Workload Estimation and Load-Balancing of Discrete Element Method

EUROHPC USER DAY 2023 Brussels 11.12.23

EuroHPC Joint Undertaking

Project: Workload Estimation and Load-Balancing of Discrete Element Method
EuroHPC used: MeluXina

Speaker: Xavier BESSERON *(Uni. of Luxembourg)*

Outline

Extended Discrete Element Method

- What is XDEM?

Parallelization of XDEM

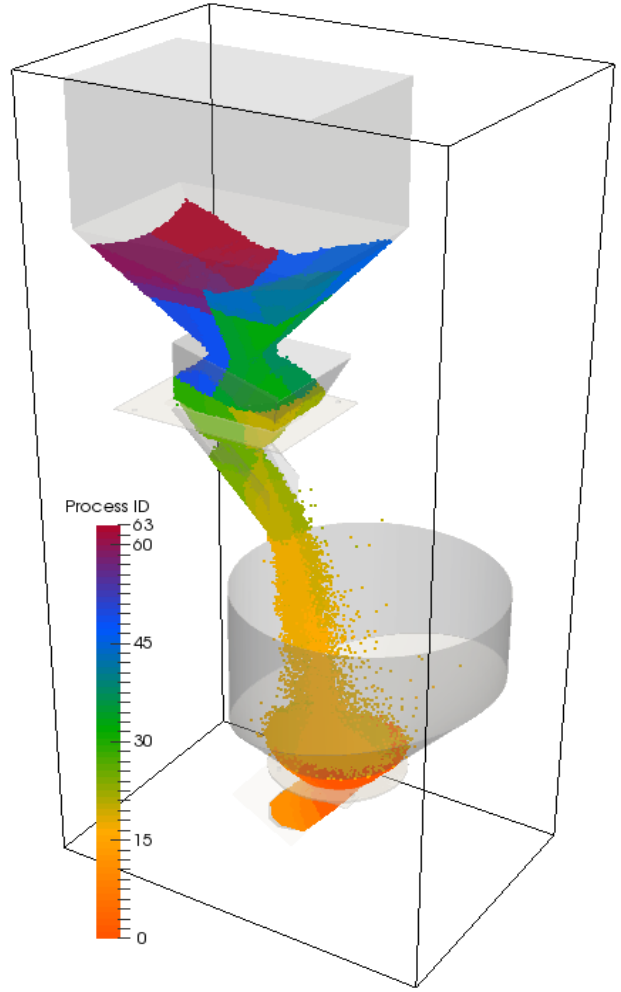
- Domain Decomposition with MPI
- Load-Balancing

Workload Estimation for XDEM

- Toward better Load-Balancing

Preliminary Results

- Load Estimation and Imbalance



Extended Discrete Element Method

What is XDEM?

What is XDEM?

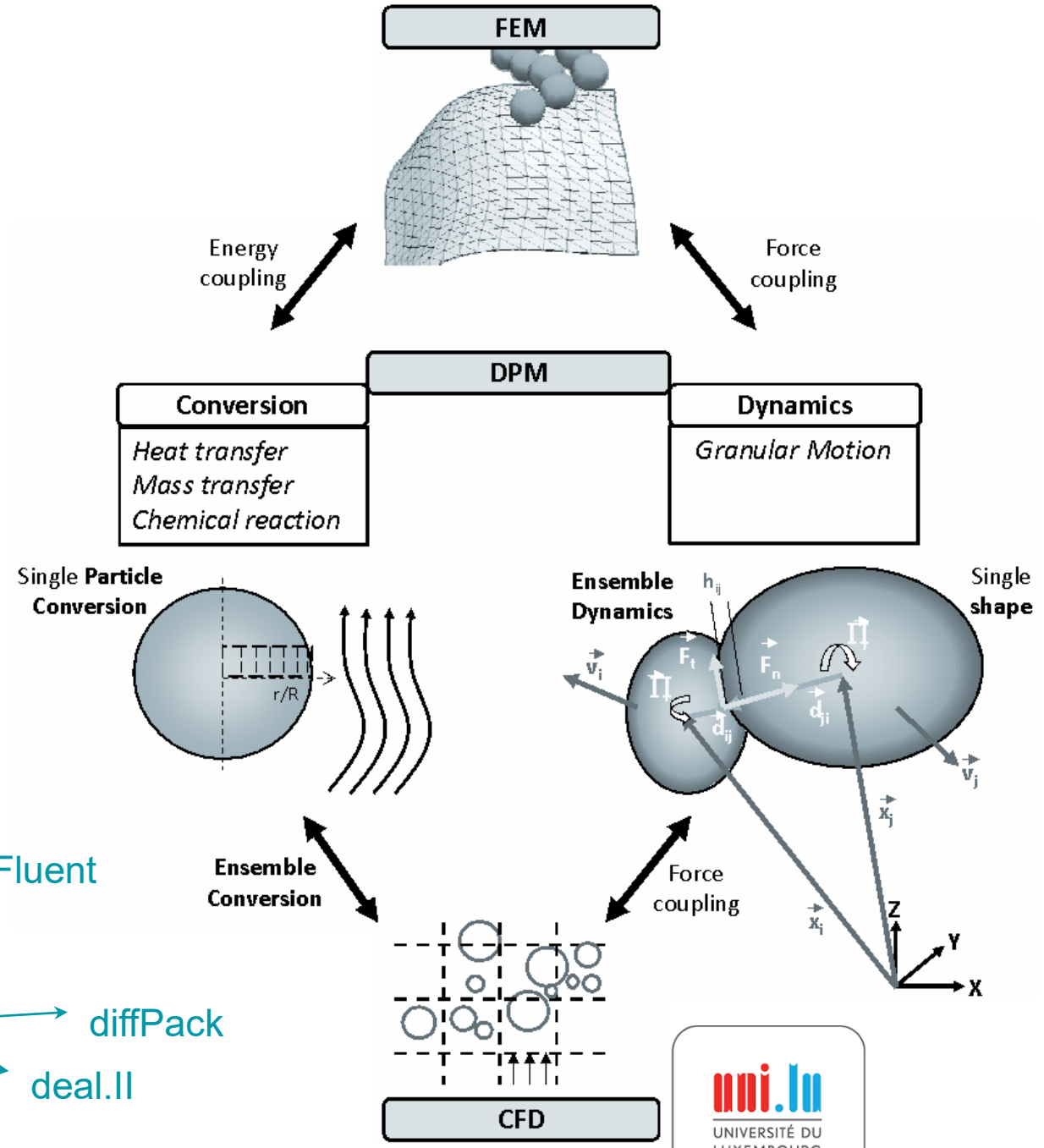
eXtended Discrete Element Method

Simulation software for

Particles Dynamics
- Force and torques
- Particle motion

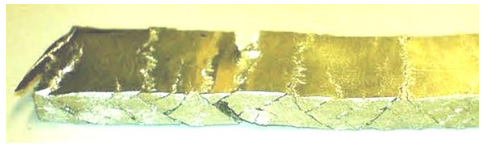
Particles Conversion
- Heat and mass transfer
- Chemical reactions

Coupled with
- Computational Fluid Dynamics (CFD)
  - OpenFOAM
  - ANSYS Fluent
- Finite Element Method (FEM)
  - diffPack
  - deal.II
  - CalculiX

