Petascale Plasma Physics Simulation Using PIC Codes (PI: W. B. Mori, UCLA)

INSIGHT
Sustainable ecosystems and society

nature

THE INTERNATIONAL WEEKLY JOURNAL OF SCIENCE

FULL SPEED AHEAD
Plasma wakefield machines — the particle accelerators of the future? PAGES 40 & 92

CENTRAL EUROPE
LIFE AFTER THE WALL
Science 25 years after the collapse of communism
PAGE 22

ENVIRONMENTAL SCIENCE
CASH, CONFLICT, CONSERVATION
To protect the planet, stop the infighting and fudging
PAGES 27, 28 & 32

INTERACTIVE NOTEBOOKS
SHARE AND SHARE ALIKE
IPython allows on-the-run analysis
PAGE 151

NATURE.COM/NATURE
6 November 2014
$10.00US $12.99CAN

Frank S. Tsung, Viktor Decyk, Weiming An, Ben Winjum

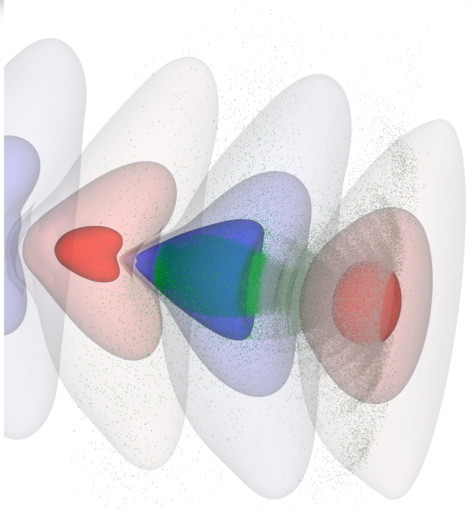
UCLA Plasma Simulation Group

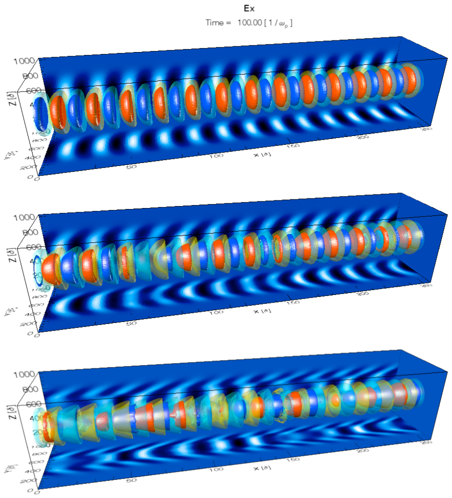
UCLA

Summary and Outline

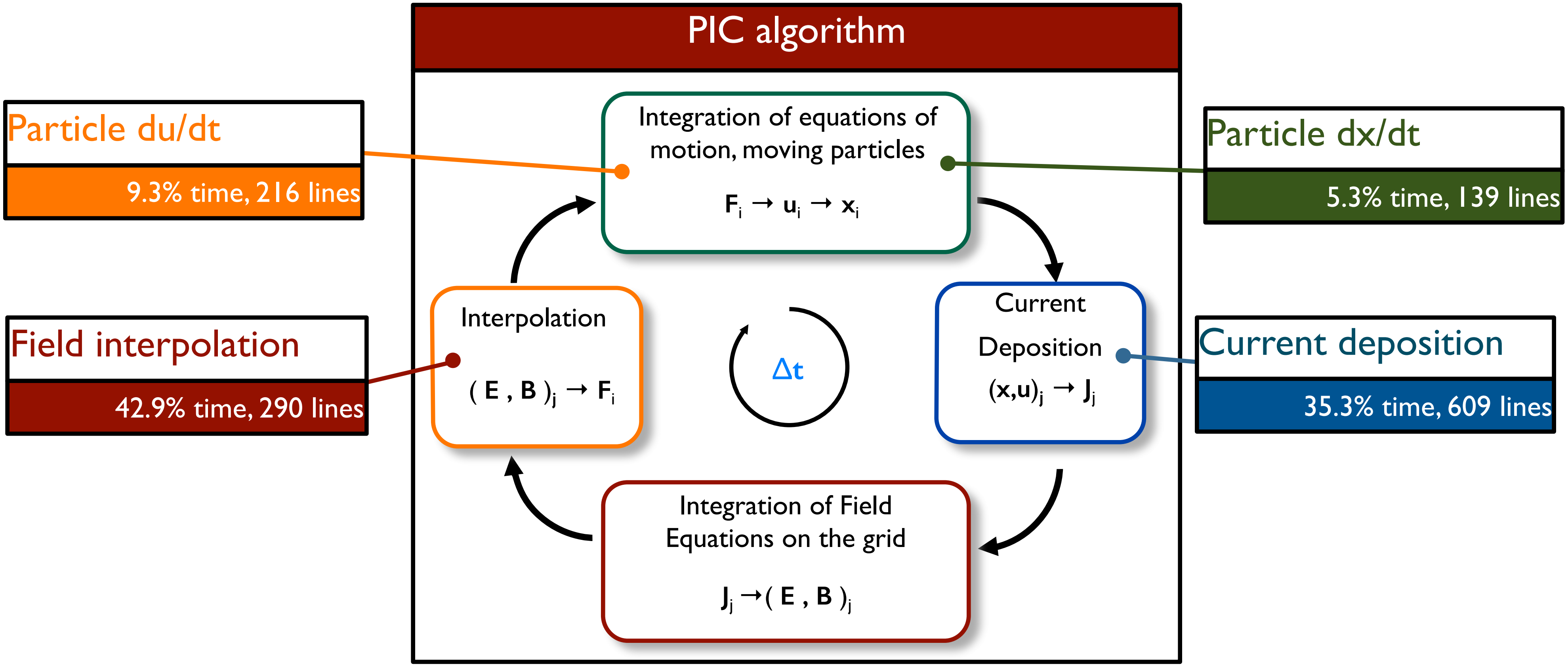
OUTLINE/SUMMARY

- Overview of the project
  - Particle-in-cell codes
  - PIC codes available @ PICKSC
- Application of OSIRIS to plasma based accelerators:
  - QuickPIC simulations of SLAC experiments
- Applications of OSIRIS to LPI's Relevant to IFE
  - SRS in indirect drive IFE targets (such as NIF).
  - Estimates of large scale LPI simulations (& the need for exascale supercomputers)
- Development works for Blue Waters and beyond (including GPU's and other emerging architectures) + the PICKSC Center @ UCLA





Profile of OSIRIS + Introduction to PIC



Particle du/dt

9.3% time, 216 lines

Field interpolation

42.9% time, 290 lines

PIC algorithm

Integration of equations of motion, moving particles

$F_i \rightarrow u_i \rightarrow x_i$

Interpolation

$(E , B)_j \rightarrow F_i$

Δt

Current Deposition

$(x,u)_j \rightarrow J_j$

Integration of Field Equations on the grid

$J_j \rightarrow (E , B)_j$

Particle dx/dt

5.3% time, 139 lines

Current deposition

35.3% time, 609 lines

- The particle-in-cell method treats plasma as a collection of computer particles. The interactions does not scale as N^2 due to the fact the particle quantities are deposited on a grids and the interactions are calculated on the grids only. Because (# of particles) >> (# of grids), the timing is dominated by the particle calculations (orbit calculation + current & charge deposition)
- The code spends over 90 % of execution time in only 4 routines
- These routines correspond to less than 2 % of the code, optimization and porting is fairly straightforward, although not always trivial.

