

How neural networks can help in kinetic Monte Carlo simulations in alloys

N. Castin, G. Bonny, D. Terentyev, L. Malerba

Structural Materials Modelling and Microstructure

Nuclear Materials Science Institute

SCK•CEN, Mol – Belgium

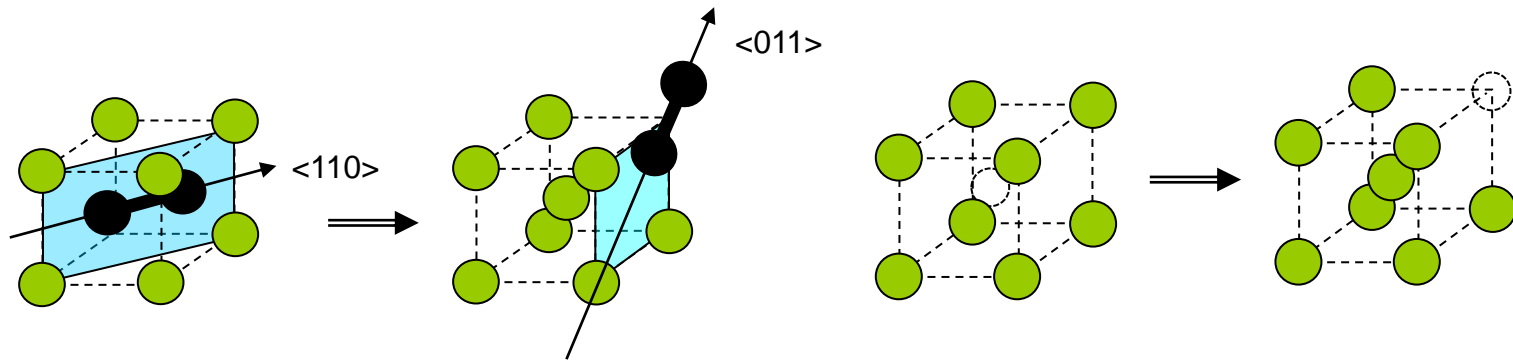
lmalerba@sckcen.be

Objective of this presentation

- Show how the use of neural networks can be helpful to develop kMC models that are truly extensions of MD models
- Emphasise that neural networks are a powerful numerical tool which does not, however, replace (or worse, obliterate) physics

Which processes are we interested in?

Point-defect diffusion-driven processes occurring under irradiation



Class “1.5” processes:

We do not know where the system will go ...

... but we hope it will end up sufficiently close to the experiment we want to simulate

And we know the system will get there via elementary processes for each of which it is *generally* possible to know initial and final state

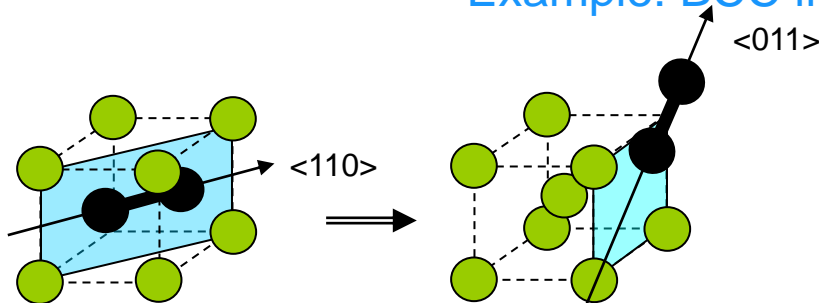
Main problem: in alloys, the actual rates of these elementary processes depend on the combinatorially large number of possible local atomic configurations

Atomistic KMC simulations to extend MD

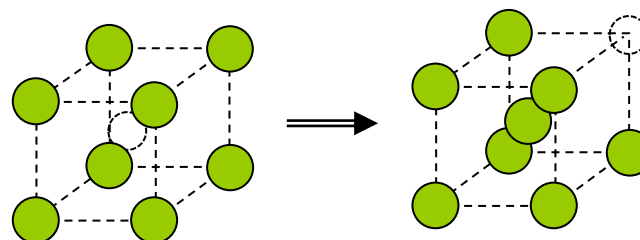
Point-defect diffusion-driven processes take too long for MD

→ Atomistic KMC simulations are widespread techniques to go beyond MD

Example: BCC iron at 600 K



SIA migration: 1 jump / 4 ps



Vacancy migration: 1 jump / 4 ns

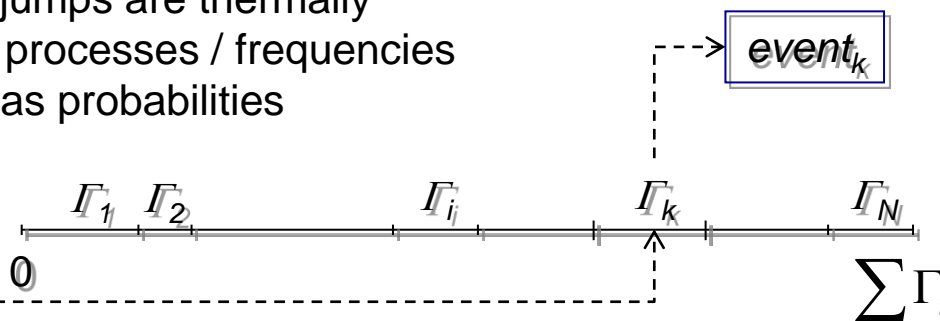
Fundamental equations:

$$\Gamma_i = \nu_0 \cdot \exp\left(\frac{-E_m^i}{k_B \cdot T}\right)$$

Diffusion jumps are thermally activated processes / frequencies are used as probabilities

Monte Carlo algorithm

Random number extraction, $R_n \in [0,1]$



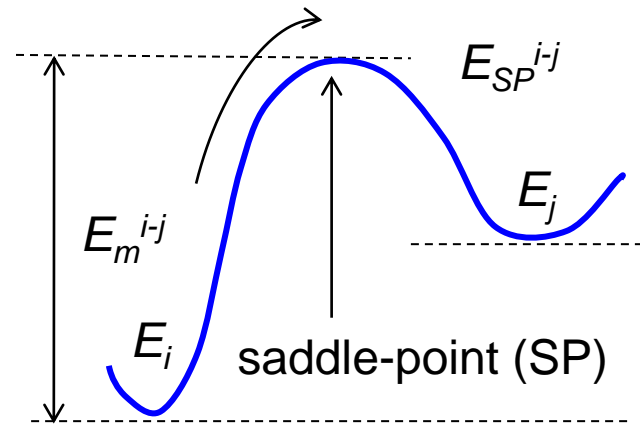
$$Dt \propto 1 / \sum G_i$$

Residence time algorithm

Most physics is contained in the migration energies, E_m^i

Migration energy calculation takes time

$$E_m^{i \rightarrow j} \stackrel{def}{=} E_{SP}^{i-j} - E_i$$



Point-defect migration energies can be calculated very accurately:

- DFT, interatomic potentials, ...
- Drag method, nudged elastic-band method, ...

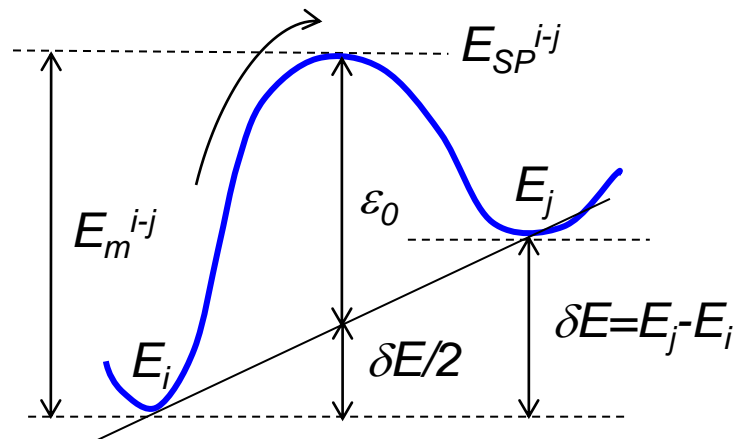
Problem: kMC simulations require migration energy in chemically changing environments to be calculated for each possible point-defect jump, at each timestep, to choose the event

- The more accurate the calculation, the less effective the timescale extension as compared to MD
- **Approximations are used to speed up on-the-fly calculation**

E_m^i are functions of the local environment

$$E_m^{i \rightarrow j} \stackrel{def}{=} E_{SP}^{i \rightarrow j} - E_i = f(c_i, c_{SP})$$

c_i = atomic configuration (positions & chemical nature) in state i



Frequent simplifying assumptions:

$$1) \quad E_m^{i \rightarrow j} = f(c_i, c_j) \stackrel{e.g.}{=} \epsilon_0 + \frac{\delta E}{2} \quad \epsilon_0 = \text{constant, determined by chemical nature of jumping atom}$$

