JÜLICH
FORSCHUNGSZENTRUM

Performance Optimisation of Coarse-Graining MD Codes IBiSCO and OCCAM

Dr. Viorel Chihaia

Jülich Supercomputing Centre

SL Molecular Systems

v.chihaia@fz-juelich.de

Jülich, JSC, 7-8 May 2013

JSC Support Activity:

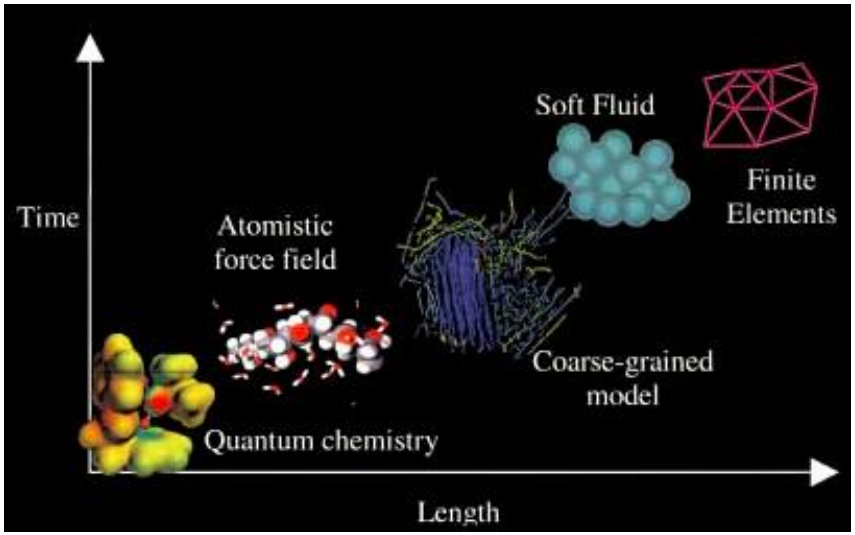
Performance Optimization of the Coarse Grained codes

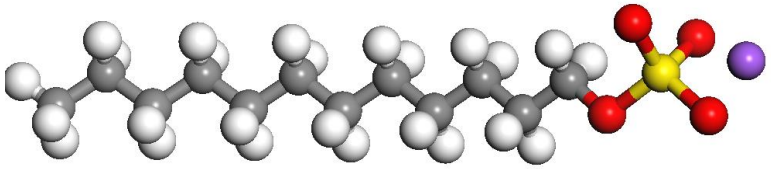
IBiSCO and OCCAM

(yasp roots)

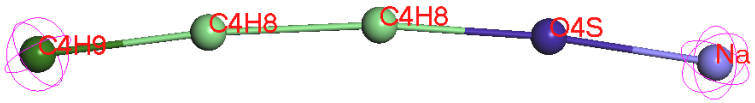
Length scales, associated time scales and computational methods

F. Müller-Plathe, ChemPhysChem, 3 (2002) 754



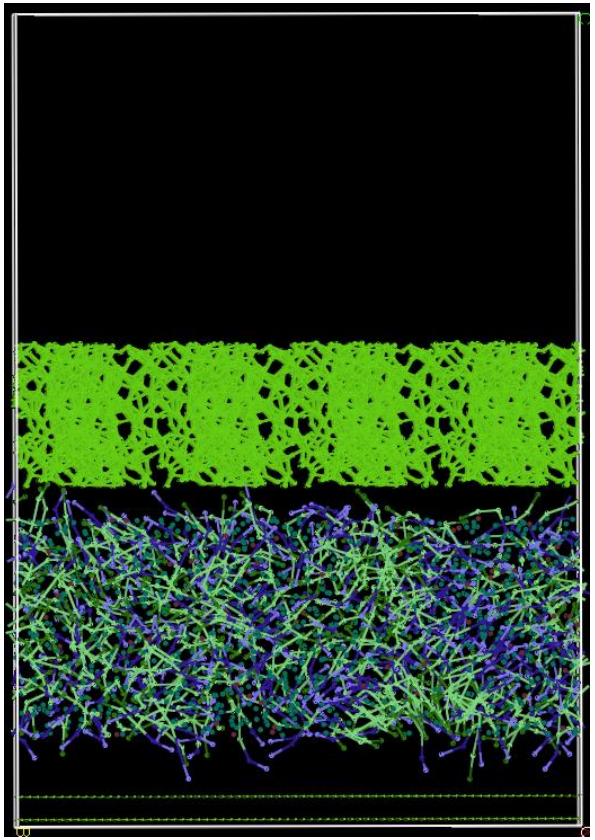


Sodium Dodecyl Sulfate: All Atoms



Sodium Dodecyl Sulfate: Coarse Grain
Beads (C4H9, C4H8, Q4S, Na)

-Less freedom degrees
-Weaker interactions
-Larger time steps



Confined SDS

IBIsCO

(It is Boltzmann Inversion software for Coarse Graining Simulations)

Prof. Florian Müller-Plathe
Theoretical Physical Chemistry Group of
Technischen Universität Darmstadt

Code Homepage:
http://www.theo.chemie.tu-darmstadt.de/ibisco/

H.A. Karimi-Varzaneh, H.-J. Qian, X. Chen, P. Carbone, Fl. Müller-Plathe,
J.Comput.Chem., 32 (2011) 1475
IBIsCO: A molecular dynamics simulation package for coarse-grained simulation

IBIsCO – J.Comput.Chem. 2011(32) 1475

Hybrid Model Atomistic - Coarse-Grained for soft material molecular dynamics simulations.

Total potential energy

$$V = V_{\text{bond}} + V_{\text{angle}} + V_{\text{torsion}} + V_{\text{nonbonded}}$$

The model employs the classical atomistic force fields and the CG force fields developed using the Iterative Boltzmann Inversion.

The bond and angle interactions can be defined as tabulated potentials or Gaussian functions:

$$V(l) = -k_B T \ln \sum_{i-1}^{n} g_i(l)$$

$$V(\theta) = -k_B T \ln \sum_{i-1}^{n} g_i(\theta)$$

Three Thermostats - Three Integration Schemes

- **Berendsen (Leapfrog Scheme)**

  H.J.C.Berendsen, et. al, *J.Chem.Phys.*, 81 (1984) 3684
  Molecular-Dynamics with Coupling to an External Bath

- **DPD (Velocity Verlet Scheme)**

  P. Español, *Phys.Rev.E.*, 52 (1995) 1734
  Hydrodynamics from Dissipative Particle

  P. Español, *Europhys. Lett.*, 40 (1997) 631
  Dissipative particle dynamics with energy conservation

  R.D. Groot and P.B. Warren, *J.Chem.Phys.*, 107 (1997) 4423
  DPD: Bridging the gap between atomistic and mesoscopic simulation

- **LA (Leapfrog Scheme)**

  C.P. Lowe, *Europhys.Lett.*, 47 (1999) 145
  An alternative approach to dissipative particle dynamics

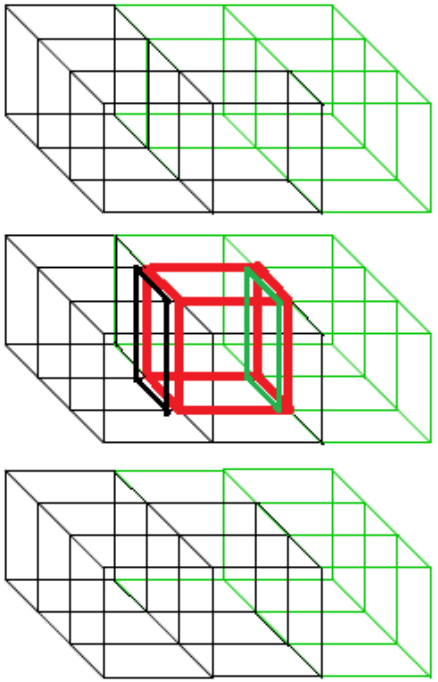
MPI Parallelized using a ***3D domain decomposition method***.

A processor sends/receives boundary data to/from 13 of its neighbour processors:

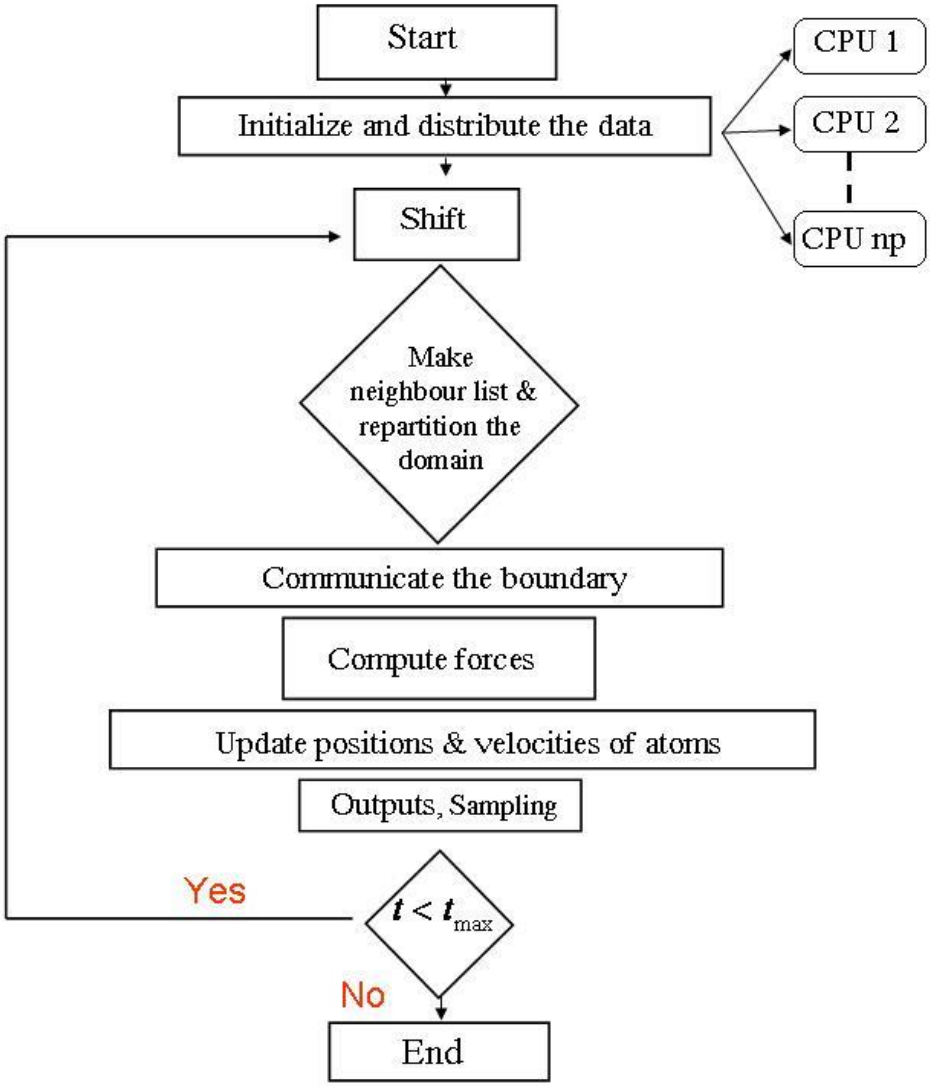
- Coordinates of the boundary particles ($r<r_{cut}$)are sent to neighbour processors,

