Recent Developments in Hadronics

V. Ivantchenko
CERN & Princeton University

27th Geant4 Collaboration Meeting, 26–30 Sept 2022

Outline

- Updates of base classes
  - G4ParticleChange
  - G4VCrossSectionDataSet
  - G4HadronicProcess
  - G4HadronicParameters
- Neutron general process
- Modifications in physics_list
- Updates in de-excitation module
  - Problems
  - Gamma transitions
  - Evaporation
  - Multi-fragmentation
- Summary and plans for 11.1

Particle Change classes clean-up

- Classes used in hadronics processes
  - G4VParticleChange – base class
  - G4ParticleChange – hadronic processes
  - G4ParticleChangeForDecay – radioactive decay
- Clean-up includes following modifications:
  - Protection against negative energy of primary or secondary
  - No limit on number of secondaries
    - SetNumberOfSecondaries(G4int) call is useful but not mandatory anymore
    - Internally std::vector is used instead of std::array
  - In G4VERBOSE mode
    - Check on secondary kinetic energy, momentum direction, time
    - Detailed printout of G4Track parameters in case of a problem
  - Different G4Exception level
    - Instead of EventShouldBeAborted now JustWarning is used

G4VCrossSectionDataSet

- Added extra interfaces to the base class for hadronic cross sections
  - virtual G4double ComputeCrossSectionPerElement(G4double kinEnergy, G4double loge, const G4ParticleDefinition* p,
    const G4Element* elm, const G4Material* mat = nullptr);
  - virtual G4double ComputeIsoCrossSection(G4double kinEnergy, G4double loge, const G4ParticleDefinition* p,
    G4int Z, G4int A, const G4Isotope* iso = nullptr, const G4Element* elm = nullptr const G4Material* mat = nullptr);
  - These methods do not use G4DynamicParticle and are useful for
    - Initialisation
    - Integral method
    - Neutron general process
- New methods without G4DynamicParticle argument are used in XS cross sections classes
  - G4NeutronInelasticXS
  - G4NeutronElasticXS
  - G4NeutronCaptureXS
  - G4ParticleInelasticXS
- Propagation to other cross section classes may be done for the next release
  - Tiny CPU improvement is expected
  - Improved capabilities for unit tests

Integral method for hadronics – G4HadronicProcess

- During a step of a charge particle cross section is changed due to the energy loss
  - In EM physics the “integral” method was developed since long time and does improve EM shower shape simulation
  - For Geant4 11.0 the method for EM was significantly updated
    - It is considered up to 3 peaks in a cross section
- The integral method is now implemented for hadronics
  - For proton and pions 3 peaks cross section shape is considered
  - For K+ one peak is considered
  - For the rest of positive hadrons and ions – increased X-section with energy is assumed
  - For the rest of negatively charged hadrons – decreased X-section with energy is assumed
  - Flags to enable/disable the integral method are available in G4HadronicParameter class
    - Separately for inelastic and elastic processes
- Tests on performance do not show significant effects on results or CPU
  - No evidence of improved shower shape for HEP yet
  - The method is the current default
  - Should affect low-energy charged hadron/ion tracking – to be confirmed

G4HadronicParameters – new parameters

```
G4bool EnableHyperNuclei() const;
void SetEnableHyperNuclei( G4bool val );
// Light hyper-nuclei may be enabled/disabled
// This flag is used both by EM and hadronic physics constructors

G4bool ApplyFactorXS() const;
void SetApplyFactorXS( G4bool val );
// Flag enabling cross section factor definition

G4int GetVerboseLevel() const;
void SetVerboseLevel( const G4int val );
// Getter/Setter of the general verbosity level for hadronics.

G4bool EnableCRCoalescence() const;
void SetEnableCRCoalescence( G4bool val );
// Boolean switch that allows to apply the Cosmic Ray (CR) coalescence algorithm
// to the secondaries produced by a string model. By default, it is disabled.
```

```
inline G4bool EnableIntegralInelasticXS() const;
inline G4bool EnableIntegralElasticXS() const;
void SetEnableIntegralInelasticXS( G4bool val );
void SetEnableIntegralElasticXS( G4bool val );
// Enable/disable integral method for main hadrons

inline G4bool EnableDiffDissociationForBGreater10() const;
void SetEnableDiffDissociationForBGreater10(G4bool val);
/// For nucleon-hadron interactions, it's not decided what to do with diffraction
/// dissociation. For the moment, they are turned off. This option allows it to
/// be turned back on. Applies to Baryon Number > 10 or # target nucleons > 10.

inline G4bool EnableNeutronGeneralProcess() const;
void SetEnableNeutronGeneralProcess( G4bool val );
// Neutron general process may be enabled/disabled
```

