Status of Geant4 Hadronic Physics

Alberto Ribon (CERN EP-SFT)

on behalf of the Geant4 Hadronic Physics Working Group

Geant4 Collaboration meeting, 22 September 2021

Hadronic Framework

Opportunity of a Major Release

- The Geant4 major release 11.0 offers the opportunity to remove and/or change old interfaces and classes
  - These changes require to edit a large number of files – in *hadronic/* , *physics_lists/* , *examples/* and *tests/* – but are transparent for users who utilize reference physics lists
  - But for user-made physics lists, and private process-level tests, the migration to G4 11.0 requires some changes, explained in the following pages
  - All the changes discussed here have been included in the June's beta release, G4 11.0.beta (== 10.7.ref06)
    - As expected, they are not affecting any physics results !

Removed many Inelastic Hadronic Processes (1/2)

- G4PionMinusInelasticProcess , G4PionPlusInelasticProcess ,
  G4KaonMinusInelasticProcess , G4KaonPlusInelasticProcess ,
  G4KaonZeroLInelasticProcess , G4KaonZeroSInelasticProcess ,
  G4ProtonInelasticProcess , G4AntiProtonInelasticProcess ,
  G4NeutronInelasticProcess , G4AntiNeutronInelasticProcess ,
  G4DeuteronInelasticProcess , G4AntiDeuteronInelasticProcess ,
  G4TritonInelasticProcess , G4AntiTritonInelasticProcess ,
  G4He3InelasticProcess , G4AntiHe3InelasticProcess ,
  G4AlphaInelasticProcess , G4AntiAlphaInelasticProcess ,
  G4LambdaInelasticProcess , G4AntiLambdaInelasticProcess ,
  G4SigmaMinusInelasticProcess , G4AntiSigmaMinusInelasticProcess ,
  G4SigmaPlusInelasticProcess , G4AntiSigmaPlusInelasticProcess ,
  G4XiMinusInelasticProcess , G4AntiXiMinusInelasticProcess ,
  G4XiZeroInelasticProcess , G4AntiXiZeroInelasticProcess ,
  G4OmegaMinusInelasticProcess , G4AntiOmegaMinusInelasticProcess
- G4IonInelasticProcess
- G4PhotoNuclearProcess , G4PhotoCaptureProcess , G4PhotoFissionProcess

Removed many Inelastic Hadronic Processes (2/2)

Replaced them with:

- G4HadronInelasticProcess* pionMinusInelasticProcess =
  new G4HadronInelasticProcess("pi-Inelastic", G4PionMinus::Definition());
- G4HadronInelasticProcess* pionPlusInelasticProcess =
  new G4HadronInelasticProcess("pi+Inelastic", G4PionPlus::Definition());
- …
- G4HadronInelasticProcess* antiOmegaMinusInelasticProcess =
  new G4HadronInelasticProcess("anti_omega-Inelastic", G4AntiOmegaMinus::Definition());
- G4HadronInelasticProcess* photonNuclear =
  new G4HadronInelasticProcess("photonNuclear", G4Gamma::Definition());
  photonNuclear->AddDataSet(new G4PhotoNuclearCrossSection);
- G4HadronicProcess* photonCapture = new G4HadronicProcess("photonNuclear", fCapture);
- G4HadronicProcess* photonFission = new G4HadronicProcess("photonFission", fFission);

Removed several old Hadronic Cross Sections (1/2)

- G4IonsKoxCrossSection , G4IonsSihverCrossSection , G4IonsShenCrossSection , G4TripathiCrossSection , G4TripathiLightCrossSection
- G4ProtonInelasticCrossSection , G4NeutronInelasticCrossSection , G4PiNuclearCrossSection , G4CrossSectionPairGG
- G4IonProtonCrossSection
- G4GeneralSpaceNNCrossSection
- G4FastPathHadronicCrossSection
- **Gheisha cross sections** :
  G4HadronCaptureDataSet , G4HadronFissionDataSet , G4HadronElasticDataSet
  G4HadronInelasticDataSet , G4HadronCrossSections

Removed several old Hadronic Cross Sections (2/2)

Replaced them with:

- G4VCrossSectionDataSet* ionInelasticXS =
  new G4CrossSectionInelastic(new G4ComponentGGNuclNuclXsc);
- G4VCrossSectionDataSet* protonInelasticXS =
  new G4BGGNucleonInelasticXS(G4Proton::Definition());
- G4VCrossSectionDataSet* neutronInelasticXS = new G4NeutronInelasticXS;
- G4VCrossSectionDataSet* pionPlusInelasticXS = new G4BGGPionInelasticXS(G4PionPlus::Definition());
  G4VCrossSectionDataSet* pionMinusInelasticXS =
  new G4BGGPionInelasticXS(G4PionMinus::Definition());
- G4VCrossSectionDataSet* antibaryonInelasticXS =
  new G4CrossSectionInelastic(new G4ComponentAntiNuclNuclearXS);
- G4VCrossSectionDataSet* ggHadInelasticXS =
  new G4CrossSectionInelastic(new G4ComponentGGHadronNucleusXsc);
- G4VCrossSectionDataSet* neutronCaptureXS = new G4NeutronCaptureXS;
- G4VCrossSectionDataSet* neutronFissionXS = new G4ZeroXS;

Removed a few old Stopping At-Rest Classes (1/2)

- G4PiMinusAbsorptionBertini
- G4KaonMinusAbsorptionBertini
- G4AntiProtonAbsorptionFritiof
- G4AntiNeutronAnnihilationAtRest
- G4SigmaMinusAbsorptionBertini
- G4AntiSigmaPlusAbsorptionFritiof

Removed a few old Stopping At-Rest Classes (2/2)

Replaced them with:

- G4HadronStoppingProcess* piMinusAbsorption =
  new G4HadronicAbsorptionBertini(G4PionMinus::Definition());
- G4HadronStoppingProcess* kMinusAbsorption =
  new G4HadronicAbsorptionBertini(G4KaonMinus::Definition());
- G4HadronStoppingProcess* antiProtonAbsorption =
  new G4HadronicAbsorptionFritiof(G4AntiProton::Definition());
- No replacement for anti-neutron absorption at rests: it is unphysical to consider that a neutral particle comes at rest
- G4HadronStoppingProcess* sigmaMinusAbsorption =
  new G4HadronicAbsorptionBertini(G4SigmaMinus::Definition());
- G4HadronStoppingProcess* antiSigmaPlusAbsorption =
  new G4HadronicAbsorptionFritiof(G4AntiSigmaPlus::Definition());

Removed RPG Model and Unused Utility Classes

- model/**rpg/**
- G4VCrossSectionBase
- G4LightMedia
- DumpFrame
- G4IsoResult
- G4HadSignalHandler
- G4GHEKinematicsVector
- G4HadReentrentException
- G4HadronicDeprecate,
- file G4Pair.hhnew

Note : no replacements !

Other Changes (1/2)

- Moved ***G4HadronInelasticProcess*** from ***management/*** to ***processes/***
  - To be consistent with G4Hadron{Elastic, Capture, Fission}Process
- Renamed ***G4HadronCaptureProcess*** as ***G4NeutronCaptureProcess***
  - It applies only to neutron projectile
- Renamed ***G4HadronFissionProcess*** as ***G4NeutronFissionProcess***
  - It applies only to neutron projectile
- Eliminated the two directories: *hadronic/**models/management/***
  *hadronic/**models/util/***
  moving their classes in: *hadronic/**management/***
  *hadronic/**util/***
  - The 3 classes: *G4V3DNucleus* , *G4VNuclearDensity* , *G4VKineticNucleon* have been moved from *hadronic/models/management/*
    to *hadronic/util/*

Other Changes (2/2)

- Clean-up useless dependencies and not existing directories
  - Several GNUmakefile, CMakeLists.txt and sources.cmake files need anyhow to be modified
- Removed all old deprecated commands and environmental variables
  - Deprecated since G4 10.7

What remains to be done

- Use modern C++ – *i.e.* C++{11,14,17,20} – in hadronic classes
  - Large number of classes need to be reviewed
  - This will happens slowly after Geant4 11.0, in minor releases, because no change in user code is needed

