ISTITUTO OFFICINA DEI MATERIALI

Ex DEMOCRITOS

pARTn: a plugin for MEP exploration as a versatile alternative to string approaches

Matic Poberznik Miha Gunde Nicolas Salles and Layla Martin-Samos

Graph

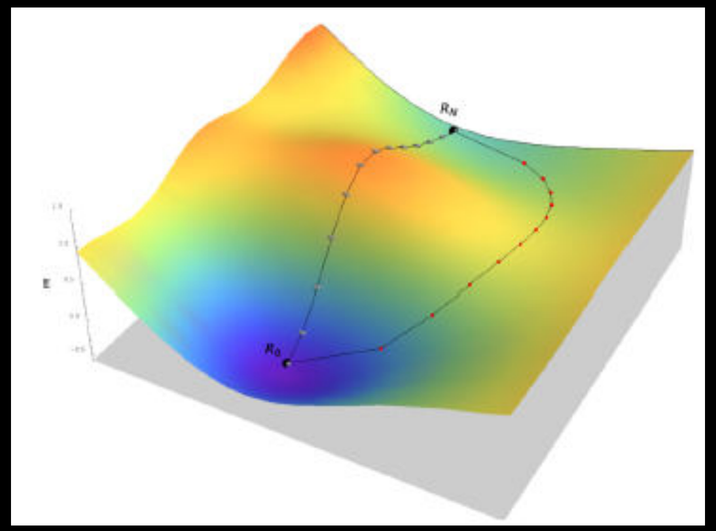
0				1				
	0			1				
		0		1				
			0	1	1			
1	1	1	1	0	1			
			1	1	0			
				1	1	0	1	1
						1	0	
						1		0