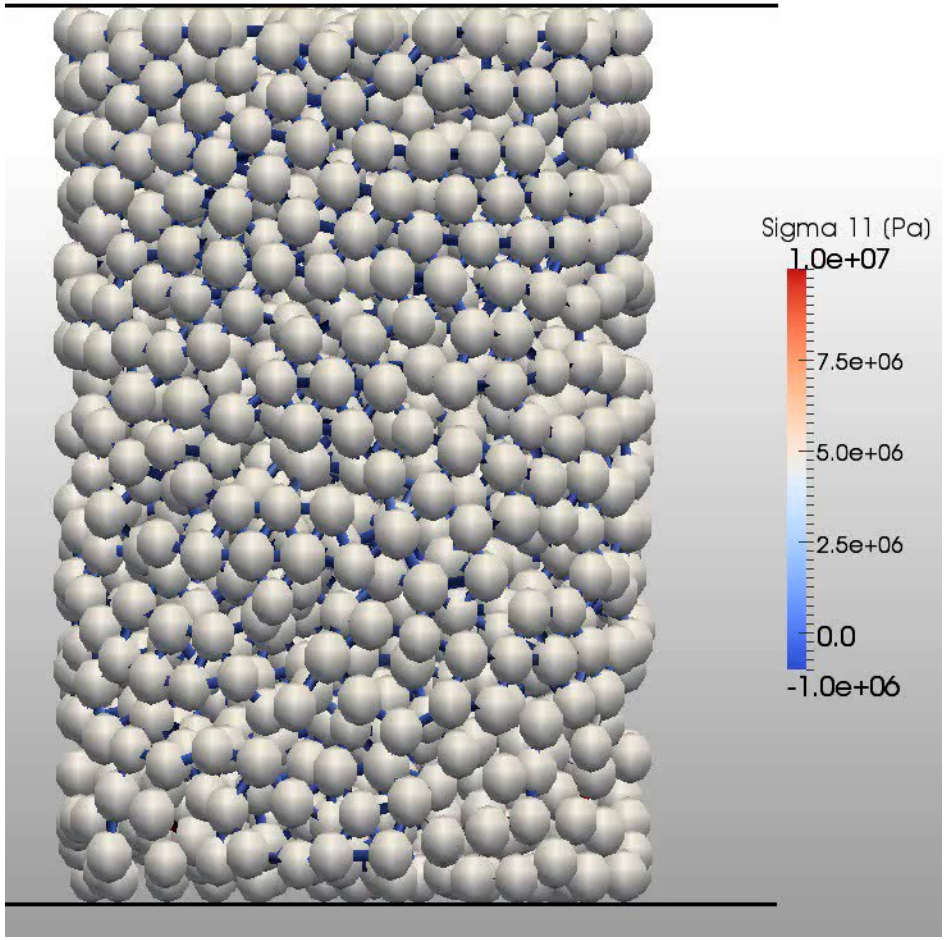


https://luxdem.uni.lu/software/

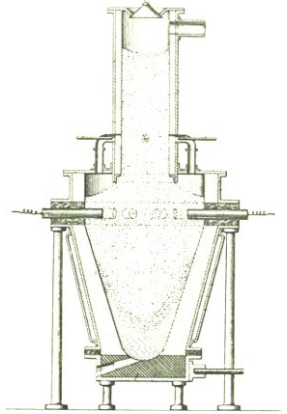
Application Examples: XDEM

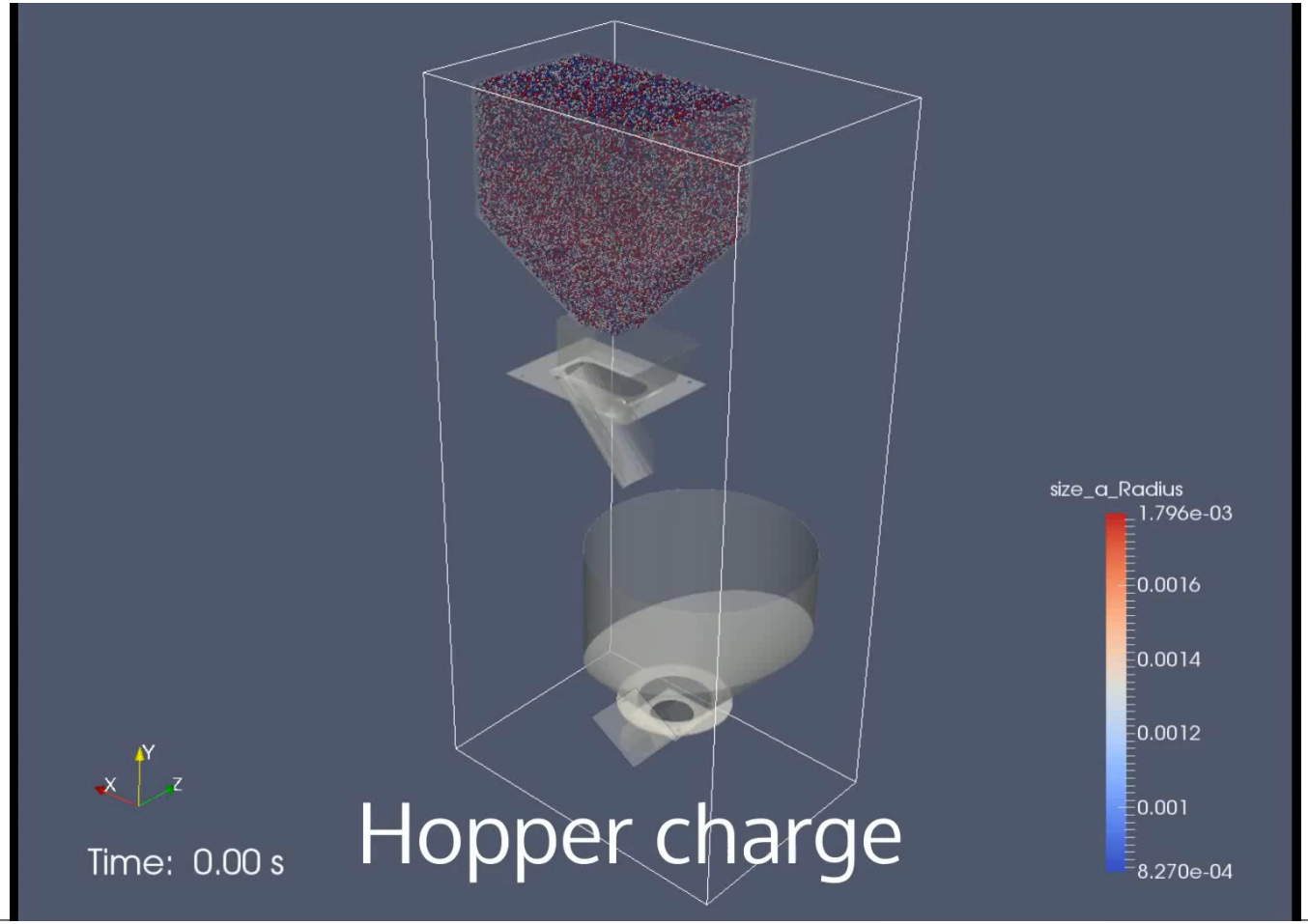


Brittle Failure

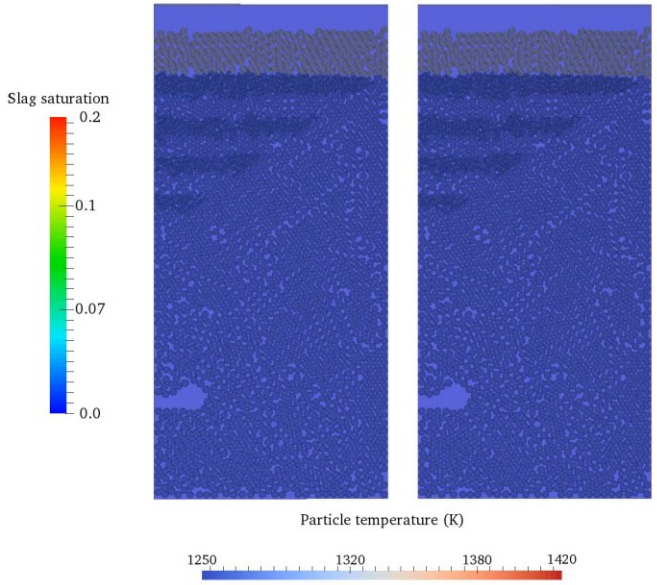


Hopper charge and discharge



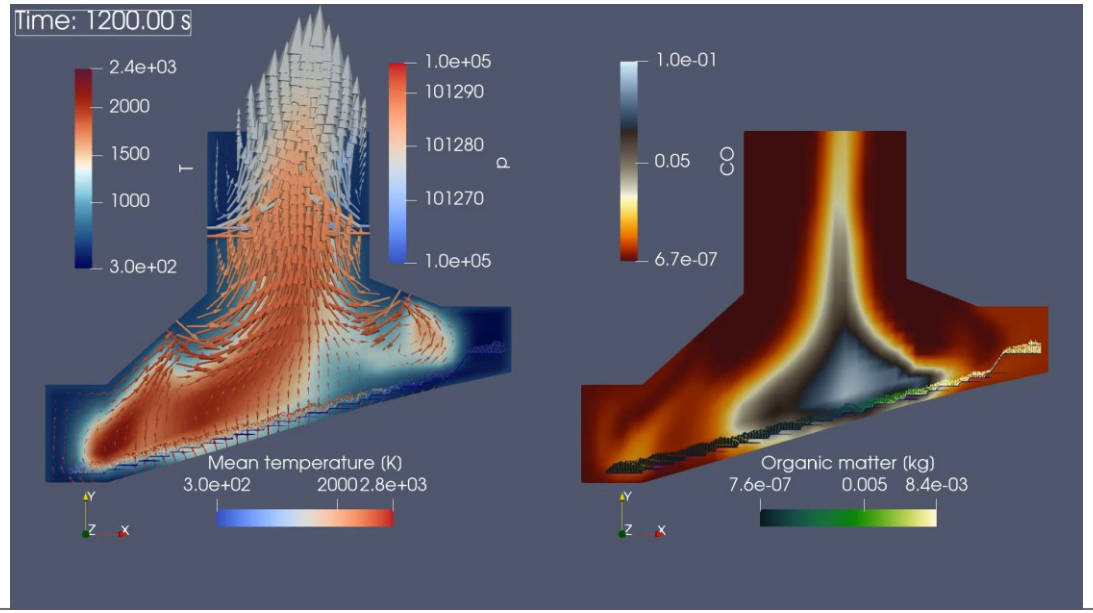


Application Examples: XDEM coupled with CFD

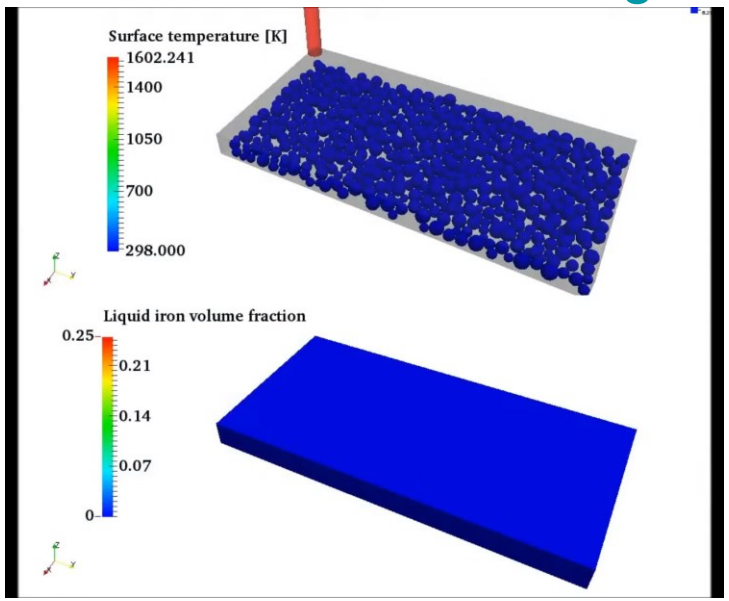