- Forces are calculated locally on the processors

- Forces / velocities are send back to the parent processors



IBIsCO


Start
Initialize and distribute the data
CPU 1
CPU 2
CPU np
Shift
Make neighbour list & repartition the domain
Communicate the boundary
Compute forces
Update positions & velocities of atoms
Outputs, Sampling
Yes
$t < t_{max}$
No
End

Changes done in the code IBIsCO

Godehard Sutmann, Rene Halver

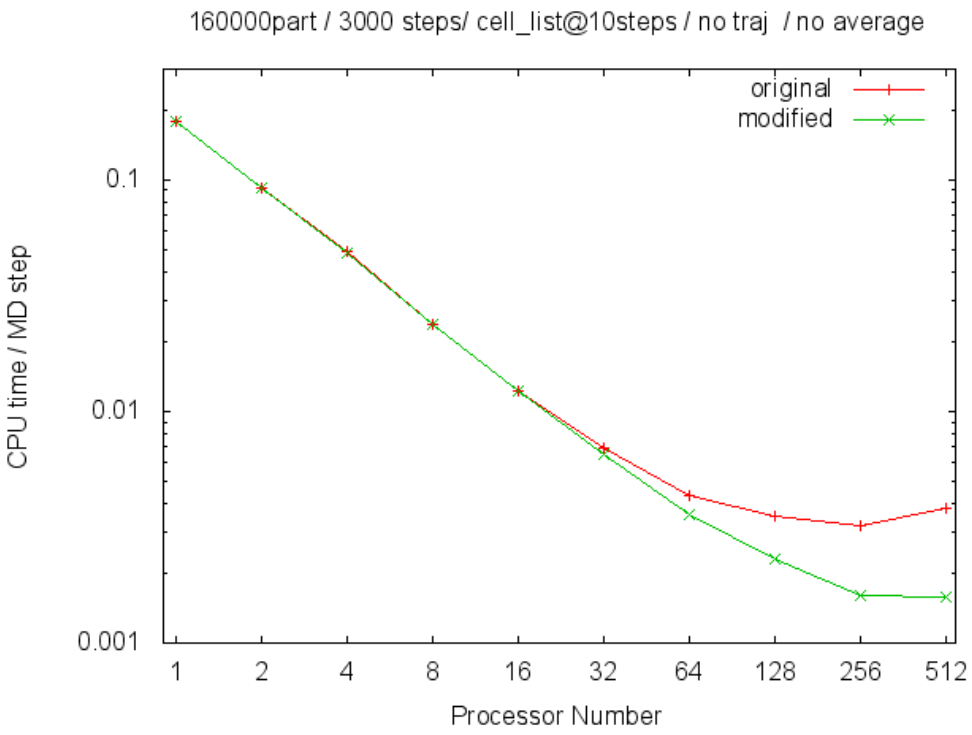
- Rewrite the subroutines *writetrj* and *average,* the two subroutines that were very time consuming

- The subroutines *takepos* and *takeen* are now called only once for the case that the subroutines *writetrj* and *average* are called during the MD simulations,

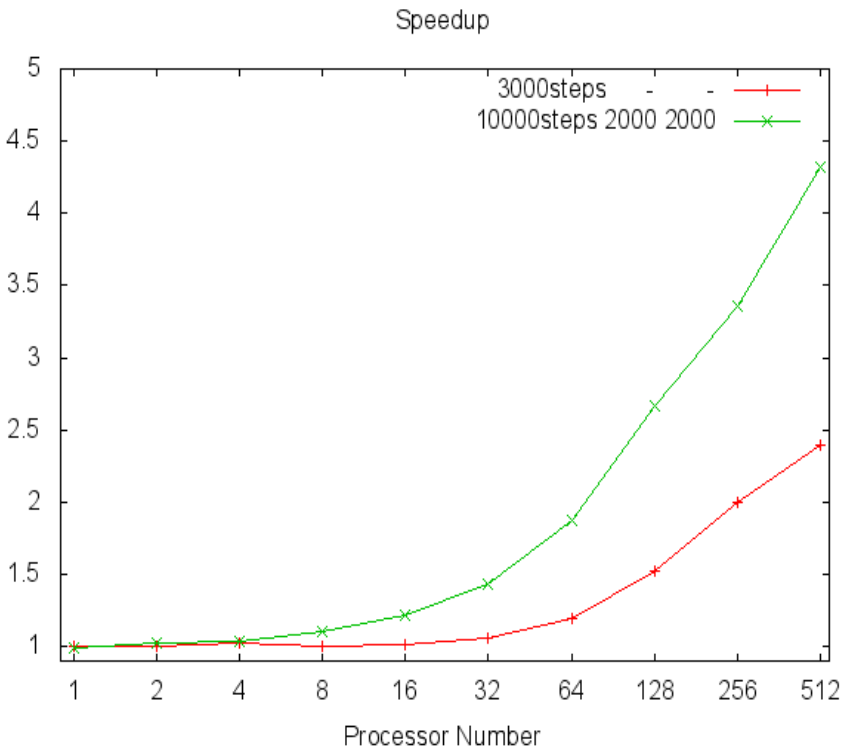
- The communication scheme of the data between the processors was changed by assembling the different components of the positions and velocities, as well as the atom and cell indices in the communication buffers DATA*n*_EXP or DATA*n*_IMP

- Rationalization of the memory usage

IBIsCO – improved parallelization

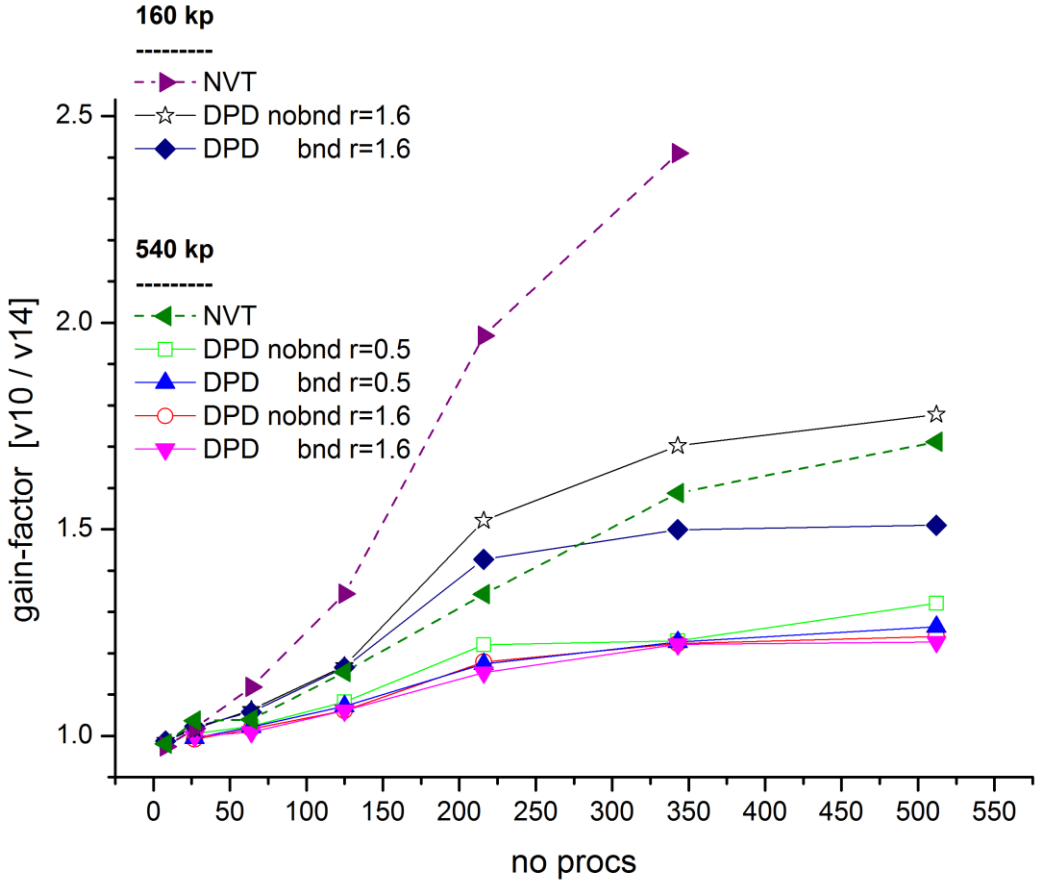


CPU time per MD step, for 3000 MD steps without trajectory writing and averaging



The gain-up of the calculations in the two sets of simulations.

IBIsCO – improved parallelization



The gain-factor versus the number of processors after the update of the inter-processor communication scheme. The simulations were done for two systems formed by 160 kp and 540 kp, respectively.

The DPD simulations are done including (bnd) or not including (nobnd) the bonded interactions in the DPD algorithm, for two different DPD cut-off distances r=0.5 (empty symbols) and 1.6 nm (filled symbols), respectively.