Improved Topological and Shape Matching techniques

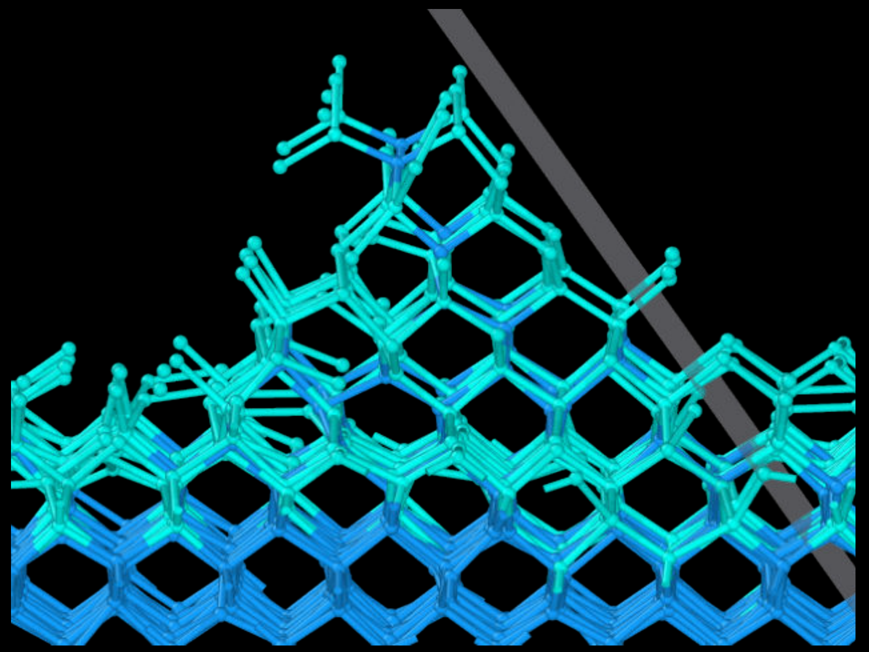
We have a dream…

Object Kinetic Monte Carlo off-lattice and on-

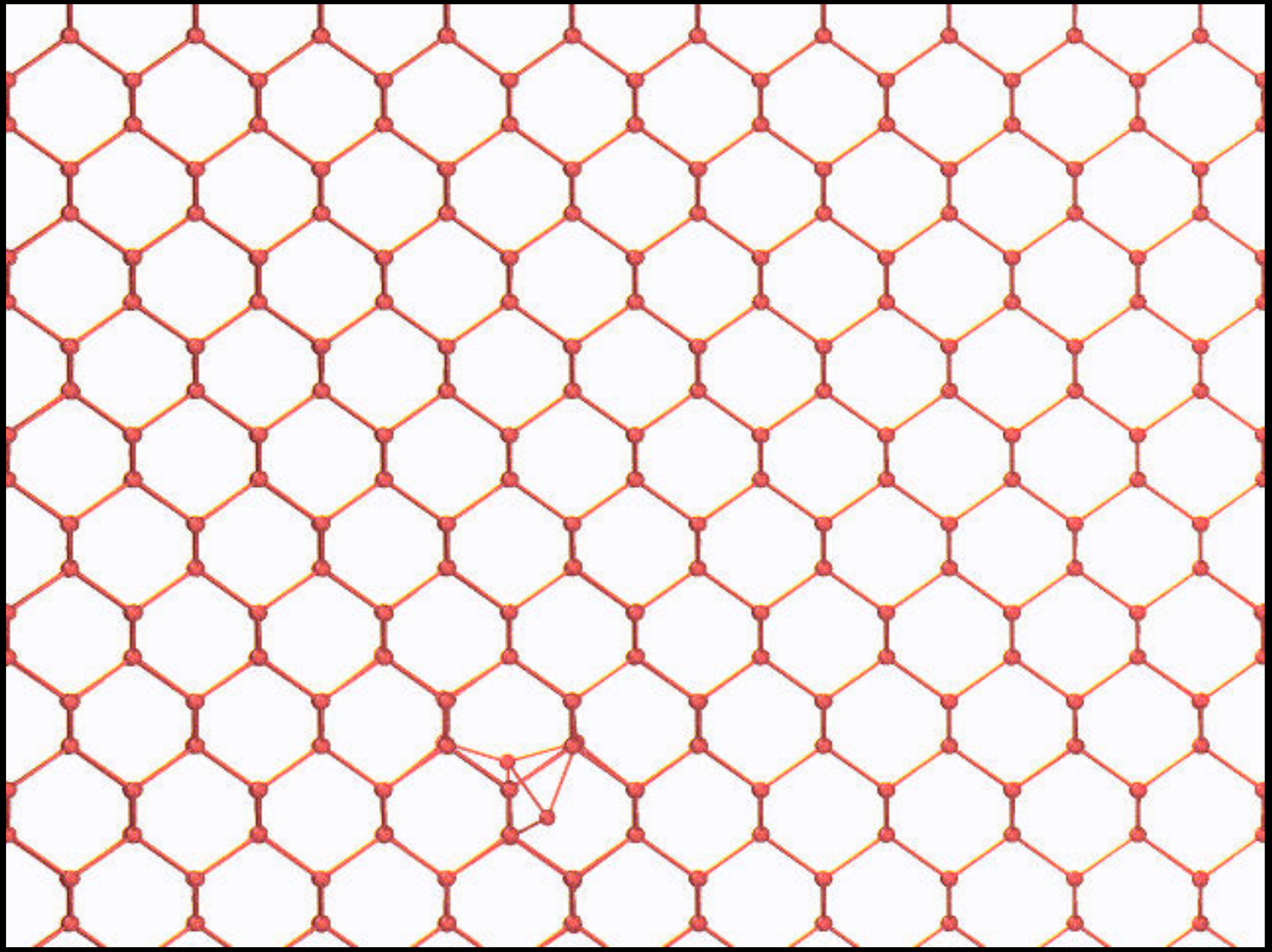
Improvements in rare-event search

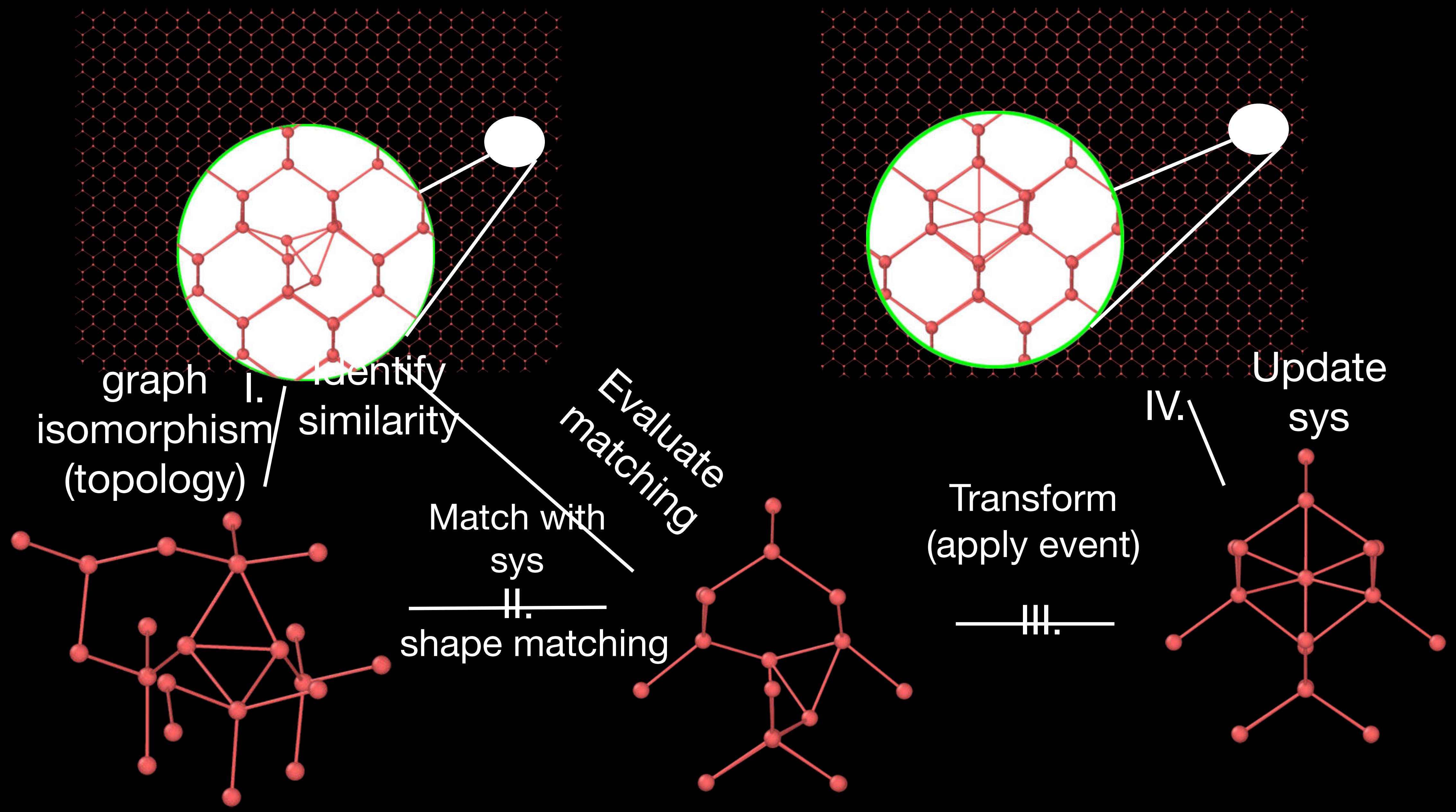


Improved Semi-Empirical potentials



ANN potentials





Graph

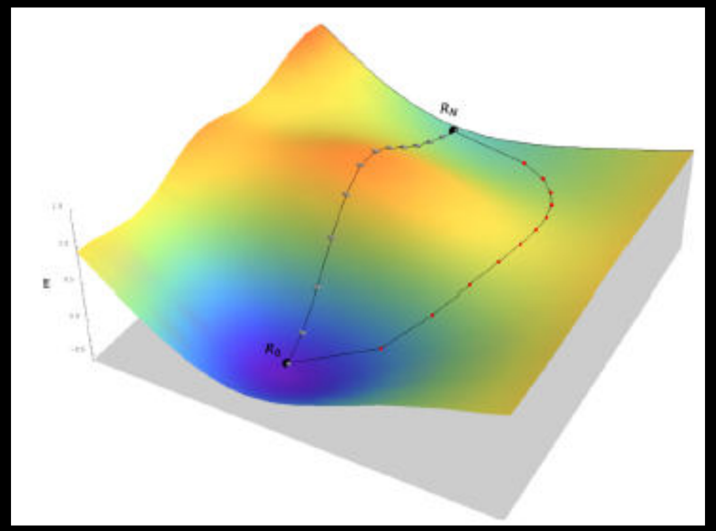
0				1				
	0			1				
		0		1				
			0	1	1			
1	1	1	1	0	1			
			1	1	0			
				1	1	0	1	1
						1	0	
						1		0

