

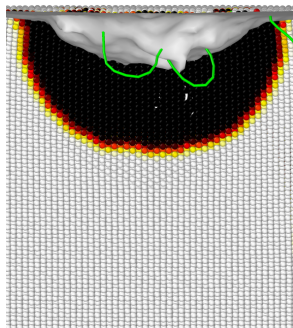
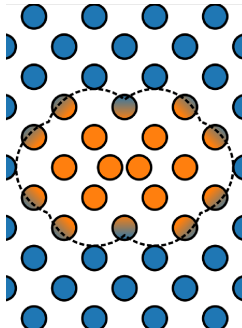
Load Balancing for Adaptive-Precision Interatomic Potentials in Materials Science

2026-08-31 | David Immel¹, Godehard Sutmann^{1,2} | 1 Jülich Supercomputing Centre 2 ICAMS

Equations of motion

$$m_i \ddot{\mathbf{x}}_i = -\nabla_i E(\mathbf{x}) = \mathbf{F}_i$$

**Decisive phenomena
may include only a
subset of atoms.**



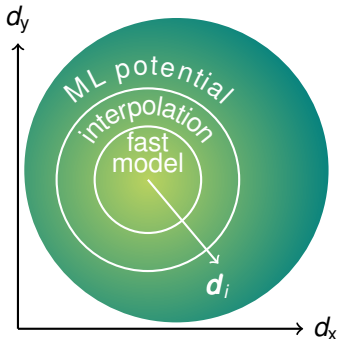
crack: M. J. Buehler et al., *Phys. Rev. Lett.* **96**, 095505 (2006)

interstitial: F. Birks et al., *arXiv:2502.19081* (2025)

nanoindentation: D. Immel et al., *Phys. Rev. Mater.* **9**, 093805 (2025)

Adaptive-precision description

characterisation per atom i → switching parameter λ_i



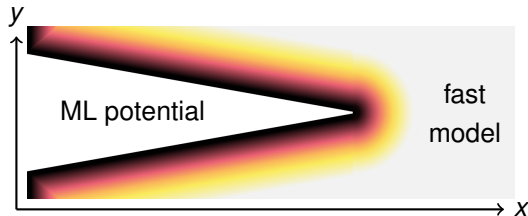
D. Immel et al., arXiv:2512.07693 (2025)

energy- or force-mixing

$$\gamma_i = \lambda_i \gamma_i^{\text{fast}} + (1 - \lambda_i) \gamma_i^{\text{ML}}$$

$$\gamma_i = E_i \quad \text{D. Immel et al., J. Chem. Phys. } \mathbf{162}, 114119 \text{ (2025)}$$

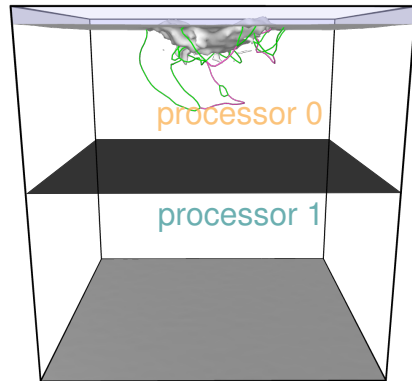
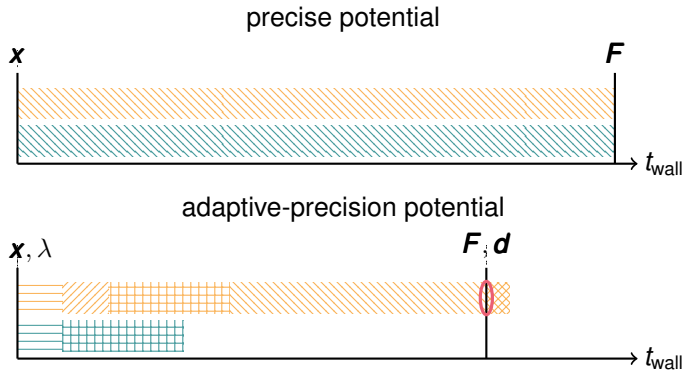
$$\gamma_i = F_i \quad \text{F. Birks et al., arXiv:2502.19081 (2025)}$$



D. Immel et al., arXiv:2512.07693 (2025)

A fast surrogate model can provide speedup compared to a ML-model.

