Detector Simulation in Particle Physics Collider Experiments – Impact and Future

V. Daniel Elvira

Special Lecture at the 1st National Workshop on Geant4 and its Application to HEP and Astrophysics

IUCAA, Pune, India, December 5th, 2022

Motivation

Throughout my career, I received many requests for material showing concrete examples on how detector simulation helps modern particle physics experiments

One of the times, John Harvey, former leader of the Software Group (SFT) at CERN, encouraged me to write a note on the topic

The note found its way to Physics Reports where it was published as a review paper:

- "Impact of detector simulation in particle physics collider Experiments", Phys. Rep. 695 (2017) 1–54

This presentation follows closely the material included in the paper and incorporates material on detector simulation software evolution

[Purposely, plots and numbers were not updated too much since publication – highlight the impact of detector simulation during startup and through the first run of the CERN LHC program]

Outline

Detector simulation is of critical importance to the success of HEP experimental programs, a determinant factor for faster delivery of outstanding physics results[0]

- Introduction
  - History, facts and numbers, modeling tracks and showers, the simulation software chain
- Detector simulation tools
  - Types of simulation, the Geant4 toolkit, physics validation
- Applications of detector simulation to HEP collider experiments
  - Simulation in data analysis, detector design & optimization, software & computing design, testing
- Modeling of particle and event properties and kinematics
  - Geometry and material effects, examples for different final states
- Economic impact and cost of simulation in HEP experiments
- Recent and current R&D projects
  - Detector simulation software in the era of heterogeneous computing
- Summary

Introduction

History, facts and numbers, simulation software tools and applications

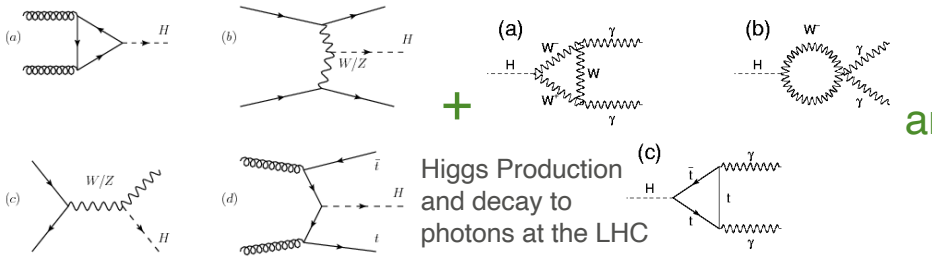
Some history

Accurate computer simulation is essential to design, build, and commission the highly complex detectors in modern HEP experiments, and to analyze & interpret their data

- Old times detector simulation
  - Simple analytic calculations, back-of-the-envelope estimates
- Era of detailed detector simulation started in late 70's early 80's
  - Electron Gamma Shower (EGS[1]), GEometry ANd Tracking (GEANT) software
- GEANT3[2] software kit to describe complex geometry, propagate particles and model interactions as they traverse different materials and EM fields
  - GEANT3 widely used by CERN, DESY, FNAL experiments. First OPAL (LEP), then L3 and ALEPH, followed by experiments at DESY and FNAL in the 90's
- Other simulation tools are FLUKA[3] and MARS[4]
- Geant4[5,6] used by most HEP experiments – limited initially, the norm in 21st century

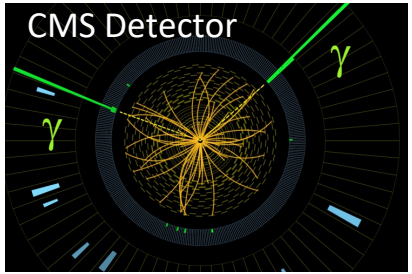
Why to simulate detectors

- Save time and money, improve the quality and accuracy of physics measurements
  - Design optimal detector, best physics at a given cost, even before fastening the first screw!
- Simulation is not magic
  - Particles cannot be “discovered” in a simulated sample which does not model them
- Simulation is essential to HEP experiments
  - Teaches physicists what mark a new particle would leave in the detector if it existed



Higgs Production and decay to photons at the LHC

and



➔ Higgs discovered in July 2012

SM Higgs prediction:
Higgs is produced at the LHC and decays to two γ’s with given properties for the event and the individual particles

Observation:
two photon events with predicted detector marks are observed

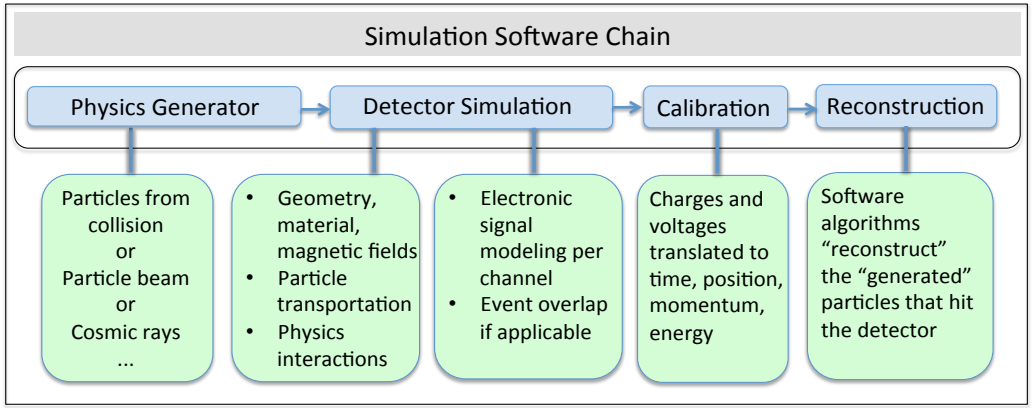
Facts and numbers

The role of detailed detector simulation in HEP experiments has grown very significantly during the last three decades. Detector simulation is now a crucial component

- LHC experiments simulate events at a speed and with physics accuracy never seen before
  - ATLAS/CMS: seconds to minutes per event, tens of billions of events since 2010
  - CDF/D0 (early 1990's): hundreds of thousands of, in comparison, poor-quality events
- Geant4-based simulation has shortened the time between data-taking and journal submission of increasingly precise physics results at the LHC
  - Other factors being detector and computing technology, a wealth of experience from pre-LHC experiments, better calibration and analysis techniques, communication tools, etc.
- In most experiments, detector simulation has taken ~ 1/2 of all computing resources
- Over the next two decades, detector simulation applications need to deliver orders of magnitude more events with increased physics accuracy and with a flat budget

A daunting challenge for detector simulation tools

Simulation software chain in a typical HEP experiment

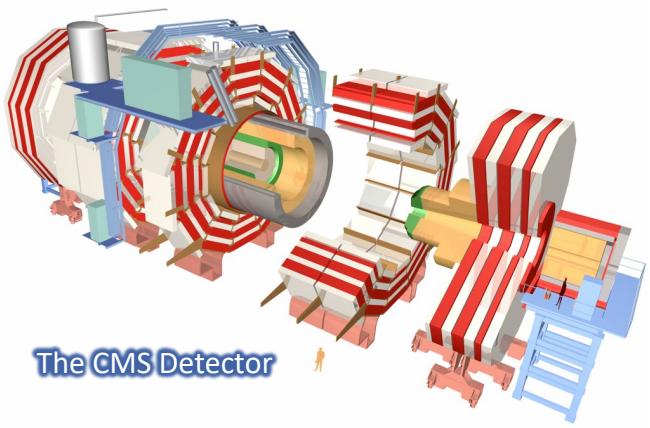


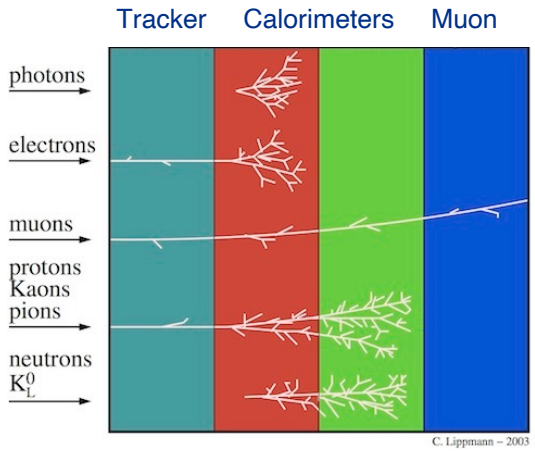
Simulation referred to as "Monte Carlo (MC) simulation"

Simulated events referred to as "MC events, or MC samples"

- Physics generator: provides the final states of the physics process of interest (Pythia, Herwig, Madgraph, Alpgen, etc. in colliders; GENIE, etc. for neutrinos)
- Detector simulation [focus of this presentation]:
  - First stage: passage of generated particles through detector material and magnetic fields
  - Second stage: detector electronics, backgrounds to collision of interest (pileup)
- Calibration: from detector quantities to physics quantities
- Event reconstruction: algorithms, typically the same, applied to real data

Particles through a collider detector: tracks and showers





Tracker measures charged particle trajectories

Calorimeter measures particle energy

Muon system, combined with tracker, measures muon trajectories

(Physics processes: energy loss, multiple scattering,…, etc. “Showers” of secondary particles produced through EM and nuclear interactions)

Charges and voltages recorded in millions of detector channels ➔ **x**, **p**, E, time measurements

