The Implementation and Usage of the Quasi-3D Algorithm in OSIRIS 4.0

Asher Warren Davidson,

Thamine Dalichaouch

OSIRIS Users and Developers Workshop

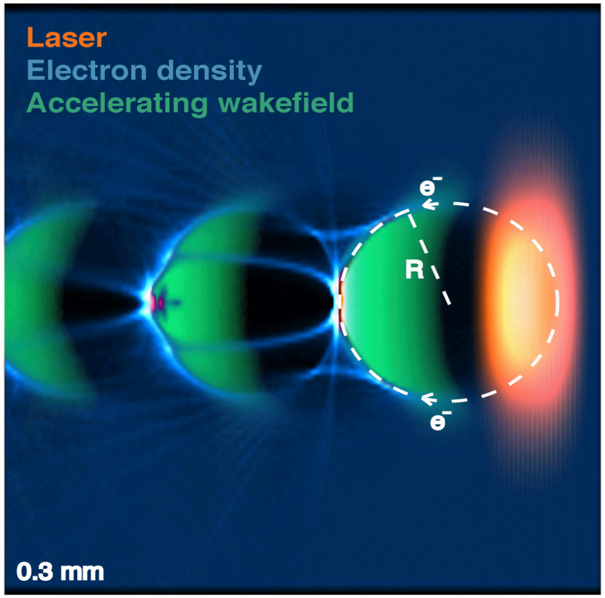
Physics & Astronomy Department

University of California, Los Angeles, CA 90095

September 18, 2017

Initial Motivation: LWFAs

The Laser Wakefield Accelerator



(Courtesy of Lu et al.)

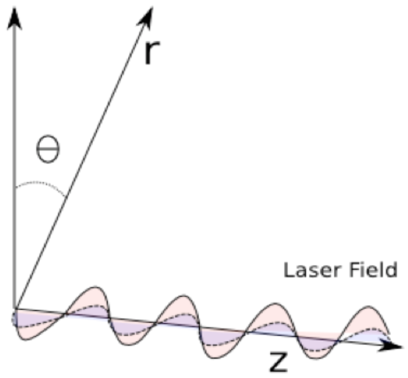
- Pushing LWFA simulations to higher particle energies become computationally expensive quickly

estimated CPU hours (3D)	P (TW)	a_0	ΔE † (GeV) Estimated
100,000	200	4.0	1.58
430,000	324	4.44	2.52
3,200,000	649	4.44	5.28
26,000,000	1298	4.44	10.57
120,000,000	2162	4.44	17.6
340,000,000	3280	4.44	26.7

[†] W. Lu et al., PRSTAB 10 (2007) 061301

2D r-z Code is a Natural Thought

it does not work for LWFAs, despite it's usefulness in other applications (PWFAs)



- The 2D slab geometry also poses geometric problems for simulating wakefield accelerators correctly

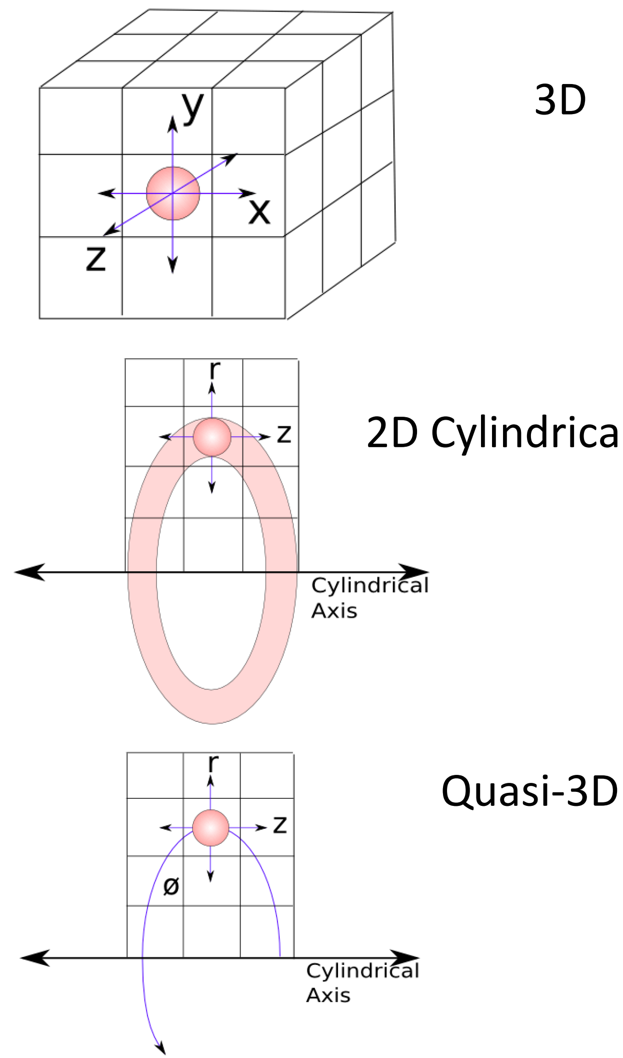
The Quasi-3D Geometry[†,††]

