Adaptively Restrained Molecular Dynamics in LAMMPS

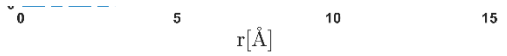
By
Semeho EDORH

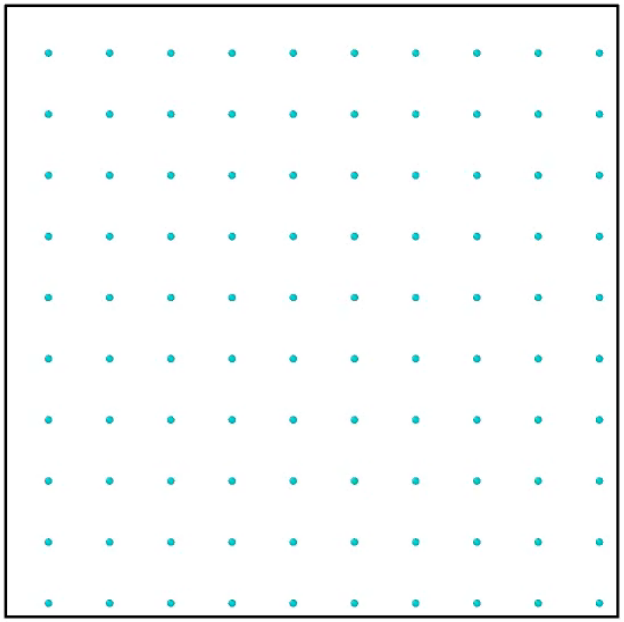
Svetlana ARTEMOVA, Semeho EDORH, Dmitriy MARIN, Krishna SINGH, *Zofia TRSTANOVA and Stephane REDON*

Classical MD simulations

- Generate a temporal evolution (Positions and momenta) of a particle system
- Retrieve macroscopic information by averaging properties along trajectories
- Expensive task due to force computations, especially for **pairwise non-bonded forces**
- Most interaction forces are computed **from the scratch** at each timestep.

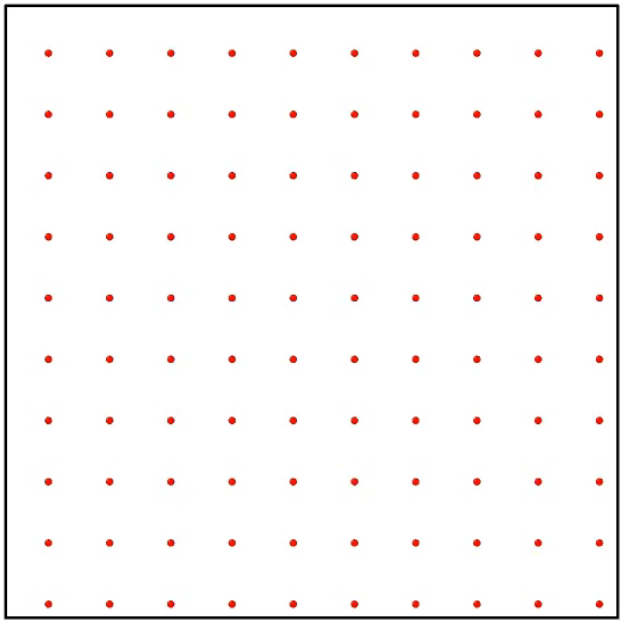
$$\begin{cases} \dfrac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}t} = \nabla_{\boldsymbol{p}} H \\ \dfrac{\mathrm{d}\boldsymbol{p}}{\mathrm{d}t} = -\nabla_{\boldsymbol{q}} H \end{cases}$$

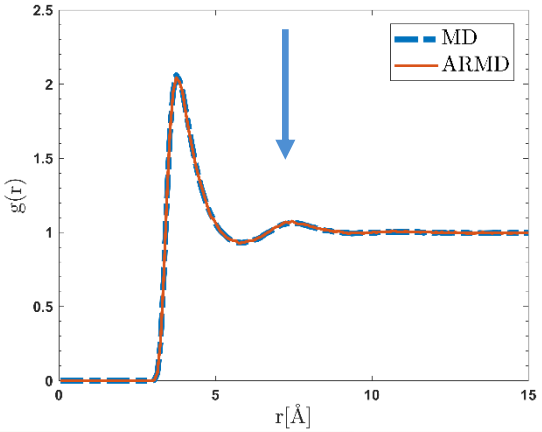




Adaptively Restrained MD simulations

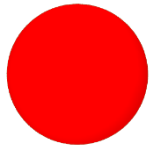
- **ARMD** freezes at each timestep some particles
- Although the phase space is explored differently, several properties are preserved
- Computationally **less expensive** when interaction forces depend upon **interatomic distances**
- Interaction forces can be **incrementally updated** at each timestep.





ARMD: How it works?

- **ARMD** switches on/off particles positional degrees of freedom based on kinetic energy
- A particle is **restrained** when its instantaneous kinetic energy is below ε_r
- Particles may gain or lose kinetic energy
- And become **active** when their kinetic energy exceeds ε_r
- Like MD, ARMD can be expressed with Hamiltonian formalism... but with a modified Hamiltonian.



[Artemova and Redon 2012]

ARMD: The Hamiltonian

- MD Hamiltonian sums kinetic energy and interaction potential

$$H(\boldsymbol{q},\boldsymbol{p}) = \frac{1}{2}\boldsymbol{p}^T M^{-1}\boldsymbol{p} + V(\boldsymbol{q})$$

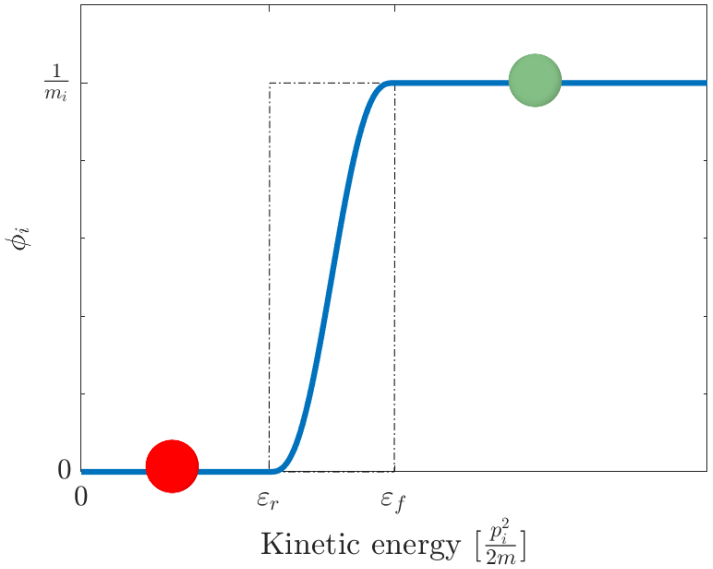
- In ARMD, the kinetic term is slightly modified

$$H_{AR}(\boldsymbol{q},\boldsymbol{p}) = \frac{1}{2}\boldsymbol{p}^T \phi(\boldsymbol{q},\boldsymbol{p})\boldsymbol{p} + V(\boldsymbol{q})$$

- The diagonal element $\phi_i(\boldsymbol{q}_i,\boldsymbol{p}_i)$ controls the state of a particle i

$$\phi_i(\boldsymbol{q}_i,\boldsymbol{p}_i) = \frac{1}{m_i}\left(1 - \rho_i\left(\frac{p_i^2}{2m_i}\right)\right)$$

- In order to allow smooth transitions and stable simulations, we introduce a second threshold ε_f



ARMD: Some properties

- ARMD can sample both NVE and NVT ensembles

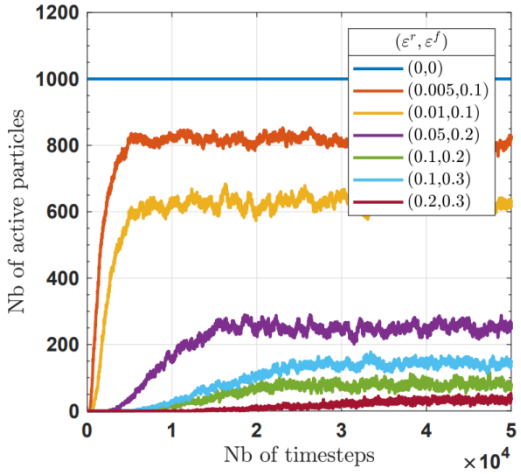
$$\begin{cases} \dfrac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}t} = \nabla_{\boldsymbol{p}} H_{AR} \\ \dfrac{\mathrm{d}\boldsymbol{p}}{\mathrm{d}t} = -\nabla V - \gamma \nabla_{\boldsymbol{p}} H_{AR} + \sigma \dfrac{\mathrm{d}\boldsymbol{W}}{\mathrm{d}t} \end{cases}$$

[Trstanova 2016]