Particle tracks and particle showers must be modeled accurately

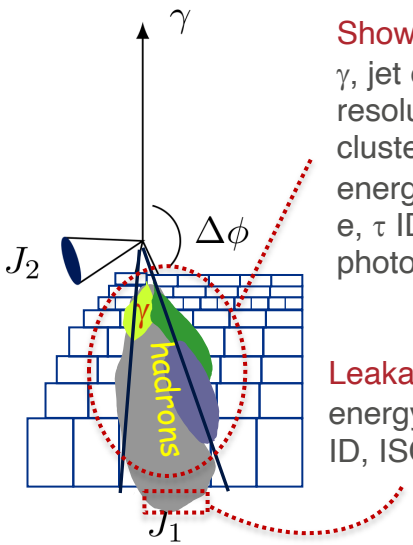
Shower modeling affects physics predictions – examples

The accuracy of the modeling of particle showers in calorimeters

(particle types and multiplicity, E and η, ϕ distribution, E response linearity and fluctuations)

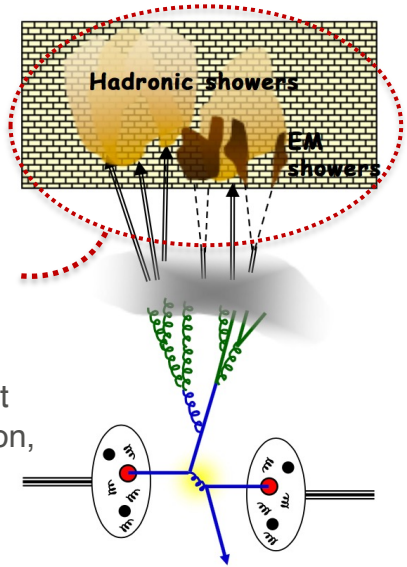
affects the degree of data-to-MC agreement for

– physics object variables, lepton identification (ID) and isolation (ISO) efficiency, etc



Shower mis-modeling affects e, γ, jet energy response and resolutions, jet multiplicity, un-clustered and out-of-cone energy in jet reconstruction, γ, e, τ ID and ISO efficiencies, di-photon and di-lepton separation

Leakage mis-modeling affects jet energy response, μ reconstruction, ID, ISO efficiencies



Impact on physics predictions:

Backgrounds with multi-jets (QCD), and leptons (EWK)

W, Z, top, Higgs mass

QCD cross sections, jet shapes, sensitivity to soft radiation

Detector simulation tools

Types of simulation, the Geant4 toolkit, physics validation

Types of simulation: toy, parametrized, full

- Toy simulation – a few simple analytical equations without a detailed geometry/field description or particle shower development
  - Zeroth order detector or physics studies
    - Output data format may not be the same as real data's , speed is a small fraction of a second/event
- Parametrized simulation – approximate geometry/field description, parametrized energy response and resolution, shower shapes
  - Computing intensive MC campaigns that would otherwise be prohibitive, i.e. parameter space scanning in BSM signal samples
    - Examples are the CDF QFL simulation (1990's) and CMS Fast Simulation framework which are tuned to test beam data, single tracks and/or full simulation
    - Output data format is typically identical to real data's, speed is of the order of a second/event

Types of simulation: toy, parametrized, full

- Full simulation – based on Geant, FLUKA, MARS with detailed geometry/field and shower description, the latter based on individual particle interactions
  - Detector and physics studies where geometry and physics accuracy are important
    - Output format same as real data's, speed is of the order of seconds to minutes per event

Full versus Fast simulation – misleading concept

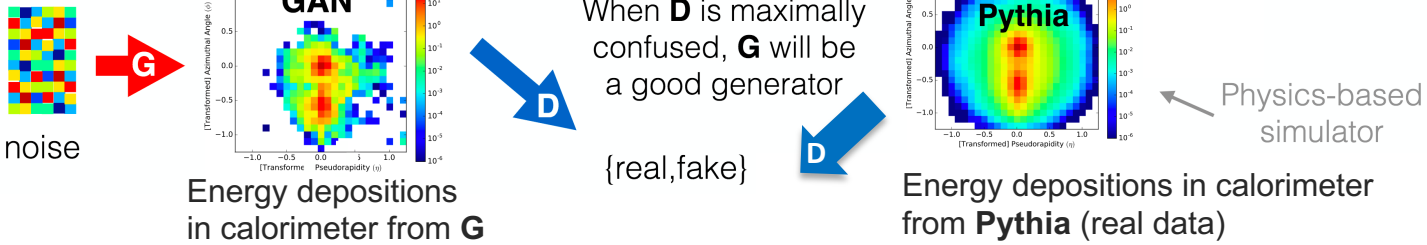
Experiments are moving towards simulation frameworks with flexibility to incorporate "fast simulation techniques" to a base Geant4 application

Tabulation, shower libraries, parametrization a la GFLASH, Machine Learning

Machine learning in detector simulation – GAN example

A Generative Adversarial Network (GAN) may be used to model particle showers

- Two Neural Networks are used (NN): Discriminator (**D**) and Generator (**G**)
- **G** takes random noise as input, initially
- **D** is a simple classifier to distinguish real data from data created by **G**
- **D** is trained from two sources: real data from physics generators (e.g., Pythia) and fake data created by **G** (noise)
- **D** classifies data received from **G** as real or fake and penalizes **G** for producing fake data
- As **G**'s output improves, **D**'s performance gets worse (cannot distinguish real data from **G**'s data)
- **G** converges into a good generator when **D** is maximally confused



Energy depositions in calorimeter from **G**

Energy depositions in calorimeter from **Pythia** (real data)

The Geant4 simulation toolkit

At the core of most full simulation applications at modern collider experiments, e.g., LHC, is the Geant4 toolkit

- International Collaboration of tens of institutions and ~120 physicists and computer professionals
- Written in OO C++, > 1 million lines of code, > 2000 C++ classes
- Used by almost all HEP experiments (10,000 users), space, and medical applications

24th G4 Collab. meeting in Jefferson Lab, USA 2019

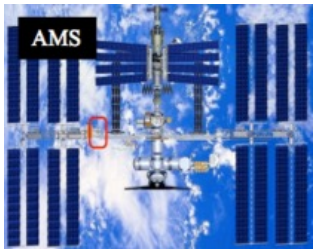
27th G4 Collab. Meeting in Rennes, France (2022)

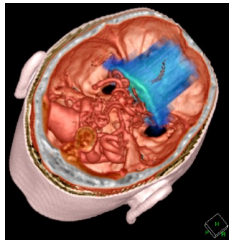
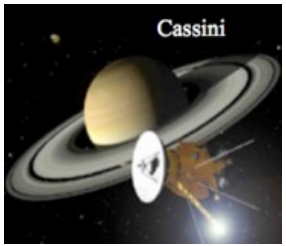
The Geant4 simulation toolkit

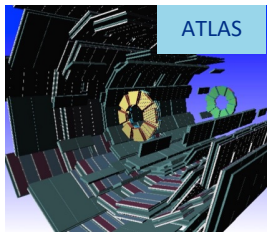
The impressive success of the Geant4-based simulation applications at the LHC experiments is the result of:

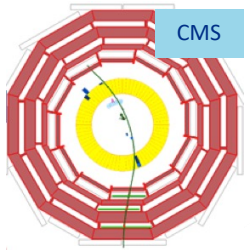
- Many years of hard work, partnership between the experiments and the Geant4 team
- A process to develop, optimize, and validate the many Geant4 physics models
- Different fora served as vehicles of communication, discussion, and information exchange

The use of Geant4 has extended to include high-energy, nuclear and accelerator physics, as well as medical science and treatment, and space exploration.



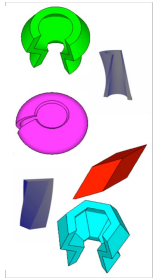




A Geant4-based simulation application

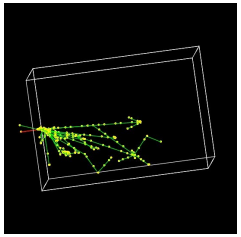
Experiments develop a ”simulation application” (software package) for their detector using Geant4 by assembling each of the following elements:

Detector geometry (shapes and materials)



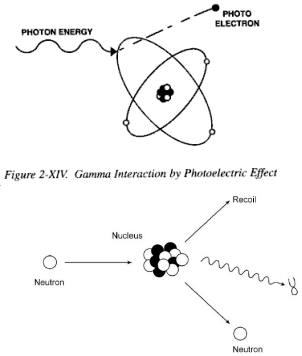
+

Particle Propagation through geometry and EM fields



+

Physics Processes



The user selects:

- Method of integration of the equation of motion, particle tracking parameters
- “Physics Lists” composed of a subset of the physics models available to describe the interaction of particles with matter for energy between 250 eV and ~100 TeV

Output is a collection of “particle trajectories” and “simulated hits” with position, time, and energy deposited in detector volumes

Validation of the detector simulation physics

A collaborative task involving the Geant4 developers and the experiments

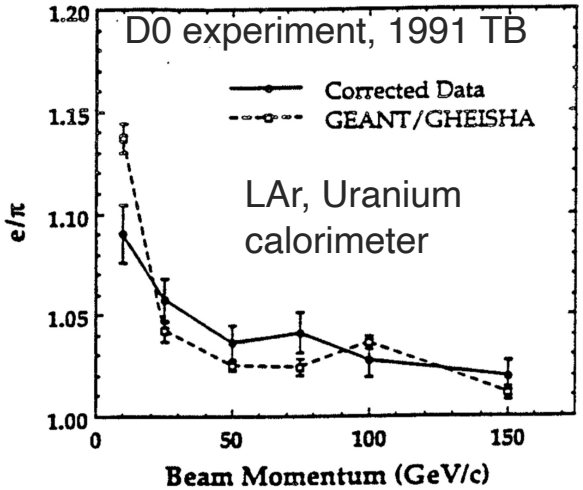
- Thin-target experiments
  - Beams of particles of different types (typically e, π, p) are directed onto thin targets made of materials typically used in HEP experiments (Be, C, Cu, Pb, Fe, etc.)
  - Measure cross sections, angular distributions, particle multiplicities
  - Examples: CALICE, HARP, NA49, NA61
  - Used by the G4 team to validate individual (G4) models at the single-interaction level
- HEP experiments
  - Collider, neutrino, muon experimental data, as well as their associated test beam results are compared to predictions from their Geant4-based simulation applications
  - Quantities are typically energy response functions, shower shapes