Can Describe a Linearly Polarized Laser

- In a traditional r-z code, the fields are solved on a 2D grid, and the macroparticle positions are in (r, z) space. There is no position in ø (The macroparticles are rings)
- Using truncated azimuthal modal decomposition[†], one results in field solver in which the fields **F**(r, z, ø) are gridded in (r, z) but continuous in ø. Such a model is ideal for problems with low-order azimuthal symmetry

$$\begin{aligned}\mathbf{F}(r,z,\phi) &= \Re\left\{\sum_{m=0}\mathbf{F}^m(r,z)e^{im\phi}\right\}\\ &= \mathbf{F}^0(r,z) + \Re\{\mathbf{F}^1\}\cos(\phi) - \Im\{\mathbf{F}^1\}\sin(\phi)\\ &\quad + \Re\{\mathbf{F}^2\}\cos(2\phi) - \Im\{\mathbf{F}^2\}\sin(2\phi)\\ &\quad + \cdots.\end{aligned}$$

- Broadly speaking, you would expect a speedup proportional to the number of cells in the longitudinal direction times two ($2n_r$) - **Hundreds of times of speedup**

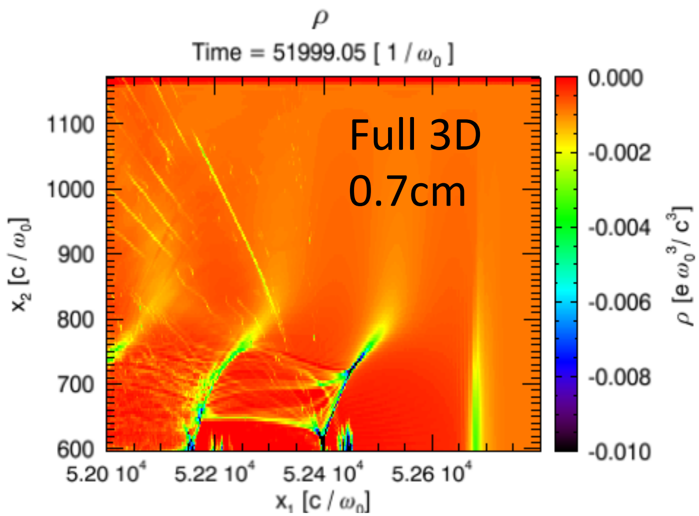


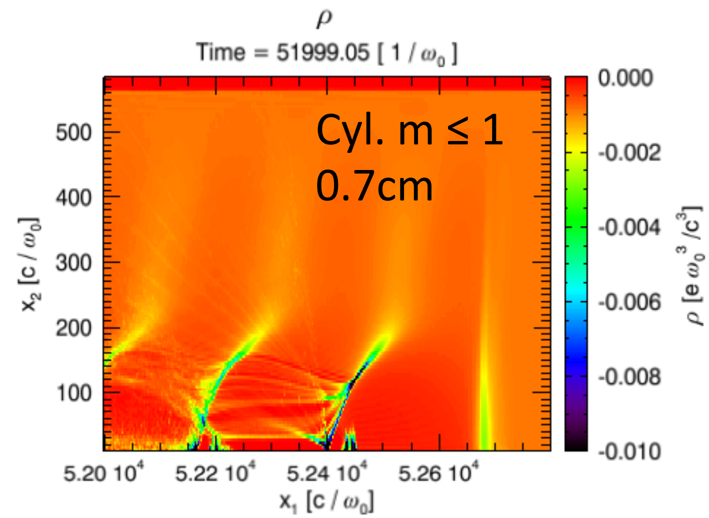
[†] F. Lifschitz et al., J. Comp. Phys. 228 (2009) 1803
[††] A. Davidson, et al., J. Comp. Phys. 281 (2015) 1063

