Certifying Synergia for CERN accelerators

James Amundson and Eric Stern

Fermilab

April 17, 2013

Fermilab

The ComPASS Project

The ComPASS Project is funded by the US DOE's SciDAC program.

Synergia

- Accelerator simulation package
  - independent-particle physics
    - linear or nonlinear
  - collective effects
    - simple or computationally intensive
  - can go from simple to complex, changing one thing at a time
- Goal: best available physics models
  - *best* may or may not mean *computationally intensive*

https://compacc.fnal.gov/projects/wiki/synergia2

please ignore certificate warning

- Designed for range of computing resources
  - laptops and desktops
  - clusters
  - supercomputers
- Goal: best available computer science for performance
  - significant interaction with computer science community

Personnel

Synergia is developed and maintained by the

Computational Physics for Accelerators group

in Fermilab's

Scientific Computing Division

James Amundson, Paul Lebrun, Qiming Lu, Alex Macridin, Leo Michelotti (CHEF), Chong Shik Park, Panagiotis Spentzouris and Eric Stern

With development contributions from Tech-X: Steve Goldhaber

Physics

- Single-particle physics are provided by CHEF
  - direct symplectic tracking
    - magnets, cavities, drifts, etc.
  - (and/or) arbitrary-order polynomial maps
  - many advanced analysis features
    - nonlinear map analysis, including normal forms
    - lattice functions (multiple definitions)
    - tune and chromaticity calculation and adjustment
    - etc.
- Apertures
- Collective effects (single and multiple bunches)
  - space charge (3D, 2.5D, semi-analytic, multiple boundary conditions)
  - wake fields
    - can accommodate arbitrary wake functions
  - electron cloud
    - proof of principle only

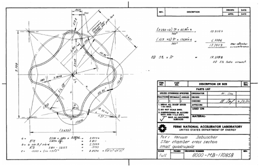
Space charge in Synergia

Variety of boundary conditions and levels of approximation

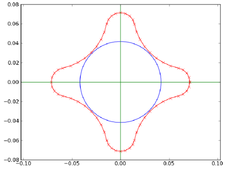
- 3D open transverse boundary conditions
  - Hockney algorithm
  - open or periodic longitudinally
- 3D conducting rectangular transverse boundary
  - periodic longitudinally
- 3D conducting circular transverse boundary
  - periodic longitudinally
- 2.5D open boundary conditions
  - 2D calculation, scaled by density in longitudinal slices
- 2D semi-analytic
  - uses Bassetti-Erskine formula
  - σ_x and σ_y calculated on the fly
- New space charge models can be implemented by the end user

Synergia aperture model

- Apertures can be associated with elements and/or steps
- 2D model
  - could be extended with slices
- Geometric
  - circular
  - elliptical
  - rectangular
  - polygon
  - wire
- Abstract
  - phase space
  - Lambertson
    - removes particles
- New apertures can be implemented by the end user



Engineering drawing of FNAL Debuncher quad cross section



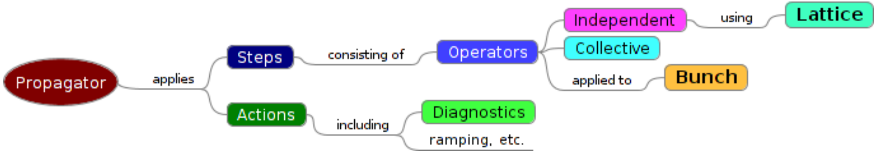
Synergia implementation: detailed, but fast (inscribed circle optimization)

Synergia 2.1 design

- Synergia 2.1 is a major milestone
  - very different from Synergia 1
  - significantly different from Synergia 2
  - designed for widespread use
- Synergia is a mix of C++ and Python
  - all computationally-intensive code is written in C++
  - user-created simulations are usually written in Python
    - pure-C++ simulations are possible
- Synergia provides a set of functions and classes for creating simulations
  - many examples available
- Virtually every aspect of Synergia is designed to be extendable by the end-user
  - code in C++ and/or Python
- Synergia 2.1 beta release scheduled for May 1, 2013

Synergia simulations

- A simulation consists of propagating a *Bunch* (or *Bunch*es) through a *Lattice*.
- Inputs: machine lattice, initial bunch parameters
- Outputs: user-selected *Diagnostics*.