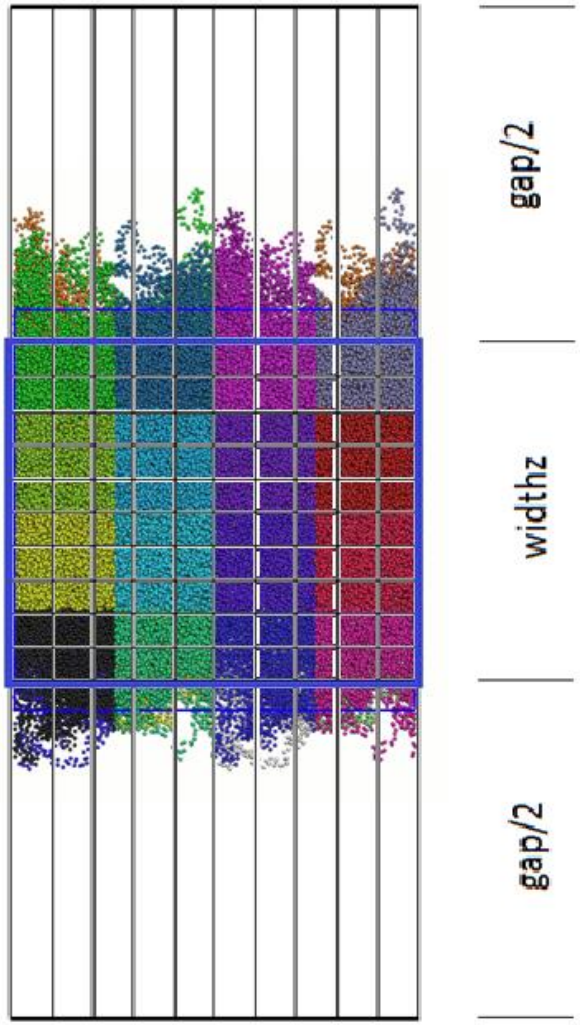
IBIsCO – Adaption for surface simulations

3D Periodicity for surface calculations:

- A huge memory must be allocated and wasted - the entire expanded box has to be covered with a very large number of cells, furthermore, most of the cells (2/3) would be empty,
- A high imbalanced load occurs - many processor domains would be empty or would have only a very reduced number of particles assigned.

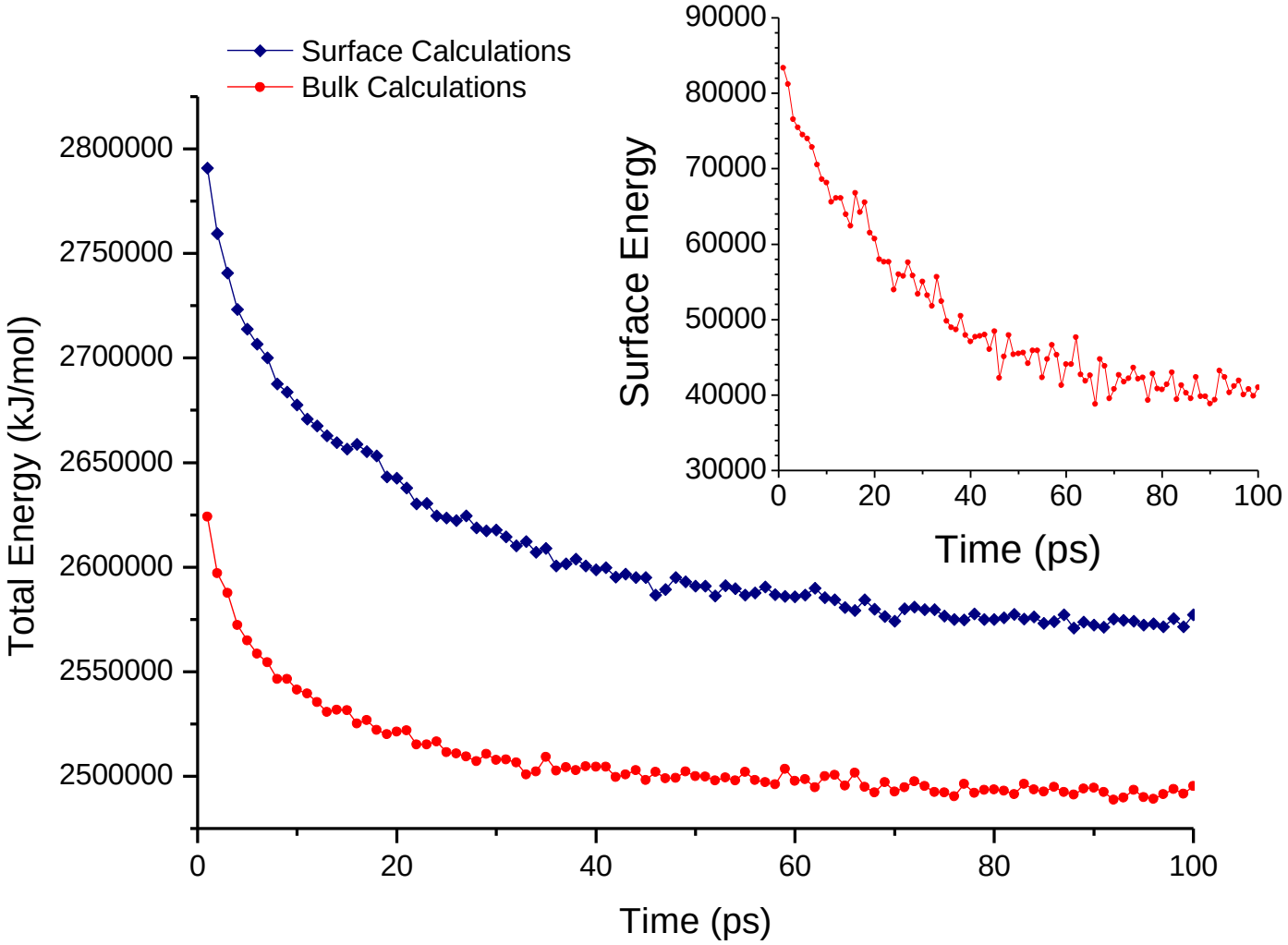
Extended Cells for surface calculations:

- The blue box corresponds to the bulk box.
- The processor domains at the top and the bottom of the surface slab are extended to include the void areas.
- The size along z of the void areas is given by the parameter gap.



The distribution of the particles onto the processor domains (the areas with different colours) for a 4x4x4 processor partition (with a 3D periodicity).

IBIsCO – Adaption for surface simulations



Evolution of the surface and bulk energies in a NVT simulation for a system of 160 kpart

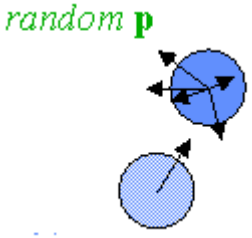
IBIsCO – Pairwise Thermostats for CG Dynamics

- The corse-graining interactions are softer and the systems are rather complex. A local thermostat is desired to enforce the *local heat dissipation*. The most efficient MD thermostat, *Nose-Hoover*, is a global one.
- All thermostats introduce unphysical features to the system dynamics.
- Proper sampling of the canonical ensemble requires that the system temperature fluctuates. Fluctuations vanish at large N $\frac{\sigma_{T_p}^2}{\langle T_p \rangle^2} = \frac{2}{3N}$

Andersen Thermostat – simplest pairwise thermostat (fictive collisions)

Each particle undergoes a collision with a fictive particle with a given probability $\Gamma\Delta\tau$. The fictive particle has a random velocity from a Maxell-Boltzman distribution, corresponding to the simulation temperature:

$$\left(\frac{m_i}{2\pi k_B T}\right)^{\frac{1}{2}} \exp\left(-\frac{m_i v_{x,i}^2}{2k_B T}\right)$$



Andersen Thermostat

Advantages:

• **Local thermostat,**

• Within the limit of an infinitely long trajectory averaged over many heat bath collisions, the Andersen thermostat rigorously generates the correct canonical ensemble probabilities.

• One can use longer time-steps without producing significant errors in the equilibrium properties

• Proper sampling of the canonical ensemble requires that the system temperature fluctuates.

Desavantages:

• It does not conserve the liniar and angular momentum,

•At high thermostating rates, the diffusion in the system is suppressed (leading to an inefficient sampling of phase space)

• The random collisions cause the velocities of particles to de-correlate much faster than the NVE dynamics. As a result, the true molecular kinetics is not preserved by the Andersen thermostat. For example, the computed diffusion constants for particles would give erroneous values.

DPD Thermostat

R.D. Groot and P.B. Warren, *J.Chem.Phys.*, 107 (1997) 4423

DPD: Bridging the gap between atomistic and mesoscopic simulation

$$\mathbf{f}_i = \sum_{j \neq i} (\mathbf{F}_{ij}^{C} + \mathbf{F}_{ij}^{D} + \mathbf{F}_{ij}^{R})$$

$$\mathbf{F}_{ij}^{C} = \begin{cases} a_{ij}(1 - r_{ij})\hat{\mathbf{r}}_{ij} & (r_{ij} < 1) \\ 0 & (r_{ij} \geqslant 1) \end{cases}$$

Conservative force (it could be any type of force)

$$\mathbf{F}_{ij}^{D} = -\gamma w^{D}(r_{ij})(\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij})\hat{\mathbf{r}}_{ij}$$

Dissipative force (mimics the thermal fluctuations)
[it cools the system]

$$\mathbf{F}_{ij}^{R} = \sigma w^{R}(r_{ij})\theta_{ij}\hat{\mathbf{r}}_{ij}$$

Random force
[it heats the system]

$\theta_{ij}(t)$ a randomly fluctuating variable with Gaussian statistics

$$\langle \theta_{ij}(t) \rangle = 0$$

$$\langle \theta_{ij}(t)\theta_{kl}(t') \rangle = (\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk})\delta(t - t')$$

$\mathbf{F}^{D}$ and $\mathbf{F}^{R}$ act along the two particles and conserve the linear and angular momentum. There is an independent random function for each *pair of particles*. The corrections may apply to non-bonded or/and bonded interactions.

DPD Thermostat

To have the correct canonical distribution function (constant NVT) the dissipative and random forces are related:

$$\omega_d(r_{ij}) = \left[\omega_r(r_{ij})\right]^2$$

Advantages

- It is local, Galelian invariant, and conserves the linear and angular momentum, and therefore preserves hydrodynamics,
- The equilibrium structure and elastic properties of the systems simulated by DPD are in good agreement with the experimental results.

Disavantages

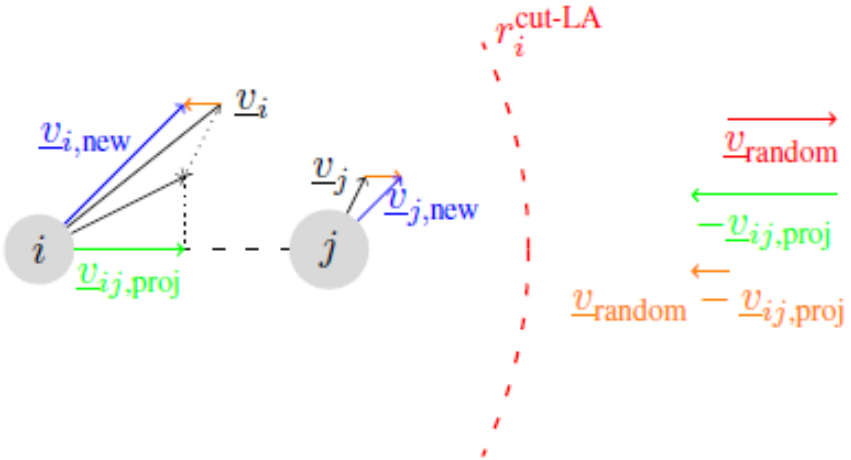
- It is not capable of controlling liquid properties such as viscousity and the diffusion constant,
- It is difficult to self-consistently integrate the stochastic equation of motion,
- The results of the simulations depend on the time step used in the simulation.

