Recent advances in quasi-static Particle-in-Cell simulations for modeling plasma accelerators

S. Diederichs,[1,2,3]

[1] Deutsches Elektronen-Synchrotron DESY, Germany
[2] Lawrence Berkeley National Laboratory, USA
[3] University of Hamburg, Germany

1. Overview of QS PIC
2. Towards faster simulations
3. HiPACE++

Particle-in-Cell simulations are critical for modelling plasma accelerators

PIC simulations are **essential**

- validating experiments
- design studies
- new concepts

and **affordable**

Particle-in-Cell simulations are critical for modelling plasma accelerators

PIC simulations are **essential**
- validating experiments
- design studies
- new concepts

and **affordable**

ESPP Accelerator R&D Roadmap
including a
"***feasibility study**, mostly theory*
*and **simulation driven***"

at 190 GeV, with ≤135 nm emittance,
and 833 pC charge

Particle-in-Cell simulations are critical for modelling plasma accelerators

PIC simulations are **essential**

- validating experiments
- design studies
- new concepts

and ~~**affordable**~~

expensive, beyond feasibility

ESPP Accelerator R&D Roadmap including a "***feasibility study**, mostly theory and **simulation driven***"

at 190 GeV, with ≤135 nm emittance, and 833 pC charge

→ Simulation costs **must** be reduced by orders of magnitude!

Different approaches for potential cost reduction

Models and algorithms
- Boosted frame
- **Quasi-static Approximation**
- Reduced wakefield models
- Advanced field solvers
- Mesh refinement

Reduced geometries
- 2D RZ
- Quasi 3D

High-Performance Computing
- Accelerated computing (mostly GPUs)
- Scalability

Different approaches for potential cost reduction

<u>Models and algorithms</u>
- ~~Boosted frame~~
- **Quasi-static Approximation**
- Reduced wakefield models
- Advanced field solvers
- Mesh refinement

+

<u>Reduced geometries</u>
- 2D RZ
- Quasi 3D

+

<u>High-Performance Computing</u>
- Accelerated computing (mostly GPUs)
- Scalability

QSA codes:

<u>3D PIC:</u>
QuickPIC, WAND-PIC
HiPACE, HiPACE++, QV3D

<u>2D RZ + reduced wakefields:</u>
Wake-T

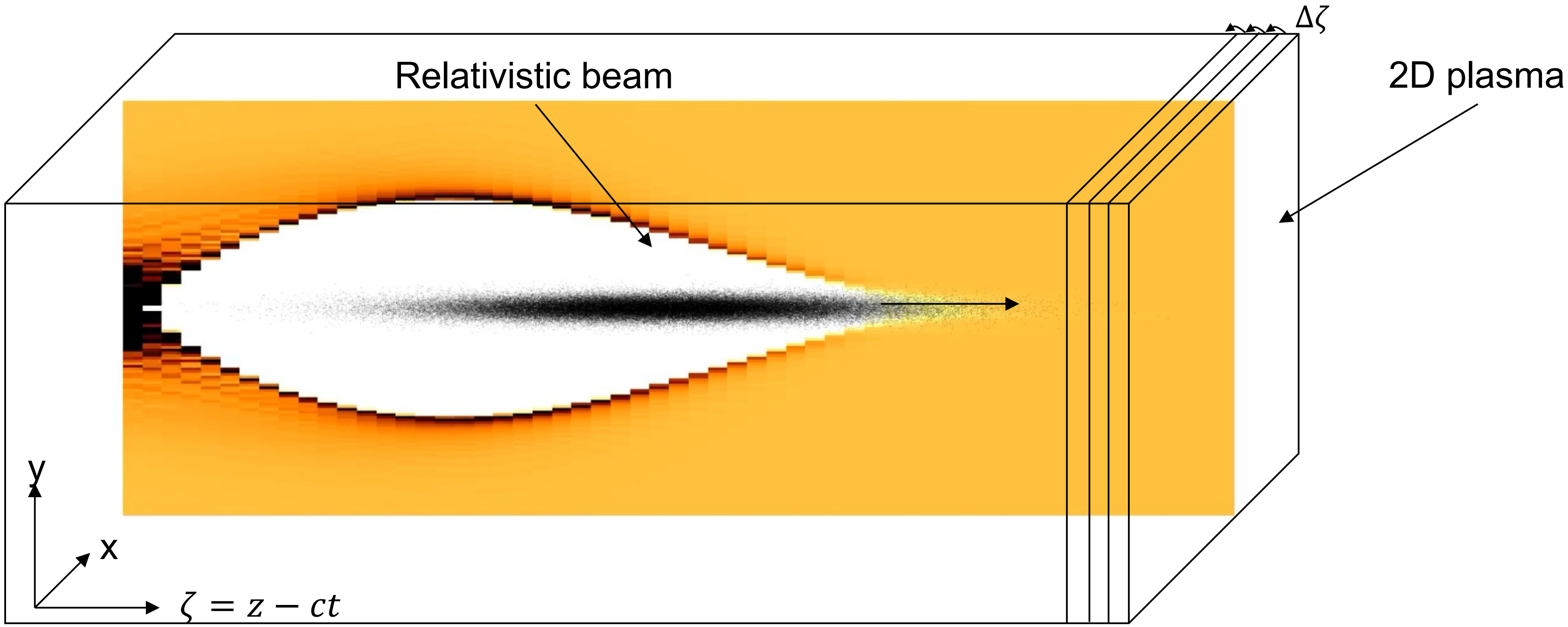
<u>2D RZ PIC:</u>
WAKE, LCODE, INF&RNO

<u>Quasi 3D PIC:</u>
QPAD

1. Overview of QS PIC
2. Towards faster simulations
3. HiPACE++

Quasi-static particle-in-cell method

separates plasma and beam evolution

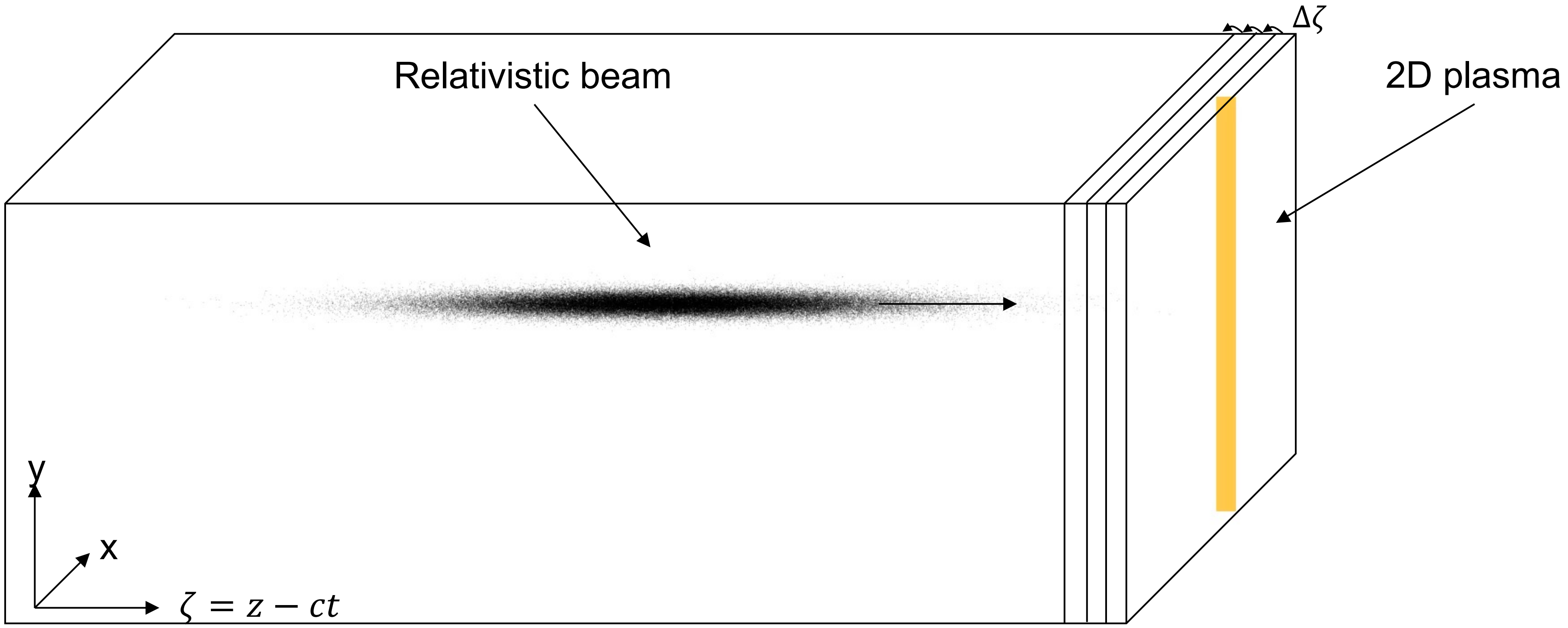


- Co-moving window
- Some time derivatives can be neglected
- Plasma evolution only depends on ζ

1. Head-to-tail plasma swipe, beam is frozen
2. At each slice: 2D PIC with Poisson solve
3. Push beam particles by **large** Δt, plasma is frozen

