Vector Electromagnetic Physics Models & Field Propagation

Guilherme Amadio (UNESP),
Ananya,
John Apostolakis, Marilena Bandieramonte, Mihaly Novak (CERN)
Soon Yung Jun (Fermilab)

21th Geant4 Collaboration Meeting
January 18 – 22, 2016
Ferrara, Italy

Outline – EM Physics Models

- Context
- Approaches
  - Tabulated (pre-simulated) reactions
  - Borrowing distributions from established G4 models
  - Re-constituted modeling
- EM processes
  - Electron models
  - Photon models
- Validation & speedup of photon models
- Outlook

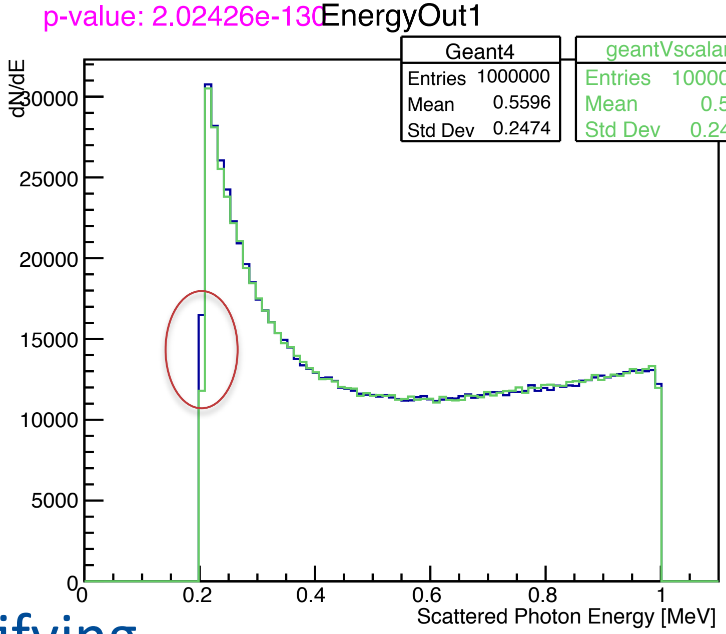
Approaches in Modeling

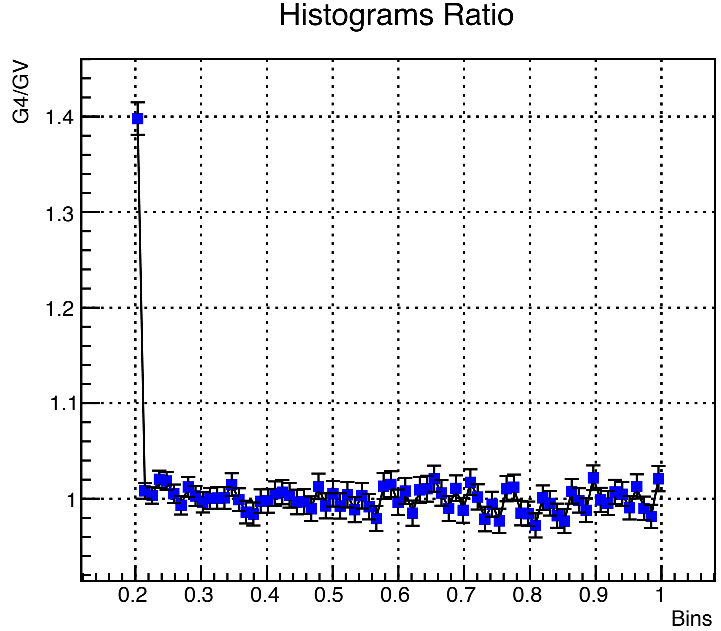
- Tabulated (pre-simulated) reactions
  - Use 5 G4 interactions per (projectile, target, energy) tuple,
  - Aims to obtain a (mostly) realistic shower development
  - Choice of production threshold in energy for each library created
  - Used as a first implementation
- **"Straight-forward" vectorization & new sampling**
  - Borrowing distributions from established G4 models
  - Allows faster 'startup' of models
  - Used for photon processes
- Re-constituted modeling "from the ground up"
  - Recalculation from first principles of models (as applicable)
  - Went back to original measurements/tabulations for available 'core' quantities
  - To date, undertaken for all electron processes

Photon Models: Sampling and its Validation

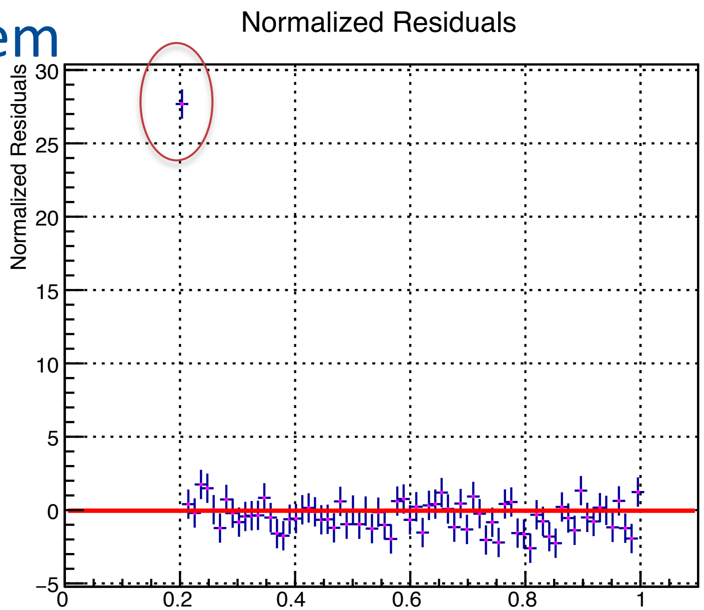
- Physics validation
  - simulate interactions at fixed input energy points
  - compare results (final states of interaction) w.r.t "Geant4" (code copied/extracted from Geant4)
- Criteria
  - χ^2 comparison & p value
  - Plot ratio of distributions, residuals
  - Q-Q plot: quantiles of two distributions (2% point vs 2% point, 4% vs 4%, …)