E is typically calculated

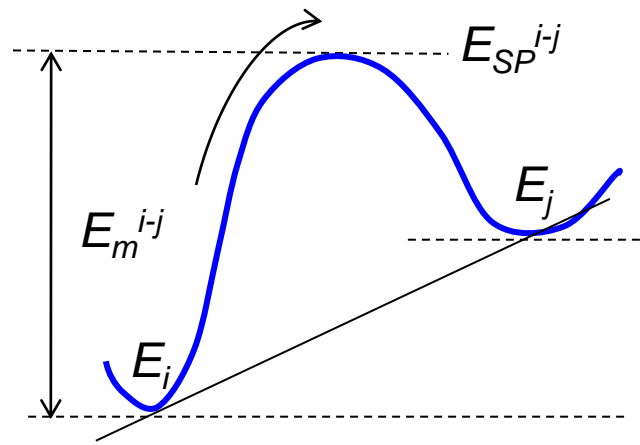
- As summation of pair energy constants extended to closest neighbours (1st & 2nd) or as cluster expansion – parameters nowadays fitted to DFT
- From interatomic potentials used on rigid lattice (relaxation is also possible, but computationally costly)

Problem: irrespective of how accurately δE is computed, correlation between thermodynamics (initial & final states) and kinetics is implicitly assumed

E_m^i are functions of the local environment

$$E_m^{i \rightarrow j} \stackrel{\text{def}}{=} E_{SP}^{i \rightarrow j} - E_i = f(c_i, c_{SP})$$

c_i = atomic configuration (positions & chemical nature) in state i



Frequent simplifying assumptions:

$$2) \quad E_{SP}^{i \rightarrow j} = f(c_{SP}^{nn})$$

c_X^{nn} = config. limited to 1st-2nd neighbour shell

$$E_i = g(c_i^{nn})$$

Typically f & g are sums of pair energy constants (nowadays fitted to DFT)

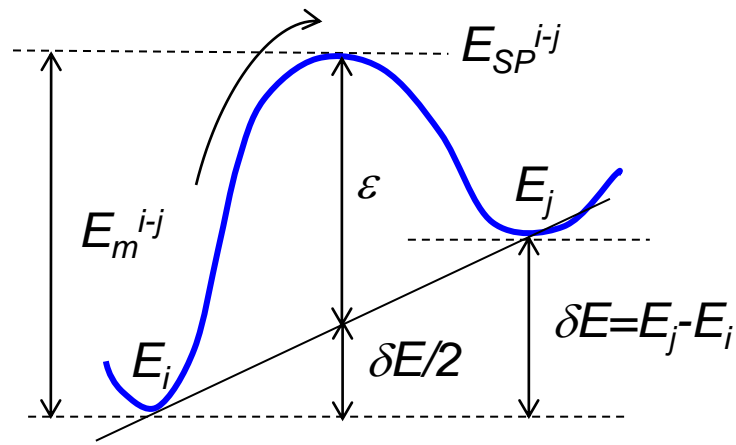
Problems:

- Medium-to-long range chemical interactions and strain field effects are disregarded (especially serious for E_{SP})
- How reliable can a summation of close neighbour pair energy constants be to reproduce complex many-body interactions from DFT (think of concentrated alloys ...)?

E_m^i are functions of the local environment

$$E_m^{i \rightarrow j} \stackrel{\text{def}}{=} E_{SP}^{i \rightarrow j} - E_i = f(c_i, c_{SP})$$

c_i = atomic configuration (positions & chemical nature) in state i



Less frequent assumption:

$$3) \quad \varepsilon = f(c_{SP}) \stackrel{\text{cluster expansion}}{=} K_0 + \sum_{\alpha} K_{\alpha} \phi_{\alpha} \quad \begin{array}{l} \text{fitted coeff.} \\ \downarrow \\ \alpha = \text{any cluster} \\ \text{defined in lattice} \\ \text{with atom at SP} \end{array}$$

$$\phi_{\alpha} = \prod_{i \in \alpha} \sigma_i \quad \sigma_i = \text{on-site variable that takes a different value depending on the type of atom at site } i \text{ of cluster } \alpha$$

$$E_i = f(c_i) \stackrel{\text{cluster expansion}}{=} E_0 + \sum_{\beta} E_{\beta} \phi_{\beta} \quad \Rightarrow \quad E_m^{i \rightarrow j} = \varepsilon + \frac{\delta E}{2}$$

← fitted coeff. ↑

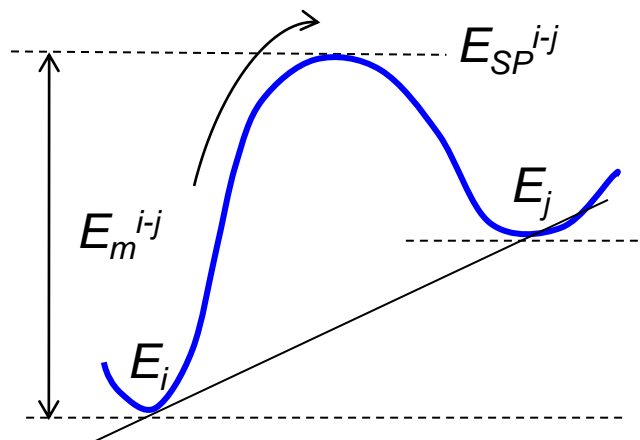
Problem:

- Choice of clusters, where to truncate expansion, ...: convergence with cluster expansion is not easy matter!

E_m^i are functions of the local environment

$$E_m^{i \rightarrow j} \stackrel{def}{=} E_{SP}^{i \rightarrow j} - E_i = f(c_i, c_{SP})$$

c_i = atomic configuration (positions & chemical nature) in state i



Our assumption:

- 4) $E_m^{i \rightarrow j} = f(\{\sigma_i; i \in A\})$ *A = sufficiently large cluster including all atoms that define the local environment around the migrating defect*

$$\Rightarrow \{\alpha\}, \{\beta\} \subset A$$

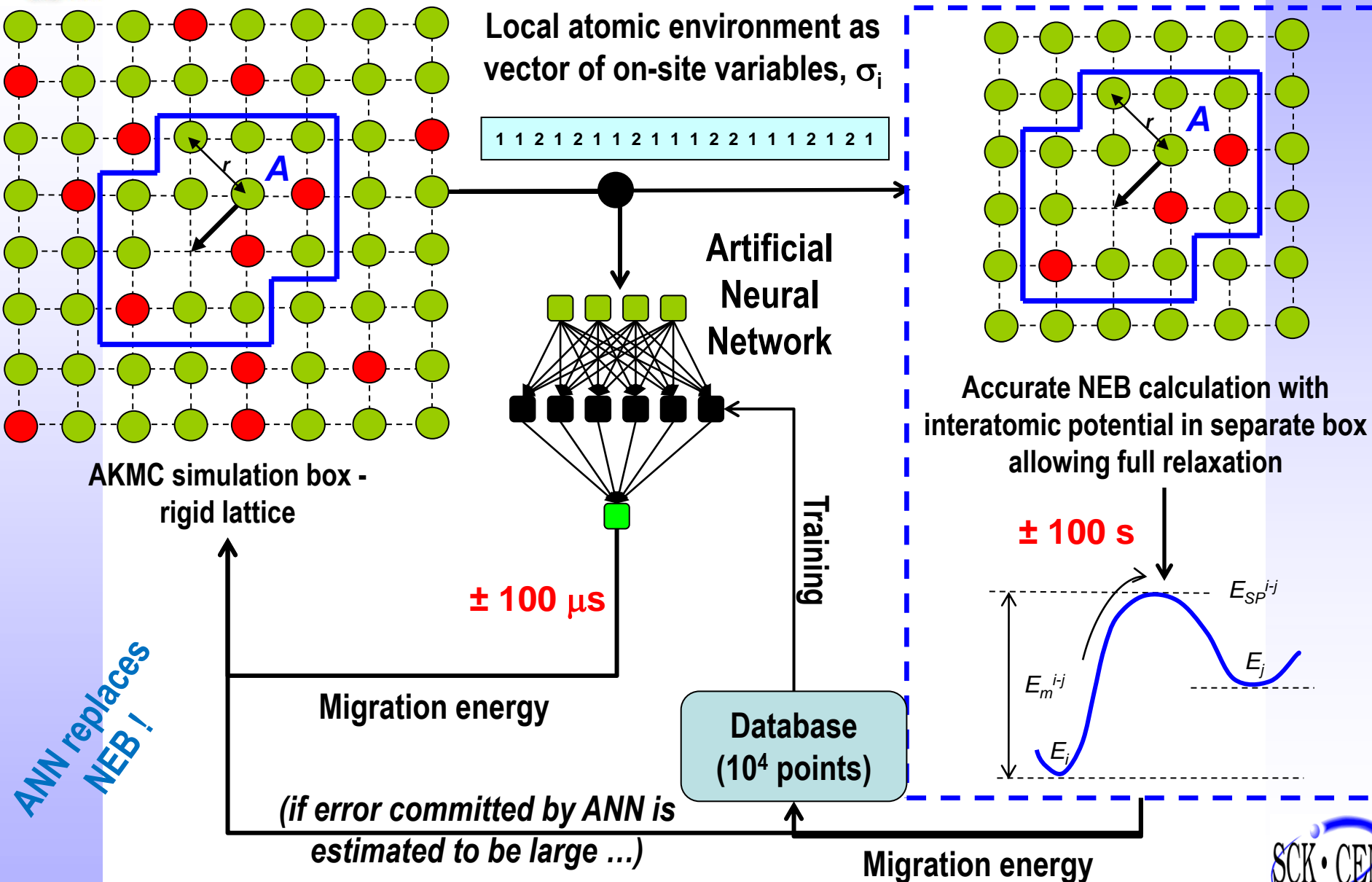
Problem:

- We do not know how this multivariable function looks like

Solution:

- Resort to powerful universal regression tool to build an approximation to this function: artificial neural network

How does it work in practice

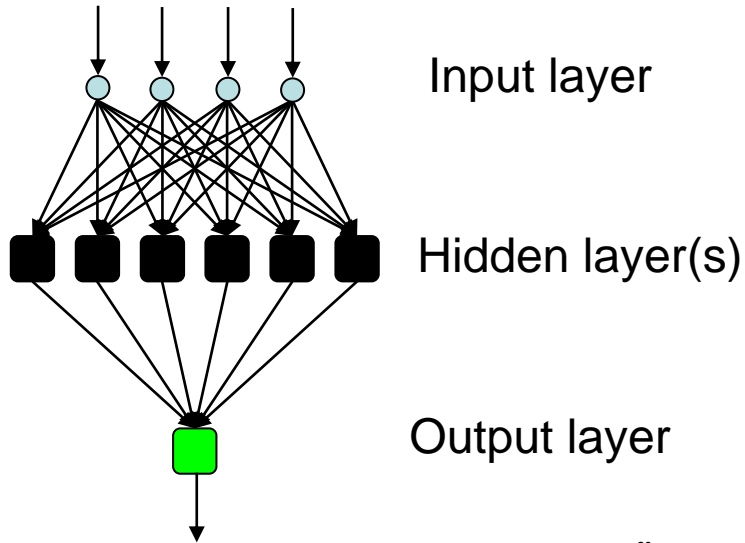


ANN replaces NEB!