Quasi-static particle-in-cell method

separates plasma and beam evolution

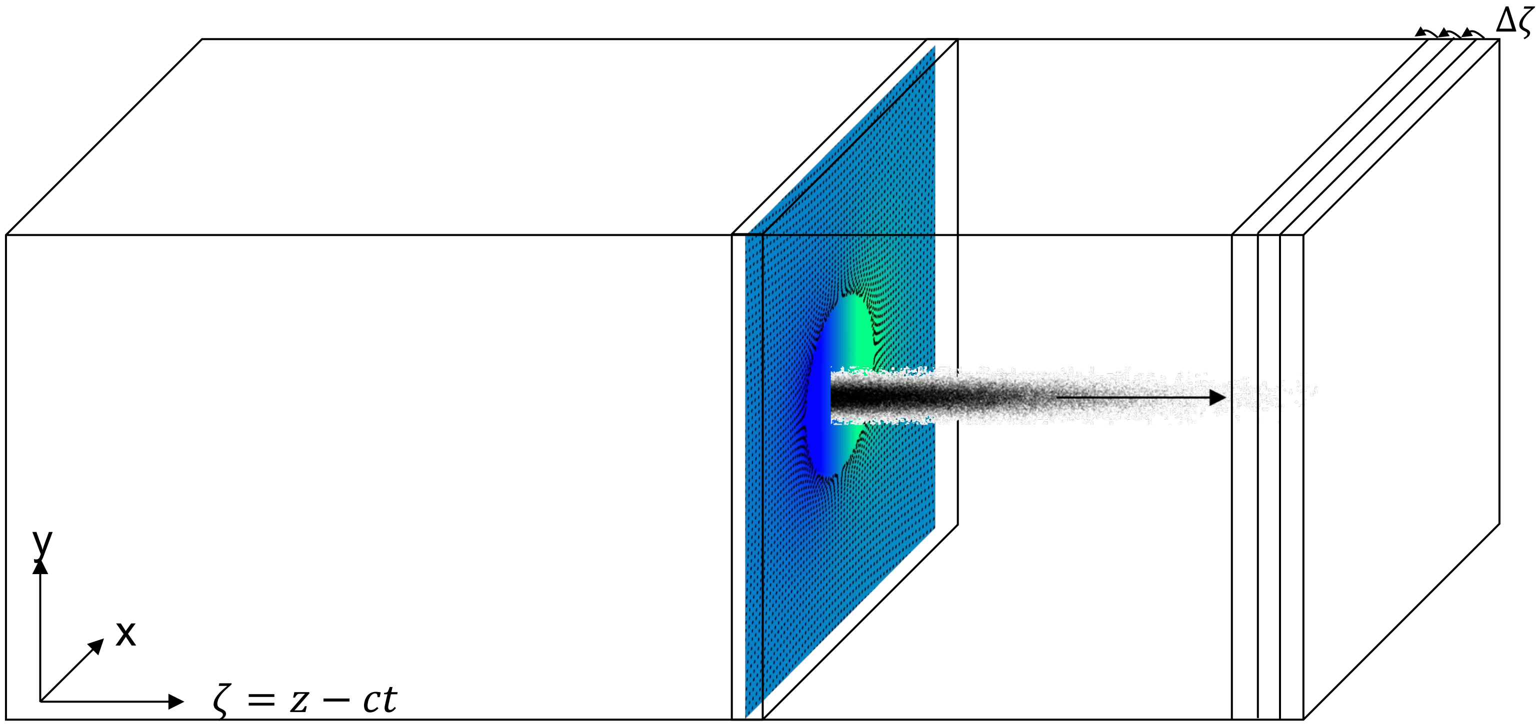


- Co-moving window
- Some time derivatives can be neglected
- Plasma evolution only depends on ζ

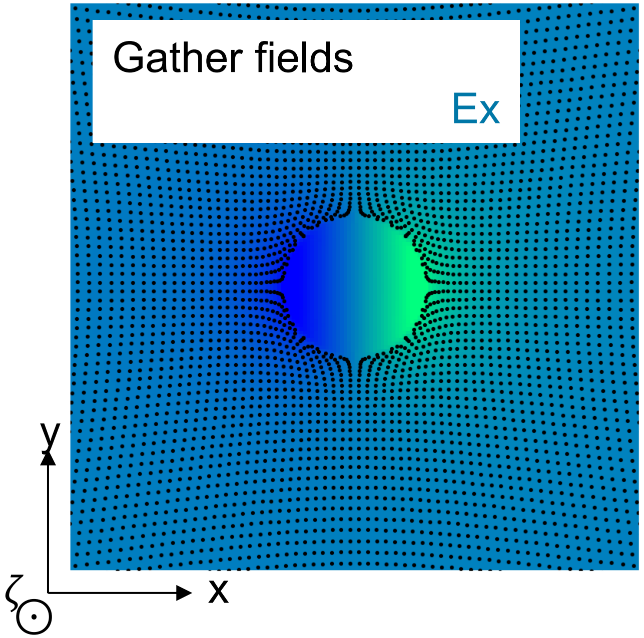
1. Head-to-tail plasma swipe, beam is frozen
2. At each slice: 2D PIC with Poisson solve
3. Push beam particles by **large** Δt, plasma is frozen

Quasi-static particle-in-cell method

separates plasma and beam evolution



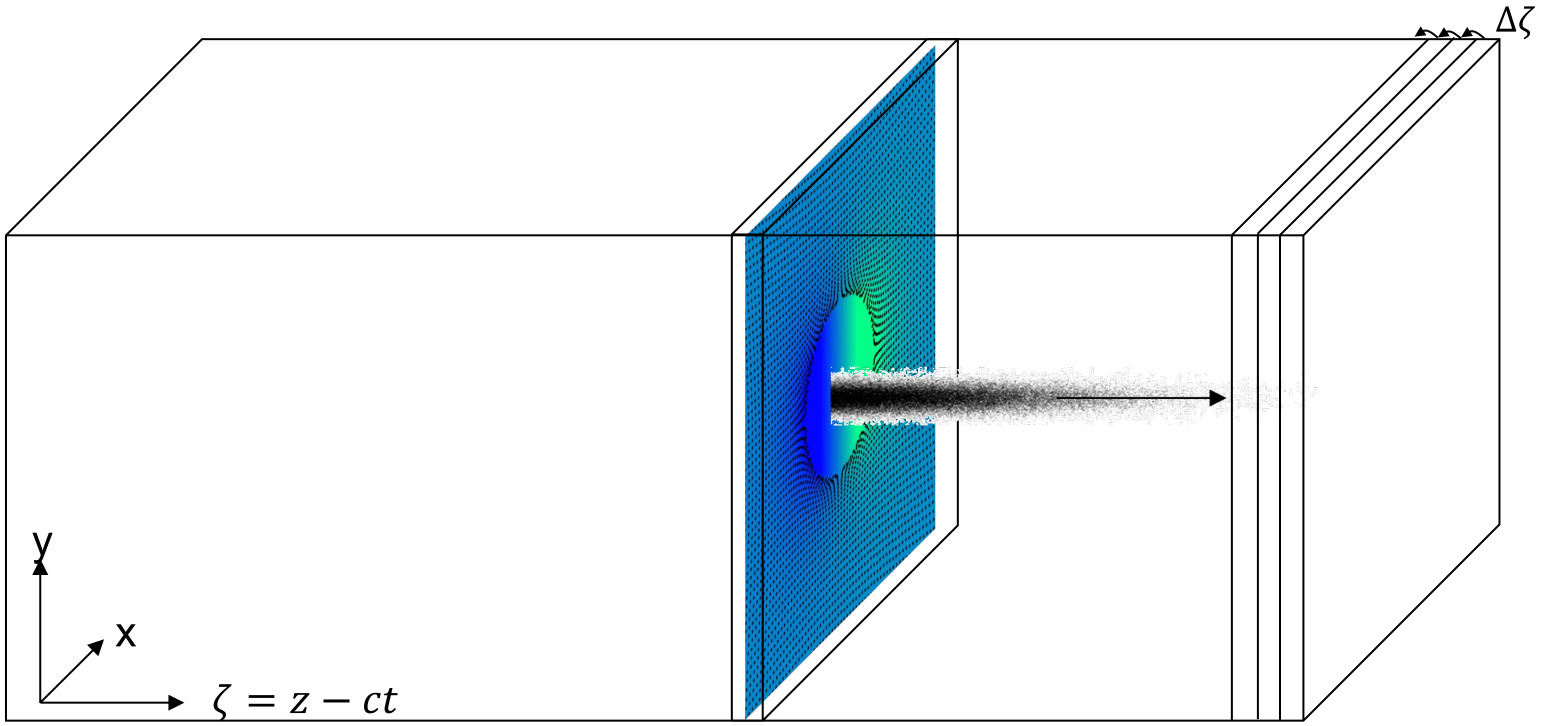
At each slice:



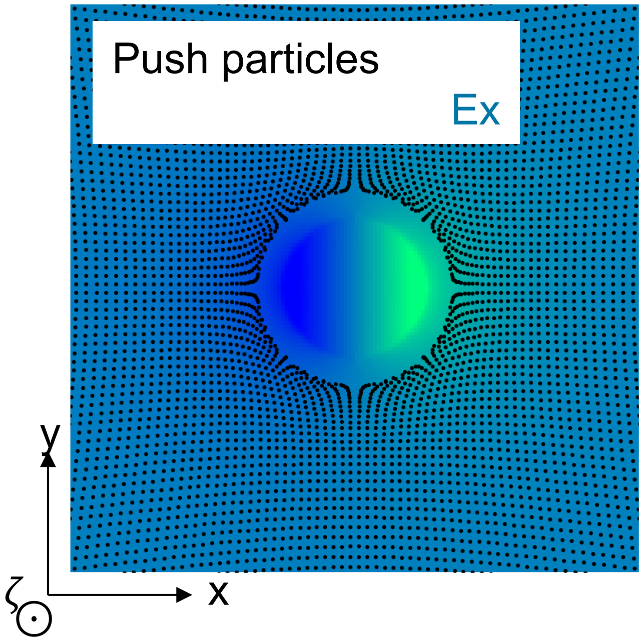
1. Head-to-tail plasma swipe, beam is frozen
2. At each slice: 2D PIC with Poisson solve
3. Push beam particles by **large** Δt, plasma is frozen