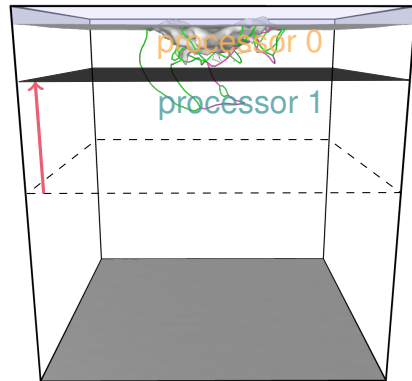
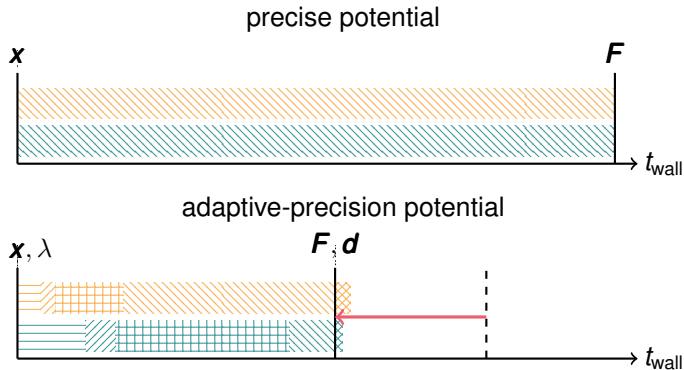
Parallel force calculation



visualised with OVITO: A. Stukowski, Model. Simul. Mater. Sci. Eng. **18**, 015012 (2009)

The slowest processor determines the runtime.

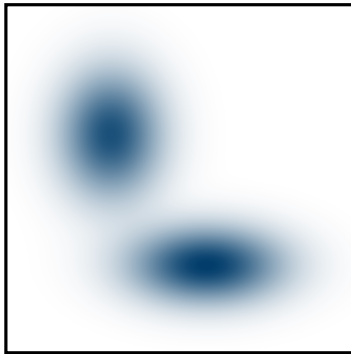
Parallel force calculation



visualised with OVITO: A. Stukowski, Model. Simul. Mater. Sci. Eng. **18**, 015012 (2009)

**The slowest processor determines the runtime.
Load balancing (LB) decreases the runtime.**

Spatial domain decompositions

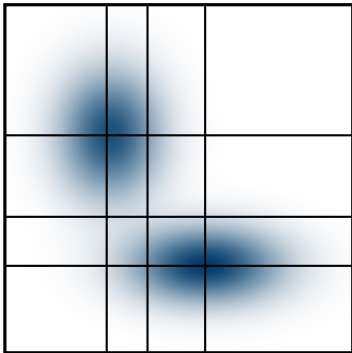


regular grid (RG)

staggered grid (SG)

irregular grid (IG)

**Each grid has unique
(dis)advantages.**



**Each grid has unique
(dis)advantages.**

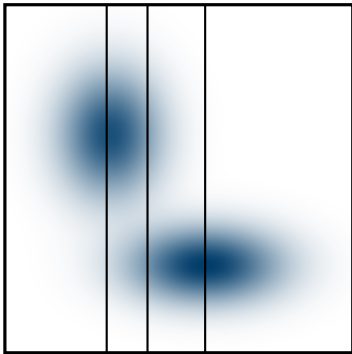
regular grid (RG)

- independent 1D decompositions
- $\text{\#DOF} = P_x + P_y - 2$
- the easiest to implement
- communication is optimised with fixed number of neighbours

staggered grid (SG)

irregular grid (IG)

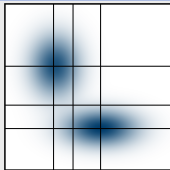
Spatial domain decompositions



Each grid has unique (dis)advantages.

SG: H. Iliev et al., JHPCS 2016, 187, (2017)

regular grid

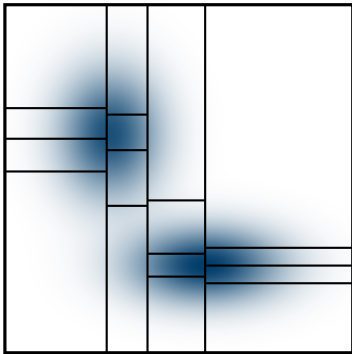


staggered grid (SG)

- recursive 1D decompositions
- $\#DOF = P - 1$
- load balancing (LB) with local information possible

irregular grid (IG)