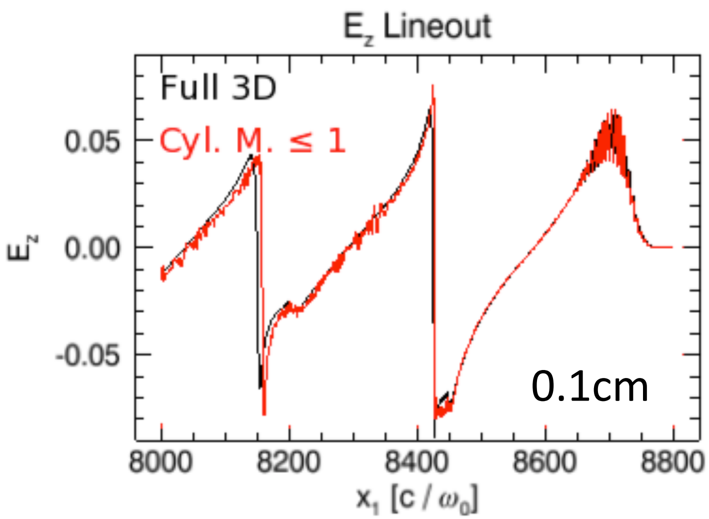
Verification of Quasi-3D Algorithm

Verification of Quasi-3D Algorithm

Comparison of Wake and Beam







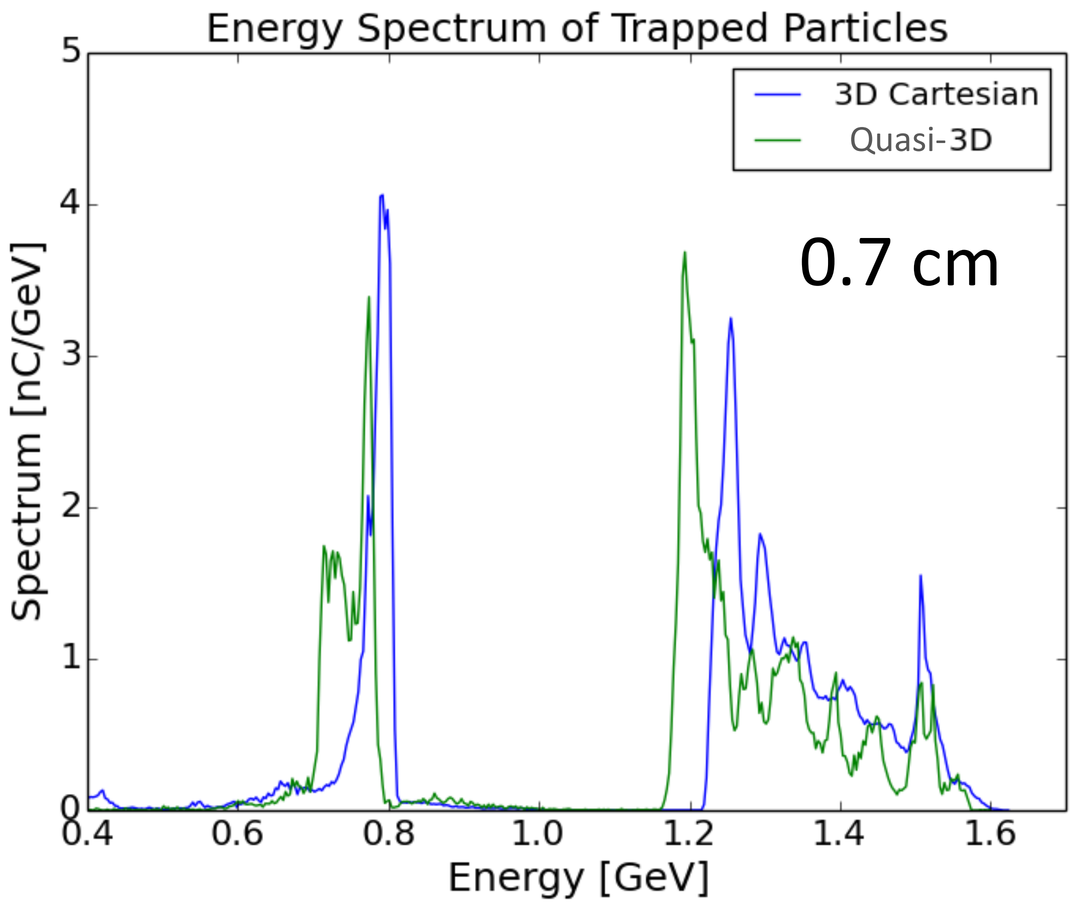
(200TW laser, n_p = 1.5x1018 ; from Lu et al.[†])