- *Diagnostics*
  - 6D means
  - 6D std deviations
  - 6x6 covariance matrix
  - 6x6 correlation matrix
  - individual particle tracks
  - dump of all particles
  - losses at locations in lattice
  - can be extended. . .

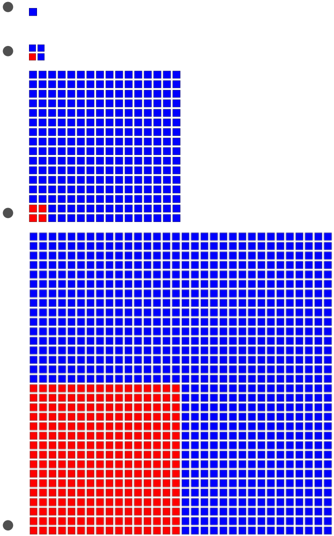
- *Actions* can specify *Diagnostics* will be applied
  - every *n* steps
  - every *m* turns
  - at specified sets of steps
  - at specified sets of turns
  - by user-specified logic
  - more

Feature: checkpointing

- Synergia simulations can be saved to disk (checkpointed) at any point
  - allows recovery from hardware failure
  - allows jobs that take longer than batch queue limits
- All simulation objects can be checkpointing
  - even, e.g., objects with open files
- Checkpointing available for both C++ and Python objects
  - *including end-user objects*
- User specifies parameters
  - every n turns
  - do p out of q total turns
  - send a message to stop at the end of next turn

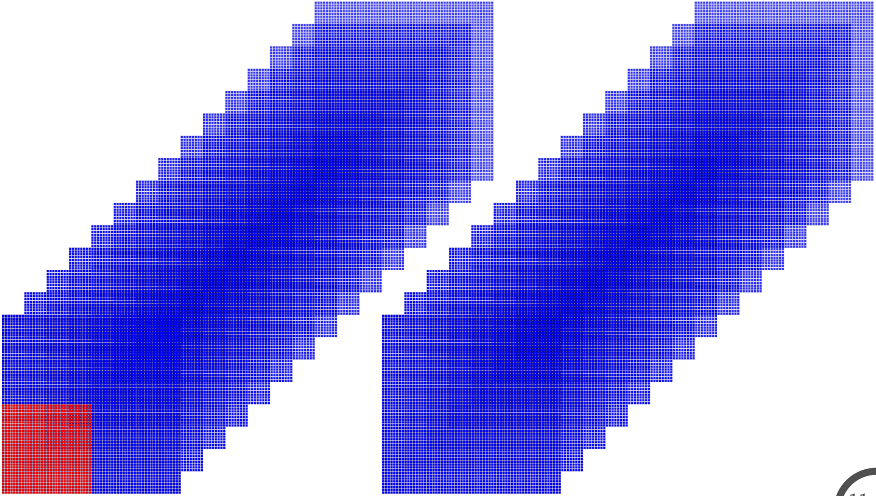
Feature: scalability

- Why?
  - statistics (many particles)
  - multiple bunches
  - take advantage of modern computing resources
- I use Synergia every day on a single CPU
- Synergia can take advantage of multiple cores on a single CPU
- We regularly run Synergia using 256 cores of a Linux cluster
- Single-bunch Synergia simulations scale well to $\approx$ 1024 cores on supercomputers such as IBM BlueGene or Cray XT



Feature: scalability

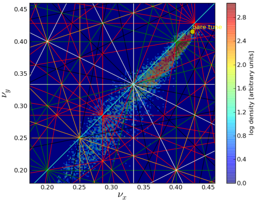
- Multi-bunch Synergia simulations have been shown to scale to 131,072 cores on Intrepid, a BlueGene/P supercomputer
  - $> 10^{10}$ particles
  - INCITE13: 80M hours ($>$ 9000 cores continuously)

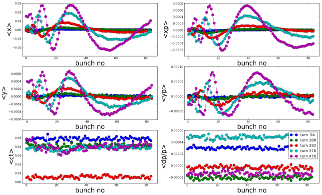


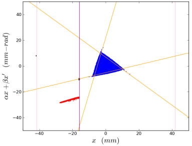
Status

Synergia 2.1 is being used for all production work in our group at Fermilab.