Improved Topological and Shape Matching techniques

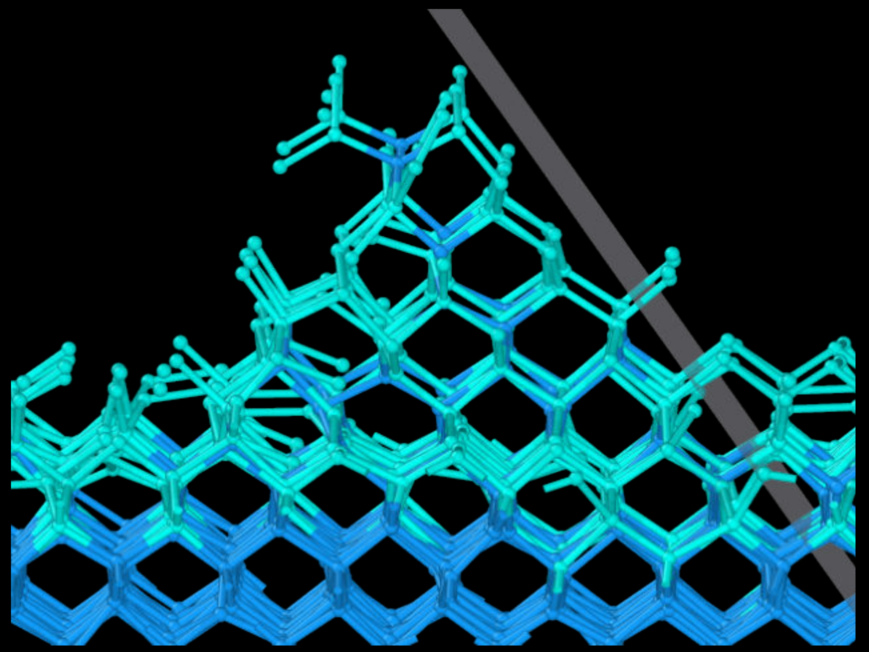
We have a dream…

Object Kinetic Monte Carlo off-lattice and on-

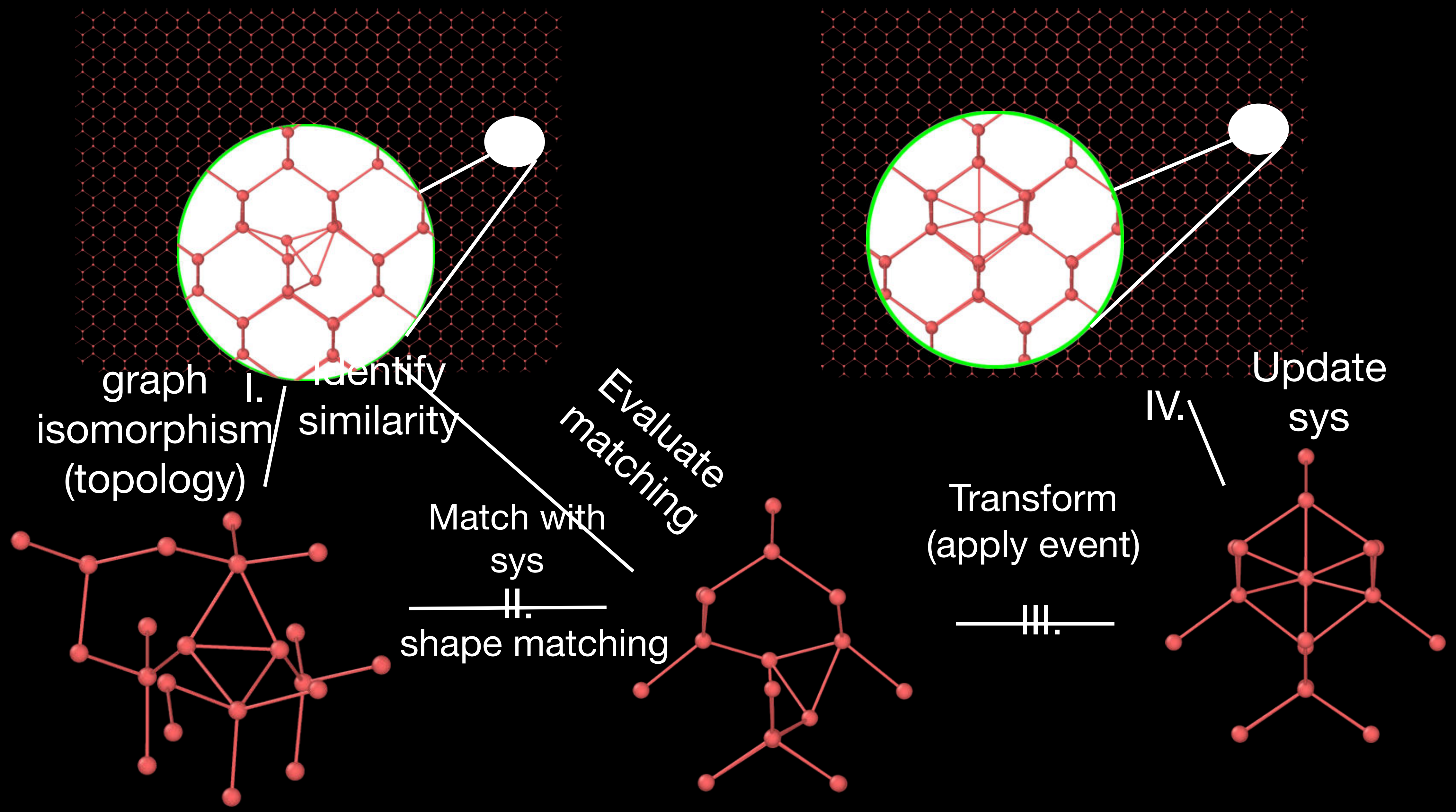
Improvements in rare-event search

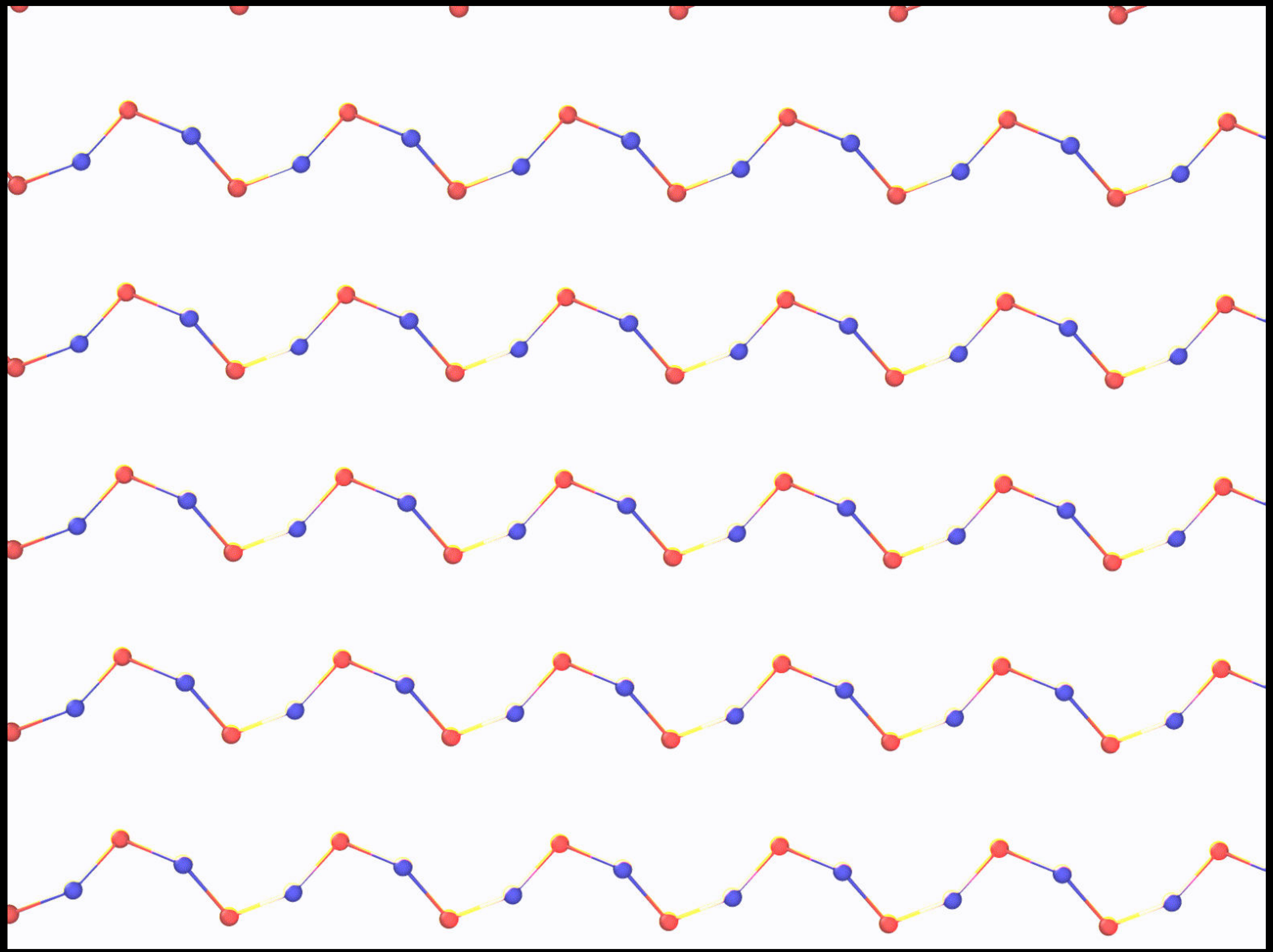


Improved Semi-Empirical potentials



ANN potentials





Once upon a time…

- Bennet or drag: move + perp. Relax
- Hard coded “constrains”
- Empty routine, like in SIESTA-1.1

- Computationally very efficient
- Not automatic (moves and jump direction by hand at each step)
- Cumbersome