Geometry	Charge [pC] (0.7cm)	ex [π mm mrad]	ey [π mm mrad]	Max Energy [GeV]
Full 3D Cartesian	340	27	30	1.57
Cyl. Mode ≤ 1	328	43	43	1.55

[†] W. Lu et al., PRSTAB 10 (2007) 061301

Verification of Quasi-3D Algorithm

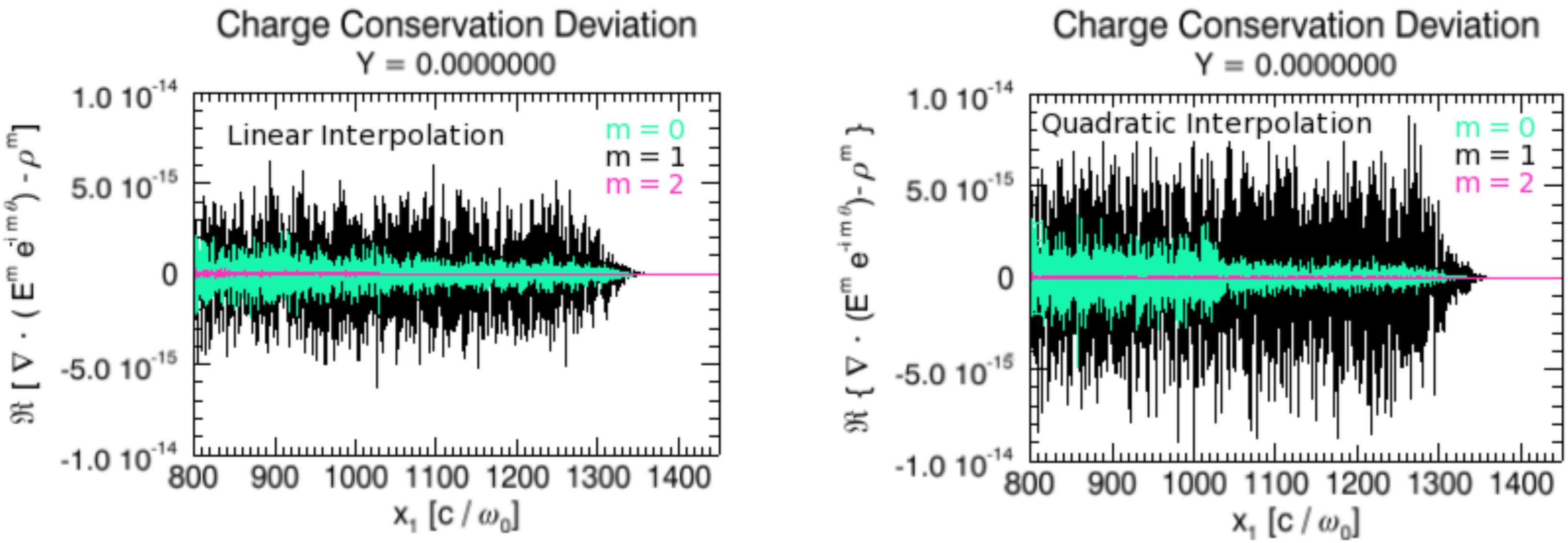
Comparison of Energy Spectra



- The total trapped charge, as well as the basic features of the mono-energetic trapped particle beams are maintained.

Verification of Quasi-3D Algorithm

Charge is Rigorously Conserved in OSIRIS



- Our Implementation ensures rigorous charge conservation†
- Charge conservation has been tested for both the real and imaginary parts of each mode, by plotting $\nabla \cdot \mathbf{E} - \rho$
- The code conserved charge to the numerical round off of the double precision floating point number

[†] A. Davidson, et al., J. Comp. Phys. 281 (2015) 1063

Using the Quasi-3D Code

How to Run in OSIRIS 4.0

Most Important Parameters

Indicate the intention to use quasi-3D geometry and associated algorithms:

```
simulation
{ algorithm = "quasi-3D", }
```

Indicate the number of modes at which to truncate expansion in ø:

```
grid
{ coordinates = "cylindrical",
  n_cyl_modes = 1, }
```

(Make sure to set coordinates as "cylindrical")

Indicate the number of particles to resolve in ø:

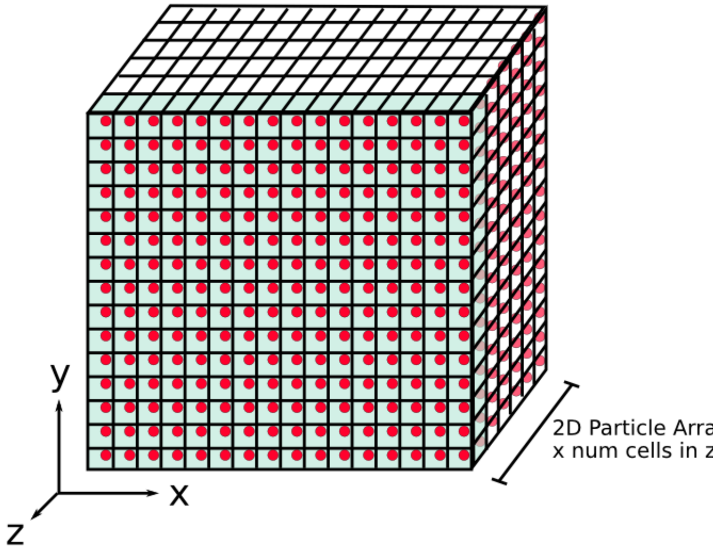
```
species
{ ...
num_par_theta = 8,
  ... }
```

Most other parameters may be used as though it is the traditional 2D Cylindrical algorithm in OSIRIS (zpulse, udist, profile, etc.)

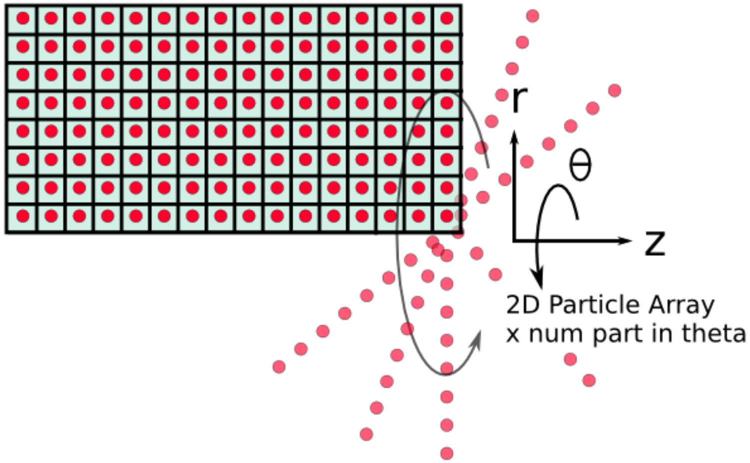
Particle Initialization

In the quasi-3D geometry, particles are initialized at equal intervals in ø

(Cartesian 3D)



(Quasi-3D; num_par_theta = 8)



Remember, although fields are decomposed into 2D grids, simulation particles exist in a continuous 3D Cartesian space

How to Run in OSIRIS 4.0

Diagnostic Parameters

Traditional diagnostic requests would output data from the cylindrically symmetric (m = 0) part of the EM field:

```
diag_emf { …
reports = "e1","e2","e3",”b1","b2","b3",
 … }
```

To output EM data for all the modes, request the following:

```
diag_emf { …
reports = "e1_cyl_m","e2_cyl_m","e3_cyl_m","b1_cyl_m","b2_cyl_m","b3_cyl_m",
… }
```

The real and imaginary part of each mode will be dumped in separate folders, inside **‘FLD/MODE-1-RE’**, **‘FLD/MODE-1-RE’**, **‘FLD/MODE-1-IM’**, etc.

More Diagnostics

The current diagnostic input follows a similar scheme:

```
diag_current { …
reports = “j1_cyl_m”, “j2_cyl_m”, “j3_cyl_m”,
 … }
```

And so does the charge diagnostic, even though simulation particles are not stored as field values decomposed as azimuthal modes:

```
diag_species { …
reports = “charge_cyl_m”,
… }
```

