PASC18
Basel Switzerland | 2-4 July 2018

Porting a Legacy Global Lagrangian PIC Code on Many-Core and GPU-Accelerated Architectures

PASC Conference, Basel

July 3rd, 2018

Noé Ohana [1], Andreas Jocksch [2], Emmanuel Lanti [1], Aaron Scheinberg [1], Stephan Brunner [1], Claudio Gheller [2], Laurent Villard [1]

[1] SPC, Lausanne
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE | SWISS PLASMA CENTER

[2] CSCS, Lugano
CSCS Centro Svizzero di Calcolo Scientifico Swiss National Supercomputing Centre

Acknowledgments: Alberto Bottino, Ben McMillan, Alexey Mishchenko, Thomas Hayward

EUROfusion

This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

Outline

Motivation

- Problem:
  - Development timescale of legacy code ORB5 [Tran, 1999, Jolliet, 2009] exceeds HPC architecture evolution timescale

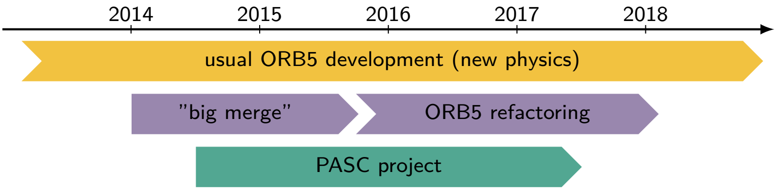
Motivation

- Problem:
  - Development timescale of legacy code ORB5 [Tran, 1999, Jolliet, 2009] exceeds HPC architecture evolution timescale
- Adopted strategy:
  - Disentangle numerical kernels and physical modules ⇒ modularization
  - More and more cores per compute node ⇒ trade multitasking for multithreading
  - Keep portability ⇒ directive-based approach (OpenMP and OpenACC)
  - Develop testbed code with fondamental kernels (PASC project)

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE
SWISS PLASMA CENTER

Recipe

- Timeline:



- Tools:
  - Profiler
    Score-P
    Scalable performance measurement infrastructure for parallel codes
  - Continuous integration tool
    Jenkins

ORB5, a multi-physics code...

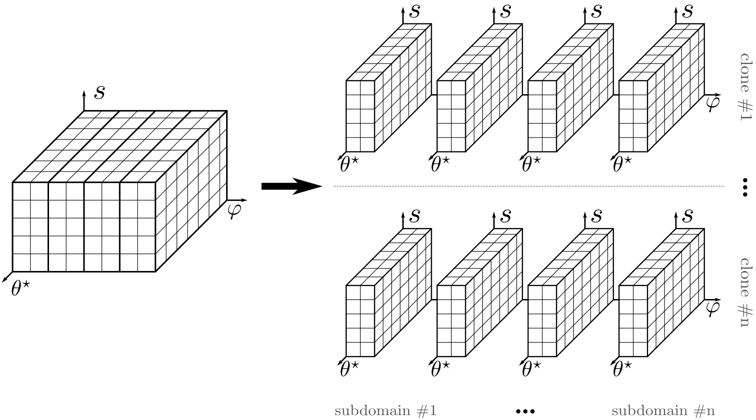
- Global gyrokinetic PIC code:
  - Equations derived from variational formulation with consistent ordering [Tronko, 2017]
  - Ad-hoc or MHD equilibrium
  - Electromagnetic perturbations
  - Different field solver options (long wavelength approximation, Padé approximation, or all orders)
  - Different gyro-averaging options ($\langle \nabla \phi \rangle$ or $\nabla \langle \phi \rangle$)
  - Multiple gyrokinetic species
  - Fixed or adaptive number of Larmor points
  - Drift-kinetic, adiabatic or hybrid electrons
  - Inter- and intra- species collisions
  - Heat sources
  - Strong flows
  - Advanced diagnostics

... with various numerical features

- Numerical features:
  - Variational formulation of field equations with finite element B-splines up to third order
  - Control variate schemes
  - Electromagnetic cancellation problem in Ampère's law solved with enhanced control variate or pullback scheme [Mishchenko, 2014]
  - Runge-Kutta of fourth order time integrator
  - Magnetic coordinates, straight-field-line
  - Field-aligned Fourier filter
  - Noise control (Krook operator, coarse graining, quadtree)
  - Original parallelization: 2 levels of MPI (domain decomposition and cloning)
  - New parallelization: hybrid MPI+OpenMP or MPI+OpenACC