How an ANN looks like

Artificial neural networks are considered as **universal approximation tools**: in theory they can reproduce any mapping between input and output variables

$$\{x_i, i=1, \dots, N\}$$



Input layer

Hidden layer(s)

Output layer

Network of simple processing units arranged in layers (signal propagated from top to bottom)

$$\{o_j, j=1, \dots, n\}$$

1 in our case

$$o = \tanh\left(\sum_{i=1}^N W_{oi} x_i + W_{jo}\right)$$

Network "synapses" to be found by training.

Input signals (on-site variables)

How an ANN is trained

	n	p	u	t	output																						
2	1	1	1	2	3	3	2	1	1	2	1	1	2	2	1	2	1	1	2	1	1	2	2	2	2	3	0.4204
1	1	2	1	1	2	2	1	2	1	3	1	1	2	3	1	2	3	3	2	2	2	2	2	1	3	1	0.4271
1	2	2	1	2	1	3	2	2	3	2	2	3	1	3	2	1	1	1	3	1	1	2	3	1	2	3	0.4546
1	2	1	1	2	2	1	2	1	3	2	2	3	2	2	3	1	3	1	2	1	1	3	2	2	2	1	0.4642
1	3	1	1	2	3	1	2	3	1	2	1	1	2	2	1	2	1	1	3	1	2	2	2	1	1	1	0.3219
2	2	1	1	2	2	1	2	2	1	2	3	1	2	2	2	2	1	1	2	1	1	2	3	1	2	1	0.3397
1	2	1	1	2	2	1	2	1	3	2	2	3	2	2	3	1	3	3	2	2	3	2	2	3	1	3	0.3514
2	2	3	2	2	3	1	3	3	2	2	1	1	2	2	1	1	1	1	1	1	1	1	1	1	1	1	0.3646
2	3	2	2	3	1	3	3	2	1	2	1	1	2	2	1	2	1	1	2	1	1	2	2	1	2	1	0.3470
2	1	1	1	2	3	3	2	3	1	2	1	1	2	2	1	2	1	2	2	2	2	3	1	3	2	0.3690	
1	3	1	1	2	3	1	2	3	1	3	1	1	2	3	1	2	3	1	2	1	1	2	2	1	2	1	0.3824
1	2	1	1	2	2	1	2	1	3	2	2	3	2	2	3	1	3	2	1	1	1	2	3	3	2	1	0.3966
2	1	1	1	2	3	3	2	1	2	1	1	2	2	1	2	1	1	2	3	1	3	2	1	3	2	0.4051	
2	1	1	1	2	3	3	2	1	2	1	1	2	2	1	2	1	1	2	1	1	2	2	2	2	3	1	0.4204
1	1	2	1	1	2	2	1	2	1	3	1	1	2	3	1	2	3	3	2	2	2	2	2	1	3	1	0.4271
1	2	2	1	2	1	2	2	2	2	2	2	2	1	2	2	1	1	1	2	2	1	2	2	1	2	2	0.4546
1	2	1	1	2	2	1	2	1	3	2	2	3	2	2	3	1	3	2	2	3	1	3	2	2	2	1	0.4642
1	3	1	1	2	3	1	2	3	1	2	1	1	2	2	1	2	1	1	3	1	2	2	2	1	1	1	0.3219
2	2	1	1	2	2	1	2	2	1	2	3	1	2	2	2	2	1	1	2	1	1	2	3	1	2	1	0.3397
1	2	1	1	2	2	1	2	1	3	2	2	3	2	2	3	1	3	3	2	2	3	2	2	3	1	3	0.3514
1	3	1	1	2	3	1	2	3	1	2	1	1	2	2	1	2	1	1	3	1	1	2	2	2	3	1	0.3941
1	1	2	1	1	2	2	1	2	1	3	1	1	2	3	1	2	3	2	2	1	1	2	2	1	2	2	0.4036
2	1	1	2	3	3	2	3	1	2	1	1	2	2	1	2	1	1	2	2	2	3	1	3	2	0.3690		
1	3	1	1	2	3	1	2	3	1	3	1	1	2	3	1	2	3	1	2	1	1	2	2	1	2	1	0.3824
2	2	1	1	2	2	1	2	2	1	2	3	1	2	2	2	1	1	2	1	1	2	3	1	2	1	0.3397	
1	2	1	1	2	2	1	2	1	3	2	2	3	2	2	3	1	3	2	2	3	2	2	3	1	3	0.3514	
2	2	3	2	2	3	1	3	2	2	1	1	2	2	1	2	2	1	3	1	1	2	2	1	2	1	0.3646	
3	1	1	2	3	3	2	1	2	2	1	1	1	2	3	3	2	1	2	2	1	1	2	3	1	2	1	0.3726
1	2	1	3	2	2	3	2	2	1	3	1	1	2	3	1	2	3	1	2	3	1	3	2	1	2	2	0.3875
1	3	1	1	2	3	1	2	3	1	2	1	1	2	2	1	2	1	1	3	1	1	2	2	2	3	1	0.3941
1	1	2	1	1	2	2	1	2	1	3	1	1	2	3	1	2	3	2	2	1	1	2	2	1	2	2	0.4036
1	2	2	1	3	1	1	2	2	3	2	1	1	1	2	3	3	2	1	2	3	1	2	2	3	2	3	0.4083

Training set: used to design ANN

Reference set: used to measure prediction error

10,000 to 100,000 examples

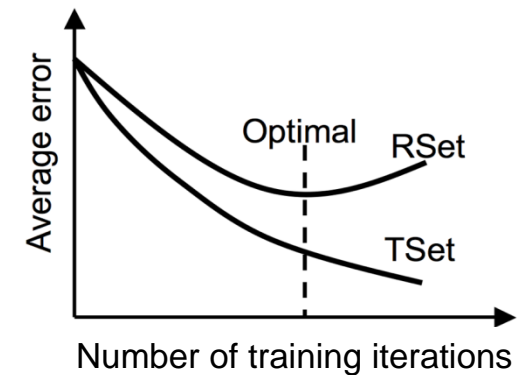
up to 600 variables

Crucial is how the set of examples is built !

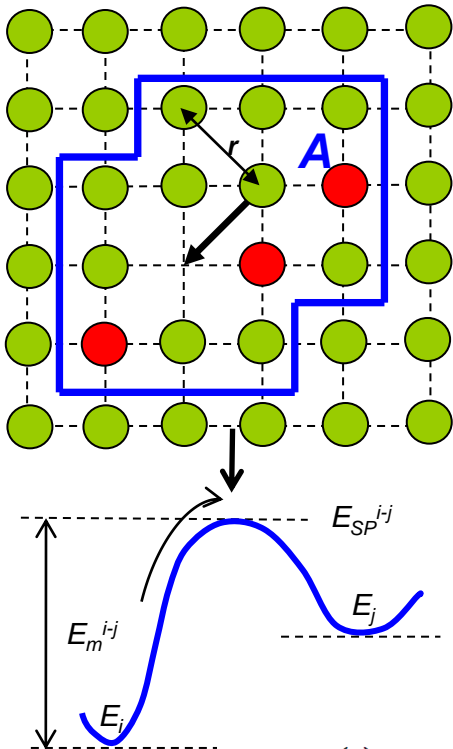
Table of examples

Divided in two non-overlapping sets

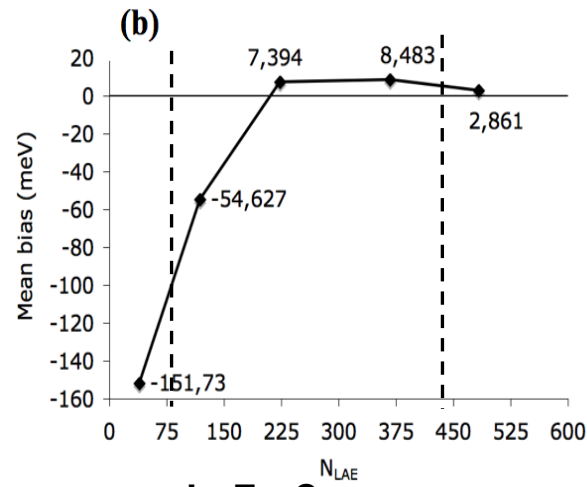
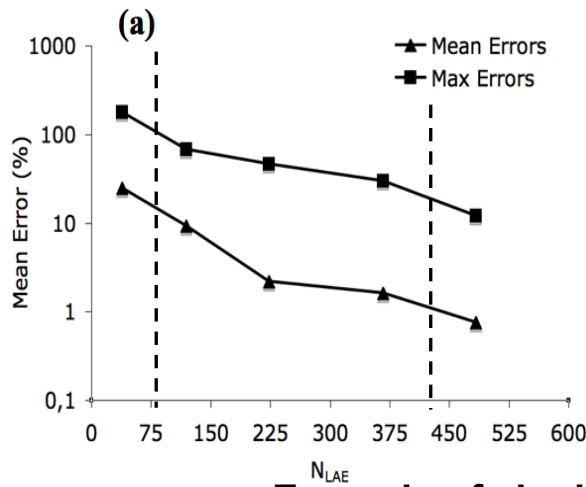
Training iterations continue until error committed on ref. set is minimum (above network becomes overspecialised)



