Use of Geant4 to simulate BTeV ECAL testbeam setup

BTeV Collaboration

Julia Yarba
Fermilab (Batavia, USA)
Dmitri Morozov
Institute for High Energy Physics (Protvino, Russia)

Geant4 User's Workshop at SLAC

February 18-22, 2002

J.Yarba, D.Morozov for BTeV

Outline :

- Why G4 in BTeV ?
- Learning curve - was it a big deal ?
- G4 to simulate BTeV/ECAL test beam data
- Other tests of G4 in BTeV
- Future plans
- Conclusions

Geant4 User's Workshop at SLAC
February 18-22, 2002

Why G4 in BTeV ? :

- BTeV is a modern project
- Looking forward to benefit from employing modern technologies, including modern software
- Currently use G3.31 as modeling engine
- Consider G4 as a new simulation package on the market

DECIDING ON REPLACING MODELING ENGINE IS A SERIOUS COMMITTTMENT, REQUIRES EXTENSIVE EVALUTION WORK

- Split the task into several steps
- Start with a small-scale prototype

Learning Curve to G4 - Was it a Big Deal ?

Julia : Extensive experince with G3, several years of working experience with C++.

Dmitri : No experience with G3 at all, some experience with C++. Had no problems installing either debug or optimized version of G4 within days. Within less than a month was able to code in G4 BTeV/ECAL test beam geometry to be exposed with electron beam of various energies, debug things, and produce first results.

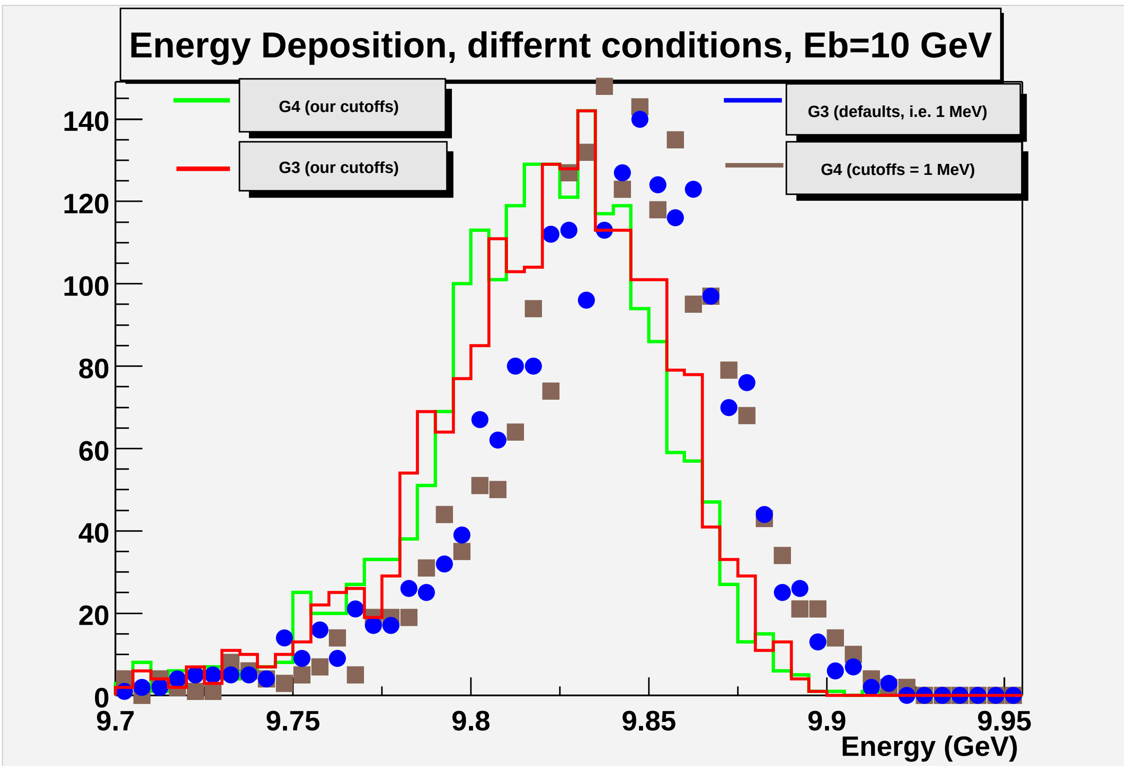
G4.3.2 : Simulation of BTeV/ECAL Test Beam Data

- Needed simulation for test beam anyway
- Made use of G4.3.2 part of test beam program because small-scale prototype is best to start with
- For "control points", had both G4.3.2 and G3.21 :
  - 5x5 matrix of Lead Tungstate crystals, each 2.7x2.7x22cm^3, wrapped in 40mk of teflon
  - Exposed the matrix with 10GeV electron, 2000 events/run
  - Production threshold = 1mm for all particles
  - Stopped tracing particles (via UserLimits/UserSpecialCuts, just like in example/novice/N05) :
    - 1MeV for γ, e^-, and e^+
    - 60KeV for γ, 0.5MeV for e^-/e^+

J.Yarba, D.Morozov for BTeV

G4.3.2 : Simulation of BTeV/ECAL Test Beam Data (cont.)