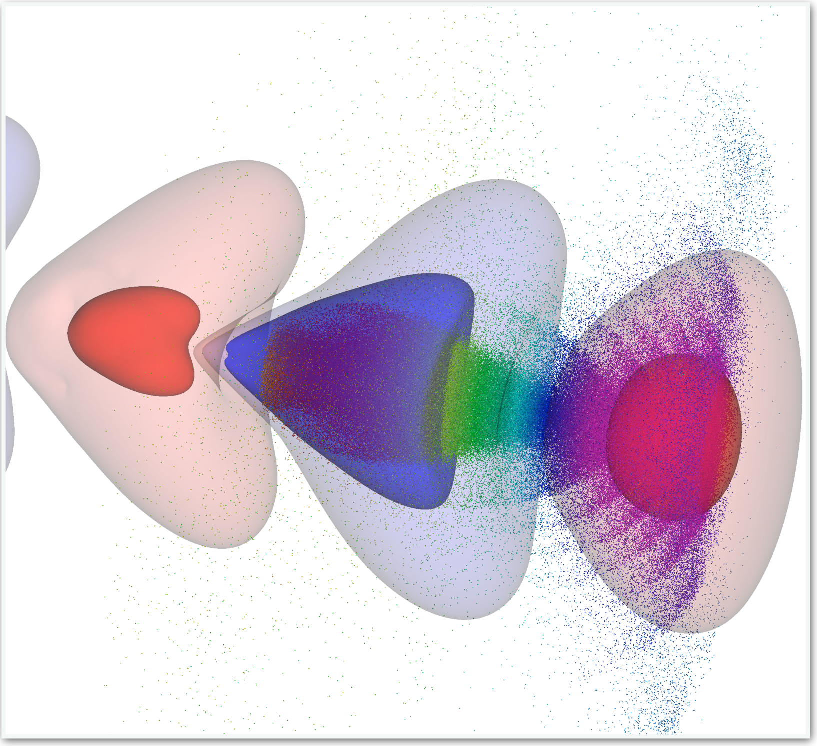
UCLA

osiris

osiris framework

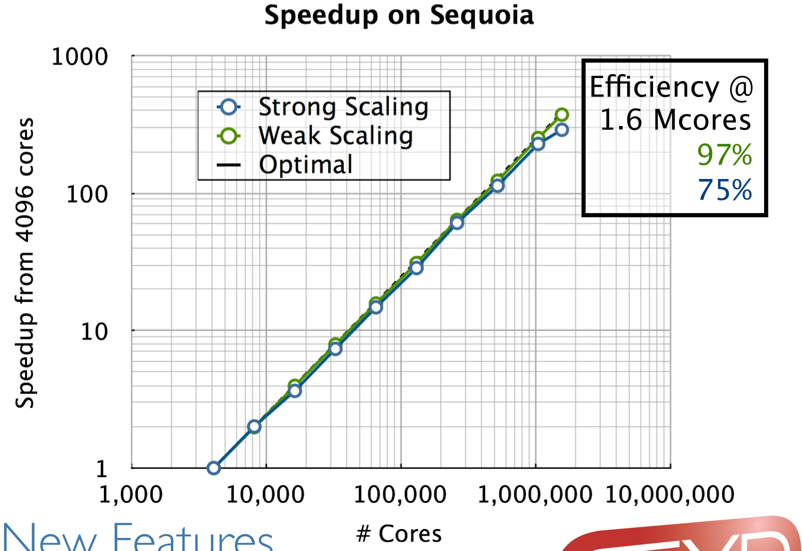
- Massivelly Parallel, Fully Relativistic Particle-in-Cell (PIC) Code
- Visualization and Data Analysis Infrastructure
- Developed by the osiris.consortium

⇒ UCLA + IST



Ricardo Fonseca: ricardo.fonseca@ist.utl.pt
Frank Tsung: tsung@physics.ucla.edu

http://cfp.ist.utl.pt/golp/epp/
http://exodus.physics.ucla.edu/



New Features

- Bessel Beams
- Binary Collision Module (to study plasmas which behave more like fluids)
- **Energy Conserving Algorithm**
- **Multi-dimensional Dynamic Load Balancing**
- **OpenMP/MPI hybrid parallelism**
- CUDA branch
- **Higher order splines**
- Parallel I/O (HDF5)
- Gridless cylindrical mode
- sustained > 2.2 PFlops on Blue Waters & good scaling on > 1.5 million cores (Sequoia supercomputer @ LLNL)

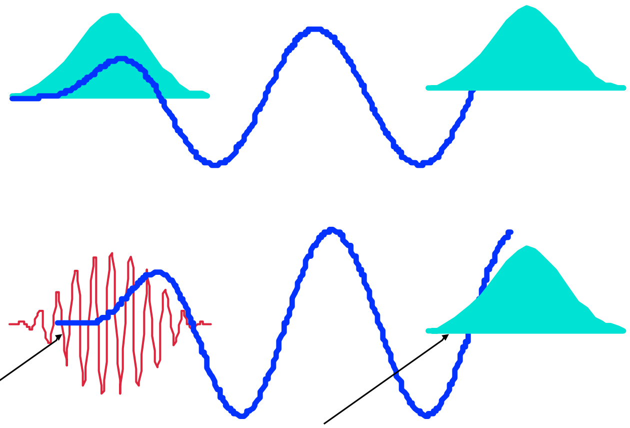
Livingston Curve for Accelerators --- Why plasmas?

Plasma Wake Field Accelerator(PWFA)

A high energy electron bunch

Laser Wake Field Accelerator(LWFA, SMLWFA)

A single short-pulse of photons

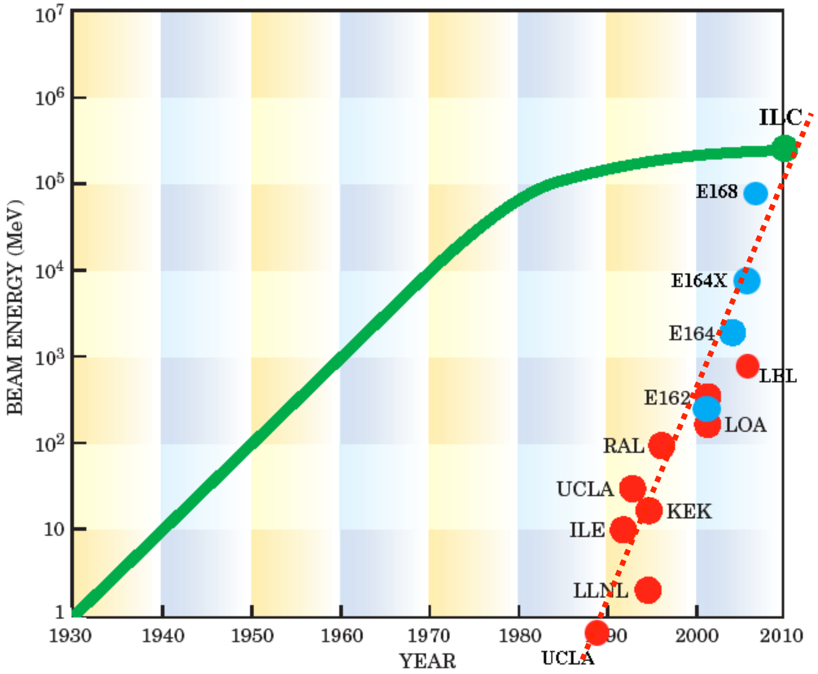


Drive beam

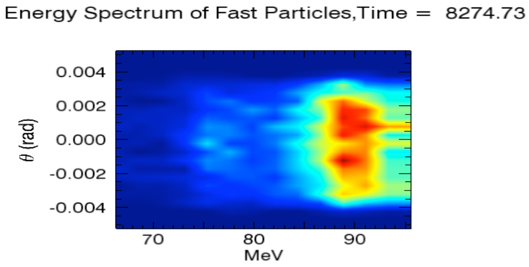
Trailing beam

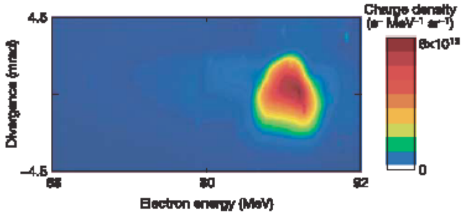
The Livingston curve traces the history of electron accelerators from Lawrence's cyclotron to present day technology.