NEB calculations



- Large number of examples to be calculated forces to use **interatomic potentials (2b discussed)**
- Relaxation of initial and final states is performed with conjugate gradients
- NEB is procedural and easily automatized, though SIAs are more delicate than vacs.
- Outside A, matrix atoms are added as fillers; PBC are used
- **How big should A be to converge to a value that does not depend on system size?**

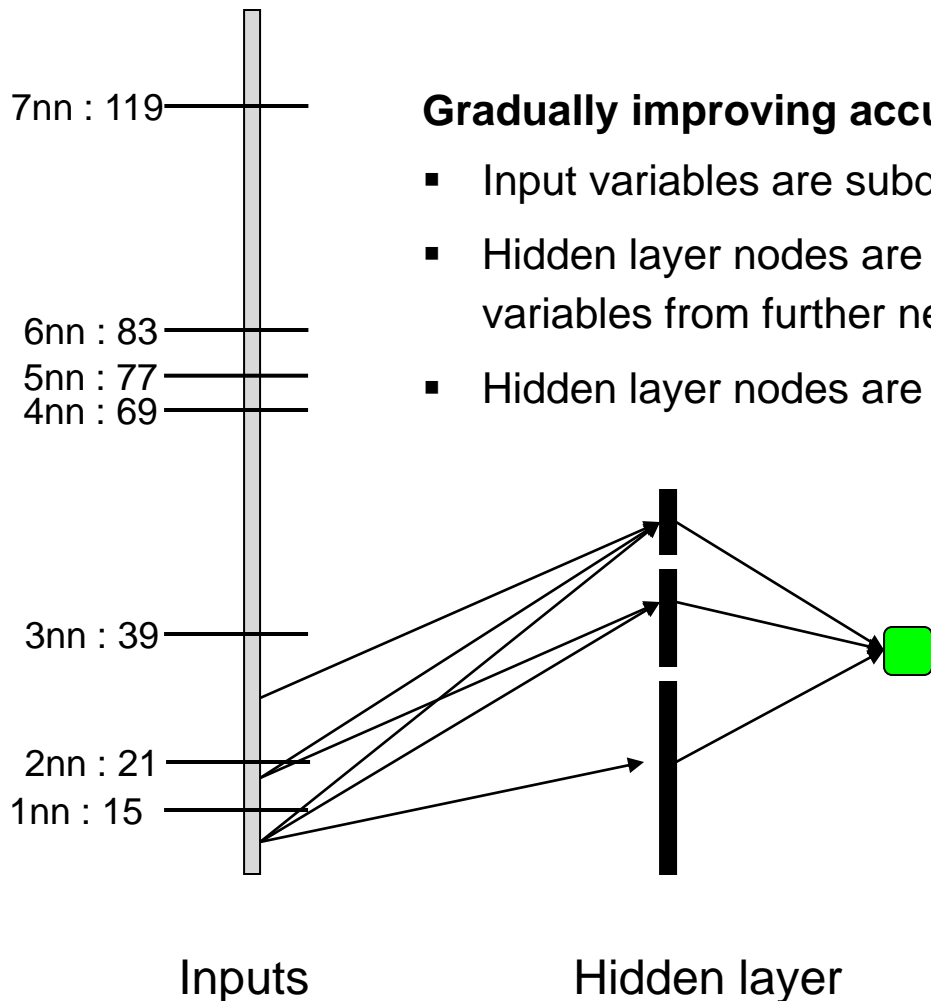


EXAMPLE of single vacancy in Fe-Cr

Incidentally, how big should A be for the ANN?

How many on-site variables need to be considered for the ANN to give a correct answer?

We let the ANN itself decide ...



Gradually improving accuracy constructive algorithm:

- Input variables are subdivided by neighbour shells
- Hidden layer nodes are progressively connected to variables from further neighbour shells
- Hidden layer nodes are added on-demand

Many advantages:

- Optimal architecture of ANN spontaneously determined
- Number of required input variables is minimised
- Training time is reduced

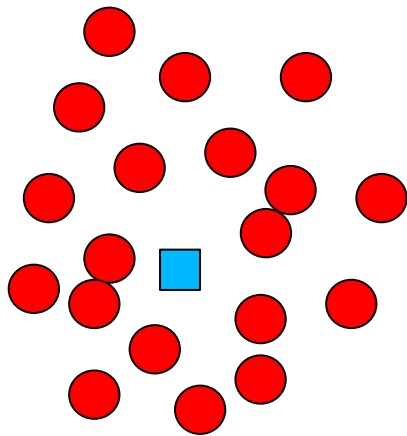
NB: Number is less than for NEB to converge ...

Examples must be physically relevant !

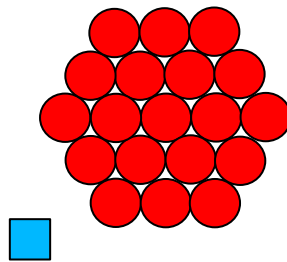
Chosen migration events must be **randomly generated** but **representative** of physical configurations encountered in the AKMC simulation.

One way is by performing AKMC simulations with simpler or preliminary E_m description and extract examples of configs.

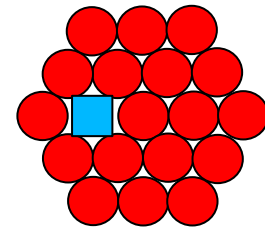
Example: Types of configurations seen by single vacancy in the Fe-Cu system, where Cu is known to precipitate and vacancy can be strongly trapped by precipitates



Vacancy in matrix



Vacancy approaching precipitate



Vacancy inside precipitate

How many examples?

- A priori, not known
- Several thousands guarantee good convergence and can be easily produced with interatomic potentials
- “Experiment” using DFT data for training not performed yet – at least a few hundreds examples are likely to be needed for reliable interpolation
- Great attention paid to evaluation of errors committed at all levels

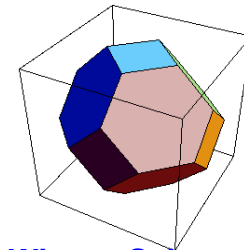
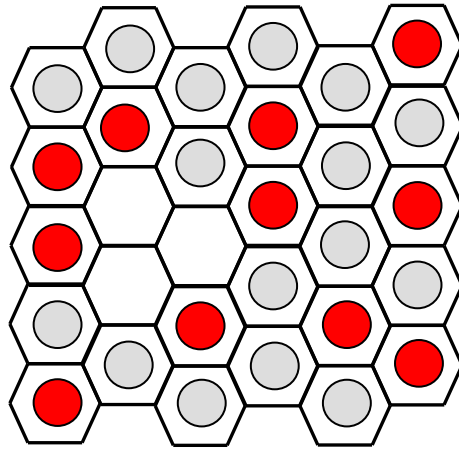
Difficulty: rigid vs relaxed system

NEB requires that initial and final state are known and at least metastable

States that are “topologically” possible on rigid lattice might be unstable once relaxed

Solution:

- On-site variables associated with Wigner-Seitz cells around lattice positions, rather than to perfect positions: it does not matter if atoms move within WS cell (so long as they remain inside)



Wigner-Seitz cell
(bcc structure)

⇒ **Stability of initial configuration is checked (if unstable, not considered for training)**

⇒ **Effect of (0 K) relaxation, thus elastic strain field, up to high order neighbour shell are implicitly accounted for**

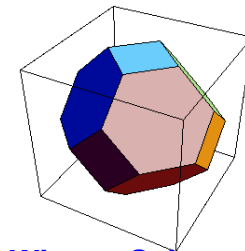
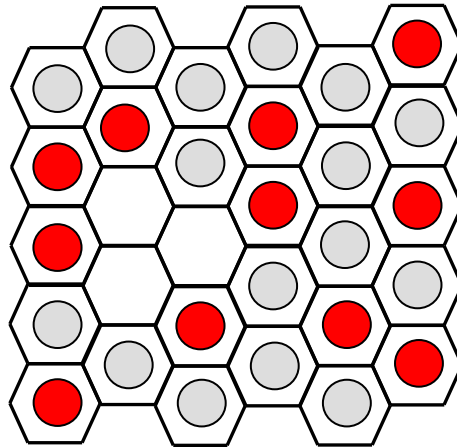
Difficulty: rigid vs relaxed system