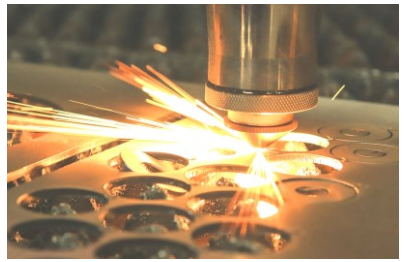
Iron & Slag production in a Blast Furnace

Wood Conversion in a Biomass Furnace



Selective Laser Melting in Additive Manufacturing





Parallelization of XDEM

Domain Decomposition with MPI

and Load-Balancing

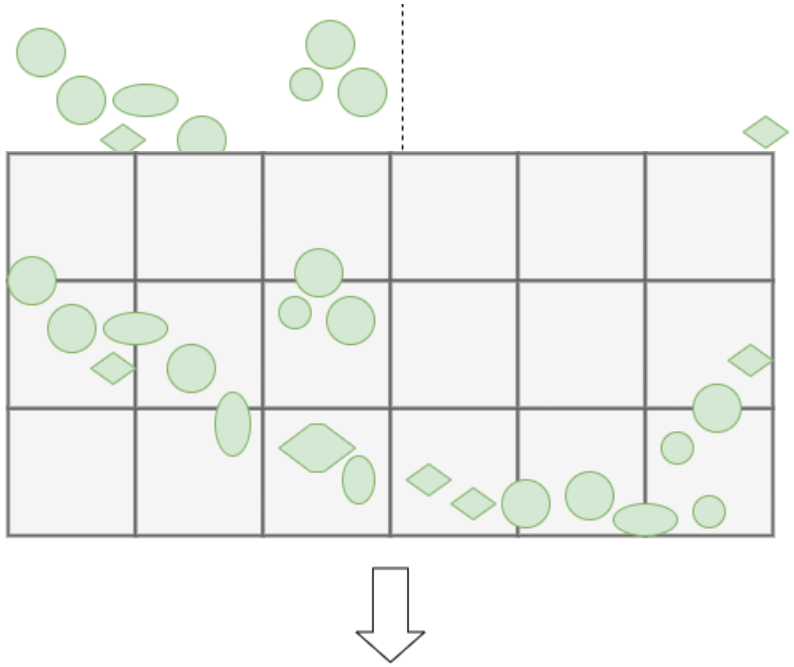
UNIVERSITÉ DU LUXEMBOURG

Domain Decomposition in XDEM

Decomposing the set of particles?