When energies from plasma based accelerators are plotted in the same curve, it shows the exciting trend that within a few years it is will surpass conventional accelerators in terms of energy.

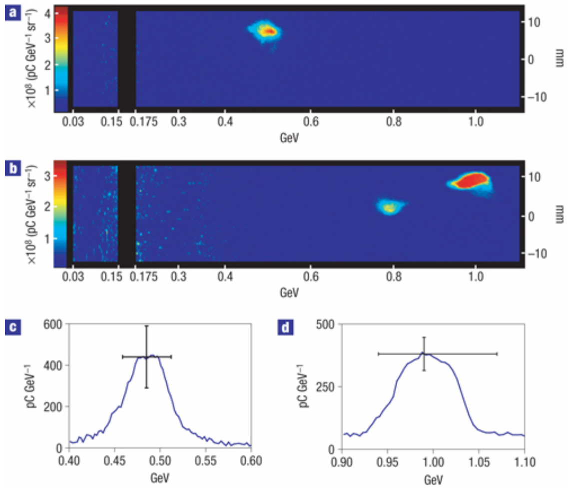


Recent Highlights (in *Nature* journals*)* in Plasma Based Acceleration (< Last 10 years) -- Simulations play a big role in all of these discoveries!!!



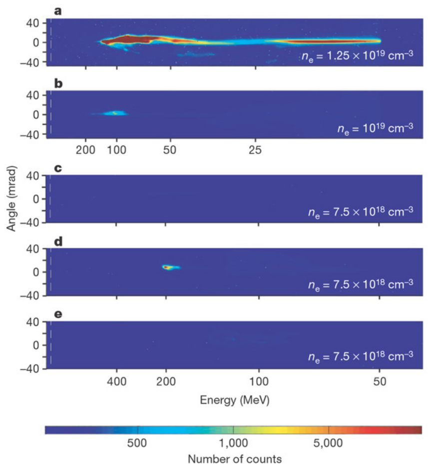


"Dream Beam" (Nature, 2004) -- 3 groups observed monoenergetic bunches using short (< 100fs) pulse lasers -- 3D simulations produced qnantitative agreements!!

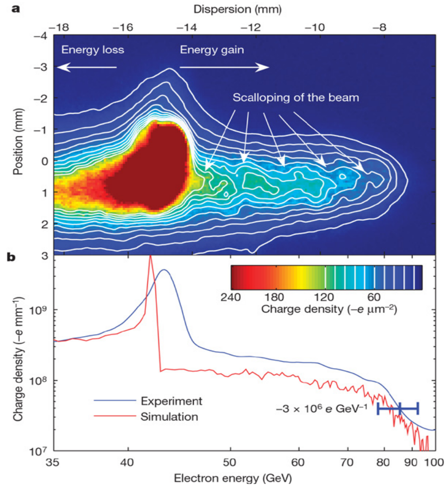


GeV LWFA in cm scale plasma

2014 "Full Speed Ahead" Cover on *Nature*



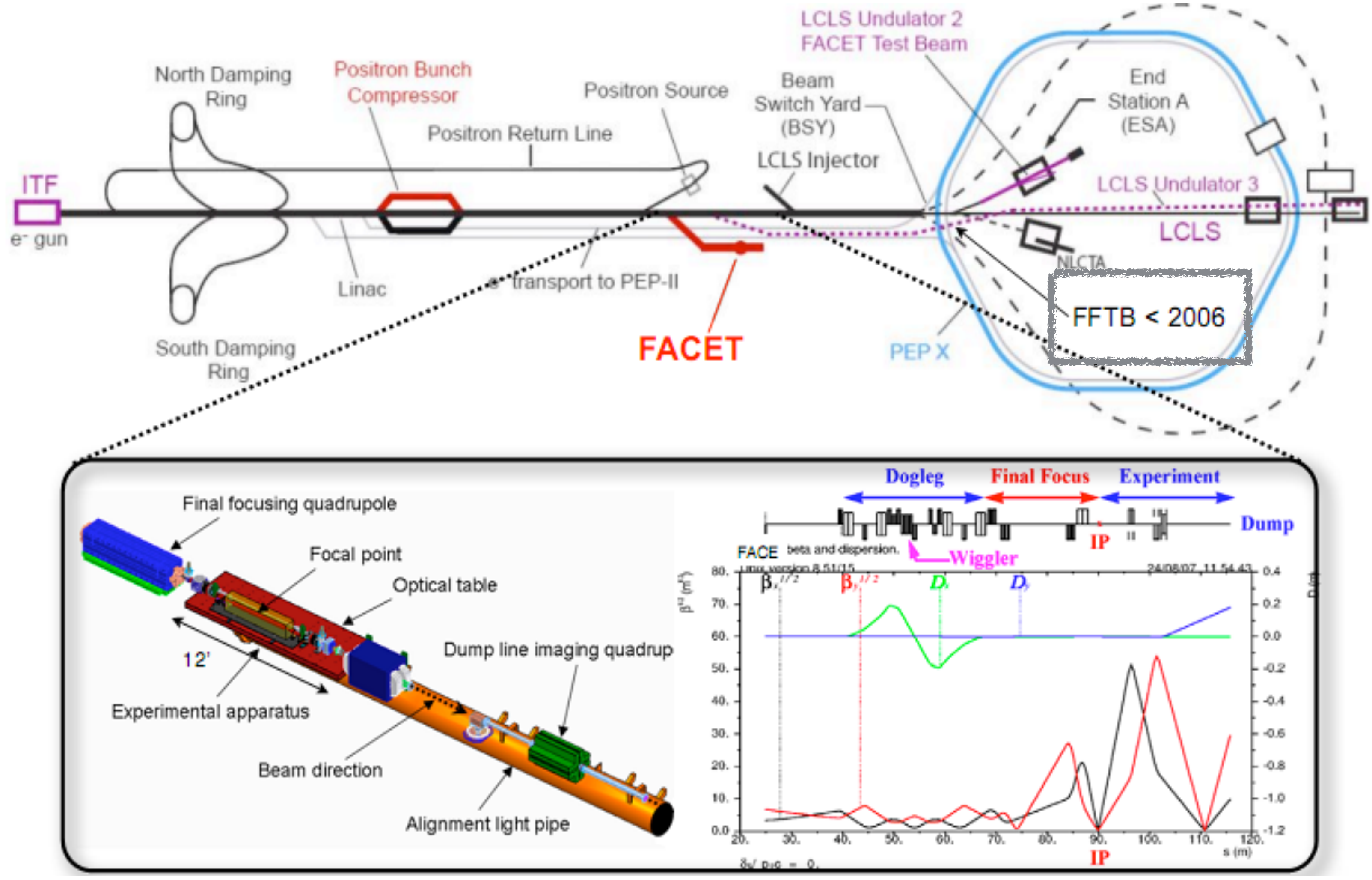
Controlled electron injection



42 GeV in less than one meter!
(i.e., 0-42 GeV in 3km, 42-85 GeV in 1m)
Simulations also identified ionization induced erosion as the limiting mechanism for energy gain