The Nudged Elastic Band

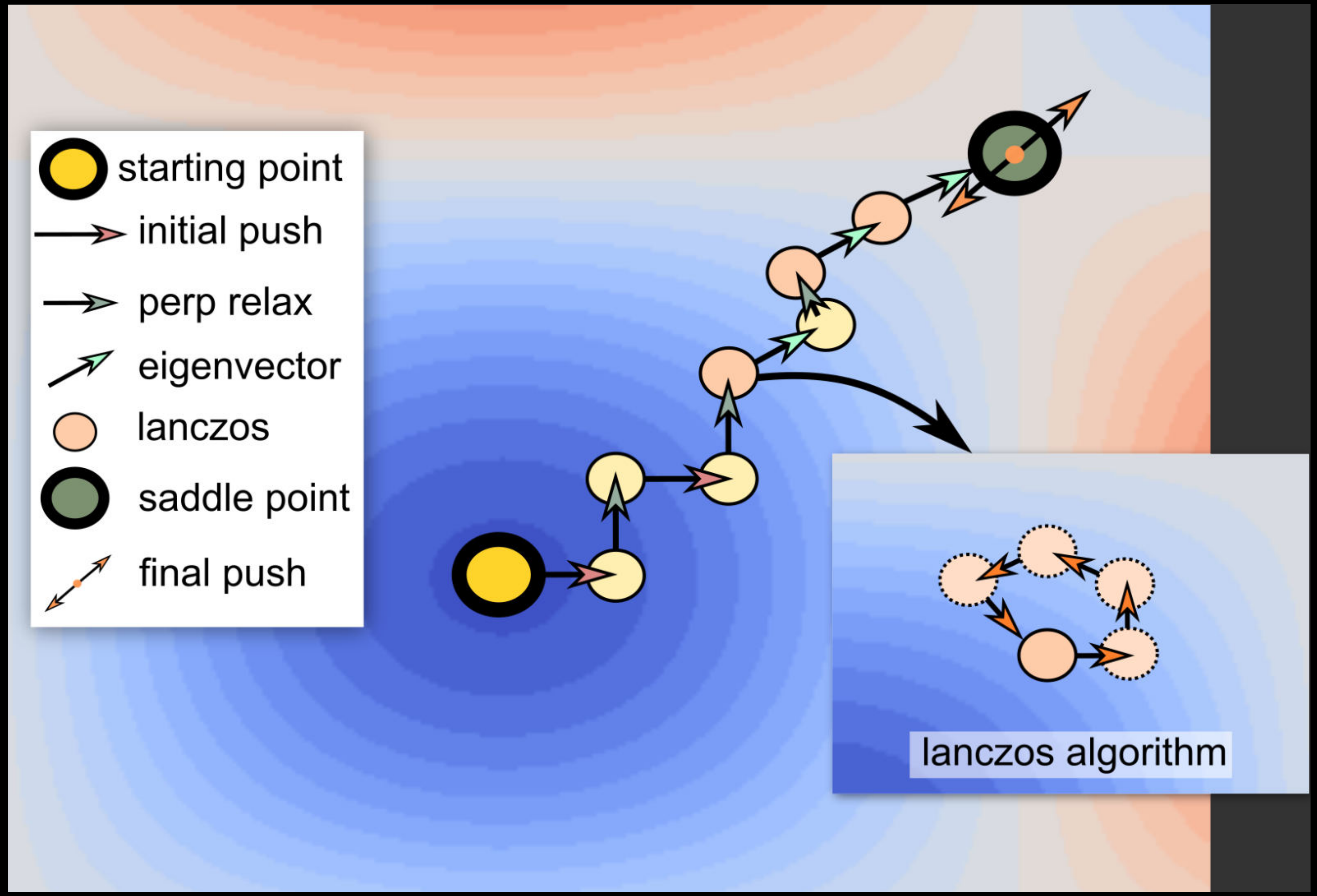
Jonsson

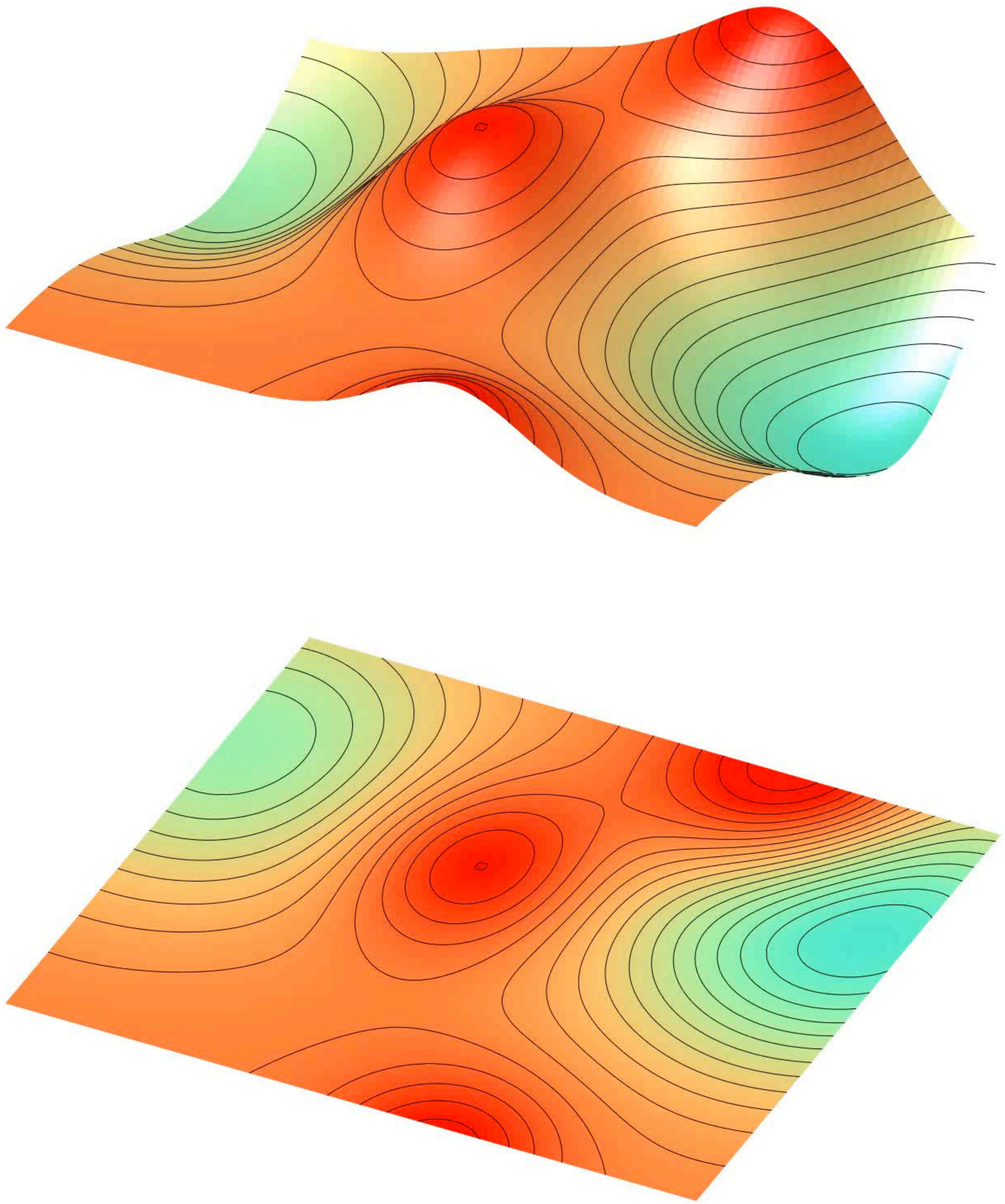
- Initial and final state
- For the saddle you need intermediate points
  - Not trivial to figure out how many
  - Cumbersome for non-trivial MEPs (more than one saddle point)
- Not easy integration in blind searches/ HTP-like searches even in the case of dimer-NEB. (“ouï-dire”dimer-NEB more unstable than ARTn, long-lasting debate between Mousseau and Jonsson)
- If you have already an idea of where the saddle point should be, you still need intermediate points -> expensive for low gain

Activation Relaxation Technique

Normand Mousseau

- Push direction
- Push
- Estimation of the Hessian (lowest “meaningful” eigenvector)
- Lowest eigenval <0 -> follow smoothed eigenvector
- Perpendicular relaxations





Courtesy Antoine Jay

Activation Relaxation Technique (ARTn)

Previous implementations

- ART moves -> Energy force engines called from inside
- ARTn-LAMMPS also implemented in k-ART
- ARTn-VASP too heavy (Nicolas Salles) (no re-use of rho, V)
- ARTn-QE* previous version -> re-use of rho and V (Antoine Jay and Nicolas Salles) -> prove of concept ARTn versatility and competitiveness against string approaches in the context of first-principle energy and force engines.
- ARTn, r-ART, d-ART… hard-coded integration of too many calculation options

** A. Jay et al, Journal of Chemical Theory and Computation 16 (10), 6726-6734 (2020)*

Activation Relaxation Technique

Previous implementations

- ART_move(R[out])
- Call Lanczos(R[inout])
  - Call engine(R[in], forces[out])
- Call ART_Perp_relax(R[inout])
  - Call engine(R[in], forces[out])

Refactoring and improvements

Back to the fundamentals

- A “jump” direction guess (default is random)
- Push
- Evaluating lowest eigenvector of the Hessian
- Following (“smoothed”) eigenvector
- Last push

Reverting the algorithm like a sock

The Energy-Force engine performs a kind of constrained-minimization