Lowe Andersen Thermostat

C.P. Lowe, *Europhys.Lett.*, 47 (1999) 145

An alternative approach to dissipative particle dynamics

For each pair of particles, the LA thermostat takes a new velocity from the Maxwell distribution, and with a given probability $\Gamma\Delta t$ corrects the component along the interparticle direction of the relative velocity of the two particles.



$$\underline{v}_i^{\text{new}} = \begin{cases} \underline{v}_i(t), & \Gamma\Delta t < \zeta \\ \underline{v}_i(t) + \dfrac{\mu_{ij}}{m_i}\left(v_{\text{rand}} - \left(\underline{v}_i - \underline{v}_j\right)\cdot\underline{\epsilon}_{ij}\right)\cdot\underline{\epsilon}_{ij}, & \Gamma\Delta t \geq \zeta\,, \end{cases}$$

$$\underline{v}_j^{\text{new}} = \begin{cases} \underline{v}_j(t), & \Gamma\Delta t < \zeta \\ \underline{v}_j(t) - \dfrac{\mu_{ij}}{m_j}\left(v_{\text{rand}} - \left(\underline{v}_i - \underline{v}_j\right)\cdot\underline{\epsilon}_{ij}\right)\cdot\underline{\epsilon}_{ij}, & \Gamma\Delta t \geq \zeta\,. \end{cases}$$

Lowe Andersen Thermostat

JÜLICH FORSCHUNGSZENTRUM

Advantages

- It conserves the linear and angular momentum, and it is Galelian invariant,
- It is a local thermostat, enhances the viscosity and preserves the hydrodynamics,
- Simply to implement in a Velocity Verlet scheme,
- It gives a correct equilibrium statistics, even for finite time steps,
- It is computationally more efficient than DPD, as it allows longer time-steps.

Disavantages

- What should be the order of the „collision“ corrections (a random order???)
- The LA algorithm is difficult to be efficiently parallelized.

NOTE: Within short time-step limit, the DPD and LA thermostats are equivalent.

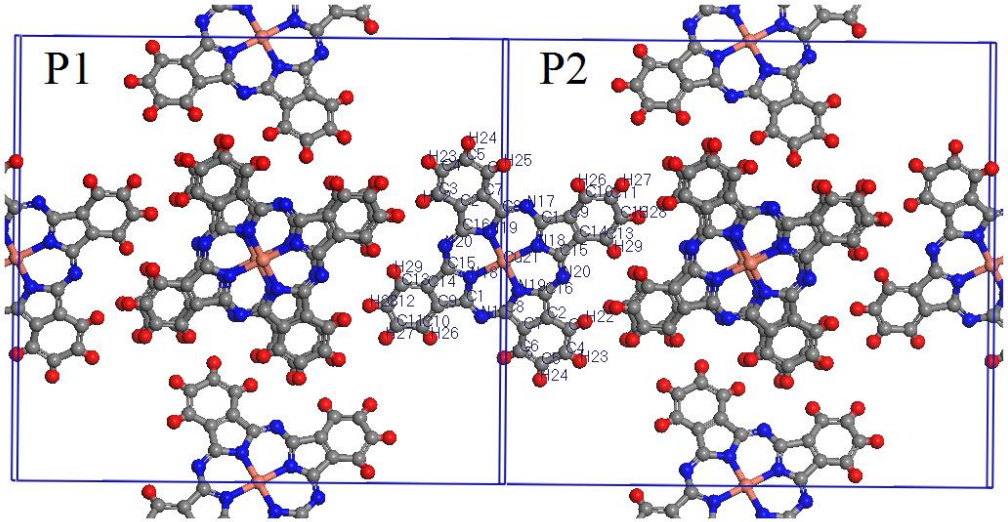
LA Thermostat – serial algorithm

For each pair of particles ($r_{ij} < r_{LA\_cut}$)

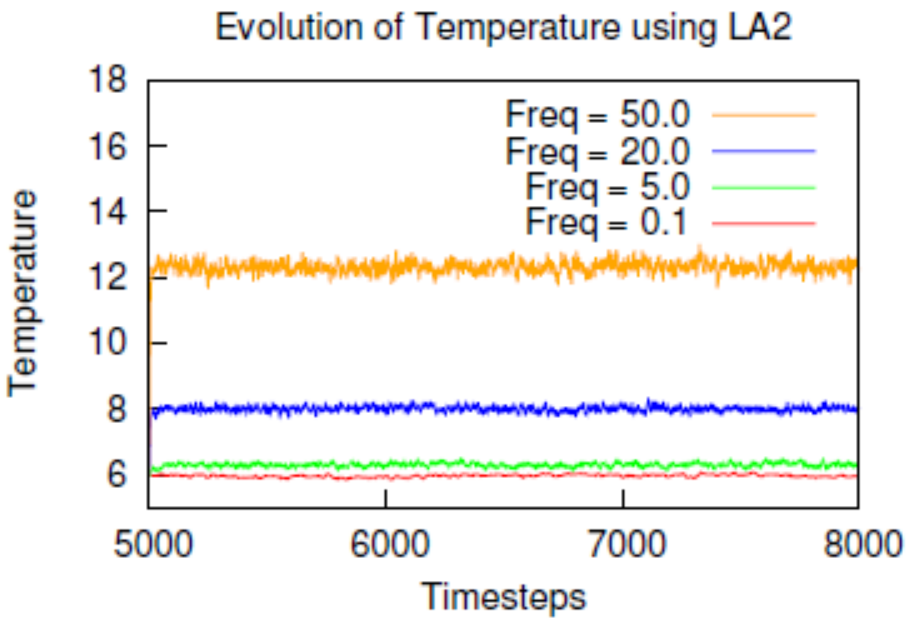
1. Determine the unit vector $\underline{\epsilon}_{ij}$ along the line connecting the centers of particle i and j.
2. Project the vector $(\underline{v}_i - \underline{v}_j)$ on the vector $\underline{\epsilon}_{ij}$ to obtain $\underline{v}_{ij,\mathrm{proj}} = \left[\left(\underline{v}_i - \underline{v}_j\right)\underline{\epsilon}_{ij}\right]\underline{\epsilon}_{ij}$.
3. Pick a random velocity $v_{\mathrm{rand}} = \left[\zeta' \sqrt{(k_B T)/(\mu_{ij})}\right]$ from the Boltzmann distribution corresponding to a given temperature. ζ' is gaussian random number and $\mu_{ij} = {(m_i m_j)}/{(m_i + m_j)}$ is the reduced mass of the pair of the two particles, where m_i and m_j are their masses.
4. Add $\left(v_{\mathrm{rand}} \cdot \underline{\epsilon}_{ij} - \underline{v}_{ij,\mathrm{proj}}\right)$ to $\underline{v}_i$ and subtract it from $\underline{v}_j$, to conserve the linear and the angular momenta.

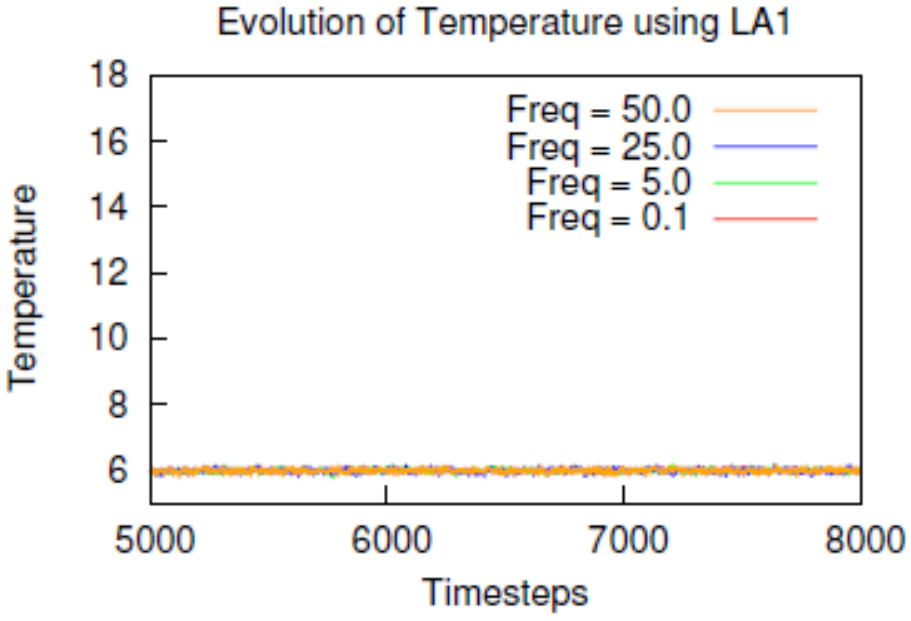
LA Thermostat - Parallelization

The correction order is important: the velocity corrections on the processor boundaries cannot be done synchronously.



LA Thermostat – Two algorithms





Felix Uhl, Ruhr-Universität Bochum

Efficient Communication Schemes for Stochastic Thermostats in parallel MD Simulations

JSC Guest Student Colloquium, September 26/27, 2012

LA Thermostat – Parallelization within IBIsCO

The original implementation :

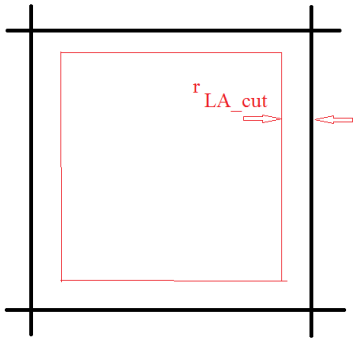
1. The velocities and the additional information about the particles are communicated from each processor to the master processor using seven MPI_GATHERV calls.
2. All elements in the vectors are rearranged in a way such that they are in order considering their absolute index.
3. The velocity correction is successively performed for each processor domain on the master processor.
4. The velocities are sent back to the host processors using three independent sending processes.