NEB requires that initial and final state are known and at least metastable

States that are “topologically” possible on rigid lattice might be unstable once relaxed

Solution:

- On-site variables associated with Wigner-Seitz cells around lattice positions, rather than to perfect positions: it does not matter if atoms move within WS cell (so long as their remain inside)



Wigner-Seitz cell
(bcc structure)

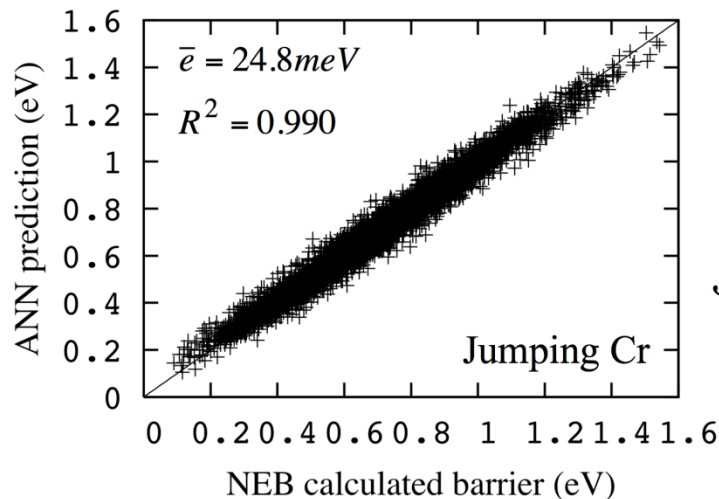
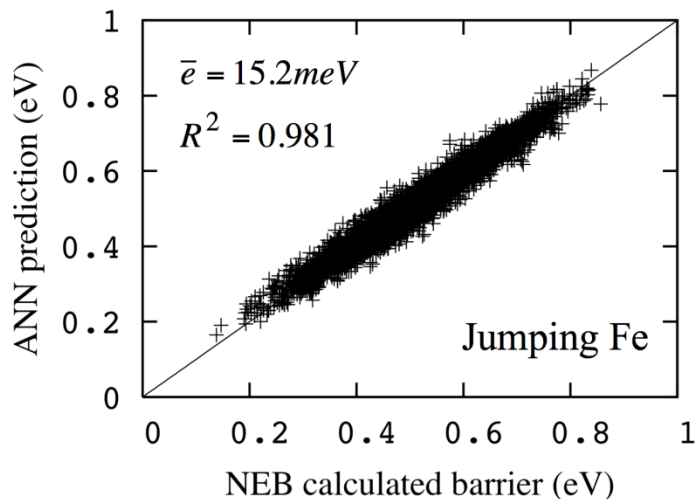
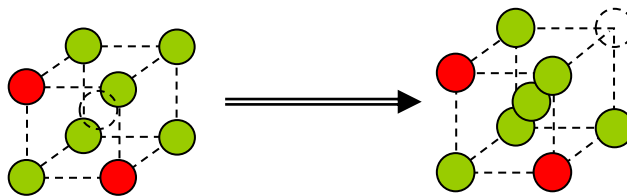
Still open problem

- What if final state is unstable?** (Especially delicate for SIAs)

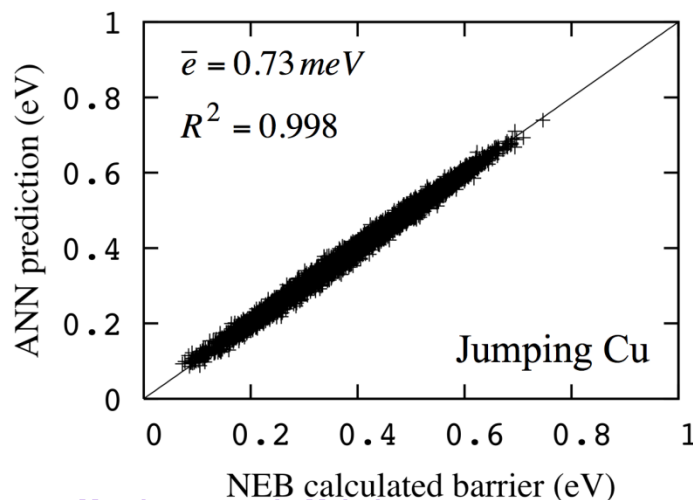
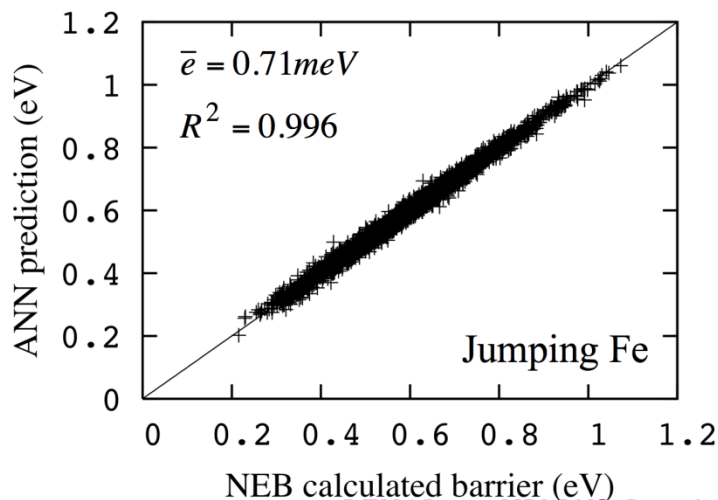
Here solution could be to train ANN to recognise unstable configurations: feasibility is demonstrated but method not yet applied

How well do ANN replace NEB ?

Single vacancy in binary alloy



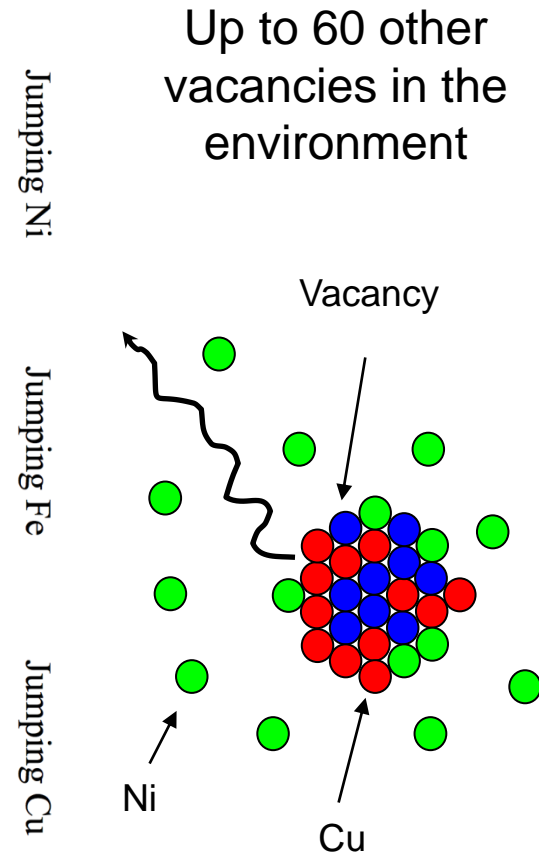
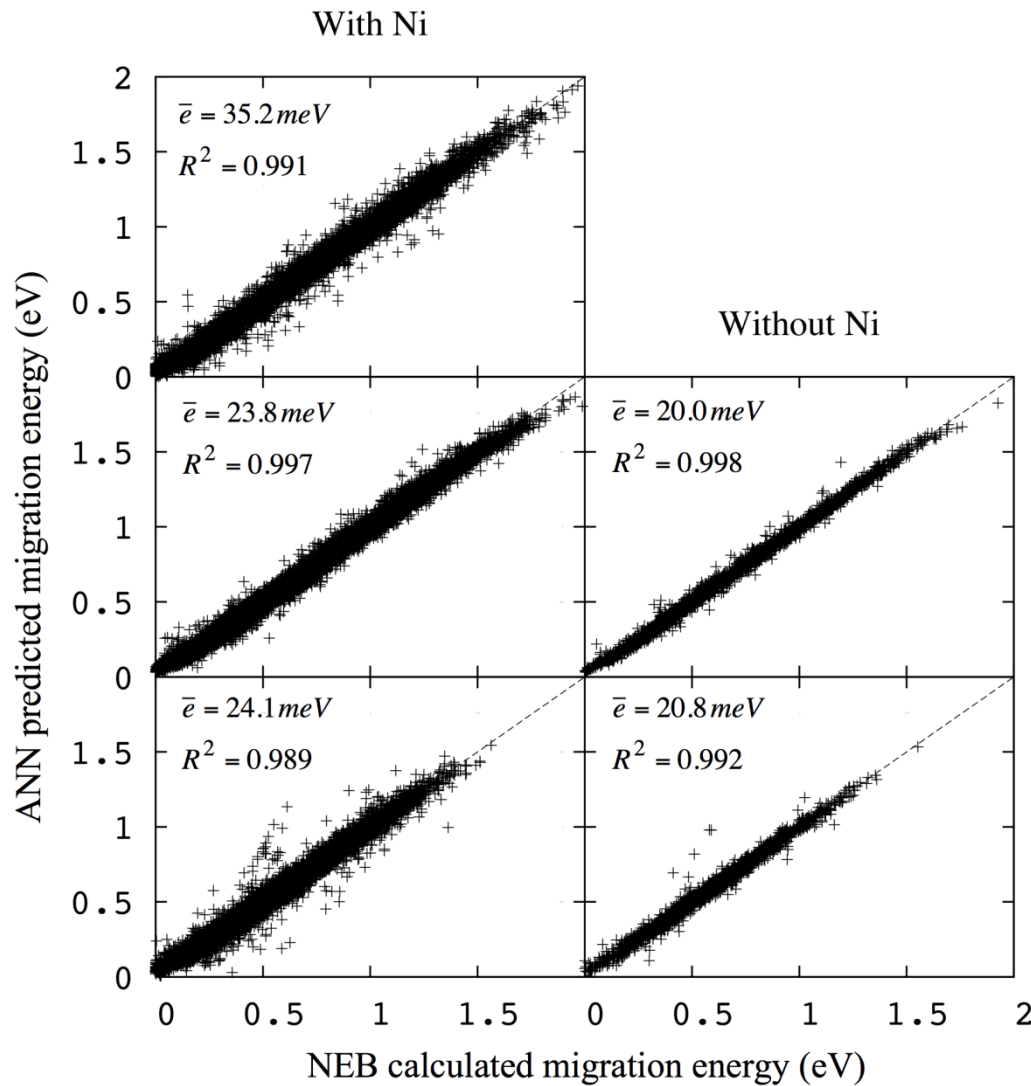
Fe-Cr alloy