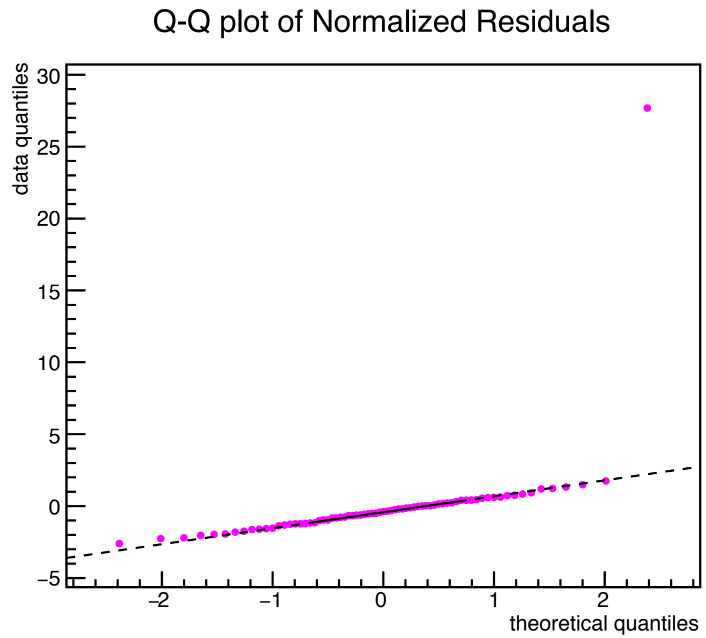
Verification of Compton Klein Nishina model – 1 MeV