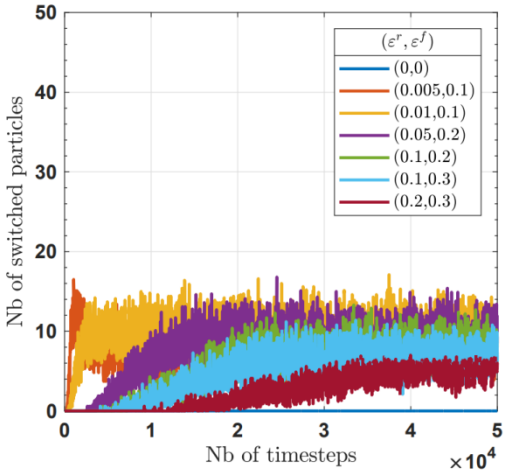
- Ensemble averages are preserved for position-dependent properties

$$\langle A \rangle_{H_{AR}} = \langle A \rangle_{H}$$

- The number of active particles is governed by the choice of $(\varepsilon_r, \varepsilon_f)$



(1) Number of active particles



(2) Number of switched particles

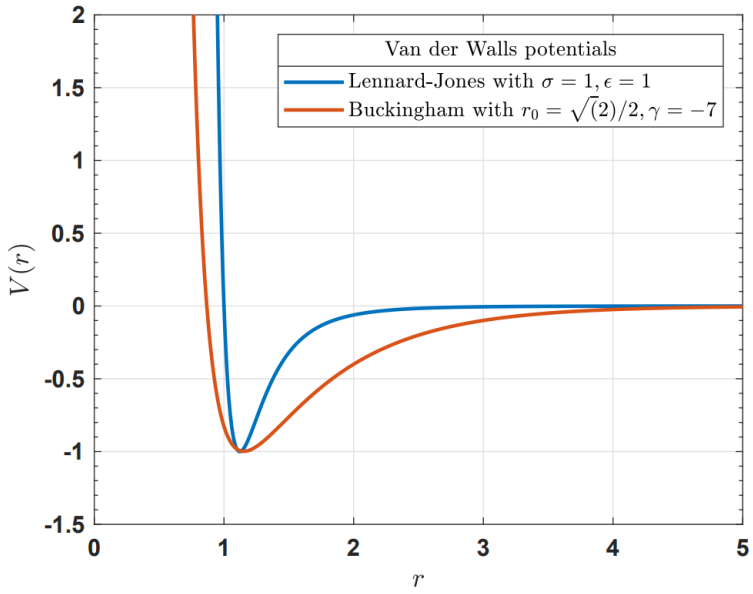
ARMD in LAMMPS

- NVE ensemble: **Fix_ARMD_NVE.cpp**
- NVT Langevin: **Fix_ARMD_Langevin.cpp**
- Integrator: **ARMD.cpp**
- Incremental algorithms for non-bonded pairwise interactions
  - ✓ General treatment for short-range interactions
  - ✓ Specialized algorithms for electrostatics
- ARMD in KOKKOS package

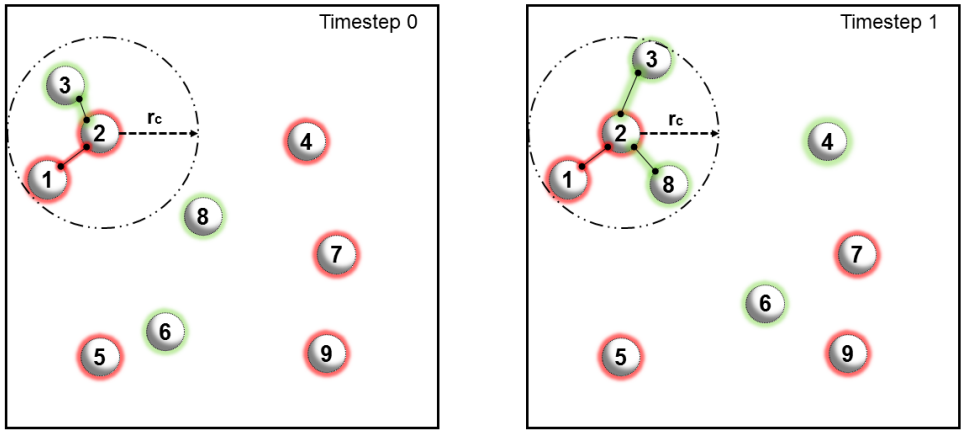
Short-range potentials

- **Several pair_***.cpp** in LAMMPS
- Van der Walls potentials (Lennard Jones, Buckingham, etc.)
- Can be truncated beyond a certain cutoff
- Efficiently computed with **Neighbor Lists**

$$V_{short} = \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j>i}}^{N} V_s(r_{ij})$$



Incremental Algorithms for short-range interactions



[Edorh 2018]

- **Restrained** Interactions between **restrained-restrained** particles
- **Active** Interactions which involve at least one active particle (**active-restrained, active-active**)
- Force decomposition:

$$\boldsymbol{F}_i = \boldsymbol{F}_i^{\text{active}} + \boldsymbol{F}_i^{\text{restrained}}$$

[Singh 2017]

Active Neighbor lists

- Active particles: $\boldsymbol{F_i = F_i^{active} + 0}$
- Restrained particles: $\boldsymbol{F_i = F_i^{active} + F_i^{restrained}}$
- $\boldsymbol{F_i^{restrained}}$ can be stored
- $\boldsymbol{F_i^{active}}$ can be efficiently evaluated with specialized Neighbor lists:

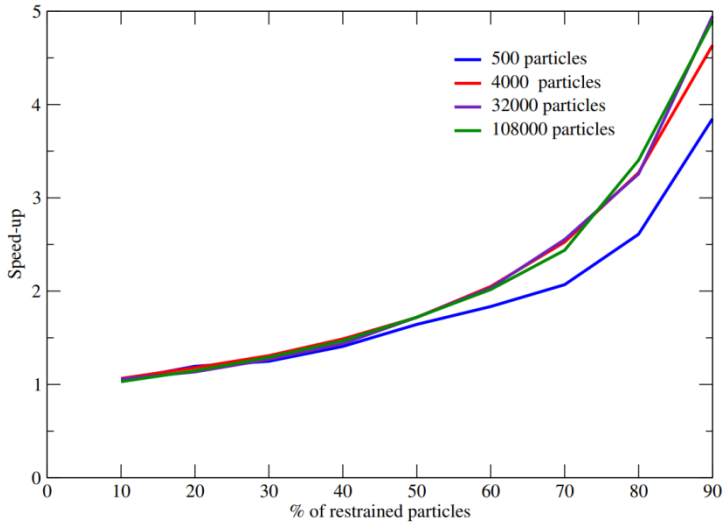
Active Neighbor lists (ANL)

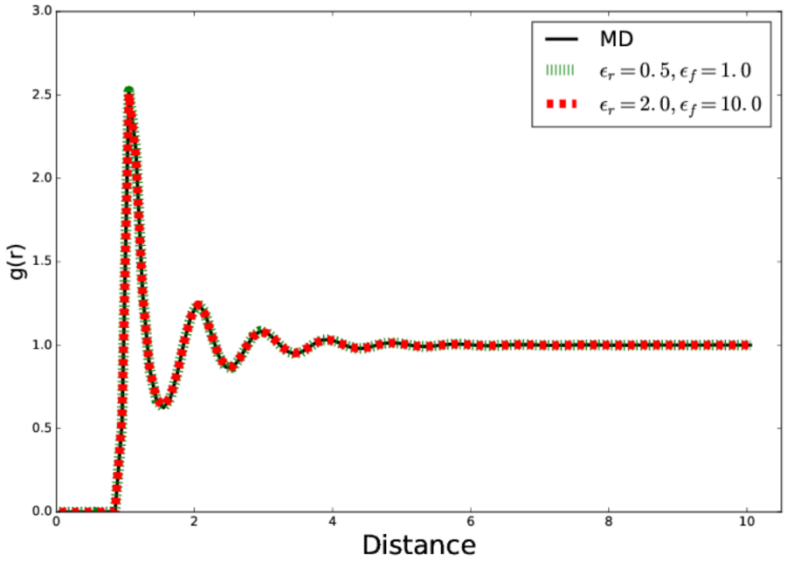
- Derived from built-in Full Neighbor lists
- Newton's 3rd Law can be enabled by **ANL**
- **ANL +ARMD** can be used without modifying any **pair_***.cpp**
- **Switched particles** can be treated with few operations.

