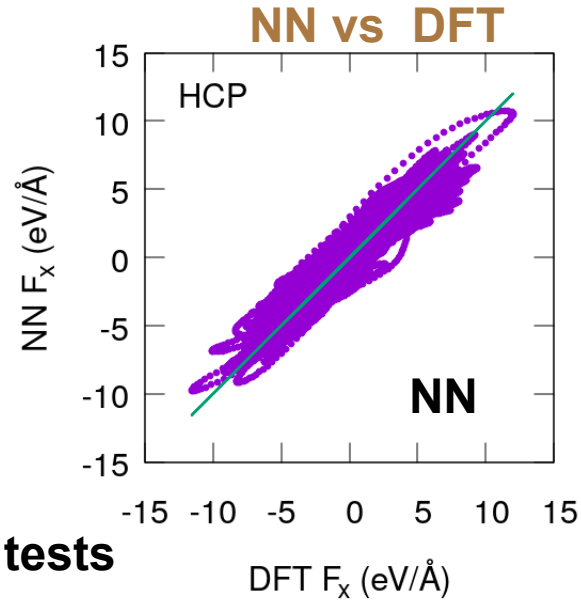


Al dislocation
700 K





HCP Al
1000-4000 K



… and many similar tests

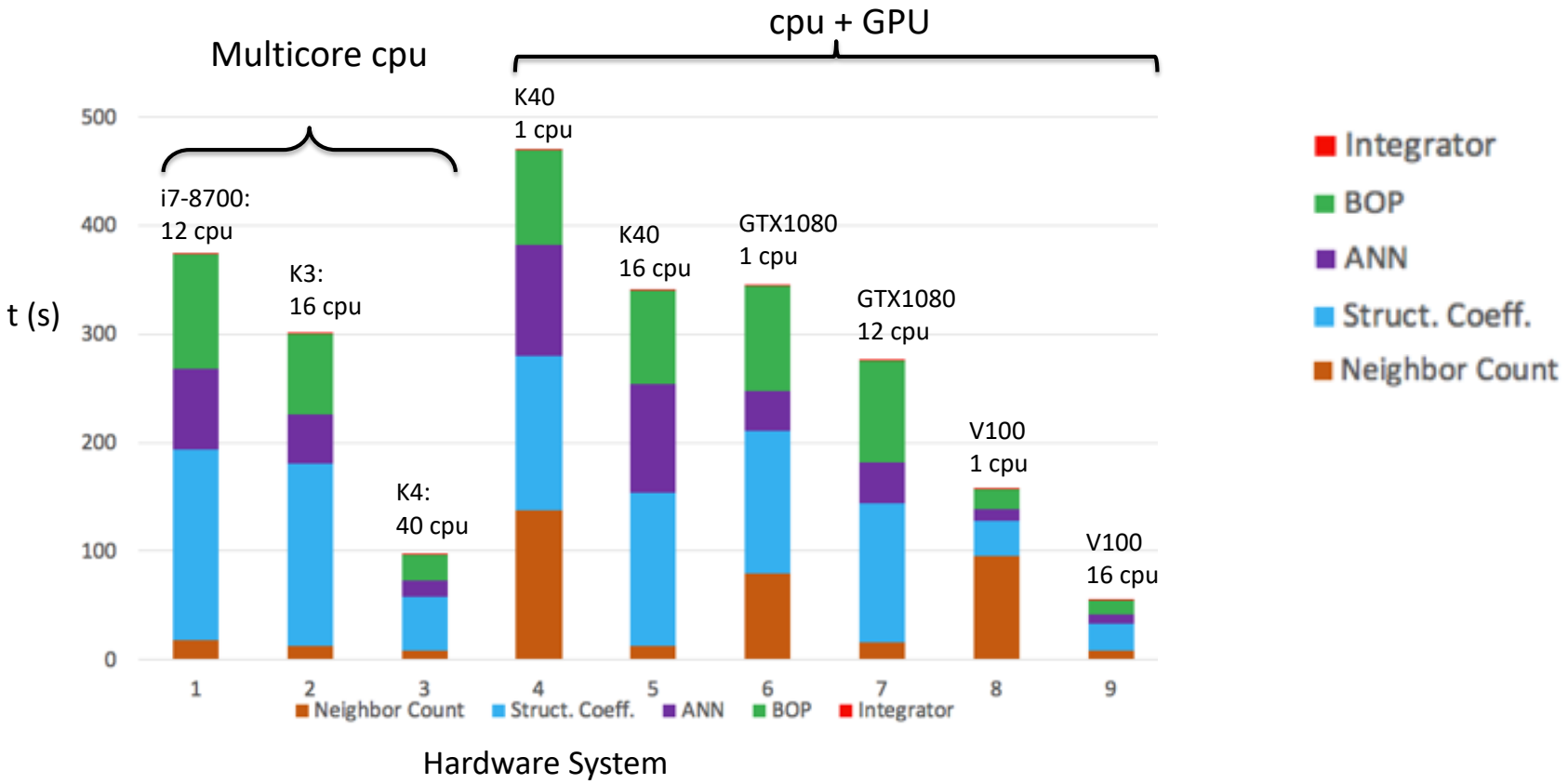
Computational performance of PINN

- ~*N* scaling
- x100 slower than traditional potentials. Much faster than DFT
- Scalable software: ParaGrandMC (V. Yamakov, NASA)

https://software.nasa.gov/software/LAR-18773-1

- MD and Monte Carlo
- Parallelized with MPI, OpenMP and GPUs

Computational performance of PINN



- Calculation of the descriptors in the bottleneck
- Relatively small overhead of PINN versus NN (~25%)

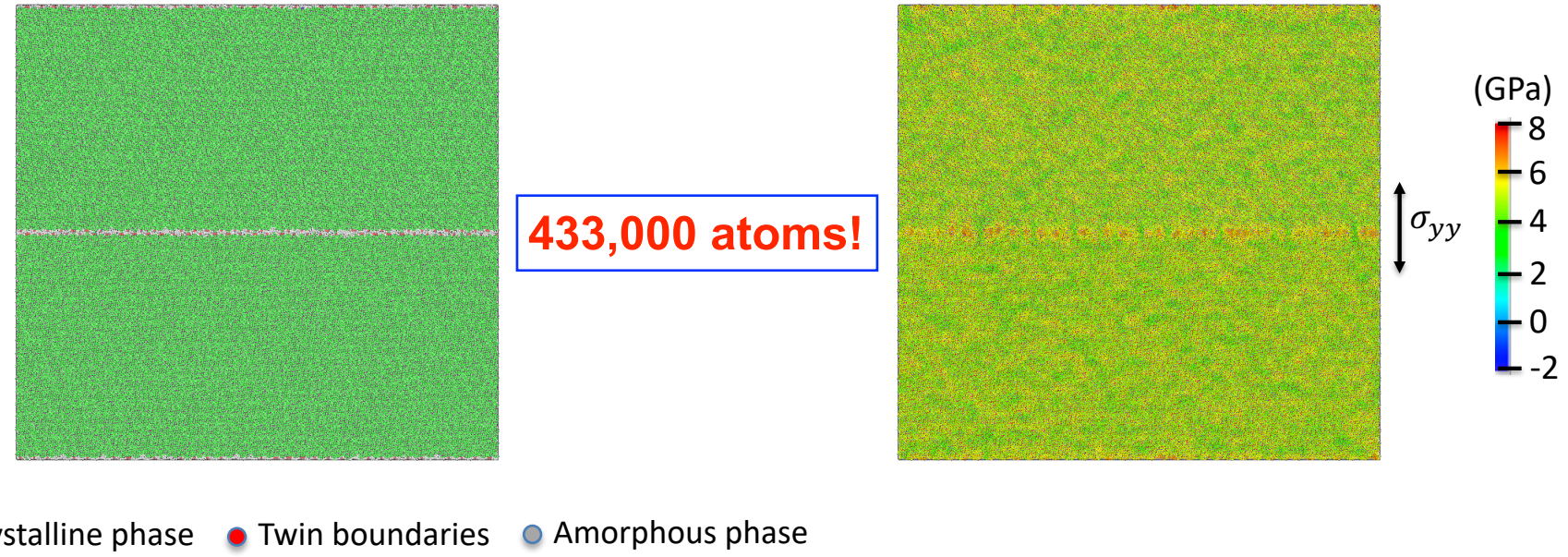
Example: large scale MD with AI PINN potential

Crack nucleation on a (557) symmetric tilt grain boundary

NVT MD at 400 K

Crystalline phase visualization

Stress field visualization



- **MD run on 4 MPI nodes (10 Skylake 6148 CPUs + V100 GPU)/node**
- **MD time 24 ps (12,000,000 MD steps)**
- **CPU time 14 hours**

Conclusions

- ML potentials emerge as next-generation models
- Main limitation – transferability. No physics – no transferability
- To be more transferable, new potential models must incorporate guidance from physics/chemistry
- Future direction of the field: ML + physics
- PINN is one example of physics-guided ML models. Shows much promise
- Existing PINN potentials: Al (done), Si (almost done), Cu, Pt and Ta (in progress)
- Multicomponent PINN in progress