Problems:

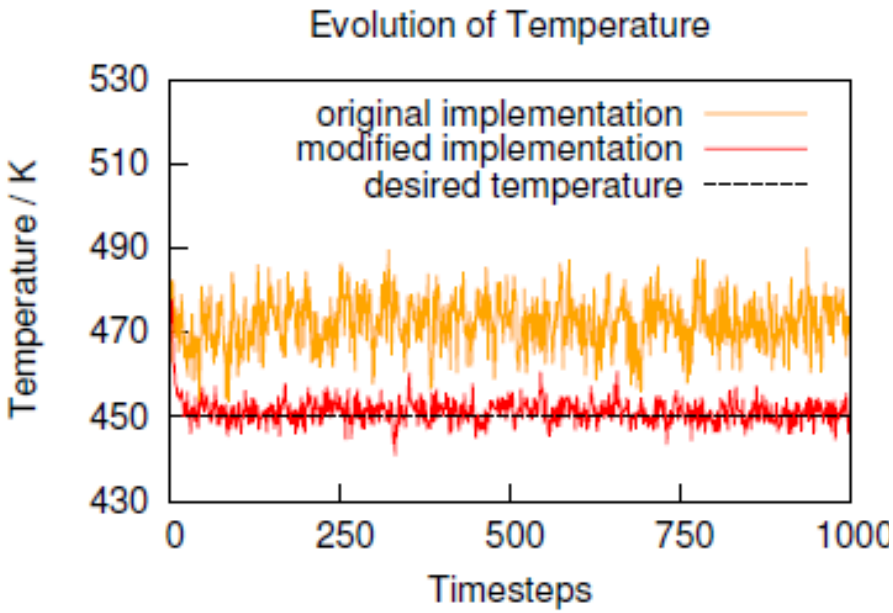
A large amount of memory is allocated on the master processor for the positions, velocities and indices of all the particles in the system. The serial character of this quite expensive method leads to a significant slowdown and an inefficient hardware usage.

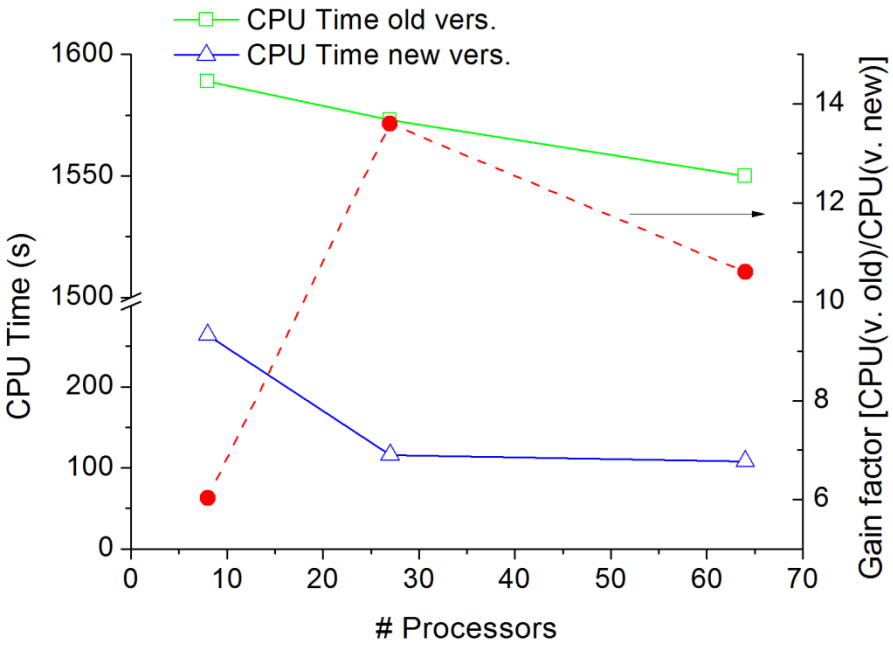
Solution:

- Do the velocity corrections for the particles localized in interior domains ($r<r_{LA\_cut}$) localy on processors,
- Communicate the corrected velocities of the particles from LA boundaries to the half of neighbour processors, and do the velocity corrections there,
- Communicate back to the parent processors the corrected velocities.



Lowe Anderson Thermostat





Comparison of the temperature evolution using the original and the modified LA implementation. Target temperature is 450K.

Comparison of the simulation times using the original and the modified LA implementation with a varying amount of processors.

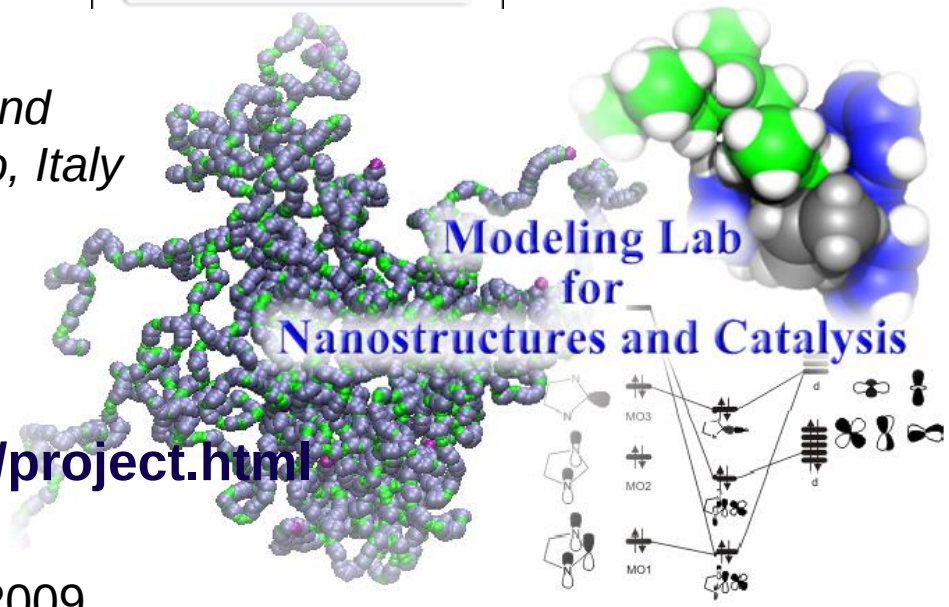
OCCAM – Support Activity

OCCAM Molecular Dynamics

JÜLICH FORSCHUNGSZENTRUM

Dr. Giuseppe Milano

Modeling Laboratory for Nanostructures and Catalysis - MoLNaC, University of Salerno, Italy



Code Homepage:
https://www.molnac.unisa.it/occam/project.html

Milano G., Kawakatsu T., *J.Chem.Phys.*, 2009
Hybrid particle-field molecular dynamics simulations for dense polymer systems

Milano G., Kawakatsu T., *J.Chem.Phys.*, 2010
Pressure calculation in hybrid particle-field simulations

Zhao Y., De Nicola A., Kawakatsu T., Milano G., J. Comput.Chem.,2012

Hybrid Particle-Field Molecular Dynamics Simulations: Parallelization and Benchmarks,

OCCAM

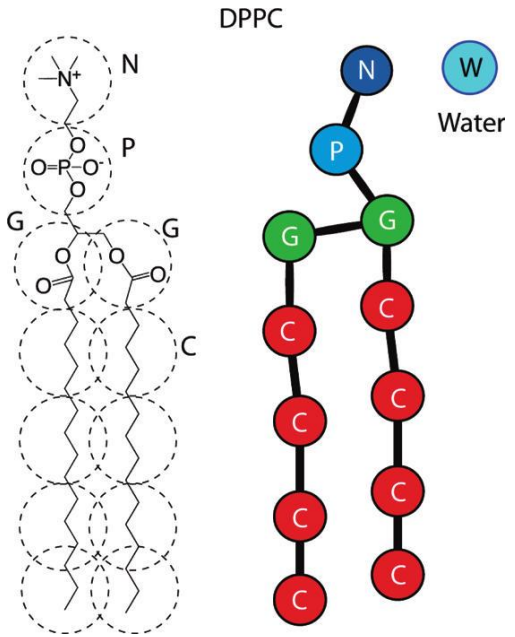
Hybrid Particle-Field-Theoretic Molecular Dynamics Simulations

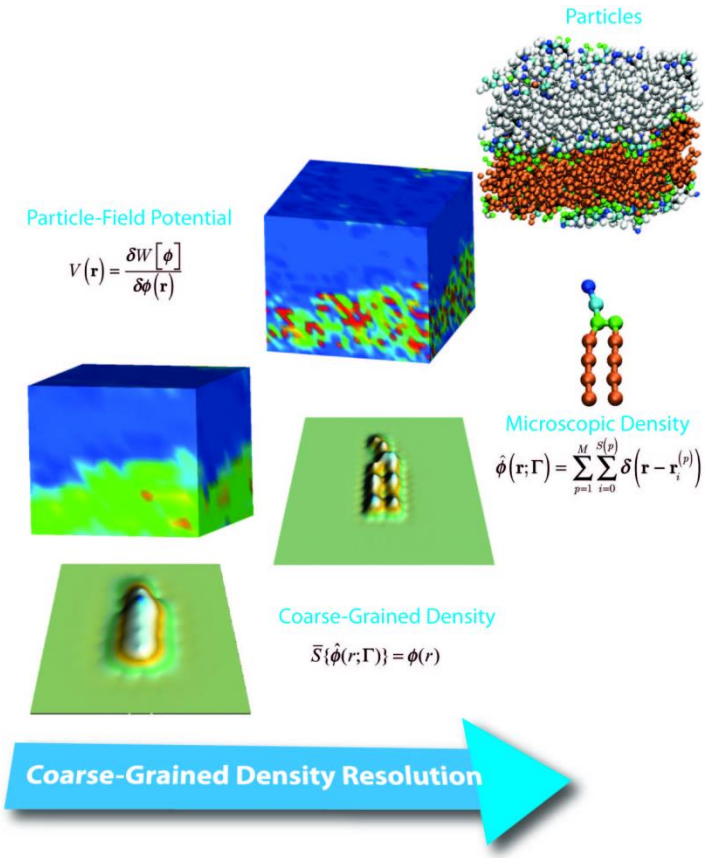
Basic Theory and Implementation

J.Chem.Phys., 2009 (130) 214106; 2010 (133) 214102.

Parallel Implementation and Code Description

J.Comput.Chem., 2012 (33) 868.