Creator Model ID

Creator Model ID (1/2)

- Unique identifier of the physics model (EM, HAD, *etc*.) responsible for the creation of a track
  - The solution in the G4 10 series is based on the class ***G4PhysicsModelCatalog*** that does the job, but many users would prefer to have a **fixed-forever model identifier**, – *i.e.* which remains the same for all applications, physics lists and G4 versions
  - The major release G4 11.0 offers the possibility to change it, according to the following solution, discussed and agreed last year:
    - **3 redundant**, useful information to identify uniquely the model creator
      - **ID** : a (large & sparse) integer that provides the type of physics (useful for analysis)
      - **Name** : a string with the name of the model (useful for debugging)
      - **Index** : a (small, contiguous) integer: the “index” of the vector of IDs & Names (useful for plotting)

Creator Model ID (2/2)

- First implementation in G4 10.7.ref**07**
  - Please take a look: there is still time to change some convention on ID values or Names given to models
- After G4 11.0, **ID**s and **Name**s of existing models will be frozen
  - Because users want to have fixed-forever identifiers
- New models that will be introduced after G4 11.0 need :
  - To have their **ID** and **Name** to be included at the bottom of the method ***G4PhysicsModelCatalog::Initialize()*** in the file ***global/management/src/G4PhysicsModelCatalog.cc***
  - **ID** and **Name** should follow some convention – clearly specified as comments in the above method
    - **Index** is for free, as the index of the ID and Name vectors corresponding to the position where the model has been added

Models & Hadronic Showers

FTF (Fritiof) String Model

- Reminder
  - Up to ~ G4 10.1, revision of FTF by V. Uzhinsky improved both thin-target data and hadronic showers
  - Then, developments in the model driven by the goal of improving thin-target data led to worse hadronic showers (*i.e.* higher energy response) --> released “stable/production” version
  - Since G4 10.5, we continue the development and recommend a more consistent Birks quenching treatment for the energy response issue
  - In G4 10.6, started the extension for charm and bottom hadrons; refinement of nucleus-nucleus interactions; improved annihilation at rest of light anti-ions
- Main developments included in G4 10.7
  - Completed the extension for charm and bottom hadronic interactions

QGS (Quark Gluon String) Model

- Reminder
  - In 2014, V. Uzhinsky started to revised QGS according to the theoretical prescriptions of Kaidalov – as an alternative string model *w.r.t.* FTF
  - In G4 10.1 new version of the string hadronization
  - In G4 10.5 new version of the string formation
  - In G4 10.6 started extension for charm & bottom hadrons; various fixes
- Main developments included in G4 10.7
  - Completed the extension for charm and bottom hadronic interactons
- QGS becomes competitive with FTF above roughly 15 – 20 GeV
  - From thin-target data
  - For lab. projectile E_{kin} below these energies, FTF is better

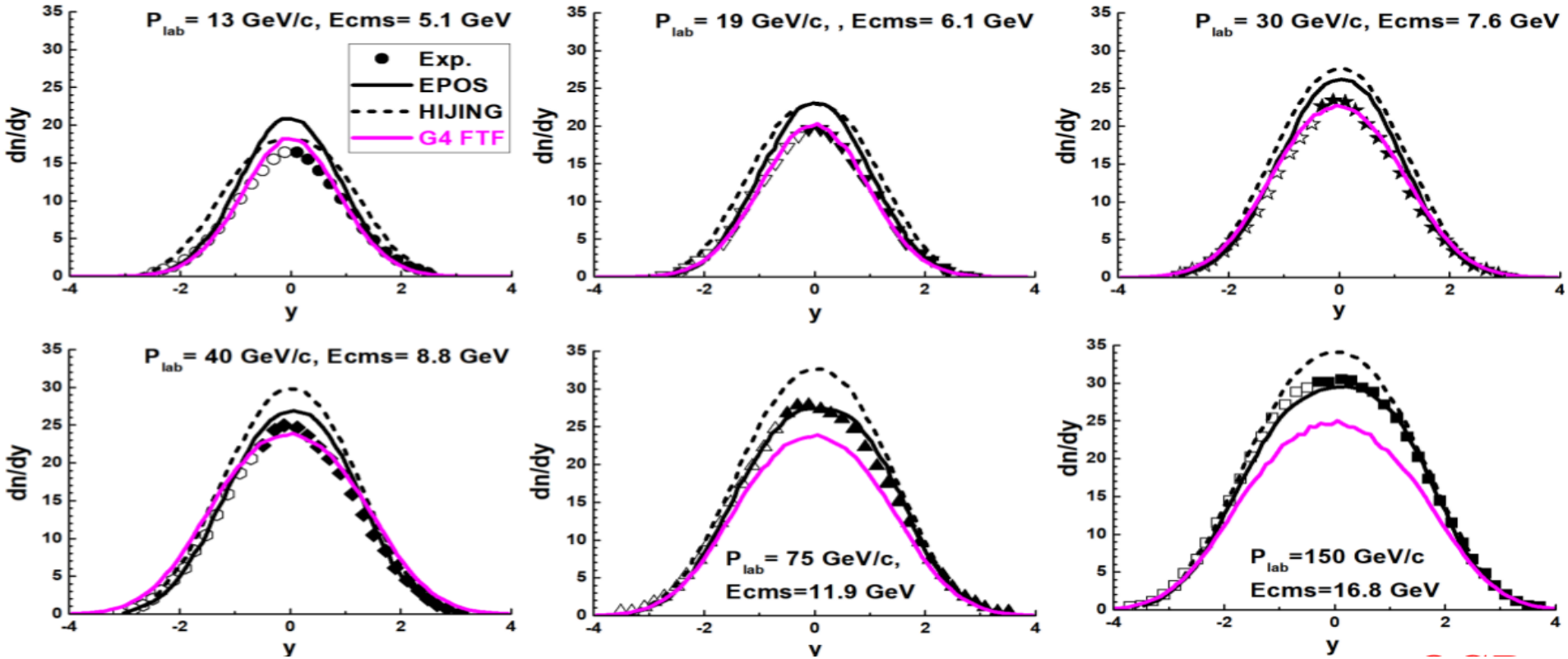
On-going Developments in String Models

- Developments for the coming release, G4 11.0
  - Validation and refinement of nucleus-nucleus interactions in FTF model
    - See next slide
    - Note that currently QGS does not handle ion projectiles
  - Clean-up for the coming, major release
  - Start extension to cover projectile hyper-nuclei and anti-hyper-nuclei
    - More on this topic later