- Particles move during the simulation
- Neighborhood relations change
- Create undetected dependencies

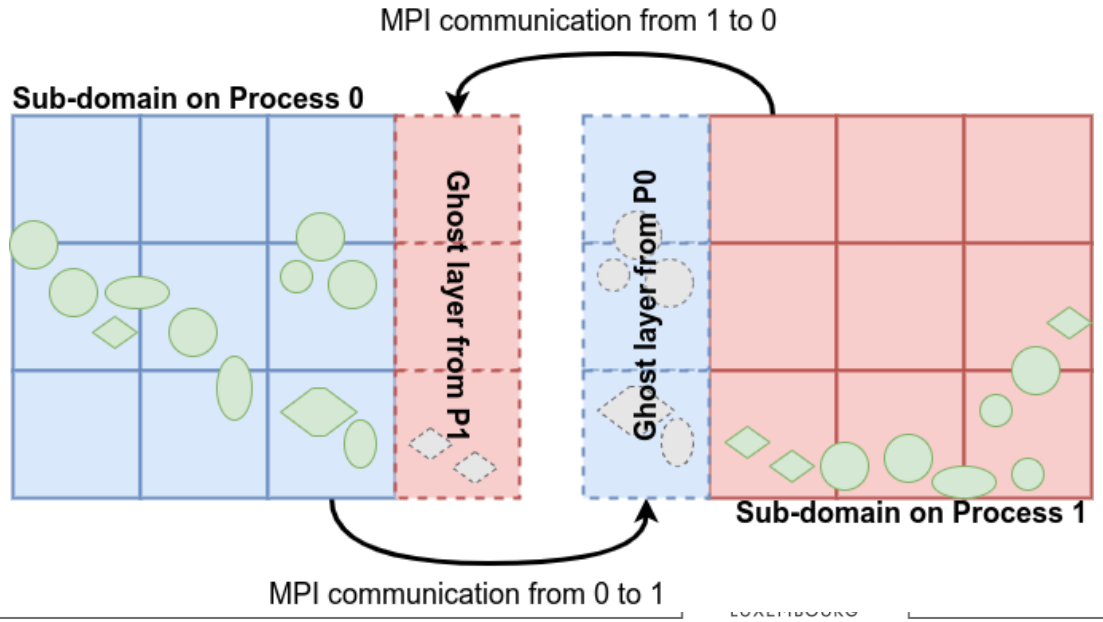
→ Would require frequent re-partitioning



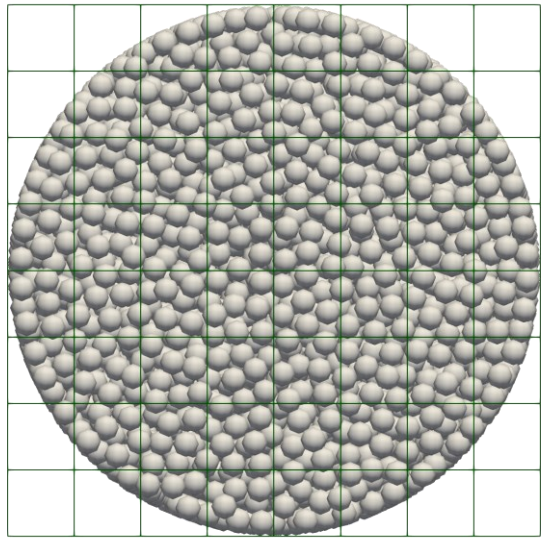
Use a static regular grid to ‘store’ particles

- Find location of a particle in constant time
- Size of grid cells adapted for collision detection
- No missing communication

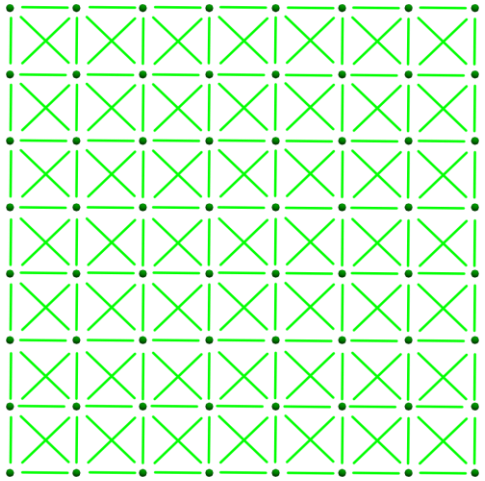
→ Re-partitioning only required in case of imbalance



Partitioning and Load-Balancing for XDEM

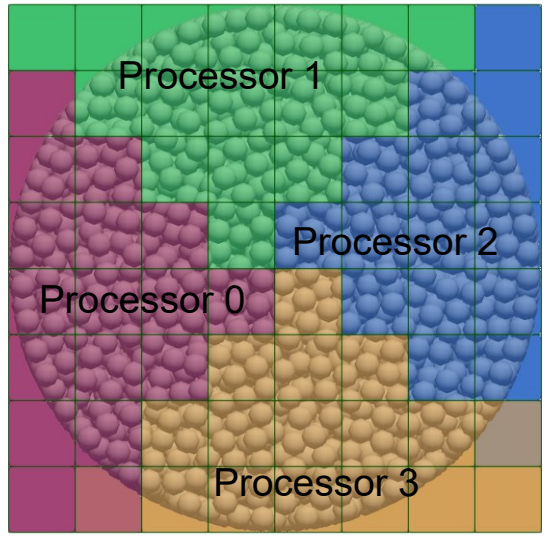


Particles in the cell grid



From grid to graph

- Node ← Cell
- Node weight ← *f*(nb particles)
  ~ Computation cost
- Edge ← Neighborhood relation
- Edge weight ← *g*(nb particles)
  ~ Communication cost
- Node Coordinates (topologic approaches)



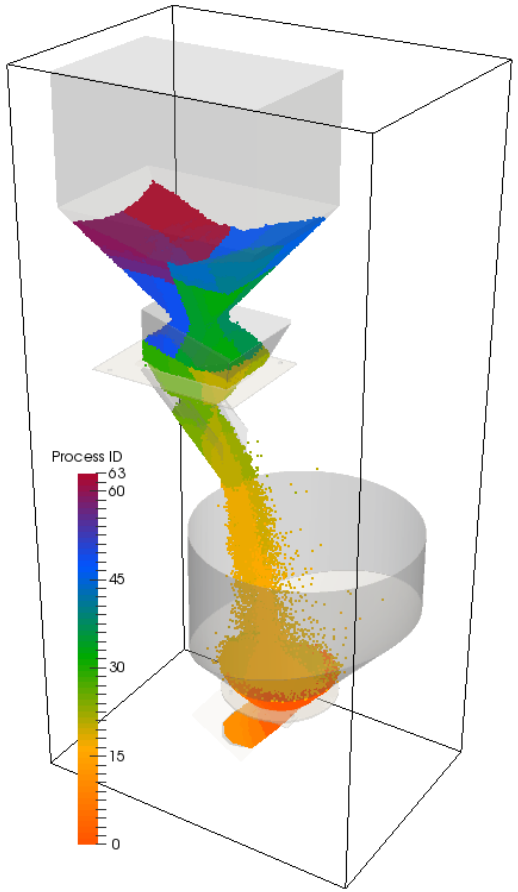
Partitioning algorithm

- Orthogonal Recursive Bisection
- METIS
- SCOTCH
- Zoltan PHG, RCB, RIB, ...
- etc.