These two sets of data are complementary: thin-targets for “first principles” G4 models tuning, HEP experiments for confirmation or small tweaks to the models

Simulation physics validation: HEP experiments – test beams

Collider experiments run test beam (TB) campaigns, used to select among physics lists, guide the G4 team on how to assemble them from individual models

Early times: D0, CDF Experiments (Tevatron, early 90's)



Slow computers – Geant3 full simulation took O(hour/event), limited TB programs, deficient communication technology

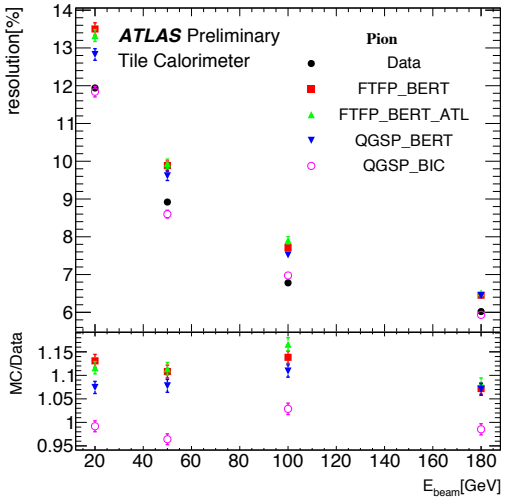
- Low statistics MC samples
- Approximations in exchange for time performance
  - D0: simplified geometry, average materials, shower truncation, full Geant3 simulation only for some analyses
  - CDF: use of parametrized simulation (QFL) tuned to minbias/TB data, then Geant3+GFLASH shower parametrizations

e/π response ratio: large statistical uncertainties in GEANT3 prediction (negligible in CMS)

Limited energy range 10-150 GeV, difficult to evaluate high energy region (2-300 GeV in CMS)

Simulation physics validation: HEP experiments – test beams

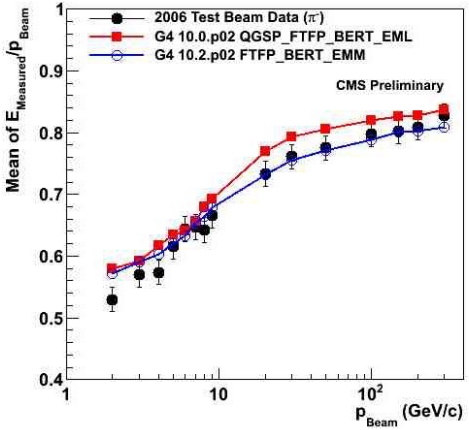
ATLAS 2002–2003 TB



Calorimeter π energy resolution (%) vs. beam energy

- Stat errors only
- MC/data ~ 1.00 -1.15

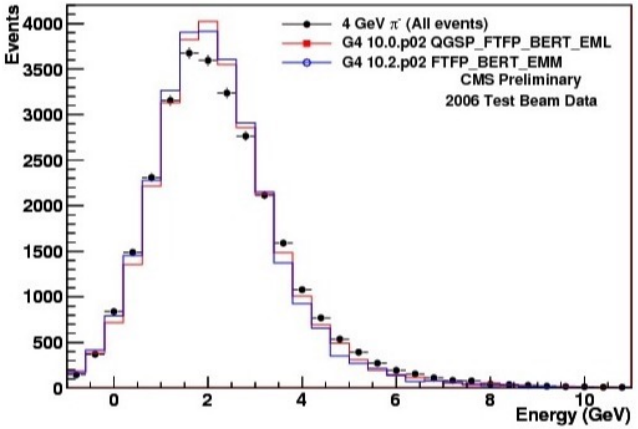
CMS 2006 TB



Calorimeter π energy response vs. beam energy

- Excellent agreement within statistical uncertainties
- MC overestimate trend below ~5 GeV

CMS 2006 TB



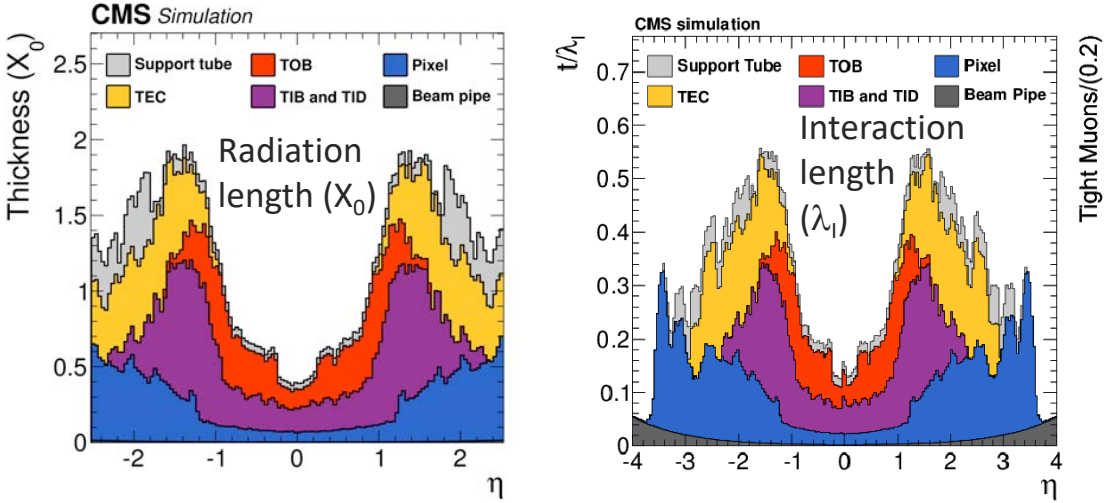
Calorimeter response function

- Good modeling of core and tails critical for jet and E_T^{miss} modeling (jet cross sections and QCD background to BSM measurements)

Note small/negligible statistical errors in simulation

Simulation physics validation: HEP experiments – material

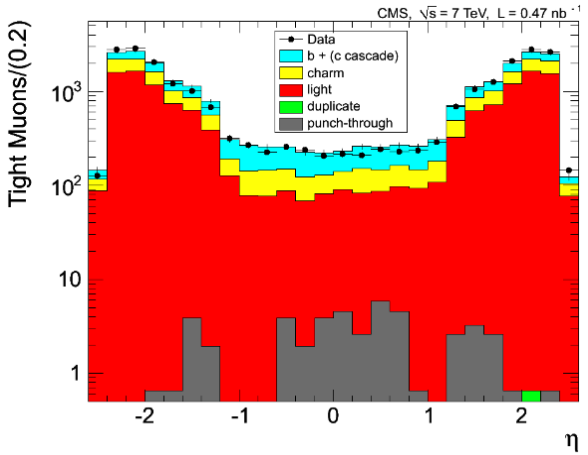
Data from collider runs used for final validation of full simulation application



Thickness of CMS silicon tracker from simulation

- Mis-modeling affects energy loss of charged particles, photon conversion (70% in CMS tracker)

Validated by weighing components of real detector

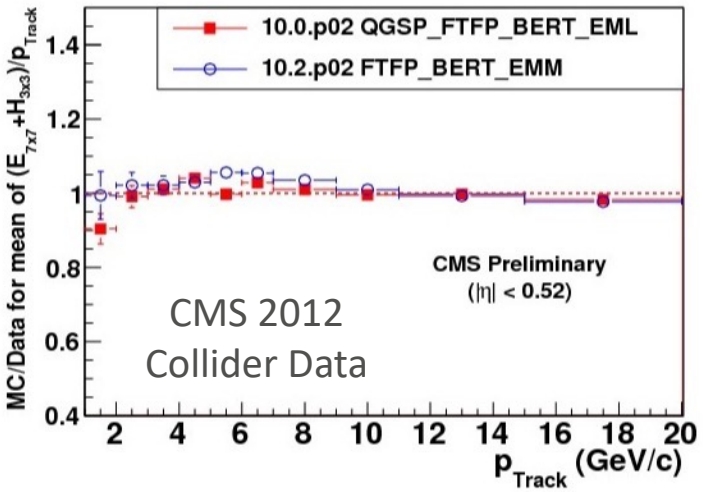


CMS inclusive μ sample (zero-bias)

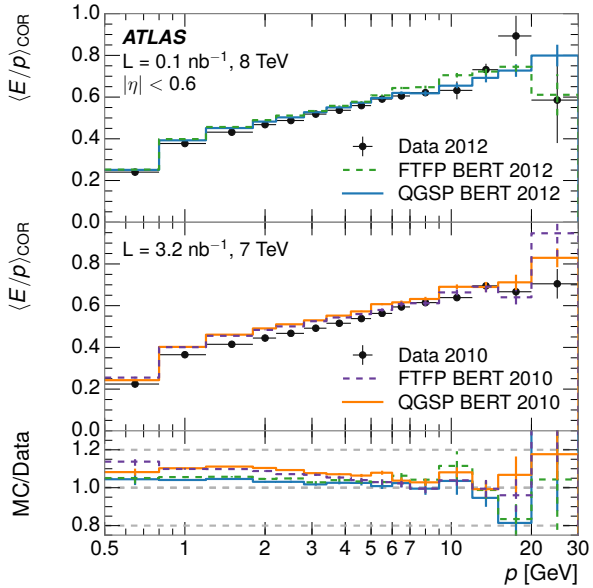
- All sub-detectors used in muon reconstruction (different materials, technologies)

Excellent agreement!