Nucleus-Nucleus Inelastic Interactions

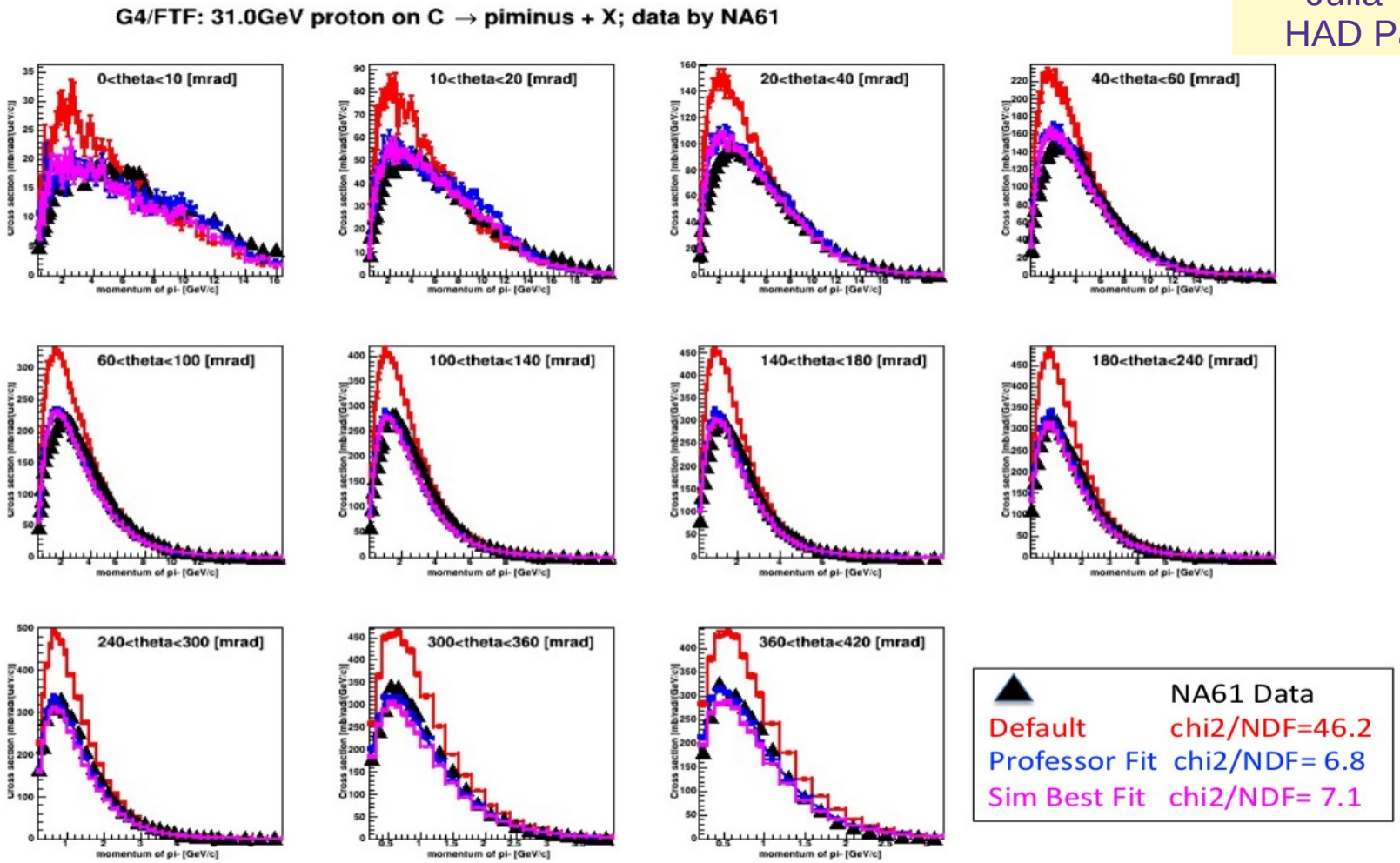
A.Galoyan, V. Uzhinsky

NA61/SHINE data on Ar-40 + Sc-45 and FTF model
5 % centrality, dn/dy of π-mesons



On-going work on Tuning FTF Parameters

Julia Yarba (FNAL)
HAD Parallel Session



Intra-nuclear Cascade Models

- **BERT** (Bertini-like cascade)
  - Stable
    - The most used, workhorse cascade model in HEP
- **BIC** (Binary Cascade)
  - Stable
    - In HEP used sometimes for evaluating systematic errors
- **INCLXX** (Liege cascade)
  - Currently stable, but with on-going extension to antiprotons
    - In HEP it is used by **ALICE** in the Tracker region (while using FTFP_BERT elsewhere)
    - Interest by ALICE and PANDA (at FAIR) on **hyper-nuclei** physics : both INCL and ABLA are capable to handle both projectile and target hyper-nuclei
    - Great interest of the AD experiments (GBAR, ASACUSA, AEgIS, ALPHA, ATRAP) and some astroparticle experiments (GAPS) for **low-energy anti-baryon annihilations**

Extension of INCLXX to antiproton

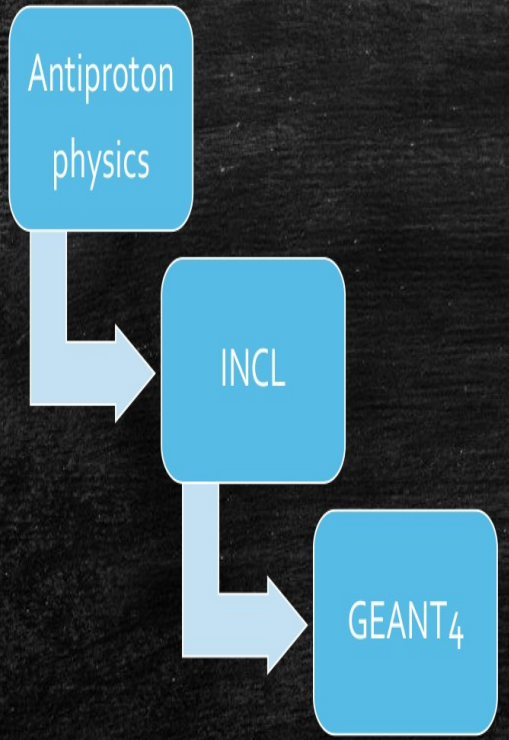
Interaction of antiprotons with nuclei

Demid Zharenov

15/09/2021

At rest annihilation

- Annihilation process
- Annihilation orbit determination
- Annihilation position
- Final states



- At rest simulation will be soon implemented and applied for new experiment at AD
- Add in flight physics (yet without $\bar{\Lambda}, \bar{\Sigma}, \bar{\Xi}$)
- Beta-version of INCL for GEANT4 with antiprotons is planned for Summer 2022

Hadronic Showers (1/2)

- Reminder:
  - Hadronic showers kept stable from G4 **10.1** to **10.4**
    - Released the “production/stable” versions of the string models (FTF & QGS) (snapshots of the development versions available in the beta releases)
  - Change of hadronic showers in G4 **10.5**
    - Finally release the development version of the string models (FTF & QGS)
    - Significant increase of energy response, to be “absorbed” with a more consistent treatment of the **Birks** quenching
  - Change of hadronic showers in G4 **10.6**
    - Mainly due to the change in the **transition region** between BERT and FTFP (from [3, 12] GeV to **[3, 6] GeV**)
  - Stable showers in G4 **10.7**
    - Extension to **charm & bottom** hadron nuclear interactions
    - Only a small change (~% level) of hadronic showers for QGS-based physics lists

Hadronic Showers (2/2) *(see plots in backup slides)*

- Current hadronic showers up to now (*i.e.* G4 10.7.ref07):
  - Stable showers for both FTF- and QGS-based physics lists
    - Similar as those of G4 10.7.p02
  - Showers of QGSP_BERT *vs.* those of FTFP_BERT
    - A bit higher energy response, larger energy fluctuations, longer and narrower shower shapes

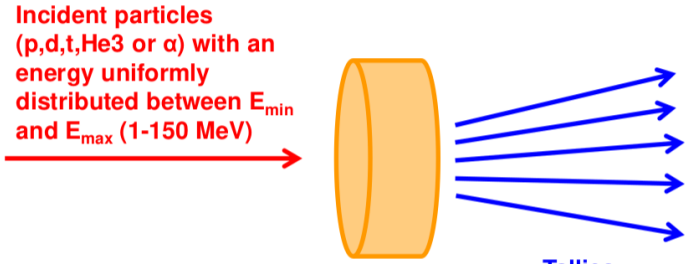
ParticleHP

- Study of (d, xn) reactions for fusion applications
  - See talk by Emilio Mendoza at the Hadronic Parallel Session
- New library G4TENDL-**1.4**, based on TENDL-**2019**, for G4 11.0
  - 562 isotopes (same as G4NDL4.6)
  - 864 MB
  - Created mainly from TENDL-2019
  - Some exceptions:
    - Proton incident: 46 isotopes from ENDF/B-VIII.0
    - Deuteron incident: 6 isotopes from ENDF/B-VIII.0 (include also JENDL/DEU-2020?)
    - Triton incident: 6 isotopes from ENDF/B-VIII.0
    - He3 incident: 5 isotopes from ENDF/B-VIII.0
    - Alpha incident: full TENDL-2019
  - 6 reactions not present in TENDL-2014 nor in ENDF/B-VII.1, where the projectile is heavier than the target: d+p, t+p, t+d, He3+p, He3+d, He3+t. They were created from the inverse reaction (p+d, p+t, …) from the ENDF/B-VII.1 data (already in the previous list).

