OPAL a State of the Art Partcicle Accelerator Framework

A. Adelmann

December 19, 2022

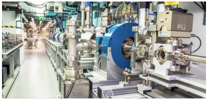
- not relevant for the exam
- purpose is to inspire you

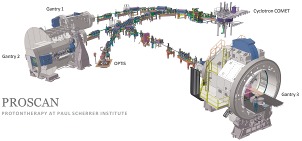
Problem Setup

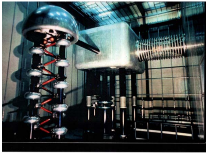
Understand Dynamical Systems !

Charged Particle Accelerators in particular

- this is a multiscale plasma-physics problem
- spatial scales: nm ... km
- velocity scales: $v \ll c \ldots v \sim c - \varepsilon$
- PIC: N is ranging up to 10^9 and bejond







Understand Dynamical Systems !

Challenges involved ...

Computing Physics Mathematics

The OPAL Ecosystem

Object Oriented Parallel Particle Library (OPAL)

https://gitlab.psi.ch/OPAL/src/wikis/home

OPAL is a versatile open-source tool for charged-particle optics in large accelerator structures and beam lines including 3D EM field calculation, collisions, radiation, particle-matter interaction, and multi-objective optimisation

- OPAL is built from the ground up as an HPC application
- OPAL runs on your laptop as well as on the largest HPC clusters
- OPAL uses the MAD language with extensions
- OPAL is written in C++, uses design patterns, easy to extend
- The OPAL Discussion Forum: https://psilists.ethz.ch/sympa/info/opal
- International team of 11 active developers and a user base of $\mathcal{O}(100)$
- The OPAL **sampler** command can generate labeled data sets using the largest computing resources and allocations available

The Active OPAL Developer Team


UKRI Science and Technology Facilities Council | ISIS Neutron and Muon Source

SLAC NATIONAL ACCELERATOR LABORATORY

PAUL SCHERRER INSTITUT PSI

MIT

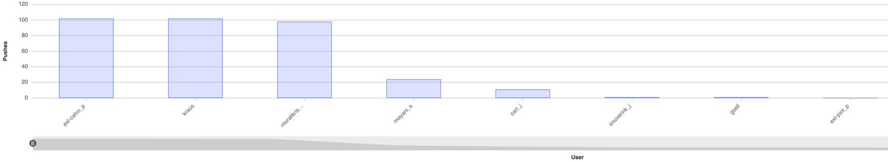
Ciemat
Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas

University of St Andrews

Challenges involved ...

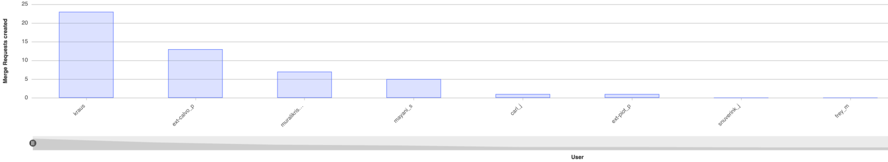
Pushes

339 pushes, more than **814.0** commits by **7 people** contributors.



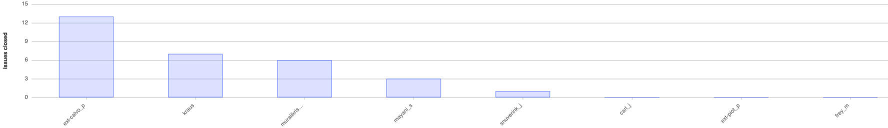
Merge requests

50 created, **45** merged.