MPI parallelization scheme

- 1D domain decomposition plus domain cloning:



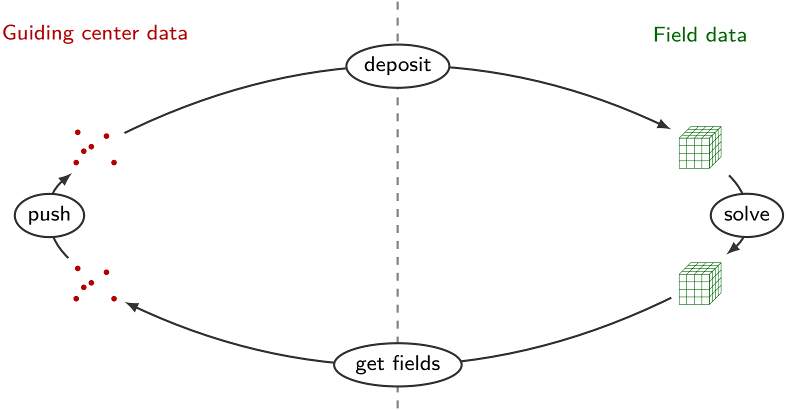
- Reduce/broadcast operations between clones
- All-to-all (parallel data transpose and particle move) and neighboor (guard cells) communications between subdomains

Key modifications prior to multithreading

- Data structures
  - Structure of arrays instead of arrays of structure
  - Pack variables for host-to-device memory transfers

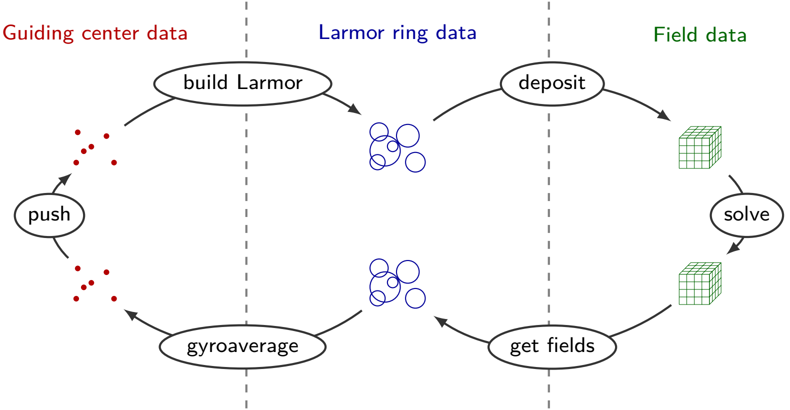
Key modifications prior to multithreading

- Data structures
  - Structure of arrays instead of arrays of structure
  - Pack variables for host-to-device memory transfers
- Modularization
  - Stages of a time step:



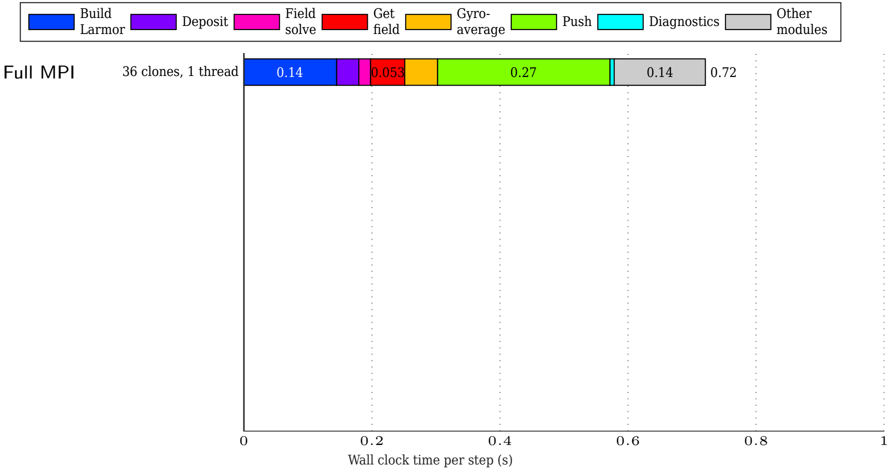
Key modifications prior to multithreading

- Data structures
  - Structure of arrays instead of arrays of structure
  - Pack variables for host-to-device memory transfers
- Modularization
  - Stages of a time step:



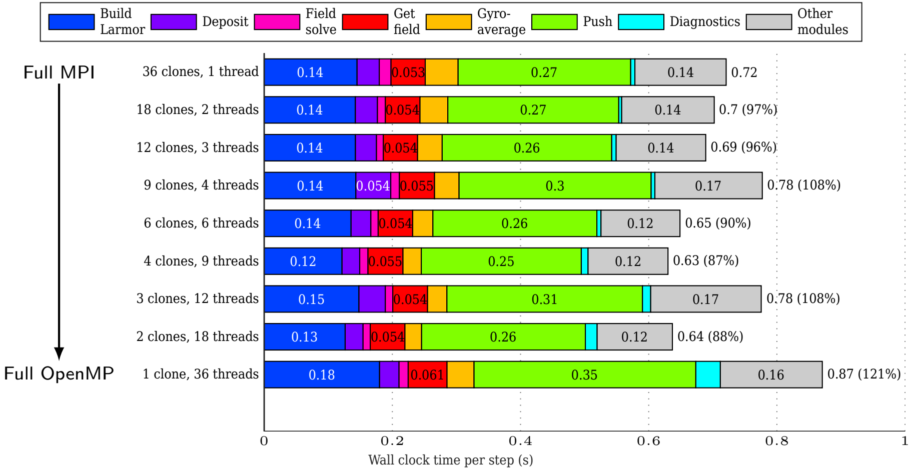
MPI+OpenMP

- Single node simulations with 10^6 ions (4 Larmor points each), 10^6 electrons, and $128 \times 32 \times 4$ cubic splines, on Broadwell CPU (2×18 cores, 2.1GHz)



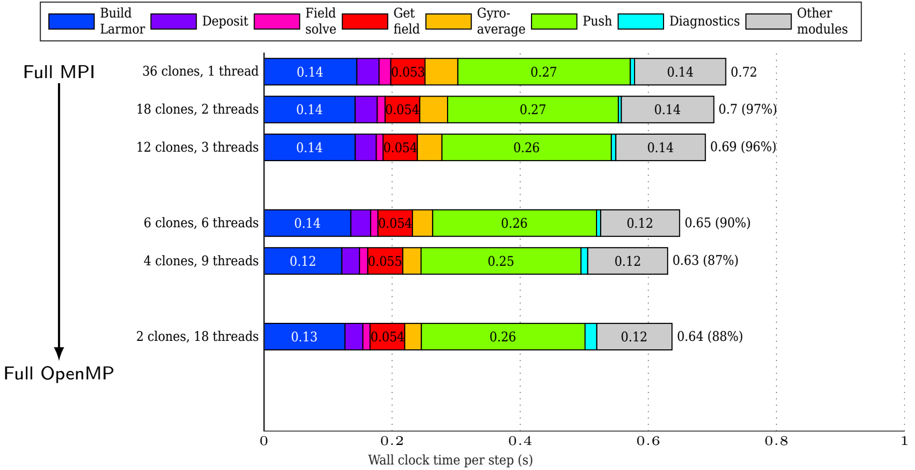
MPI+OpenMP

- Single node simulations with 10^6 ions (4 Larmor points each), 10^6 electrons, and $128 \times 32 \times 4$ cubic splines, on Broadwell CPU (2×18 cores, 2.1GHz)



MPI+OpenMP

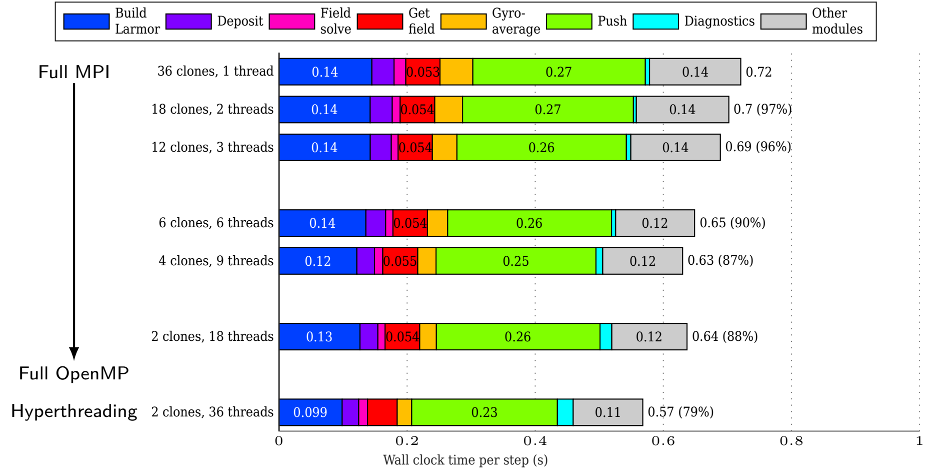
- Single node simulations with 10^6 ions (4 Larmor points each), 10^6 electrons, and $128 \times 32 \times 4$ cubic splines, on Broadwell CPU (2×18 cores, 2.1GHz)