FACET— Plasma based accelerator experiments @ LCLS (linac coherent light source)

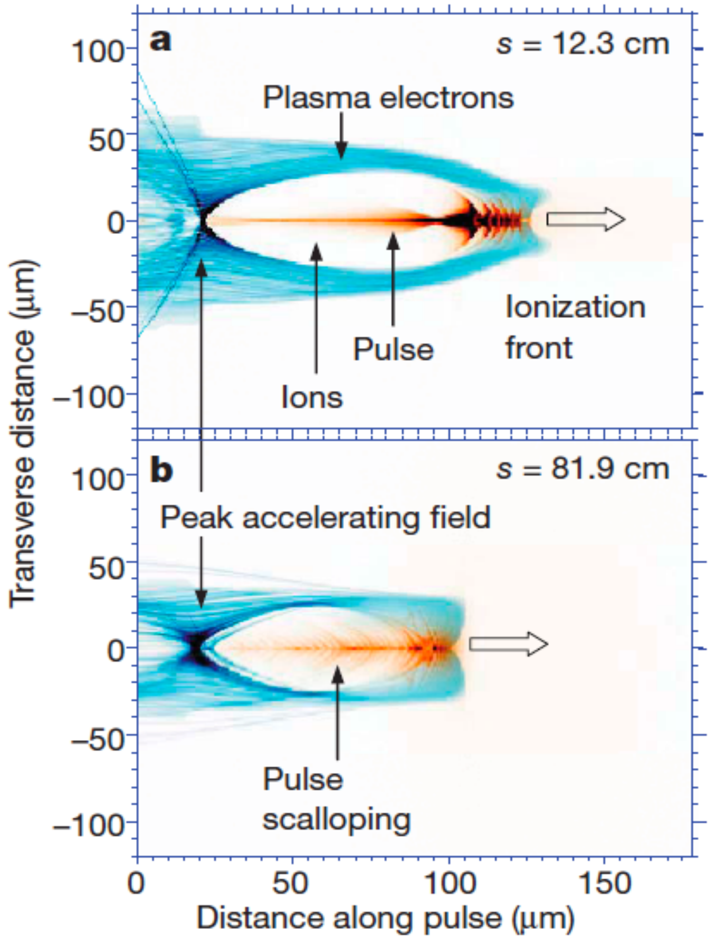
Facility for Advanced Accelerator Experimental Tests

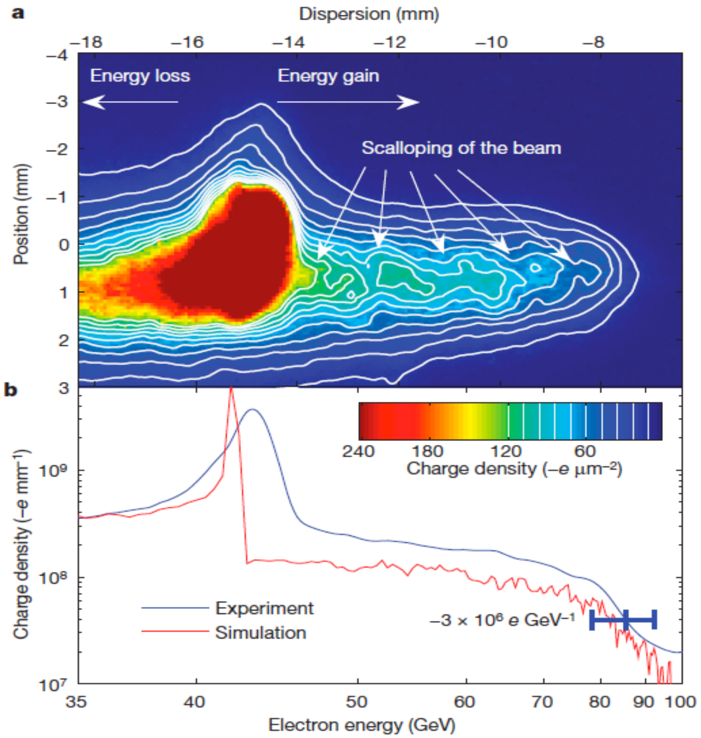


FACET is a new facility to provide high-energy, high peak current e^- & e^+ beams for PWFA experiments at SLAC.

Single Bunch e$^-$ Driven PWFA (Blumenfeld *et al*, *Nature* (2007)).