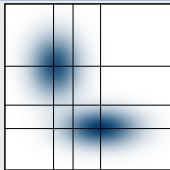
Spatial domain decompositions



**Each grid has unique
(dis)advantages.**

SG: H. Iliev et al., JHPACS 2016, 187, (2017)

regular grid

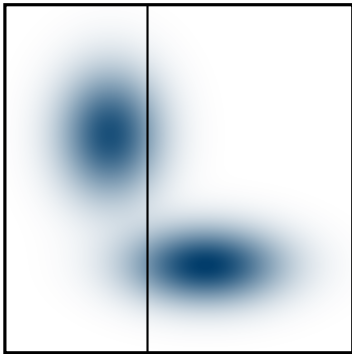


staggered grid (SG)

- recursive 1D decompositions
- $\#DOF = P - 1$
- load balancing (LB) with local information possible

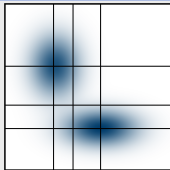
irregular grid (IG)

Spatial domain decompositions

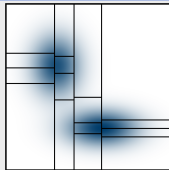


Each grid has unique (dis)advantages.

regular grid



staggered grid



irregular grid (IG)

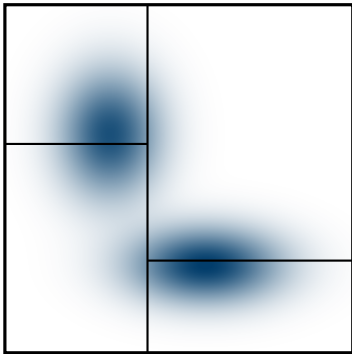
- recursive coordinate bisections
- surface-to-volume ratio can be optimised
- $\#DOF = P - 1$

SG: H. Iliev et al., JHPCS 2016, 187, (2017)

RCB: M. J. Berger et al., IEEE Trans.

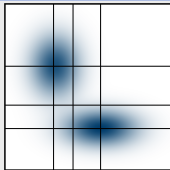
Comput. **C-36**, 570–580 (1987)

Spatial domain decompositions

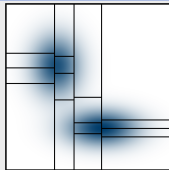


Each grid has unique (dis)advantages.

regular grid



staggered grid



irregular grid (IG)

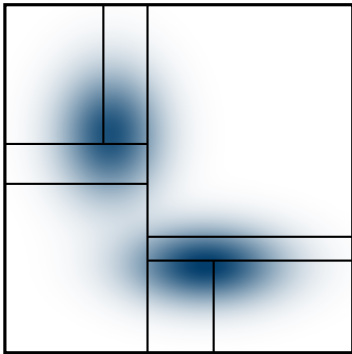
- recursive coordinate bisections
- surface-to-volume ratio can be optimised
- $\#DOF = P - 1$

SG: H. Iliev et al., JHPCS 2016, 187, (2017)

RCB: M. J. Berger et al., IEEE Trans.

Comput. **C-36**, 570–580 (1987)

Spatial domain decompositions



Each grid has unique (dis)advantages.

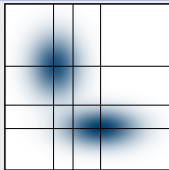
SG: H. Iliev et al., JHPACS 2016, 187, (2017)

RCB: M. J. Berger et al., IEEE Trans.

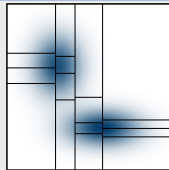
Comput. **C-36**, 570–580 (1987)

D. Immel, G. Sutmann

regular grid



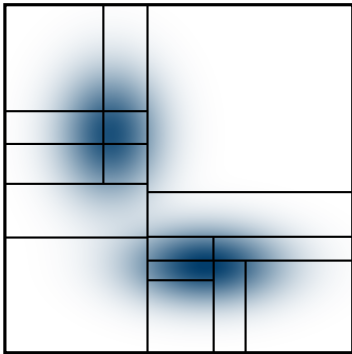
staggered grid



irregular grid (IG)

- recursive coordinate bisections
- surface-to-volume ratio can be optimised
- $\#DOF = P - 1$

Spatial domain decompositions



Each grid has unique (dis)advantages.

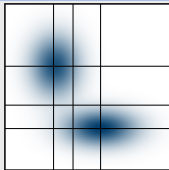
SG: H. Iliev et al., JHPACS 2016, 187, (2017)

RCB: M. J. Berger et al., IEEE Trans.