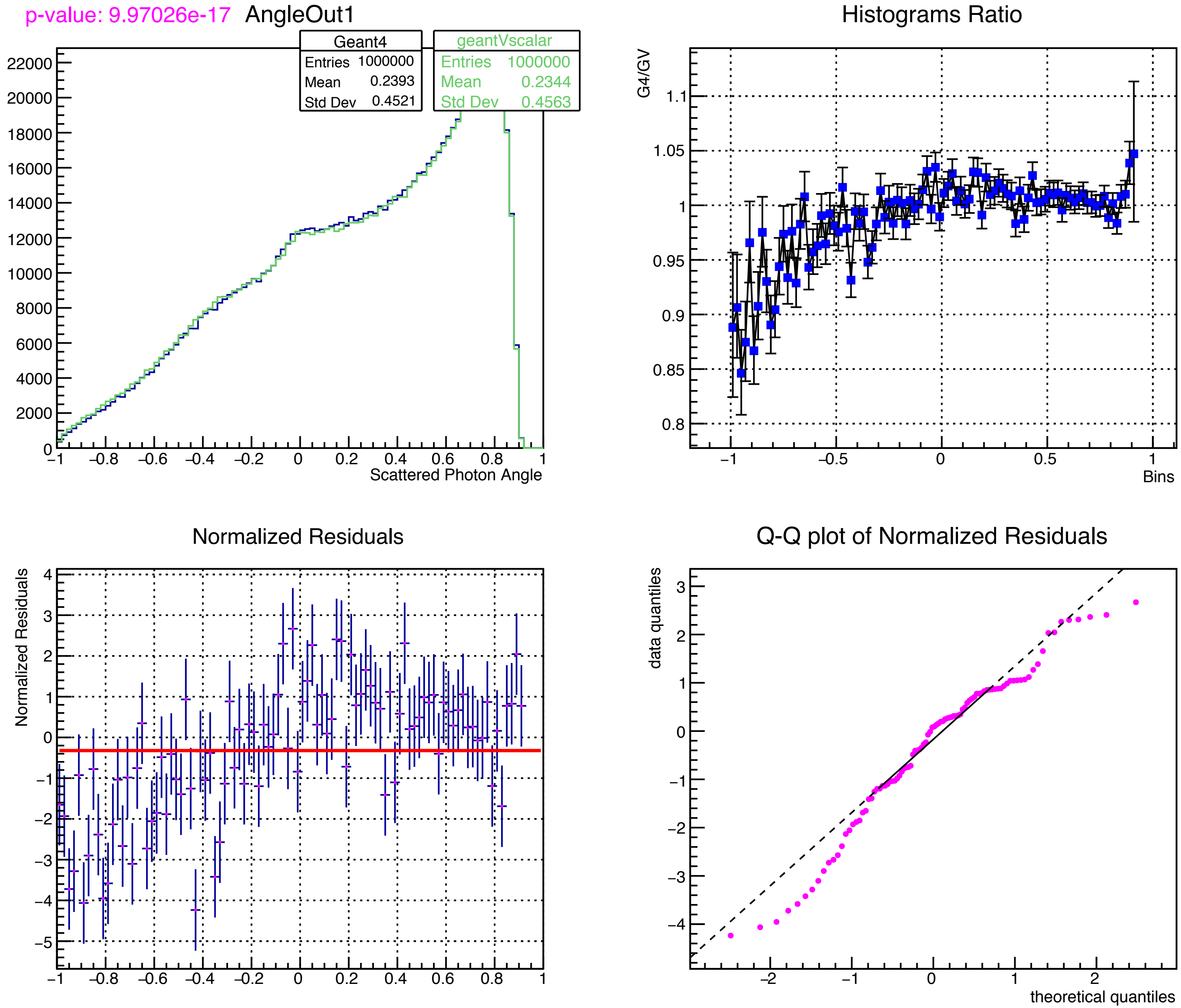


Identifying problem points



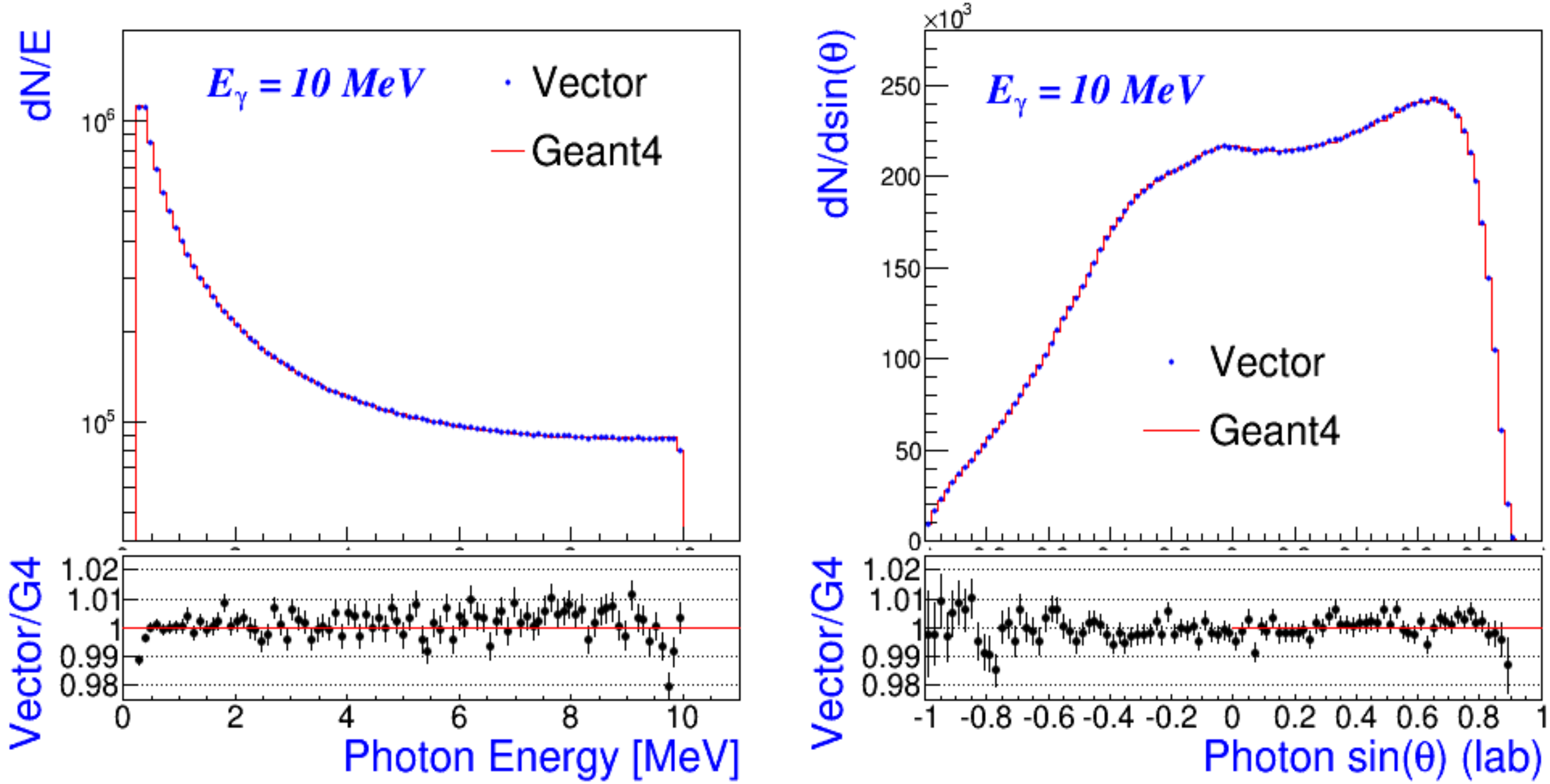


Verification of Compton Klein Nishina model – 100 MeV



Preliminary Validation: Alias Sampling Method vs. Geant4

- Compton: scattered photon energy and angular distributions with input photon energy = 10 MeV (Vector)



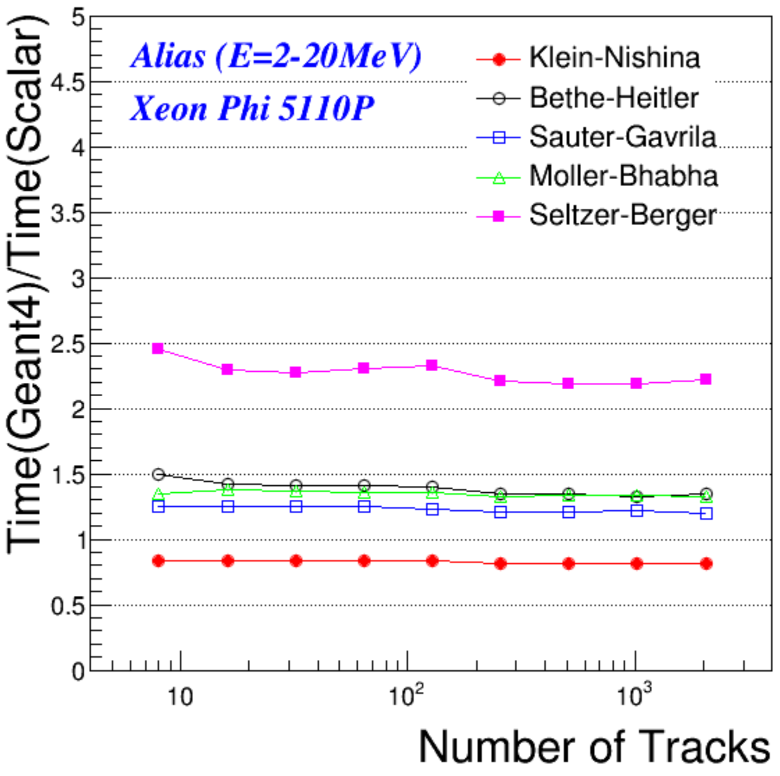
1%-level agreement up to 100 MeV with alias table size [100,200]