PWFA: Plasma Wake Field Acceleration



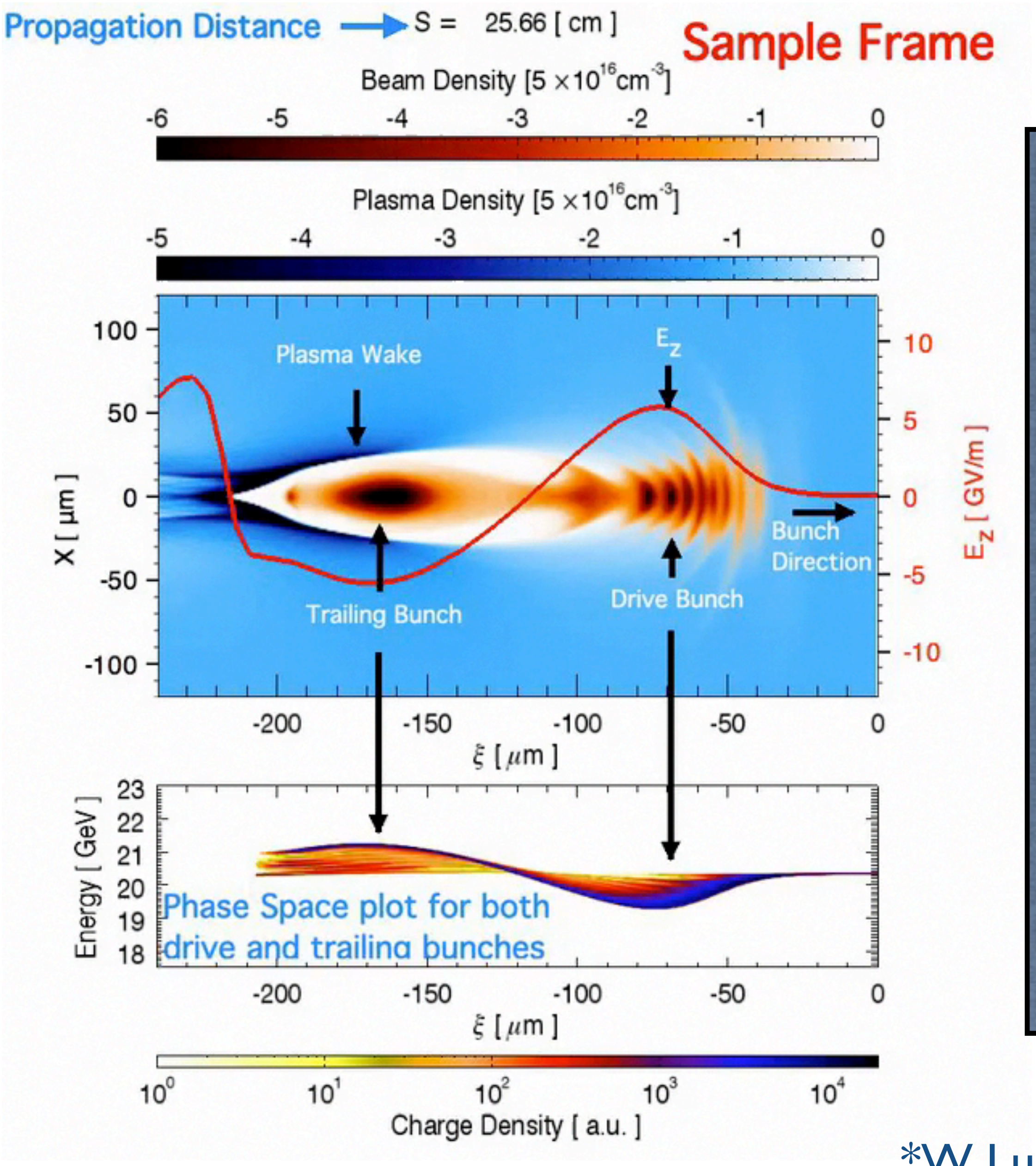


In the 2007 experiment, done @ the FFTB facility @ SLAC, used a single bunch which serves both as the driving bunch and the witness bunch. In the experiment, the initial energy of the electron beam was ~42 GeV (after 3kM) and the **peak** energy is doubled after < 1 meter of plasma. The above plots show good agreements between experimental results and experiments, and the simulation also shed lights on the limitation of the 2007 experiment.

However, the experiment demonstrated acceleration, but the electrons created have a very large energy spread and cannot be used to study high energy physics. The goal of the 2014 experiment is to change this and to demonstrate that an accelerator can be built using plasma based techniques.

* Ian Blumenfeld, et. al., Nature 445, 741 (2007)

Two-Bunch e$^-$ PWFA



In the 2014 experiment, the electron beam is split into two, a driving beam and a trailing beam. The trailing beam has enough charge such that it can modify the wake, and cause the wake to flatten. The flat wake causes all of the electrons to be accelerated at the same rate, leading to a high quality beam with a narrow energy spread (< 1% energy spread). The initial energy of the beam is ~20GeV (1.5kM) and it gains 2GeV after only 36cm of plasma.

A typical QuickPIC simulation of two-bunch PWFA will use 4096 processors and cost around 16000 cpu-hours.

*W. Lu, PRL(2006) and M.Tzoufras, PRL (2008)

UCLA

Two-Bunch e^- Driven PWFA

INSIGHT
Sustainable ecosystems and society

nature

THE INTERNATIONAL WEEKLY JOURNAL OF SCIENCE

FULL SPEED AHEAD

Plasma wakefield machines — the particle accelerators of the future? PAGES 40 & 92

CENTRAL EUROPE
LIFE AFTER THE WALL
Science 25 years after the collapse of communism
PAGE 22

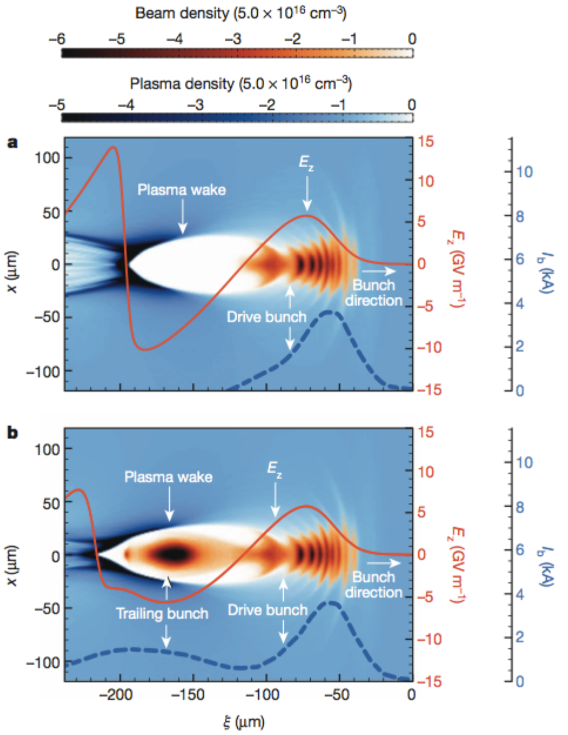
ENVIRONMENTAL SCIENCE
CASH, CONFLICT, CONSERVATION
To protect the planet, stop the infighting and fudging
PAGES 27, 28 & 32

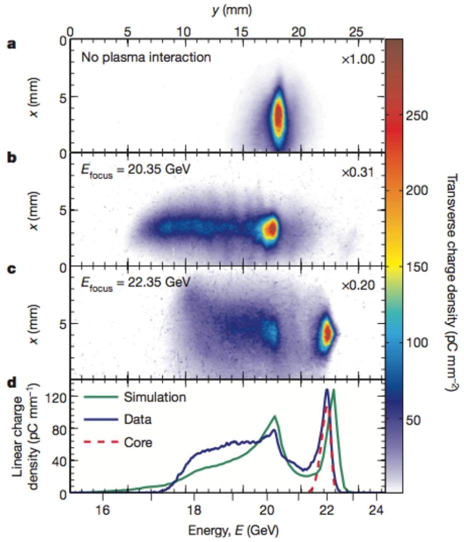
INTERACTIVE NOTEBOOKS
SHARE AND SHARE ALIKE
IPython allows on-the-run analysis
PAGE 151

NATURE.COM/NATURE
6 November 2014

$10.00US $12.99CAN

*M. Litos et. al, 515, 92 Nature (2014)





And here are some figures taken from the *Nature* article, and the image which was chosen for the cover.

As I reported earlier (and the energy spectrum of the electrons are shown on the right), in the 2014 experiment, the particles started with at 20 GeV, and after 36cm of plasmas, some of the particles lost energy but the trailing bunch gained 2GeV with a very small (< 1%) energy spread, and the quantitative agreements between our simulation results and experiments are quite good.

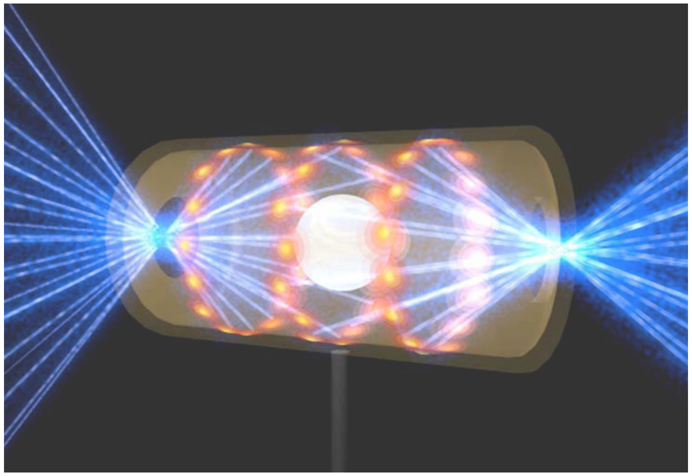
In 2015 the experiments focus on the acceleration of positrons and I hope to talk to you about these results next year.

Laser Plasma Interactions in IFE

- IFE (inertial fusion energy) uses lasers to compress fusion pellets to fusion conditions. The goal of these experiments is to extract more fusion energy from the fuel than the input energy of the laser. In this case, the excitation of plasma waves via LPI (laser plasma interactions) is detrimental to the experiment in 2 ways.
  - Laser light can be scattered backward toward the source and cannot reach the target
  - LPI produces hot electrons which heats the target, making it harder to compress.