Simulation physics validation: HEP experiments – physics runs



ATLAS 2010/2012 Collider Data →



MC-to-data ratio: calorimeter energy / tracker momentum (single tracks, minbias samples)

- Demonstrates excellent modeling of hadron energy response linearity after calibration, using two independent measurements: calorimeter energy and tracker momentum

MC models data within < 5 % above 0.5 (1) GeV for ATLAS (CMS)

Applications of detector simulation to HEP Collider experiments

Data analysis, detector design and optimization, software & computing design, development and testing

Applications of simulation to data analysis

A few examples of applications to data analysis and interpretation:

- Data-driven methods
  - Techniques applied to real collider data to measure ***physics backgrounds, calibration & alignment factors, resolutions, identification & reconstruction efficiencies, fake rates***, etc.
  - Based on detector properties, conservation laws, mathematical tools and analysis
    - Applied to detector-level data and detector-level simulated data as if it were real data
- Closure tests
  - Verify data-driven measurements are correct within the quoted uncertainties
    - Comparing detector level MC measurement with MC truth information

      $T = (MC^{\text{reco-level}} - MC^{\text{truth}}) / MC^{\text{truth}} \sim 0$ within the uncertainty of the method
- Modeling of signal samples
  - SM precision measurements (e.g., top, W/Z/Higgs), Beyond the Standard Model (BSM) searches
  - Fast simulation to scan large theory parameter space (e.g., SUSY)

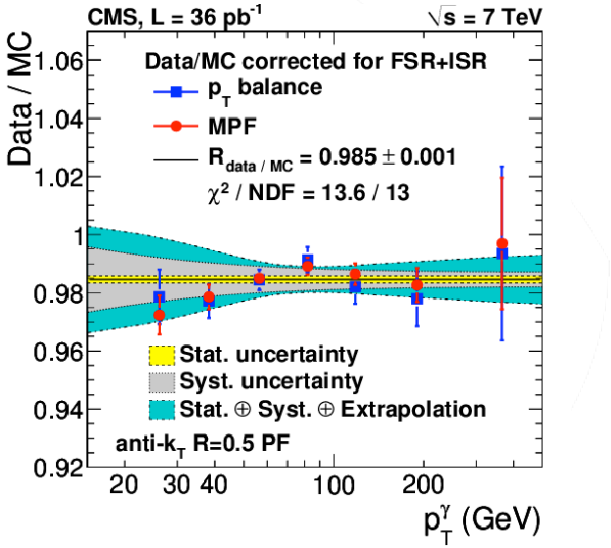
Applications of simulation to data analysis – data-driven methods

Corrections in data analysis mostly from MC truth with small ”scale factors (SF)”

- SF calculated as ratio of data-driven measurements in detector-level collider data and MC
- The trick is that systematic uncertainties “cancel” in the SF ratio – same method!

- Jet energy response (R_{jet}) or “ jet energy scale” (JES)
  - $R_{jet}^{\text{truth-MC}} = p_T^{\text{jet reco-MC}}/p_T^{\text{jet particle-level-MC}}$
  - Data-driven methods use di-object p_T balance: multijet, γ+jets, Z+jets samples (conservation laws)
  - $R_{jet} \sim p_T^{jet}/p_T^{\gamma,Z}$ and $SF=R_{jet}^{\text{reco-data}}/R_{jet}^{\text{reco-MC}}$

$JES = R_{jet}^{\text{truth-MC}} \times SF$, with SF ~ 0.98 +/- 1-2%



Accuracy improves as SF→1 within a small uncertainty – excellent MC modeling of the data

Applications of simulation to data analysis – closure tests

Data-driven methods need to be demonstrated with “closure tests” (T)

- Lack of closure (T≠ 0, outside error band)
  - Indicates the need to go back to the drawing board and understand biases in the procedure excellent MC modeling needed!
- Limitations of simulation at D0 (early 1990’s)
  - **Geant3:** approximate geometry, average material, partial validation of response linearity with data, showers at 95% of total energy deposited (soft contributions, out-of-cone effects missed)
  - **Parametrized “a la CDF” simulation not viable**: no central magnetic field until 2001 ⟹ no single particle response measurement for response tuning

Cause of delay in a number of physics measurements

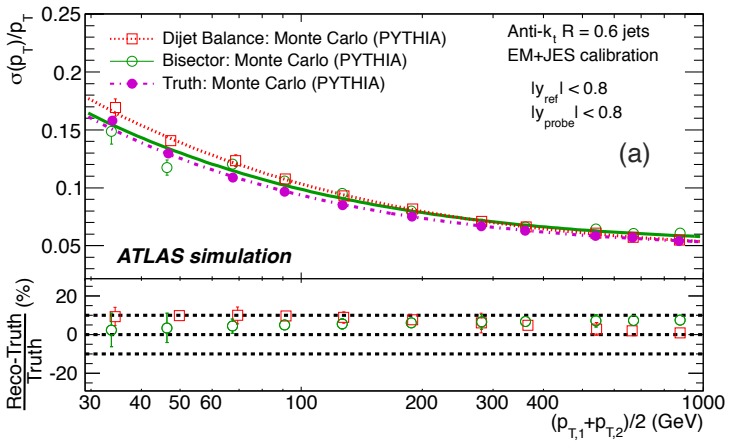
Jet cross sections and other QCD measurements –delayed 1992 ⇒ 2000 until JES error ≤ 3%

(Lack of large/accurate MC samples to demonstrate data-driven methods by closure for JES)

Applications of simulation to data analysis – closure tests

- Verify data-driven methods are accurate within quoted uncertainties

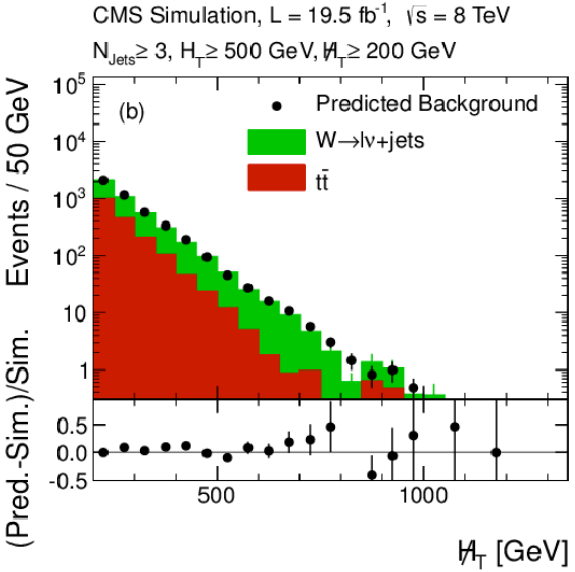
  T = [(data-driven prediction) – (MC truth value)] / (MC truth value)



Jet energy Resolution

MC data-driven prediction from dijet asymmetry:
A = ($p_T^{jet\ 1}$-$p_T^{jet\ 2}$) / =($p_T^{jet\ 1}$+$p_T^{jet\ 2}$)

Method closes within < 5%



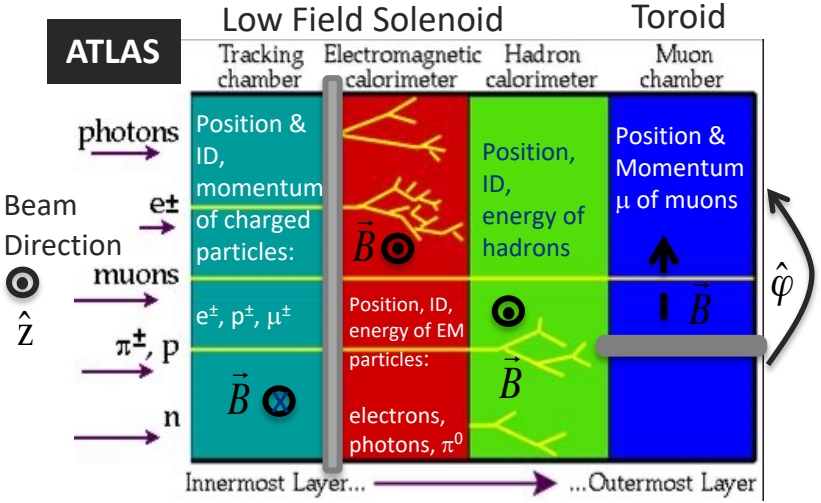
MC data-driven prediction from inclusive lν sample after analysis cuts and lepton efficiency corrections

Closure within stat. errors

$W \to e/\mu\,\nu$ and $t\bar{t}$ backgrounds to multijet + H_T^{miss} Supersymmetry (SUSY) search

Simulation in detector design and optimization

To design a HEP detector, different components, sizes, and are modeled and optimized in simulation for best physics performance



Tracker (in Si detector) optimized varying pixel and strip density, number of layers, angular coverage, amount of material

Calorimeter optimized varying angular coverage and hermeticity, transverse granularity, longitudinal segmentation, materials

Muon system optimized varying wire chamber density, number of layers in the radial direction, angular coverage.