The plugin acts on the engine forces -> R are replaced by an appropriated F

Engine (at each step)

- Call engine_forces(forces[out])
- Call pART(R[in], forces[in],forces_pART[out])
  - Choose “move” type
    - Lanczos(forces_pART[inout])
    - Push(forces_pART[out])
    - Perp_projection(forces_pART[inout])
- Engine_move(forces_pART[in])

Key needs

- Full control on the relaxation algorithm (FIRE for instance)
- In principle, only one Interface routine -> one patch point

Advantages

- Easy maintenance;
- Easier porting;
- Fully compatible with improvements to rho, fft, engine parallelisation, interpolation and re-use of wfc, rho and/or V in relaxation/MD;
- Simpler for users as it gives the impression to still run “only” the engine;
- Tutorials might be focussed on ART and not on how-to-run a specific engine;
- Fully compatible with engine HTP scripts, post-processing and pre-processings
- In systems for which the position of the saddle is predictable at a glance -> refine saddle VERY FAST

Implementation details

Lanczos

Miha Gunde

- Iterative algorithm, obtains extremum eigenvalue and corresponding vector.
- Single step: matrix-vector, generation of next vector, diagonalization of a matrix

Matrix-vector: $\underbrace{H\, d\mathbf{R}}_{A|v\rangle} = -\underbrace{d\mathbf{F}}$

We can compute

The very first vector is random normalised displacement: $|v_0\rangle = d\mathbf{R}_0$

Next vectors:

$$|v_{i+1}\rangle = A|v_i\rangle - \alpha_i|v_i\rangle - \beta_i|v_{i-1}\rangle - \sum_{j=0}^{i}\langle v_{i+1}|v_j\rangle\langle v_j|$$

$$M = \begin{pmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \beta_3 & & \\ & \beta_3 & \alpha_3 & \beta_4 & \\ & & & \cdots & \\ & & \beta_{m-1} & \alpha_{m-1} & \beta_m \\ & & & \beta_m & \alpha_m \end{pmatrix}_{m \times m}$$

Loop until lowest eigenvalue converges $\quad M|e_M\rangle = \lambda_R|e_M\rangle$

Store all generated vectors in a matrix, needed for reconstructing H eigenvector.