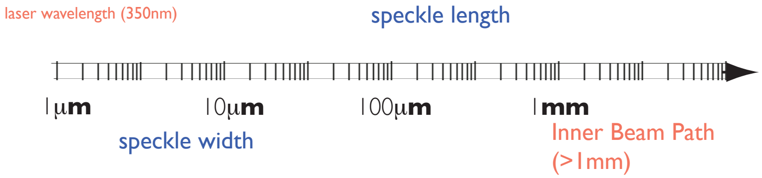
- The LPI problem is very challenging because of the various scales involved
  - The spatial scale spans from sub-micron (which is the laser wavelength) to mille-meters (which is the length of the plasma).
  - The temporal scale spans from a femto-second(which is the laser period) to nano-seconds (which is the duration of the fusion pulse)

Laser Plasma Interactions

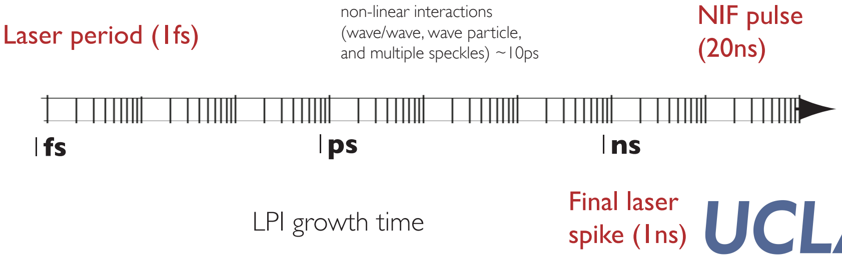


NIF
National Ignition Facility

Lengthscales



Timescales

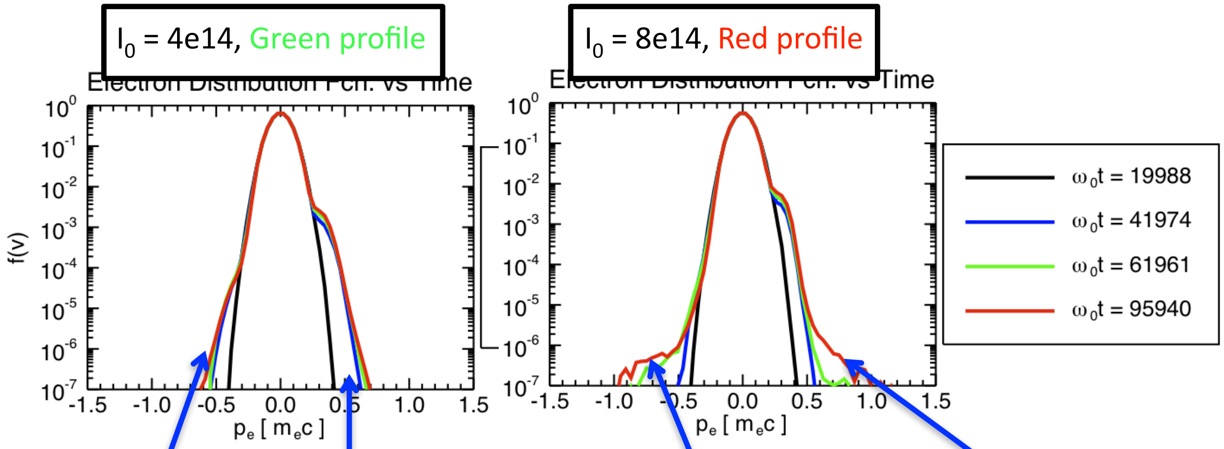


Currently most kinetic simulations of LPI's for NIF are done in 1D

- 1D simulations are quick and allow for methodical parameter scans and comparisons with linear theory. Currently, experimentalists @ NIF can re-construct plasma conditions (such as density and temperature) using a hydro code, and LPI information can be calculated using these plasma conditions.
  - Hydro conditions → NIF uses 1D fluid postprocessing tools such as SLIP/NEWLIP:

    Predict the frequency and reflectivity of the most unstable LPI
  - Hydro conditions → 1D OSIRIS simulations:

    Similar capabilities + detailed information about energy partition, backscattered light, and energetic electrons (which can also be compared against experiments). We can also identify the various processes that create these energetic electrons. In the plot below (where we show f(v)), we can identify the physical processes that lead to the various kinks in the dist. func..



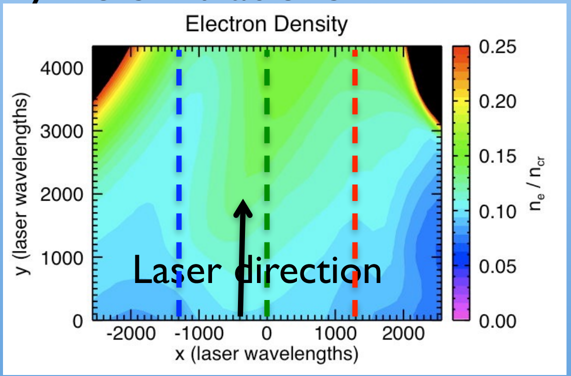
Due to backscatter

Due to LDI of backscatter

Due to LDI of rescatter

Due to rescatter of initial SRS

I_{laser} = 2 – 8 x 10^{14} W/cm^2
λ_{laser} = 351nm,
T_e = 2.75 keV,
T_i = 1 keV, Z=1,
t_{max} up to 20 ps
Length = 1.5 mm
Density profiles from NIF hydro simulations

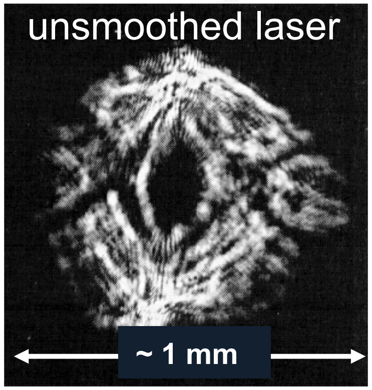


14 million particles
~100 CPU hours per run
~1 hr on modest size supercomputer

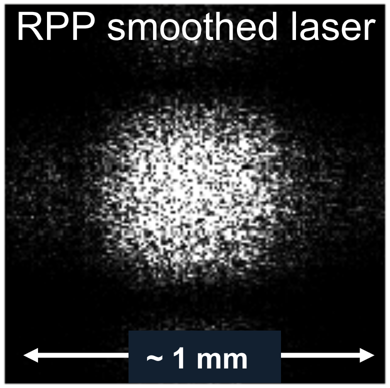
We have simulated stimulated Raman scattering in multi-speckle scenarios (in 2D)

- Although the SRS problem is 1D (i.e., the instability grows along the direction of laser propagation). The SRS problem in IFE is not strictly 1D -- each "beam" (right) is made up of 4 lasers, called a NIF "quad," and each laser is not a plane wave but contains "speckles," each one a few microns in diameter. These hotspots are problematic because you can have situations where according to linear theory, the "averaged" laser is LPI unstable only inside these "hotspots". And the LPI's in these hotspots can trigger activities elsewhere. The multi-speckle problem are inherently 2D and even 3D.