Emilio Mendoza (CIEMAT)
HAD Parallel Session

Verification

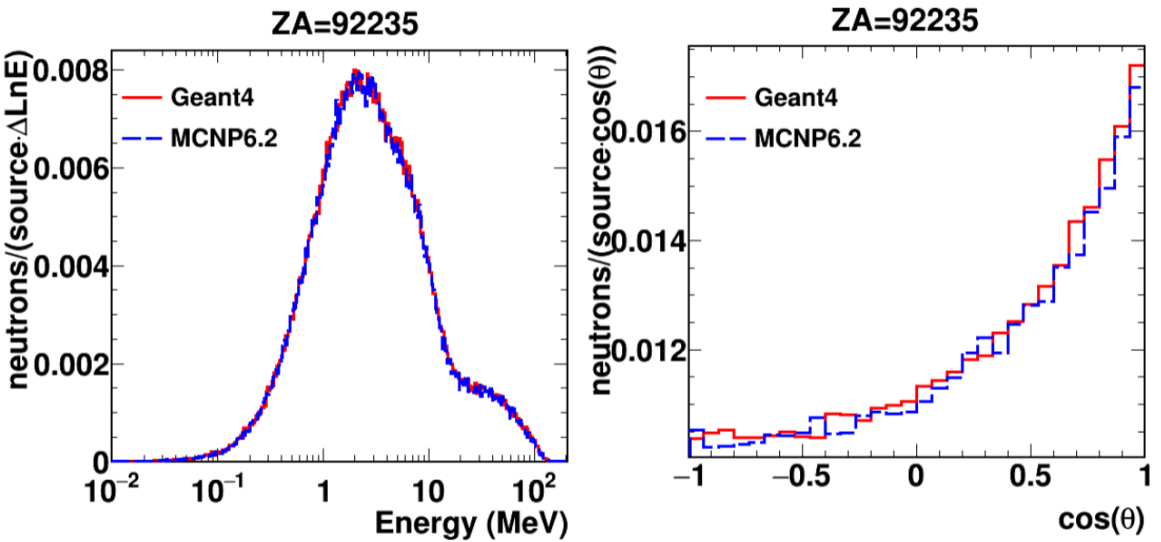
We have performed a verification of ParticleHP+G4TENDL1.4 by performing a set of similar simulations in Geant4 and MCNP6.2.



Target: a cylinder (2 cm radius + 5 cm length) made of a monoisotopic material, with density = 1 g/cm^3.

Tallies:
- Source particle flux inside the target (energy dependent)
- Neutron flux inside the target (energy dependent).
- 2D Energy-angular distributions of the neutrons and gamma-rays exiting the target.

Examples (incident protons)



Energy (left) and angular (right) distributions of the neutrons leaving the target in the p+^{235}U simulations.

Other Activities

Testing & Validation

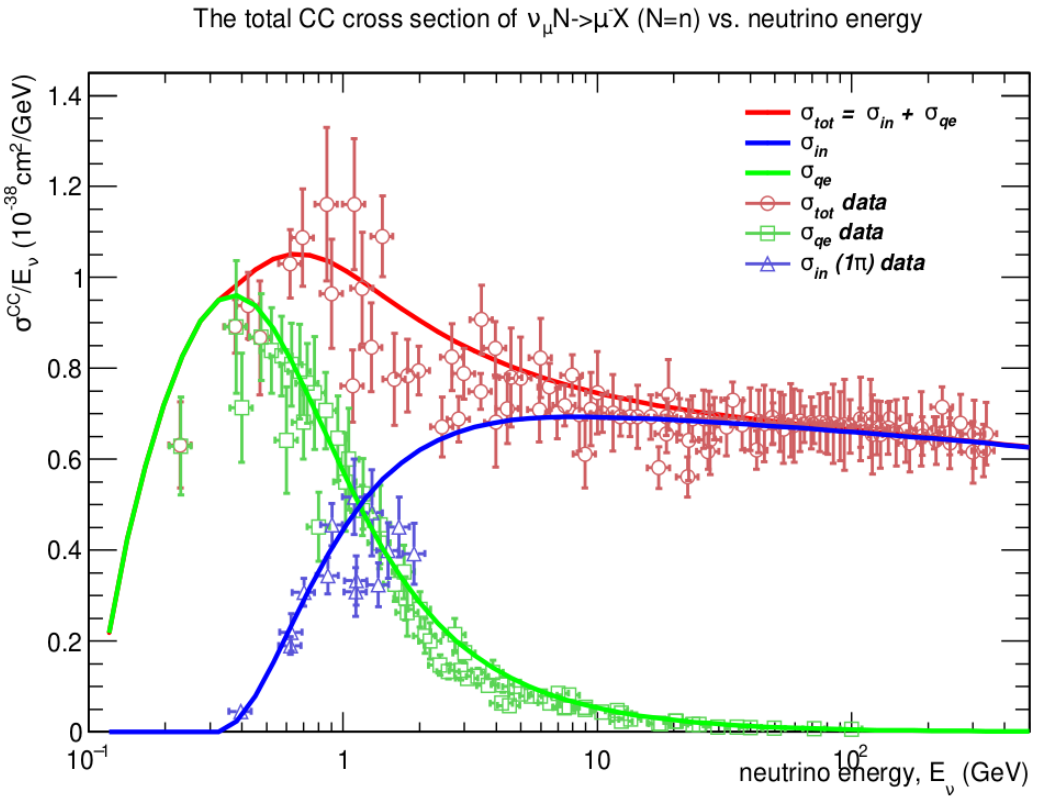
- geant-val
  - Used regularly for testing of hadronic showers
  - On-going work to include more tests
    - *E.g.* some of those present in test22 or some of those used with DoSSiER
- FNAL validation
  - test47 (BNL & MIPP data), CMS 2006 test-beam, CMS collider data
- Re-started the activity of importing calorimeter test-beams
  - Lorenzo Pezzotti, new fellow at CERN EP-SFT
  - On-going work for ATLAS TileCal, and new Dual-Readout calorimeter
  - To be included in geant-val

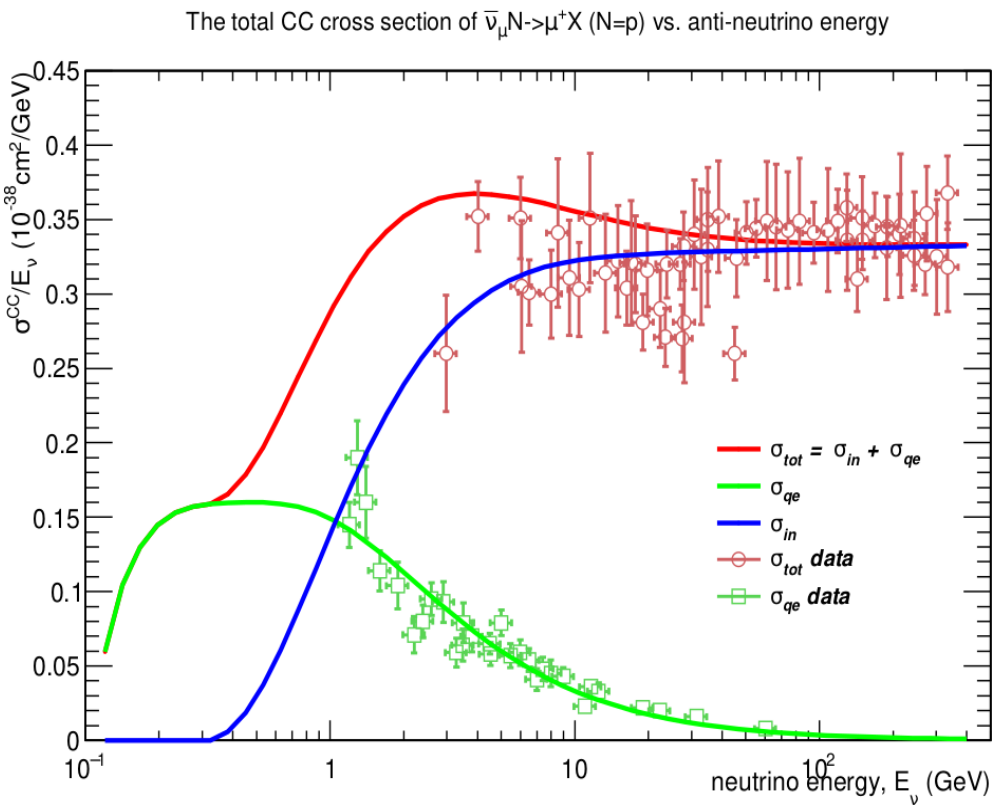
G4GammaNuclearXS

- Work by student Kutsenko Bogdan and by V. Ivanchenko
- New class based on IAEA photo-nuclear data library (Aug 2020) included in G4PARTICLEXS**4.0** dataset (G4 10.7.ref07)
  - Before we had only one photo-nuclear cross section in Geant4 − the old (2002) Kossov's (CHIPS) class *G4PhotoNuclearCrossSection*
  - The new class is activated by default in all reference physics lists
  - The IAEA evaluated photo-nuclear data library, in ENDF-6 format, covers the energy range **0 – 130 MeV** (including the important Giant Dipole Resonance energy region) for 219 isotopes
  - For elements without IAEA data – for all energies – and for all isotopes above 150 MeV, the *G4PhotoNuclearCrossSection* is used
    - With linear interpolation between the two in the region 130 – 150 MeV