[Singh 2017]

Benchmark: NVT simulation of LJ particles

- **FCC** lattice of density 0.8442
- Truncated **Lennard-Jones** potential
- Cutoff distance $r_c = 2.5\sigma$
- Neighbor list updated each 20 timesteps

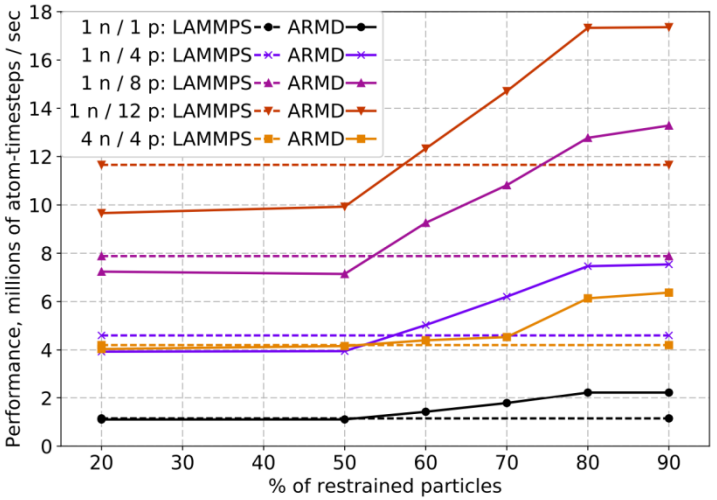


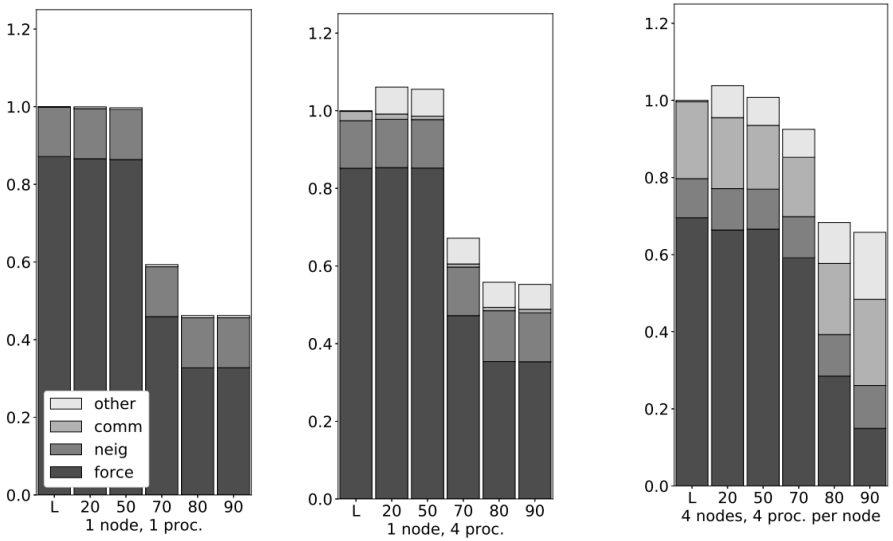


[Singh 2017]

Benchmark: NVT simulation of LJ particles + MPI

- **MPI** enabled
- **864K** particles
- 8/16 CPUs Intel Xeon E5540 per node
- Gigabit Ethernet network





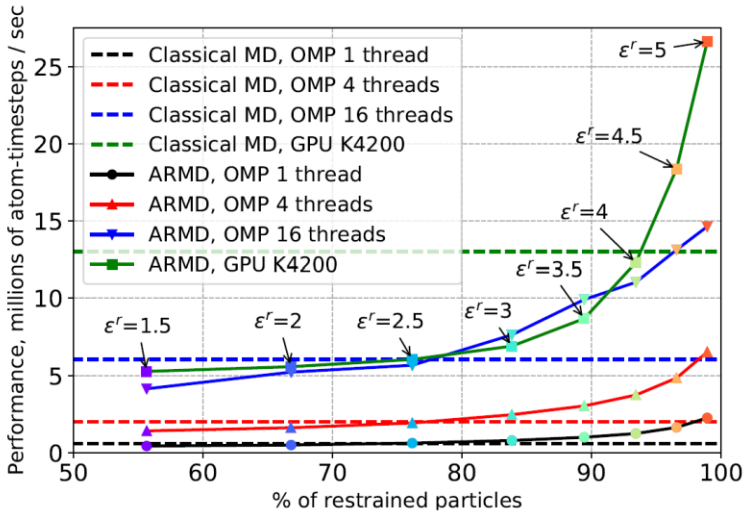
Breakdown of wall-clock time for 1 and 4 nodes with 4 processes per each node normalized by LAMMPS timing for different percentage of restrained particles

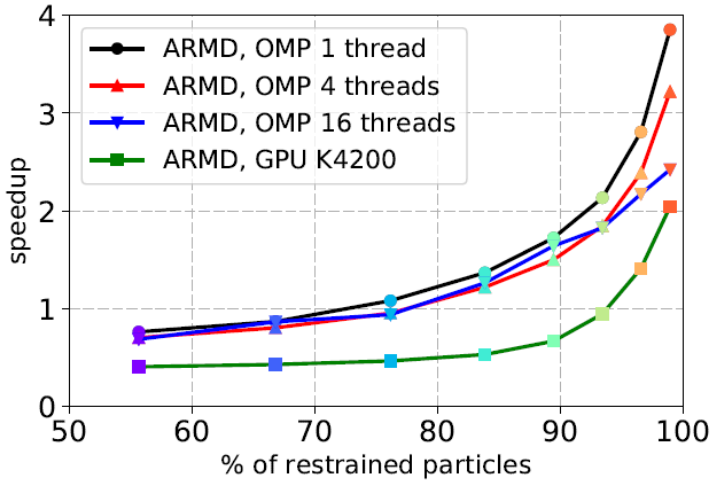
Other – Load balancing, ARMD routines for switched particles (ANL and force computations of switched particles), position & momenta update; comm – communications; neig – neighbor list construction; force – force computation

[Marin 2017]

ARMD and KOKKOS

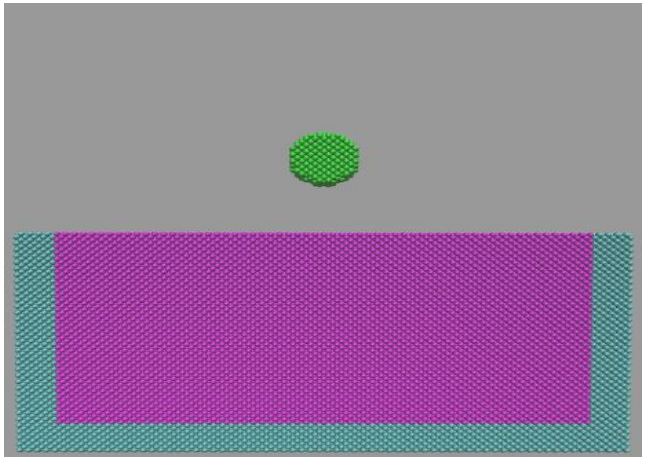
- **FCC** lattice of density 0.8442
- Truncated **Lennard-Jones** potential
- Cutoff distance $r_c = 2.5\sigma$
- Neighbor list updated each 20 timesteps

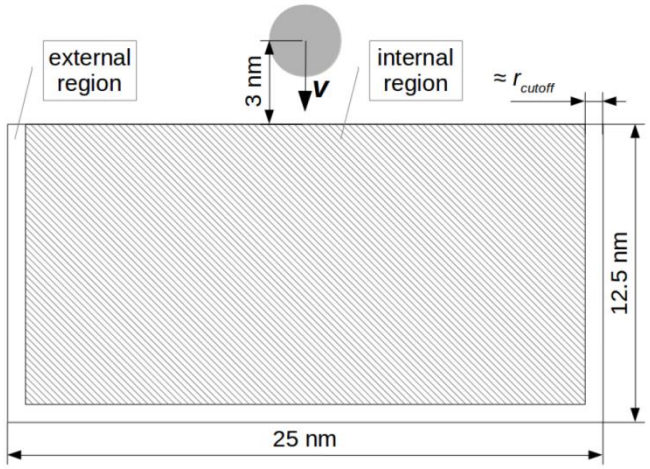




Benchmark: nano-projectile impact on surface

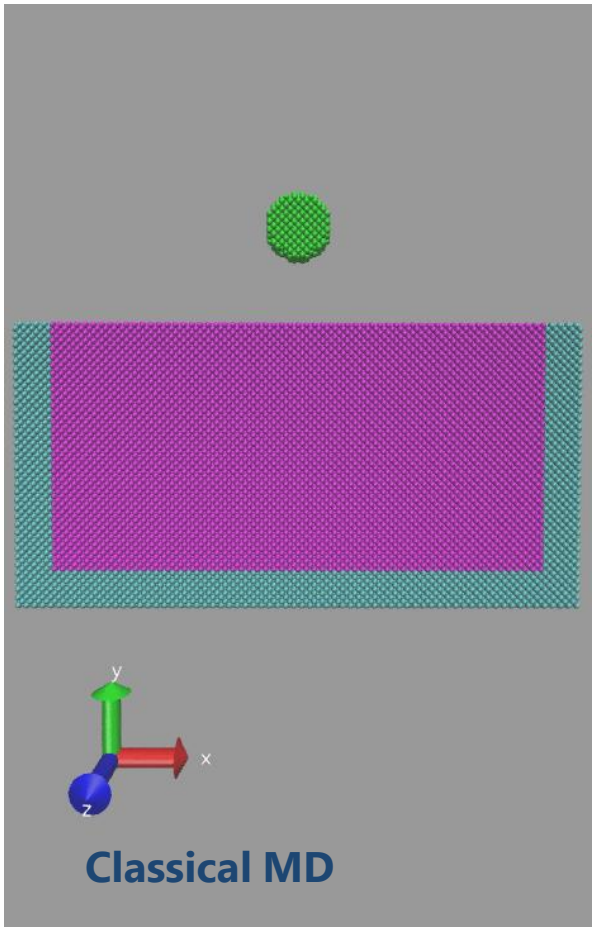
- **Nano-projectile**: Cu_{1000} (r = 1.415 nm)
- **Substrate**: Cu, **677K** atoms
  (L = 25 nm; H = 12.5 nm)
- **Nano-projectile velocity**: 5 km/sec
  (8.25 eV per atom, total: 8.25 keV)
- Truncated **Lennard-Jones** potential
- Neighbor list update each 20 timesteps