- We have been using OSIRIS to look at SRS in multi-speckle scenarios. In our simulations we observed the excitation of SRS in under-threshold speckles via:
  - "seeding" from backscatter light from neighboring speckles
  - "seeding" from plasma wave seeds from a neighboring speckle.
  - "inflation" where hot electrons from a neighboring speckle flatten the distribution function and reduce plasma wave damping.
- The interaction of multiple speckles is a highly complex process and is ideally suited for PIC simulations



Early 1980's



Post 1990's

PIC simulations of 3D LPI's is still a challenge, and requires exa-scale supercomputers, this will require code developments.

	2D multi-speckle along NIF beam path	3D, 2 speckles	3D, multi-speckle along NIF beam path
Speckle scale	50 x 8	2 x 1	10 x 10 x 5
Size (microns)	150 x 1500	18 x 9 x 120	28 x 28 x 900
Grids	9,000 x 134,000	1,000 x 500 x 11,000	1,700 x 1,700 x 80,000
Particles	300 billion	620 billion	22 trillion
Steps	470,000 (15 ps)	180,000 (5 ps)	540,000 (15 ps)
Memory Usage*	7 TB	18 TB	1.6 PB
CPU-Hours	8 million	9 million	1 billion (2 months on the full BW)

Designing New Particle-in-Cell (PIC) Algorithms on GPU’s

On the GPU, we apply a local domain decomposition scheme based on the concept of tiles.

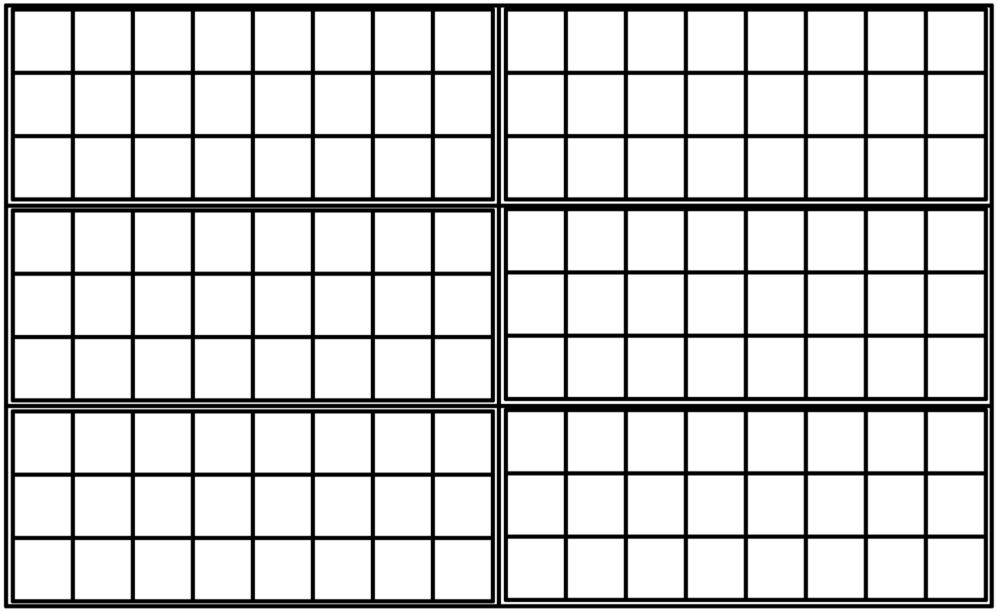
Particles ordered by tiles, varying from 2 x 2 to 16 x 16 grid points
On Fermi M2090:

- On each GPU, the problem is partitioned into many tiles, and the code associate a **thread block** with each tile and particles located in that tile

We created a new data structure for particles, partitioned among threads blocks (i.e., particles are sorted according to its tile id, and there is a local domain decomposition within the GPU), within the tile the grid and the particle data are aligned and the loops can be easily parallelized.

We created a new data structure for particles, partitioned among threads blocks:

```
   dimension part(npmax,idimp,num_blocks)
```



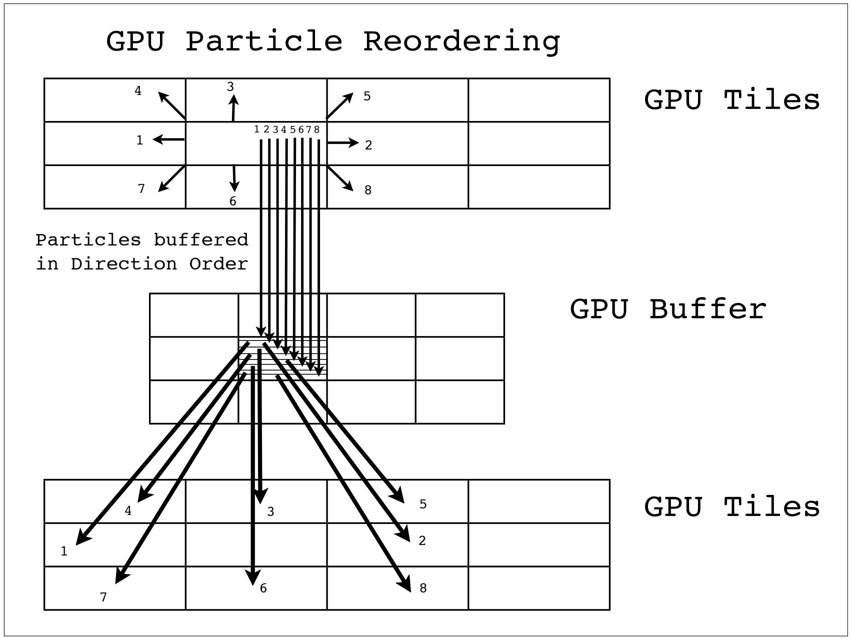
Designing New Particle-in-Cell (PIC) Algorithms:

Maintaining Particle Order

Three steps:
1. Particle Push creates a list of particles which are leaving a tile
2. Using list, each thread places outgoing particles into an ordered buffer it controls
3. Using lists, each tile copies incoming particles from buffers into particle array

A "particle manager" is needed to maintain the data alignment. This is done every timestep.
- Less than a full sort, **low overhead** if particles already in correct tile
- **Essentially message-passing**, except buffer contains multiple destinations

In the end, the particle array belonging to a tile has no gaps
- Particles are moved to any existing holes created by departing particles
- If holes still remain, they are filled with particles from the end of the array



Evaluating New Particle-in-Cell (PIC) Algorithms on GPU: **Electromagnetic Case**

2-1/2D EM Benchmark with 2048x2048 grid, 150,994,944 particles, 36 particles/cell
optimal block size = 128, optimal tile size = 16x16

GPU algorithm also implemented in OpenMP

```
Hot Plasma results with dt = 0.04, c/vth = 10, relativistic
              CPU:Intel i7    GPU:Fermi M2090  OpenMP(12 CPUs)
Push             66.5 ns.        0.426 ns.          5.645 ns.
Deposit          36.7 ns.        0.918 ns.          3.362 ns.
Reorder           0.4 ns.        0.698 ns.          0.056 ns.
Total Particle  103.6 ns.        2.042 ns.          9.062 ns (11.4x speedup).

The time reported is per particle/time step.
The total particle speedup on the Fermi M2090 was 51x compared to 1 Intel i7 core.
```

Field solver takes an additional 10% on GPU, 11% on CPU.

OK, so how about multiple CPU/GPU's?

Designing New Particle-in-Cell (PIC) Algorithms: **Multiple GPUs**

Multiple GPUs can be controlled with MPI

- Merge MPI and GPU algorithms: we use MPI for inter-GPU communications.