OCCAM: Particle – Density Field Interaction

The evaluation of the non-bonded forces between particles is replaced by an evaluation of an external potential dependent on the local density in the point where a particle is situated. The mean field potential acting on a particle of type A, in interaction with a particle of type B, at position r is given by

$$V_A(\mathbf{r}) = k_B T[\chi_{AA}\phi_A(\mathbf{r}) + \chi_{AB}\phi_B(\mathbf{r})] + \frac{1}{\kappa}(\phi_A(\mathbf{r}) + \phi_B(\mathbf{r}) - 1).$$

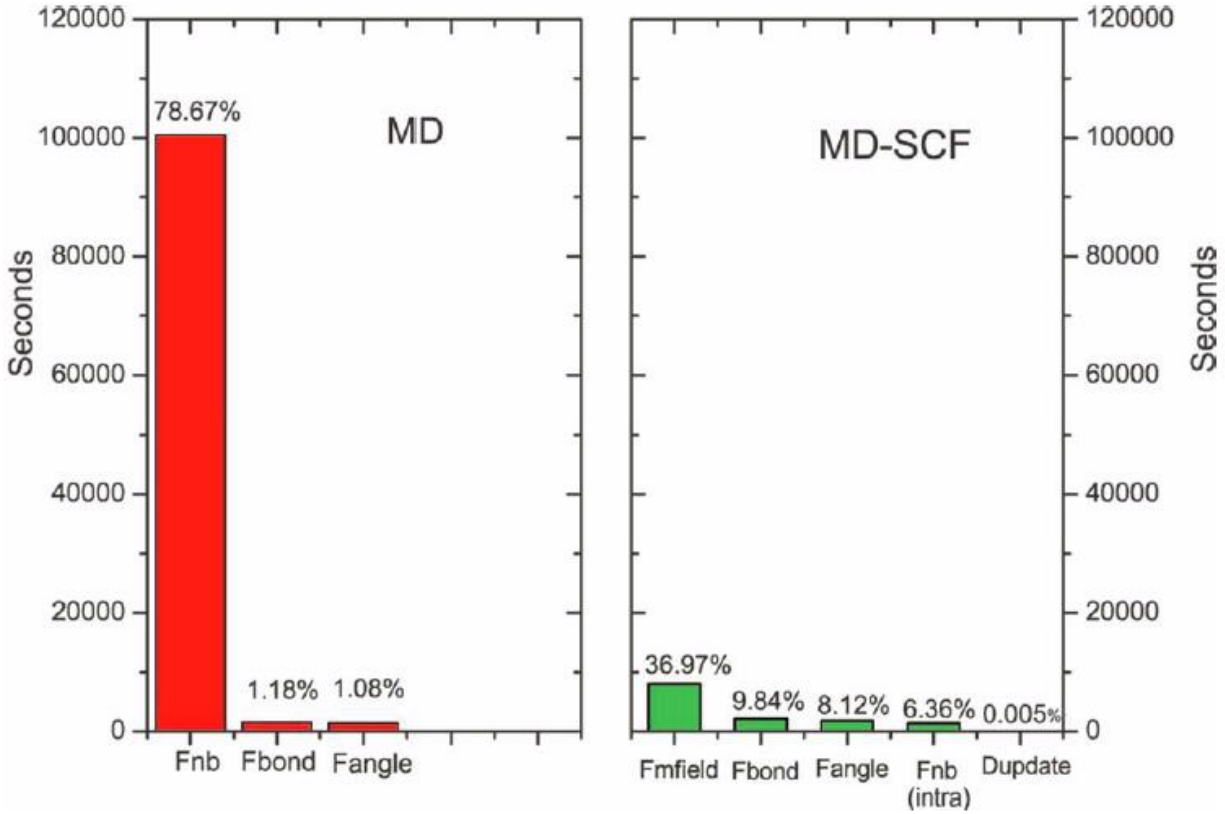
The mean field force of a particle of type B on a particle of type A is given by

$$F_A(\mathbf{r}) = -\frac{\partial V_A(\mathbf{r})}{\partial \mathbf{r}} = -k_B T\left(\chi_{AA}\frac{\partial \phi_A(\mathbf{r})}{\partial \mathbf{r}} + \chi_{AB}\frac{\partial \phi_B(\mathbf{r})}{\partial \mathbf{r}}\right) - \frac{1}{\kappa}\left(\frac{\partial \phi_A(\mathbf{r})}{\partial \mathbf{r}} + \frac{\partial \phi_B(\mathbf{r})}{\partial \mathbf{r}}\right)$$

where:

- $\phi_A(\mathbf{r})$ is the density of particles of type A in the position **r**
- χ_{AB} is the mean field parameters for the interaction of a particle of type A with the density fields determined by particles of type B (obtained from the Flory-Huggins approach), and
- κ is the system compressibility (assumed to be sufficiently small)

OCCAM

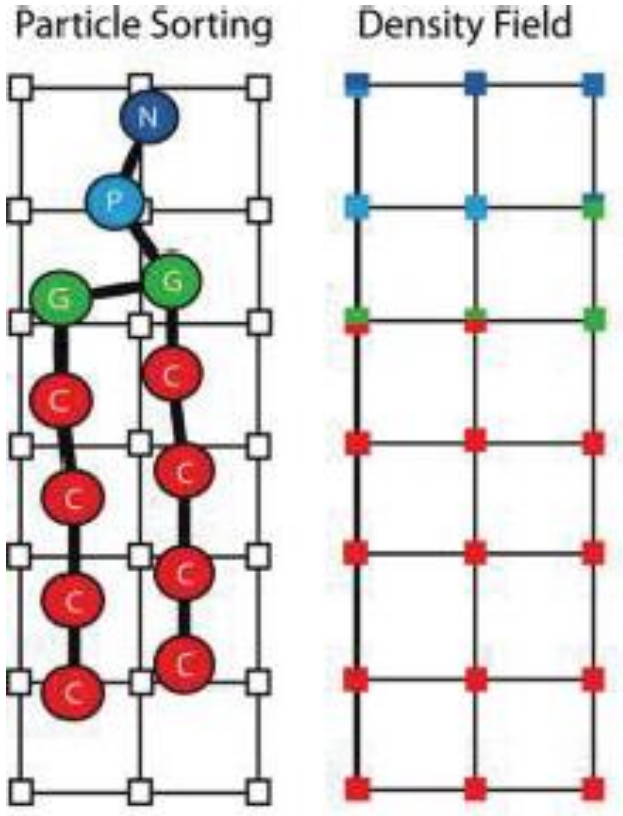


Comparison of serial MD and MD-SCF profiles using gprof v2.17 profiler. The system RC1 including 30 kPart has been simulated for 1,000,000 time steps with update frequency Δt_{update} 300 time steps. The computational times (in second) corresponding to nonbonded, particle-field, bond, and angle forces are shown.

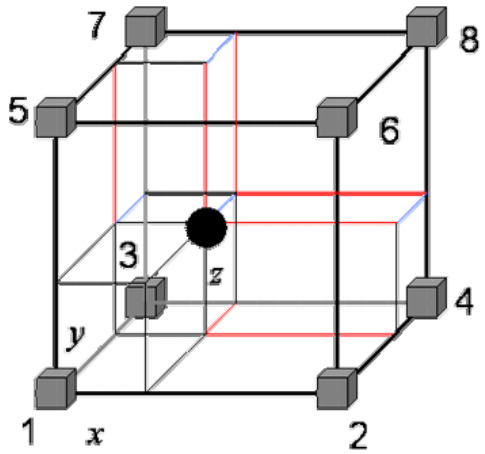
Y. Zhao, A. De Nicola, T. Kawakatsu, G. Milano, J.Comput.Chem., 33 (2012) 868
Hybrid Particle-Field Molecular Dynamics Simulations: Parallelization and Benchmarks

OCCAM

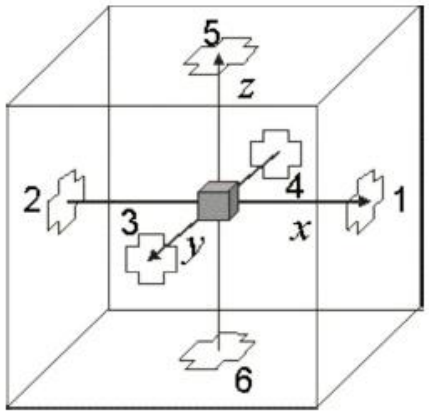
$$\hat{\phi}(\mathbf{r};\Gamma) = \sum_{p=1}^{M} \sum_{i=0}^{S(p)} \delta\left(\mathbf{r} - \mathbf{r}_i^{(p)}\right)$$



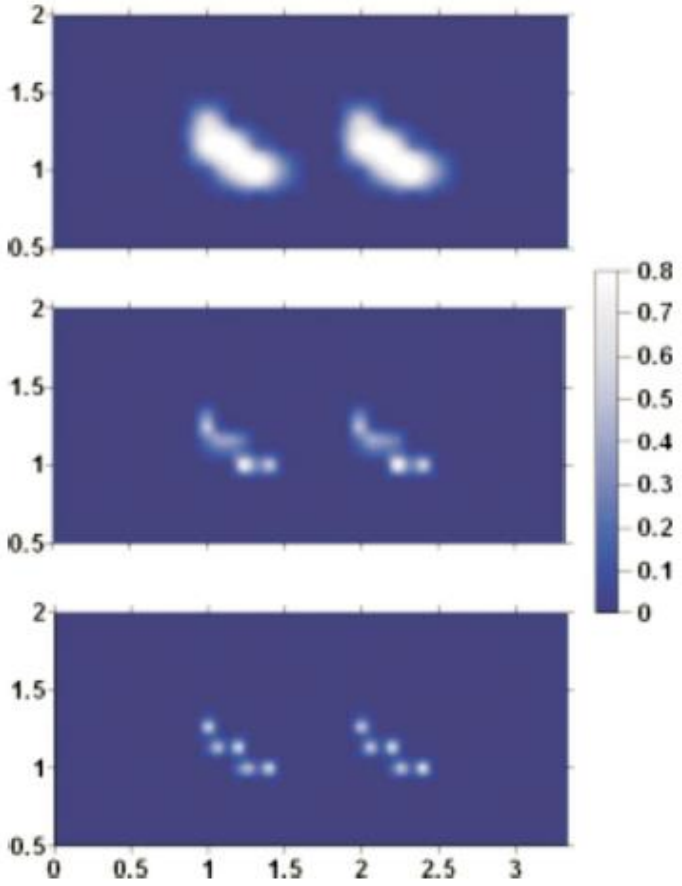
OCCAM - Density Field



Particle fraction assignment for the case of 3D lattice.



