Identified hadron spectra analysis applying parallel Geant simulation for future

Gábor Bíró

Eötvös Loránd University/Wigner RCP, Hungary

G.G. Barnaföldi, MTA Wigner FK

G. Papp, Eötvös Loránd University

F. Carminati, M. Novák, S. Wenzel, CERN PH-SFT

Zimányi Winter School On Heavy Ion Physics

01-05.12.2014.

Outline

- **Why** do we need simulations?
  - Tsallis – Pareto fits as motivation
- **How** much simulation do we need?
  - Requirements of the present and future
- **How** can it be faster?
  - Introducing Geant Vector Prototype
- **How** faster is it now?
  - Speedtest of elementary volumes
- **Summary**

Why do we need simulations?

- Monte Carlo simulations are essential in todays HEP
  - Event generators
  - Detector simulations
- Response of huge detector systems needs to be precisely predicted
- Tsallis – Pareto distributions fit identified hadron spectras well in a broad p_T range
  (arXiv: 1409.5975v1)

$$E\frac{\mathrm{d}N}{\mathrm{d}p_T} = A \cdot m_T \left(1 + (q-1)\frac{m_T}{T}\right)^{-\frac{1}{q-1}}$$

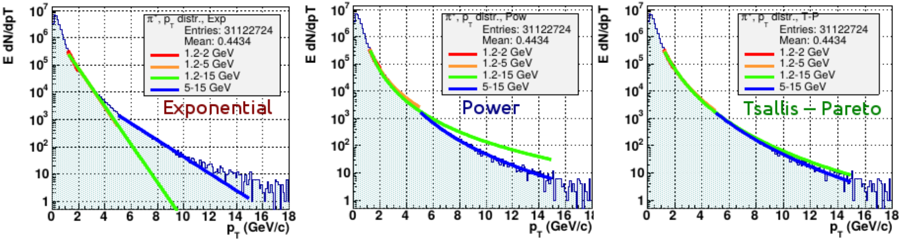
Why do we need simulations?

- **Example:** fits of π^+ spectra with various p_T ranges, made by pp Monte Carlo simulation

$$f_{exp}(p_T) = A_{exp} \cdot e^{-b \cdot p_T} \quad ,$$

$$f_{pow}(p_T) = A_{pow} \cdot p_T^{-n} \quad ,$$

$$f_{TP}(p_T) = A_{TP} \cdot p_T \cdot m_T \left(1 + \frac{q-1}{T} E\right)^{-\frac{1}{q-1}}$$

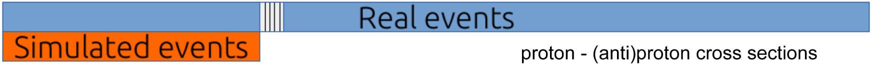


- 10^6 pp events with $\sqrt{s} = 14$ TeV generated with PYTHIA6.4
- Tsallis – Pareto fits well

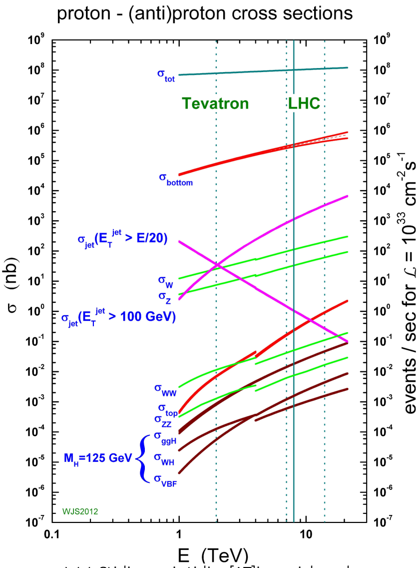
χ^2/ndf, Pion

p_T [GeV]	[1,2:2]	[1,2:5]	[1,2:15]	[5:15]
Exp	112,37	623,89	254,12	3,01
Pow	1,71	161,27	214,12	1,37
TP	**0,45**	**12,21**	**10,39**	**1,14**

How much simulation do we need?



- **Ideal:** amount of simulated data ≈ real data
  - **Number of events at LHC:** $\mathcal{O}(\cdot 10^8)$ **/ s**
  - **Necessary time for Monte Carlo with ALICE geometry:** $3.8\ ms/track$
- **Necessary time to simulate 1 s of ALICE data:** $\mathcal{O}(days)$
- **Precision scales with** $1/\sqrt{N}$
- **Probability decreases with the increasing „interestingness"**



W.J. Stirling, w.j.stirling[AT]imperial.ac.uk

We need time...

How much simulation do we need?