Neutrino – Nuclear interactions

- On-going work by Vladimir Grichine





Production of Isomers

- Changed the half-life time threshold for isomer production to **1 ns**
  - Before it was **1 μs**
    - G4Track objects to be transported by Geant4 are created for isomers with half-life time above this threshold
  - The motivation is to avoid unphysical large missing energy events as seen by NA61/SHINE calorimeter simulations
    - Due to unphysical large Lorentz boosts of the decay products of isomers whose half-life time threshold was below threshold and therefore were not transported, hence did not loose quickly their kinetic energy by ionization as they should do...
  - CPU overhead measured to be below ~1%
    - So we get better physics with negligible overhead !

New Cut for Very-Late Radioactive Decays

- Introduced a default threshold parameter of **10^27 ns** to kill ions at rest with sampled decay time larger than this threshold
  - The default value corresponds to twice the age of the Universe
    - But it can be changed via a UI command, e.g. to set it to 1000 years:
      /process/had/rdm/thresholdForVeryLongDecayTime *3.1536e+10 second*
  - It applies only to analogue mode
  - It applies to the sampled decay time, not to the average lifetime
  - Motivation: not to save CPU time (although it does, very little), but to avoid to get counter-intuitive (but correct) results on energy deposition by running Geant4 out-of-the-box
    - *E.g*. more energy deposit in a block of PbWO4 than the beam energy with RDM
    - Due to very late radioactive decays, *e.g.* from natural isotopes W183, Pb204, W180
    - Real effect, depending on the “proton cut”, not observable in practice

Radioactive Decay of Stable Ions at Rest

- Stable ions at rest have RD as process defining the step
  - There is nothing wrong with this, and negligible overhead, because:
    - Radioactive Decay is assigned to ***G4GenericIon***
    - It is allow for a process to be invoked and return no secondaries (which is what is happening as soon the decay table of the ion is *nullptr*)
  - However, it can be confusing for users
- One solution is to separate between stable and unstable ions
  - And assign Radioactive Decay only to the unstable ions
  - Likely the best solution, but tricky to implement efficiently in MT mode…
- A simpler solution is to define an empty ***G4NoProcess***
  - Which is not assigned to any particle, but used for stable ions at rest in ***G4SteppingManager::InvokeAtRestDoItProcs***

Hyper-nuclei

Light Hyper-nuclei and Anti-Hyper-nuclei (1/2)

- ALICE has required the transport of hyper-triton in Geant4
  - Request made during the Geant4 Technical Forum on 23 March 2020
- This is the updated, complete list they need “for the next ~10 years” (6+6 new particles; a more general treatment maybe needed later...)

		PDG code: **10LZZZAAAI**
A = 3		
– **hyper-triton**	**(p – n – Λ°)**	1010010030
A = 4		
– **hyper-hydrogen-4**	**(p – n – n – Λ°)**	1010010040
– **hyper-helium-4**	**(p – p – n – Λ°)**	1010020040
– **double-hyper-hydrogen-4**	**(p – n – Λ° – Λ°)**	1020010040
– **double-hyper-double-neutron**	**(n – n – Λ° – Λ°)**	1020000040
A = 5		
– **hyper-helium-5**	**(p – p – n – n – Λ°)**	1010020050

+ their **anti-nuclei**

Light Hyper-nuclei and Anti-Hyper-nuclei (2/2)

- ALICE is interested in those light hyper-nuclei & anti-hyper-nuclei for kinetic energies **below ~ 15 GeV** and in their **Tracker** region
  - Their decay lengths are of the order of a few **cm**
    - So it is enough to cover the beam pipe and the Inner Tracker System (ITS)
  - ALICE uses the equivalent of "**FTFP_INCLXX**" in the Tracker region
    - And FTFP_BERT elsewhere
- Proposal for the release G4 11.0
  - Introduce 6 new light hyper-nuclei and their 6 anti-particles
  - Enable the possibility to switch on EM physics (ionization) for these hyper-nuclei and anti-hyper-nuclei, for at least a few physics lists
  - Extend the interfaces & utilities of hadronic physics to handle hyper-nuclei projectiles – while the physics (hadronic cross sections and models) will be developed next year, for G4 11.1

Backup

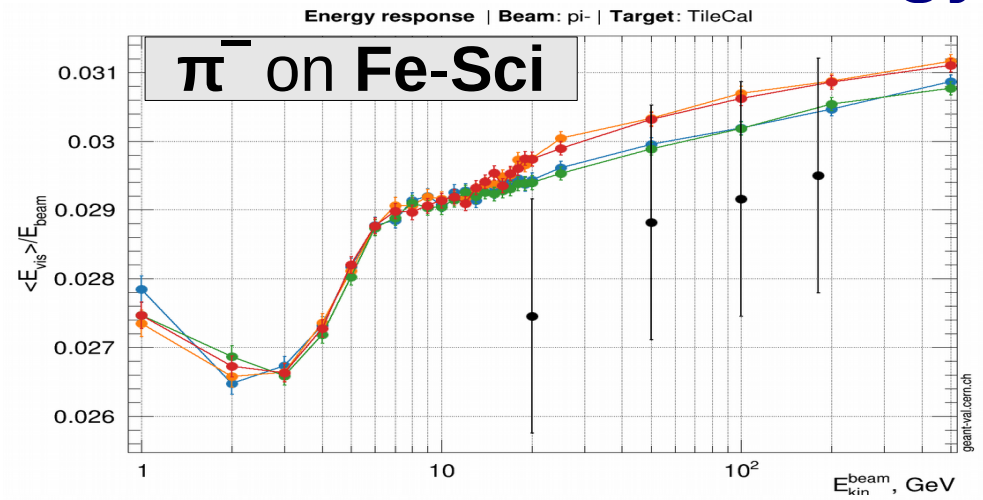
Pion− showers

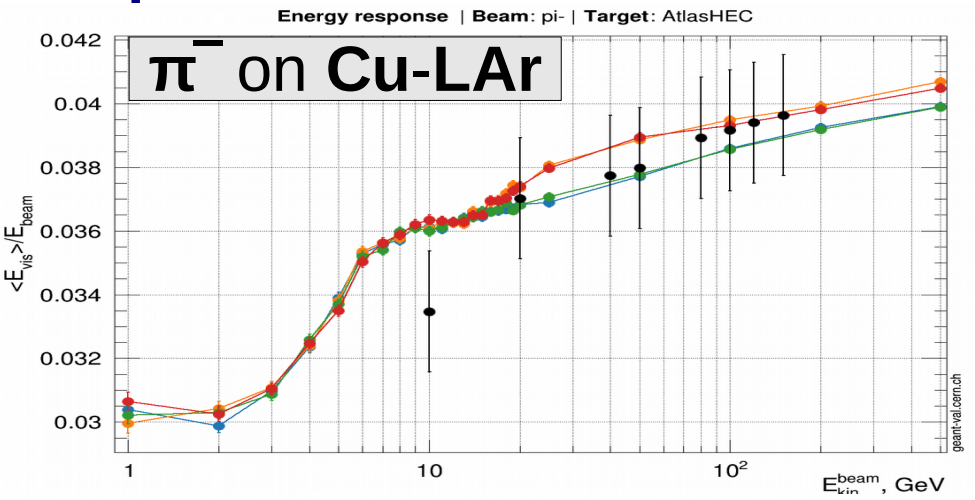
G4 10.7.ref07 FTFP_BERT
G4 10.7.p02 FTFP_BERT