Quasi-static particle-in-cell method

separates plasma and beam evolution



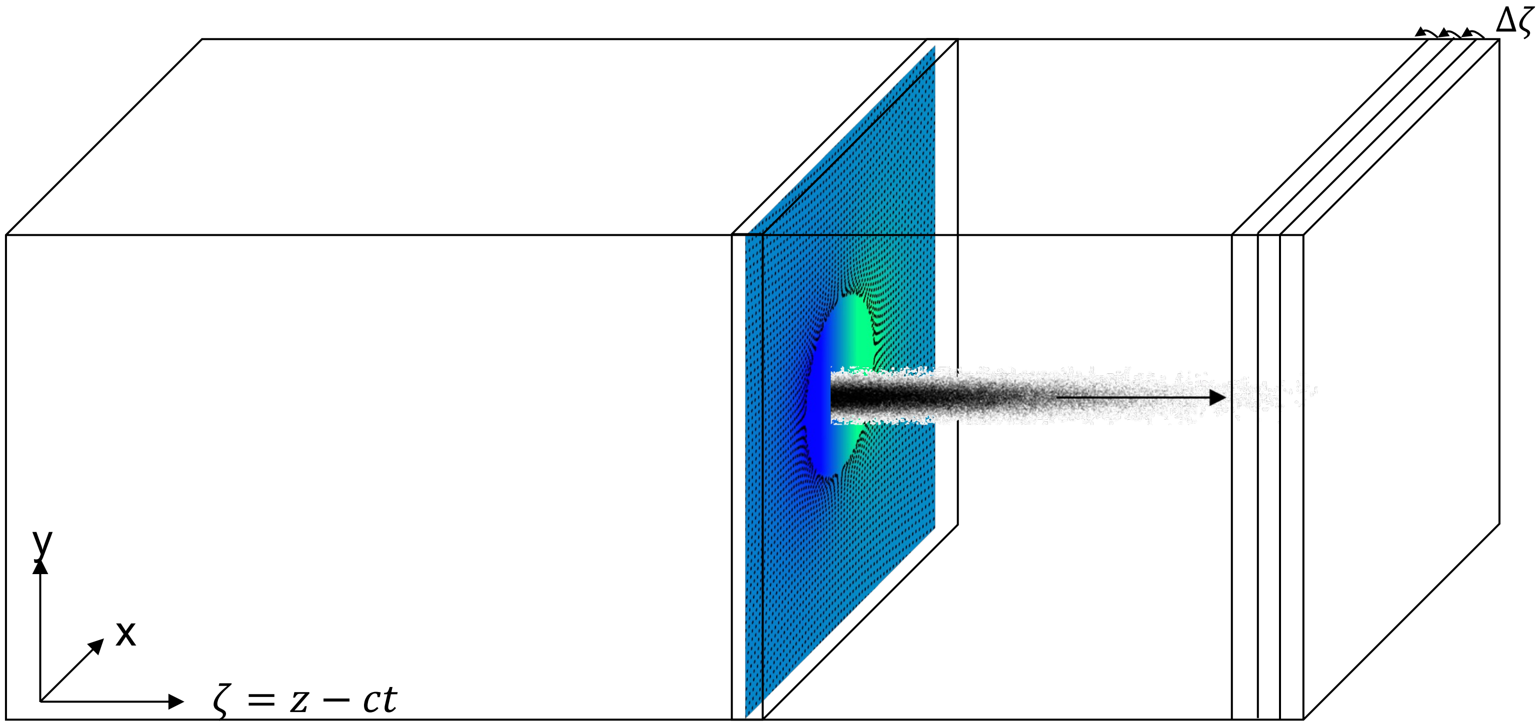
At each slice:



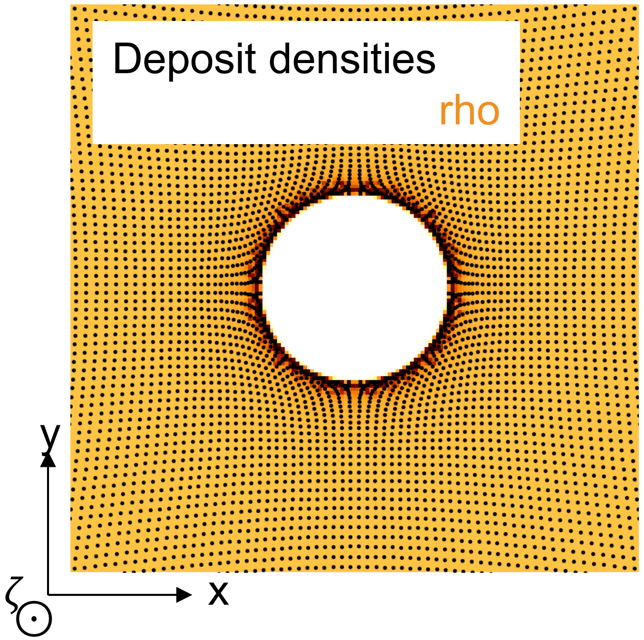
1. Head-to-tail plasma swipe, beam is frozen
2. At each slice: 2D PIC with Poisson solve
3. Push beam particles by **large** Δt, plasma is frozen

Quasi-static particle-in-cell method

separates plasma and beam evolution



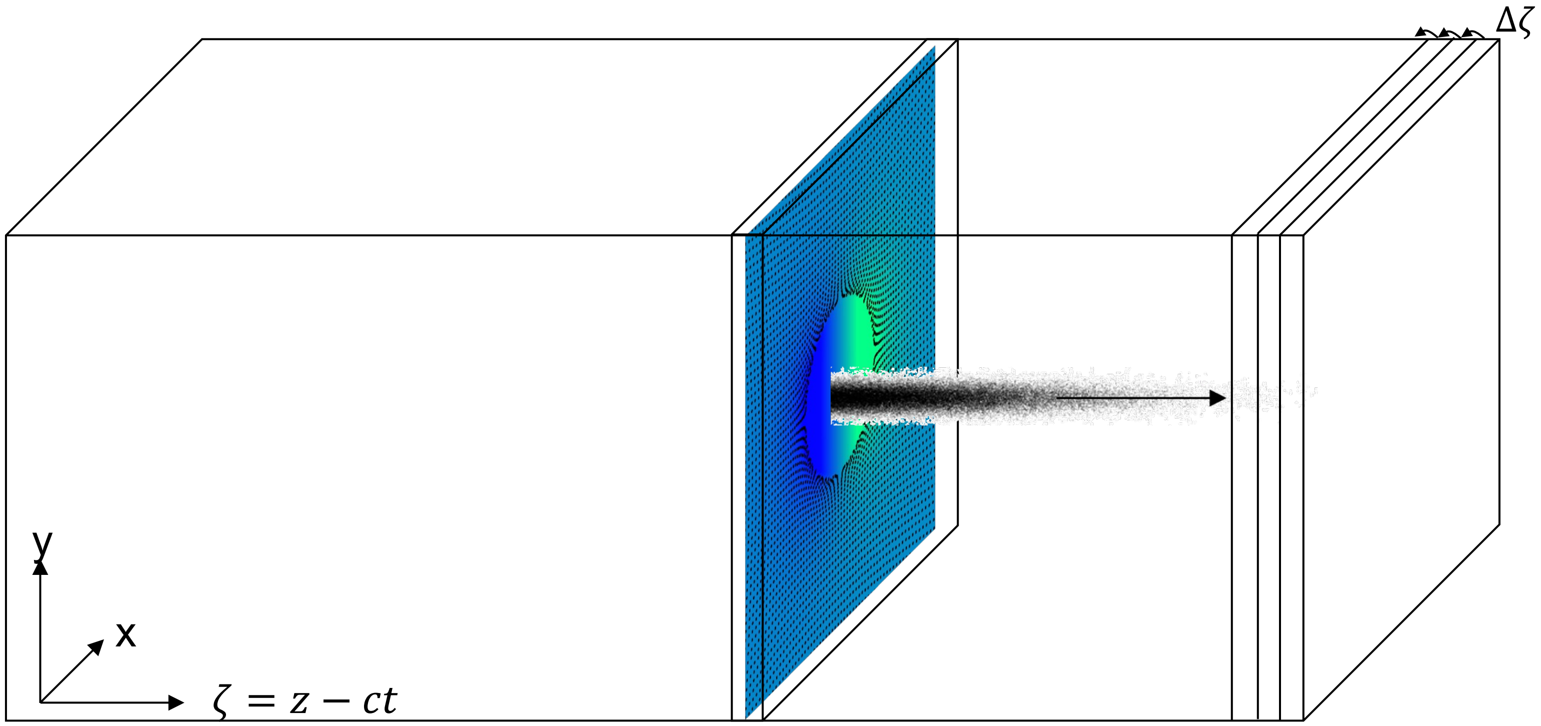
At each slice:



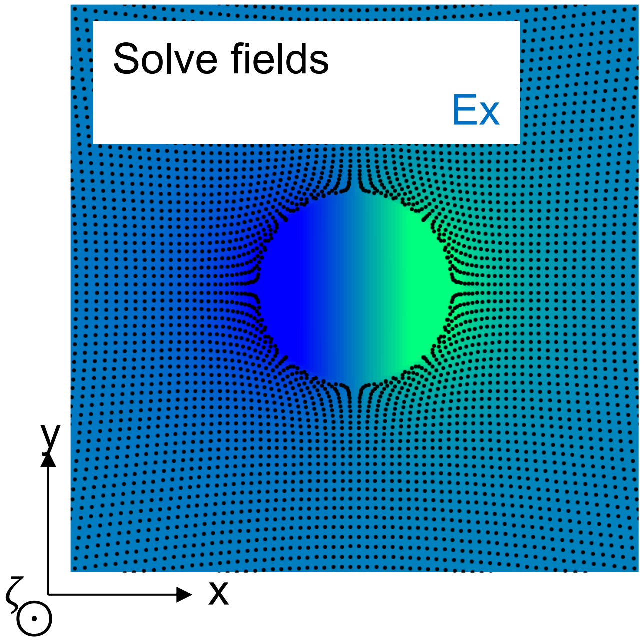
1. Head-to-tail plasma swipe, beam is frozen
2. At each slice: 2D PIC with Poisson solve
3. Push beam particles by **large** Δt, plasma is frozen