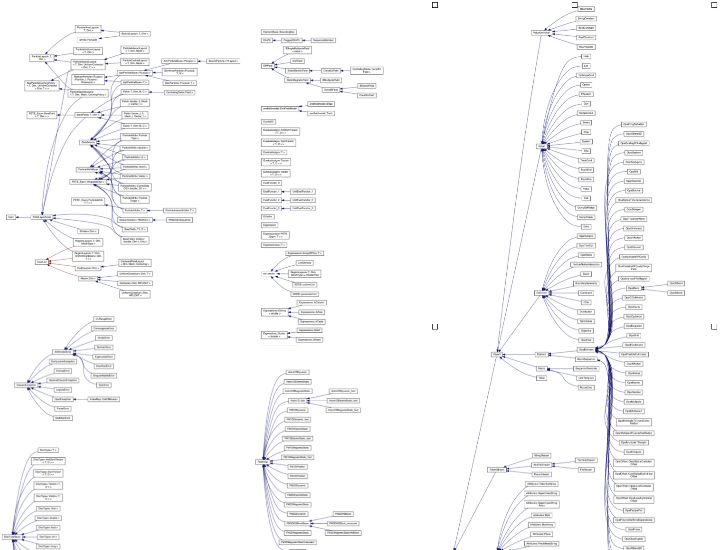


Issues

35 created, **30** closed.



Challenges involved ...



The Governing Equations

$f_s \subset (\mathbb{R}^3 \times \mathbb{R}^3), \mathbb{R} :\to \mathbb{R}$ and s are the species.

$$\frac{\partial f_s}{\partial t} + \mathbf{v} \cdot \nabla_x f_s + \frac{q_s}{m_s}(\mathbf{E} + \mathbf{v} \times \mathbf{B}) \cdot \nabla_v f_s = \frac{\partial f_s}{\partial t}_{\text{coll}},$$

$$\left.\begin{aligned} \partial_t \mathbf{E} - c^2 \mathbf{curl}\, \mathbf{B} = \frac{\mathbf{J}}{\varepsilon_0}, \qquad \nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0}, \\ \partial_t \mathbf{B} + \mathbf{curl}\, \mathbf{E} = 0, \qquad \nabla \cdot \mathbf{B} = 0, \end{aligned}\right\} \qquad \text{Maxwell's equations}$$

where the source terms are computed by

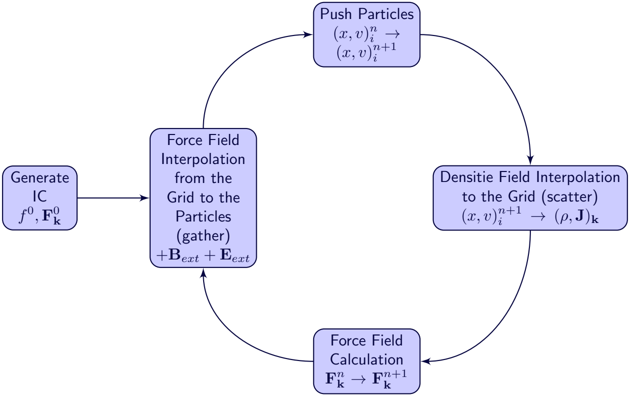
$$\rho = \sum_s q_s \int f_s \, d\mathbf{v}, \qquad \mathbf{J} = \sum_s q_s \int f_s \mathbf{v} \, d\mathbf{v}.$$

In some cases Maxwell's equations can be replaced by a reduced model like Poisson's equation. The electric and magnetic fields $\mathbf{E}$ and $\mathbf{B}$ are superpositions of external fields and self-fields (space charge),

$$\mathbf{E} = \mathbf{E}_{\text{ext}} + \mathbf{E}_{\text{sc}}, \quad \mathbf{B} = \mathbf{B}_{\text{ext}} + \mathbf{B}_{\text{sc}}.$$

We do Particle-In-Cell (ES-PIC)

With $f^0 \in \mathbb{R}^{3\times3\times1}$ we denote the initial phase space. The discrete force field is $\mathbf{F_k}$ with $\mathbf{k} = \{i, j, k\}$. A particle i is denoted by $(x, v)_i \in \mathbb{R}^3$ and pushed from time steps n to $n + 1$:



PAUL SCHERRER INSTITUT

Vlasov-Poisson Equation

When neglecting collisions, and taking advantage of the electrostatic approximation, the Vlasov-Poisson equation describes the (time) evolution of the phase space $f(\mathbf{x}, \mathbf{v}; t) > 0$ when considering electromagnetic interaction with charged particles.