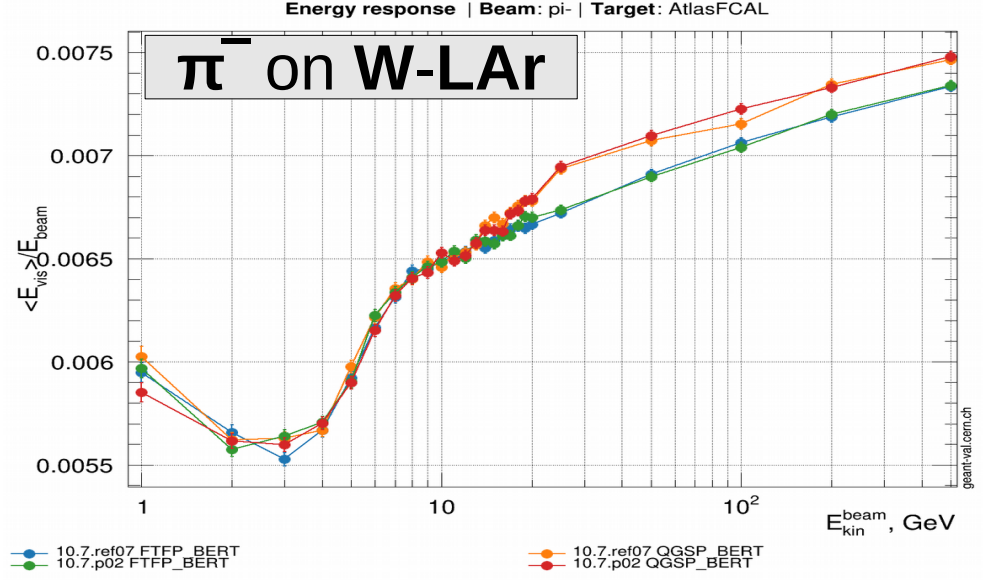
G4 10.7.ref07 QGSP_BERT
G4 10.7.p02 QGSP_BERT

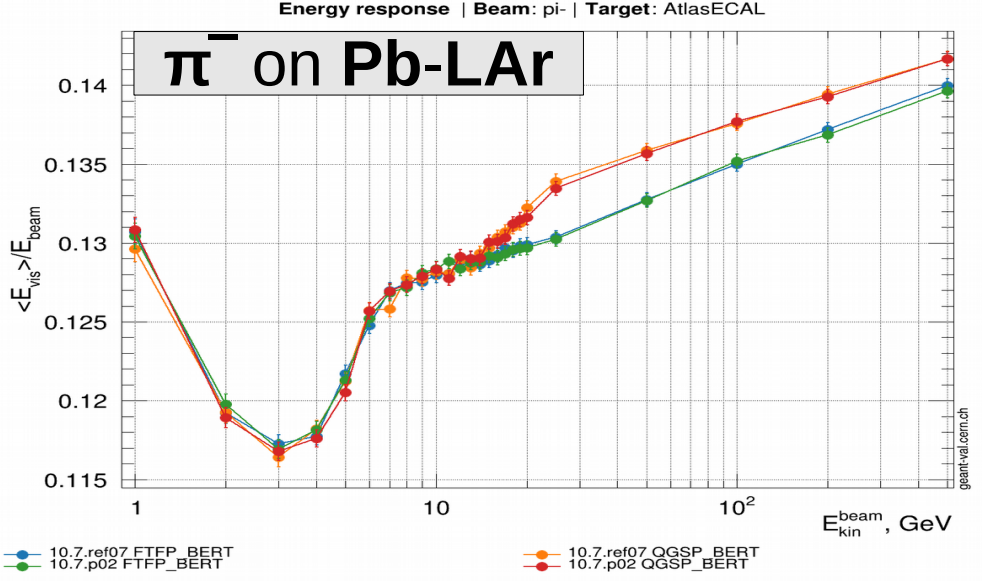
Note : conventional Birks treatment
(easier and no experimental h/e to fit !)