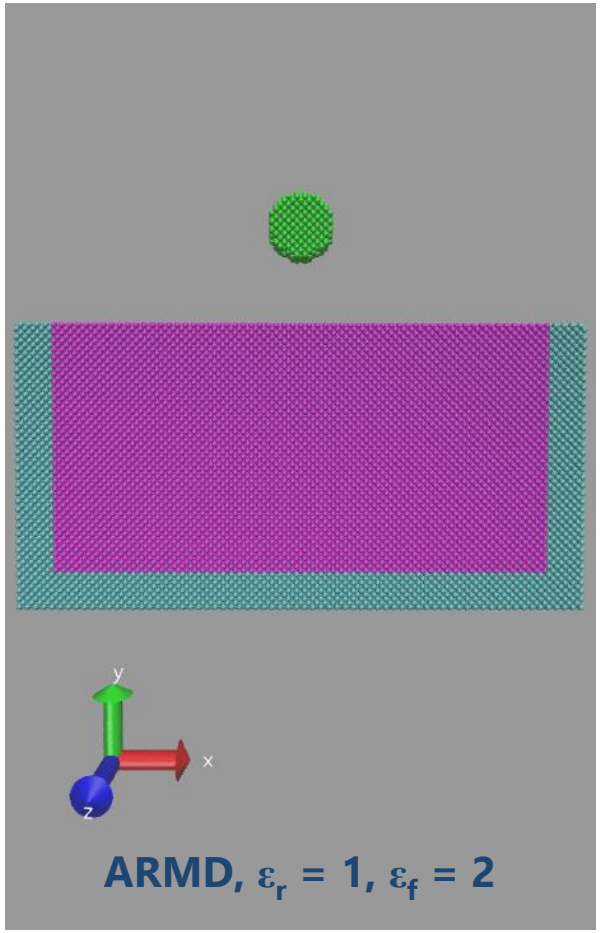


[Marin 2018]

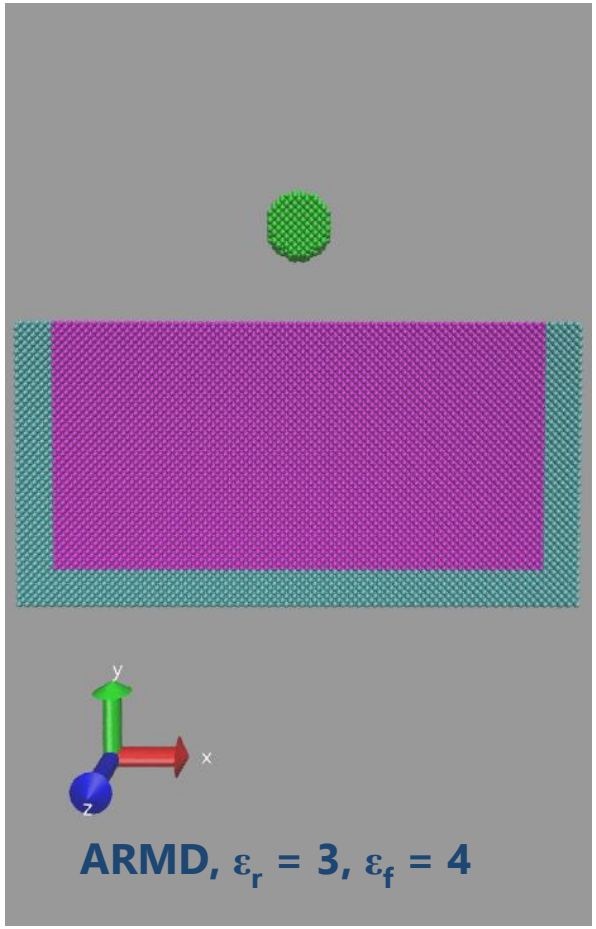
Benchmark: nano-projectile impact on surface



Classical MD



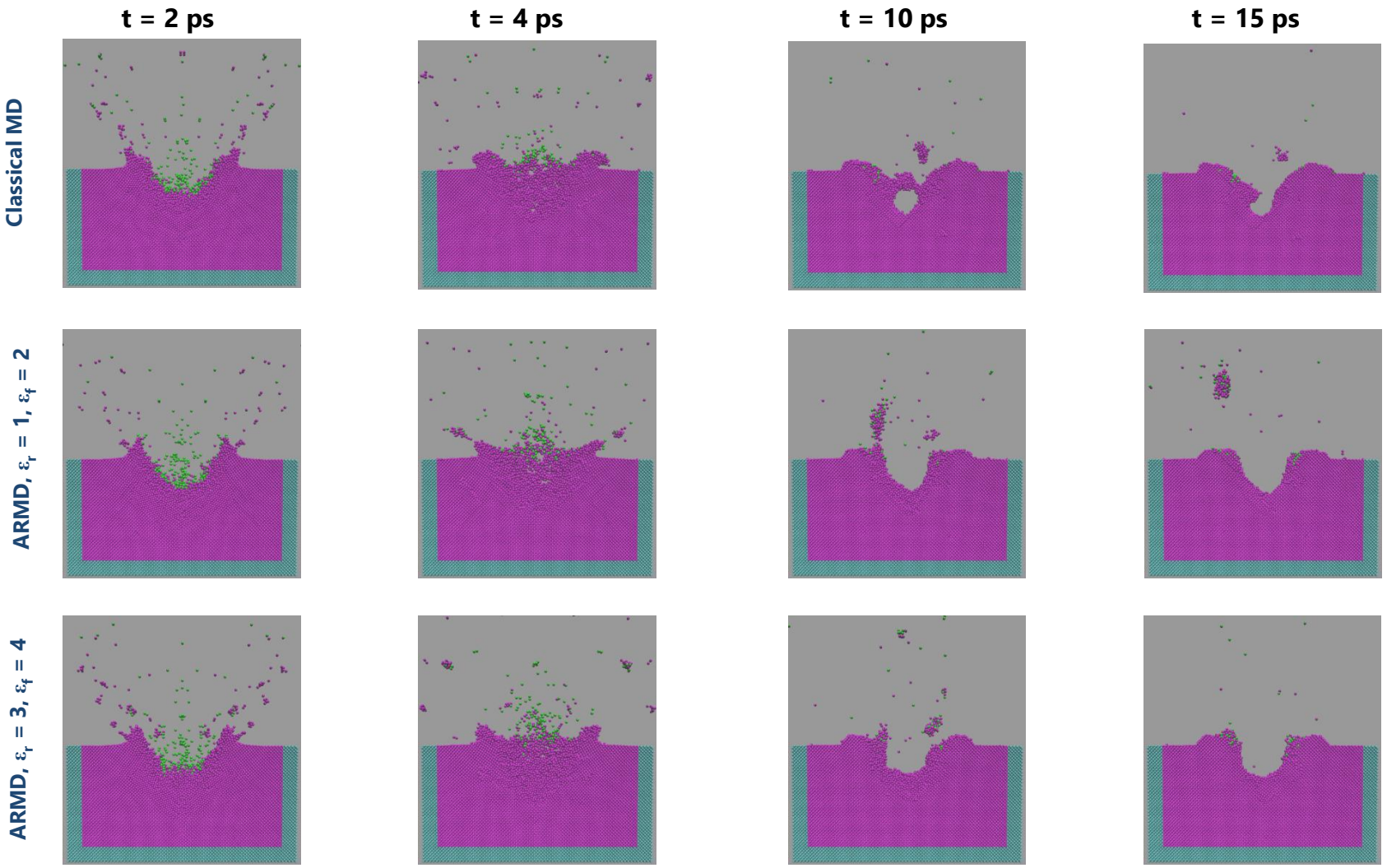
ARMD, $\varepsilon_r = 1$, $\varepsilon_f = 2$