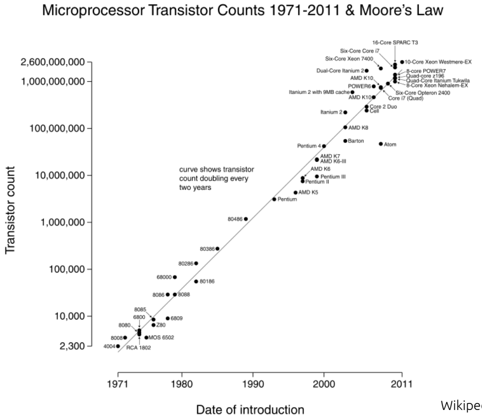
We need time...

http://www.kgionline.com/distributed-computing.jpg

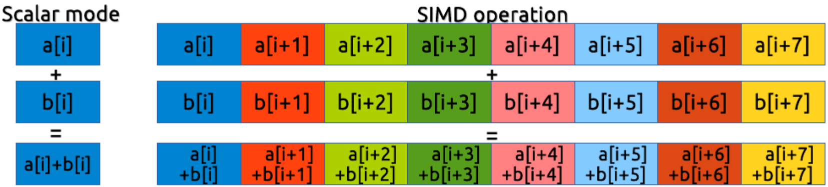
...or we need more CPUs...

How can it be faster?

- ...or we can exploit micro-parallelism!
- Moore's law: increasing number of transistors
- Vector micro-parallelism: SIMD (Single Instruction Multiple Data)
- Parallelisation in instruction level

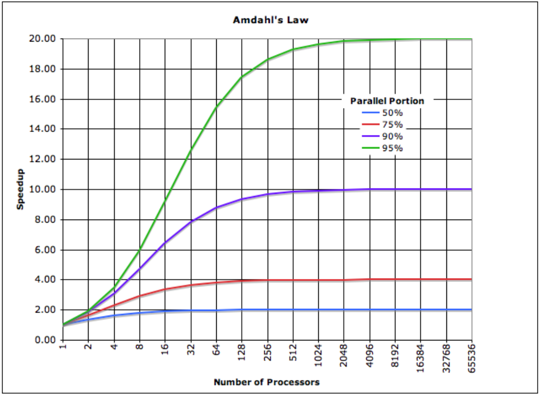


Wikipedia



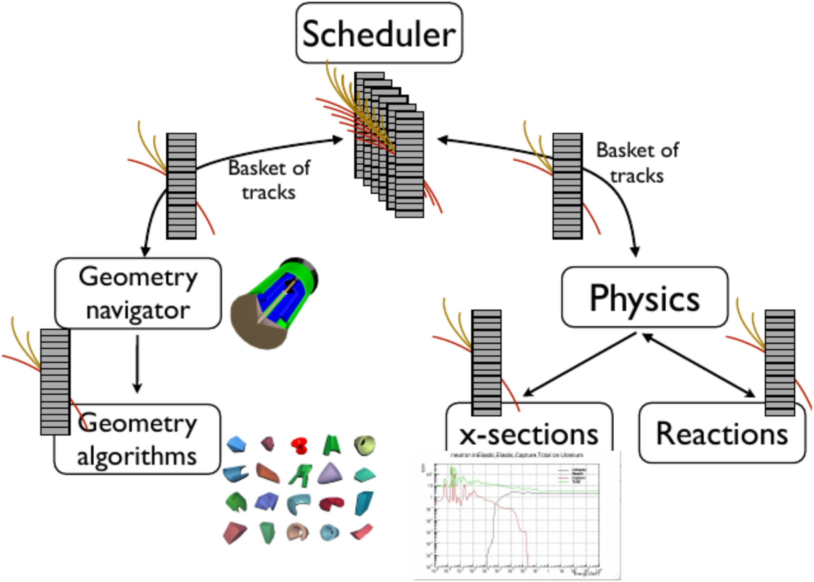
How can it be faster?

- Preliminary studies showed, that particle transport in most detector geometries is very much localized
  (2012 J. Phys.: Conf. Ser. 396 022014)
- GeantV Vector Prototype: new generation of particle transport simulation
  (2014 J. Phys.: Conf. Ser. 513 052006, 2014 J. Conf. Phys.: Conf Ser. 052013)
- Main goals: exploit the opportunities for parallel execution offered by the hardwere without losing portability and maintainability



Wikipedia

How can it be faster?