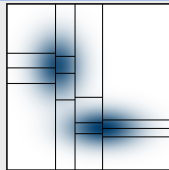
Comput. **C-36**, 570–580 (1987)

D. Immel, G. Sutmann

regular grid



staggered grid



irregular grid (IG)

- recursive coordinate bisections
- surface-to-volume ratio can be optimised
- $\#DOF = P - 1$

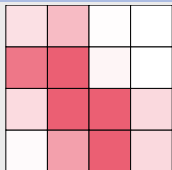
Spatial domain decompositions

- load per processor L_p
- load imbalance

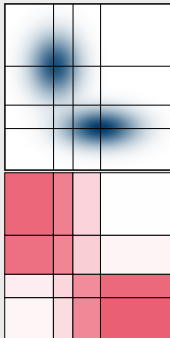
$$I = \frac{\max_p (L_p)}{\langle L_p \rangle_p}$$

- aim: $I = 1$

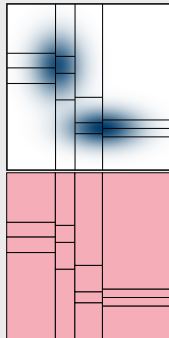
without LB



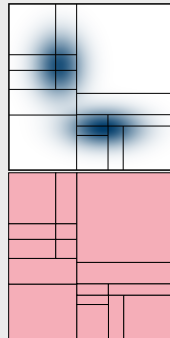
regular grid



staggered grid

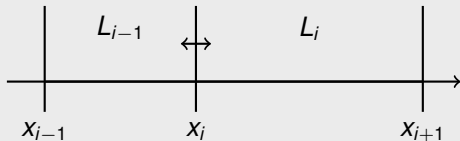


irregular grid



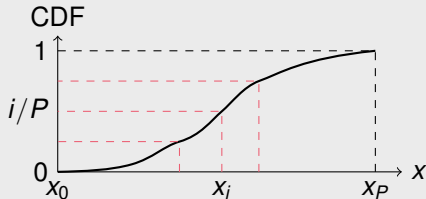
Every load-balancing strategy is beneficial.

iterative



- only local information required
- iterates to convergence
- for regular and staggered grids

CDF-based



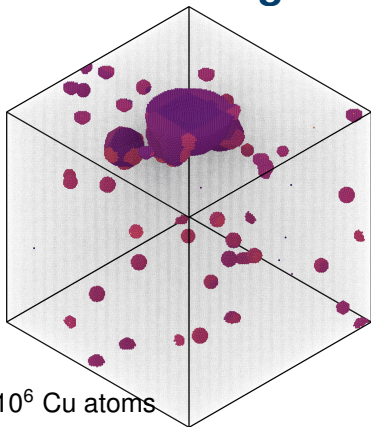
- global information required
- for all grids

- ALL (A Load-balancing Library): iterative and CDF-based LB for RG and SG
- LAMMPS (particle simulator): CDF-based approaches for RG and IG

ALL: gitlab.jsc.fz-juelich.de/SLMS/loadbalancing

LAMMPS: github.com/lammps/lammps

Load balancing of an APIP

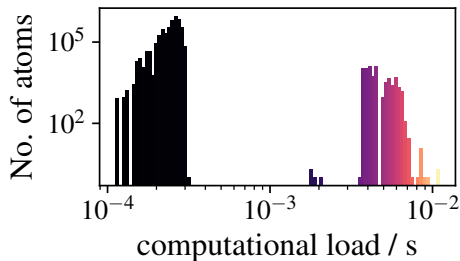


$4 \cdot 10^6$ Cu atoms

visualised with OVITO: A. Stukowski, Model. Simul. Mater. Sci. Eng. **18**, 015012 (2009)

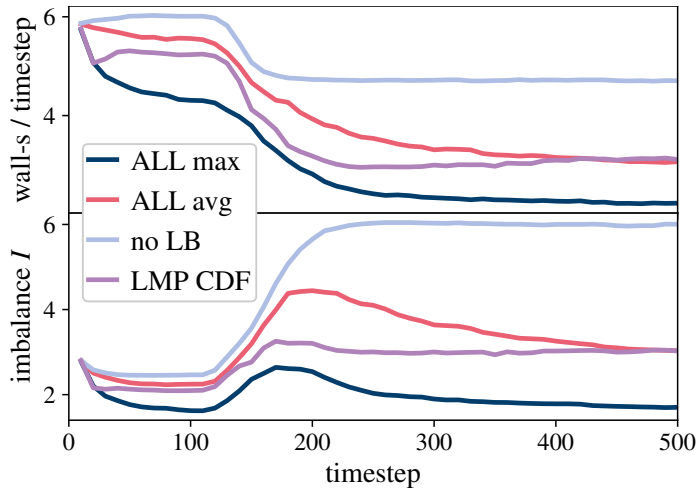
JURECA-DC: Jülich Supercomputing Centre, JLSRF 7, A182 (2021)