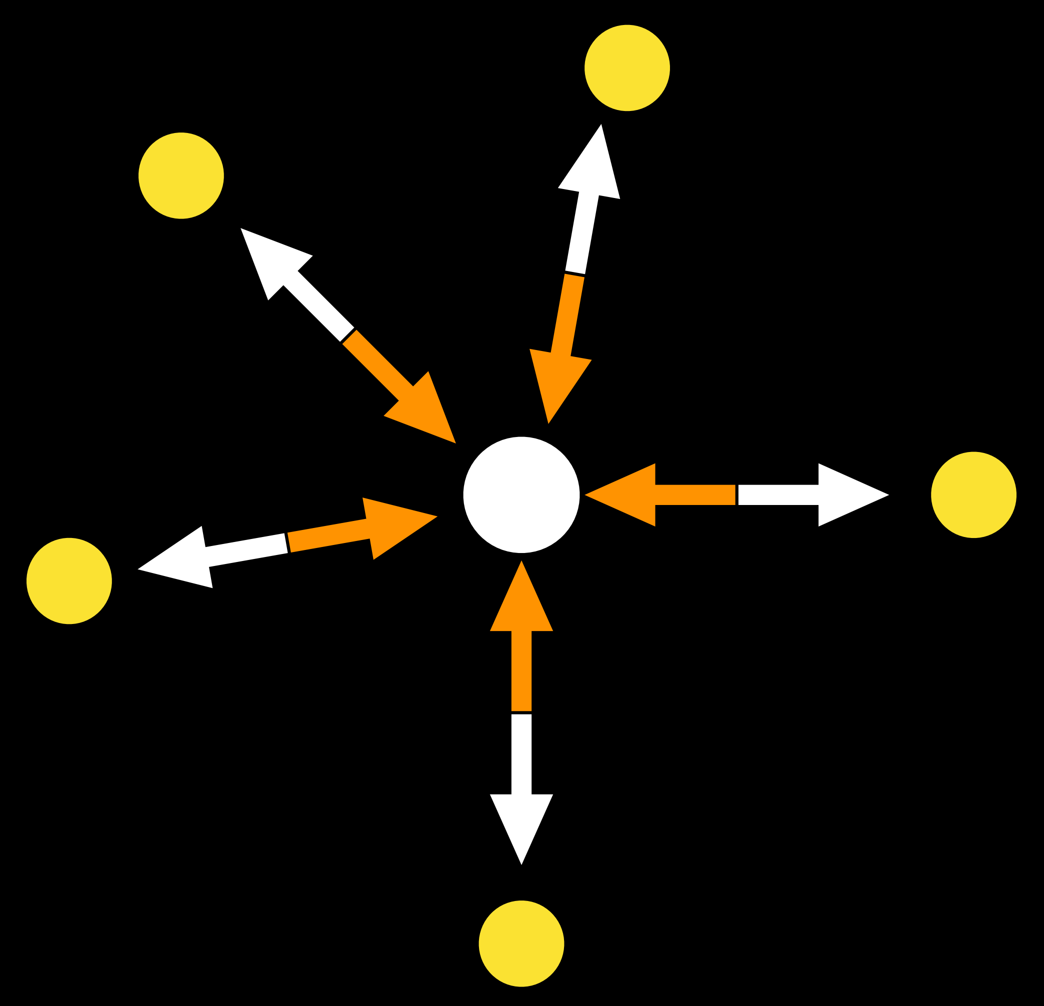
$$|x\rangle = V|e_M\rangle$$

C. Lanczos, Journal of Research of the National Bureau of Standards. 45 (4): 255–282, 1950
I. U. Ojalvo, M. Newman, AIAA Journal 8:7, 1234-1239, 1970

In practice

- Receive force corresponding to current positions;

- Calculate the next Lanczos vector $d\mathbf{R}_{i+1}$;
  - Store the needed factors α_i and β_i for the matrix M ;
  - Diagonalize M with current size $(i+1) \times (i+1)$ (lapack);
  - If lowest eigenvalue converged: $dR_{i+1} = \boxed{-\sum_{j=0}^{i} dR_j} + |x\rangle$
  - Else: $dR_{i+1} = \boxed{-\sum_{j=0}^{i} dR_j} + |v_{i+1}\rangle$
  - Based on result of Lanczos, change internal flags
- Transform. $d\mathbf{R}_{i+1}$ into a force ;
- Return force, corresponding to the desired move.



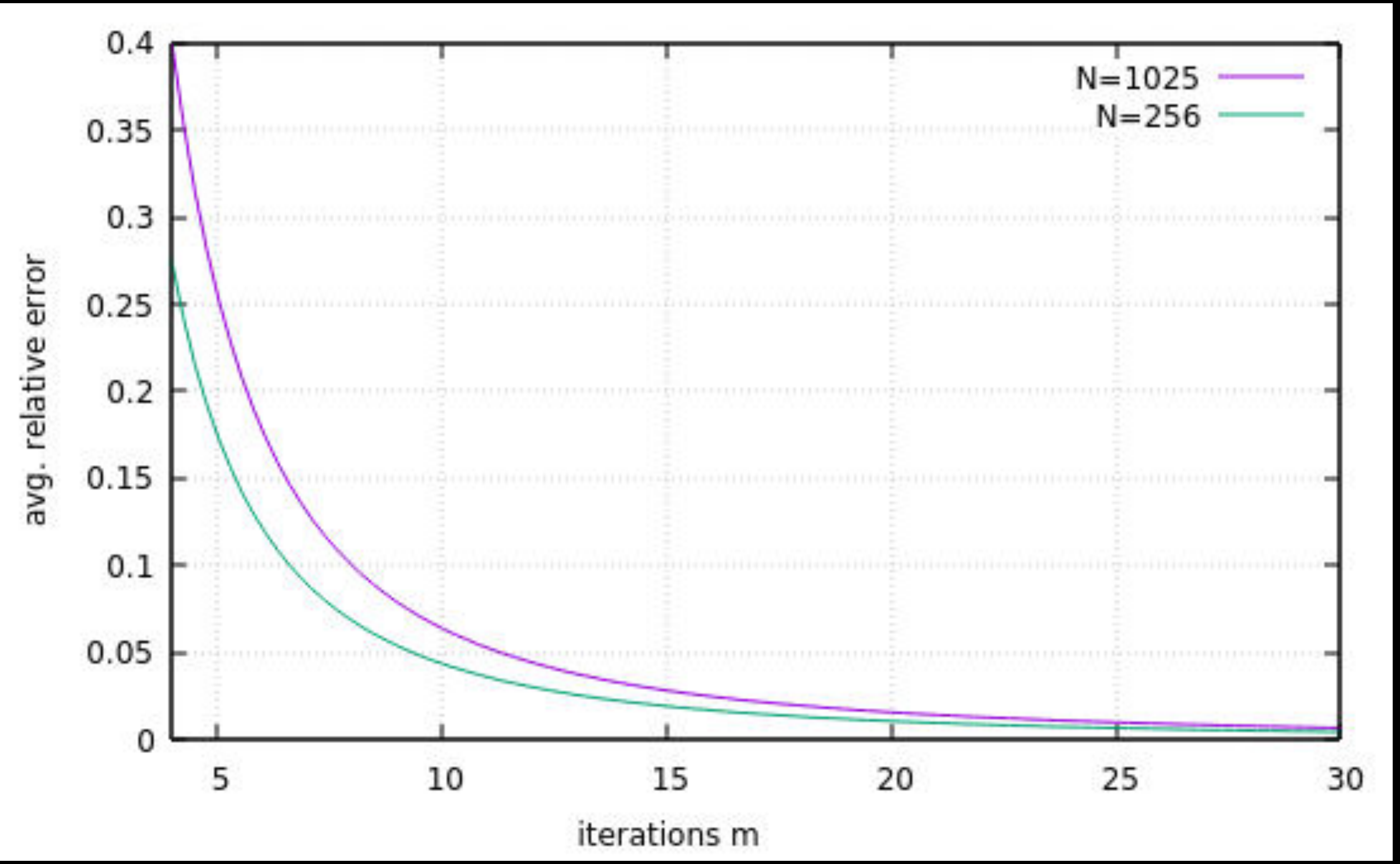
Further changes

Lanczos steps (iterative convergence criteria)

- Original ARTn implementation used a fixed number of Lanczos steps (m=15), following Ref. [1] tests.