MPI+OpenMP

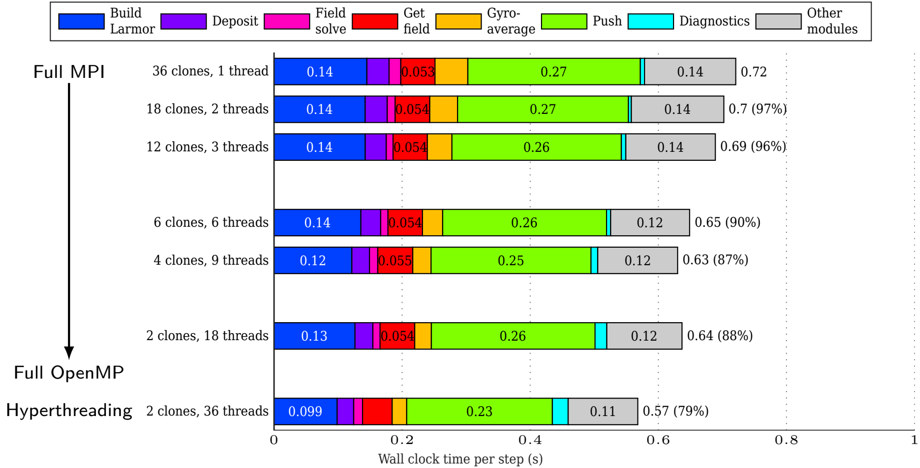
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE — SWISS PLASMA CENTER

- Single node simulations with 10^6 ions (4 Larmor points each), 10^6 electrons, and $128 \times 32 \times 4$ cubic splines, on Broadwell CPU (2×18 cores, 2.1GHz)



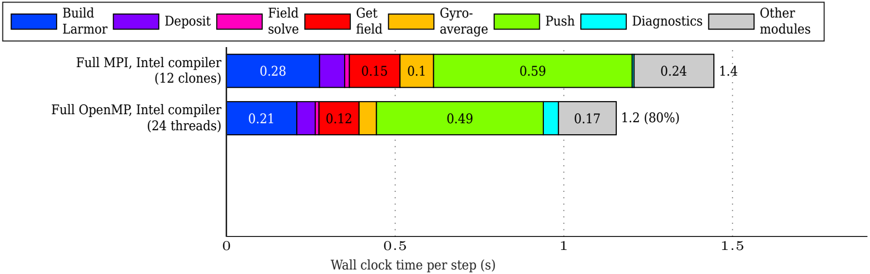
MPI+OpenMP

- Single node simulations with 10^6 ions (4 Larmor points each), 10^6 electrons, and $128 \times 32 \times 4$ cubic splines, on Broadwell CPU (2×18 cores, 2.1GHz)