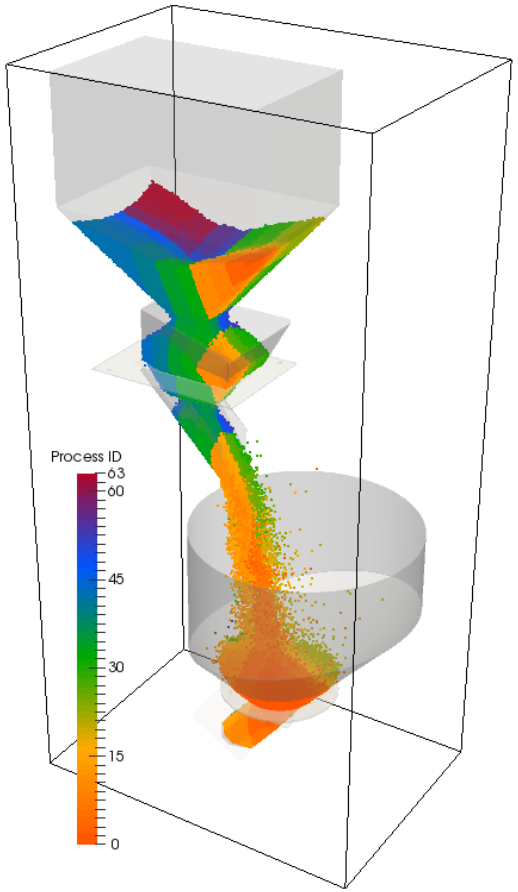
Objectives

- Balance the computation cost
- Minimize the communication cuts

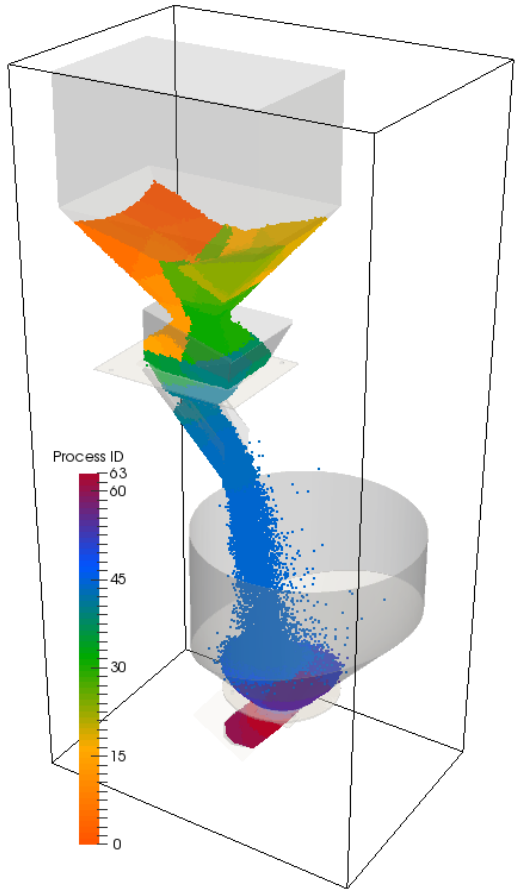
Example of Load-Balancing



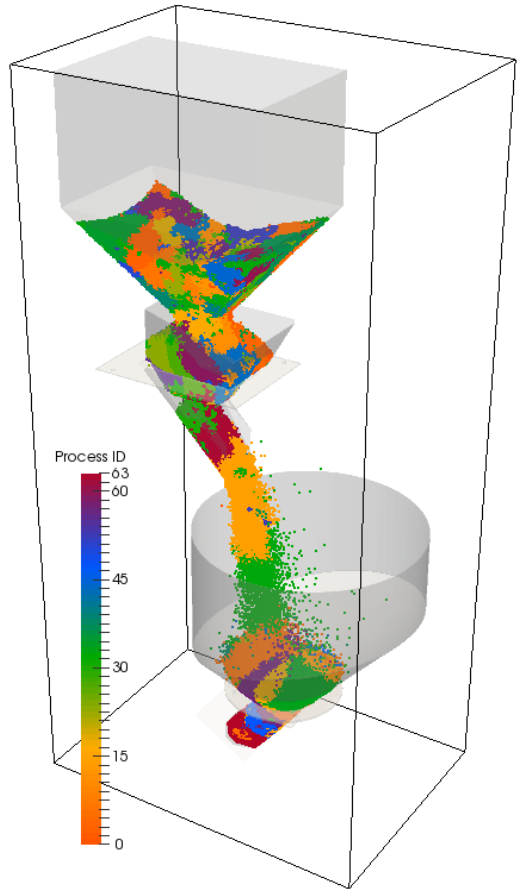
Zoltan RCB
(Recursive Coordinate Bisection)



ORB
(Orthogonal Recursive Bisection)



Zoltan RIB
(Recursive Inertial Bisection)



SCOTCH K-way

UNIVERSITÉ DU LUXEMBOURG

Workload Estimation for XDEM

Toward better Load-Balancing

Main Computations Phases in XDEM

Interaction Detection

Broad Phase: Fast but approximate scan to identify the pairs of particles that *could* interact

- ➔ uses an approximate shape (bounding volume)

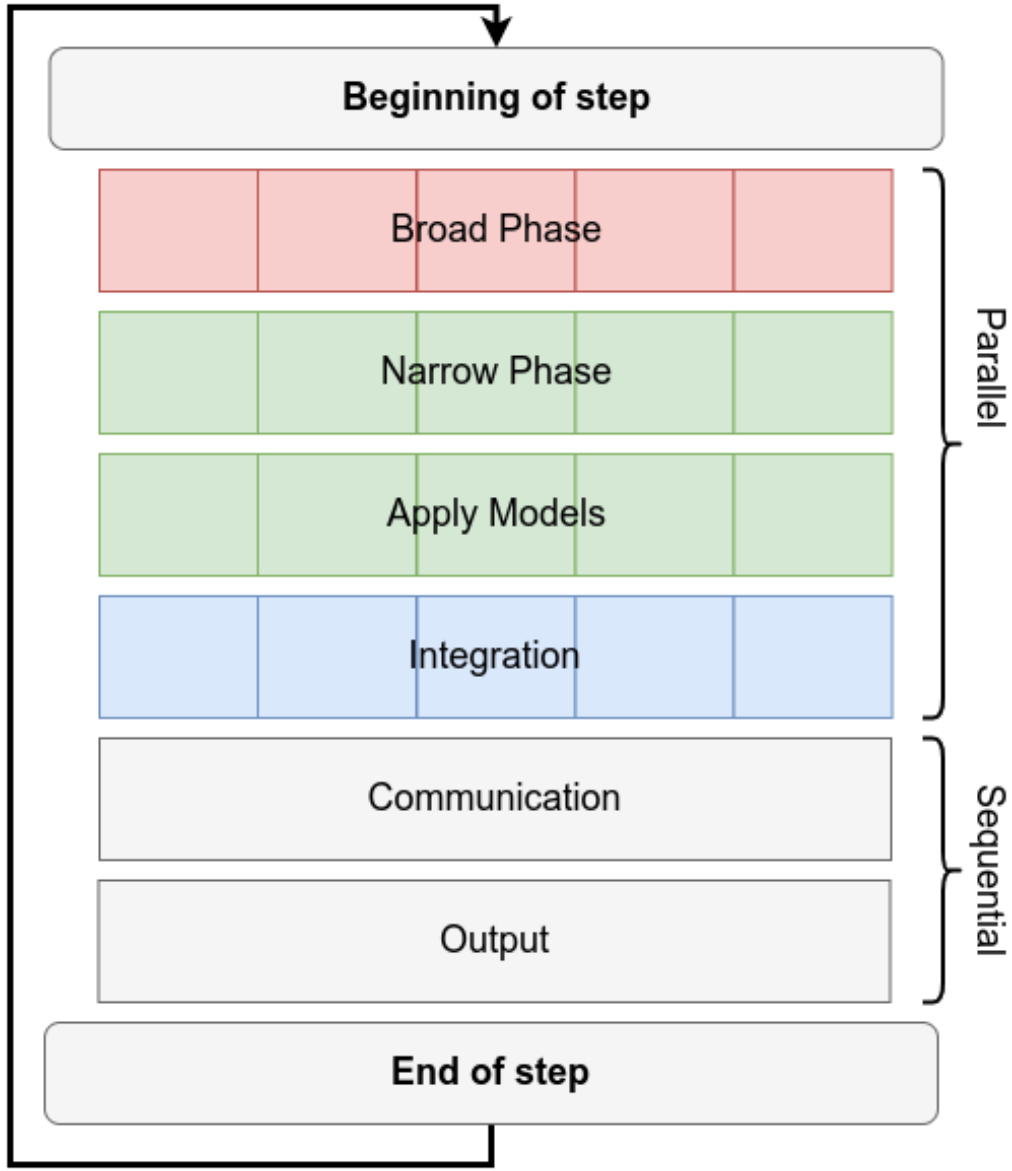
Narrow Phase: Precise collision detection on the particle pairs identified in the broad-phase