- Theoretically Lanczos average relative error is upper bounded[2] by:

$$0.1\left(\frac{\ln(3N)}{m}\right)^2$$



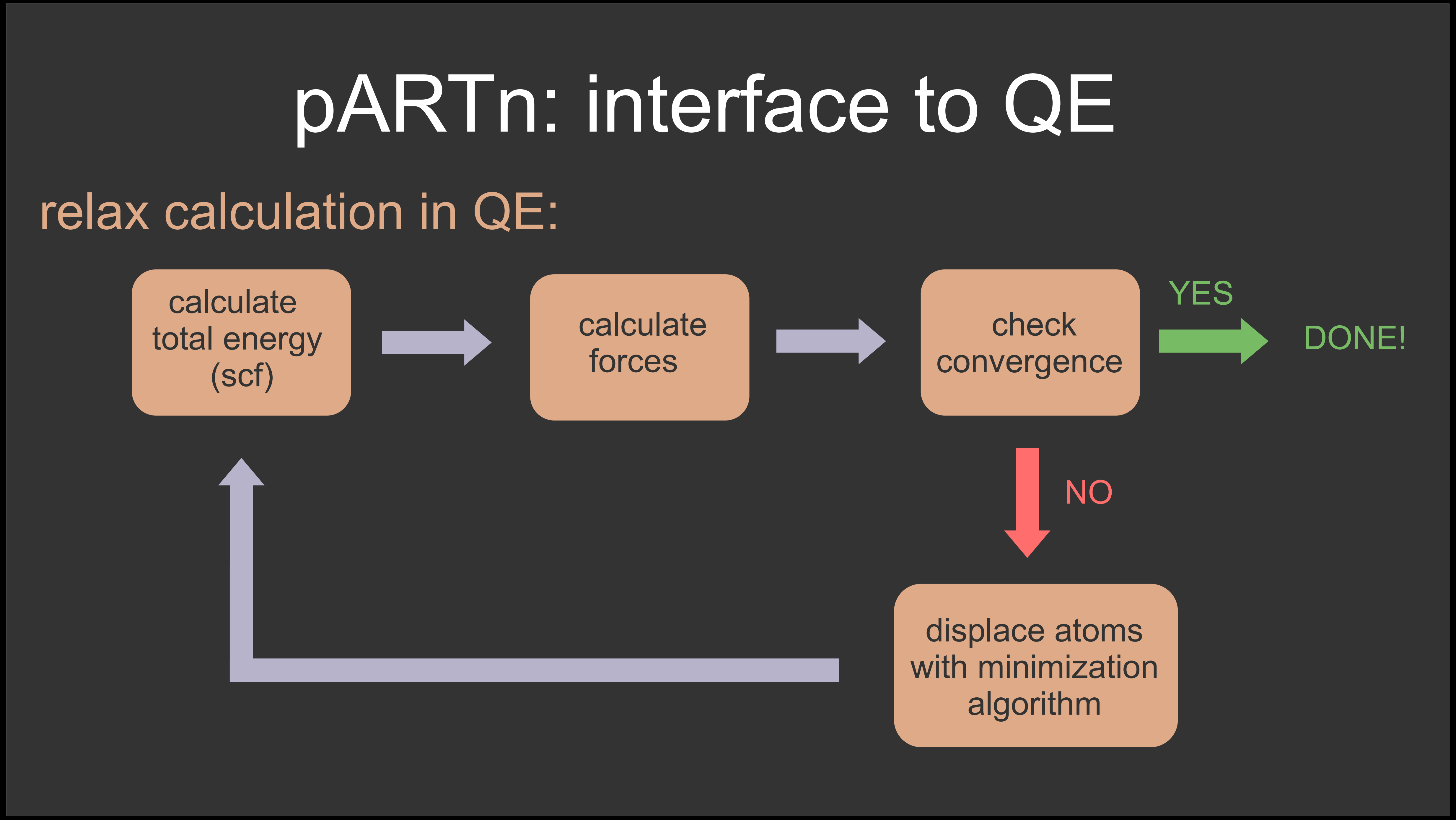
- + re-use of previous eigenvector as $|v_0\rangle$

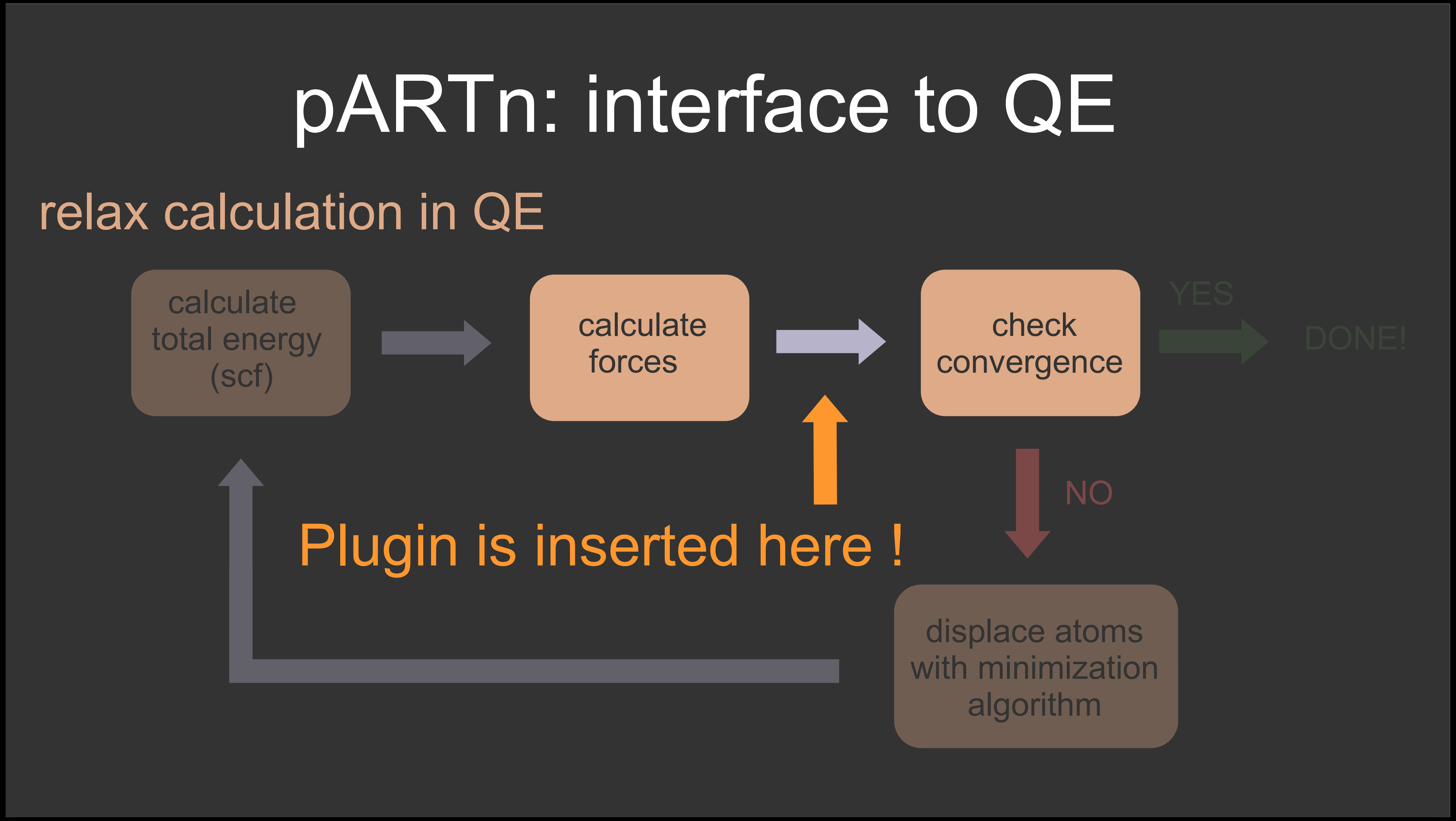
- The number of F evaluations reduced!