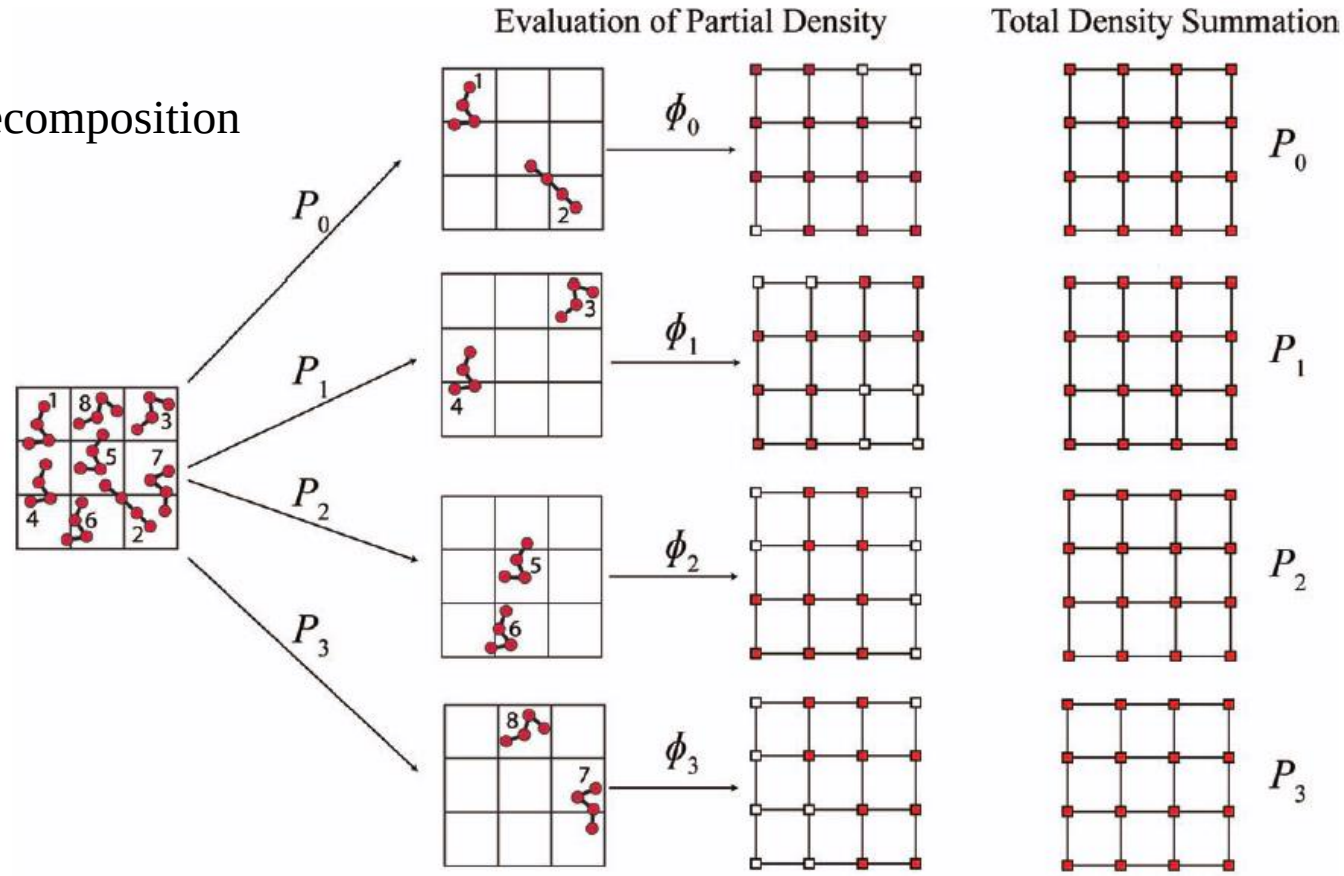
The density gradients are defined on the center of each face staggered lattice points of the cube surrounding a density lattice point.



2D density maps corresponding to a test configuration containing two *n-pentane molecules in all trans conformation* laying in the *xy plane at different grid resolutions.* ***The grid size is a parameter of the potential.***

OCCAM - Parallelization

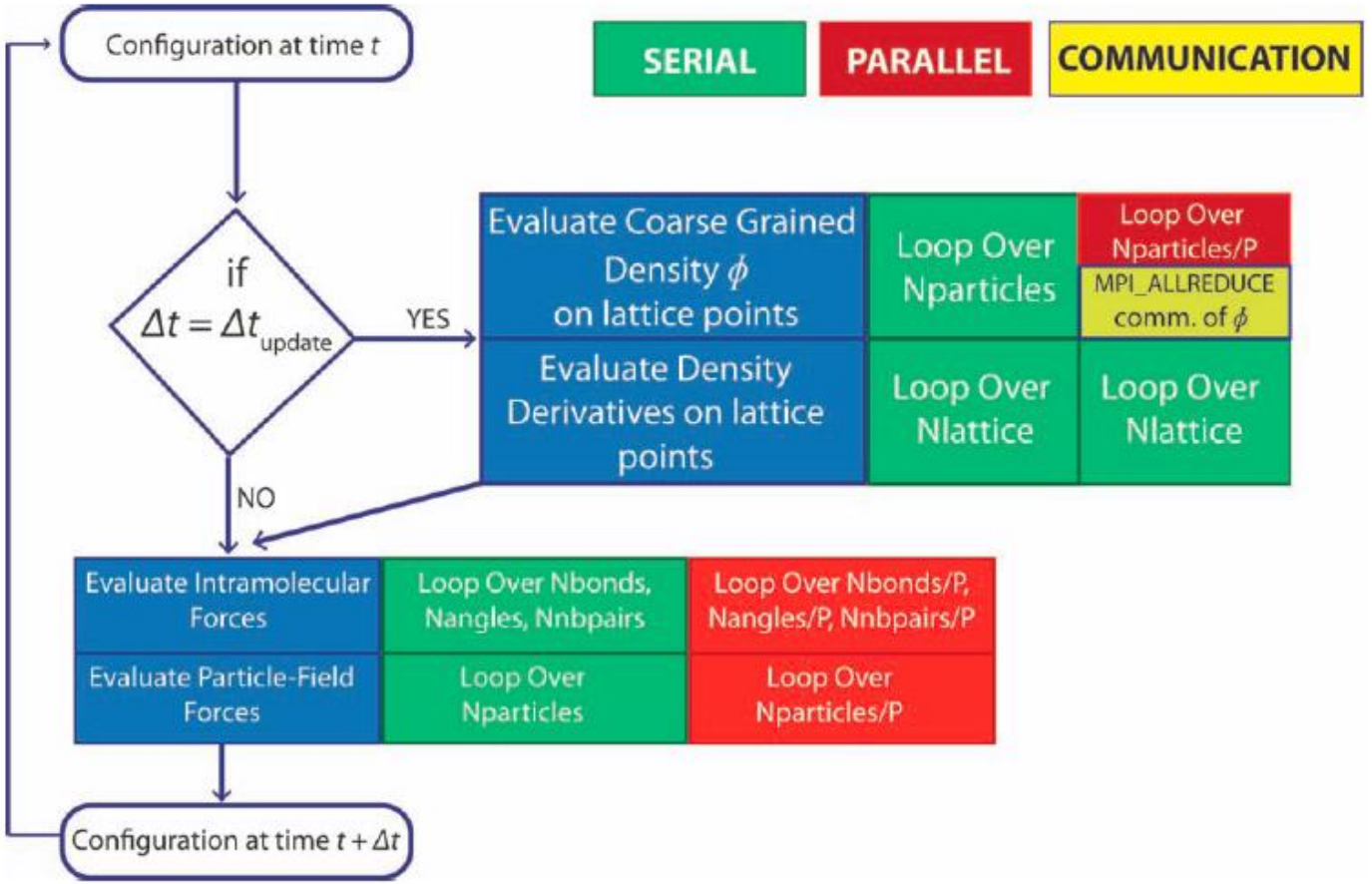
Particle decomposition scheme



Density-update parallelization shown in an example of eight molecules assigned to four processors

Y. Zhao, A. De Nicola, T. Kawakatsu, G. Milano, J.Comput.Chem., 33 (2012) 868
Hybrid Particle-Field Molecular Dynamics Simulations: Parallelization and Benchmarks

OCCAM - parallelization



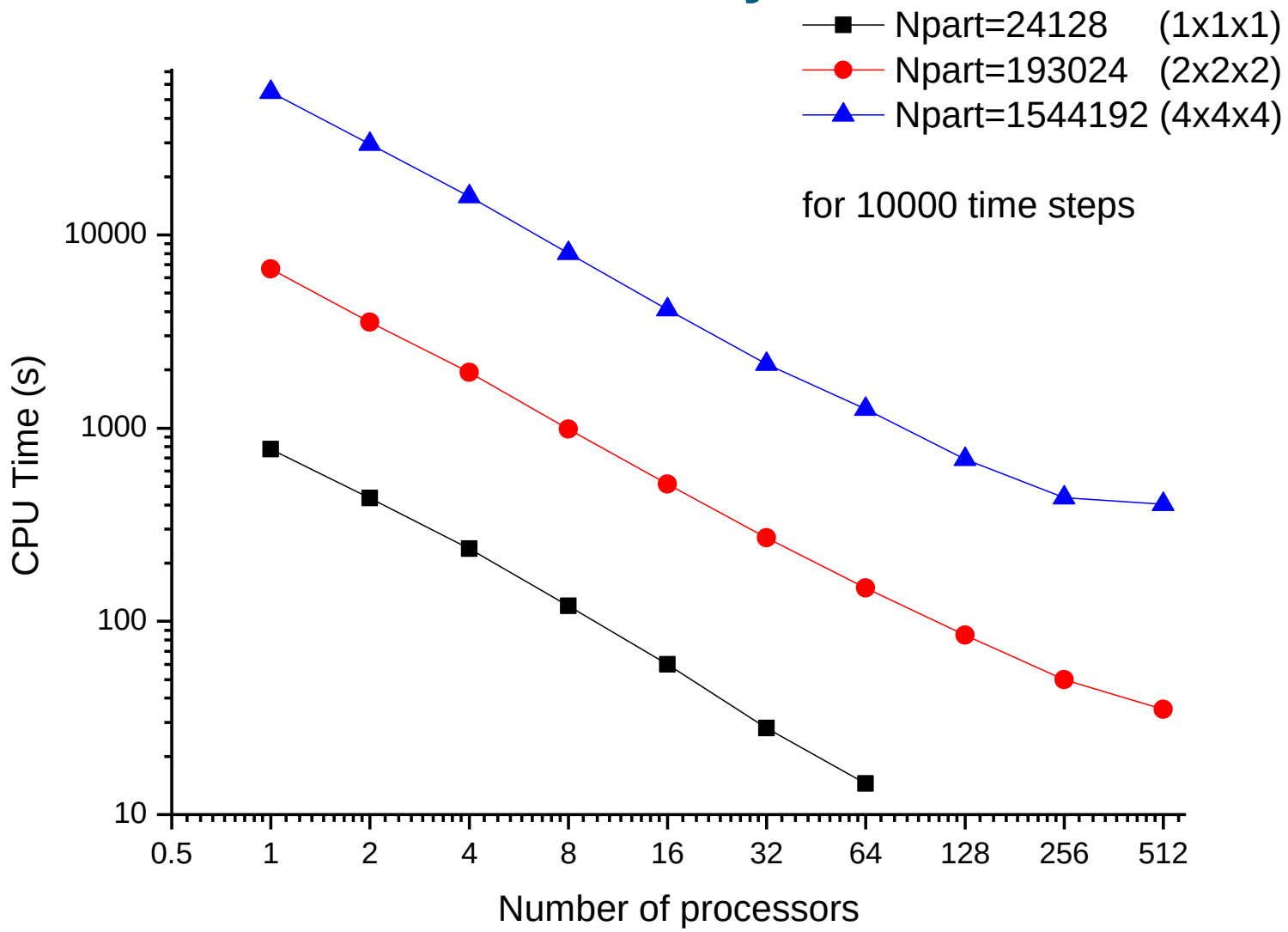
Add on the grid the density from different processors using MPI_ALLREDUCE