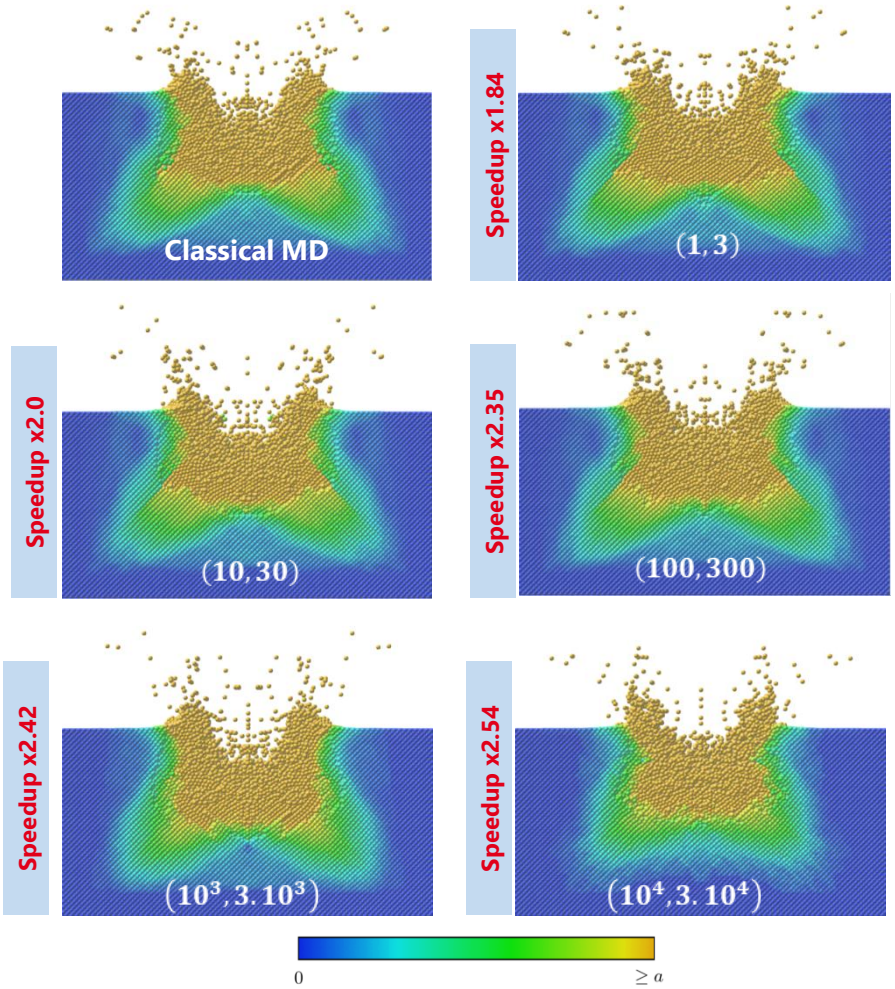
ARMD, $\varepsilon_r = 3$, $\varepsilon_f = 4$

Parameters ε_r, ε_f are in eV.
Nano-projectile: 8.25 eV per atom
Slice: 1 nm

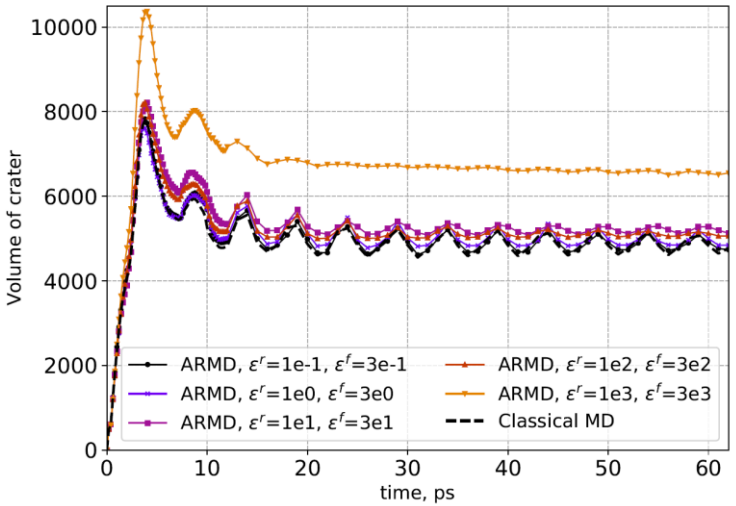
Benchmark: nano-projectile impact on surface



Benchmark: nano-projectile impact on surface



Snapshots at t=1ps of crater formation in the substrate.
Color depicts particle displacement compared to their initial state, **a** is the lattice constant, (ϵ_r, ϵ_f) are in eV.



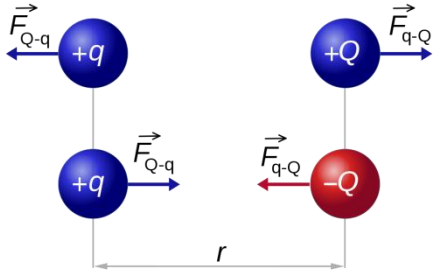
The cluster volume as a number of missing atoms in the substrate.

✓ The relative error of the calculated crater volume for ARMD compared to classical MD is less than 12% for parameters $(\epsilon_r, \epsilon_f) = (0.1, 0.3), (1, 3), (10, 30), (100, 300)\ eV$

[Marin 2018]

Incremental algorithms for Electrostatics

- Long-range interactions (Electrostatics) are extremely expensive
- Electrostatics are efficiently handled by splitting short- and long- range contributions:
  Pair.cpp + **kspace.cpp**
- They can also benefit from adaptive restraints
- **Short-range terms are enhanced by ANLs**
- **Long-range terms require more specific algorithms**
- Improved algorithms
  - ✓ **Ewald summation**
  - ✓ **P3M**
  - ✓ **Meshed continuum method**



$$|\vec{F}_{Q-q}| = |\vec{F}_{q-Q}| = k\frac{|q \times Q|}{r^2}$$

$$V_{elec} = \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j>i}}^{N} \frac{1}{4\pi\epsilon_r} \frac{q_i q_j}{r_{ij}}$$

Ewald summation

- Coulomb point charges are smeared with Gaussians
- Forces are split into Real- and Fourier- spaces contributions: $\boldsymbol{f}_i^{short} + \boldsymbol{f}_i^{long}$
- Real space : $\boldsymbol{f}_i^{short} = \frac{1}{4\pi\epsilon_0} \sum_{j \in \mathcal{N}(i)} q_i q_j \frac{\boldsymbol{x}_{ij}}{|\boldsymbol{x}_{ij}|^2} \left[\frac{2}{\sigma\sqrt{2\pi}} \exp\left(\frac{|\boldsymbol{x}_{ij}|^2}{2\sigma^2} \right) - \operatorname{erfc}\left(\frac{|\boldsymbol{x}_{ij}|}{\sigma\sqrt{2}} \right) \right]$
- K-space : $\boldsymbol{f}_i^{long} = \frac{1}{2V\epsilon_0} \sum_{\boldsymbol{k} \neq 0} \frac{\exp(-\sigma^2 k^2/2)}{k^2} \left(\overline{S(\boldsymbol{k})} \frac{\partial S(\boldsymbol{k})}{\partial \boldsymbol{x}_i} + S(\boldsymbol{k}) \frac{\partial \overline{S(\boldsymbol{k})}}{\partial \boldsymbol{x}_i} \right)$

$$= \frac{q_i}{V\epsilon_0} \sum_{\boldsymbol{k} \neq 0} \frac{e^{-\sigma^2 k^2/2}}{k^2} \boldsymbol{k} \left(\sin(\boldsymbol{k}.\boldsymbol{x}_i) \mathbb{R}\mathrm{e}(S(\boldsymbol{k})) - \cos(\boldsymbol{k}.\boldsymbol{x}_i) \mathbb{I}\mathrm{m}(S(\boldsymbol{k})) \right)$$

✓ **Structure factor :** $S(\boldsymbol{k}) \equiv \sum_{j=1}^{N} q_j e^{i\boldsymbol{k}.\boldsymbol{x}_j}$

❖ **σ and k affect accuracy/speed!**

Incremental Ewald summation

- Real space : handled with **ANLs**
- Evaluation of the structure factor :

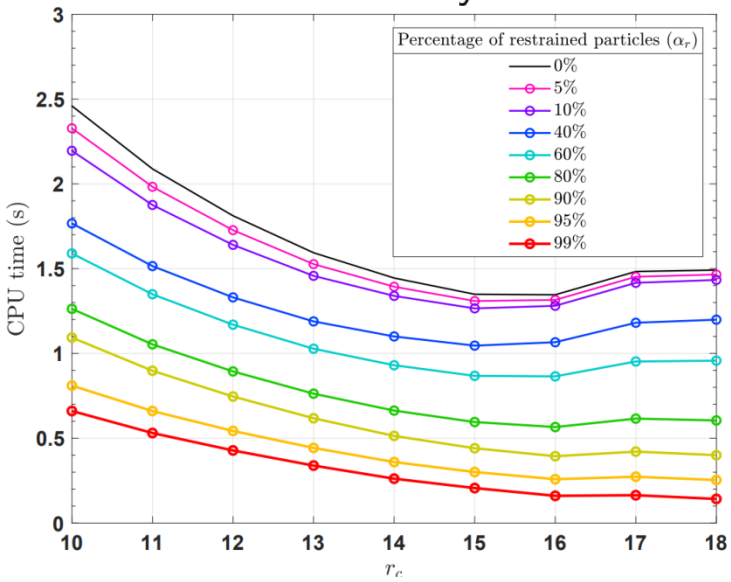
$$\begin{aligned} S(\boldsymbol{k}) &= S^{(a)}(\boldsymbol{k}) + S^{(r)}(\boldsymbol{k}) \\ &= \sum_{active} q_j e^{i\boldsymbol{k}.\boldsymbol{x}_j} + \sum_{restrained} q_j e^{i\boldsymbol{k}.\boldsymbol{x}_j} \end{aligned}$$