Fe-Cu alloy

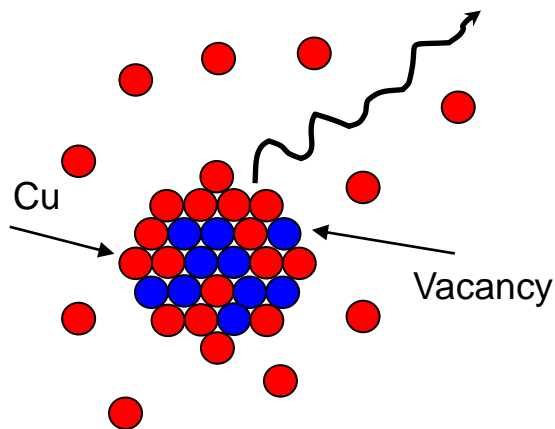
How well do ANN replace NEB ?

Many vacancies in ternary alloy

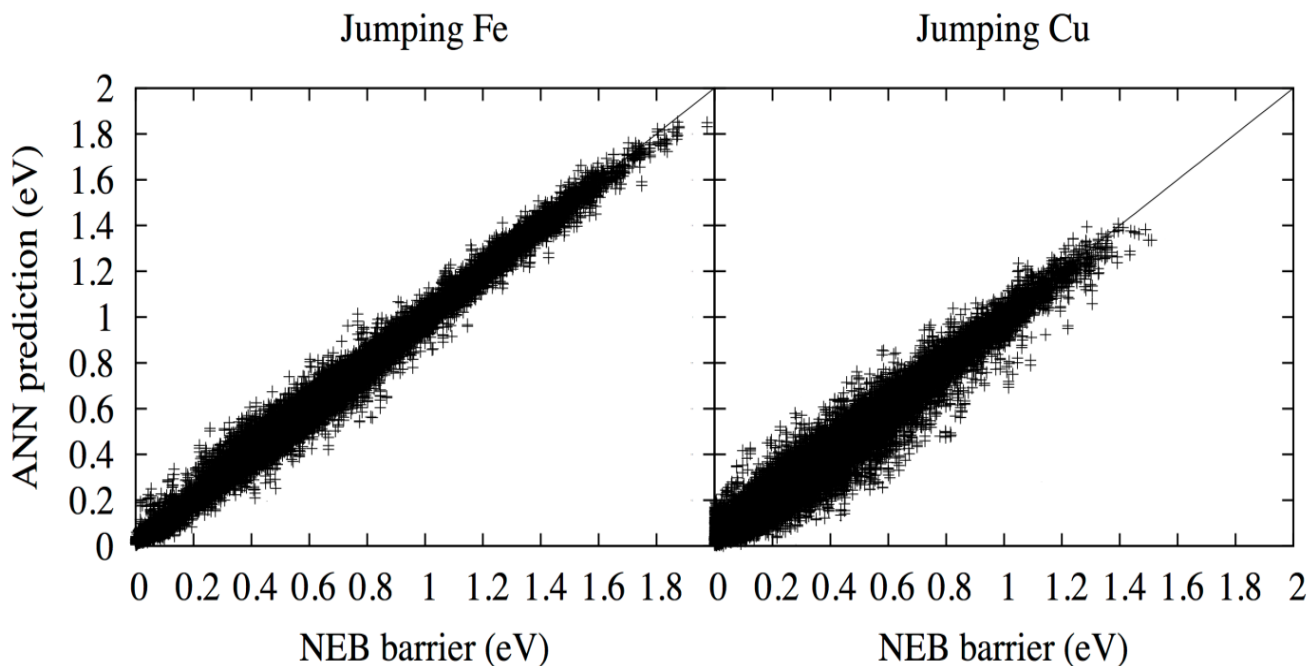


How well do ANN replace NEB ?

Even more vacancies in binary alloy

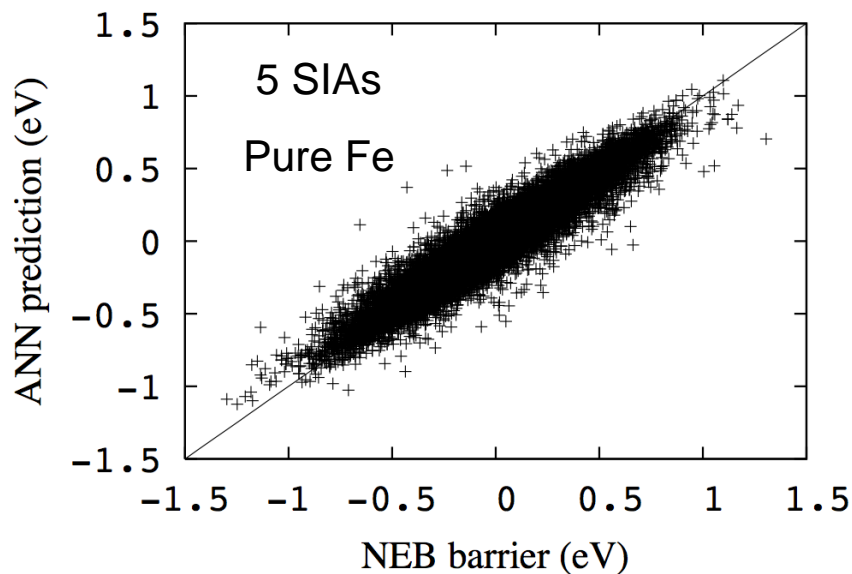
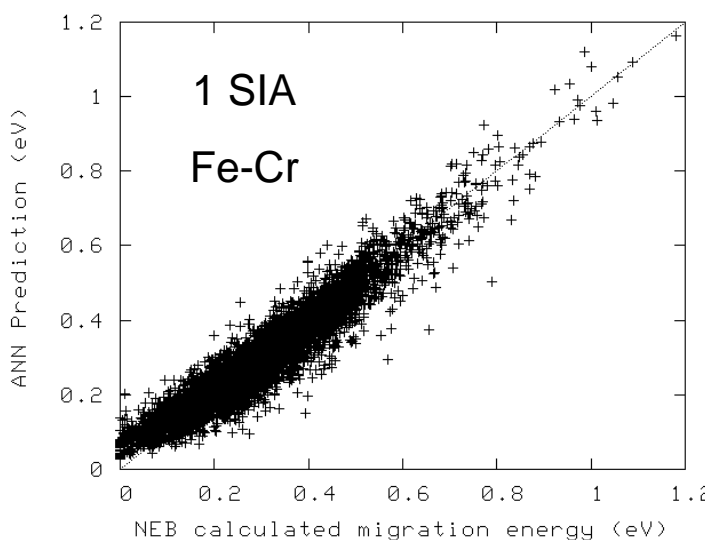
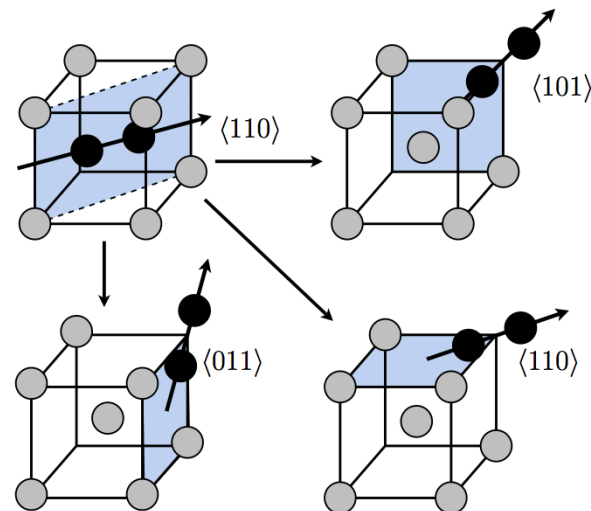


Up to 250 other vacancies in the environment

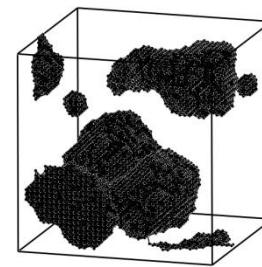
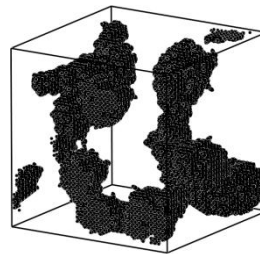
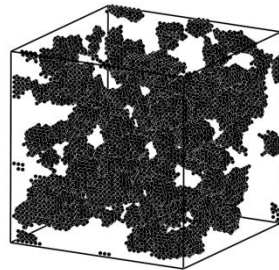
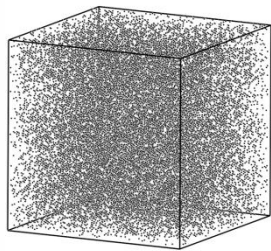


How well do ANN replace NEB ?