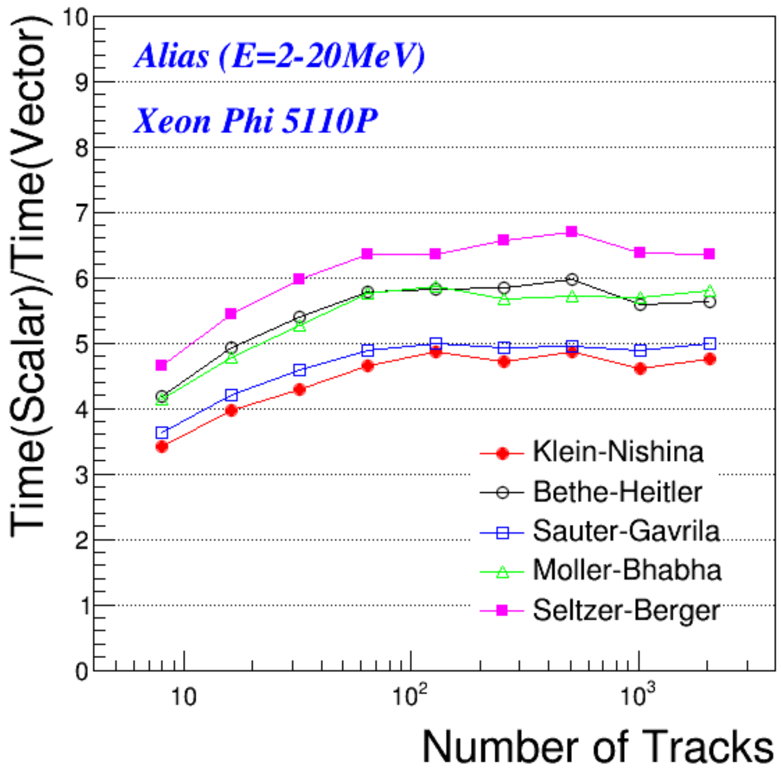
Validation and Performance Evaluation (Preliminary)

- Performance measurement
  - 10 experiments, 100 repetitions per experiment
  - time for simulating N secondary interactions with a exponentially falling input energy spectrum in the range;
    - input energy range = [2MeV:20MeV] with 16 Z-elements
    - also tested in E [10keV:1MeV], E [10MeV:1GeV], E [10GeV:1TeV]
    - note that performance of Geant4 depends on the energy range
  - sample and store secondary particles, and update primary tracks
  - speedup = scalar/vector or scalar/GPU as the number of input particles (tracks) for the same task

Preliminary Performance: Alias Sampling Method

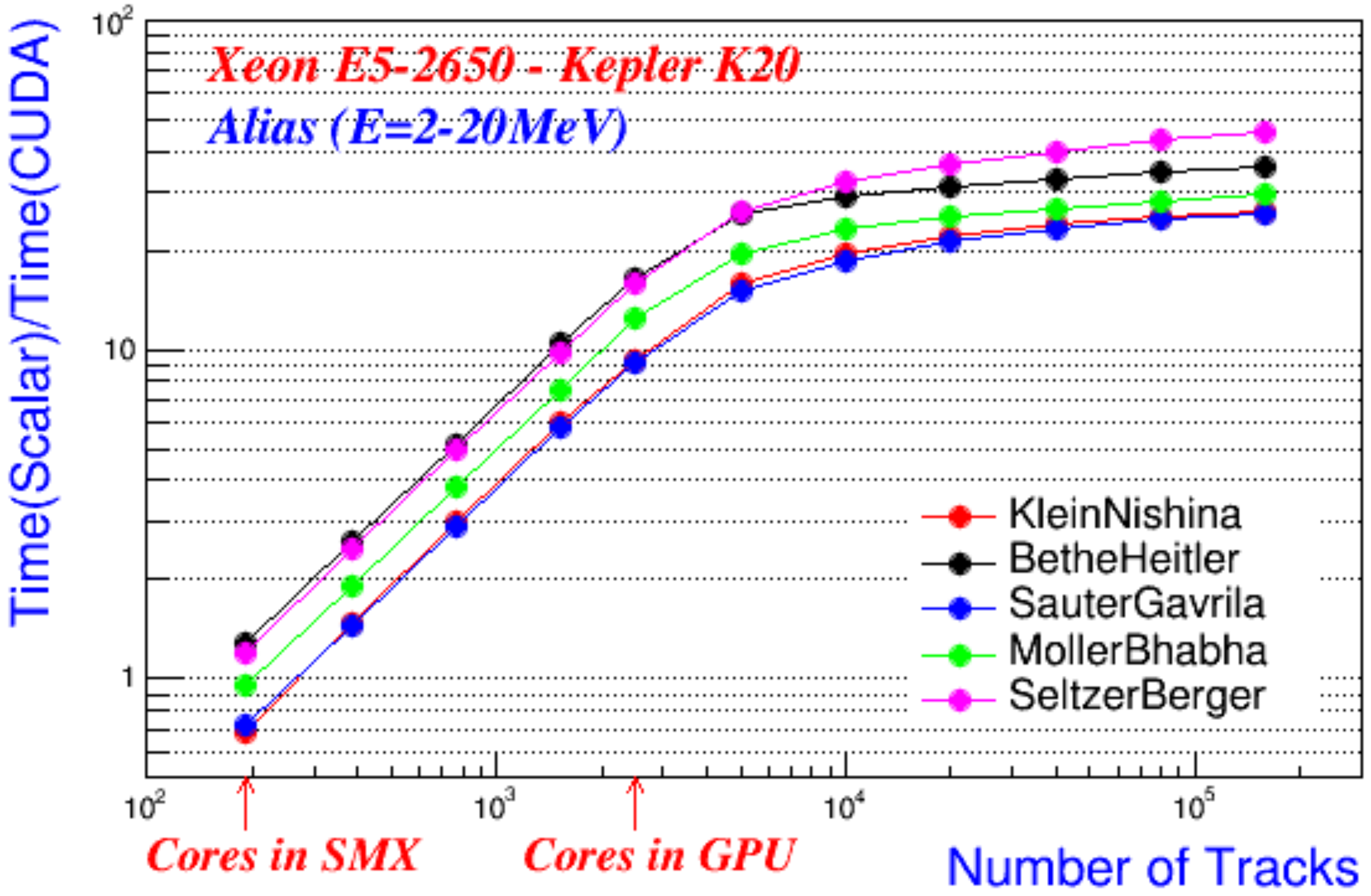
- Vector performance
  - input particle energy: 2-20 MeV (valid range for all models)
  - using 16 elements (random for each track)
  - MIC (Intel Xeon Phi 5110P 60 cores @ 1.053 GHz) - 8 vector pipelines for double precision – see also SSE/AVX in backup





Preliminary Performance: Alias Sampling Method – GPU

- GPU
  - GPU: Nvidia Kepler (K20), 2496 cores @ 0.7 GHz - <<<26,192>>>
  - Host: Intel Xeon E5 – 2650 @ 2.60 GHz