Neutron general process

- G4GammaGeneralProcess is adopted both by ATLAS and CMS
  - In ATLAS it provides ~4% speedup for Run3 MC production
  - In CMS it provides ~2% speedup for Run3 MC production
- G4NeutronGeneralProcess (NGP) – new combined process
  - Should optimize GetPhysicsInteractionLength(..) method
  - It includes processes
    - Neutron elastic
    - Neutron inelastic
    - Neutron capture
    - Neutron killer
  - In these processes the cross sections are used
    - G4NeutronElasticXS
    - G4NeutronInelasticXS
    - G4NeutronCaptureXS
- Expected few % speedup for HEP experiments

Some implementation details

- NGP is released inside hadronic/processes sub-directory
  - It is using only hadronic sub-libraries nothing from EM
- In NGP 3 energy intervals will be considered
  - T < 1 keV simplified cross sections
    - This threshold should be optimized
    - Low energy neutrons will be killed mainly by the time cut
  - 1 keV < T < 20 MeV
    - Detailed cross sections with many bins per decade
    - Optimized selection of process and target isotope
  - T > 20 MeV – standard HEP tables
    - Mainly element vise x-sections
  - Time cut by default 10 microsecond, may be changed via Set method
- Problems to include into reference Physics Lists
  - Solution for FTFP_BERT was found out and it is working fine
  - Nightly tests fail for several applications, which uses general biasing
    - No investigation of a problem was done
  - Temporary solution:
    - custom physics configuration may be implemented by ATLAS, CMS, and any other user application, where general biasing is not used

Modifications in hadronic physics lists

- To ensure early initialization of hadronic parameters
  - G4PhysListUtil::InitialiseParameters()
    - Called in several places of physics construction
    - Should be called in any custom physics
- To access a hadronic process on top of any reference Physics List, use methods of G4PhysListUtil
  - static G4VProcess* FindProcess(const G4ParticleDefinition*, G4int subtype);
  - static G4HadronicProcess* FindInelasticProcess(const G4ParticleDefinition*);
  - static G4HadronicProcess* FindElasticProcess(const G4ParticleDefinition*);
  - static G4HadronicProcess* FindCaptureProcess(const G4ParticleDefinition*);
  - static G4HadronicProcess* FindFissionProcess(const G4ParticleDefinition*);
  - static G4NeutronGeneralProcess* FindNeutronGeneralProcess();
- To change hadronic cross section of top of any reference Physics List, use methods of HadProcesses utility
  - static G4bool AddInelasticCrossSection(const G4ParticleDefinition*, G4VCrossSectionDataSet*);
  - static G4bool AddInelasticCrossSection(const G4String&, G4VCrossSectionDataSet*);
  - static G4bool AddElasticCrossSection(const G4ParticleDefinition*, G4VCrossSectionDataSet*);
  - static G4bool AddElasticCrossSection(const G4String&, G4VCrossSectionDataSet*);
  - static G4bool AddCaptureCrossSection(G4VCrossSectionDataSet*);
  - static G4bool AddFissionCrossSection(G4VCrossSectionDataSet*);

Problems of de-excitation module

- De-excitation module is responsible for simulation of decays of excited fragments
  - Multi-fragmentation (disabled by default)
  - Emission of light fragments (Evaporation, GEM, FermiBreakUp)
  - Gamma de-excitation including internal conversion (IC)
- De-excitation is performed step by step
  - Decay channels are selected randomly depending on probability
  - It is completed when all fragments are stable
    - Lifetime of a stable fragment should be above 1 ns
- There were several problem reported by different users
  - The most detailed analysis was done by A. Svetlichnyi and colleagues: https://indico.cern.ch/event/1106118/contributions/4693132/attachments/2376453/4059593/Kinetic_G4_Tech_Forum_Svetlichnyi.pdf
  - After 11.0ref02, QGSP_BIC in simplified calorimeter test provides energy deposition is increased by 3-4%
  - Problems were studied and number of revisions were added to the de-excitation module