Y. Zhao, A. De Nicola, T. Kawakatsu, G. Milano, J.Comput.Chem., 33 (2012) 868
Hybrid Particle-Field Molecular Dynamics Simulations: Parallelization and Benchmarks

OCCAM – Code Optimization

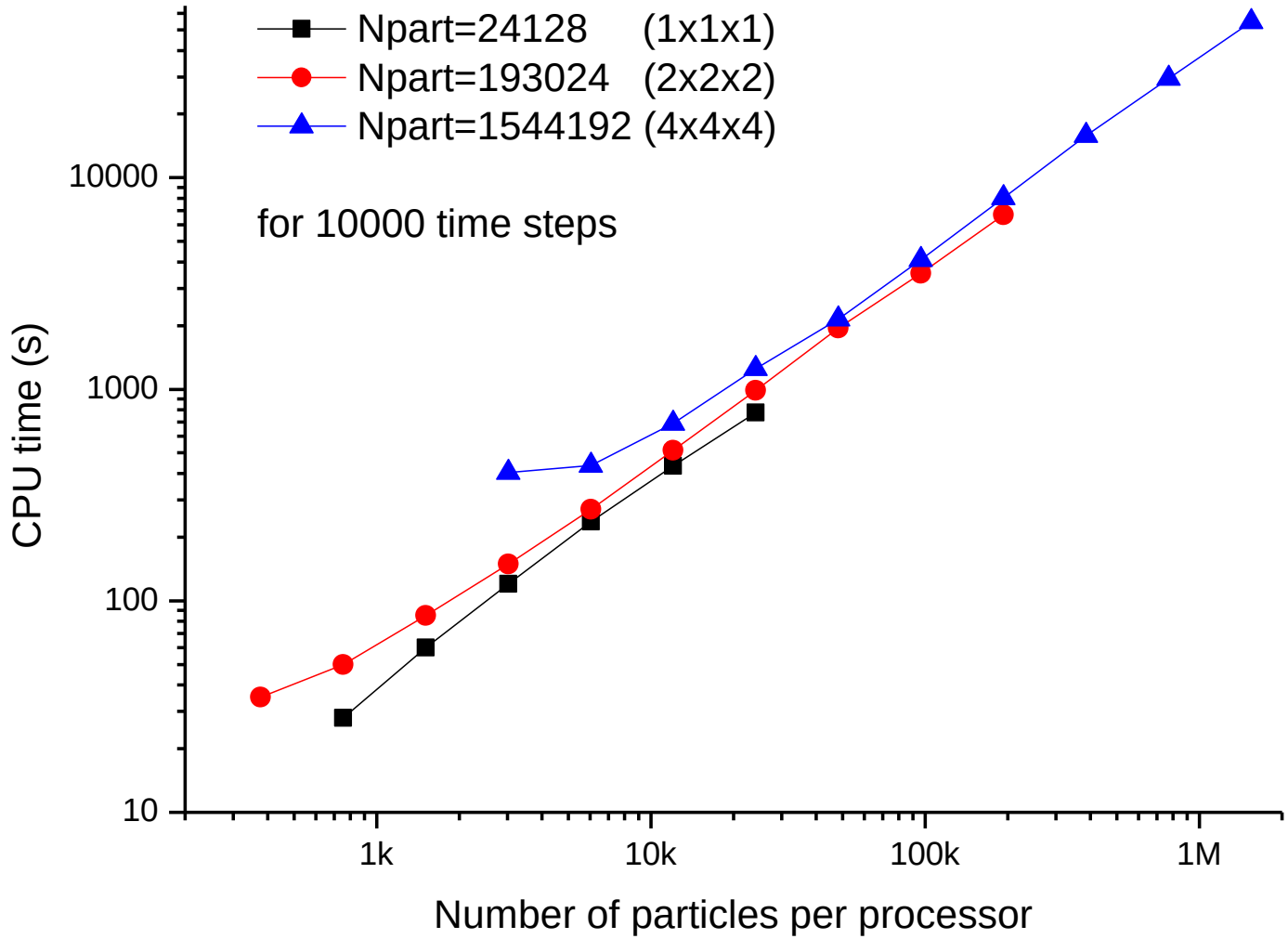
- We changed the way to open these files, to decrease the number of files that are simultaneously opened,
- The code structure was optimized. To avoid the problems generated by the very large temporary matrices vertex and cell, the call of the subroutine indexing can be moved after the subroutine getdata that will read the parameters of the grid (mx,my,mz)
- The memory usage was rationalized . We tried to reduce the size of allocated memory by eliminating some of the large arrays (see Table 1). The size of matrix d_pos was reduced from (natmpia*ntype,ntype) to (natmpia,ntype), and the size of the vertices index was reduced from v(8,ncell*ntype) to v(8,ncell).
- For large systems, the size nvert*ntype of the buffer density matrix dmpi is too large to be efficiently used in the data transfer operations MPI_ALLREDUCE. The size of the array dmpi can be reduced to nvert by calling the MPI_ALLREDUCE separately for each species of the particles.

OCCAM – Parallel Efficiency



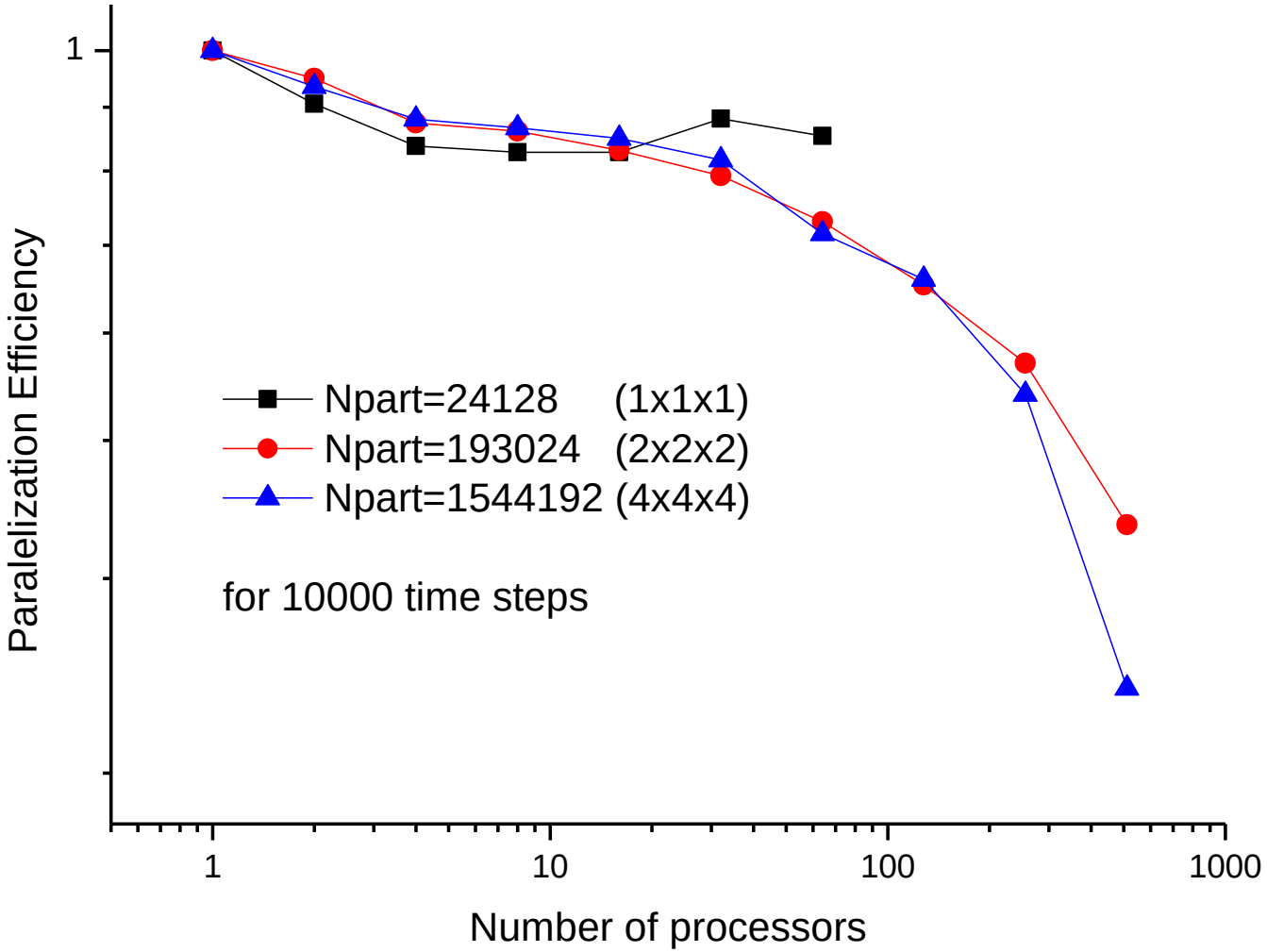
CPU Time versus number of processors for several system sizes.

OCCAM – Parallel Efficiency



CPU Time versus number of particles per processor for several system sizes.

OCCAM – Parallel Efficiency



Parallelization efficiency versus the number of processors for several system sizes

OCCAM – New parallelization

- The grid is decomposed on domains assigned to the processors,
- Each processor manages the particles localized on ist associated domain,
- The densities and gradients projected on the grid are localy calculated,
- The particles localized on the processor boundaries are sent to half of its neighbour processors,
- Each processor calculate the bonded and nonbonded forces on the particles from ist own domain and imported from neighbours,
- Each processor exports the nonbonded particle-field forces of the imported boundary particles to their parent processors.
- Each processor evaluate add the imported forces to the forces calculated localy and integrate the equation of motion of the particles localized to their domains,
- With some frequency, the particles moved from their initial domains are assigned to the new parent processors.