Next steps & Outlook

- Integration into GeantV tracking loop – with validation
  - a single gamma processes (in progress)
  - multiple gamma processes
  - electron process(es)
- Full set of electron & gamma processes in tracking loop
  - Validation of EM showers
  - Profiling & optimisation
  - Speedup evaluation
- Neutron processes
  - Process cross section – full reevaluation (BARC)
  -

Electron processes and models

- Model for multiple scattering prepared
  - Portable between G4 and GeantV
  - Approach & validation presented in 2015 Geant4 workshop
  - Released as ‘GS’ Mult.Scat. model in Geant4 10.2 (Dec 2015)

- Models for bremmstrahlung
  - Prepared new ‘ground-up’ implementation of brem
  - Fully integrated density & LPM effects
  - Identified correction for LPM effect in G4 brem model (fixed)
- Models for ionisation

- Status
  - Further validation underway
  - Integration into GeantV tracking is in preparation

VecPhys Summary

- Demonstrated feasibility of implementing electromagnetic physics models for scalar, Vector(SIMD) and GPU (SIMT) architectures with common source code
- Integrated different sampling methods (alias, adaptations of accept/reject) for models of photon processes
- New electron models ‘from the ground up’ – multiple scattering fully validated in Geant4 simulations
- Validating physics results with Geant4 and evaluating computing performance
- Outlook
  - Optimize further for both SIMD and SIMT
  - Integrate in the GeantV framework
  - Measure performance in full simulation

Magnetic Field Propagation

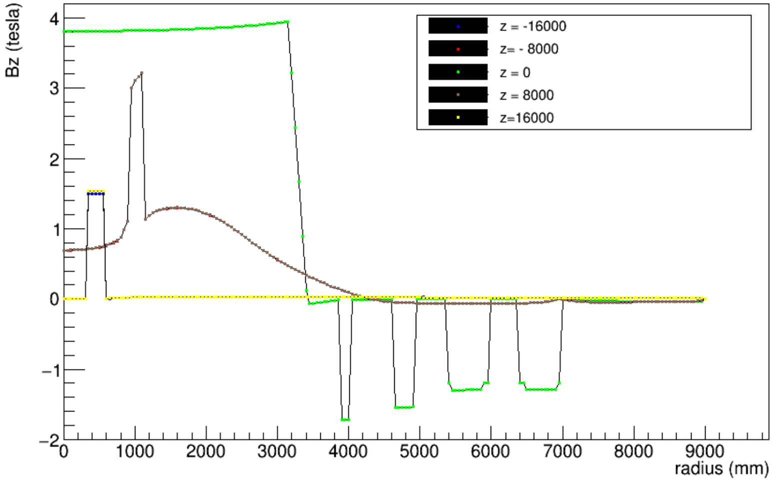
Ananya
John Apostolakis

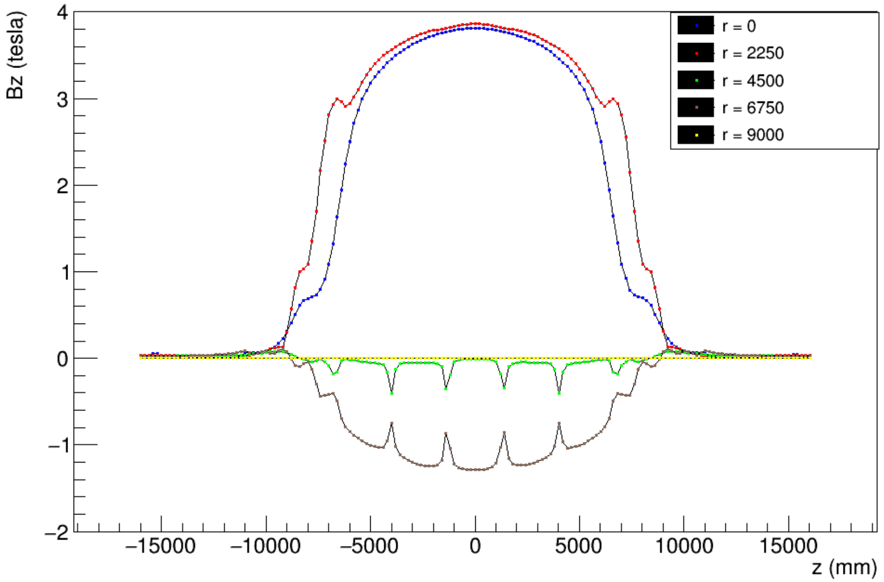
Outline

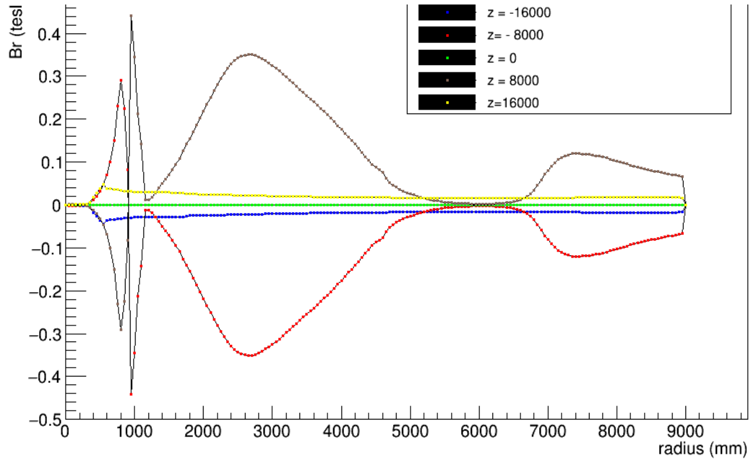
- GeantV and magnetic field tracking
- The CMS magnetic field
  - Approximation using Bilinear Interpolation
  - Vectorization and memory layout
  - Optimization
  - VTune Analysis
- Vectorization of Integration of motion

Problem statement

- Start with sample values of 2D CMS field.
- Assume phi-symmetric field.
- Find magnetic field given a point in 3D space.