[1] M.-C. Marinica, F. Willaime, and N. Mousseau, Phys. Rev. B 83, 094119, 2011
[2] J. Kuczynski, H. Wozniakowski, Siam J. Matrix Anal. 13, 1094, 1992

QE interface

Matic Poberznik





Plugin in

- Number of atoms (size of arrays)
- **Forces**
- **FIRE variables and parameters (dt,v, alpha)**
- **Convergence thr**
- Fixed atoms from engine
- + structure
- **Root and engine_comm**
- **IO Paths**

Plugin out

- ***“Forces”***
- **FIRE variables and parameters**
- **Convergence thr**

Few technicalities

- Compiled as a library, then included in the paths of make.inc
- ARTn input parameters read from file
- Only root or ionode call pARTn
- A single call to the driver interface routine after the calculation of engine total force

```
LOGICAL :: lconv
!
! ARTn convergence flag
!
lconv = .false.
!
IF ( ionode ) THEN
   CALL artn(force,etot,epsf,nat,ityp,atm,tau,at,alat,istep,if_pos,vel,dt,fire_alpha_init,lconv,prefix,tmp_dir)
ENDIF
IF ( lconv ) THEN
   WRITE (*,*) "ARTn calculation converged, stopping"
   STOP 1
END IF
```

From positions (R) to Forces (F)

Now exploiting FIRE, but extendable to steepest decent or quick-min

Push and Lanczos

$$\mathbf{F}(\mathbf{x}(t)) = \frac{\Delta \mathbf{R} m}{\Delta t^2}$$

FIRE parameters:

$$\mathbf{v}(t) = 0$$
$$\alpha = 0$$
$$N_{P>0} = 0$$

Perpendicular relaxation

$$\mathbf{F}(\mathbf{x}(t)) = \mathbf{F}(\mathbf{x}(t)) - \mathbf{F}_{\text{para}}(\mathbf{x}(t))$$

FIRE parameters:

first step:

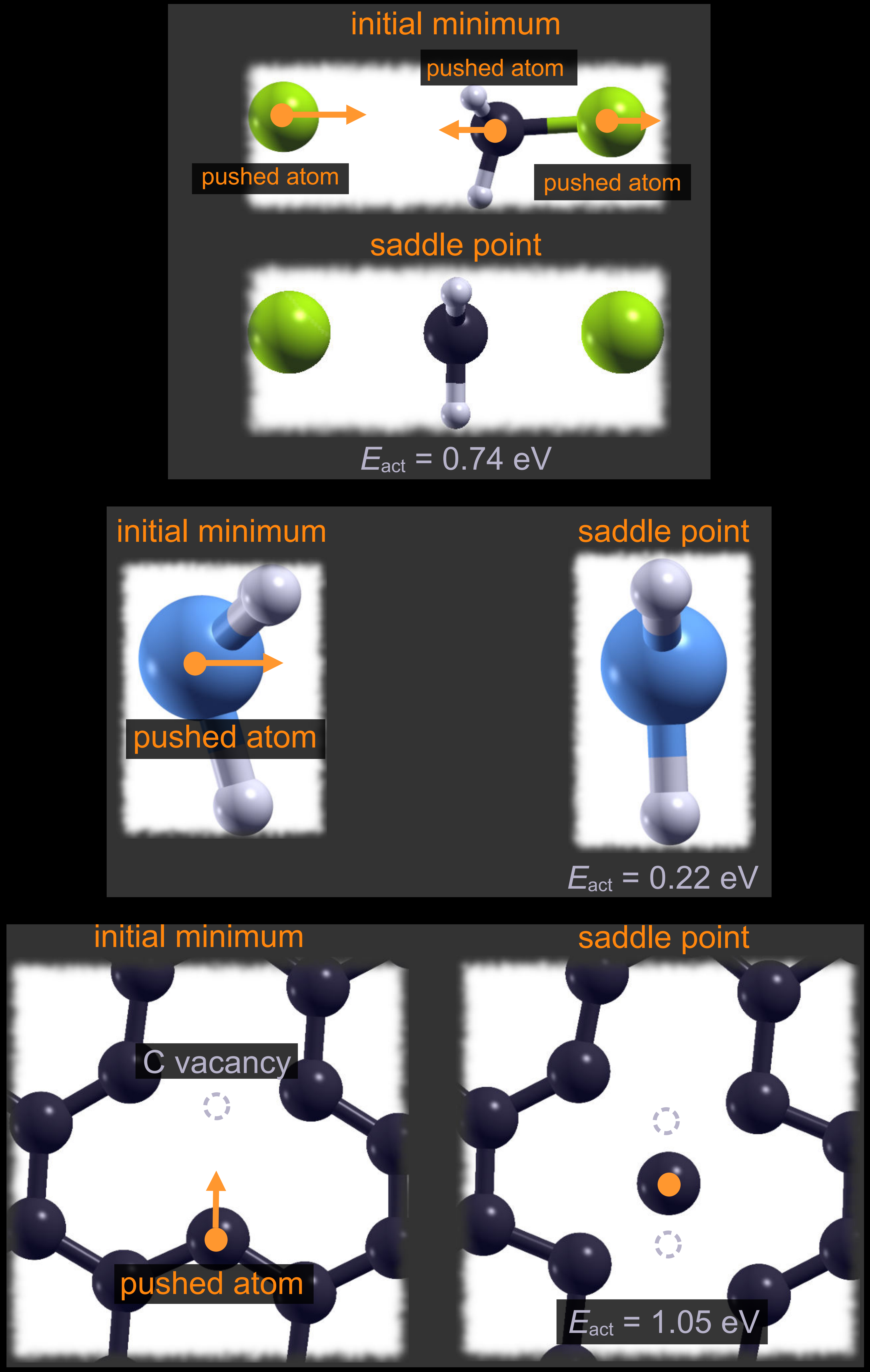
$$\mathbf{v}(t) = 0$$
$$\alpha = \alpha_{\text{init}}$$
$$\Delta t = \Delta t_{\text{init}}$$

other steps:

$$\mathbf{v}(t) = \mathbf{v}(t) - \mathbf{v}_{\text{para}}(t)$$

Few examples
(Serious benchmark in progress)

	Cl transfer	NH3 inversion	Li@ graphene
NEB	172	68	38 (freezing option)
pARTn	60	33	83



Acknowledgements

MAMMASMIAS *relaxed* Consortium

Dr. Anne Hemeryck
Dr. Nicolas Richard
Prof. Normand Mousseau
Prof. De Gironcoli
Dr. Antoine Jay
Thomas Jarrin
Ruggero Lot
Dr. Gabriela Herrero

MAGNELIQ
No 899285

European Commission