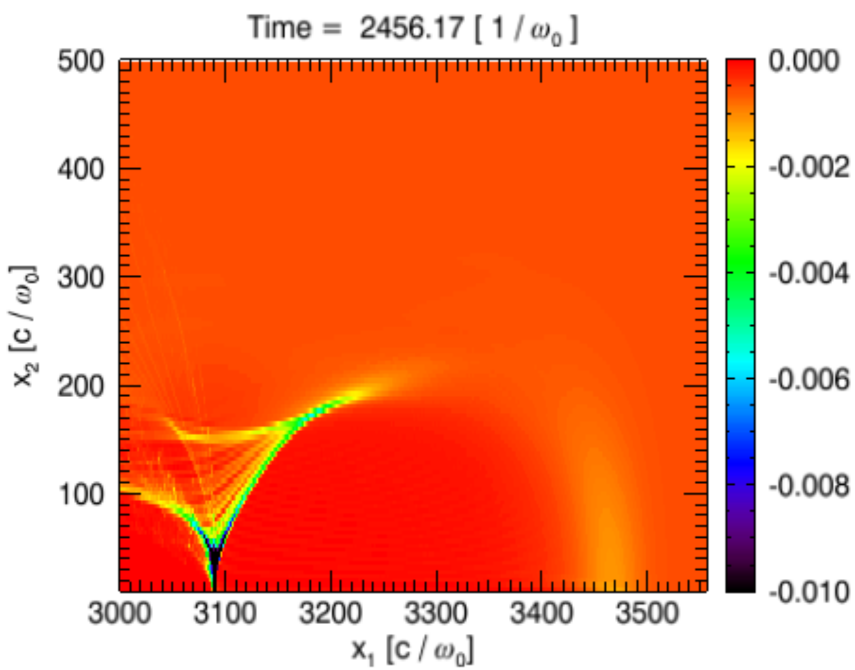
Full 3D information can be constructed by summing existing modes:

$$
\begin{aligned}
\mathbf{F}(r,z,\phi) &= \Re\left\{\sum_{m=0} \mathbf{F}^m(r,z)e^{im\phi}\right\} \\
&= \mathbf{F}^0(r,z) + \Re\{\mathbf{F}^1\}\cos(\phi) - \Im\{\mathbf{F}^1\}\sin(\phi) \\
&\quad + \Re\{\mathbf{F}^2\}\cos(2\phi) - \Im\{\mathbf{F}^2\}\sin(2\phi) \\
&\quad + \cdots.
\end{aligned}
$$

More Diagnostics

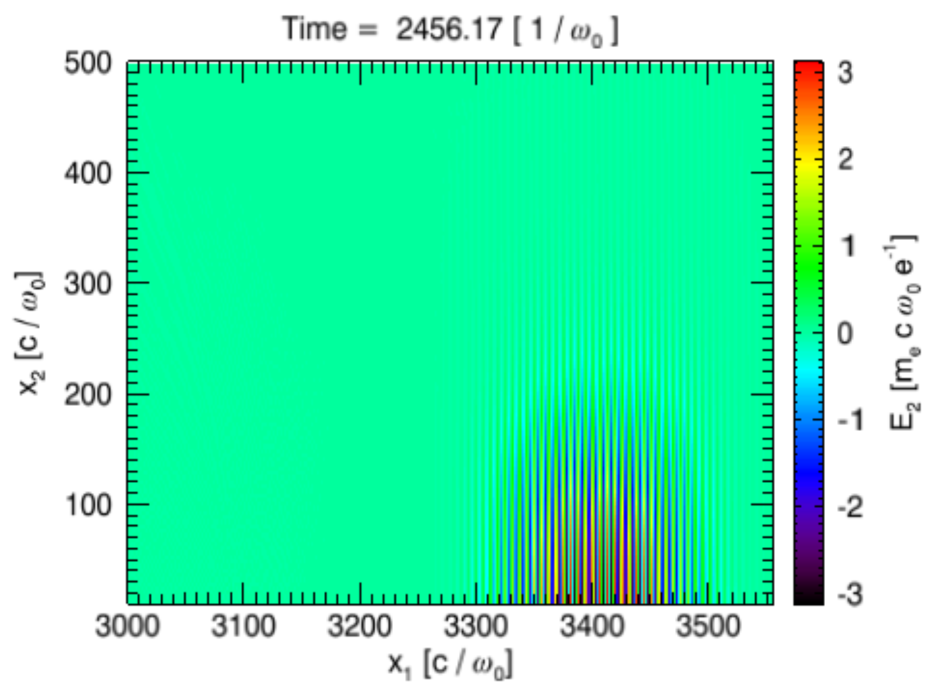
Output may sometimes present clear meaning without summing modes