Quasi-static particle-in-cell method

separates plasma and beam evolution



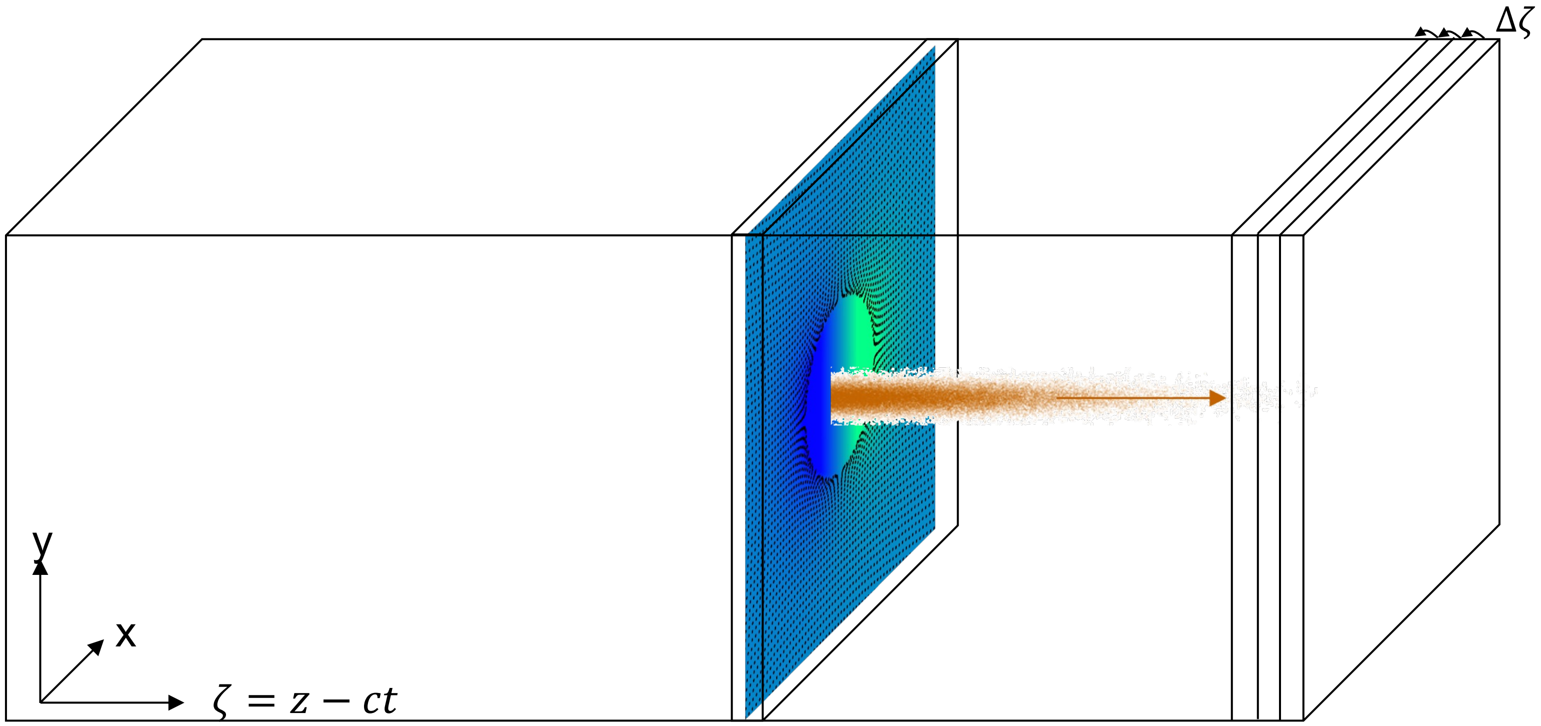
At each slice:



1. Head-to-tail plasma swipe, beam is frozen
2. At each slice: 2D PIC with Poisson solve
3. Push beam particles by **large** Δt, plasma is frozen

Quasi-static particle-in-cell method

separates plasma and beam evolution



1. Head-to-tail plasma swipe, beam is frozen
2. At each slice: 2D PIC with Poisson solve
3. Push beam particles by **large** $\boldsymbol{\Delta t}$, plasma is frozen

Explicit field solver

QSA PIC without predictor-corrector

- QSA PIC solves Maxwell equations in **co-moving frame** under **quasi-static approximation**[1]

$$\nabla_\perp^2 \psi = -\frac{1}{\epsilon_0}\left(\rho - \frac{1}{c} j_z\right)$$

$$E_x - c\,B_y = -\partial_x \psi$$

$$E_y + c\,B_x = -\partial_y \psi$$

$$\nabla_\perp^2 E_z = c\mu_0\left(\partial_x j_x + \partial_y j_y\right)$$

$$\nabla_\perp^2 B_x = \mu_0\left(-\partial_y j_z + \partial_\zeta j_y\right)$$

$$\nabla_\perp^2 B_y = \mu_0\left(\partial_x j_z - \partial_\zeta j_x\right)$$

$$\nabla_\perp^2 B_z = \mu_0\left(\partial_y j_x - \partial_x j_y\right)$$

Poisson equations:
solvable by FFT or multigrid solvers

[1] An et al., JCP 250 165-177 (2013)

Explicit field solver

QSA PIC without predictor-corrector

- QSA PIC solves Maxwell equations in **co-moving frame** under **quasi-static approximation**[1]

$$\nabla_\perp^2 \psi = -\frac{1}{\epsilon_0}\left(\rho - \frac{1}{c} j_z\right)$$

$$E_x - c\,B_y = -\partial_x \psi$$

$$E_y + c\,B_x = -\partial_y \psi$$

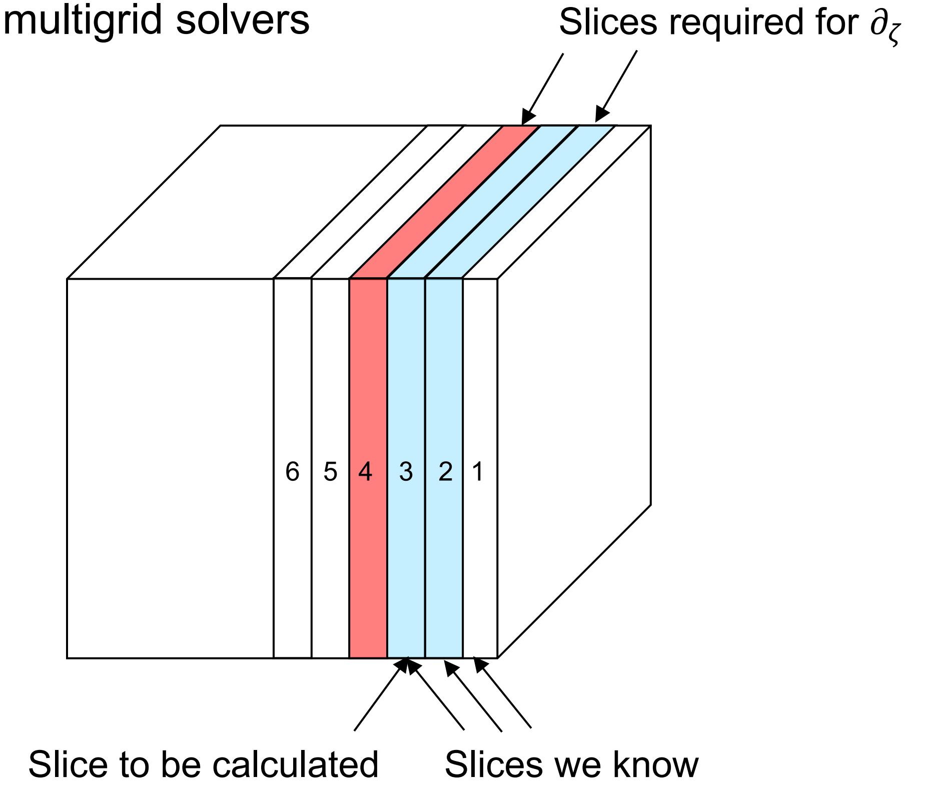
$$\nabla_\perp^2 E_z = c\mu_0 \left(\partial_x j_x + \partial_y j_y\right)$$

$$\nabla_\perp^2 B_x = \mu_0 \left(-\partial_y j_z + \boldsymbol{\partial_\zeta j_y}\right)$$

$$\nabla_\perp^2 B_y = \mu_0 \left(\partial_x j_z - \boldsymbol{\partial_\zeta j_x}\right)$$

$$\nabla_\perp^2 B_z = \mu_0 \left(\partial_y j_x - \partial_x j_y\right)$$

Poisson equations:
solvable by FFT or multigrid solvers



- Longitudinal derivatives not known!
  Predictor-corrector loop → expensive
  (involving particle push, current deposition, field solve)

[1] An et al., JCP 250 165-177 (2013)

Explicit field solver

QSA PIC without predictor-corrector

- QSA PIC solves Maxwell equations in **co-moving frame** under **quasi-static approximation**[1]

$$\nabla_\perp^2 \psi = -\frac{1}{\epsilon_0}\left(\rho - \frac{1}{c} j_z\right)$$

Poisson equations:
solvable by FFT or multigrid solvers

$$E_x - c\, B_y = -\partial_x \psi$$

$$E_y + c\, B_x = -\partial_y \psi$$

$$\nabla_\perp^2 E_z = c\mu_0 \left(\partial_x j_x + \partial_y j_y\right)$$

$$\nabla_\perp^2 B_x = \mu_0 \left(-\partial_y j_z + \boldsymbol{\partial_\zeta j_y}\right)$$

$$\nabla_\perp^2 B_y = \mu_0 \left(\partial_x j_z - \boldsymbol{\partial_\zeta j_x}\right)$$

$$\nabla_\perp^2 B_z = \mu_0 \left(\partial_y j_x - \partial_x j_y\right)$$

Derivatives can be expressed explicitly[2,3] with elaborate source term S

$$\nabla_\perp^2 B_\perp - \frac{n^*}{1+\psi} B_\perp = -[e_z \times S]$$

Screened Poisson equation:
solvable by multigrid solvers

T. Wang et al., from
Cornell University,
University of Texas at Austin

- Longitudinal derivatives not known!
  Predictor-corrector loop → expensive
  (involving particle push, current deposition, field solve)

[1] An et al., JCP 250 165-177 (2013)
[2] Wang et al., PoP 24, 103117 (2017)
[3] Wang et al., arXiv:2012.00881v2 (2022)

Explicit field solver

QSA PIC without predictor-corrector

- QSA PIC solves Maxwell equations in **co-moving frame** under **quasi-static approximation**[1]

$$\nabla_\perp^2 \psi = -\frac{1}{\epsilon_0}\left(\rho - \frac{1}{c} j_z\right)$$

Poisson equations:
solvable by FFT or multigrid solvers

$$E_x - c\, B_y = -\partial_x \psi$$

$$E_y + c\, B_x = -\partial_y \psi$$

$$\nabla_\perp^2 E_z = c\mu_0 \left(\partial_x j_x + \partial_y j_y\right)$$

$$\nabla_\perp^2 B_x = \mu_0 \left(-\partial_y j_z + \boldsymbol{\partial_\zeta j_y}\right)$$

$$\nabla_\perp^2 B_y = \mu_0 \left(\partial_x j_z - \boldsymbol{\partial_\zeta j_x}\right)$$

$$\nabla_\perp^2 B_z = \mu_0 \left(\partial_y j_x - \partial_x j_y\right)$$

Derivatives can be expressed explicitly[2,3] with elaborate source term S

$$\nabla_\perp^2 B_\perp - \frac{n^*}{1+\psi} B_\perp = -[e_z \times S]$$

Screened Poisson equation:
solvable by multigrid solvers

- Longitudinal derivatives not known!
  Predictor-corrector loop → expensive
  (involving particle push, current deposition, field solve)

T. Wang et al., from
Cornell University,
University of Texas at Austin

Implemented in
WAND-PIC[3], HiPACE++

Full comparison difficult due to different numerical parameters

Personal experience:
Explicit solver
→ converges faster
→ has increased stability
→ is easier to use!