- Sin and Cos Terms can be saved when particle i is restrained

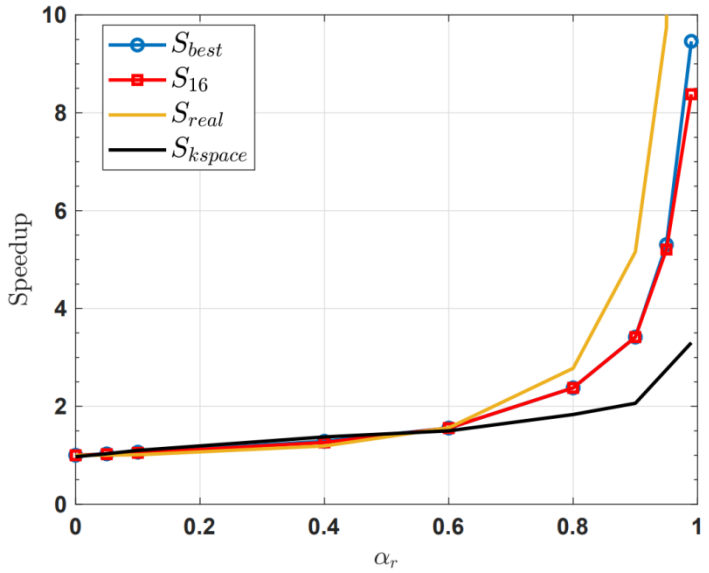
$$\begin{aligned} \boldsymbol{f}_i^{long} &= \frac{1}{2V\epsilon_0} \sum_{\boldsymbol{k}\neq 0} \frac{\exp(-\sigma^2 k^2/2)}{k^2} \left(\overline{S(\boldsymbol{k})} \frac{\partial S(\boldsymbol{k})}{\partial \boldsymbol{x}_i} + S(\boldsymbol{k}) \frac{\partial \overline{S(\boldsymbol{k})}}{\partial \boldsymbol{x}_i} \right) \\ &= \frac{q_i}{V\epsilon_0} \sum_{\boldsymbol{k}\neq 0} \frac{e^{-\sigma^2 k^2/2}}{k^2} \boldsymbol{k} \left(\sin(\boldsymbol{k}.\boldsymbol{x}_i)\, \mathbb{R}\mathrm{e}(S(\boldsymbol{k})) - \cos(\boldsymbol{k}.\boldsymbol{x}_i)\, \mathbb{I}\mathrm{m}(S(\boldsymbol{k})) \right) \end{aligned}$$

Incremental Ewald summation

- SPC/E Water : 36000 particles
- Box : 70X70X70 A^3
- Relative accuracy: 10^{-5}



CPU time per timestep for various percentage of restrained particles. Several real space cutoff were employed in order to find the best configuration. r_c is in angstrom.



Speedup as function of the proportion of restrained particles.

❖Good acceleration but still slower than P3M

Particle Particle Particle Mesh (P3M)

- Ewald summation… but long-range terms are treated on a grid
- Charges are sampled on a mesh

$$\rho_g(\boldsymbol{x}_p) = \frac{1}{h^3}\sum_{i=1}^{N} q_i W(\boldsymbol{x}_p - \boldsymbol{x}_i)$$

- Electric field/Coulomb potential is retrieved with Fast Fourier Transforms

$$\mathscr{E}(\boldsymbol{x}_p) = \overleftarrow{\mathrm{FFT}}\left[i\boldsymbol{k}\overrightarrow{\mathrm{FFT}}[\rho_g] \times \tilde{G}_{opt} \right](\boldsymbol{x}_p)$$

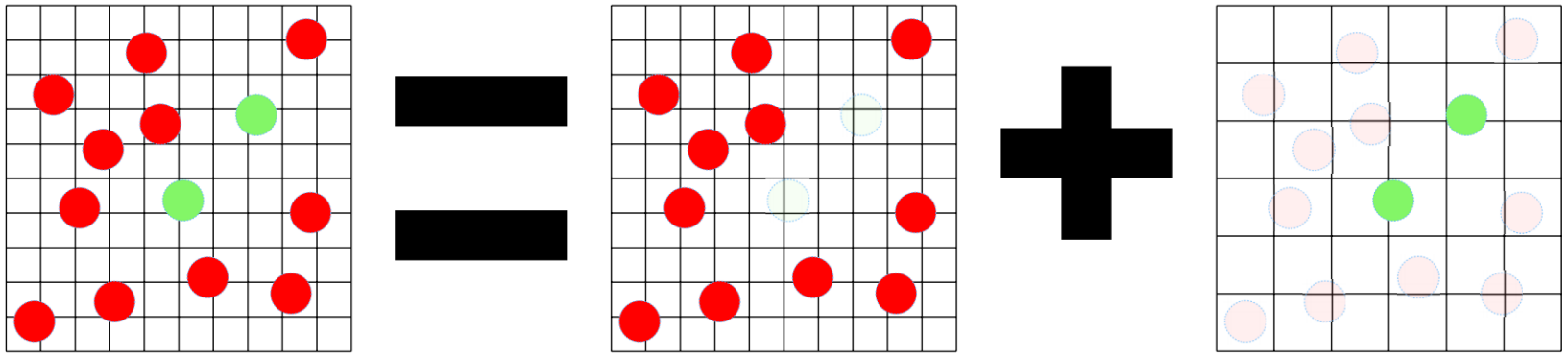
- Forces are interpolated from the electric field

$$\boldsymbol{f}_i^{long} \simeq q_i \sum_{\boldsymbol{x}_p \in \Omega_h} \mathscr{E}(\boldsymbol{x}_p) W(\boldsymbol{x}_p - \boldsymbol{x}_i)$$

❖ **Grid size and sigma controls accuracy/speed!**

Incremental Particle Particle Particle Mesh (IP3M)

- Divide and Conquer



- Grid sizes can be linked to the accuracy of computations

$$M_2 \simeq M\left[\frac{1}{\gamma}\sqrt{\frac{N_a}{N}}\right]^{\frac{1}{P}}$$

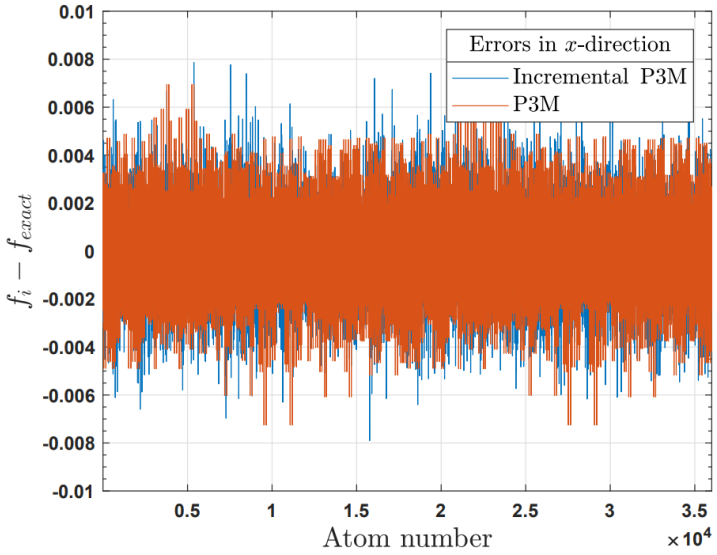
$$(\Delta f) \simeq (\Delta f_{1+2}) \Longrightarrow \gamma = \frac{\sqrt{2}}{2}$$

- Electric field is evaluated at atomic positions
- Problem 1 can be solved once.

❖Valid only when particles can't switch state

Incremental Particle Particle Particle Mesh (IP3M)

- SPC/E Water : 36000 particles
- Box : $70 * 70 * 70$ Å^3
- Relative accuracy: 10^{-5}



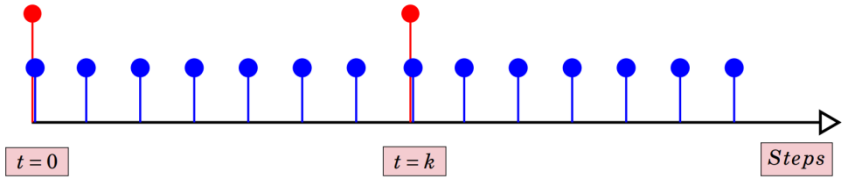
Error in forces (x-direction) of incremental Particle Particle Particle Mesh