- energy-mixing with APIP package of LAMMPS
- switching parameter λ_i calculated from time average
- load $l_{p,i}$ is estimated per atom i and processor p
- force-calculation time as load $L_p = \sum_i l_{p,i}$ is measured
- 128 cores as 3D grids on JURECA-DC



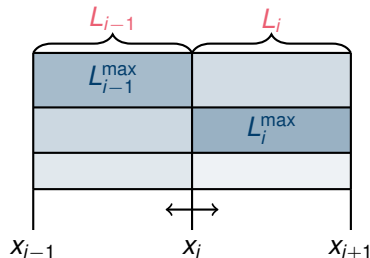
The load-distribution changes spatially and temporally.

APIP with regular grid

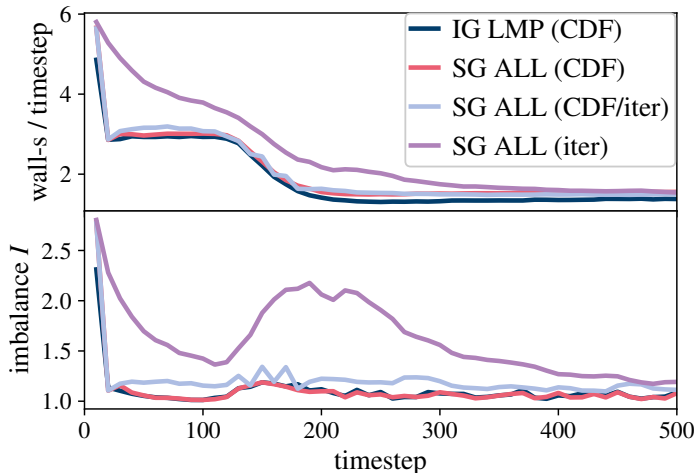


Results

- **no LB** is the slowest
- **iterative LB** converges against **CDF-based LB**
- **LB of maxima** outperforms LB of the average