[1] An et al., JCP 250 165-177 (2013)
[2] Wang et al., PoP 24, 103117 (2017)
[3] Wang et al., arXiv:2012.00881v2 (2022)

Reminder: ~100 nanometer emittances requested

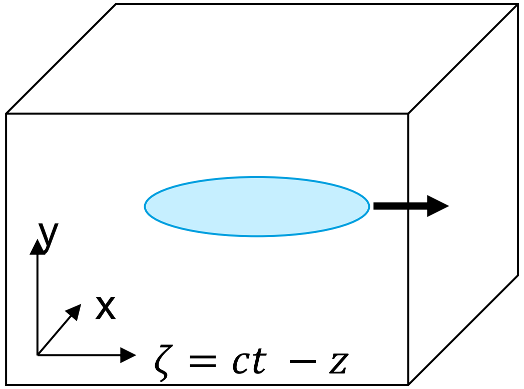
Orders of magnitude speedup required

1. Overview of QS PIC
2. Towards faster simulations
3. HiPACE++

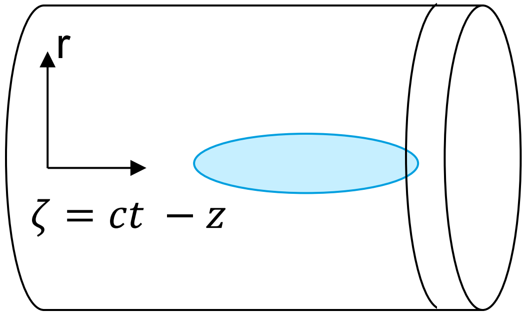
Reduced geometries enable massive speedup

: ~ 100 nanometer emittances requested!

3D



2D RZ



$\partial_\theta = 0$

→ **Extremely efficient**, $> \mathbf{10^4}$ **speedup**[2]
only correct for problems with cylindrical symmetry

2D RZ + reduced wakefields:
Wake-T

2D RZ PIC:
WAKE[1]*, LCODE, INF&RNO*

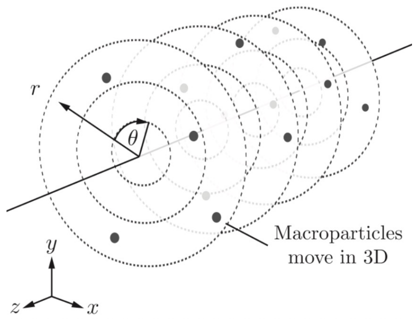
[1] Mora, Antonsen PoP 1997
[2]Benedetti et al., AIP conference proceedings (2017)

QSA PIC and azimuthal decomposition

Obtaining 3D physics at the cost of 2D simulations

Fourier decomposition into higher order azimuthal modes

- First proposed by Lifschitz et al. for Calder Circ[1]
  also implemented in open source code FBPIC[2]
- High accuracy for close-to-axisymmetric problems



From [3]

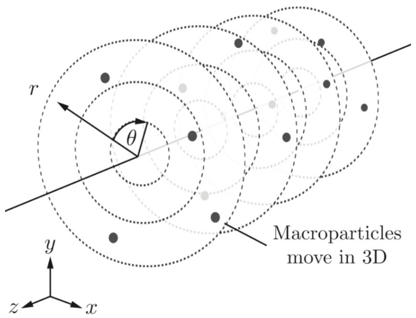
[1] Lifschitz et al., JCP 228, 5 (2009)
[2] https://github.com/fbpic/fbpic
[3] https://fbpic.github.io/overview/pic_algorithm.html

QSA PIC and azimuthal decomposition

Obtaining 3D physics at the cost of 2D simulations

Fourier decomposition into higher order azimuthal modes

- First proposed by Lifschitz et al. for Calder Circ[1]
  also implemented in open source code FBPIC[2]
- High accuracy for close-to-axisymmetric problems

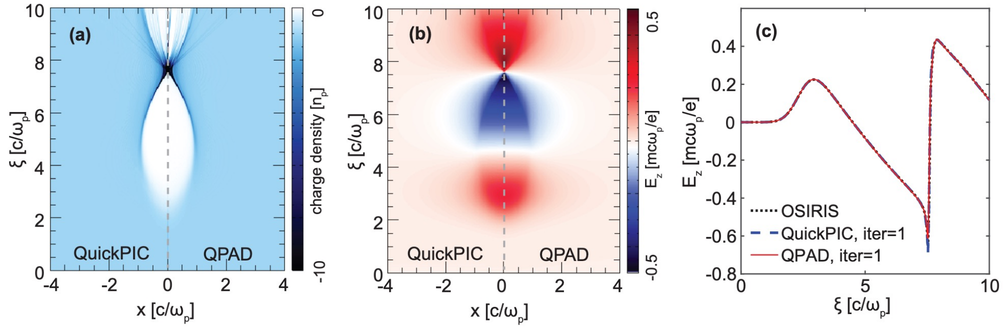


From [3]

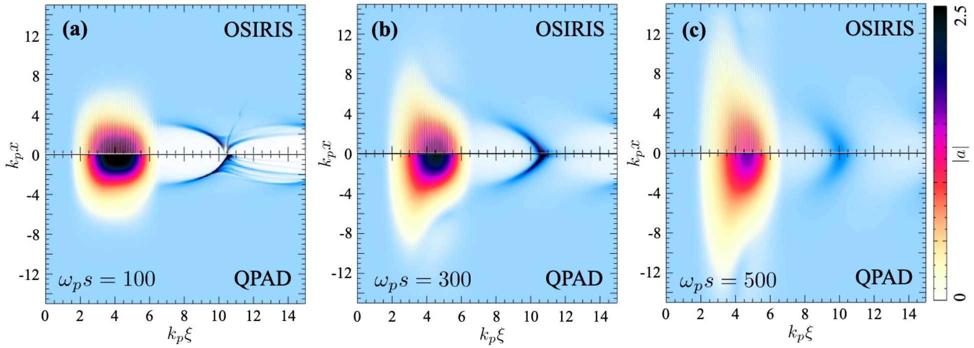
[1] Lifschitz et al., JCP 228, 5 (2009)
[2] https://github.com/fbpic/fbpic
[3] https://fbpic.github.io/overview/pic_algorithm.html
[4] Li et al., CPC 261, 107784, (2020)
[5] Li et al., JCP 111599, (2022)

QPAD[4]: QuickPIC+ Azimuthal Decomposition

- Large time steps due to QSA
- Reduced computational costs due to reduced geometry



- Laser envelope solver[5]: $> \mathbf{10^4}$ **speedup**



Reminder: ~100 nanometer emittances requested ***in 3D***

Orders of magnitude speedup required

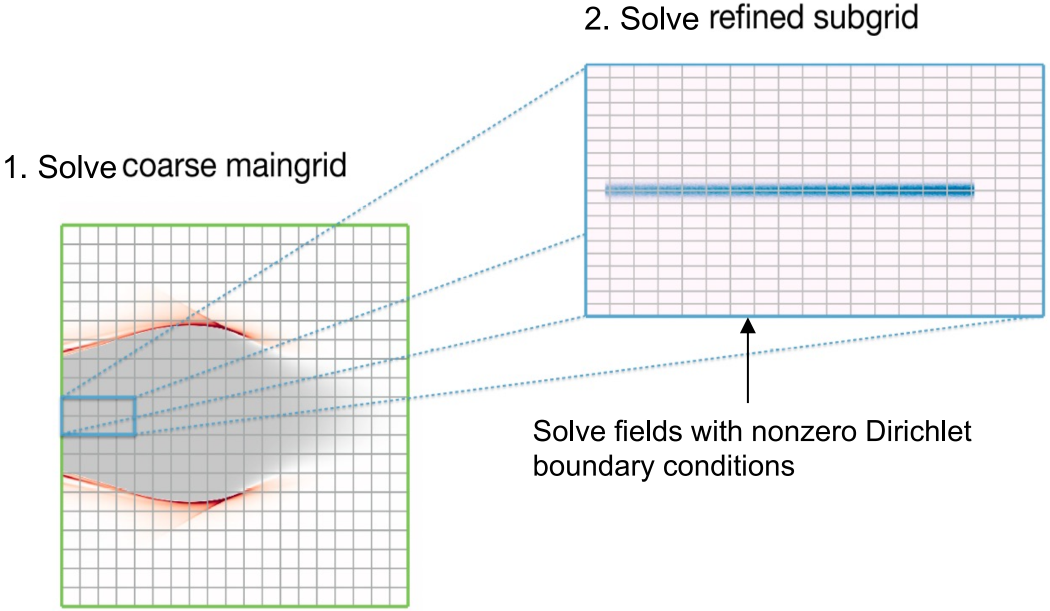
Mesh refinement in QSA PIC

Enables ultra-high resolution to model collider-relevant emittance beams

: ~ 100 nanometer emittances requested!