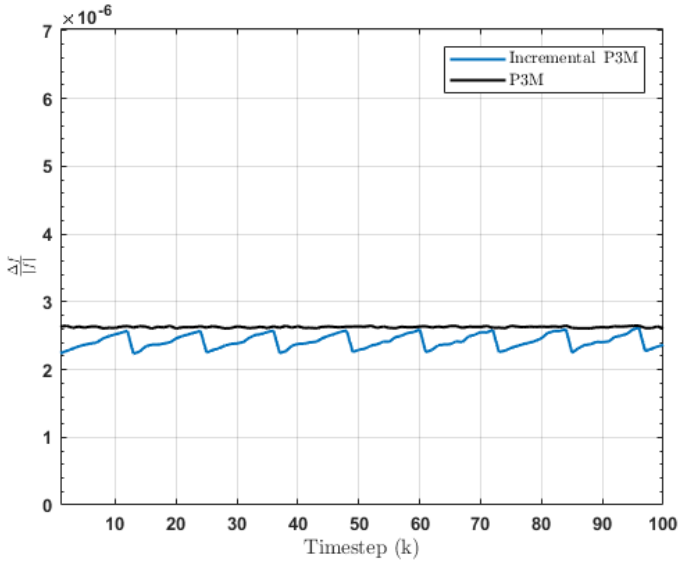
α_a	M	M_1	M_2	Δf	Δf_{1+2}	S_{long}
$\epsilon \simeq 10^{-4}$ and $\Delta f^{target} = 3.8 \times 10^{-2}$						
1%	36	40	30	2.50×10^{-2}	1.90×10^{-2}	×1.9
5%	36	40	30	2.50×10^{-2}	2.32×10^{-2}	×1.8
10%	36	40	32	2.50×10^{-2}	2.92×10^{-2}	×1.7
20%	36	40	36	2.50×10^{-2}	2.05×10^{-2}	×1.1
40%	36	40	36	2.50×10^{-2}	2.19×10^{-2}	×1.0
$\epsilon \simeq 10^{-5}$ and $\Delta f^{target} = 3.9 \times 10^{-3}$						
1%	60	64	45	2.85×10^{-3}	3.01×10^{-3}	×2.2
5%	60	64	50	2.85×10^{-3}	2.82×10^{-3}	×1.8
10%	60	64	54	2.85×10^{-3}	2.79×10^{-3}	×1.34
20%	60	64	60	2.85×10^{-3}	2.46×10^{-3}	×1.01
40%	60	64	60	2.85×10^{-3}	2.70×10^{-3}	×1.0
$\epsilon \simeq 10^{-6}$ and $\Delta f^{target} = 3.4 \times 10^{-4}$						
1%	100	108	72	2.96×10^{-4}	2.86×10^{-4}	×3.1
5%	100	108	80	2.96×10^{-4}	2.87×10^{-4}	×2.2
10%	100	108	81	2.96×10^{-4}	2.85×10^{-4}	×1.9
20%	100	108	90	2.96×10^{-4}	2.89×10^{-4}	×1.5
40%	100	108	90	2.96×10^{-4}	2.95×10^{-4}	×1.4
$\epsilon \simeq 10^{-7}$ and $\Delta f^{target} = 3.8 \times 10^{-5}$						
1%	180	180	120	2.37×10^{-5}	2.83×10^{-5}	×3.1
5%	180	180	135	2.37×10^{-5}	2.99×10^{-5}	×2.5
10%	180	180	144	2.37×10^{-5}	3.02×10^{-5}	×2.3
20%	180	180	150	2.37×10^{-5}	3.00×10^{-5}	×1.8
40%	180	180	160	2.37×10^{-5}	2.95×10^{-5}	×1.5

✓We can also handle switching particles

Incremental Particle Particle Particle Mesh (IP3M)

- **Problem 1** is solved every $\boldsymbol{k}$ steps
- **Problem 2** is slightly modified and account for the fact that some restrained particles have been displaced (by being active at least once)
- When the rate of switches is constant, we can estimate the number of particles that have been active and the suitable grid sizes
- $\boldsymbol{k}$ and $\boldsymbol{M_2}$ can be tuned to give the best performance
- **Speedup is less important**





Evolution of the RMS force error of IP3M.
90% of particles are restrained 1% of particles switch at each timestep.
Speedup of long- (resp. short-) range calculations **1.35** (resp. **3.9**).

Meshed Continuum Method (MCM)

- Point charges are replaced by **smooth densities** which have a compact support
- The induced potential is the solution of a Poisson equation solved with a **multigrid** solver: $\Delta\Phi = \rho^{sm}$
- Coulomb quantities are interpolated from a gridded solution
- A Particle-Particle **near-field correction** is used to properly retrieve electrostatics
- O(N) scaling but slower than P3M.
- **Sampling** of the RHS of the Poisson equation and the **near-field correction** are the most extensive task.

❖ **The grid size and the support of the density control accuracy/speed!**

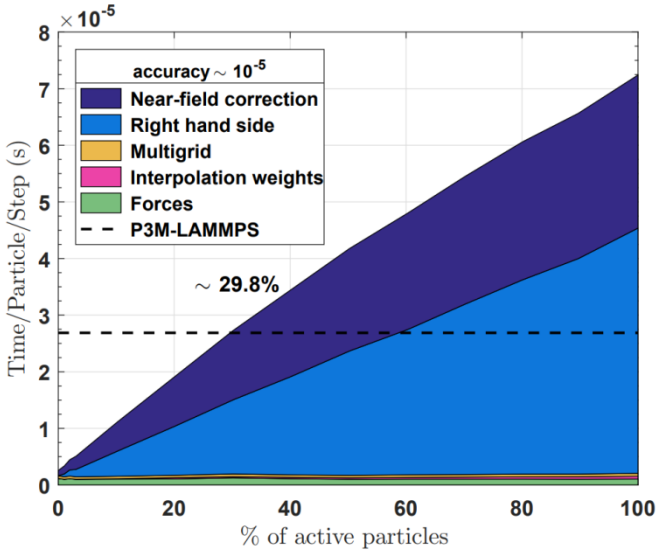
[Bolten 2008, Arnold 2013]

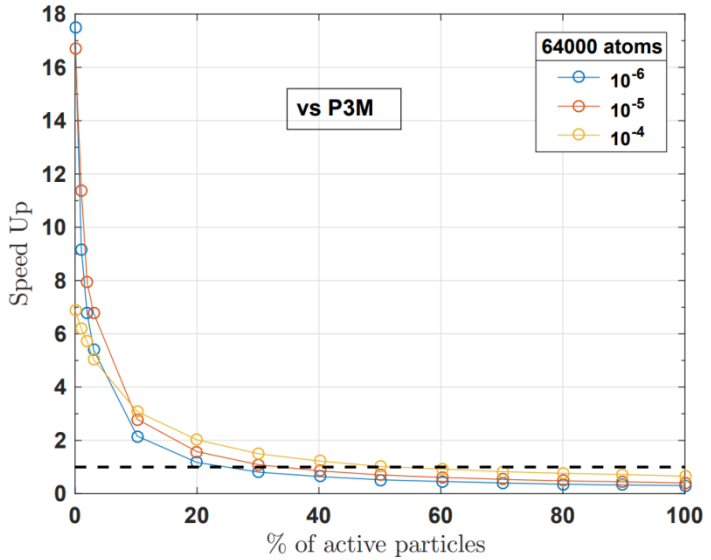
Incremental Meshed Continuum Method (IMCM)

- Sampling of the RHS can benefit from a divide and conquer

$$\rho^{sm} = \rho^{(Active)} + \rho^{(Restrained)}$$

- The Particle-Particle near-field correction is treated with pair_***.cpp : **ANLs**
- **IMCM** outperforms **P3M** for low numbers of active particles.



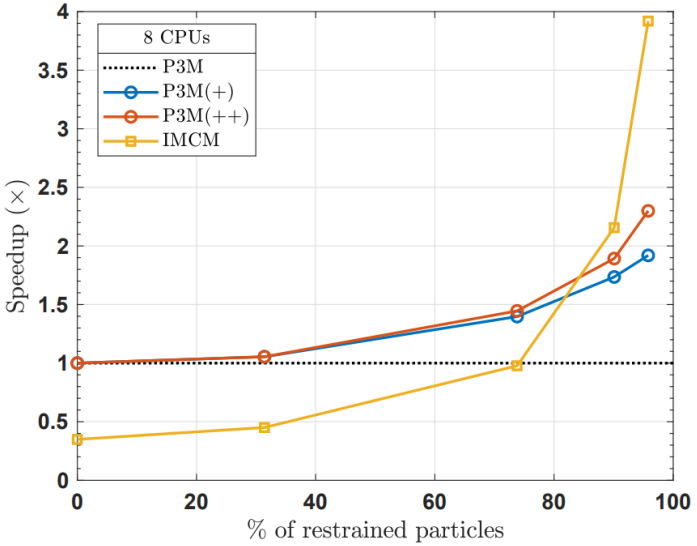


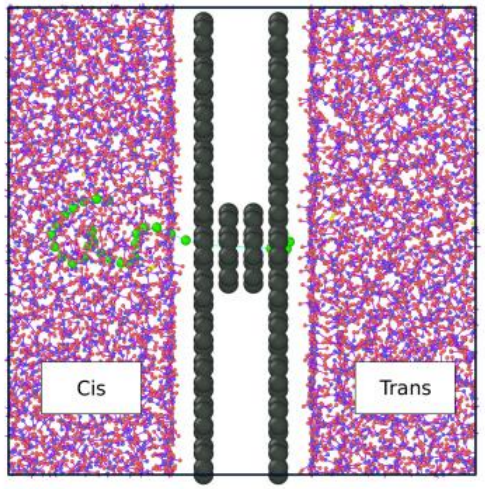
IMCM vs P3M for various proportions of restrained particles.
NaCl system (64000 atoms in 113^3 Å^3 box)

[Edorh 2018]

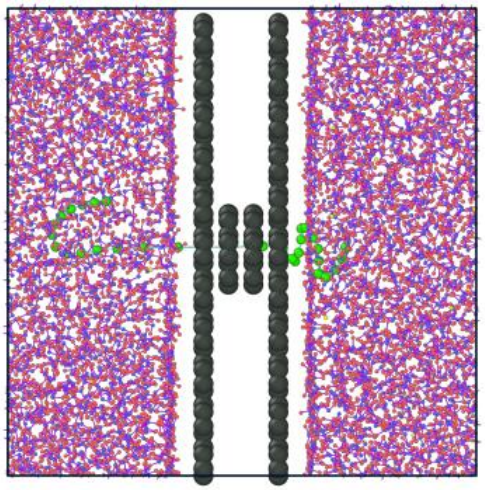
Application : Polyelectrolyte translocation