- Fermilab Main Injector
  - space charge, multipoles, detailed apertures, orbit bumps
- Fermilab Booster
  - space charge, wakes, 84 bunches
- Fermilab Debuncher (Mu2e experiment)
  - space charge, ramping, resonant extraction
- Hybrid MPI-OpenMP and MPI-GPU versions







Requested upgrades for CERN accelerators

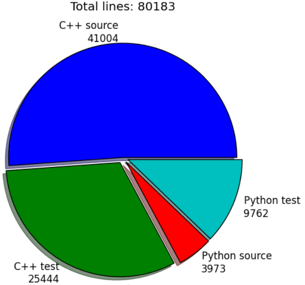
In Fall 2012 we received a list of requests for Synergia in order to be useful for LIU. Some were already available, some were already on the to do list, a few were new.

Status:

- Read MadX lattice files
  - Done. Was on to do list.
- Higher harmonics in RF cavities
  - Done. Was on to do list.
- Test particle tracking
  - New feature. In progress.
- Bend edge effects
  - New feature. Should be done by the end of April.
- Foils
  - New feature. Need to choose model.
- Manual
  - Started! Biggest item on to do list.
- Validation with space charge trapping benchmark.
  - See next slides...

Space charge trapping benchmark

- We do a great deal of testing in Synergia
  - ≈ 80k lines of code
    - excluding CHEF
  - $>$ 40% tests



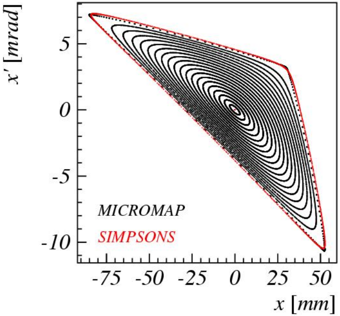
However, it is important to show that we can reproduce non-trivial simulations done with other programs.

- Space charge trapping benchmark in GSI SIS18
  - `http://web-docs.gsi.de/~giuliano/research_activity/trapping_benchmarking/main.html`
- Discussed elsewhere at this workshop
- *The aim of the code benchmarking is to confirm the space charge induced trapping of particles in a bunch during long term storage.*

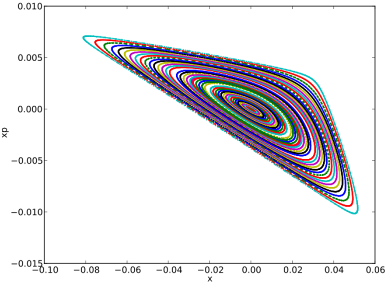
Benchmark step 1

Phase space with sextupole and no space charge.

Benchmark



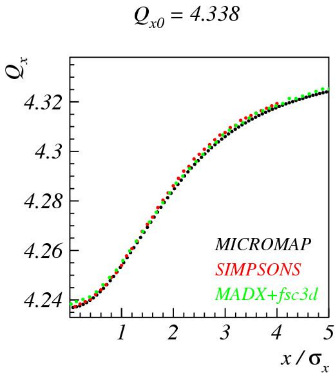
Synergia



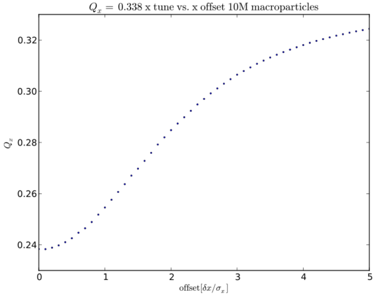
Benchmark step 2

Phase space with sextupole and no space charge.