Transverse mesh refinement in QSA PIC:
Convoluted QSA PIC with new boundary conditions[1]

- nm-scale resolution possible!
- Implemented in HiPACE, HiPACE++, QuickPIC, and INF&RNO

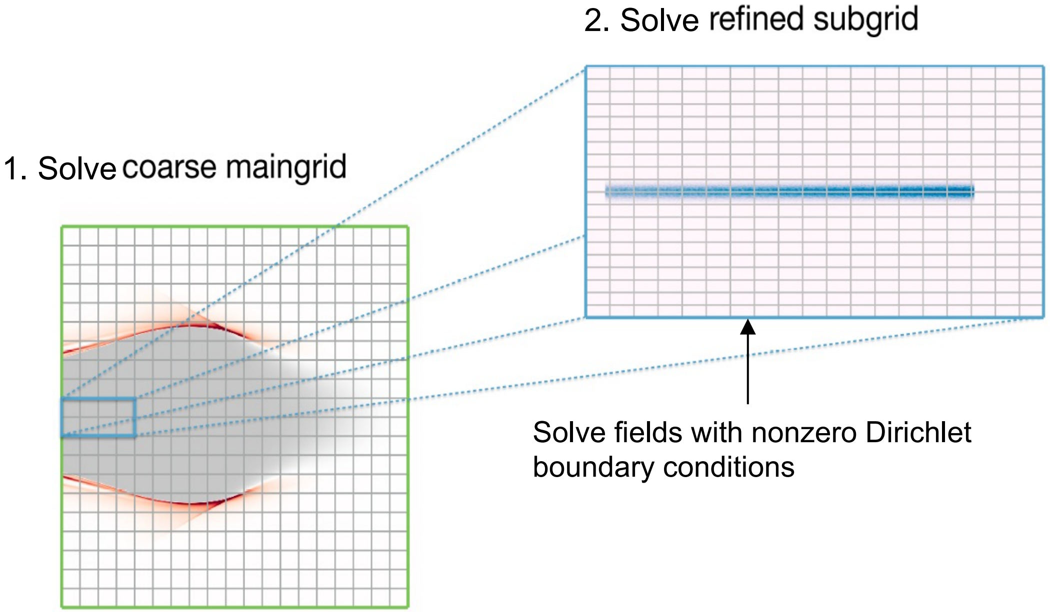


[1] Mehrling et al., 2018 AAC proceedings (2019)

Mesh refinement in QSA PIC

Enables ultra-high resolution to model collider-relevant emittance beams

: ~ 100 nanometer emittances requested!



Transverse mesh refinement in QSA PIC:
Convoluted QSA PIC with new boundary conditions[1]

- nm-scale resolution possible!
- Implemented in HiPACE, HiPACE++, QuickPIC, and INF&RNO
- Currently <u>no plasma exchange</u> between meshes

→ **Key developments** needed to meet roadmap goals!

[1] Mehrling et al., 2018 AAC proceedings (2019)

Supercomputer landscape is evolving

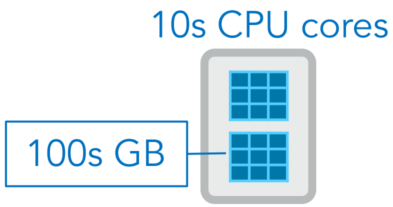
1. Overview of QS PIC

2. Towards faster simulations

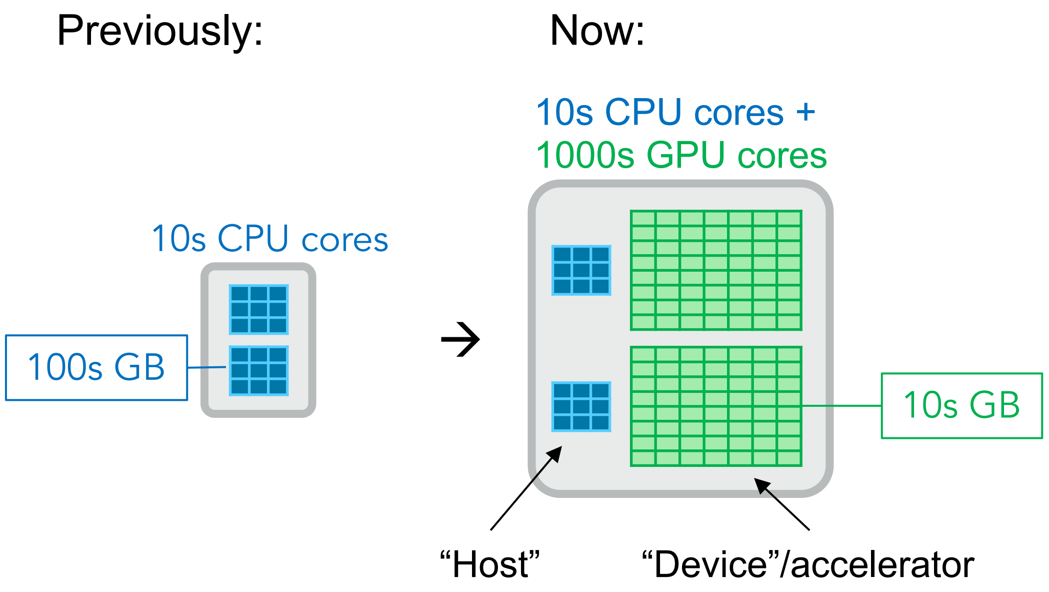
3. HiPACE++

Supercomputer landscape is evolving

Previously:



Supercomputer landscape is evolving



GPUs:

- Highly performant: 14 of the top 20 of the top500 GPU-based[1]
- Energy efficient → **reduce environmental impact**

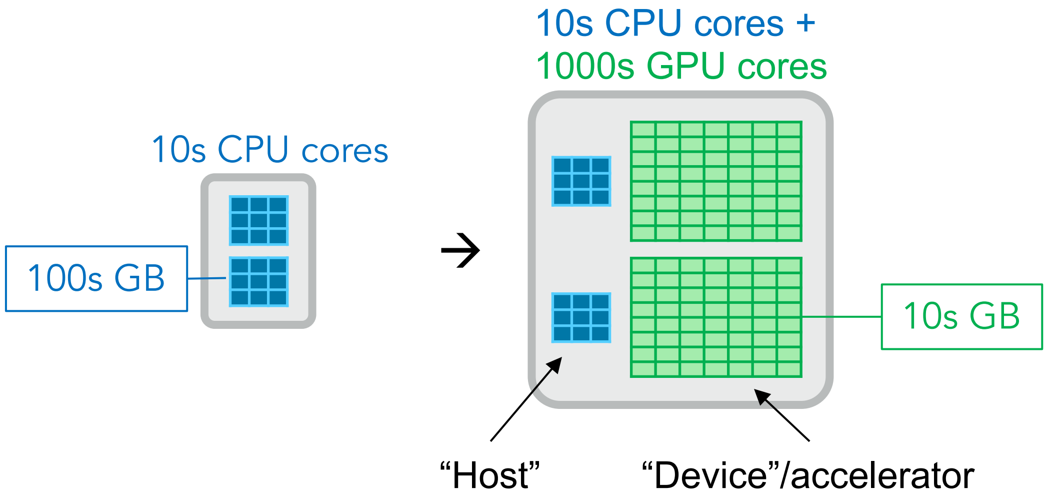
Top 20 of the Green500[1]

Rank	Name	Hardware
1	Frontier TDS	AMD Instinct MI250X
2	Frontier	AMD Instinct MI250X
3	LUMI	AMD Instinct MI250X
4	Adastra	AMD Instinct MI250X
5	ATOS THX.A.B	NVIDIA A100
6	MN-3	Xeon Platinum 8260M
7	SSC-21 Scalable Module	NVIDIA A100
8	Tethys	NVIDIA A100
9	Wilkes-3	NVIDIA A100
10	Athena	NVIDIA A100
11	Phoenix - 2022	NVIDIA A100
12	HiPerGator AI	NVIDIA A100
13	Snellius Phase 1 GPU	NVIDIA A100
14	Perlmutter	NVIDIA A100
15	Karolina, GPU partition	NVIDIA A100
16	MeluXina - Accelerator Module	NVIDIA A100
17	Alex	NVIDIA A100
18	NVIDIA DGX SuperPOD	NVIDIA A100
19	JUWELS Booster Module	NVIDIA A100
20	JURECA Data Centric Module	NVIDIA A100

[1] https://www.top500.org/lists/green500/2022/06/

Supercomputer landscape is evolving

Previously: Now:



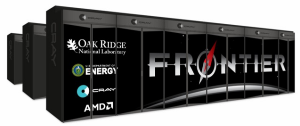
GPUs:

- Highly performant: 14 of the top 20 of the top500 GPU-based
- Energy efficient → **reduce environmental impact**
- Inherently massively parallel devices

- require specific code
  → portability layers

NVIDIA CUDA

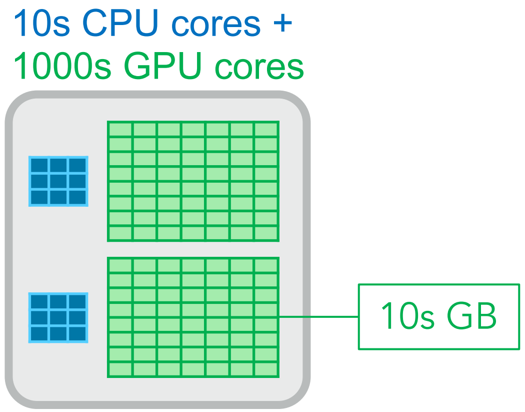
FUJITSU

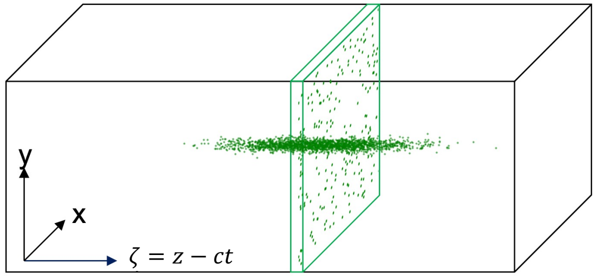


AMD

intel (expected 2022)

Quasi-static PIC is a prime candidate for GPU computing





GPUs:	Massive parallelism	moderate memory (10s GB)
QSA PIC:	✓ many particles, many cells	✓ only 2D slices, 2D plasma, and beams

HiPACE++: quasi-static PIC on GPU

- full re-writing of HiPACE (DESY, LBNL) for accelerated (GPU) computing
- Built on top of AMReX[1] (LBNL)
  Data structures and communications
  - **Performance-portability layer**
- openPMD[2] I/O
- Collaboration with the ECP WarpX[3,4] team at LBNL
- HPC programming standards
  - Documented, **open-source**, open-repository, CMake
  - Automated testing (CI)