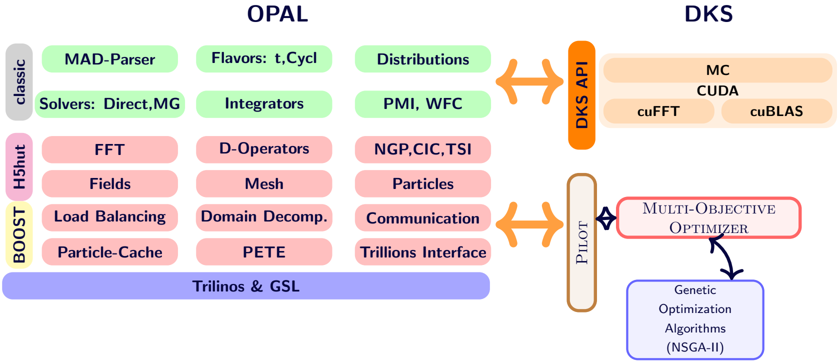
$$\frac{df}{dt} = \frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla_x f + \frac{q}{m}(\mathbf{E}(\mathbf{x}, t) + \mathbf{v} \times \mathbf{B}(\mathbf{x}, t)) \cdot \nabla_v f = 0. \qquad (1)$$

Solving with ES-PIC

- Hockney and Eastwood, $h_x(t), h_y(t), h_z(t)$, $M = M_x \times M_y \times M_z$
- SAAMG-PCG solver with geometry [AA et al., JCP, **229** 12 (2010)]
- change M during simulation (many different field solver instances)
- adaptive in Δt
  [M. Toggweiler, AA, et al. J. Comp. Phys. **273** (2014)]
- modern computational architectures

Software Architecture

MPI based + HW accelerators + Optimiser [N. Neveu, AA, et al. (2019)]



Time Dependent Phase Space

GPU & FFT Poisson solver

[AA et al., CPC **207** (2016)], [U. Locans, AA, et al., CPC **215** (2017)]

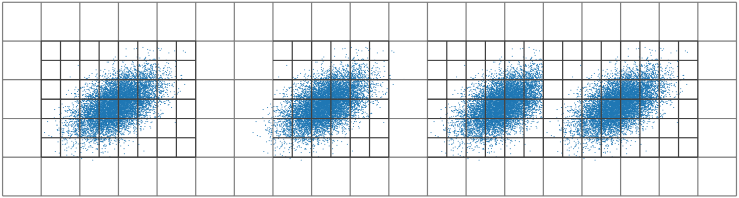
Example: simulation for the PSI Ring Cyclotron.
Host code 8 cores: 2x Intel Xeon Processor E5-2609 v2
Accelerator: Nvidia Tesla K20 or Nvidia Tesla K40

FFT size	DKS	Total time (s)	OPAL speedup	Solver t (s)	Solver speedup
64x64x32	no	324.98		22.53	
	K20	**311.17**	$\times$**1.04**	**7.42**	$\times$**3**
	K40	**293.7**	$\times$**1.10**	**7.32**	$\times$**3**
128x128x64	no	434.22		206.73	
	K20	**262.74**	$\times$**1.6**	**32.15**	$\times$**6.5**
	K40	**245.08**	$\times$**1.8**	**25.87**	$\times$**8**
256x256x128	no	2308.05		1879.84	
	K20	**625.37**	$\times$**3.6**	**202.63**	$\times$**9.3**
	K40	**542.73**	$\times$**4.2**	**160.87**	$\times$**11.7**
512x512x256	no	3760.46		3327.14	
	K40	**716.86**	$\times$**5.2**	**302.49**	$\times$**11**