More powerful or weaker magnets allow for more compact (CMS) or larger (ATLAS) detector designs

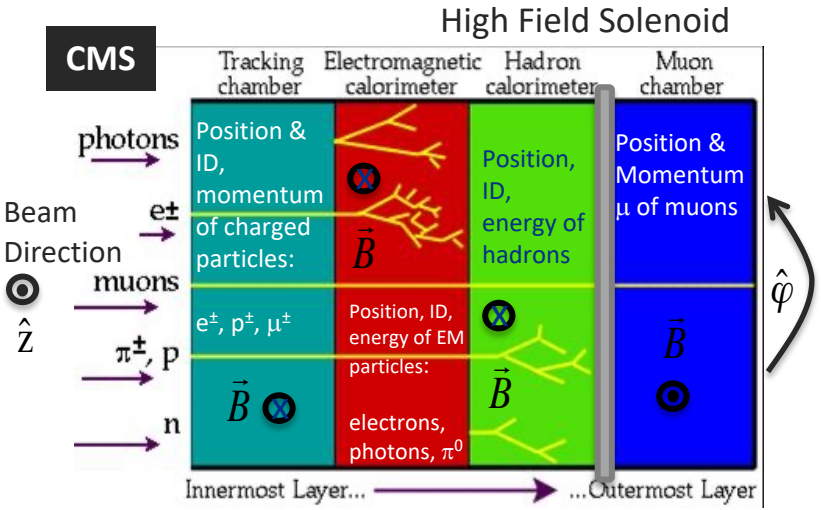
MC campaigns consist of millions of events generated with different detector scenarios

- Make the case for a design, optimize parameters for best physics, impact of de-scoping

(Interesting: detector configurations also adapt to play to the strengths of the Geant4 simulation toolkit)

Simulation in detector design and optimization

To design a HEP detector, different components, sizes, and are modeled and optimized in simulation for best physics performance



Tracker (in Si detector) optimized varying pixel and strip density, number of layers, angular coverage, amount of material

Calorimeter optimized varying angular coverage and hermeticity, transverse granularity, longitudinal segmentation, materials

Muon system optimized varying wire chamber density, number of layers in the radial direction, angular coverage.

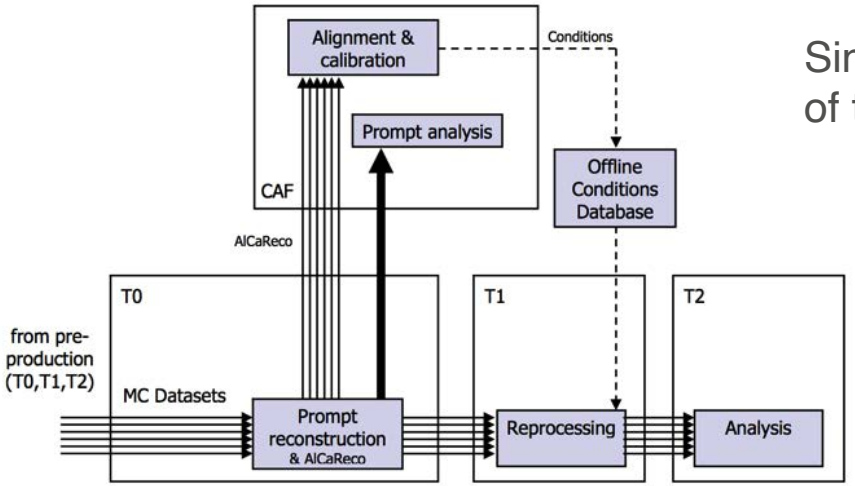
More powerful or weaker magnets allow for more compact (CMS) or larger (ATLAS) detector designs

MC campaigns consist of millions of events generated with different detector scenarios

- Make the case for a design, optimize parameters for best physics, impact of de-scoping

(Interesting: detector configurations also adapt to play to the strengths of the Geant4 simulation toolkit)

Simulation in software and computing design and testing



Simulation is essential to develop each element of the workflow and data flow for data handling

- Worldwide LHC Computing Grid (WLCG) divided in four tiers: 0, 1, 2, 3
- Each tier performs difference services: acquisition, reconstruction, simulation, storage, data analysis

Combined procedure tested in Computing, Software, and analysis challenges (CSA) in CMS

- System stress tested at 25%, 50%, and 75% capacity in 2006, 2007, and 2008
- 150 million events simulated, trigger rates modeled, and data reconstructed, skimmed, calibrated
- Data transfers between centers, monitoring of event file size, memory and CPU consumpti

The realism of these tests resulted in computing systems performing as predicted

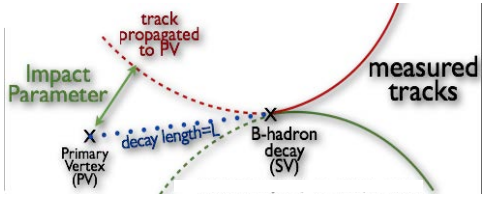
Modeling of particle and event properties and kinematics

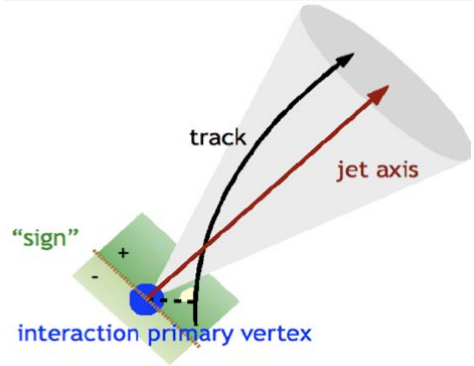
Tagging of heavy quarks, W, Z, and photon event distributions, missing transverse energy distributions

Modeling of particles and event properties: b jets

Modeling of b-jet reconstruction/identification is a critical simulation benchmark

- SM measurements: top decays to b, W and flavor tied to EWSB mechanism
- BSM searches: SUSY and EWSB related through hierarchy problem





b-jet identification (b-tagging) depends on impact parameter of charged-tracks and reconstructed decay vertices in the jet, lepton presence

Impact parameter (IP) is the point of closest approach between the track and the primary vertex

- b-quarks have positive IP while light jets have IP ~ 0
  - Resolution effects give positive and negative values in a real detector

Good simulation of IP variables is necessary for accurate measurement of b-tagging efficiencies and fake rates using data-driven methods

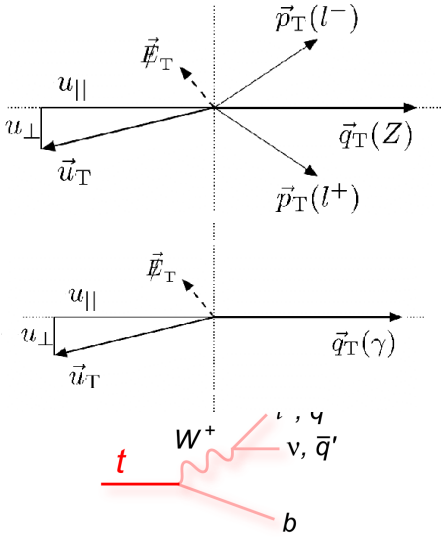
- Derived from data-driven methods applied to samples of jets with a muon

Need excellent modeling of material budget, energy loss, ionization, multiple scattering, noise, pileup mainly in tracker

Modeling of particles and event properties: W, Z, photons

Gauge bosons are at the core of SM measurements (W, Z, top mass and properties) and contribute backgrounds to most BSM searches

- Topologies and kinematics of W/Z/γ + jets events must be modeled with high accuracy
  - Generators are limiting factor for accuracy, particularly in multi-jet events with heavy flavor



W/Z + jets background typically estimated from data in BSM searches

- Simulation is used to study (di-)lepton + jets control samples to design data-driven methods (distribution shapes, variable correlation, etc)