Nuclear level data

- Information on a level includes
  - Excitation energy
  - Lifetime
  - Spin-parity
  - List of transitions with probabilities of internal conversion (IC)
- The list of nuclear levels is incomplete
  - For unstable fragments smaller number of measurements are available
  - Even for main isotopes there are levels with unknown properties
  - Fermi level of an unstable fragment is known approximately
- In the recent dataset we have
  - Total size in the memory 56M, without IC data – 8M
  - 171303 nuclear levels
  - 15653 levels are “floating”
  - 1263 floating levels have no known transitions
  - There are levels with identical energy
    - 984 floating levels have excitation energy equal to the previous level
    - 438 floating levels have energy equal to the next level
    - At least 65 floating levels with equal energy have different lifetime
  - There are differences in level description in radioactive decay data and photon evaporation data

Photon evaporation code updates

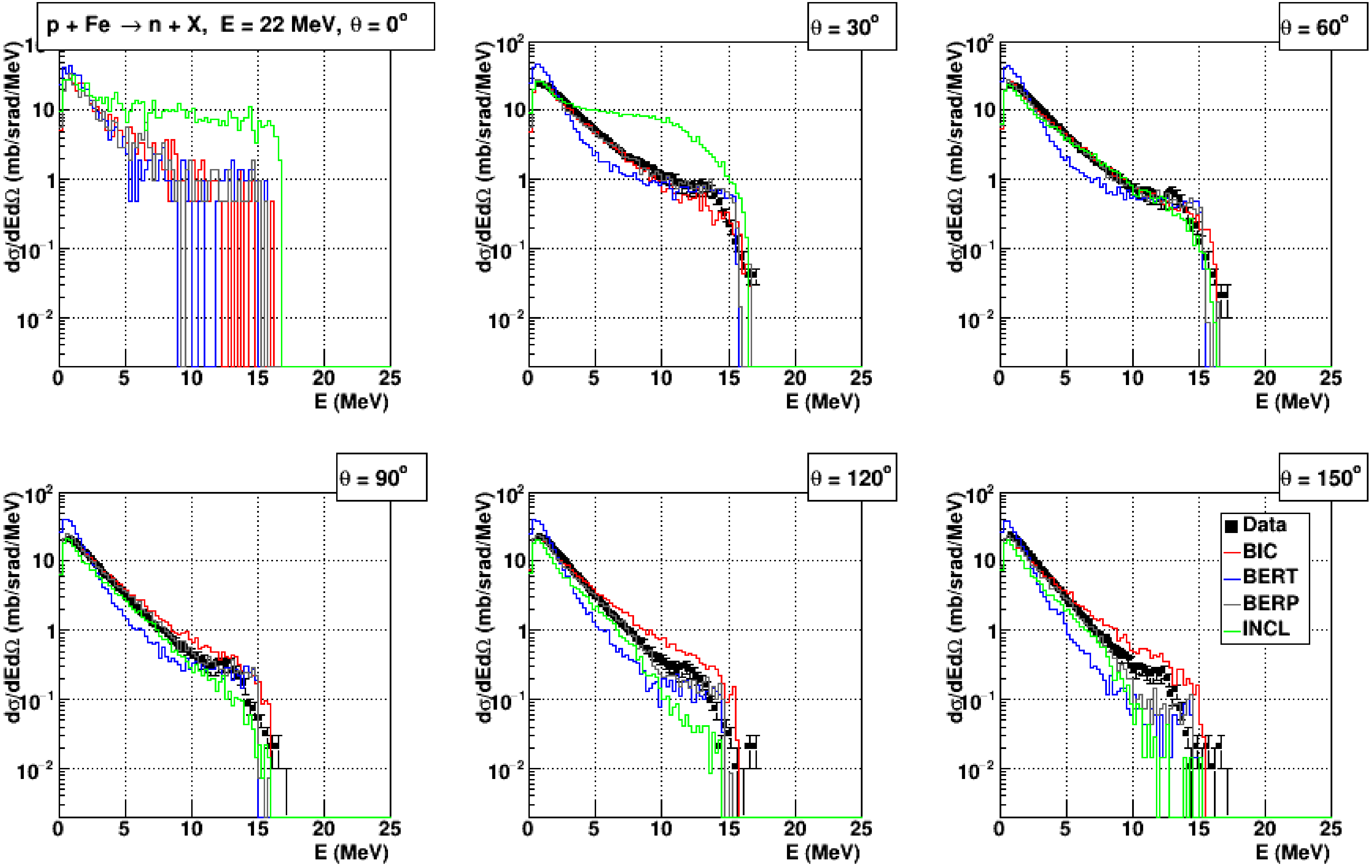
- For Geant4 11.0ref03 number of fixes were introduced
  - If initial excitation level is a floating level a search for the close normal level is performed
  - If G4Fragment has excitation energy out of tolerance=10 eV to a known nuclear level or this level has no known transitions this excitation state assumed to be continuum and not discrete
    - Continues and discrete transitions are handled differently
  - Final state in gamma de-excitation is always a known excitation level
    - The same approach for neutron and light ion emission
  - A new flag “IsLongLived()” is added to G4Fragment with the lifetime >1 ns
    - Such state is “stable” and will not be further decayed
    - This allows transportation and radioactive decay of such isotopes
- The situation was improved with 11.0ref04
  - No fake states are produced
  - Bugs in radioactive decay channels are fixed
- Bug reports on incorrect probabilities of the internal conversion and on correlated gammas are not yet addressed