Adaptive Mesh Refinement (AMR) in OPAL

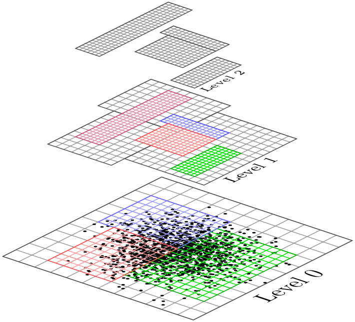
SNF sponsored Ph.D. project M. Frey [M. Frey et al. (2020)]

- **Requirements on Particle-in-Cell (PIC) Model:**
  - Solving large-scale N-body problems of $\mathcal{O}(\mathbf{10^9 ... 10^{10}})$ **particles** coupled with Maxwell's equations
  - High resolution to cover tiny halo effects $\Longrightarrow$ Extremely fine mesh of $\mathcal{O}(\mathbf{10^8 ... 10^9})$ **grid points**
- **Bottlenecks:**
  - Waste of memory and resolution in regions of void
- **Solution:**
  - Block-structured adaptive mesh-refinement (AMR)



Adaptive Mesh Refinement (AMR) in OPAL

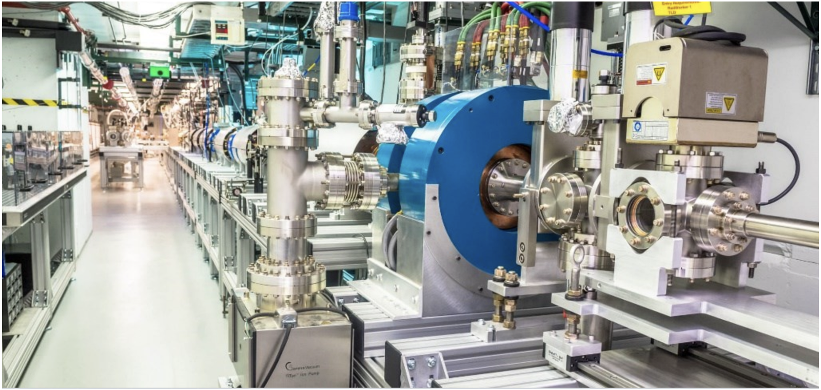
- General interface to AMR libraries (in use: AMReX[1])
- Hardware independent implementation (CPU/GPU/XXX)



[1] https://amrex-codes.github.io/amrex/

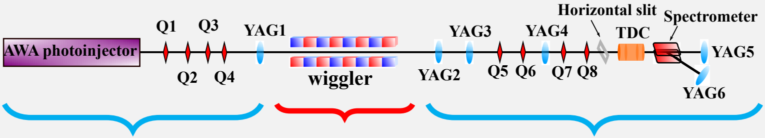
The OPAL full Electromagnetic - a case study

The Need for a Full EM Solver



The Need for a Full EM Solver

AWA Wiggler Experiment



Only space charge

(electrostatic approximation okay)

Space charge and radiation

(requires full EM)

Only space charge

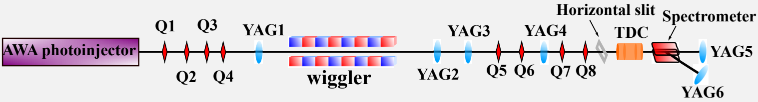
(electrostatic approximation okay)

Need a model/tool that can model this beamline - economically - start-to-end.

OPAL electrostatic solver (only space charge):

The Need for a Full EM Solver

AWA Wiggler Experiment



Only space charge

(electrostatic approximation okay)

Space charge and radiation

(requires full EM)

Only space charge

(electrostatic approximation okay)