- ➔ uses the actual shape (sphere, cube, cylinder, etc.)
- ➔ calculates the distance/overlap between particles

Apply Models: Apply the physics models to each pair of interacting particles

- ➔ accumulate contributions to each particle:
  Contact → *force*, *torque*, ...
  Conduction/Radiation → *heat flux, ...*

Integration: Update the particle states by integrating the contributions from all the interacting partners



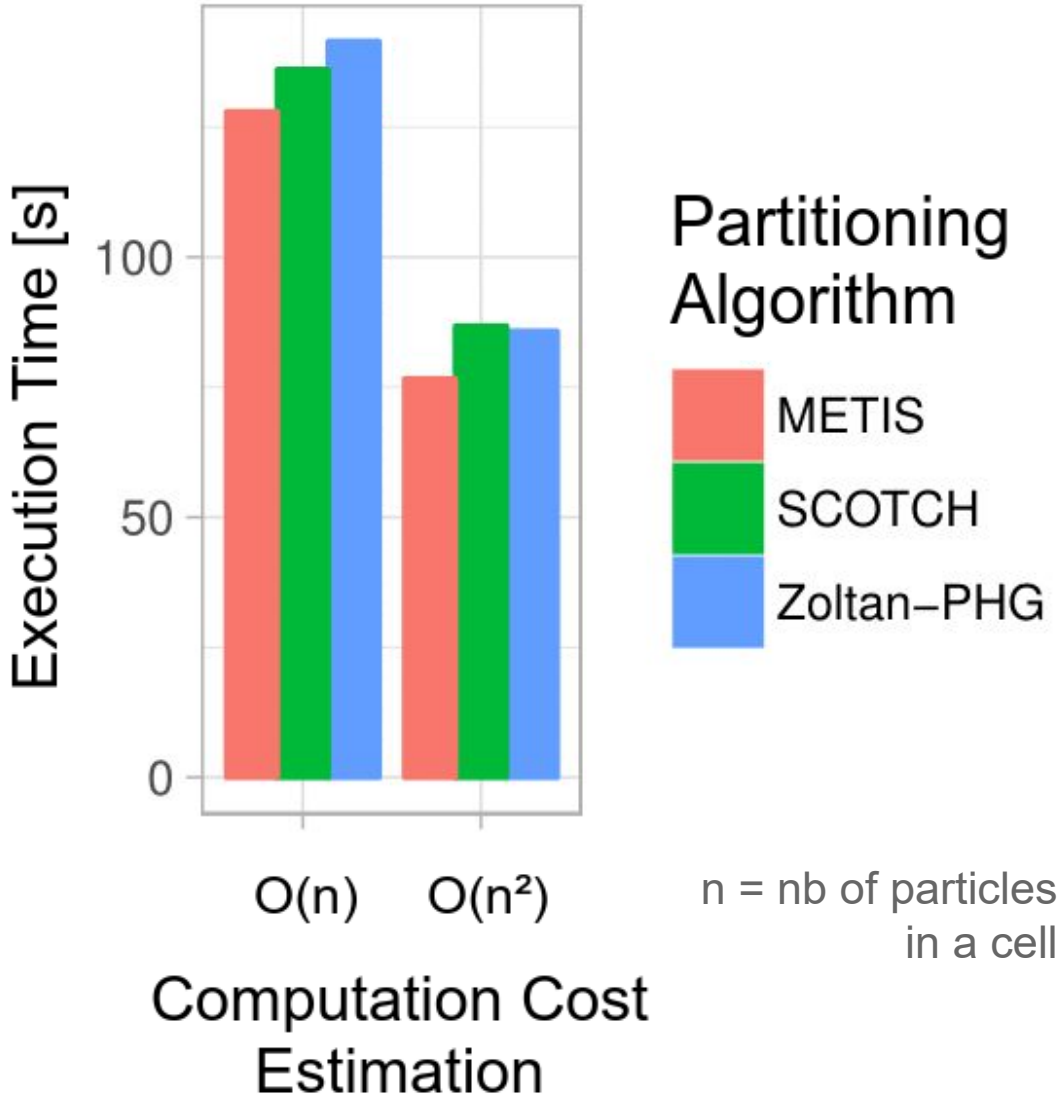
Weight estimation for load-balancing

How to estimate the computing cost ?

- Difficult to measure at the level of a single cell
- Multiple phases and different complexities

Computation Phase	*Complexity*
Broad-phase	O((nb particles in cell)^2)
Narrow-phase	O(nb interactions)
Apply Models	O(nb interactions)
Integration	O(nb particles)

- Nb of interactions is difficult to estimate



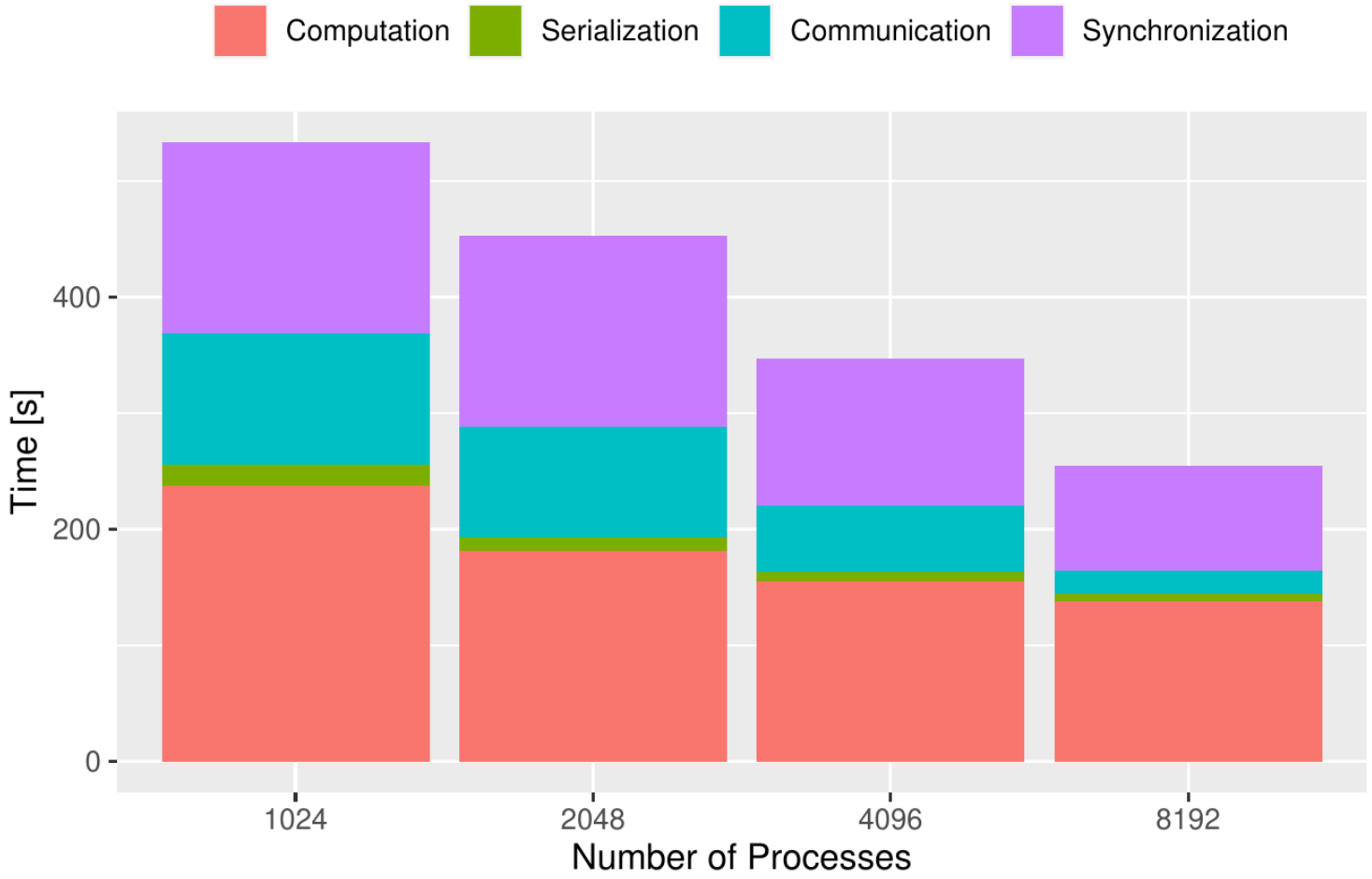
→ **Workload estimation** has a significant impact on the **load-balancing** and on the **performance**