Single self-interstitial migrating in binary alloy (FeCr) or in iron but with other SIAs around

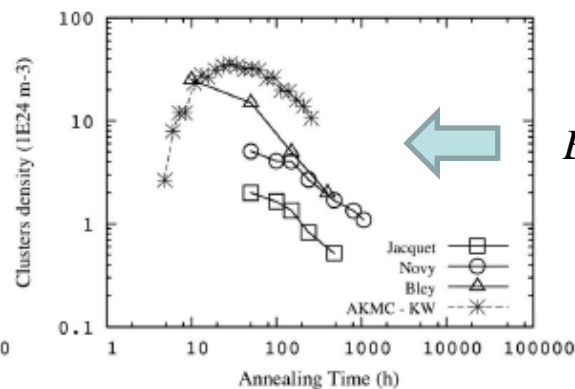
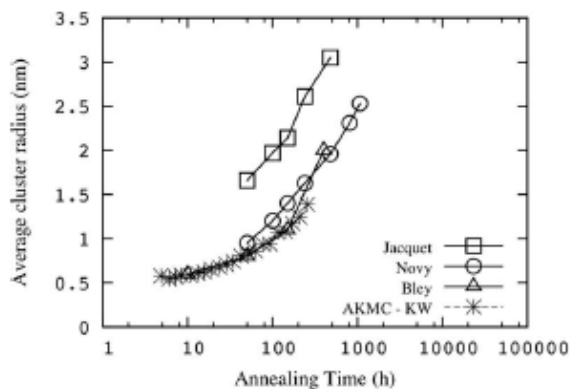


Application: 500°C ageing in Fe20%Cr

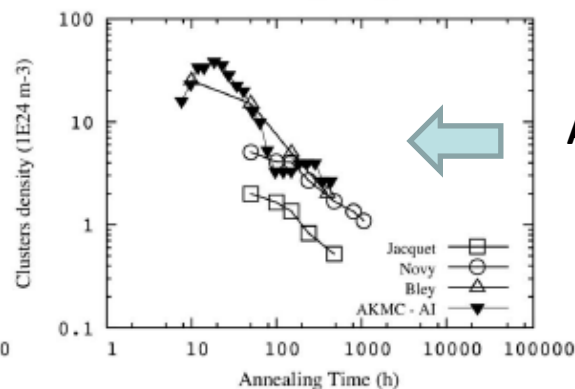
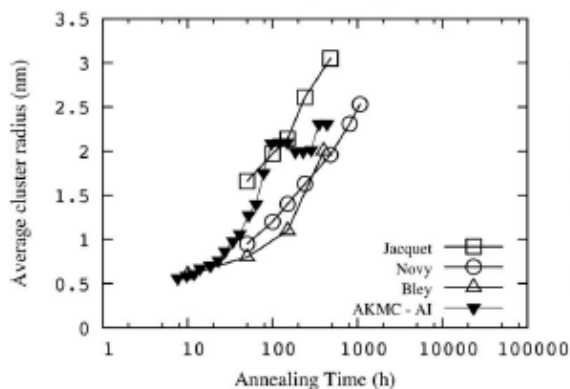


074507-10 N. Castin and L. Malerba

J. Chem. Phys. 132, 074507 (2010)



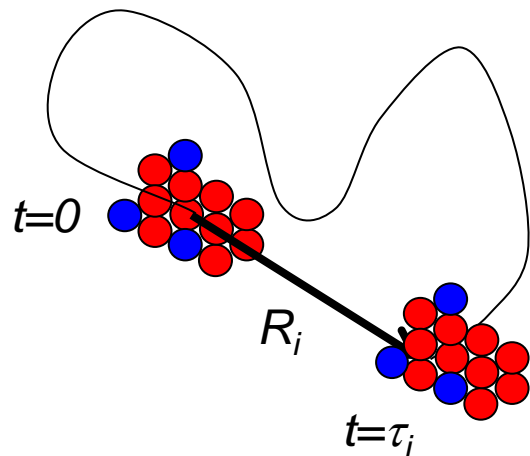
$$E_m^i = \epsilon_0 + \frac{\delta E}{2}$$



ANN

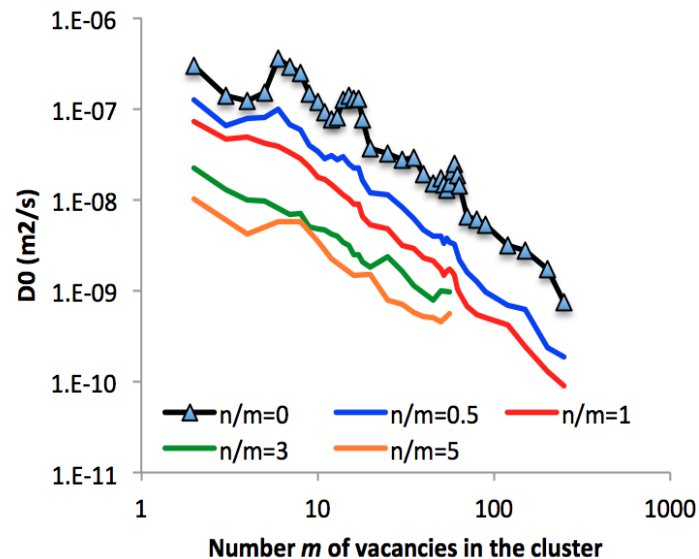
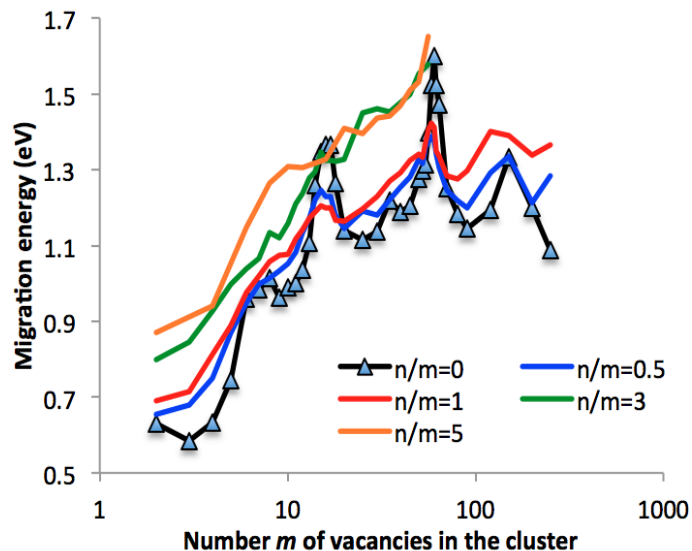
Use of ANN provides simultaneously good prediction of ppt size and density

Application: mobility of Vac_mCu_n clusters



$$D(T) = \frac{1}{N^{sim}} \sum_{i=1}^{N^{sim}} \frac{R_i^2}{6\tau_{life}^i}(T)$$

$$D(T) = D_0 \exp(-E_m^D/k_B T)$$



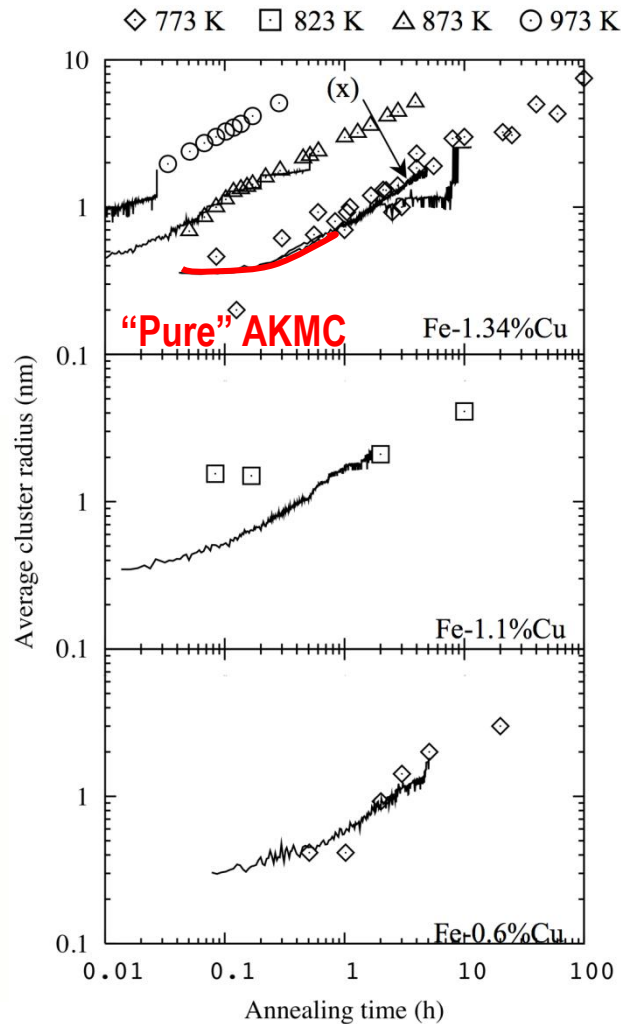
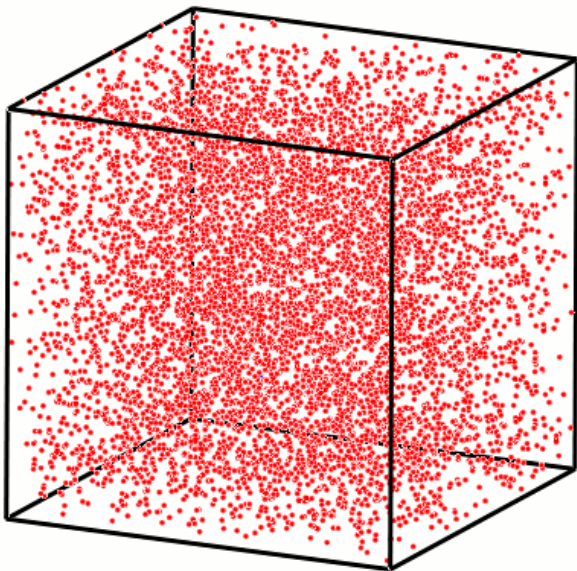
Further application of ANN: interpolate and extrapolate values of $D(T, size, \dots)$, to be plugged in Object kMC models