MC truth used to predict sub-dominant SM backgrounds

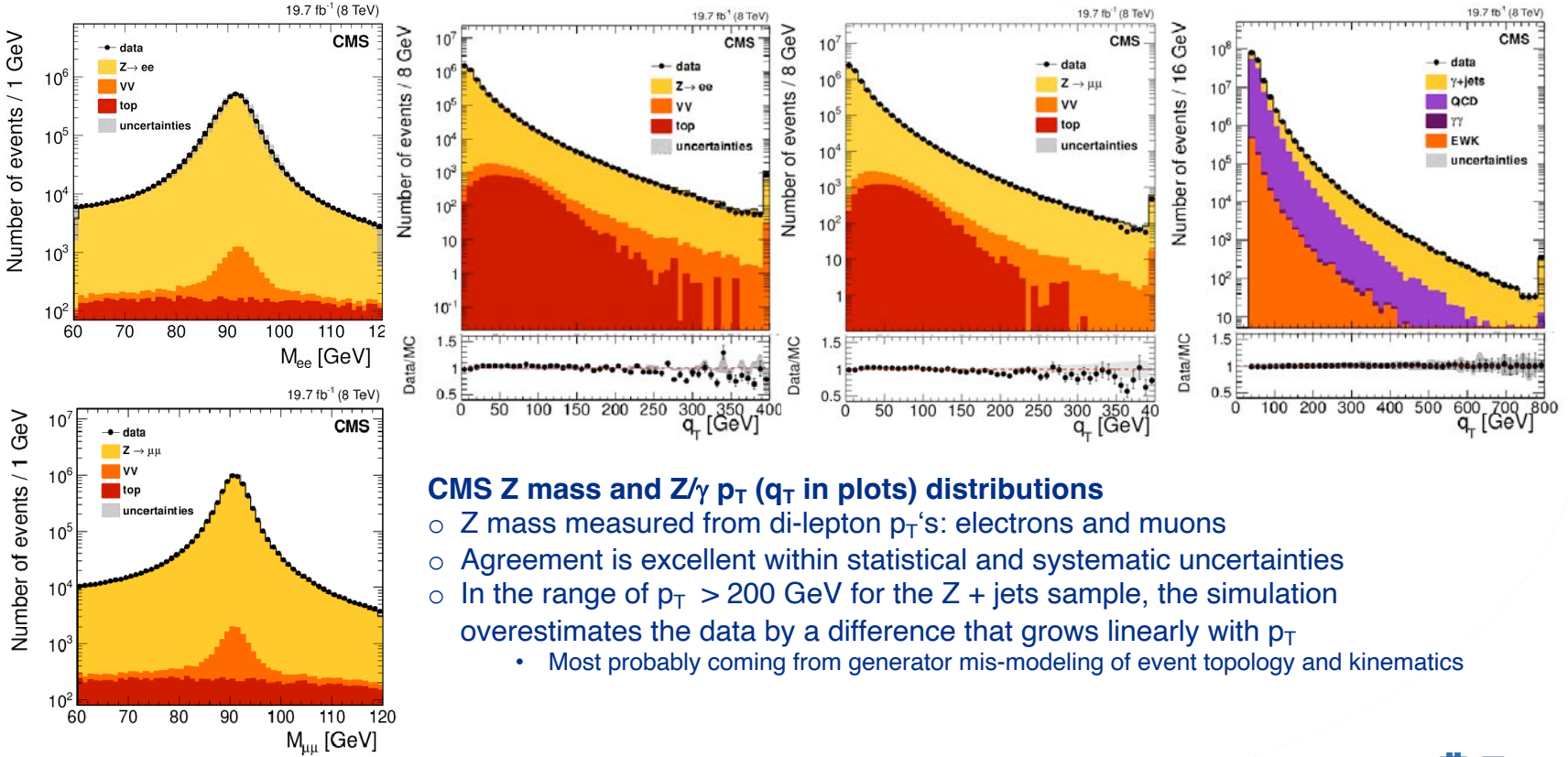
- $VH, t\bar{t}, t\bar{t}Z, t\bar{t}W, t\bar{t}H$

Detector simulation accuracy enters through modeling of γ, e/μ, jets, and E_T^{miss} (E_t^{miss} coming from neutrino in W decay, energy resolution in hadronic recoil)

- material budget in tracker, EM and hadron calorimeter showers

$$M_T^W = \sqrt{2p_T^l p_T^\nu \left(1 - \cos(\phi_l - \phi_\upsilon)\right)} \qquad M^Z = \sqrt{2p^{l1}p^{l2}\left(1 - \cos(\phi_{l1} - \phi_{l2})\right)}$$

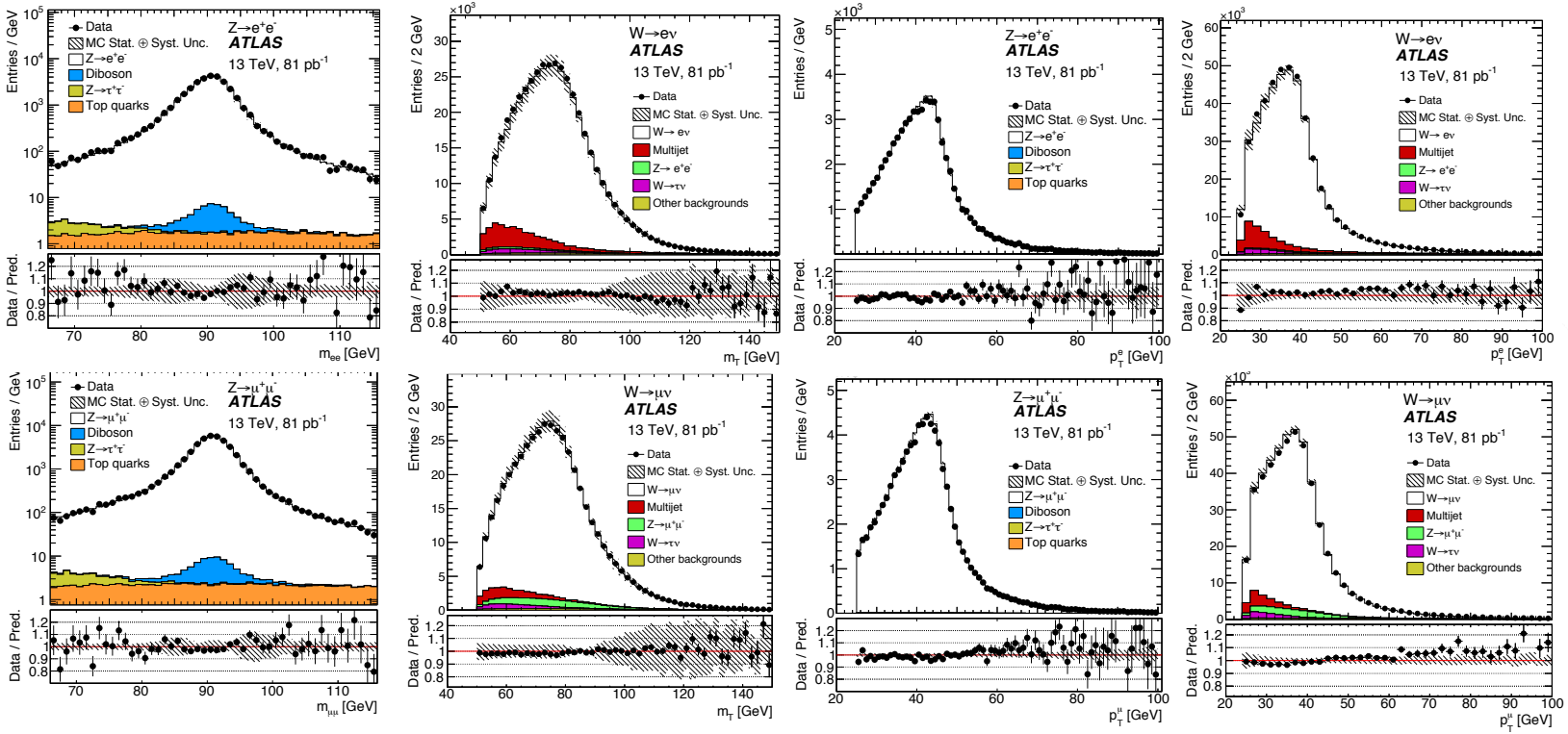
Modeling of particles and event properties: W, Z, photons



CMS Z mass and Z/γ p_T (q_T in plots) distributions

- Z mass measured from di-lepton p_T's: electrons and muons
- Agreement is excellent within statistical and systematic uncertainties
- In the range of p_T > 200 GeV for the Z + jets sample, the simulation overestimates the data by a difference that grows linearly with p_T
  - Most probably coming from generator mis-modeling of event topology and kinematics

Modeling of particles and event properties: W, Z, photons



ATLAS W/Z mass and e/μ p_T distributions in the electron and muon W/Z decay channels

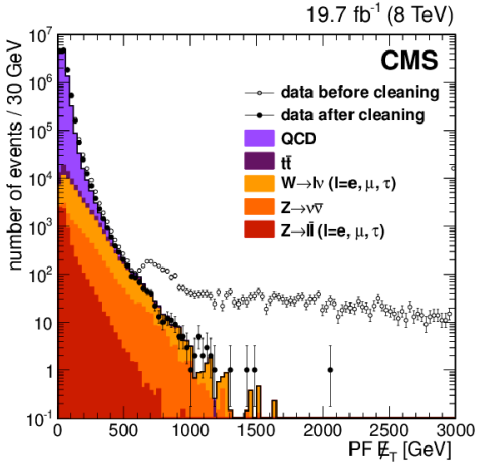
- Impressive agreement within <10% in the domain ranges with good statistics

Modeling of particles and event properties: missing E_T

Event missing transverse energy: E_T^{miss} or $\vec{\not{E}}_T = -\sum_{\text{particles}} (p_x \hat{\mathbf{i}} + p_y \hat{\mathbf{j}})$

Modeling E_T^{miss} is among the most challenging simulation tasks: depends on all types of particles, hadronic showers from jets, and un-clustered energy

- Paramount importance in BSM SUSY, ED, dark matter searches, Higgs characterization
- Intrinsic low-med (high) E_T^{miss} in SM (BSM) searches, or E_T^{miss} from detector mis-measurement



E_T^{miss} distribution for CMS di-jet events before and after applying the software algorithms to remove events with spurious E_T^{miss}

- Agreement is excellent > 500 GeV and worsens below 500 GeV as the QCD contribution increases and becomes dominant
- E_T^{miss} QCD background estimates in SM/BSM analyses typically not taken from MC
  - Shower fluctuations and un-clustered energy not modeled accurately enough
  - Impossible to demonstrate that all sources of spurious events in the tails have been identified and modeled in the MC with the correct rates

Low-med E_T^{miss} from invisible decays (neutrinos) better modeled than high E_T^{miss} tails in multi-jet samples with origin in resolution or detector malfunction

Economic impact and cost of simulation in HEP collider experiments

The CMS case, Geant4

Economic impact/cost of simulation in HEP collider experiments

We define “simulation chain” physics generation, interaction with matter (G4), readout modeling, reconstruction, analysis

- Took 85% of CPU resources used by CMS, while G4 module took 40% of total (Run 1, 2)
- ATLAS’s simulation application was 8-9 times slower than CMS’ in 2018, and used significantly more resources than CMS in physics generation
- Rest of resources used in reconstruction and analysis of real collider data

CMS in more detail (taken from analysis of 2012/May 2015-May 2016 periods)