Energy Deposition in PbWO4 Simulated by G4.3.2 and G3.21



- "Our cutoffs" : tracing cuts = 60KeV for γ, 0.5MeV for e^-/e^+
- Recorded energy deposition changes with tracing cuts and is fairly independent of the package

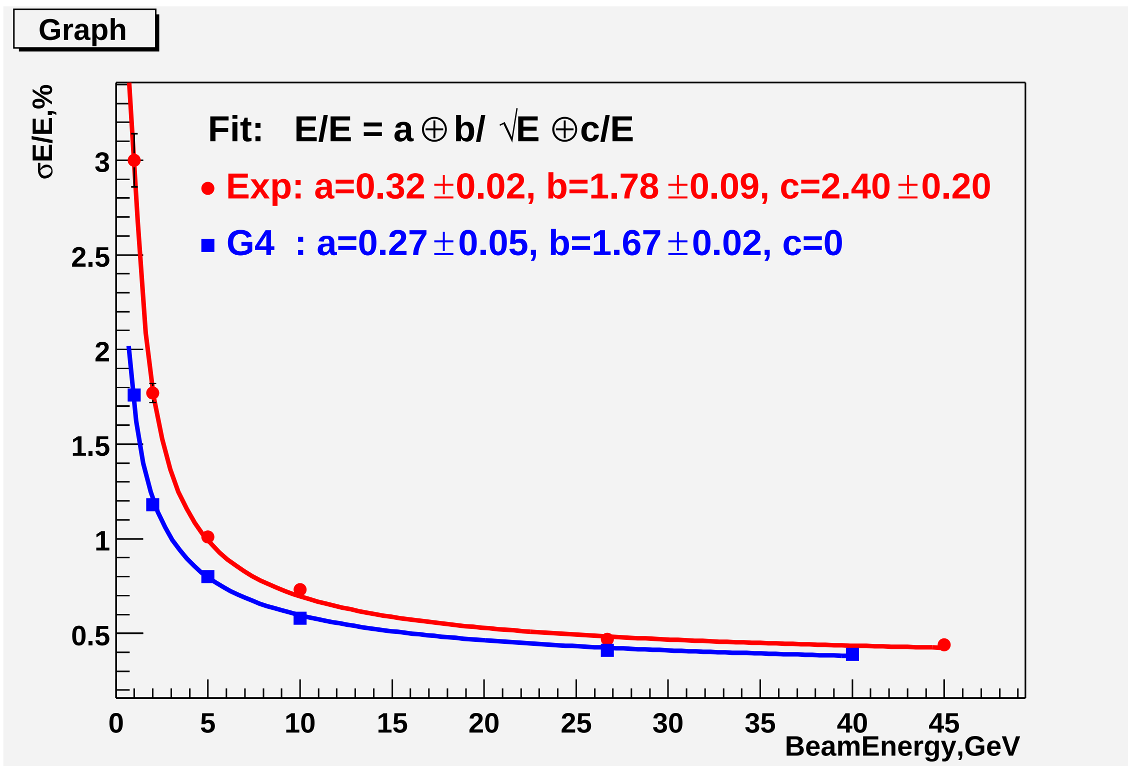
G4.3.2 : Simulation of BTeV/ECAL Test Beam Data (cont.)

- "Exposed" the matrix of PbWO4 crystals with e^- beam with energy in the range from 1GeV to 40GeV to obtain theoretical energy resolution
- Set production threshold at 1mm for all particles
- Set tracing cuts at 60KeV for γ, 0.5MeV for e^-/e^+
- To account for photostatistics variation, assumed light output to be 5 ph.e./MeV
- Did NOT account for errors from multiple scattering upstream the prototype
- Did NOT account for non-uniformity of light output along crystal

J.Yarba, D.Morozov for BTeV

G4.3.2 : Simulation of BTeV/ECAL Test Beam Data (cont.)

Energy Resolution Simulated by G4.3.2, Compared with Experimental Test Beam Data



- Stochastic terms in agreement
- Difference mostly due to errors from multiple scattering and non-uniformity of light output along crystal

G4.3.2 vs G3.21: Speed Concern

Speed Factor :

- 2.6 with 1MeV cut for γ, e^-, and e^+
- 2.3 with 60KeV cut for γ, 0.5MeV cut for e^-, e^+ (however, later realized that we were not tracing e^-/e^+ below 1.13MeV in G4, while in G3 we were tracing them above 0.5MeV...)