Hybrid O/AKMC

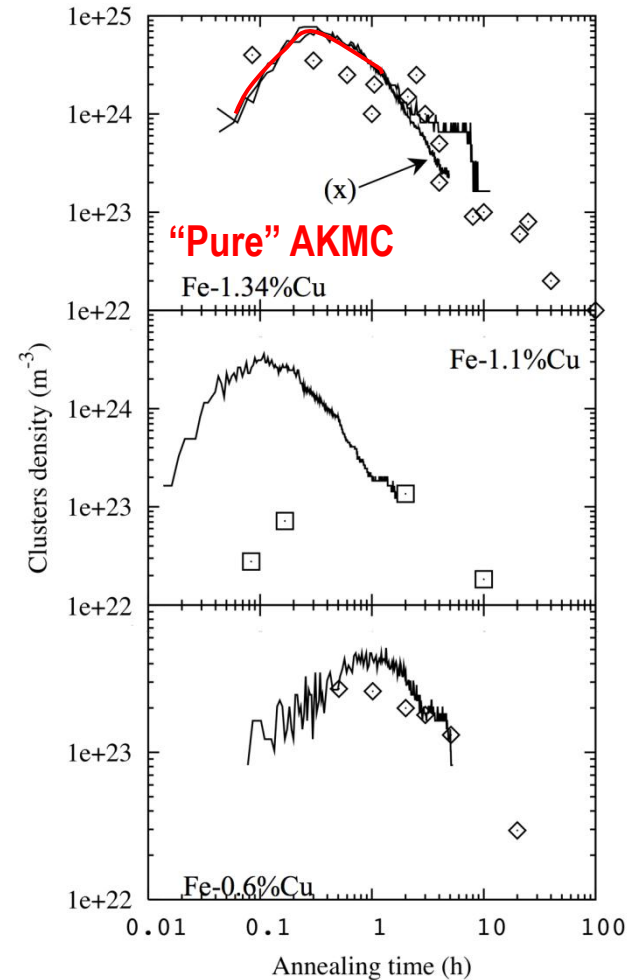
Ageing of Fe_x%Cu at different temperatures

Below a certain size ANN-based AKMC is used.

Above, Cu ppts treated as objects with pre-defined properties predicted by trained ANN



Castin et al., JCP 135, 064502 (2011)



- Mobility of Cu ppts crucial to explain experimental kinetics of pptation
- Hybrid approach instrumental to reach sufficiently long timescales

Still open challenge: large strain fields

➤ SIA clusters:

- *Many initial and final configurations that are 'topologically' possible on rigid lattice are unstable*
- *Likely that hybrid or simplified approach will have to be adopted*

➤ Presence of grain boundaries or other interfaces; dislocations; phases with different crystallography:

- *These problems have not been attacked, yet*
- *Simplified approaches might be necessary*
- *Rigorous approaches may turn out to be possible but highly sophisticated*
 - ✓ *Cost and benefit considerations will be the guide*

Use of interatomic potentials as reference: *is it a limiting factor?*

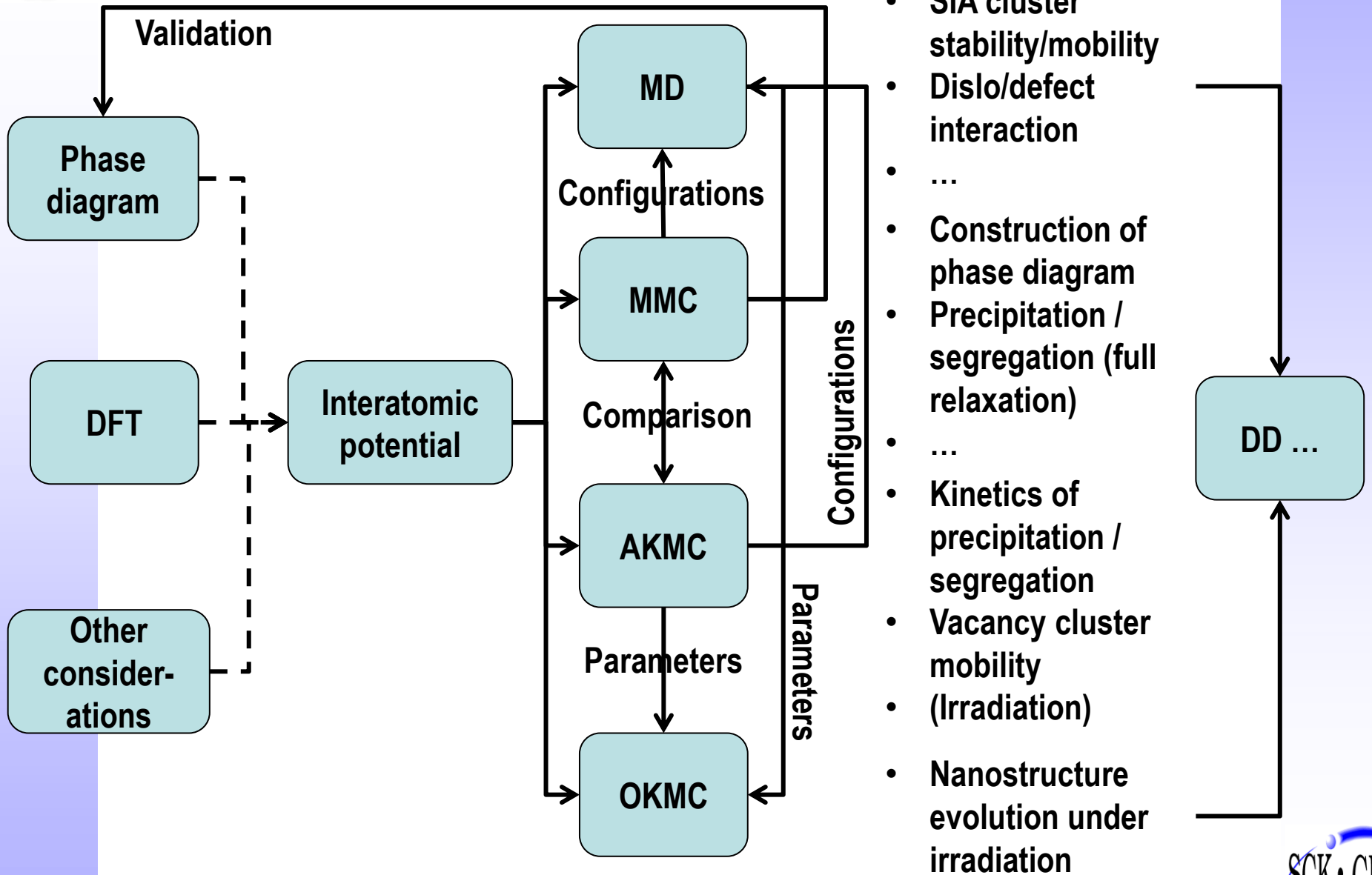
Quite obviously, model cannot do any better than interatomic potential used as reference → ***DFT reference would be better*** – but:

- Different DFT approximations give different values
- How big should box be for NEB with DFT to converge?
- How large is error committed using pair energies or cluster expansion to predict DFT reference data?

Use of interatomic potential provides consistency:

- True extension of MD (besides high T vibrational entropy effects)
- Results from different models and approaches can be combined or compared

Our global approach



- SIA cluster stability/mobility
- Dislo/defect interaction
- ...
- Construction of phase diagram
- Precipitation / segregation (full relaxation)
- ...
- Kinetics of precipitation / segregation
- Vacancy cluster mobility (Irradiation)
- Nanostructure evolution under irradiation

Acknowledgements

FP7/GETMAT & PERFORM60 Projects for funding

People involved in the early development of this methodology:

F. Djurabekova, R. Domingos, G. Cerchiara

Da Ruan (*in memoriam*)