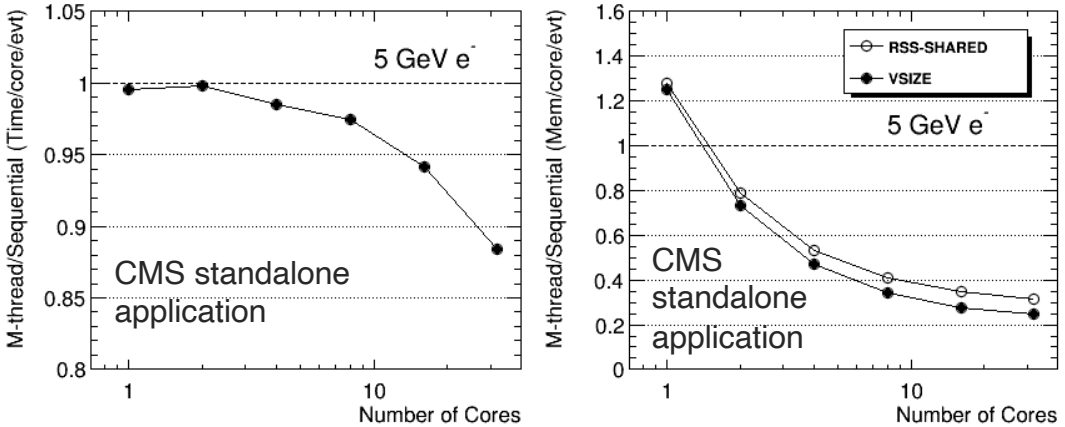
- 540k/860k core months corresponding to 45/70k CPU cores at full capacity (half in G4)
- Purchasing cost is 5/8 million dollars
- Cost of physical hardware including life-cycle, operation, maintenance
  - 0.9 cents/core hour (FNAL), or 1.4 cents/core hour (what FNAL paid industry in 2017)
- Annual cost of simulation in CMS: 3.5-6.2/5.5-10 million dollars
- Improvements of 1%, 10%, 35% in G4 time performance would render 50-80k, 500-800k, 1.8-2.8M dollars savings to CMS

Computing needs of HL-LHC program are an order of magnitude higher, depends on simulation and reconstruction solutions implemented – reconstruction will take a larger fraction (pileup)

Economic impact/cost of simulation in HEP collider experiments

Design, development, validation, operation, support of simulation toolkits, such as Geant4, as well as development of the experiment applications add to the cost

- Integrated over time, investment on G4 totals approximately ~ 500 person-years or > 100M dollars
- How much more experiment design, optimization, commissioning, operation, as well as the physics programs would have costed without Geant simulation?



Geant4 introduced multithreading capabilities in 2013 – event level

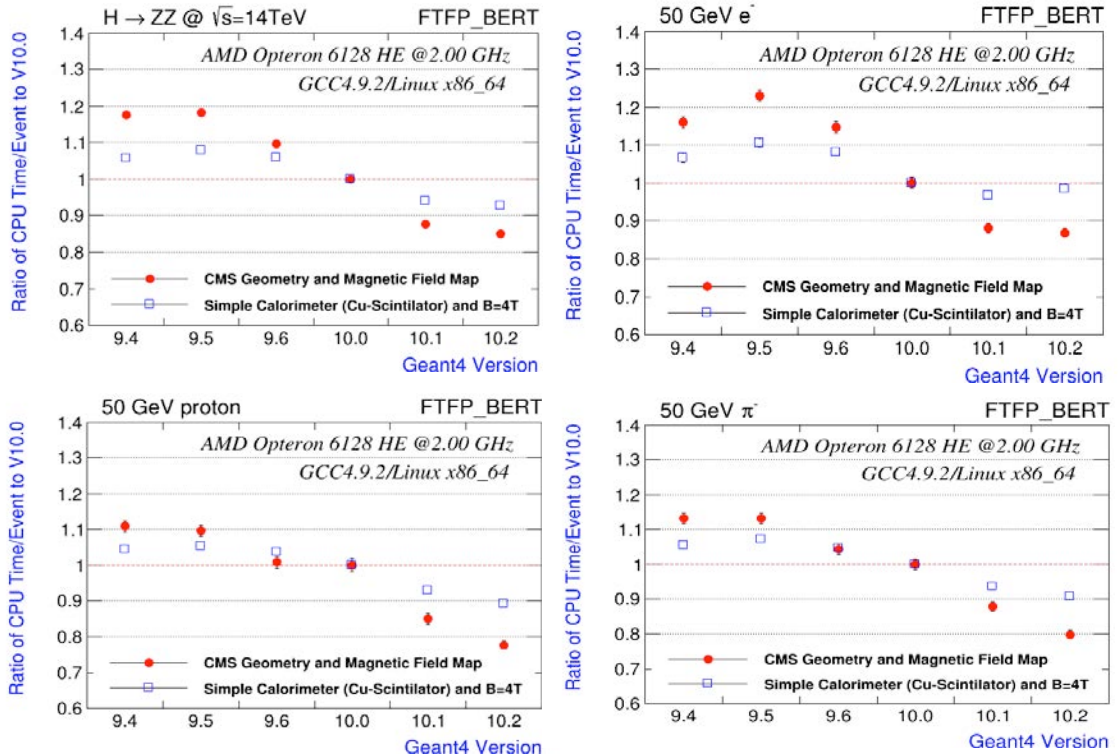
- Time performance does not improve, deviates from perfect scaling: ~10% for 30 cores
- Memory use improves significantly ~170MB in first event, ~30MB by each additional thread

Corollaries: 1- the cost of physics software is a significant fraction of the cost of detectors

2- the cost of simulation and reconstruction should be a factor in detector design

Economic impact/cost of simulation in HEP collider experiments

The G4 Collaboration has gone to great lengths to improve computing performance



During the 2010-2015 period:

- Time performance improvement was of the order of 35% (simple calorimeter & CMS standalone)
- Double digits CPU improvement while physics accuracy also improved

Remember a 35% faster G4 means ~2-3M dollars/year savings in CMS

(or we can do 35% more simulation)

But this is not enough for HL-LHC!

(and technology changes call for adaptation)

Recent and current R&D projects

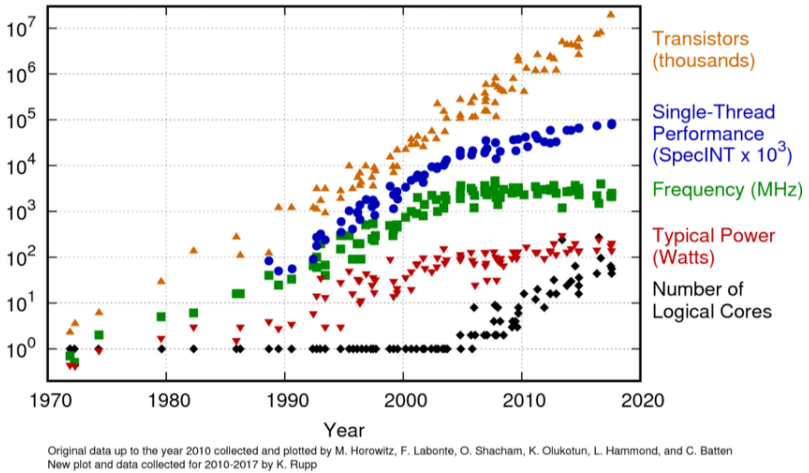
The future demands better physics accuracy and increased speed by means of novel programming techniques and modern computing architectures

A brave new world in computing

A paradigm change in computing architecture

- **Dennard Scaling (DS):** power use by silicon device/volume independent on the number of transistors
- **Moore's Law:** transistor density doubles every two years
- **Clock speed (CS):** increased 1,000 times in 1970-2000

Computer speed doubled every 2 years while cost halved



Break down of DS (leakage current), **Moore's Law** (atom sized devices), **CS** (too much power)

Evolution towards heterogeneous systems with multi-core machines using co-processors (e.g., GPUs) and complex memory configurations

Computing in HEP is not business as usual any more

Involves redesigning the **computing model for HEP**

- **Experiment software frameworks (e.g., CMSSW)** will need to use heterogenous resources locally and remotely, including supercomputing centers and commercial facilities
- **Data Management Model** to handle data access, transfer, processing across a diverse set of computing systems
- **Adapt or re-engineer almost every piece of software**, including common software tools for event generation, detector simulation and end-user analysis, as well as reconstruction algorithms
- **Portability tools** to avoid re-writing software for different computing hardware

Departure from stability of the past when the "same old software" would run faster and cheaper in future machines without adaptation or re-engineering

R&D for transition and delivery of production level software is heavy in labor

- person power with rare and expensive talents

A community effort

- R&D to address S&C challenges in the next decade must be a community effort
  - Future experiments face similar challenges
  - Resources are limited, cannot afford duplicated efforts
- The HEP Software Foundation (HSF) was formed in 2015
  - Followed early incarnations starting with a “Workshop on Concurrency in the many-Cores Era” (FNAL, 2011), and “Annual Concurrency Forum Meetings” (FNAL-2013, CERN-2014)
  - HSF facilitates coordination and common efforts in S&C across HEP in general
  - No formal organization, no representation from funding agencies, research institutions or experiments
  - A “coordination team” composed of volunteers contribute their time
- The community white paper
  - Year-long process, many working groups, two major workshops
  - Massive community engagement: 310 authors from 124 institutes, 14 chapters
  - **Published in Comp. and Soft. for Big Sci.:** https://doi.org/10.1007/s41781-018-0018-8
  - **Dedicated white paper on detector simulation:** https://arxiv.org/abs/1803.04165