Re{ Density (m=0) }



Wake Structure

Re { E2 (m = 1) }

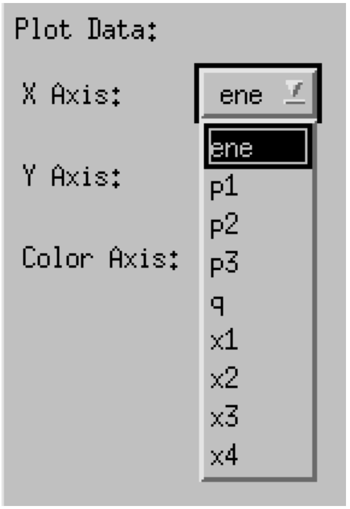


Laser

How to Run in OSIRIS 4.0

RAW Particle Diagnostics

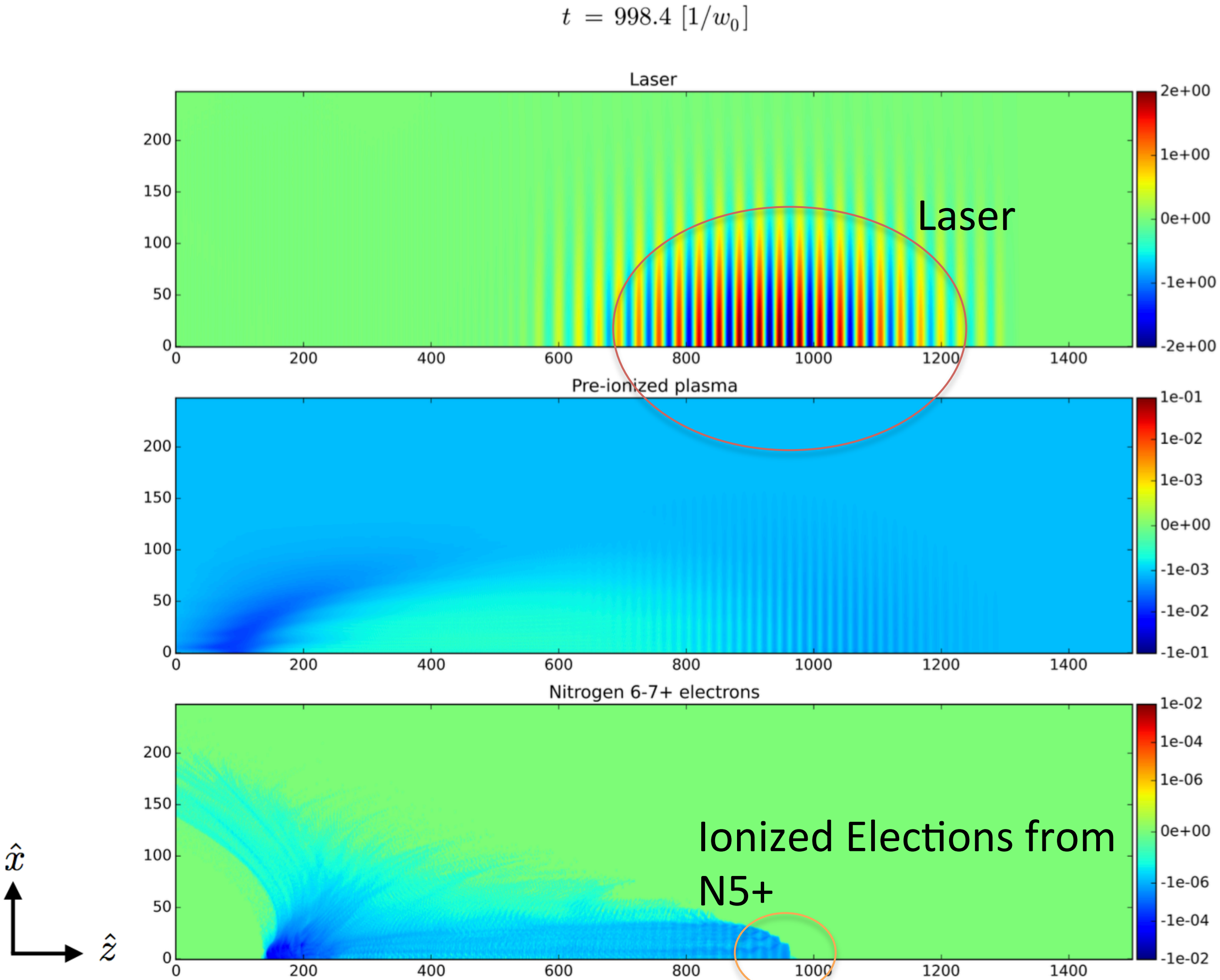
- The input for RAW particle diagnostics are identical to typical 2D cylindrical simulations
- However, you will observe not **2** but **4 coordinates** in the output
- Due to the internal management of coordinate values in OSIRIS, the particle coordinates are stored in the order of (x1, x2, x3, x4) = (z, r, x, y), respectively
- The momentum values correspond to (p1, p2, p3) = (p_z, p_x, p_y)
- Aside from the stored ‘r’ values, he RAW particle data is in Cartesian Coordinates



(GUI output menu from vis2D)

Future Implementation

of Ionization-Injection into Quasi-3D



(Courtesy of Thamine Dalichaouch)