OPAL electrostatic solver (only space charge):

$$\left\{\begin{array}{l} \boldsymbol{\nabla}\cdot\boldsymbol{E} = \dfrac{\rho}{\varepsilon_0}, \quad \boldsymbol{\nabla}\wedge\boldsymbol{E} = -\cancelto{0}{\dfrac{\partial \boldsymbol{B}}{\partial t}}, \\ \boldsymbol{\nabla}\cdot\boldsymbol{B} = 0, \quad \boldsymbol{\nabla}\wedge\boldsymbol{B} = \mu_0 \boldsymbol{j} + \cancelto{0}{\dfrac{1}{c^2}\dfrac{\partial \boldsymbol{E}}{\partial t}}, \\ \mathbf{E} = \mathbf{E}_{\text{ext}} + \mathbf{E}_{\text{self}}, \quad \mathbf{B} = \mathbf{B}_{\text{ext}} + \mathbf{B}_{\text{self}}, \end{array}\right. \Rightarrow \left\{ \underbrace{\boldsymbol{\nabla}^2\Phi = \rho/\varepsilon_0,} \right.$$

in co-moving frame with appropriate boundary conditions.

Adelmann et al. (2019), DOI:10.48550/ARXIV.1905.06654

Mithra: Full EM Solver from First-Principles

ELSEVIER

Computer Physics Communications

Volume 228, July 2018, Pages 192-208

MITHRA 1.0: A full-wave simulation tool for free electron lasers ☆

Arya Fallahi [a], Alireza Yahaghi [a], Franz X. Kärtner [a, b]

Maxwell equations rearranged into wave equations:

$$\begin{cases} \nabla \cdot \boldsymbol{E} &= \dfrac{\rho}{\varepsilon_0}, \\ \nabla \wedge \boldsymbol{E} &= -\dfrac{\partial \boldsymbol{B}}{\partial t}, \\ \nabla \cdot \boldsymbol{B} &= 0, \\ \nabla \wedge \boldsymbol{B} &= \mu_0 \boldsymbol{j} + \dfrac{1}{c^2}\dfrac{\partial \boldsymbol{E}}{\partial t}, \end{cases} \Rightarrow \begin{cases} \nabla^2 \boldsymbol{A} - \dfrac{1}{c^2}\dfrac{\partial^2 \boldsymbol{A}}{\partial t^2} &= -\mu_0 \boldsymbol{j}, \\ \nabla^2 \phi - \dfrac{1}{c^2}\dfrac{\partial^2 \phi}{\partial t^2} &= -\dfrac{\rho}{\varepsilon_0}. \end{cases} \tag{2}$$

Integrate wave equations with non-standard FDTD, in co-moving frame.

A. Fallahi et al. (2018), DOI:10.1016/j.cpc.2018.03.011

J.-L. Vay (2007), DOI: 10.1103/PhysRevLett.98.130405

PAUL SCHERRER INSTITUT PSI

The AWA POP Experiment

ELSEVIER

Computer Physics Communications

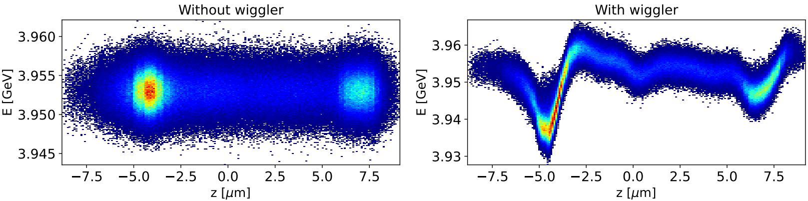
Volume 280, November 2022, 108475

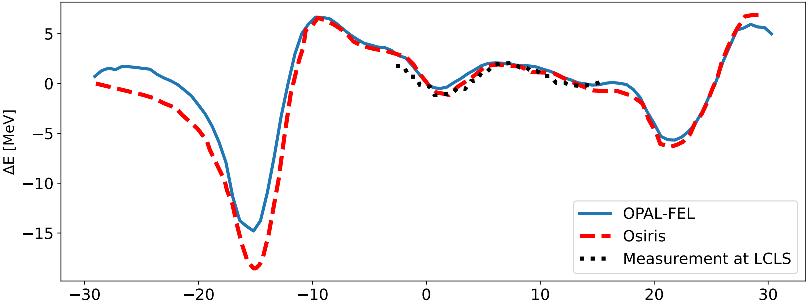
Benchmarking collective effects of electron interactions in a wiggler with OPAL-FEL ☆