Are We Paying Twice as Much in Terms of CPU for Simulating the Same Detector Response ?

G4.3.2 vs G3.21: Speed Concern(cont.)

Further testing with a μ^- :

- Switched OFF all processes but transportation, hit the matrix of crystals on the front, on the side, under angle $\neq$ 90degrees : speed is exactly the same as of G3.21
- Turned ON multiple scattering :
  deviation from original (px,py,pz) is very similar to G3.21;
  speed is pretty much the same as of G3.21
- Tried to switch to switch other physics processes one by one to estimate changes in speed and physics, but ended up with added all of them at once (otherwise G4 crashes on segfault) :
  the speed factor became 6 and up vs G3.21 (depends on the energy of μ);
  the number of secondaries produced in G4 was much larger than in G3.21
- Learned from a colleague that all physics processes must added, then a user can switch OFF selected processes via user's interface, but did not have time to do the excersise

G4.3.2 vs G3.21: Speed Concern(cont.)

Looking at the number of secondaries produced in EM showers in G3.21 and G4.3.2 and the number of steps made by each particle :

- Used 1mm production threshold, 1MeV tracing cuts for γ, e^-, and e^+
- Noticed that G4.3.2 generates much more secondaries than G3.21
- Noticed much more neutrals making 1-2 steps in G4.4.3 than in G3.21
- Realized that 1mm production cut means slightly above 1MeV for e^- and e^+ but much lower for γ;
  knew that in G3.21 tracing cut applies at the begining of step and in G4.3.2 - at the end of step
- Tried to adjust production threshold for γ (about 9.5cm) :
  noticed the speed improve (factor 2.25 vs G3.21 instead of 2.6)
  BUT !!! also noticed that recorded energy deposition in the matrix of crystals went down instead of going up

G4.3.2 vs G3.21: Speed Concern(cont.)

- Knew that recording of deposited energy happened via SensitiveDetector;
  looked at the stepping loop (G4SteppingManager(2).cc) and realized that SD was called for AlongStep and PostStep processes but could not see it called for AtRest processes;
  since had e+ annihilation defined as AtRest process, following the examples, suspected that certain number of γs produced via annihilation might have been "lost"

J.Yarba, D.Morozov for BTeV

G4.4.0 vs G4.3.2 vs G3.21: Another Speed Concern

- Recently gave a try to G4.4.0 (patch with bug fixes included)
- Used production threshold 1mm for all particles, tracing cut 1MeV for all particles
- Realized that the the output in terms of recorded energy deposition in the matrix of PbWO4 crystals remained essencially the same but the speed dropped by factor 1.85 vs G4.3.2, which makes factor of almost 5 vs G3.21
- Heard concerns about CLHEP 1.7 (required for G4.4.); rebuilt G4.3.2 with CLHEP 1.7 and saw the speed drop by about factor 1.25 (but NOT 1.85 !)
- CLHEP people at Fermilab are working on improving the package; a few days ago tried G4.3.2 built with "local" version of CLHEP 1.7 which included a number of fixes from Mark Fishler and saw the speed improved by about 10% - it's to say that about a half of the loss has been gained back; hope for more progress in CLHEP

G4 vs G3.21: Speed Concern(cont.)

- Are we using G4 in a wrong way ?
- Any tricks about geometry/cuts/tracing ?
- Any issues with simulating physics processes ?
- Any issues with other packages G4 uses ?

BTeV hopes to get answers from G4 team...

J.Yarba, D.Morozov for BTeV

G4: Tracking in a Magnetic Field

- Stefano Magni, Daniela Lunesu of INFN-Milan
- Toy model of BTeV vertex detector : planes in a box; B=constant.
- Turn off all physics processes.
- Trace out and back. Compare intersections with each plane out and back. Compare with analytical calculation.
  - G4 out of the box:
    Errors of many microns; unacceptable for BTeV.
    Worrisome !
  - There must be adjustable parameters : we are working on finding the best set of parameters; expect final results on precision and timing soon
  - Adjustable parameters need documentation !
- Will test with $B \neq$ constant.

Future Plans :

- Continue G4 evaluation with small-scale prototypes
- Ongoing test beam program for every subdetector; plan to compare test beam data vs G4 output, for package validation as well for for self-education
  - Pixels : plan to use G4 to simulate data
  - Calorimetry : Compare G4 vs existing π^- data
  - Other Subsystems : Compare G4 vs data, once available
- If feel confident after accomplishing all of the above, we will implement full-scale BTeV geometry in G4, for further testing

Conclusions :

- BTeV is currently using G3.21
- G4 evaluation for use in BTeV is work in progress
- We appreciate the experience
- Along with positive news, we get concerns : Speed, Precision, Validity of Physics Processes
- We expect progress in future releases and we will continue with our tests