- Gas ionization has been successfully implemented in OSIRIS 3.0
- This will be implemented into 4.0 in the future

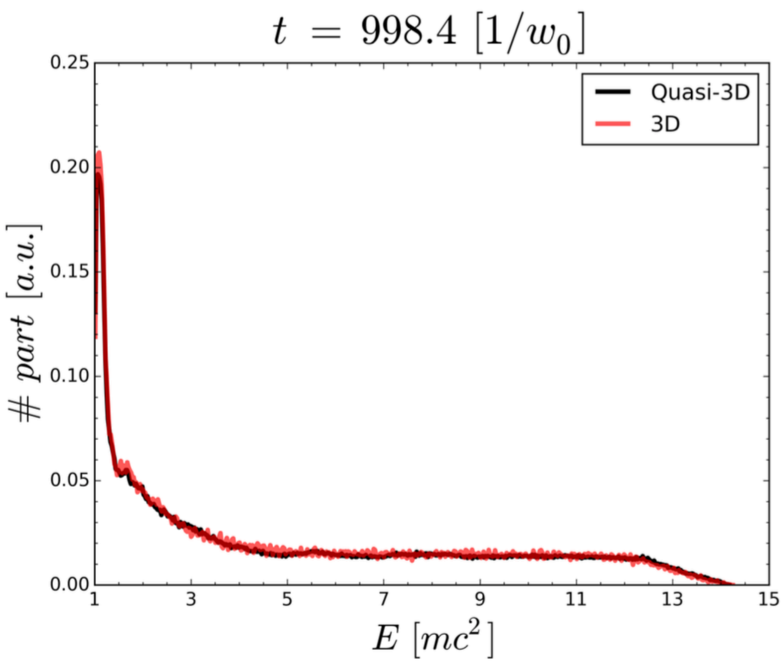
- **Parameters**
  - $a_0 = 2.0$
  - $w_0 = 14\ \mu m$
  - FWHM = 26 fs

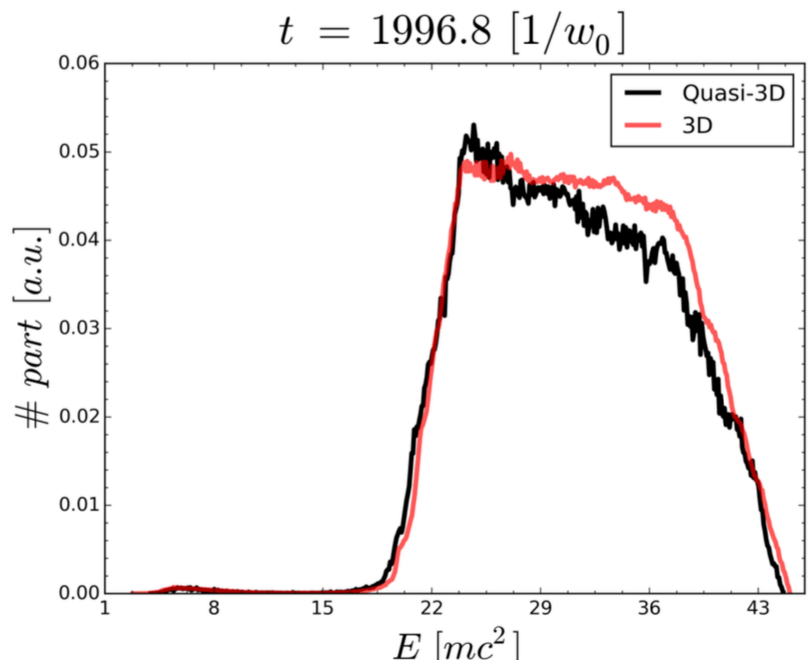
  - $n_p = 2x10^{18}\ cm^{-3}$
  - plasma length : 4cm

Future Implementation

of Ionization-Injection into Quasi-3D

- Ionization-injection may be used to inject mono-energetic beams into a LWFA in a controlled manner
- Here we may compare the energy spectrums of injected particles at two moments in time for a full 3D Cartesian simulation and a Quasi-3D simulation





(Courtesy of Thamine Dalichaouch)

Summary

- The quasi-3D geometry has been implemented into the OSIRIS 4.0 framework
- Verification of the effectiveness of the code and comparison with full 3D results have been presented
- The basic input deck parameters and user guidelines have been presented
- Ask me any questions about the quasi-3D code and its currently supported features
- Thamine can answer any questions regarding the implementation of neutral gas ionization in the quasi-3D geometry

The End