Benchmark



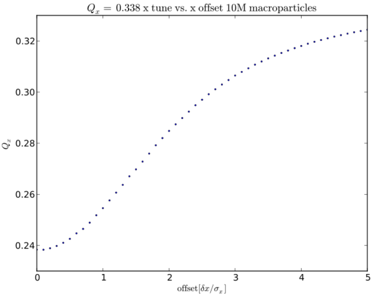
Synergia



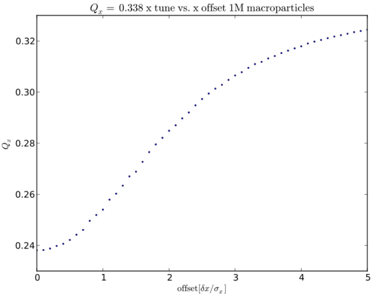
Some observations from tracking single particles in PIC (1)

Statistics have observable effects.

10 million particles



1 million particles



Some observations from tracking single particles in PIC (2)

- Single-particle tracking can reveal bugs that are washed out in collective diagnostics (beam widths, emittances, etc.)
  - Found a very subtle bug arising from two overlapping optimizations.
    - Space charge kicks were shuffled once every 10000 steps.
    - Pathological (*really* well-hidden bug).
    - Not apparent in collective diagnostics.
    - Statistics made it difficult to sort out.
    - Found some kicks $> 25\sigma$ from mean.

Some observations from tracking single particles in PIC (3)

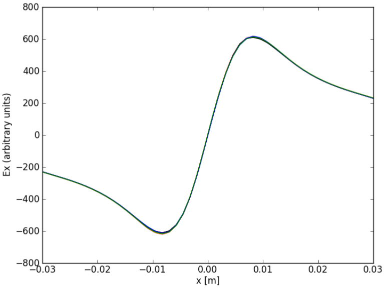
- A particle at $(0, 0)$ experiences no forces (under the right conditions, of course...)
  - Therefore, it should not leave $(0, 0)$.
- In a PIC simulation, a particle starting at $(0, 0)$ does not behave that way.
  - Why?
  - Does the center become hollow?

Why the $(0,0)$ particles moves

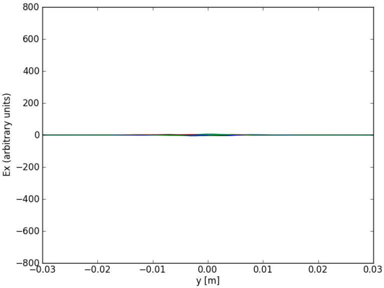
The figures below each contain ten curves from ten different turns in a PIC simulation with 1M particles.

- 1M particle simulation
- 10 field evaluations, different turns
- field units arbitrary, but consistent across plots

$E_x(x,0,0)$



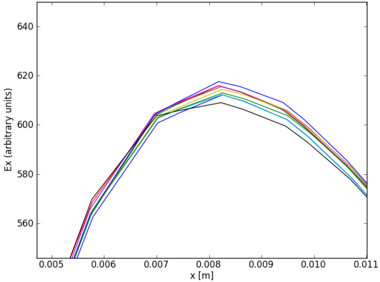
$E_x(0,y,0)$



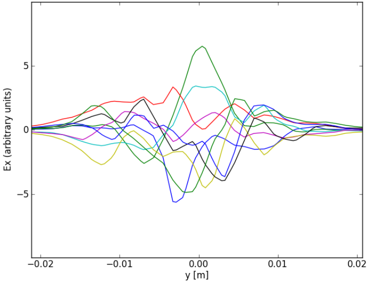
Why the $(0, 0)$ particles moves

Zooming in:

$E_x(x, 0, 0)$



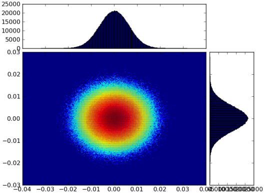
$E_x(0, y, 0)$



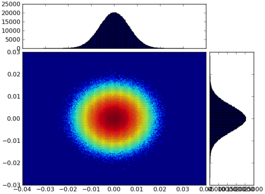
- relative errors along x-axis are small
- similar magnitude errors along y-axis, but relative to 0(!)
  - sum of different curves tends to 0

Does the center become hollow?