We started with existing 2D MPI codes from UPIC Framework

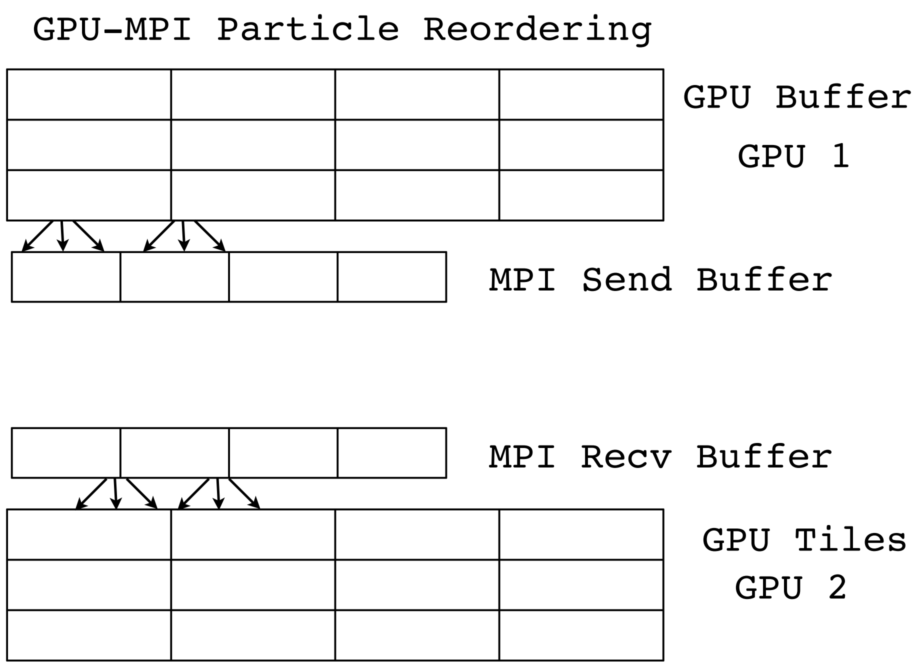
- Replacing MPI push/deposit with GPU version was no major challenge

With multiple GPUs, we need to integrate two different particle partitions

- MPI and GPU each have their own particle managers to maintain particle order

Only the first/last row or column of tiles on GPU interacts with neighboring MPI node

- Particles in row/column of tiles collected in MPI send buffer
- Because of the local domain decomposition on the GPU, table of outgoing particles are also sent
- Table is used to determine where (i.e., which tile) incoming particles must be placed, therefore, this table allows for the particles to end up at the right GPU and the right tile at the end of the MPI message passing.



GPU-MPI Particle Reordering

GPU Buffer

GPU 1

MPI Send Buffer

MPI Recv Buffer

GPU Tiles

GPU 2

Evaluating New Particle-in-Cell (PIC) Algorithms on GPU: **Electrostatic Case**

2D ES Benchmark with 2048x2048 grid, 150,994,944 particles, 36 particles/cell
optimal block size = 128, optimal tile size = 16x16. Single precision. Fermi M2090 GPU

```
Hot Plasma results with dt = 0.1
              CPU:Intel i7         1 GPU            24 GPUs            108 GPUs
Push             22.1 ns.        0.327 ns.        13.4 ps.            3.46 ps.
Deposit           8.5 ns.        0.233 ns.        11.0 ps.            2.60 ps.
Reorder           0.4 ns.        0.442 ns.        19.7 ps.            5.21 ps.
Total Particle   31.0 ns.        1.004 ns.        49.9 ps.           13.10 ps.

The time reported is per particle/time step.
The total particle speedup on the 108 Fermi M2090s compared to 1 GPU was 77x (>70% efficient),

We feel that we can improve on the current efficiency.  Currently, field solver (which uses FFT) takes an
additional 5% on 1 GPU, 45% on 2 GPUs, and 73% on 108 GPUs.  And we believe the efficiency should be
higher for PIC codes with a finite-difference solver.
```

PIC Algorithms on future architectures are largely a hybrid combination of previous techniques

- Vector techniques from Cray (old fashioned vector Crays)
- Blocking techniques from cache-based architectures
- Message-passing techniques from distributed memory architectures

Scheme should be portable to other architectures with similar hardware abstractions (such as the intel Phi)

Further information available at:
http://www.idre.ucla.edu/hpc/research/

Source codes available at the **UCLA PICKSC web-site**

UCLA Particle-in-Cell and Kinetic Simulation Software Center (PICKSC), NSF funded center whose Goal is to provide and document parallel Particle-in-Cell (PIC) and other kinetic codes.

http://picksc.idre.ucla.edu/

Planned activities

- Provide parallel skeleton codes for various PIC codes on traditional and new parallel hardware and software systems.
- Provide MPI-based production PIC codes that will run on desktop computers, mid-size clusters, and the largest parallel computers in the world.
- Provide key components for constructing new parallel production PIC codes for electrostatic, electromagnetic, and other codes.
- Provide interactive codes for teaching of important and difficult plasma physics concepts
- Facilitate benchmarking of kinetic codes by the physics community, not only for performance, but also to compare the physics approximations used
- Documentation of best and worst practices, which are often unpublished and get repeatedly rediscovered.
- Provide some services for customizing software for specific purposes (based on our existing codes)

Key components and codes will be made available through standard open source licenses and as an open-source community resource, contributions from others are welcome.

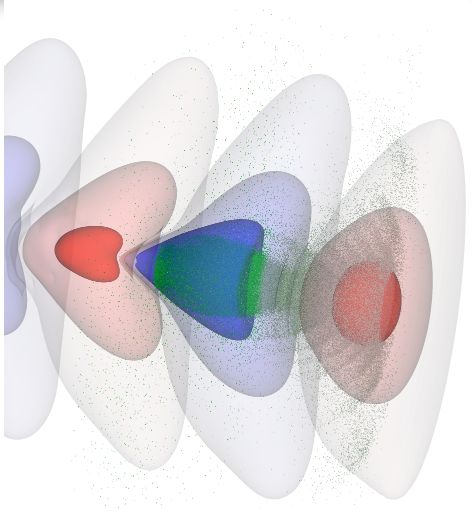
And we are hiring good post-docs! (please contact me or Prof. Warren Mori, the PI of this project)

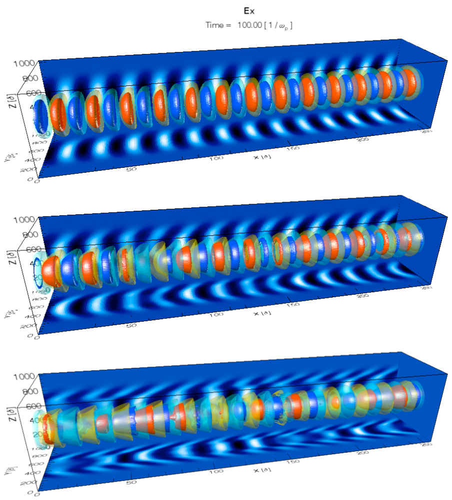
UCLA

Summary and Outline

OUTLINE

- Overview of the project
  - Particle-in-cell codes
  - PIC codes available @ PICKSC
- Application of OSIRIS to plasma based accelerators:
  - QuickPIC simulations of SLAC experiments
- Applications of OSIRIS to LPI's Relevant to IFE
  - SRS in indirect drive IFE targets (such as NIF).
  - Estimates of large scale LPI simulations (& the need for exascale supercomputers)
- Development works for Blue Waters and beyond (including GPU's and other emerging architectures) + the PICKSC Center @ UCLA





Special Thanks to Galen and the Blue Waters Team!