The GeantV R&D project

The GeantV[7] prototype was designed to take the opportunities offered by modern computing architectures (2013-2019)

Paradigm shift – Particle-level parallelization (not event-level as G4). One thread may process particles from different events

Use parallelization techniques

- Instruction pipelining → parallel instruction handling within a processor
- Data locality and explicit vectorization
  - Single Instruction Multiple Data (SIMD)
  - Work offloading to accelerator

Computing and Software for Big Science (2021) 5:3
https://doi.org/10.1007/s41781-020-00048-6

ORIGINAL ARTICLE

GeantV

CERN, FNAL, UNESP (Brazil) BARC (India)
5-12 FTE/yr for 7 years

Results from the Prototype of Concurrent Vector Particle Transport Simulation in HEP

G. Amadio[1] · A. Ananya[1] · J. Apostolakis[1] · M. Bandieramonte[1,2] · S. Banerjee[3] · A. Bhattacharyya[4] · C. Bianchini[5,6] · G. Bitzes[1] · P. Canal[3] · F. Carminati[1] · O. Chaparro-Amaro[7] · G. Cosmo[1] · J. C. De Fine Licht[1] · V. Drogan[1,8] · L. Duhem[9] · D. Elvira[3] · J. Fuentes[10] · A. Gheata[1] · M. Gheata[1,11] · M. Gravey[12] · I. Goulas[1] · F. Hariri[1] · S. Y. Jun[3] · D. Konstantinov[1,17] · H. Kumawat[4] · J. G. Lima[3] · A. Maldonado-Romo[7] · J. Martínez-Castro[7] · P. Mato[1] · T. Nikitina[1,13] · S. Novaes[5] · M. Novak[1] · K. Pedro[3] · W. Pokorski[1] · A. Ribon[1] · R. Schmitz[15] · R. Seghal[4] · O. Shadura[1,14] · E. Tcherniaev[1] · S. Vallecorsa[1,13] · S. Wenzel[1] · Y. Zhang[1,16]

Received: 14 June 2020 / Accepted: 31 October 2020 / Published online: 3 January 2021

Abstract
Full detector simulation was among the largest CPU consumers in all CERN experiment software stacks for the first two runs of the Large Hadron Collider. In the early 2010s, it was projected that simulation demands would scale linearly with increasing luminosity, with only partial compensation from increasing computing resources. The extension of fast simulation approaches to cover more use cases that represent a larger fraction of the simulation budget is only part of the solution, because of intrinsic precision limitations. The remainder corresponds to speeding up the simulation software by several factors, which is not achievable by just applying simple optimizations to the current code base. In this context, the GeantV R&D project was launched, aiming to redesign the legacy particle transport code in order to benefit from features of fine-grained parallelism, including vectorization and increased locality of both instruction and data. This paper provides an extensive presentation of the results and achievements of this R&D project, as well as the conclusions and lessons learned from the beta version prototype.

Keywords Detector simulation · Particle transport · Concurrency · Parallelism · Vectorization

Speedup achieved not worth the effort of further development, but many of the products delivered were integrated into Geant4 resulting in improved physics and computing performance

Fermilab

A detector simulation R&D program

- The case for detector simulation R&D is strong:
  - Detector simulation toolkits, such as Geant4, are more that 25 years old and were designed for sequential programing and homogeneous computing on CPUs
    - GeantV work uncovered significant opportunity for code simplification, optimization, factorization
    - Supercomputing facilities available to HEP require effective use of their resources
    - Simulation code may not be compatible with commercial hardware in the 2030s if not adapted

- A detector simulation R&D program must be comprehensive and flexible including:
  - Toolkit modifications to take opportunities offered by heterogeneous computing architectures
  - Application of AI techniques for high-fidelity fast simulation
  - Adaptation of full/fast simulation workflows for effective/efficient use of HPCs with accelerators
  - Portability solutions to keep up with rapid evolution of computer hardware

AdePT – accelerated demonstrator of EM particle transport

R&D project started by the CERN Software group to demonstrate a realistic complete simulation workflow on GPU

The AdePT prototype[8] goals are:

- Understand opportunities and limitations for GPU usage in full HEP simulation
- Evaluate feasibility, effort needed, and performance expectations for a full HEP application that offloads EM shower simulation to GPUs
- Leverage existing experience and products including vectorized libraries, ALICE GPU reconstruction, LHCb Allen GPU trigger, performance portability libraries

Demonstrator consists of a GPU transport engine

- Dynamic kernel scheduling, management of workflow and state data
- Adapt, develop, or optimize GPU-friendly transport components
- Understand constrains/hard limits and find solutions for data handling, memory management, kernel scheduling, GPU performance

Celeritas – a GPU-based particle transport for HEP simulation

Partnership between the Exascale Computing Project (ECP) and the Computational HEP US Department of Energy programs in the USA: ANL, FNAL, ONL

Celeritas[9] would be an adaptation of Geant4 to have GPU accelerated transport

- Not a replacement of G4 but will include a mechanism to incorporate GPU acceleration
- Utilize leadership class hardware (GPUs)
- Support a complete set of physics models required by HEP experiments
- Leverage recent R&D products

Demonstrator consists of the GPU implementation of a transport loop

- Transport step loop on GPU including EM field, secondaries, scoring in user-selected cells
- EM physics for γ and $e^{\pm}$, hadron proxy for shower performance testing
- Performance testing, Figure of Merit (FOM) measurements on Summit/Volta GPU hardware
  - FOM = N_{evts}/time over full machine resource

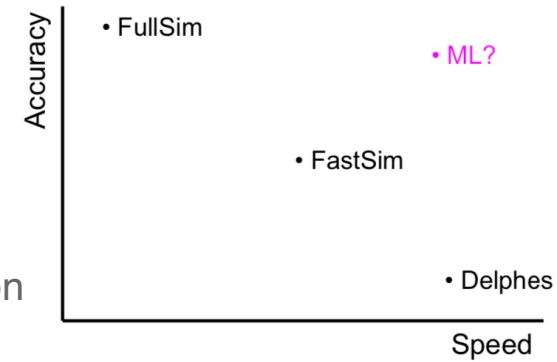
Machine learning for Simulation

Parameterized simulation is ~100x faster than G4 and models physics to within ~10%

- Reason why ATLAS processes < 25% of MC with ParSim and CMS only signal samples

- ML algorithms can be used to generate or de-noise simulation output (typically calorimeter showers using GANs or CNNs - significant R&D activity in progress)
  - ***Pros:*** may achieve higher accuracy than parametrized simulation, faster results than G4, inference can be accelerated on coprocessors (GPUs, FPGAs, etc), avenue to utilize HPCs
  - ***Cons:*** may need large training datasets and time, extrapolation may be unreliable

- Different approaches are possible
  - ***Replace (part of) FullSim:*** increase speed while preserving physics accuracy
  - ***Replace (part of) FastSim:*** decrease speed (slightly) while increasing accuracy
  - ***End-to-end:*** map generated-to-reconstructed information with no dedicated simulation step



In summary

- Physics fidelity of current simulation software, coupled with the speed of contemporary computers, has enabled HEP experiments to perform tasks that scientists could only dream about in the past
  - Detector design, stress-test of computing infrastructure, development of reconstruction software, data-driven techniques for calibration and analysis, etc.
- Detector simulation R&D for future experiments
  - Detector simulation uses a significant fraction of the HEP computing resources
  - Software tools are > 25 years old and may not run on 2030's computer platforms
  - Recently published R&D work points to opportunities for code simplification, modernization, optimization, and adaptation for use on accelerators
- My vision is for a diverse and flexible detector simulation program
  - Support for strong Geant4 development teams around the world
  - R&D program with heterogeneous computing and AI components
    - A consensus on common solutions across institutional and international boundaries must follow the initial pilot and demonstrator phase
    - Any Geant4-in-GPU-based solution will require a multi-FTE/year effort for many years

References

[0] V. Daniel Elvira, Impact of detector simulation in particle physics collider experiments, Physics Reports 695 (2017) 1-54. DOI: 10.1016/j.physrep.2017.06.002. https://arxiv.org/abs/1706.04293

[1] R. Ford, W. Nelson, The EGS Code System - Version 3, Stanford Linear Accelerator Center Report SLAC-210.

[2] R. Brun, F. Bruyant, M. Maire, A. McPherson, P. Zanarini, Geant3 CERN- DD-EE-84-1.

[3] A. Ferrari, P. Sala, A. Fasso, , J. Ranft, FLUKA: a multi-particle transport code, CERN-2005-10 (2005), INFN/TC 05/11, SLAC-R-773.

[4] The MARS Code System http://mars.fnal.gov.

[5] S. Agostineli et al. (Geant4 Collaboration), Gean4-a simulation toolkit, Nucl. Instrum. Meth. A 506 (2003) 250–303. doi:10.1016/S0168- 9002(03)01368-8.

[6] J. Allison et al. (Geant4 Collaboration), Recent developments in Geant4, Nucl. Instrum. Meth. A 835 (2016) 186–225. doi:10.1016/j.nima.2016.06.125.

[7] G. Amadio et al., GeantV: Results from the prototype of concurrent vector particle transport simulation in HEP, submitted to Software and Computing for Big Science, https://arxiv.org/abs/2005.00949v3

[8] AdeTP presentation in the Geant4 R&D Forum, September 15th 2020: https://indico.cern.ch/event/942142/sessions/363813/#20200915

[9] Celeritas presentation in the Geant4 R&D Forum, April 14th 2020: https://indico.cern.ch/event/904385/