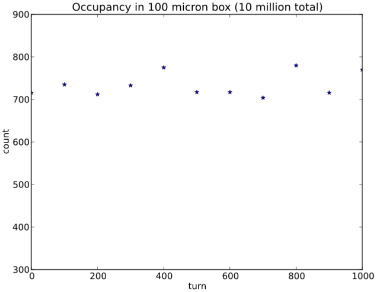
$x - y$ distributions
initial



after 1000 turns



Particle occupancy within 100μ

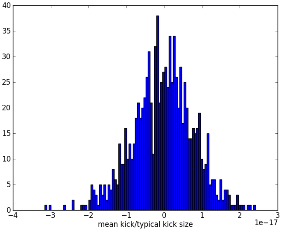


Conclusion: particles moving away from the origin are compensated by particles moving toward the origin.

Some observations from tracking single particles in PIC (4)

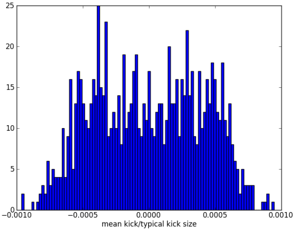
- With open boundary conditions, space charge cannot move the beam.
- Is this true in our PIC simulations?
  - Our 3D and 2D solvers produce zero net space charge kick by construction
  - 2.5D does not have this property

3D mean kicks



scale: $\approx 10^{-17}$

2.5D mean kicks



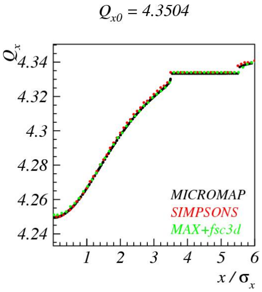
scale: $\approx 10^{-3}$

14 orders of magnitude difference!

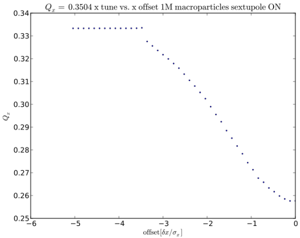
Benchmark step 4

The fourth step is to benchmark the dependence of a test particle tunes from its amplitude when the sextupole is on.

Benchmark



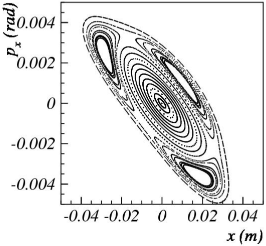
Synergia



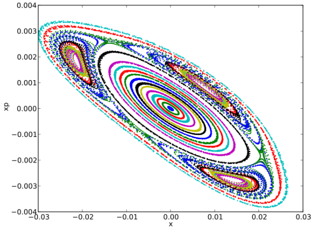
Benchmark step 5

The fifth step is to benchmark the phase space with test particles when the sextupole is on and in presence of space charge.

Benchmark



Synergia



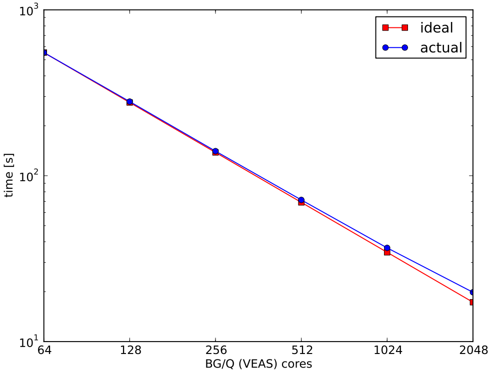
Conclusions

- Synergia 2.1 is
  - capable of both simple and complex simulations
  - being actively used in production
  - designed for end users
  - extensible
  - scalable
  - being enhanced for LIU
- Benchmarking is proceeding well

Backup slides

Strong scaling

Performed large-scale scaling benchmarks on pre-release BlueGene/Q machine at Argonne Leadership Computing Facility: Strong scaling, *i.e.*, fixed problem size ($32 \times 32 \times 1024$ grid, 100 grid cells per particle, trivial apertures)



Weak scaling

Performed large-scale scaling benchmarks on production BlueGene/P machine at Argonne Leadership Computing Facility: Weak scaling, *i.e.*, fixed ratio (problem size)/(compute size) ($32 \times 32 \times 1024$ grid, 100 grid cells per particle, trivial apertures)