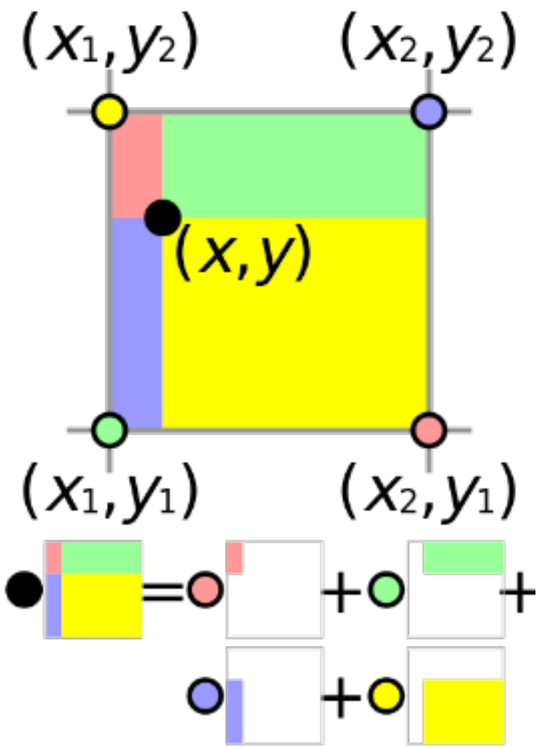






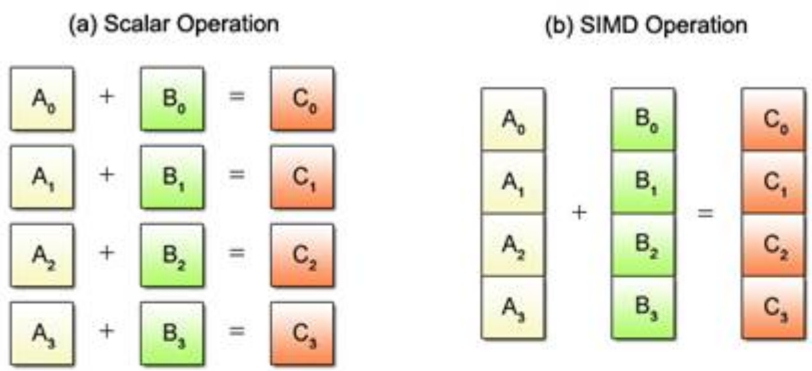
WHAT DO WE DO?

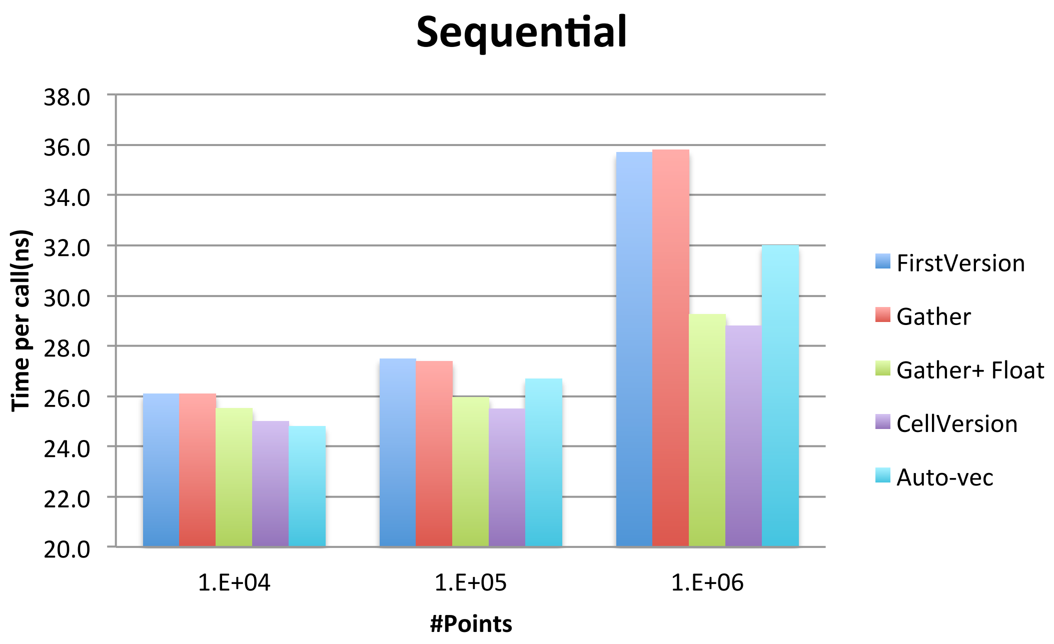
- Read given 2D map.
- Find corresponding magnetic field using bilinear interpolation on values from map.