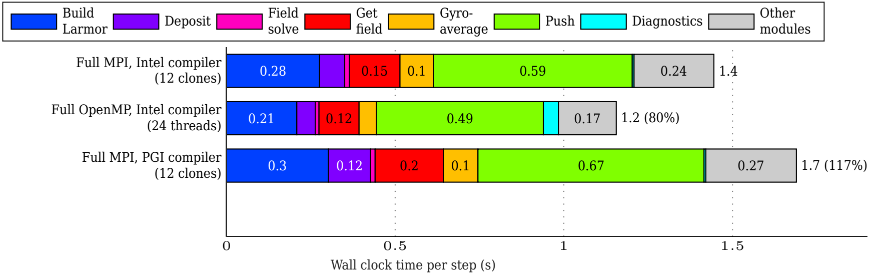


- Up to 20% speed-up by filling sockets with OpenMP threads

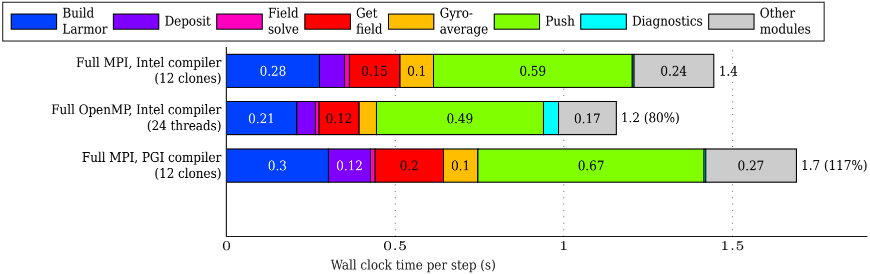
- Single node simulations with 10^6 ions (4 Larmor points each), 10^6 electrons, and $128 \times 32 \times 4$ cubic splines, on Haswell CPU (12 cores, 2.6GHz)



- Single node simulations with 10^6 ions (4 Larmor points each), 10^6 electrons, and $128 \times 32 \times 4$ cubic splines, on Haswell CPU (12 cores, 2.6GHz)

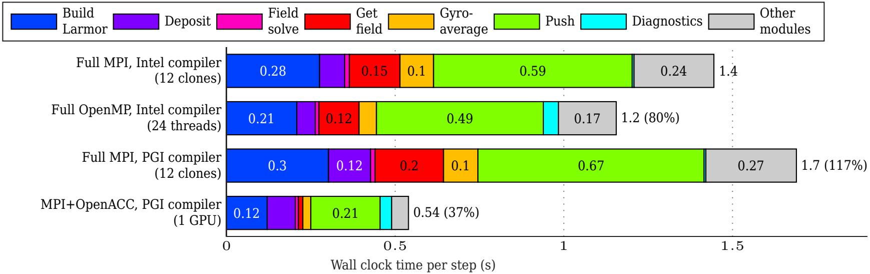


- Single node simulations with 10^6 ions (4 Larmor points each), 10^6 electrons, and $128 \times 32 \times 4$ cubic splines, on Haswell CPU (12 cores, 2.6GHz)



- PGI compiler ∼20% slower than Intel

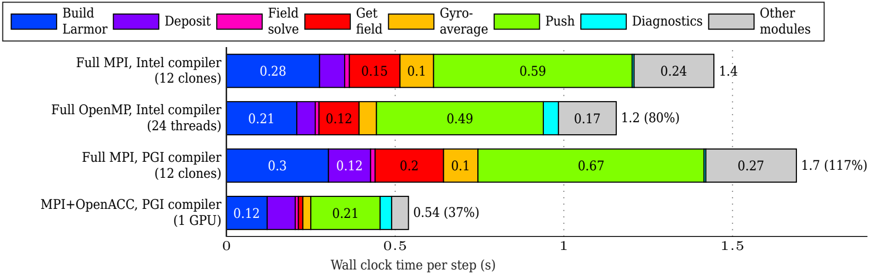
- Single node simulations with 10^6 ions (4 Larmor points each), 10^6 electrons, and $128 \times 32 \times 4$ cubic splines, on Haswell CPU (12 cores, 2.6GHz)



- PGI compiler ∼20% slower than Intel

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE
SWISS PLASMA CENTER

- Single node simulations with 10^6 ions (4 Larmor points each), 10^6 electrons, and $128 \times 32 \times 4$ cubic splines, on Haswell CPU (12 cores, 2.6GHz)

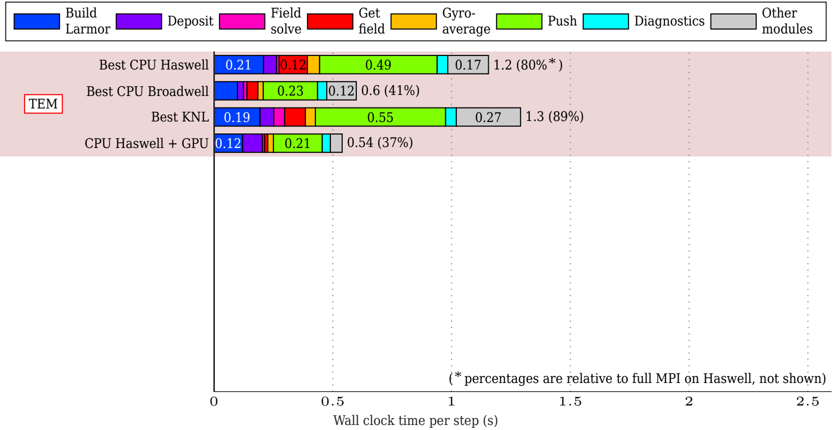


- PGI compiler ∼20% slower than Intel
- MPI+OpenACC version ∼2.7 times faster than MPI only, and ∼2.2 times faster than MPI+OpenMP