Energy Response

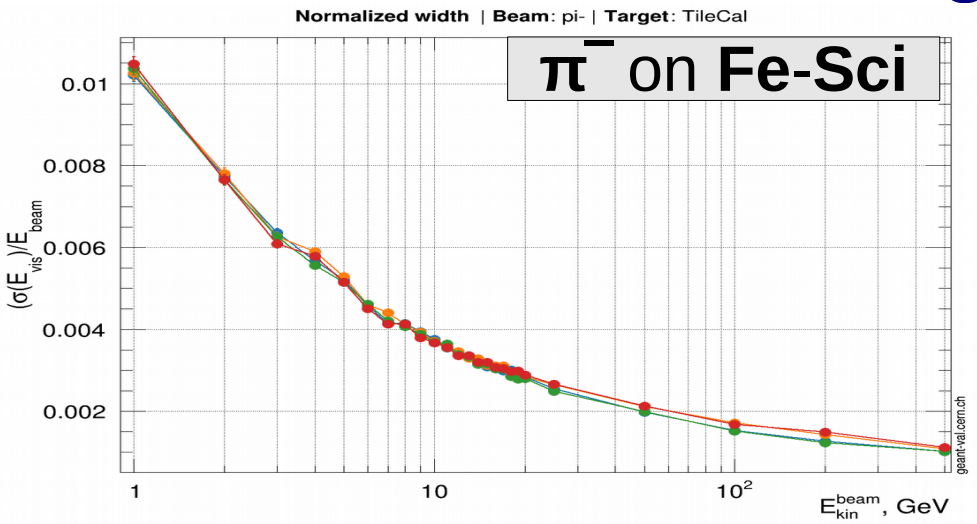


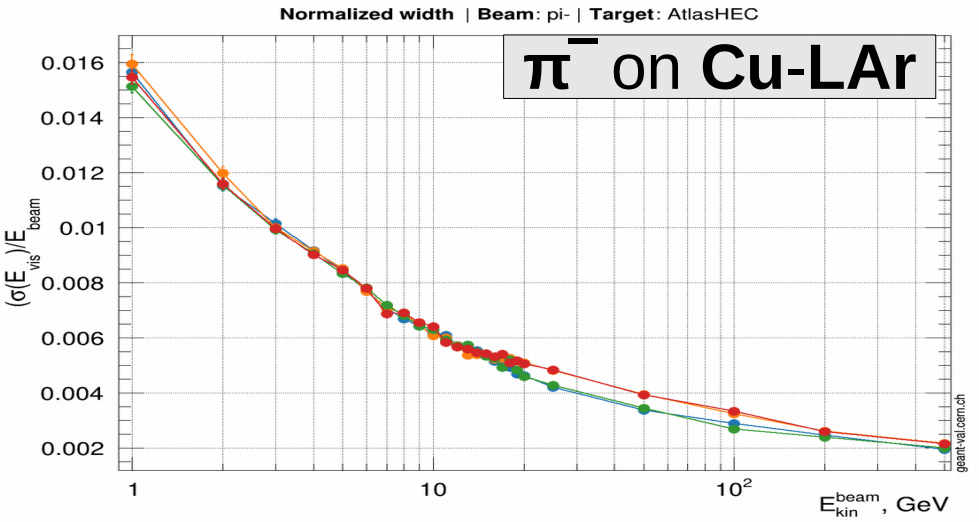


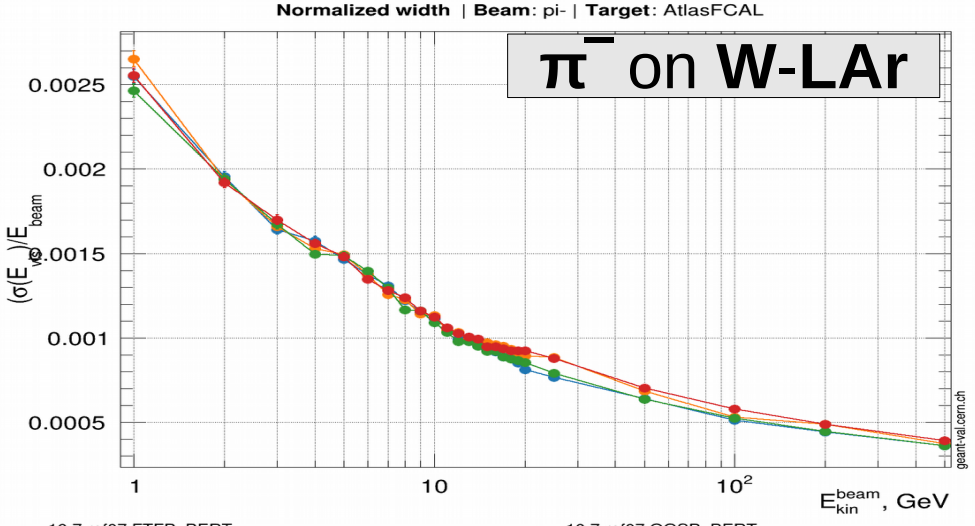


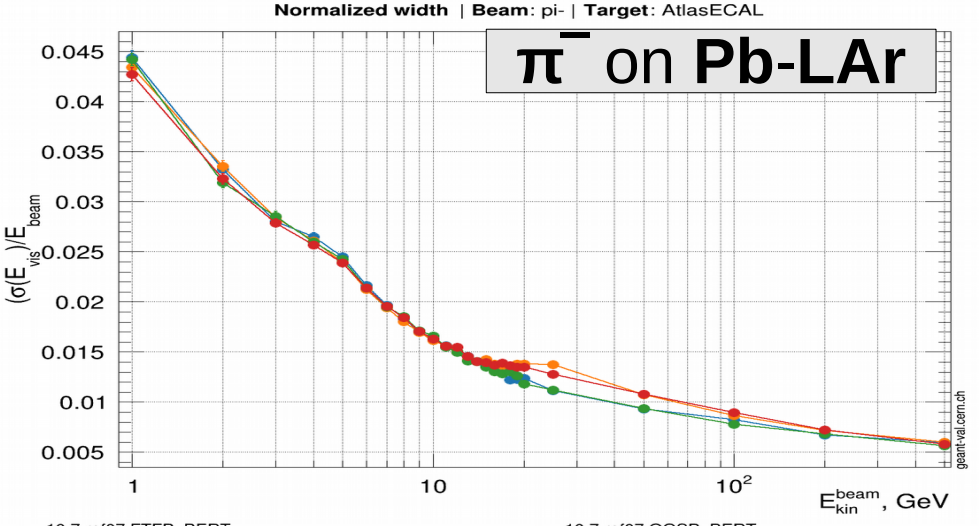


Energy Width

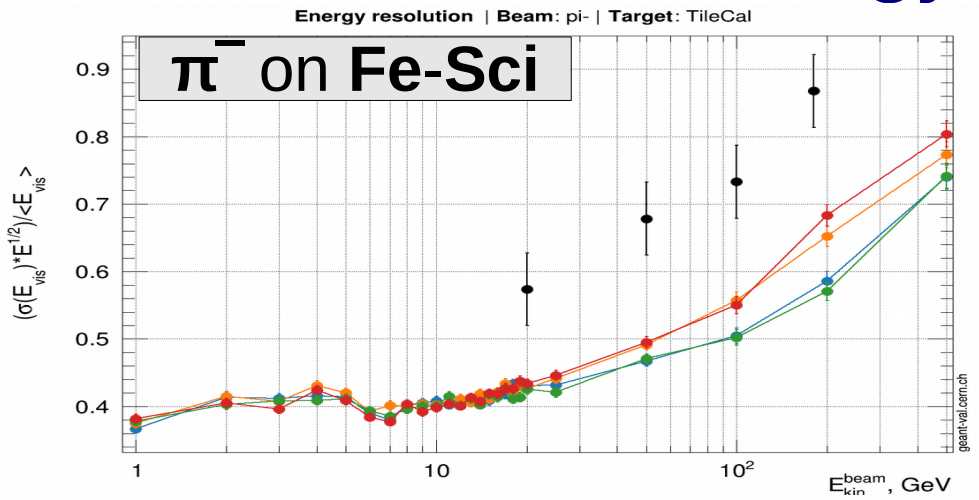


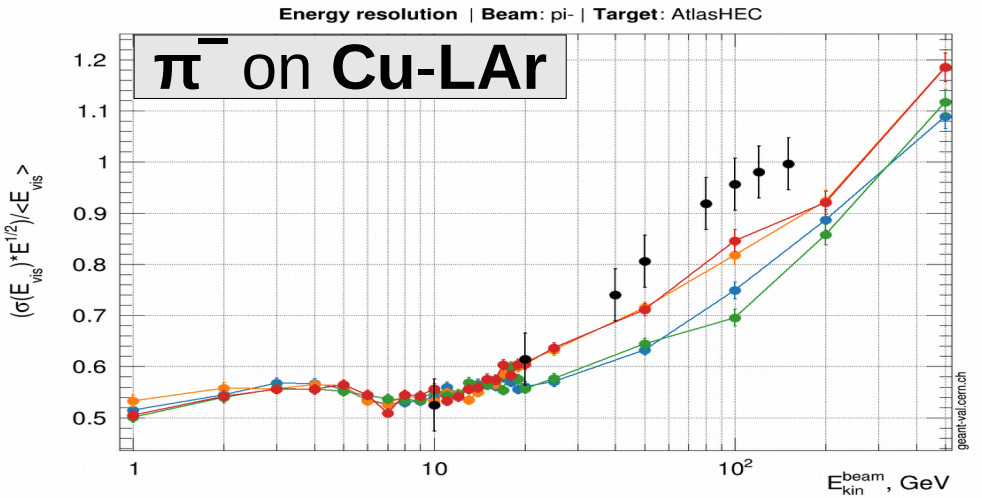