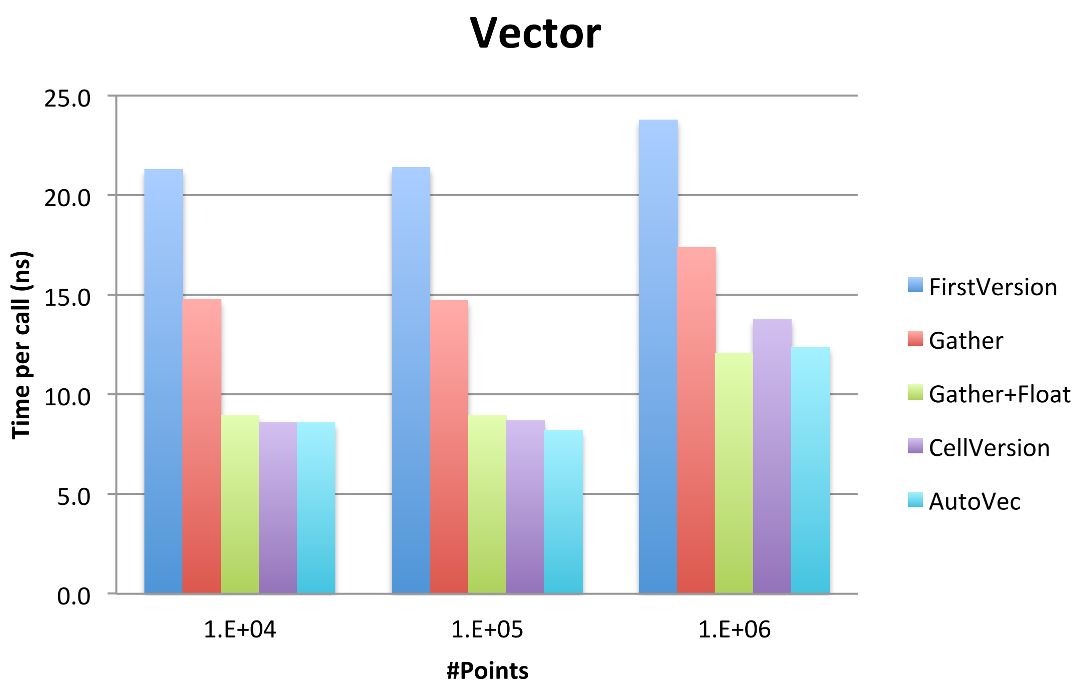


Vectorization

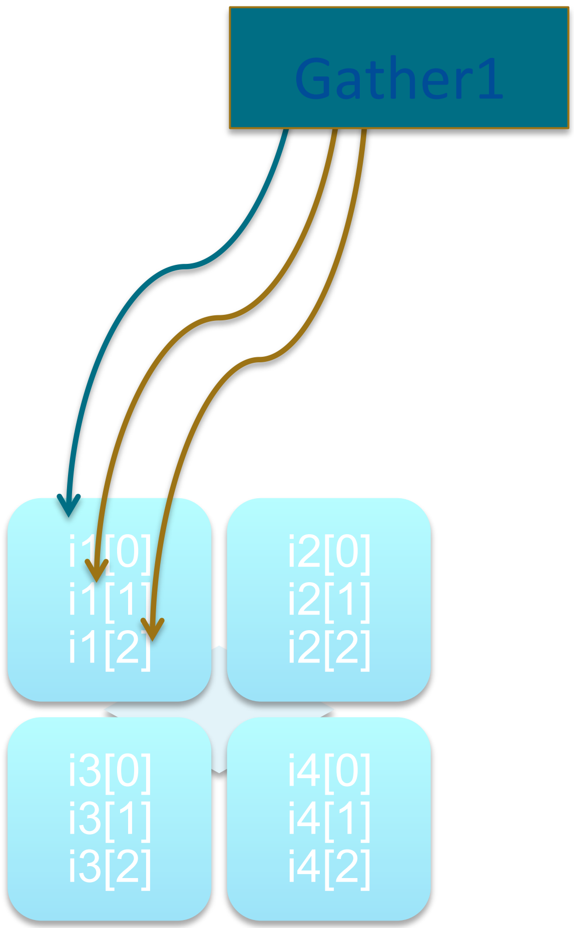
- *"**Vectorization**" (simplified) is the process of rewriting a loop so that instead of processing a single element of an array N times, it processes (say) 4 elements of the array simultaneously N/4 times.*
- What are we doing?
  - Processing multiple particles/tracks simultaneously