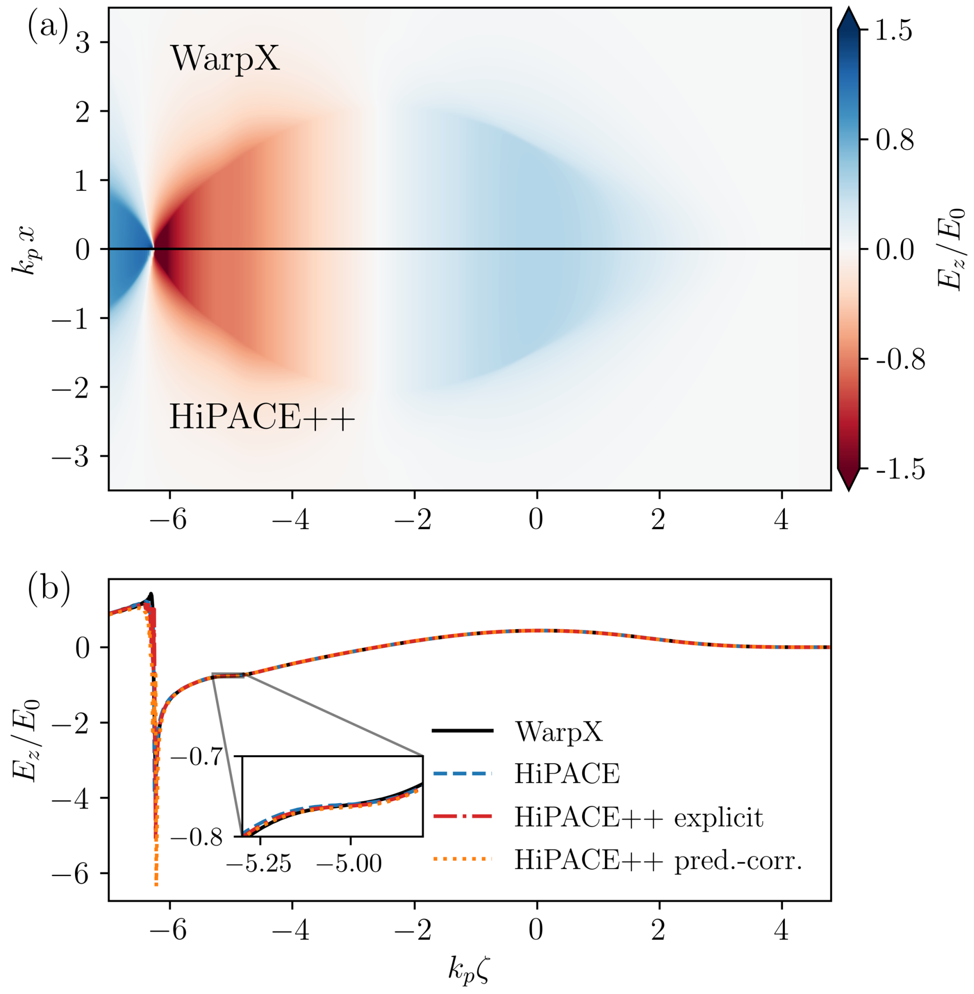
HiPACE++

3D quasi-static PIC code
Developed by DESY & LBNL:
https://github.com/Hi-PACE/hipace
Language: C++
Doc: https://hipace.readthedocs.io
2021, just starting

[1] https://github.com/AMReX-Codes/amrex
[2] https://github.com/openPMD/openPMD-standard
[3] https://github.com/ECP-WarpX/WarpX
[4] Myers et al., Parallel Comput. 2021

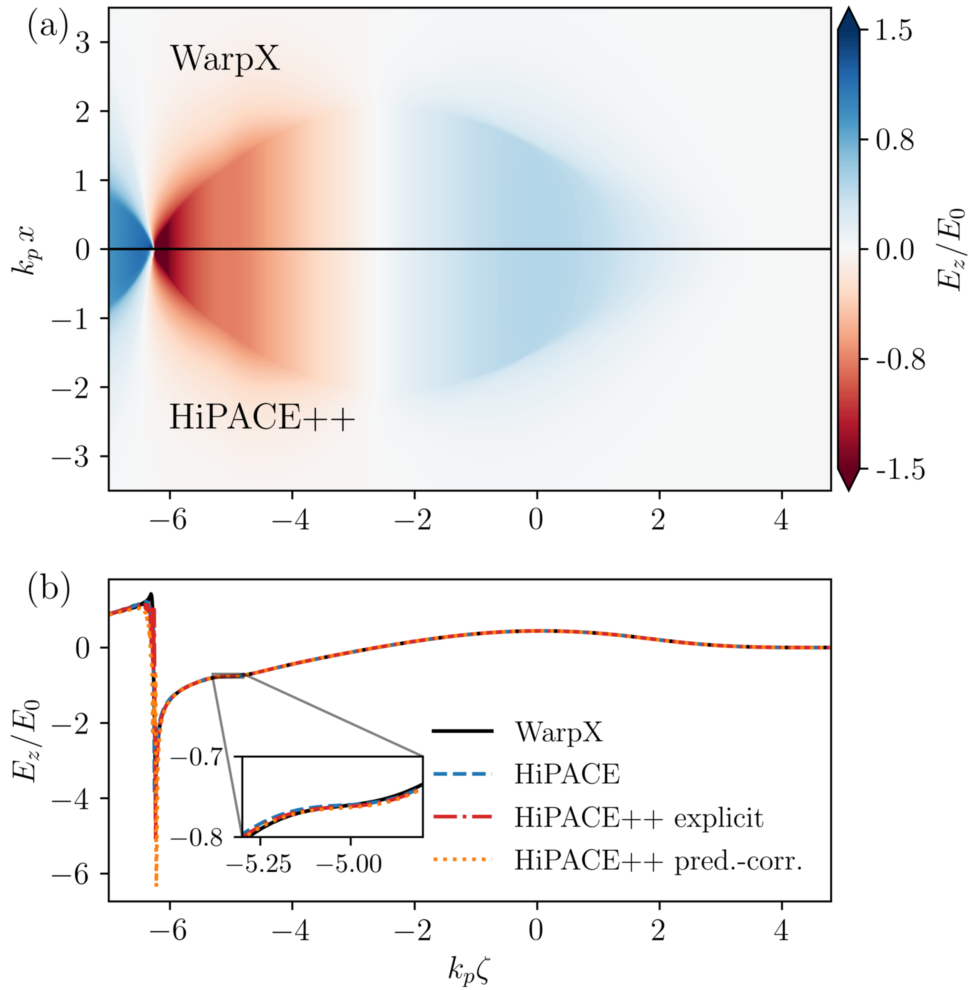
Single GPU easily outperforms many CPU cores

Benchmark and performance

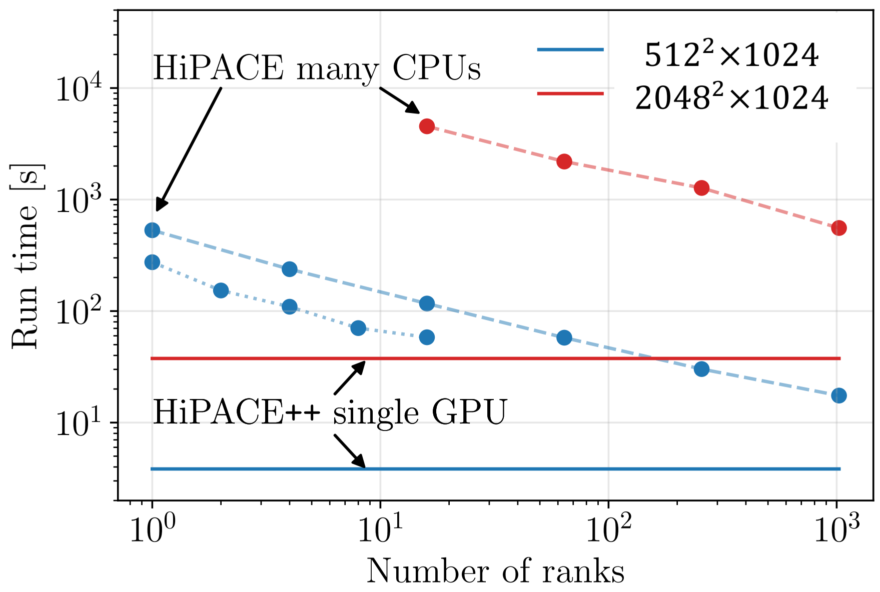


Single GPU easily outperforms many CPU cores

Benchmark and performance



Performed on the JUWELS Booster



NVIDIA A100 vs. transversely parallelized on many CPUs

10x faster, 1000x cheaper for production runs

Node-to-node comparison: estimated to be **~10x more energy-efficient**

Challenging simulation scenarios become affordable

On GPU-equipped supercomputers and laptops

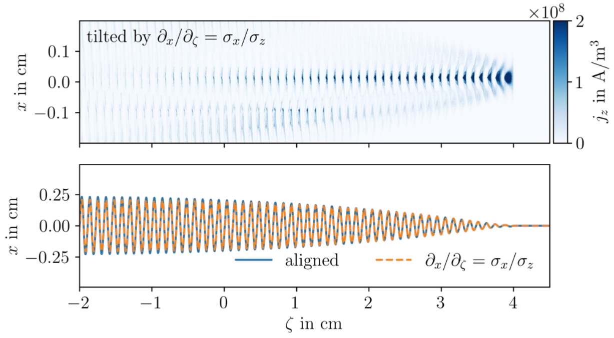
Challenging simulation scenarios become affordable

On GPU-equipped supercomputers and laptops

*Our allocation: 26 000 node hours
"large allocation": 156 000 node hours

Illustration 1:
Tilted proton bunch in AWAKE

Challenge?
Long beam, many beam particles, long simulation box, long plasma



Costs: 1 node hour*
on the JUWELS Booster (16 nodes),
in FP64, $512 \times 512 \times 2048$ grid points,
120×10^6 beam particles, **400 time steps**

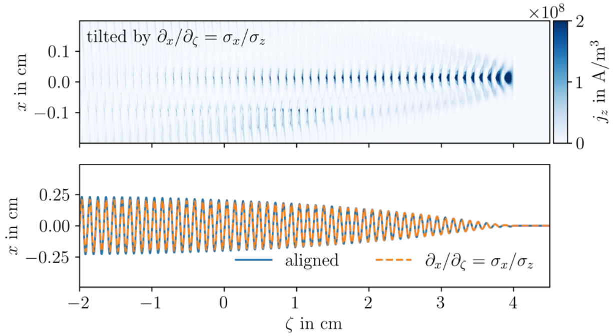
Challenging simulation scenarios become affordable