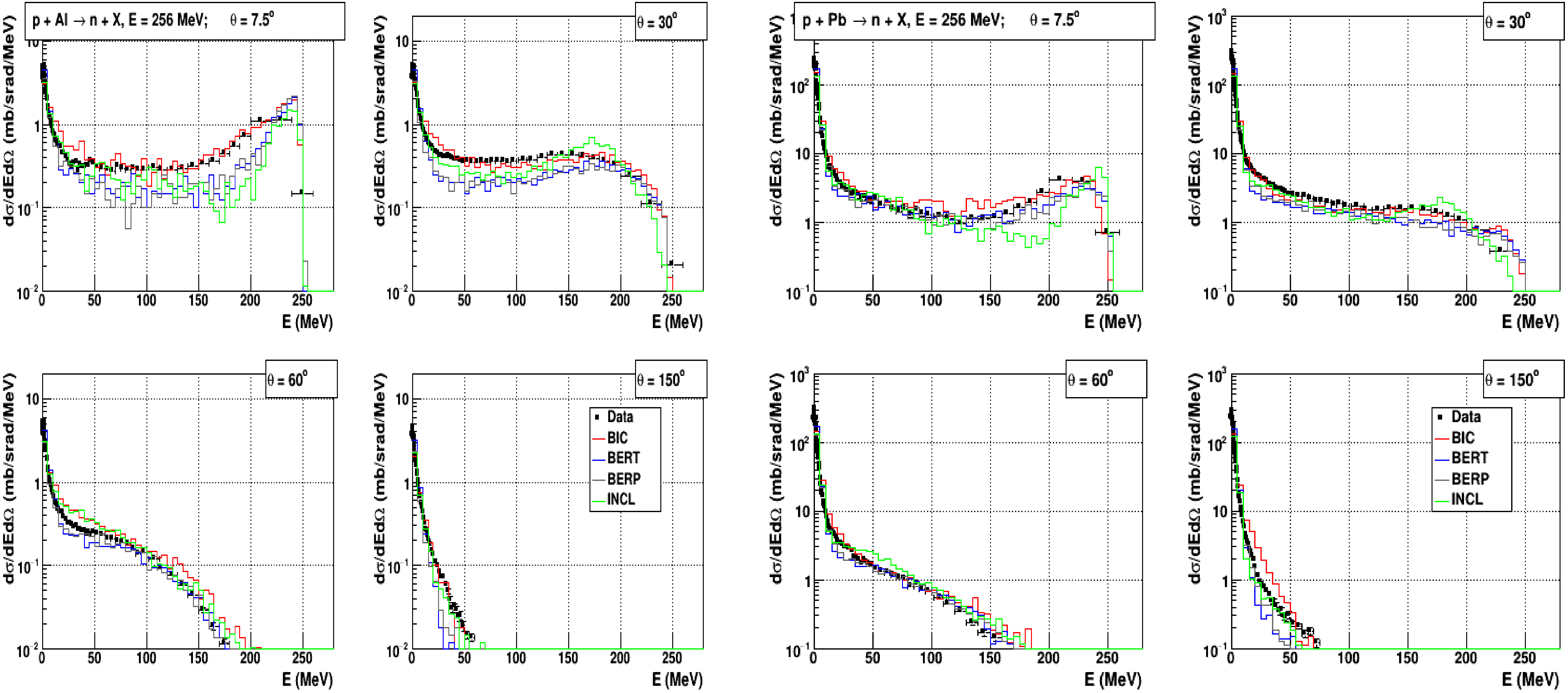
Evaporation code was improved

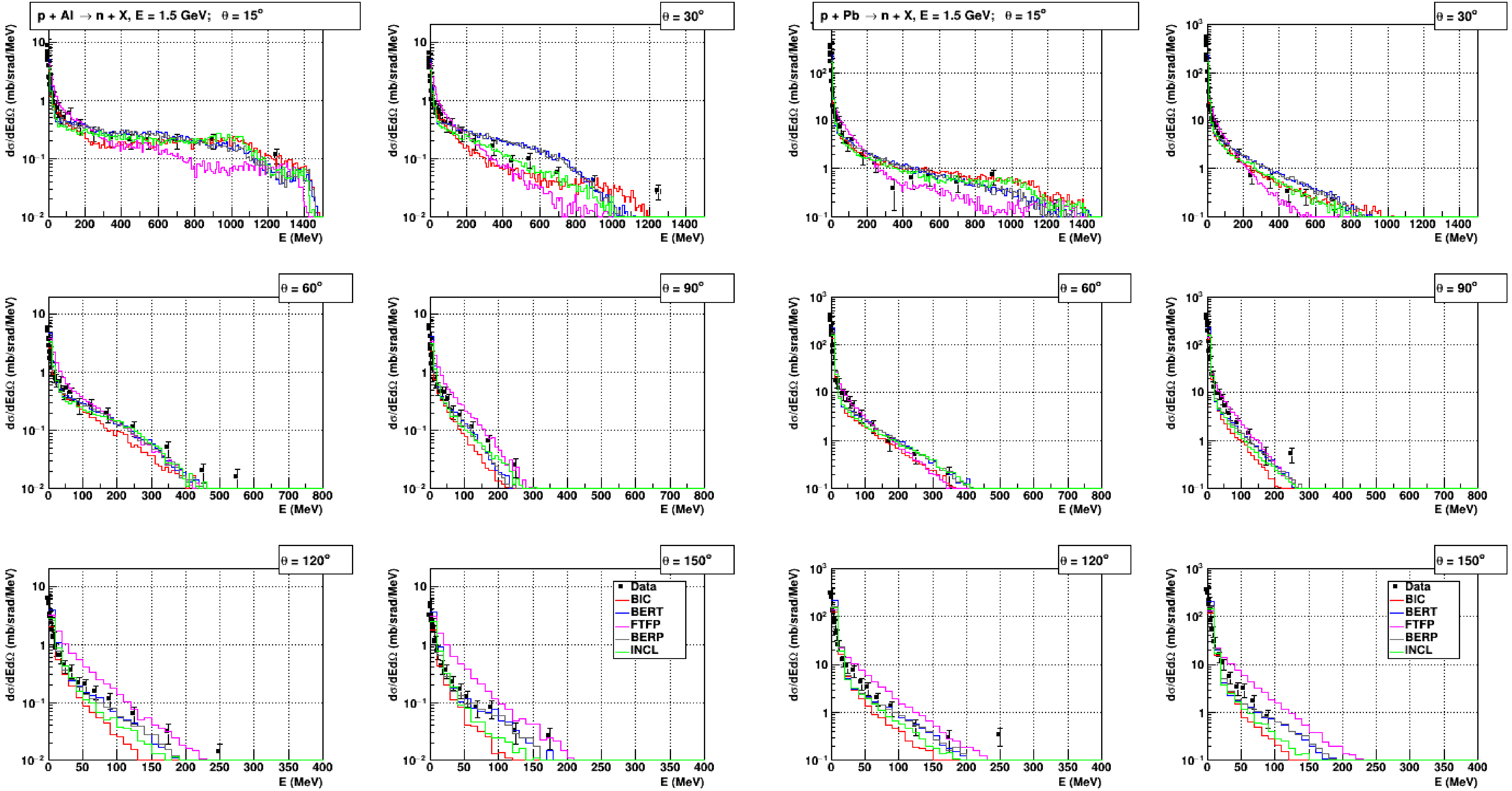
- Algorithm of numerical integration of the differential probability of light fragment emission was significantly updated
  - Step of integration is reduced and optimized
    - More computations of differential x-sections
  - During integration over emitted fragment kinetic energy the cross section is analyzed, maximum cross XS and parameters of exponential tail are identified
  - Sampling of final state is updated:
    - limit on number of attempts is increased from 100 to 1000
    - Sampling of final kinetic energy is performed more effectively using two areas
      - Const and exponential regression
- Inverse cross section expression used in computation of emission probability is revised
  - For charged fragment emission an extra factor is used for kinetic energy
    - (0.5 – 1)*CB (Coulomb Barrier)
    - similar function was introduced by V. Grichine for ion-ion x-sections
- Formulas used in evaporation code were verified
  - It was confirmed that they correspond to prescriptions of publications used in these models
- Few problems in multi-fragmentation code were fixed
  - Not yet recommended for users