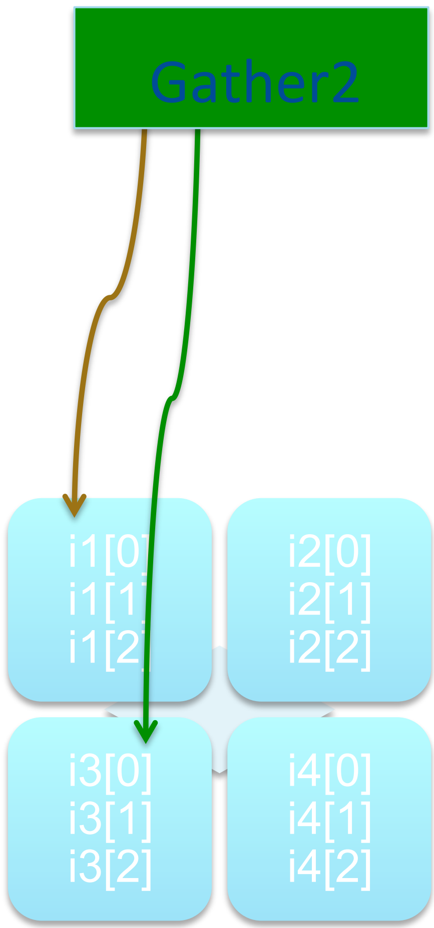


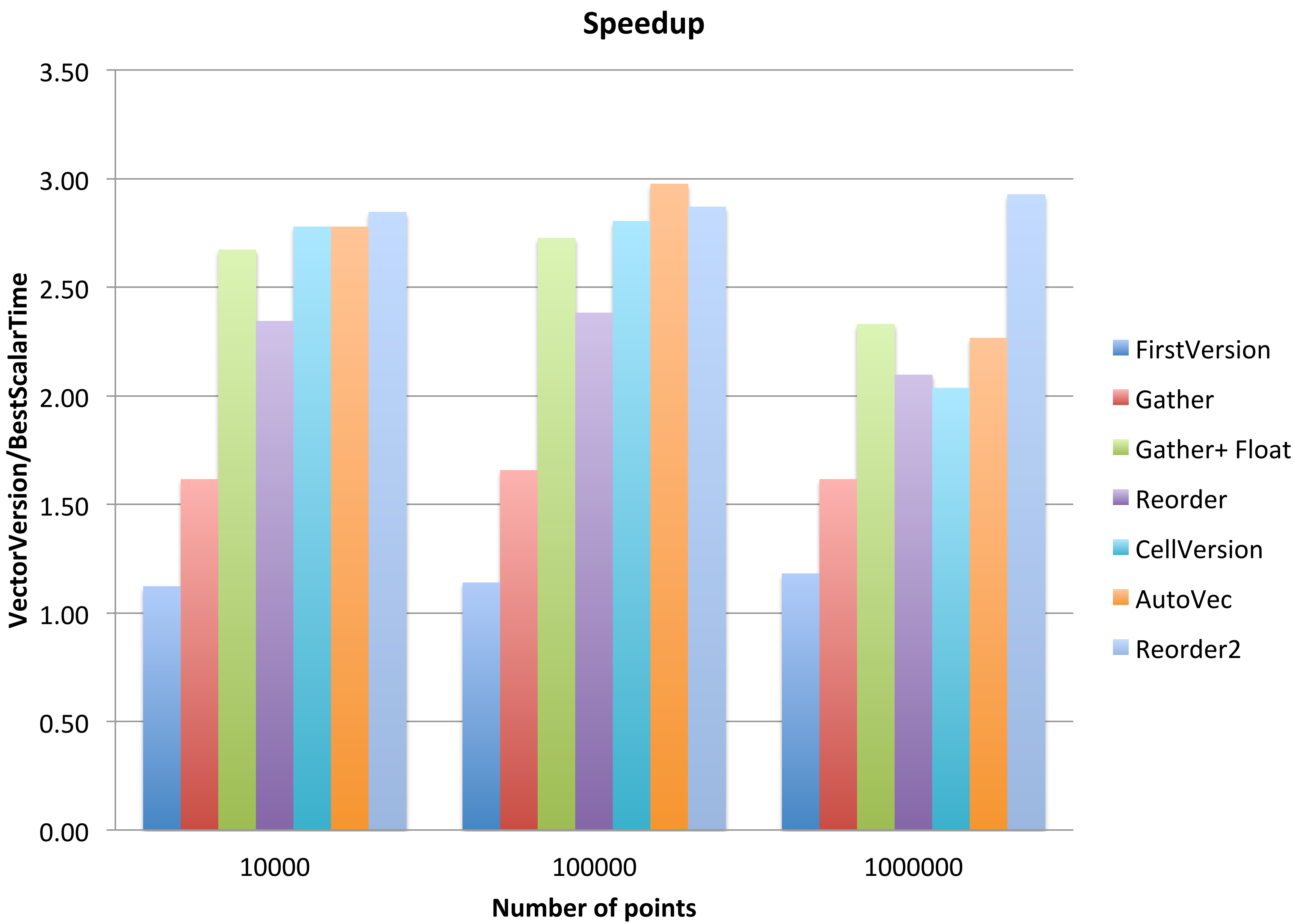

Sequential
38.0
36.0
34.0
32.0
30.0
28.0
26.0
24.0
22.0
20.0
Time per call(ns)
1.E+04
1.E+05
1.E+06
#Points
FirstVersion
Gather
Gather+ Float
CellVersion
Auto-vec


Vector
25.0
20.0
15.0
10.0
5.0
0.0
Time per call (ns)
1.E+04
1.E+05
1.E+06
#Points
FirstVersion
Gather
Gather+Float
CellVersion
AutoVec

Different Ordering of gather






Speedup
VectorVersion/BestScalarTime
3.50
3.00
2.50
2.00
1.50
1.00
0.50
0.00
10000
100000
1000000
Number of points
FirstVersion
Gather
Gather+ Float
Reorder
CellVersion
AutoVec
Reorder2

observations

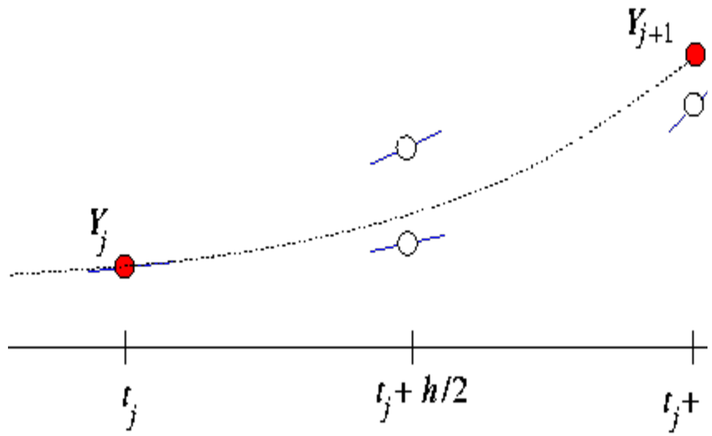
- Speedup factor of ~3
- Semi-realistic benchmark:
  - Half the points are new; the other half are ‘moved’ near to previous values.
  - Exponential random distribution.
  - Time reduced by ~5%. Likely effect is from cache.
- Difference in performance from changing doubles to floats:
  - 3-20% for sequential
  - 30-40% for vector version
- Difference in performance from changing order of memory operations:
  - 5-7% for sequential
  - 5-20% for vector version

Integrating motion

		Vecotrizable
Magnetic Field	$\vec{B}$	✓
Force	$\vec{F} = \frac{1}{m} \vec{p} \times \vec{B}$	✓
Equation of motion (ODE)	$\frac{d\vec{\mathbf{x}}}{ds} = \frac{\vec{\mathbf{p}}}{\lvert\vec{\mathbf{p}}\rvert}$ $\frac{d\vec{\mathbf{p}}}{ds} = \frac{1}{\mathbf{p}} \overrightarrow{\mathbf{F}(v)}$	✓

Vectorizable

Runge Kutta ✓



Input : y_o

Output: y_1, Δy_1

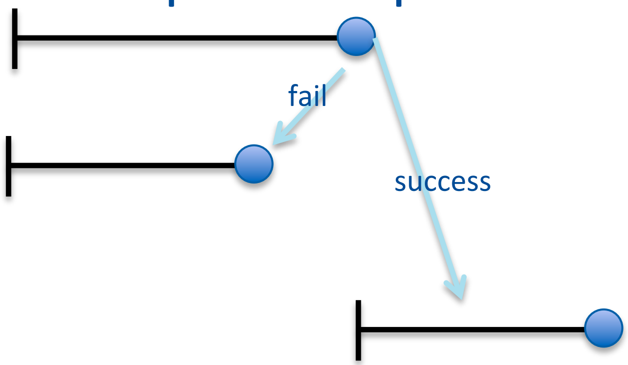
Driver

Error Control

Adaptive Stepsize

✗

(Not Naively vectorizable)



Field Propagation - Outlook

- Vectorized field propagation is in progress
  - Field & equation of motion vectorised
  - Helix and different RK steppers vectorised
  - ‘Driver’: different methods for keeping vectorisation when handling multiple tracks are under investigation

Backup slides

Vtune analysis

		Elapsed Time	Instructions Retired	CPI Rate	Back-end Bound	Memory Bound	Core Bound	Port Utilization
Sequential (nRep = 200)	Reorder2 (Haswell Xeon)	8.166	31.7B	0.947	0.786	0.282	0.505	0.327
Vector (nRep = 500)	Reorder2 (Haswell Xeon)	7.665	33.8B	0.776	0.697	0.143	0.555	0.463