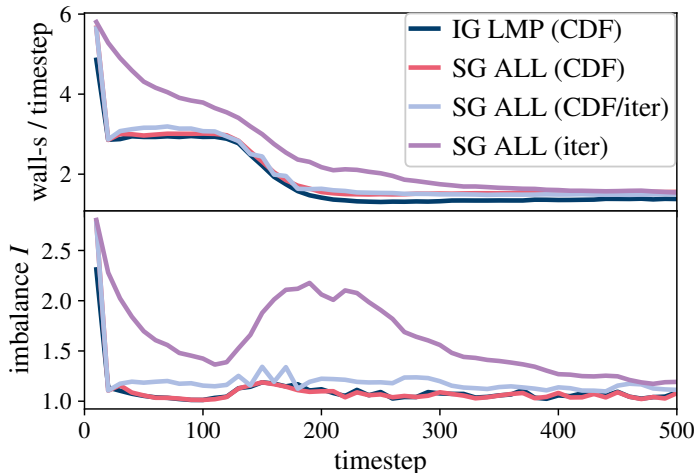
APIP with staggered/irregular grid



Results

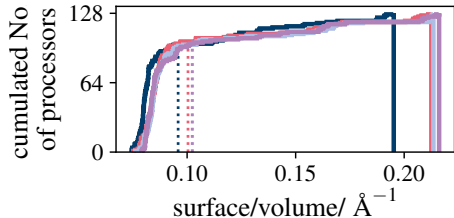
- CDF-based: $I \approx 1$

APIP with staggered/irregular grid

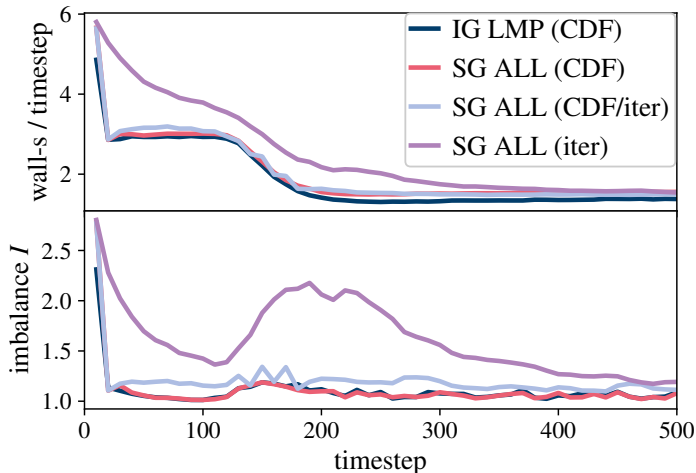


Results

- CDF-based: $I \approx 1$
- **IG** is faster than **SG**

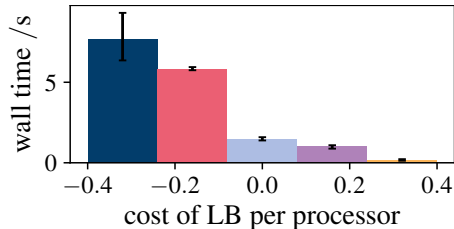


APIP with staggered/irregular grid



Results

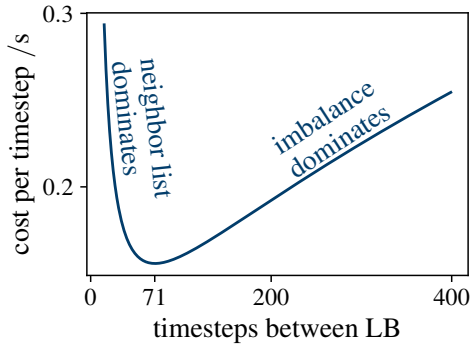
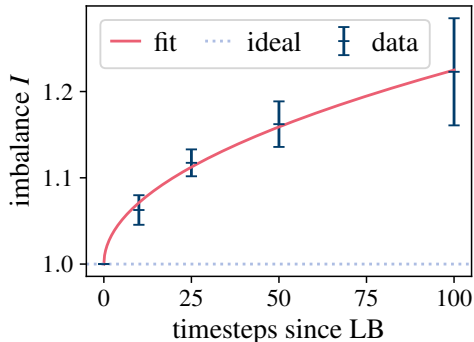
- CDF-based: $I \approx 1$
- **IG** is faster than **SG**
- LB of **IG** is faster than LB of **SG**



Load-balancing frequency

“costs” dependent on load balancing

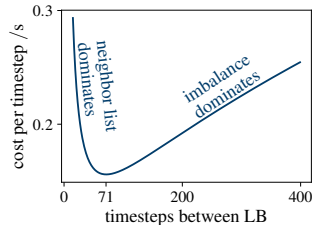
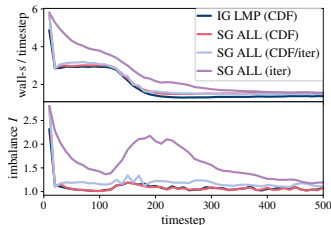
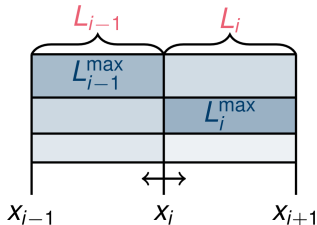
- LB itself
- neighbour list rebuild in LAMMPS
- idle time due to imbalance



The ideal LB frequency

- depends on the system
- can be estimated.

Conclusion: load balancing APIs



regular grid

Balancing the maximum with ALL is the best strategy.

staggered / irregular grid

Both reach a nearly perfectly balanced system ($I \approx 1$).

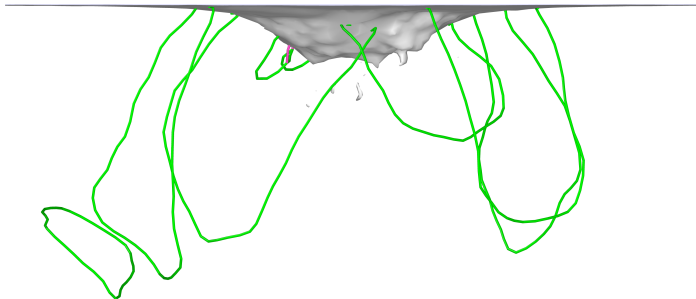
optimal LB frequency

- system-dependent
- estimable

ALL in LAMMPS:
ALL:

github.com/d-immel/lammps_ap/tree/allpkg
gitlab.version.fz-juelich.de/SLMS/loadbalancing





Load Balancing for Adaptive-Precision Interatomic Potentials in Materials Science

2026-08-31 | David Immel¹, Godehard Sutmann^{1,2} | 1 Jülich Supercomputing Centre 2 ICAMS