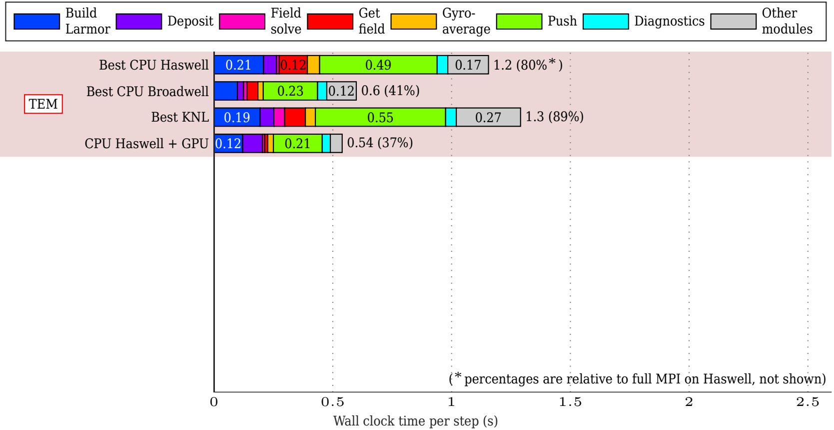
Architecture comparisons

SWISS PLASMA CENTER

- Different physics (electro-static or -magnetic, adiabatic or kinetic electrons, adaptive number of Larmor points, ...) with similar particle and cell resolutions



- Different physics (electro-static or -magnetic, adiabatic or kinetic electrons, adaptive number of Larmor points, ...) with similar particle and cell resolutions

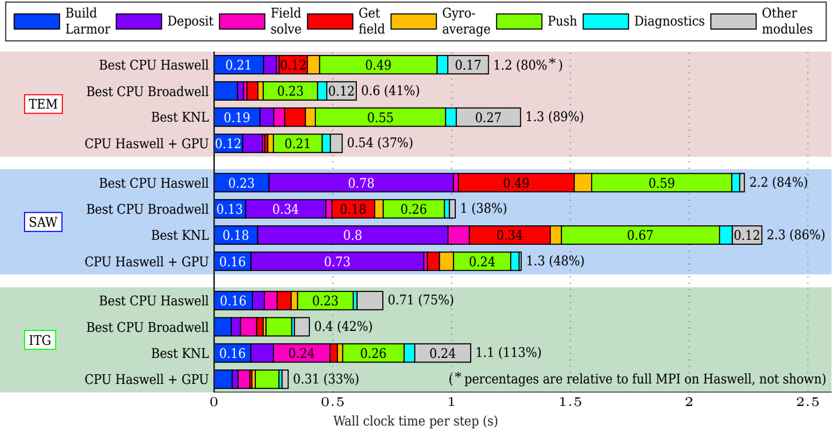


- KNL (64 cores, 1.3GHz) performance similar to Haswell CPU

Architecture comparisons

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE SWISS PLASMA CENTER

- Different physics (electro-static or -magnetic, adiabatic or kinetic electrons, adaptive number of Larmor points, ...) with similar particle and cell resolutions