Arnau Albà [a], Jimin Seok [b, c], Andreas Adelmann [a], Scott Doran [b], Gwanghui Ha [b], Soonhong Lee [b], Yinghu Piao [b], John Power [b], Maofei Qian [b], Eric Wisniewski [b], Joseph Xu [b], Alexander Zholents [b]

Benchmarking OPAL-FEL: LCLS Experiment

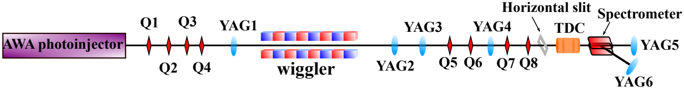
Experiment at LCLS tested wiggler effects in **radiation dominated regime**.

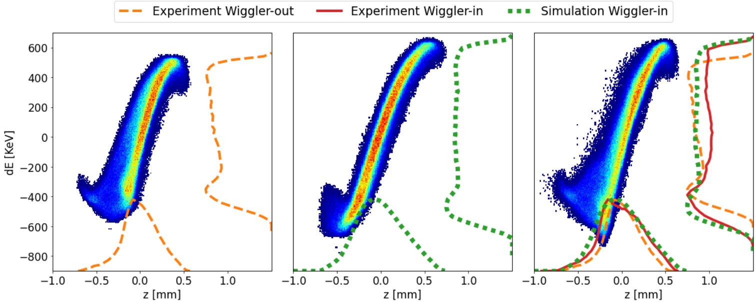




The AWA POP Experiment

Testing wiggler effects in **space charge dominated regime**.





Performance Portability

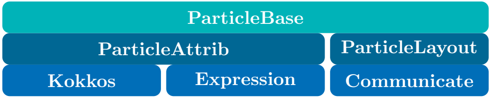
Exascale & Performance Portability Programming

- **Exascale** refers to the capability to perform a billion billion (a quintillion) operations per second.
    - FRONTIER is No 1 on https://www.top500.org/, its AMD Optimized 3rd Generation EPYC 64C 2GHz with 8,730,112 cores.
    - world record 1,102.00 PFlop/s result based on Linpack Performance (Rmax).
- **Performance Portability Programming** Kokkos is a templated C++ library that provides abstractions to allow a single implementation of an application kernel to run efficiently on different kinds of hardware, such as GPUs, Intel Xeon Phis, or many-core CPUs.
    - Kokkos maps the C++ kernel onto different back end languages such as CUDA, OpenMP, or Pthreads. The Kokkos library also provides data abstractions to adjust (at compile time) the memory layout of data structures like 2d and 3d arrays to optimize performance on different hardware. https://github.com/kokkos/kokkos

IPPL - Particles and Fields

Particles

- **Memory layout:** Struct of arrays
  Each attribute (e.g., position or velocity) is a Kokkos::View



Fields

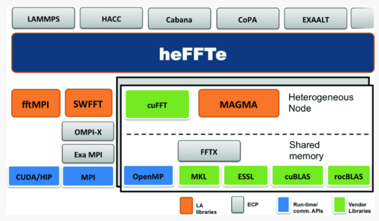
IPPL - Particles and Fields V 2.0

- IPPL core

	D-Operators	**NGP,CIC**
Fields	**Mesh**	**Particles**
Load Balancing	**Domain Decomp.**	**Communication**

IPPL - Particles and Fields V 2.0

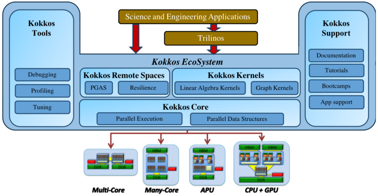
- heFFTe



heFFTe	D-Operators	NGP,CIC
Fields	Mesh	Particles
Load Balancing	Domain Decomp.	Communication