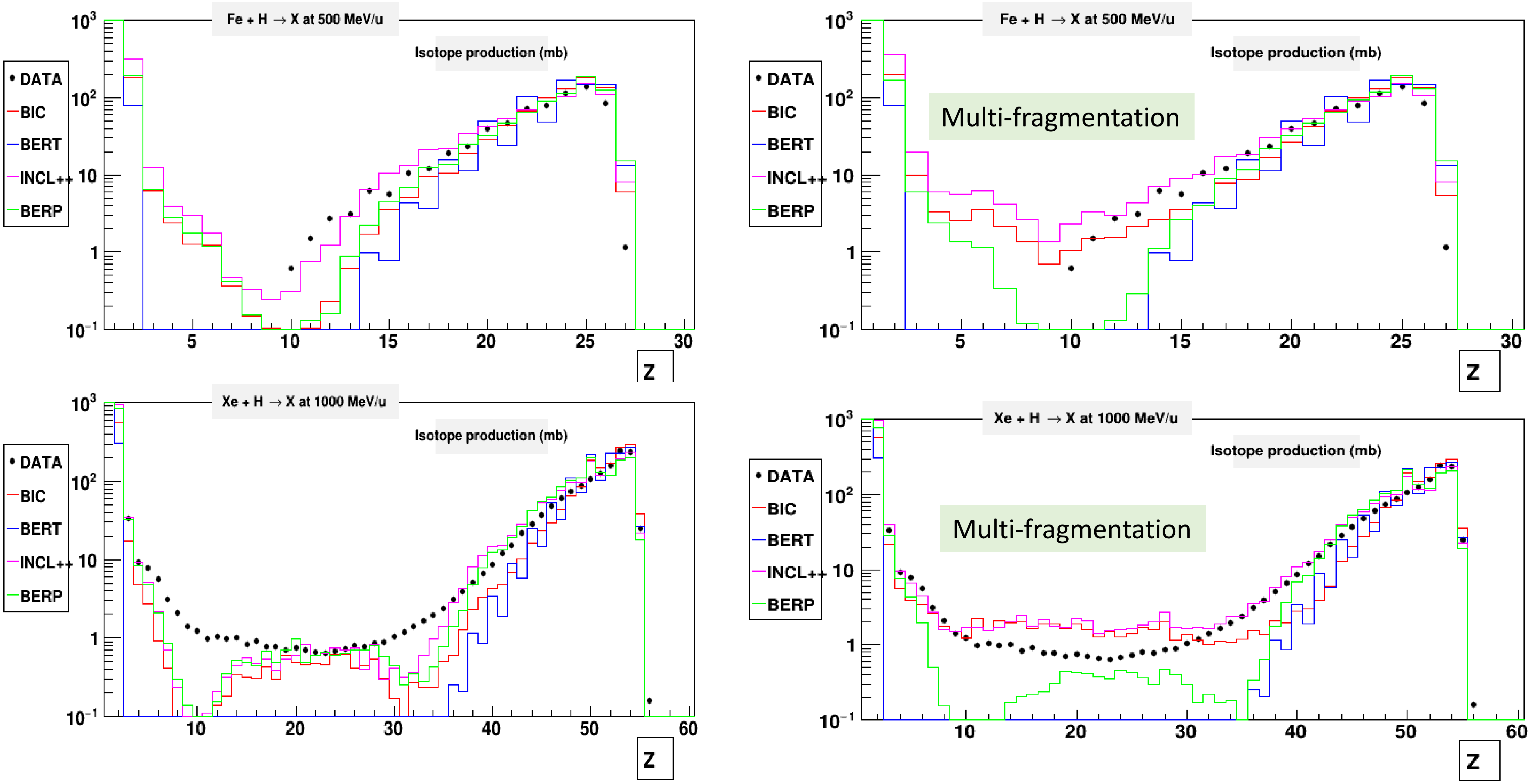
Test results for 11.0.7

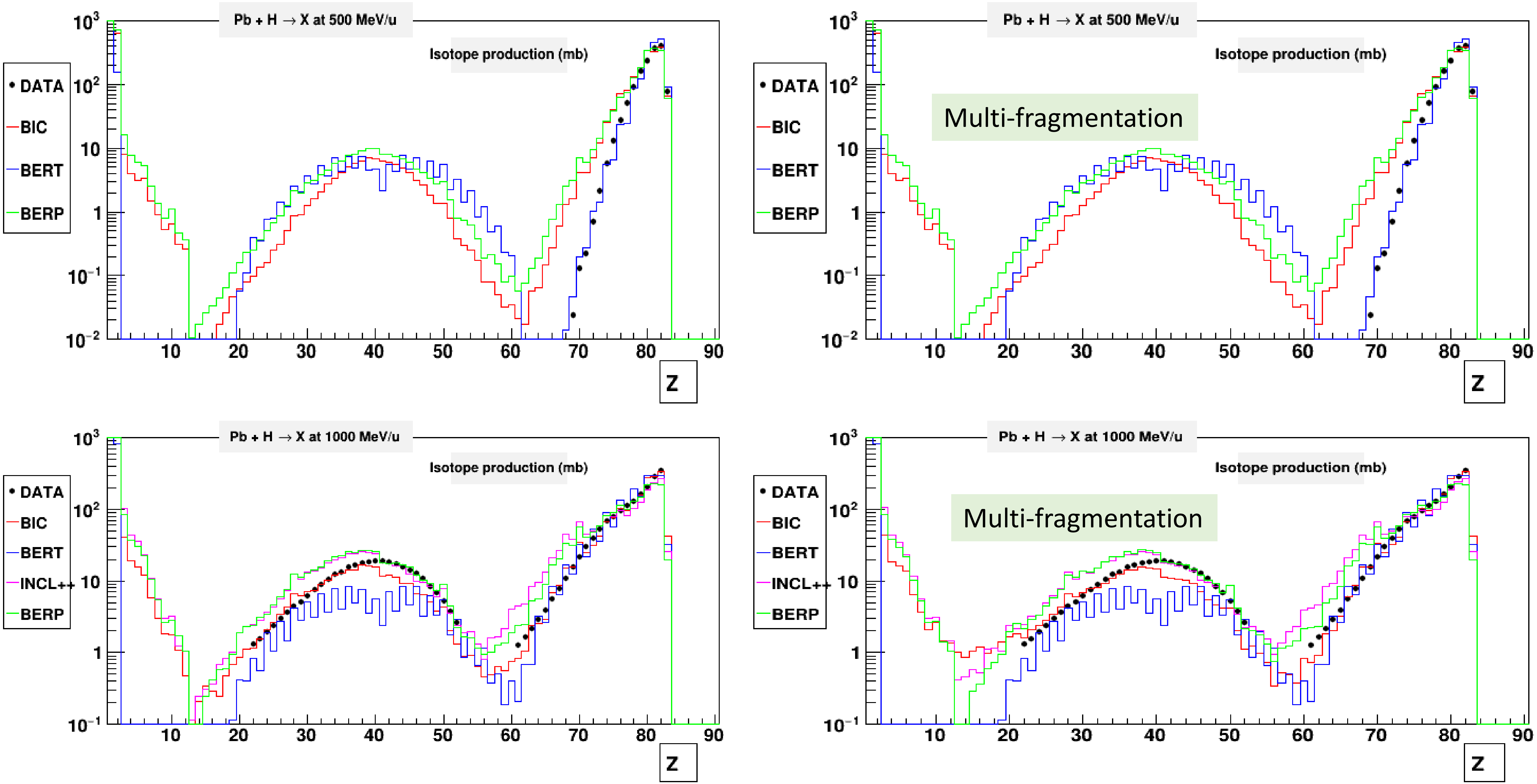
- Full set of results for test30
  - http://vnivanch.web.cern.ch/vnivanch/hadronic/test30/geant4-11-00-ref-07/
- Isotope production results for IAEA benchmark
  - http://vnivanch.web.cern.ch/vnivanch/hadronic/iaea/geant4-11-00-ref-07/