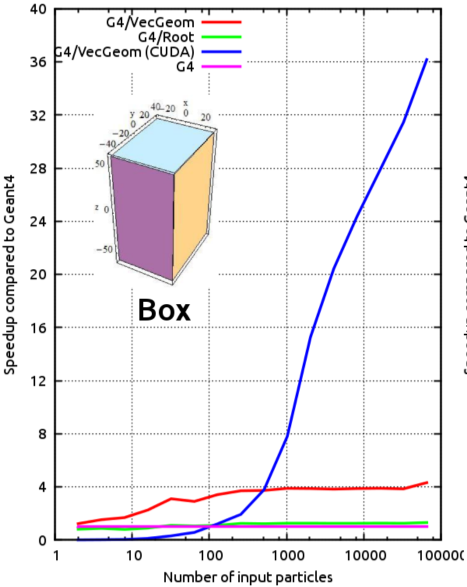


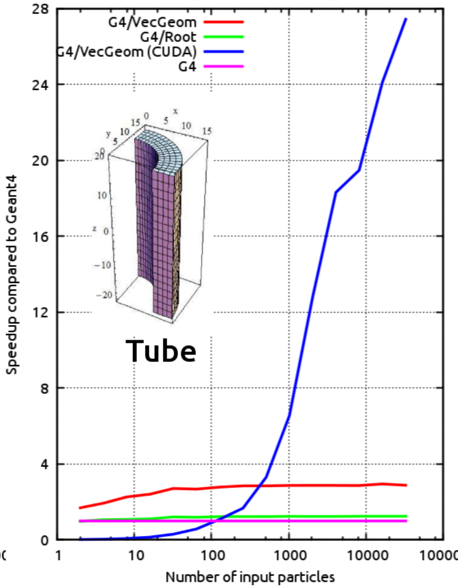
How can it be faster?

- VecGeom: the Vectorized Geometry for Geant-V
- Geometry calculations: calculate the distance for the closest/next volume boundary for a given particle
- Vectorized calculation: find the distances of vectors of particles, using SIMD operations
- High degree of portability thanks to trait classes
- One single code for scalar and vector context and for GPUs as well
  - SIMD instructions: using Vc vector library (http://code.compeng.uni-frankfurt.de/projects/vc)
  - GPU: through generic programming with C++ templates
    - Supports NVidia CUDA
    - OpenCL: work in progress...
  - Benchmarking elemetary shapes: calculate the distances of random points

How faster is it now?

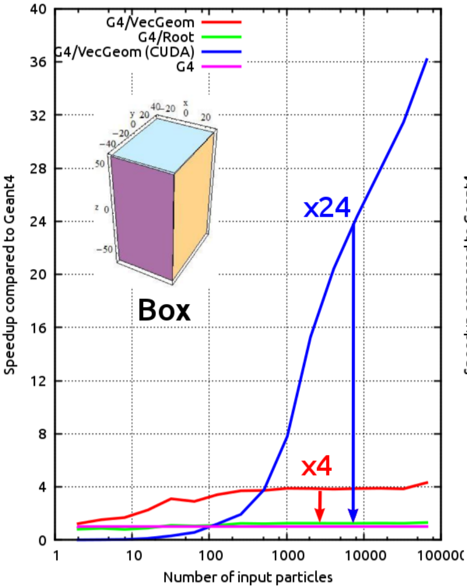
Distance To Out, 5000 Repetitions, Intel® Core™ i7-920 Processor + NVidia Tesla C2050

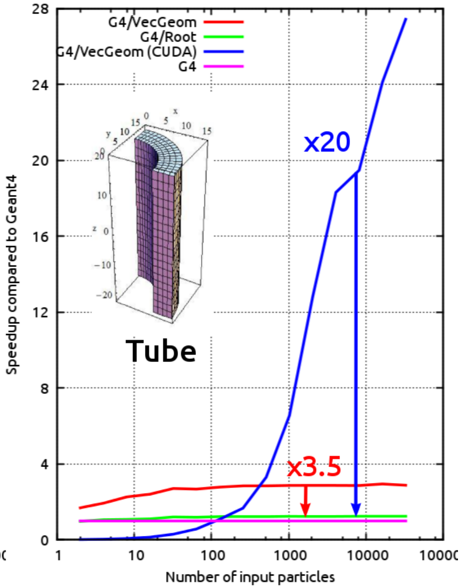




How faster is it now?

Distance To Out, 5000 Repetitions, Intel® Core™ i7-920 Processor + NVidia Tesla C2050





Summary

- There is a need to accelerate future detector simulations significantly
- Vectorized geometry code already shows $\sim 3.5x$ speedup with CPU and more with NVidia GPU
- More speedtests with complex geometries (CMS, ALICE) are in progress
- The speedup obtained with the geometry routines is promising, but there is still a lot of work to do (vectorized physics, scheduler)

Thank you for your attention!