Preliminary Results

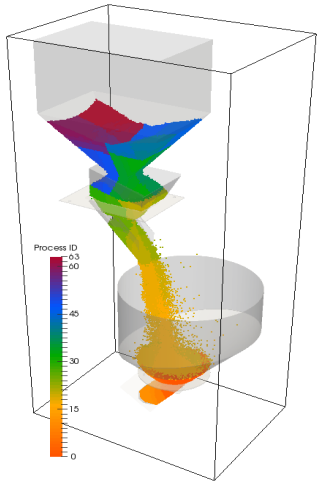
Load Estimation and Imbalance

Profiling large scale execution

- Use ‘extra’ synchronizations to isolate the phases in the execution



→ **Time spent in synchronization indicates imbalance between the processes**

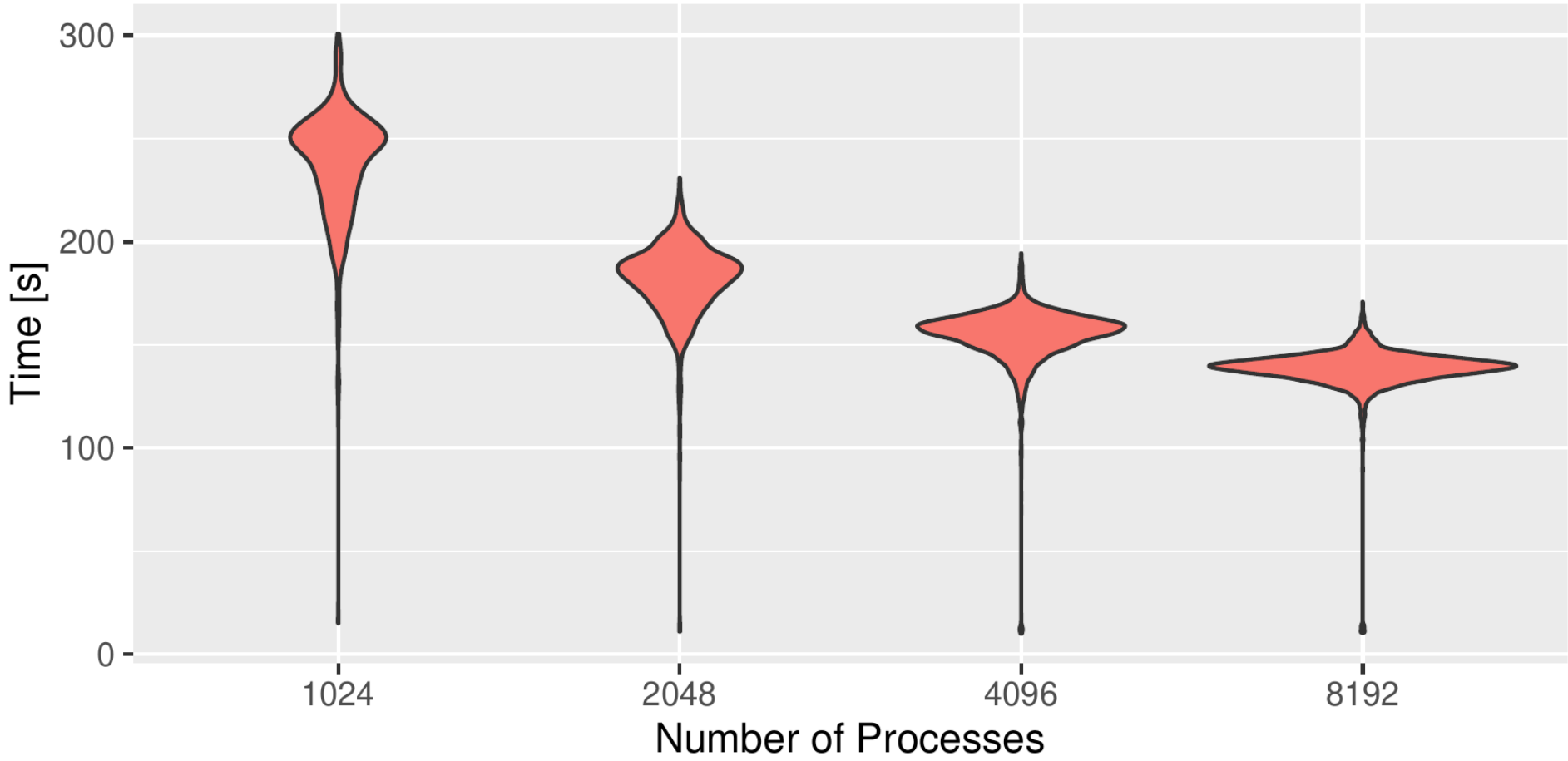


Hopper discharge with 5.5M particles

1000 timesteps
Partitioner: Zoltan-RCB
Cost function: $1+n^2$

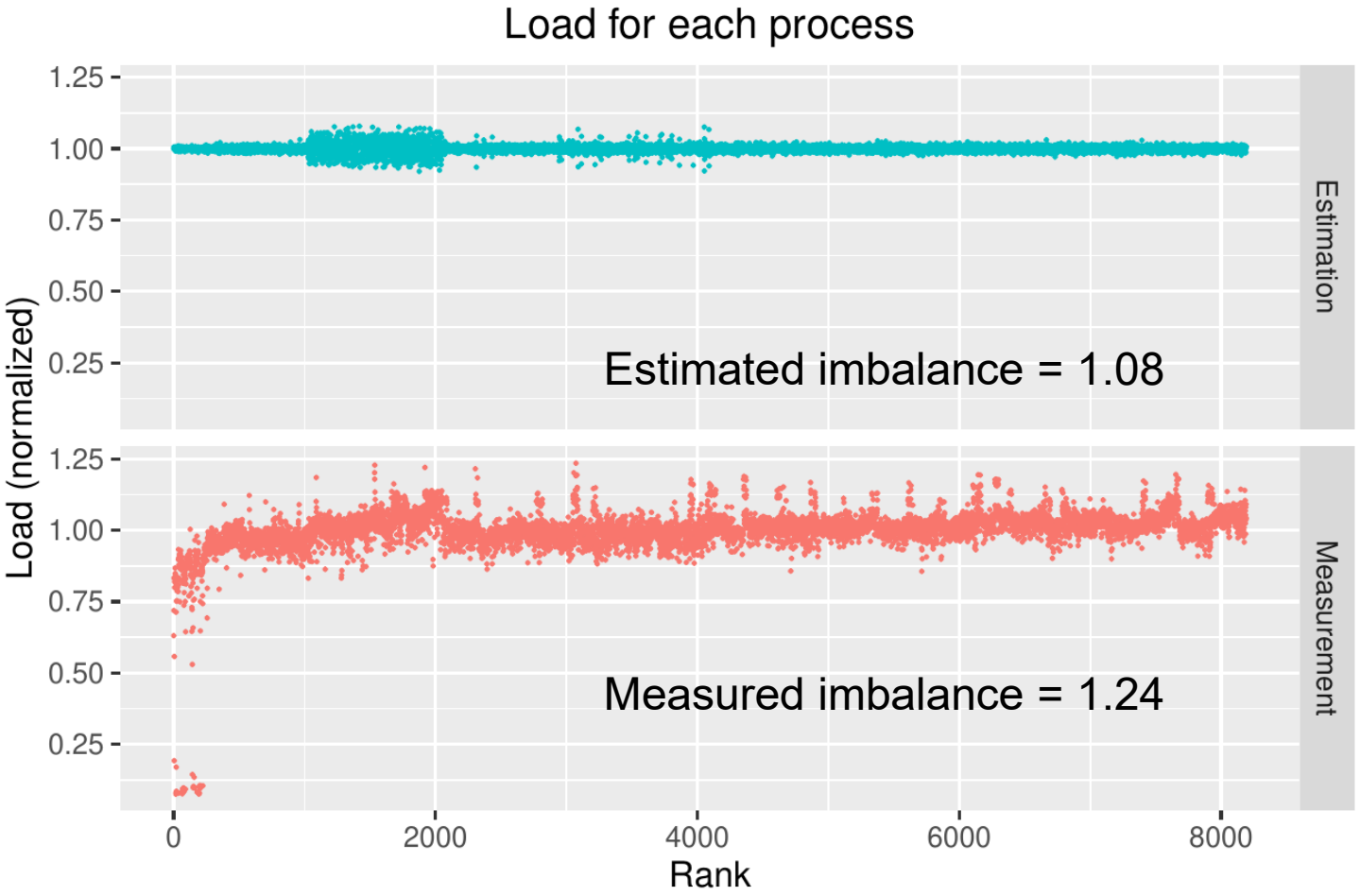
Measured Imbalance

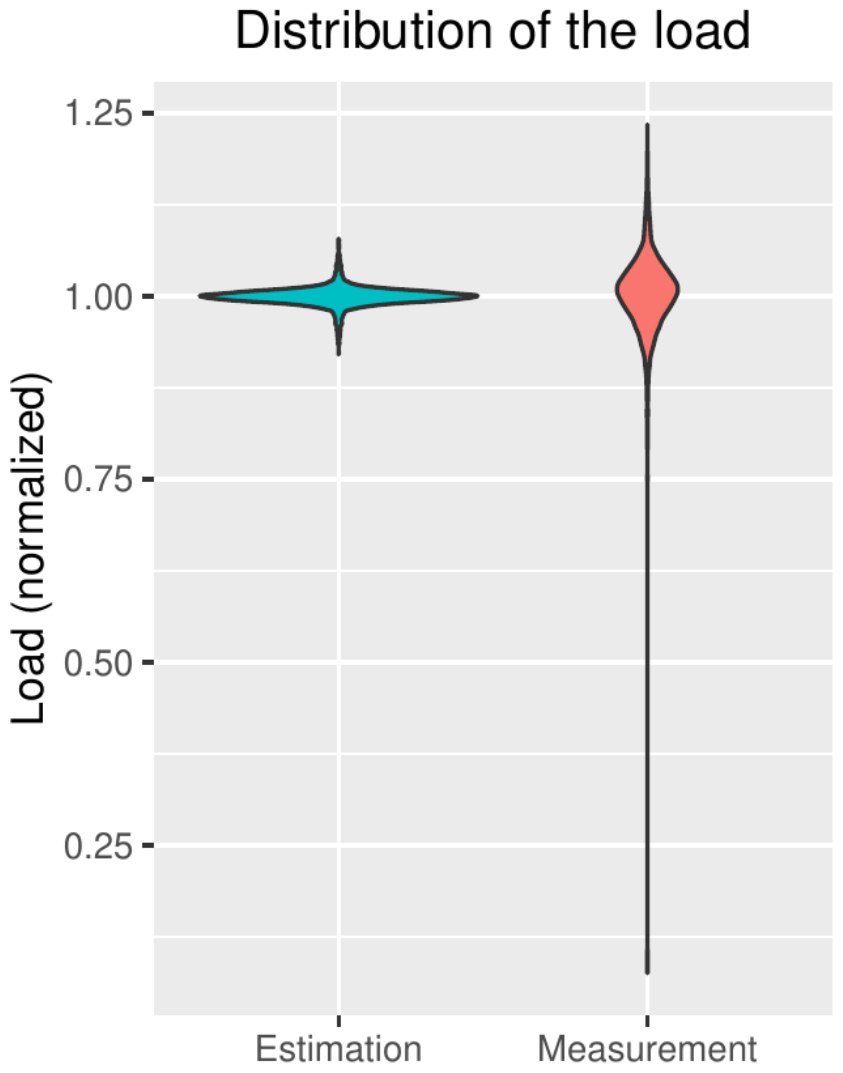
Distribution of the computation time (excluding communication-related time)



Measured Imbalance:	1.26	1.27	1.26	1.24

Estimated Load vs Measured Load





- Discrepancy between the estimated and the measured load