On GPU-equipped supercomputers and laptops

*Our allocation: 26 000 node hours
"large allocation": 156 000 node hours

Illustration 1:
Tilted proton bunch in AWAKE

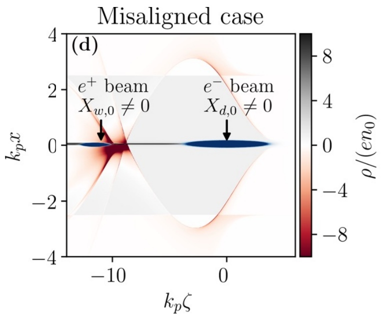
Challenge?
Long beam, many beam particles, long simulation box, long plasma



Costs: 1 node hour*
on the JUWELS Booster (16 nodes),
in FP64, **512×512×2048** grid points,
120×10^6 beam particles, **400 time steps**

Illustration 2:
e^+ acceleration in a plasma column

Challenge?
High-resolution, sharp plasma spikes, many time steps, many beam particles



Costs: 18 node hours*
on the JUWELS Booster (32 nodes),
in FP64, **4096×4096×2048** grid points,
225×10^6 beam particles, **750 time steps**

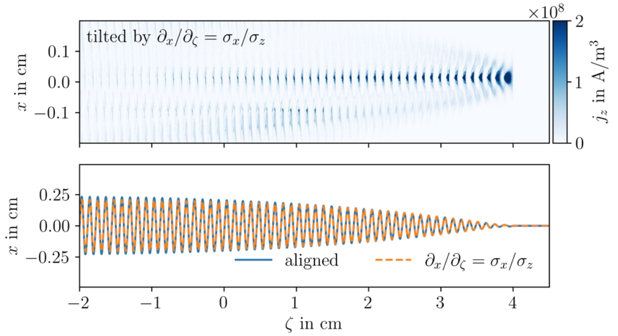
Challenging simulation scenarios become affordable

On GPU-equipped supercomputers and laptops

*Our allocation: 26 000 node hours
"large allocation": 156 000 node hours

Illustration 1:
Tilted proton bunch in AWAKE

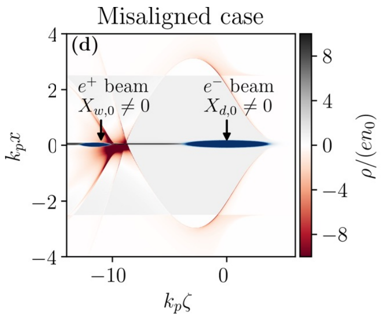
Challenge?
Long beam, many beam particles, long simulation box, long plasma



Costs: 1 node hour*
on the JUWELS Booster (16 nodes),
in FP64, **512×512×2048** grid points,
120×10^6 beam particles, **400 time steps**

Illustration 2:
e^+ acceleration in a plasma column

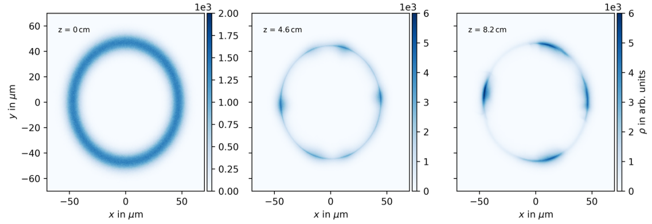
Challenge?
High-resolution, sharp plasma spikes, many time steps, many beam particles



Costs: 18 node hours*
on the JUWELS Booster (32 nodes),
in FP64, **4096×4096×2048** grid points,
225×10^6 beam particles, **750 time steps**

Illustration 3:
e^+ acceleration with a hollow-core driver
(Jain et al., PRL 2015)

Challenge?
Standard 3D simulation



Costs: 1 hour on a laptop
(NVIDIA RTX2070)
in FP32, **1024×1024×1024** grid points,
10^7 beam particles, **300 time steps**

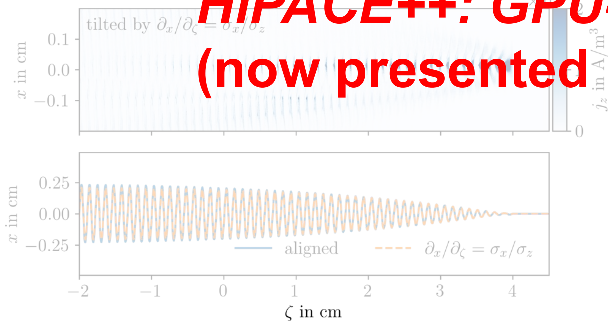
Challenging simulation scenarios become affordable

On GPU-equipped supercomputers and laptops

*Our allocation: 26 000 node hours
"large allocation": 156 000 node hours

Illustration 1:
Tilted proton bunch in AWAKE

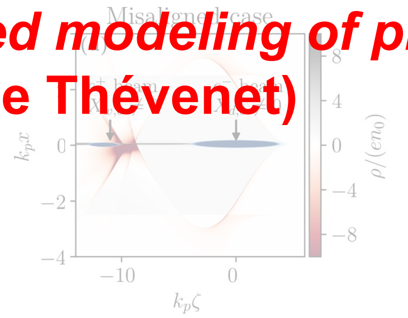
Challenge?
Long beam, many beam particles, long simulation box, long plasma



Costs: 1 node hour* on the JUWELS Booster (16 nodes), in FP64, $512\times512\times2048$ grid points, 120×10^6 beam particles, **400 time steps**

Illustration 2:
e^+ acceleration in a plasma column

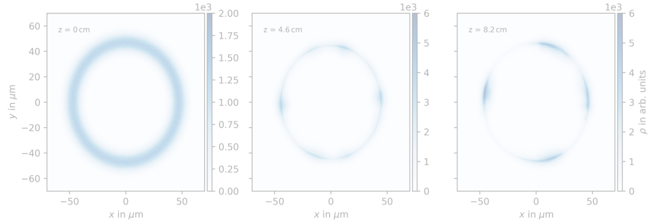
Challenge?
High-resolution, sharp plasma spikes, many time steps, many beam particles



Costs: 18 node hours* on the JUWELS Booster (32 nodes), in FP64, $4096\times4096\times2048$ grid points, 225×10^6 beam particles, **750 time steps**

Illustration 3:
e^+ acceleration with a hollow-core driver (Jain et al., PRL 2015)

Challenge?
Standard 3D simulation



Costs: 1 hour on a laptop (NVIDIA RTX2070) in FP32, $1024\times1024\times1024$ grid points, 10^7 beam particles, **300 time steps**

More information on our poster tonight:
HiPACE++: GPU-accelerated modeling of plasma wakefield accelerators
(now presented by Maxence Thévenet)

Summary

- QSA PIC community is on the right path
- Many advances in different directions: reduced geometries, mesh refinement, utilizing modern HPC

Significant effort required to fulfill promised results!

Outlook

Questions by the organizers

1) Future developments needed and planned

Sustained, future-oriented code development needed: ***Open source, automated testing, community standards (openPMD), portability***
Focus on achieving roadmaps: ***mesh refinement***

2) Do the planned activities address the requirements from funded projects (AWAKE, EuPRAXIA, …) and from various roadmaps for plasma accelerators? Are there urgent holes?

3) Does simulations and theory require its own roadmap or is work adequately driven/supported through funded projects and through overall plasma accelerator roadmaps?

Outlook

Questions by the organizers

1) Future developments needed and planned

Sustained, future-oriented code development needed: ***Open source, automated testing, community standards (openPMD), portability***
Focus on achieving roadmaps: ***mesh refinement***

2) Do the planned activities address the requirements from funded projects (AWAKE, EuPRAXIA, …) and from various roadmaps for plasma accelerators? Are there urgent holes?

Yes, simulation activities are designed to meet these requirements!

3) Does simulations and theory require its own roadmap or is work adequately driven/supported through funded projects and through overall plasma accelerator roadmaps?

Outlook

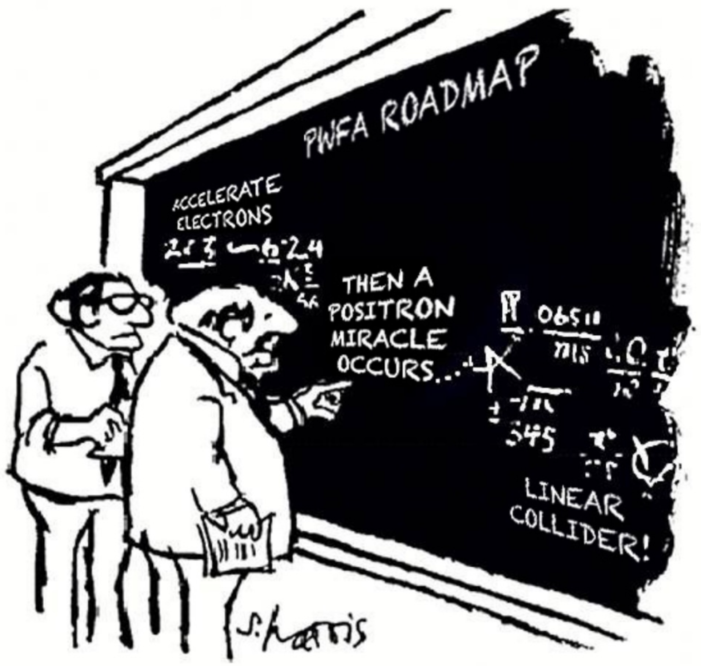
Questions by the organizers

1) Future developments needed anc

Sustained, future-oriented code dev ***ed testing,***
community standards (openPMD)
Focus on achieving roadmaps: ***mes***

2) Do the planned activities address VAKE, EuPRAXIA, …) and from various roadmaps for plasma accele

Yes, simulation activities are design



"I think you should be more explicit here in step two."

from C. Lindstrøm, PhD thesis, 2019

3) Does simulations and theory require its own roadmap or is work adequately driven/supported through funded projects and through overall plasma accelerator roadmaps?

Yes*, otherwise we will soon need another "positron miracle"*

"The needs for simulating (…) nm emittance bunches (…) require further development in this area."