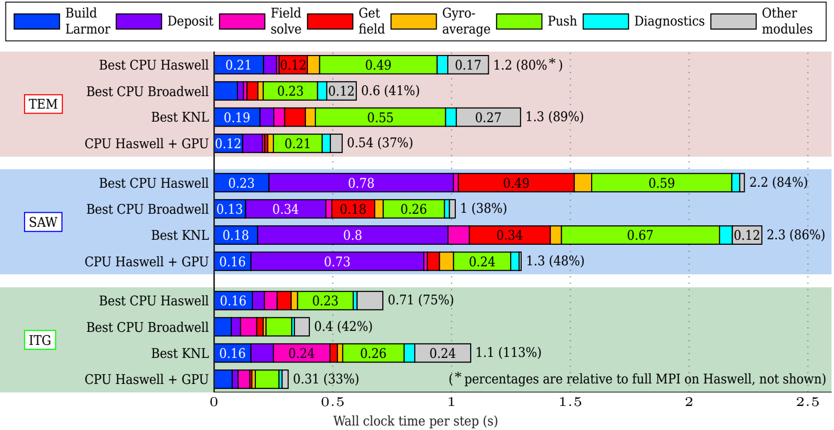


- KNL (64 cores, 1.3GHz) performance similar to Haswell CPU

Architecture comparisons

SWISS PLASMA CENTER

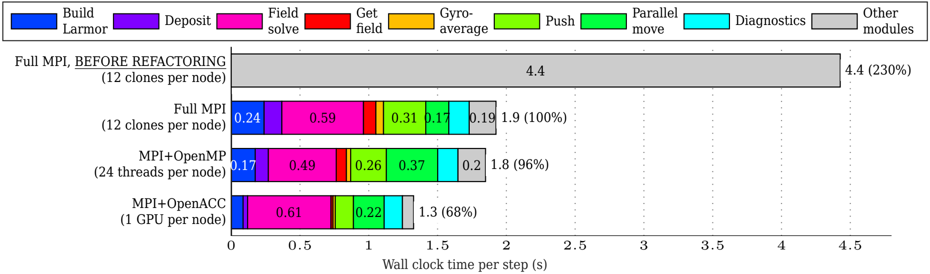
- Different physics (electro-static or -magnetic, adiabatic or kinetic electrons, adaptive number of Larmor points, ...) with similar particle and cell resolutions



- KNL (64 cores, 1.3GHz) performance similar to Haswell CPU
- Weaker GPU performance for SAW due to control variate iterations on CPU
- GPU performance similar to Broadwell CPU
- Exact factors are case-dependent

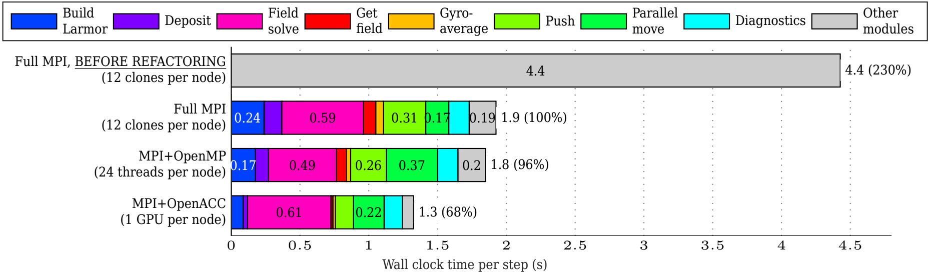
Application to production run

- ITG production run on 256 nodes with $256 \cdot 10^6$ ions (4 Larmor points each) and $256 \times 512 \times 256$ cubic splines



Application to production run

- ITG production run on 256 nodes with $256 \cdot 10^6$ ions (4 Larmor points each) and $256 \times 512 \times 256$ cubic splines



- Timings of communication-bound stages (field solver and parallel move) become non-negligible.
- Multithreaded versions bring less speed-up than on sinle node because they do not improve inter-node communication.
- Full MPI version already 2.3 times faster than before refactoring.

Conclusion

- Achievements:
  - Single code making efficient use of different architectures
  - Good compromise between efficiency, ergonomy, and "future-proofness" (flexibility to accomodate for new physics and new architectures)
- Future work:
  - Turn on sorting when particle-to-field operations are significant
  - Reduce amount of communications in solver

Bibliography

SWISS PLASMA CENTER

- T.M. Tran, K. Appert,M. Fivaz, G. Jost, J. Vaclavik and L. Villard
  Global gyrokinetic simulation of ion-temperature-gradient-driven instabilities using particles
  Theory of Fusion Plasmas, Int. Workshop (Editrice Compositori, Bologna, Societa Italiana di Fisica), 45, 1999
- S. Jolliet
  Gyrokinetic particle-in-cell global simulations of ion-temperature-gradient and collisionless-trapped-electron-mode turbulence in tokamaks
  PhD thesis, EPFL, 2009
- A. Mishchenko, A. Könies, R. Kleiber, and M. Cole
  Pullback transformation in gyrokinetic electromagnetic simulations
  Physics of Plasmas, 21, 2014
- N. Tronko, A. Bottino, C. Chandre and E. Sonnendruecker
  Hierarchy of second order gyrokinetic Hamiltonian models for particle-in-cell codes
  Plasma Physics and Controlled Fusion, 59, 2017