p + Fe → n + X, E = 22 MeV, θ = 0°
θ = 30°
θ = 60°
θ = 90°
θ = 120°
θ = 150°
dσ/dEdΩ (mb/srad/MeV)
E (MeV)
Data
BIC
BERT
BERP
INCL


p + Al → n + X, E = 256 MeV; θ = 7.5°
θ = 30°
θ = 60°
θ = 150°
p + Pb → n + X, E = 256 MeV; θ = 7.5°
θ = 30°
θ = 60°
θ = 150°
dσ/dEdΩ (mb/srad/MeV)
E (MeV)
Data
BIC
BERT
BERP
INCL


p + Al → n + X, E = 1.5 GeV; θ = 15°
θ = 30°
p + Pb → n + X, E = 1.5 GeV; θ = 15°
θ = 30°
θ = 60°
θ = 90°
θ = 60°
θ = 90°
θ = 120°
θ = 150°
θ = 120°
θ = 150°
dσ/dEdΩ (mb/srad/MeV)
E (MeV)
Data
BIC
BERT
FTFP
BERP
INCL


Fe + H → X at 500 MeV/u
Isotope production (mb)
DATA
BIC
BERT
INCL++
BERP
Z
Fe + H → X at 500 MeV/u
Isotope production (mb)
Multi-fragmentation
Xe + H → X at 1000 MeV/u
Isotope production (mb)
Xe + H → X at 1000 MeV/u
Isotope production (mb)
Multi-fragmentation


Pb + H → X at 500 MeV/u
Isotope production (mb)
DATA
BIC
BERT
BERP
Z
Pb + H → X at 500 MeV/u
Isotope production (mb)
Multi-fragmentation
Pb + H → X at 1000 MeV/u
Isotope production (mb)
DATA
BIC
BERT
INCL++
BERP
Z
Pb + H → X at 1000 MeV/u
Isotope production (mb)
Multi-fragmentation

Summary and plans for 11.1

- New features are available for 11.1
  - Neutron general process
  - Integral method for hadrons
- De-excitation module improvements
  - Resolved problems introduced in previous releases
    - Levels without data on transitions
    - Floating levels
    - Sampling of evaporation of neutrons and light ions
  - Energy response for hadronic calorimeters are back to normal
- Short term plan for 11.1
  - Establish 1 ns time limit for all components of hadronic physics
  - Do not enable multi-fragmentation by default
  - Check Fermi BreakUp model
  - To add optional X-sections and fragmentation of light hyperneuclei