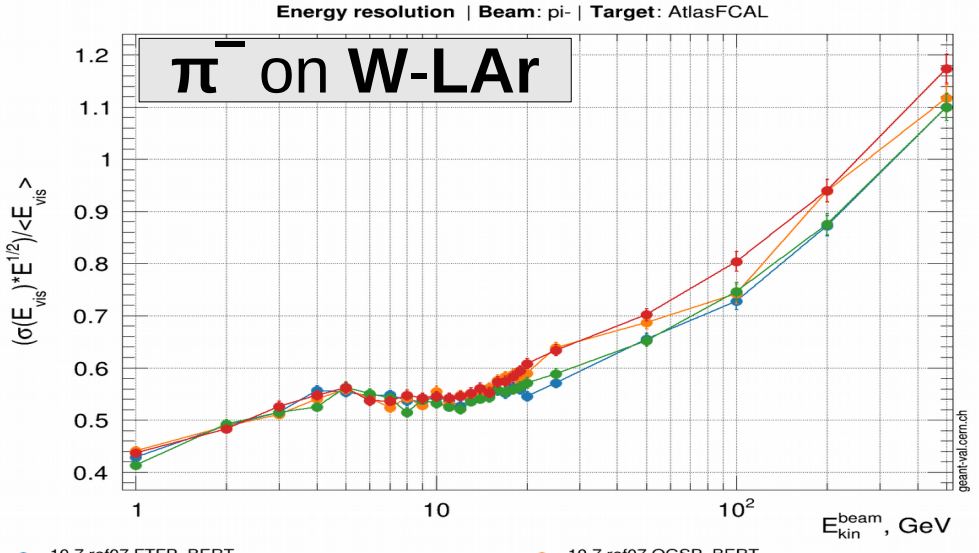


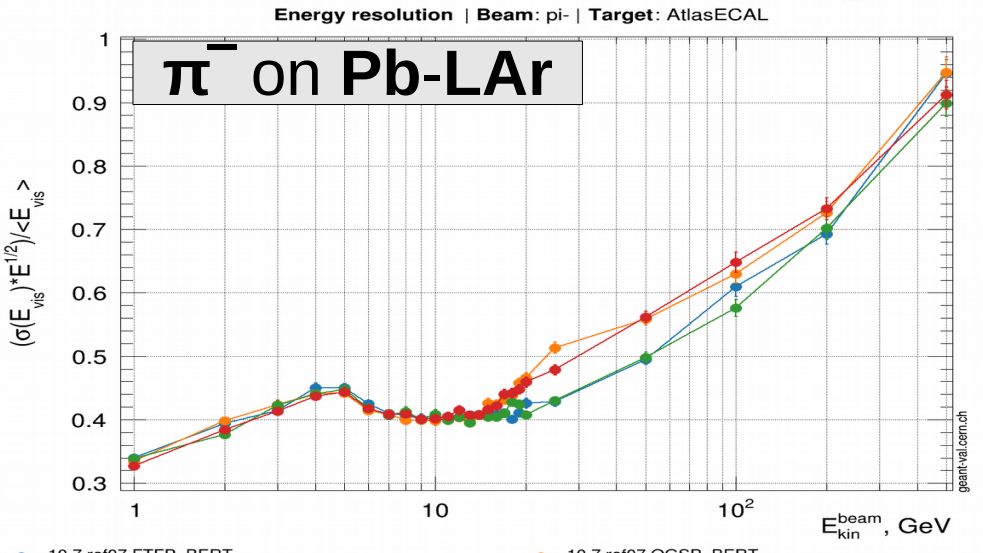


Energy Resolution

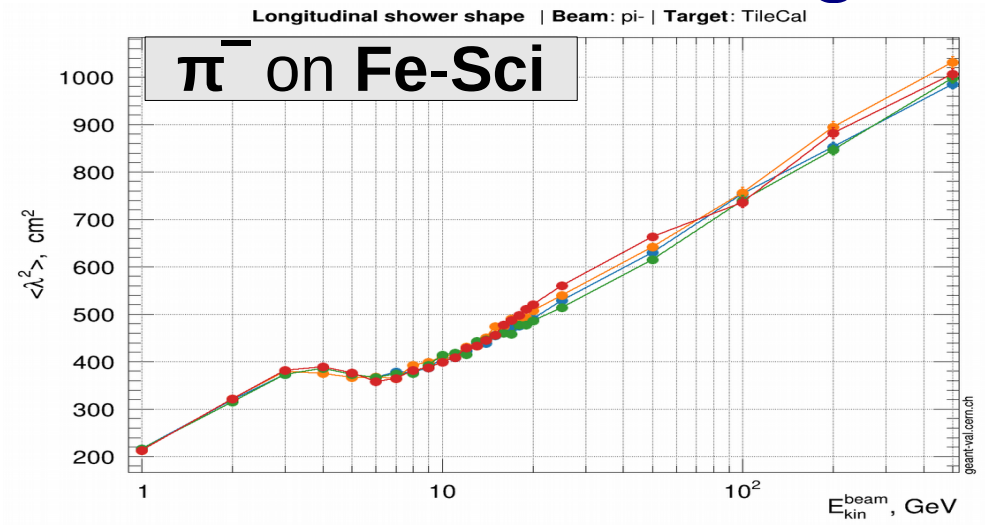


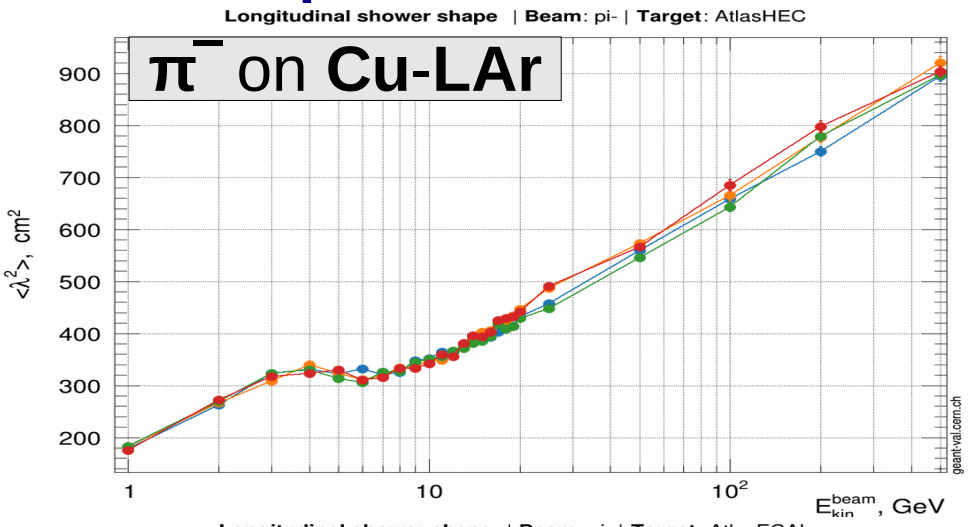


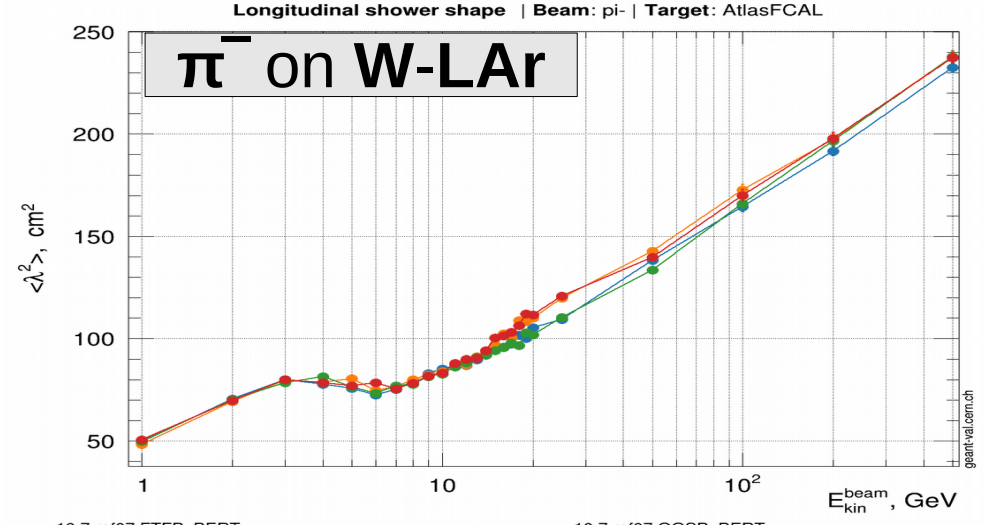


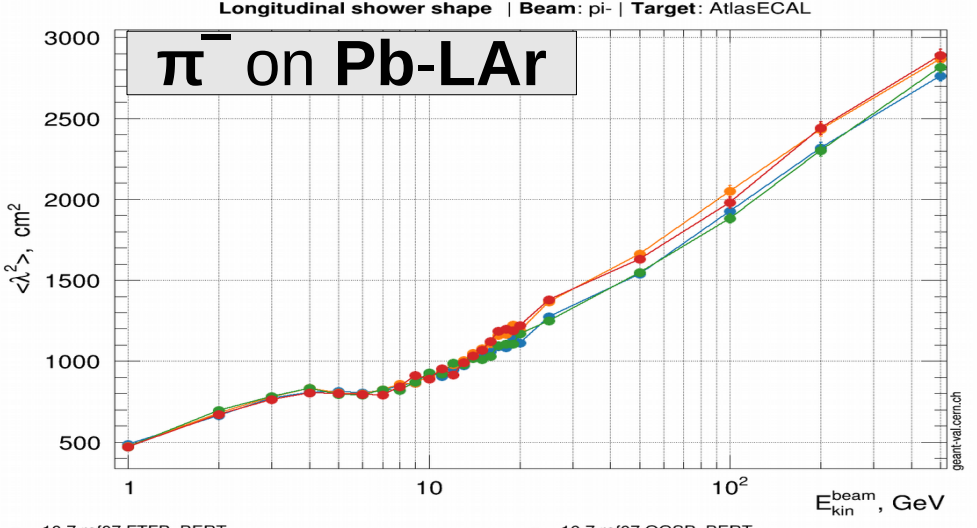


Longitudinal Shape









Lateral Shape