IPPL - Particles and Fields V 2.0

- Kokkos https://github.com/kokkos/kokkos-tutorials/blob/main/Intro-Short/KokkosTutorial_Short.pdf

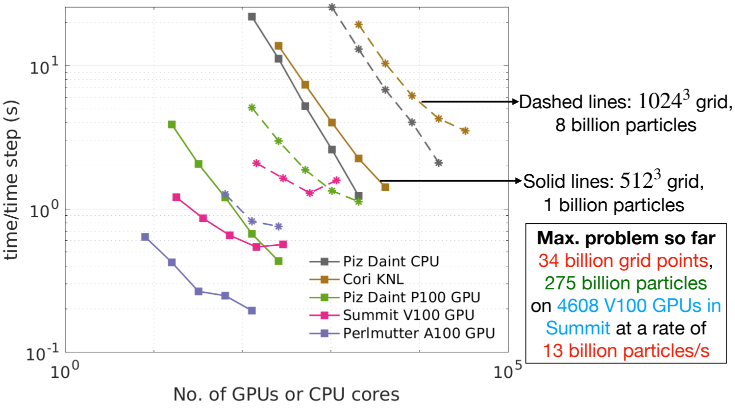


heFFTe	D-Operators	NGP,CIC
Fields	Mesh	Particles
Load Balancing	Domain Decomp.	Communication

Kokkos

Piz Daint Scaling - Linear Landau Damping

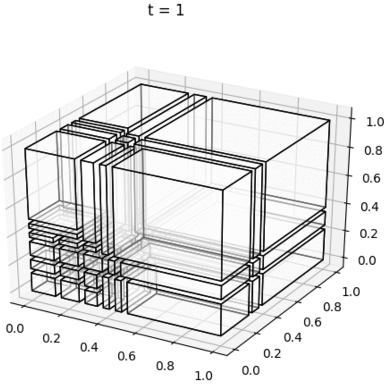
https://arxiv.org/abs/2205.11052 (Alessandro Vinciguerra, Michael Ligotino)



Max. problem so far
34 billion grid points,
275 billion particles
on 4608 V100 GPUs in Summit at a rate of
13 billion particles/s

ORB

64 ranks, 32^3 grid - work of Michael Ligotino http://amas.web.psi.ch/people/aadelmann/ETH-Accel-Lecture-1/projectscompleted/cse/ORBMichael.pdf



PAUL SCHERRER INSTITUT

References I

[arXiV:1905.06654] Andreas Adelmann, Pedro Calvo, Matthias Frey, Achim Gsell, Uldis Locans, Christof Metzger-Kraus, Nicole Neveu, Chris Rogers, Steve Russell, Suzanne Sheehy, Jochem Snuverink and Daniel Winklehner, OPAL a Versatile Tool for Charged Particle Accelerator Simulations, (2019), arXiv:1905.06654

[AA et al., JCP, **229** 12 (2010)] A. Adelmann, P. Arbenz, et al., J. Comp. Phys, 229 (12): 4554 (2010)

[N. Neveu, AA, et al. (2019)] Phys. Rev. Accel. Beams, **22**, 5, 054602, 2019

[U. Locans, AA, et al., CPC **215** (2017)] U. Locans, AA, et al., CPC **215** "71 - 80" (2017)

[AA et al., CPC **207** (2016)] AA, et al., CPC, dx.doi.org/10.1016/j.cpc.2016.05.013

[M. Toggweiler, AA, et al. J. Comp. Phys. **273** (2014)] , J. Comp. Phys. **273** : 255-267 (2014)

[Y. Ineichen, AA, et al. (2012)] Computer Science - Research and Development, pp. 1-8. Springer, Heidelberg, 2012.

[M. Frey et al. (2020)] M. Frey, U. Locans, A. Adelmann, On Architecture and Performance of Adaptive Mesh Refinement in an Electrostatics Particle-In-Cell Code, https://www.sciencedirect.com/science/article/pii/S0010465519302905