- 60K particles
- Polyelectrolyte: 64 beads
- Pore length: 10 Å
- Electric field inside the pore
- LJ + Electrostatics

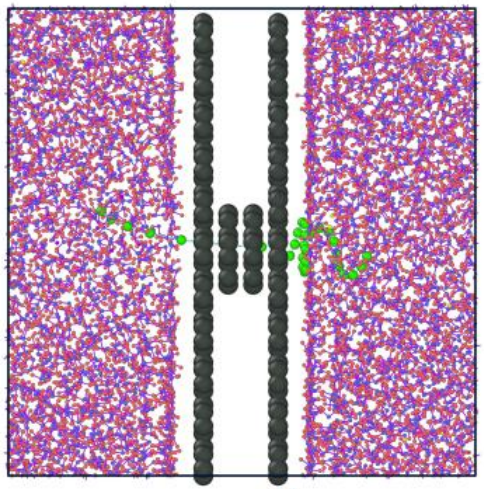




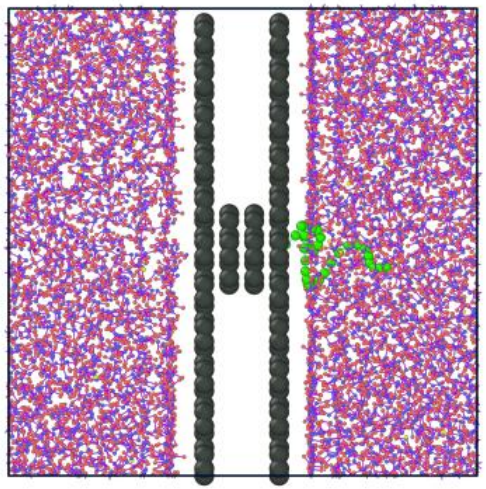
(1) $t = 0$



(2) $t = \frac{\tau}{3}$



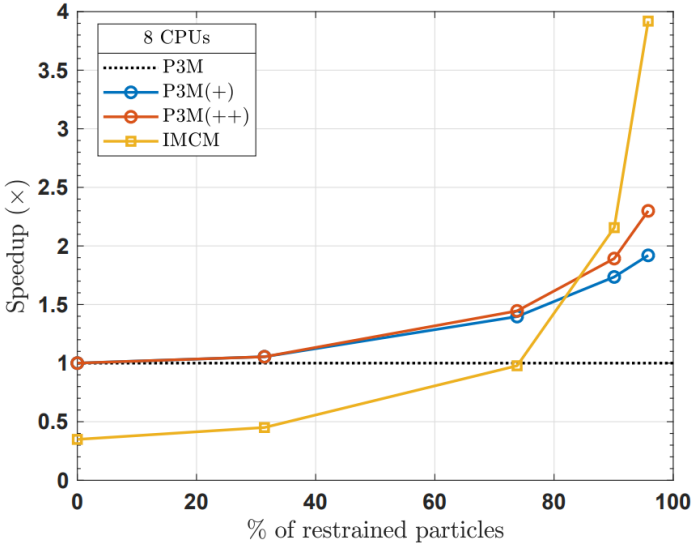
(3) $t = \frac{2\tau}{3}$

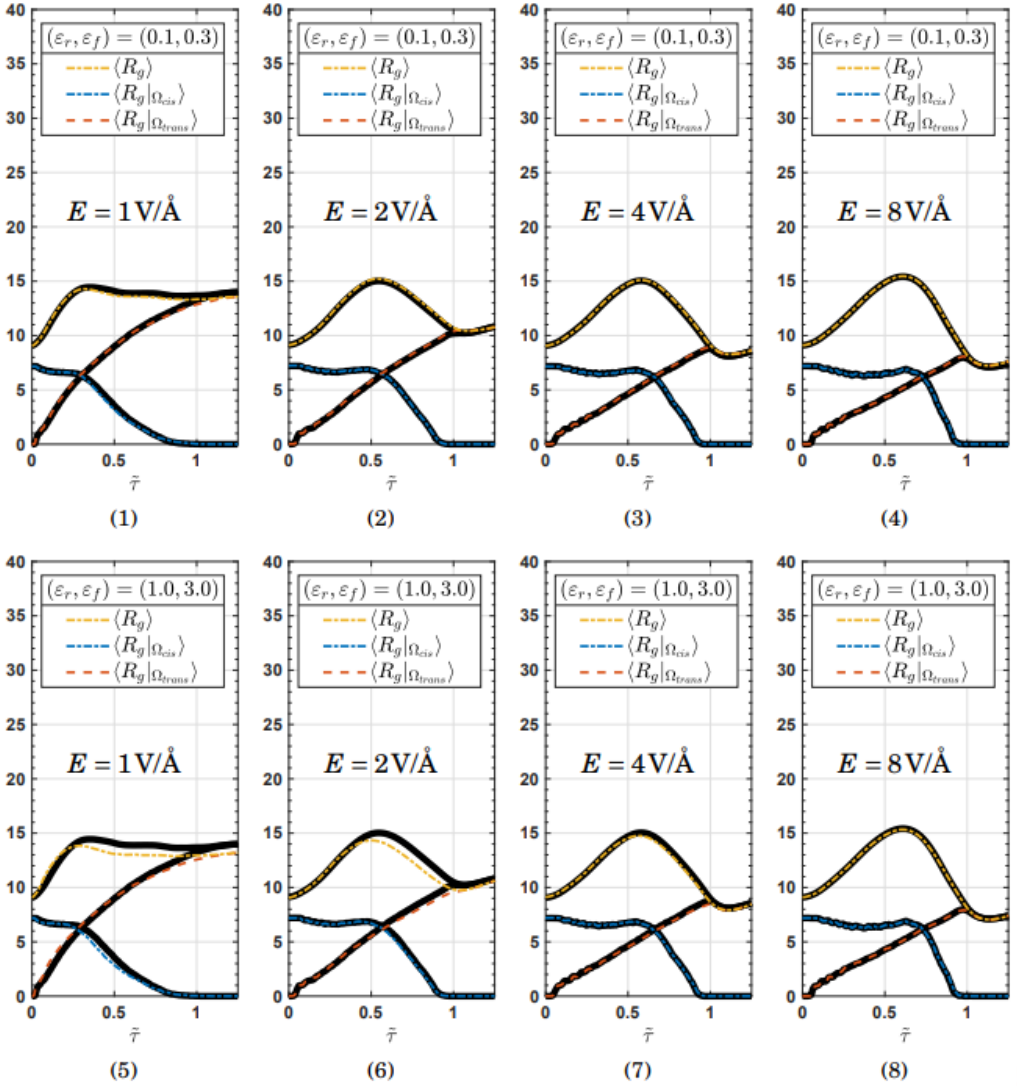


(4) $t = \tau$

Application : Polyelectrolyte translocation

- 60K particles
- Polyelectrolyte: 64 beads
- Pore length: 10 Å
- Electric field inside the pore
- LJ + Electrostatics





Future work

- Incremental Multilevel Summation Method
- Improve performance on multicore architectures
- Applications such as channels in membrane proteins, ion implantation, molecular docking, protein folding
- Study of correlated motions (Essential dynamics using ARMD)

Publications

- Svetlana Artemova and Stephane Redon. **Adaptively restrained particle simulations**. Physical Review Letters, 109(19):190201, nov 2012. ISSN 00319007. doi: 10.1103/PhysRevLett.109.190201.
- Zofia Trstanova and Stephane Redon. **Estimating the speed-up of adaptively restrained Langevin dynamics.** Journal of Computational Physics, 336:412–428,2017.
- Gabriel Stoltz and Zofia Trstanova. **Stable and accurate schemes for langevin dynamics with general kinetic energies.** arXiv preprint arXiv:1609.02891, 2016.
- Krishna Kant Singh and Stephane Redon**. Adaptively Restrained Molecular Dynamics in LAMMPS**. Modelling and Simulation in Materials Science and Engineering, 2017. ISSN 1361651X. doi: 10.1088/1361-651X/aa7345.
- Krishna Kant Singh, Dmitriy F. Marin, and Stephane Redon. **Parallel adaptively restrained molecular dynamics**. In High Performance Computing & Simulation(HPCS), 2017 International Conference on, pages 308–314. IEEE, 2017.
- Krishna Kant Singh and Stephane Redon. **Single-pass incremental force updates for adaptively restrained molecular dynamics**. Journal of Computational Chemistry, 2017. ISSN 1096-987X. doi: 10.1002/jcc.25126.
- Semeho Prince A. Edorh and Stéphane Redon. **Incremental update of electrostatic interactions in adaptively restrained particle simulations**. Journal of Computational Chemistry, 2018. doi: 10.1002/jcc.25215.
- Dmitriy F. Marin, and Stephane Redon. **Parallel implementation of Adaptively Restrained Molecular Dynamics in LAMMPS using Kokkos**. The International Journal of High Performance Computing Applications,(in review), 2018.

https://team.inria.fr/nano-d/