- The load-balancing depends on a good load estimation

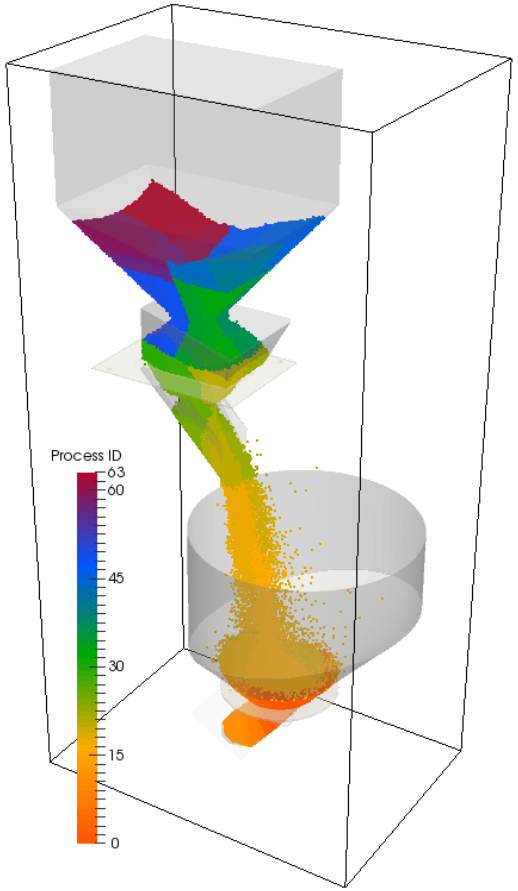
→ **Propose an accurate load estimation function for XDEM** (work-in-progress)

Thank you for your attention !

Question?

Xavier Besseron
University of Luxembourg

LuXDEM Research Team
https://luxdem.uni.lu



We acknowledge EuroHPC JU for awarding this project access to MeluXina.

EUROHPC USER DAY 2023 Brussels 11.12.23

EuroHPC Joint Undertaking

Project: Workload Estimation and Load-Balancing of Discrete Element Method
Period: Sep. 2023 – Aug. 2024
EuroHPC used: MeluXina

HPC uni.lu UNIVERSITÉ DU LUXEMBOURG

Some results presented in this research were carried out using the HPC facilities of the University of Luxembourg