Developments in INCL/ABLA (Oct'09)

Pekka Kaitaniemi
pekka.kaitaniemi@gmail.com

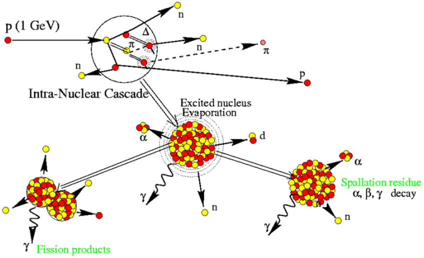
Helsinki Institute of Physics
CEA/Saclay

October 19th, 2009
Revision: c5fa648ab4f5b7bfe25f3290c98413de1549e499

INCL intra-nuclear cascade and ABLA de-excitation

Projectile	p, n, π, deuteron, triton, He3, alpha
Energy range	150 MeV - 3 GeV
Target nuclei	Carbon - Uranium

Table: Model validity range



Interactions (isospin dependence):

NN ⟶ NN

NN ⟶ NΔ

NΔ ⟶ NN
NΔ ⟶ NΔ
ΔΔ ⟶ ΔΔ

No NΔ ⟶ ΔΔ
Δ = Δ33 (1232 MeV)
(No other baryonic resonances)
No πN ⟶ πN
No πN ⟶ 2πN ,
but Δ ⇄ πN

- INCL tracks particles (p, n, π, Δ) and their binary collisions.
- Physics list `QGSP_INCL_ABLA`.
- Extended projectile support up to Carbon.

Use-cases and implementation

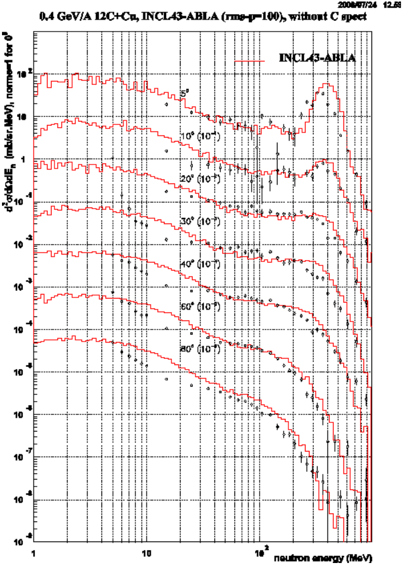
Use-cases:

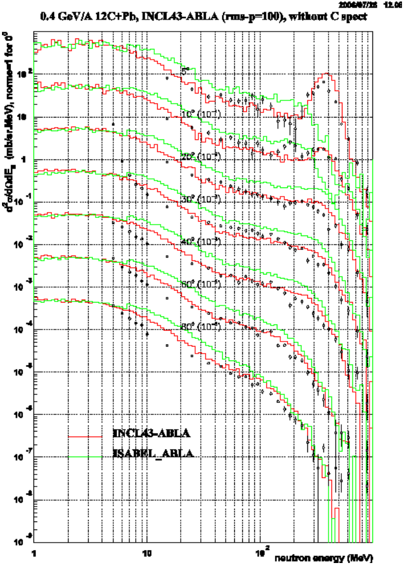
- Treatment of tumors by using light ion beams
- Radioprotection of manned spacecrat irradiated by cosmic rays
- Damage to spacecraft electronics

Implementation:

- Projectile consists of A nucleons
- Projectile nucleons that don't hit the nucleus form a "spectator nucleus" (de-excitation using Fermi break-up)
- Target treated using either ABLA or Fermi break-up

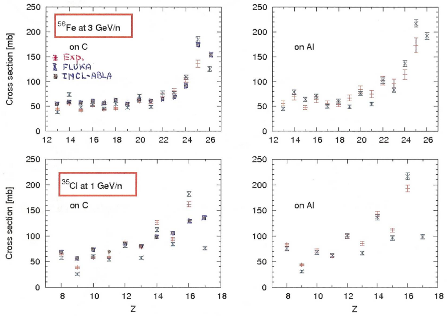
INCL4.3: Carbon-12 @ 400 MeV/nucleon





INCL4.3: Fragmentation, comparison against Fluka

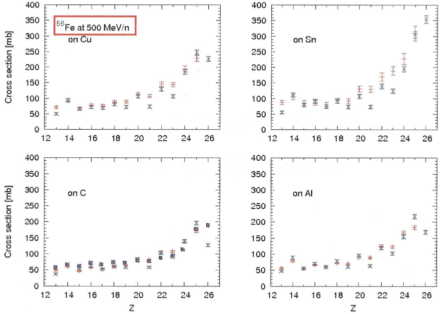
Fragmentation: 3 GeV/n Fe, 1 GeV/n Cl ions



Exp. and MC (FLUKA) charge fragment cross sections for 3 GeV/n Fe and 1 GeV/n Cl ions (exp. data from http://fragserver.lbl.gov/main.html)

Alfredo Ferrari, NUFRA2007

Fragmentation: 500 MeV/n Fe ions



Exp. and MC (FLUKA) charge fragment cross sections for 500 MeV/n Fe ions (exp. data from http://fragserver.lbl.gov/main.html)

Alfredo Ferrari, NUFRA2007

Availability in Geant4 and future plans

Carbon projectile support in INCL will be available for Geant4 9.3 release.

Short term:

- Various approaches for the beam spectators (excitation energy at large impact parameters).
- More symmetry between the target (light) and the beam
- Tests on complex systems (transport part with GEANT4 and Hadrontherapy)

Long term:

- Strong focus on improving the light ion support
- A more radical transformation of the code is needed (ion-ion part to be done by D. Mancusi)...

INCL OO redesign goals

Goals:

- Only one INCL physics model
- Develop the next generation of the INCL codebase
- Modularize design, make the system extensible
- Get rid of old cruft
- Better (framework) support for nucleus-nucleus collisions (D. Mancusi, possibly starting in the summer of 2010)

Traditionally INCL has been a standalone code and we have provided interfaces to transport codes:

- C++ translation and Geant4
- FORTRAN version and LAHET

INCL++ initialization

- Nucleus: a collection of nucleons

```
G4INCL::Nucleus *theNucleus = new G4INCL::Nucleus(6,
12);
theNucleus->initializeParticles();
```

- Propagation model: predicts the collisions (a.k.a. avatars) between particles and transports the particles inside the nucleus.

```
G4INCL::Propagation::IPropagationModel *pm
= new G4INCL::Propagation::StandardPropagationModel();
```

- Assign the nucleus to the propagation model:

```
pm->setStoppingTime(70.0);
pm->setNucleus(theNucleus);
```

INCL++ avatar processing loop

1. Get the avatar with the smallest time and propagate particles to that point in time.
   `G4INCL::IAvatar *avatar = pm->propagate();`
2. Channel is responsible for calculating the outcome of the selected avatar:
   `G4INCL::IChannel *channel = avatar->getChannel();`
3. Final state tells what changed:
   `G4INCL::FinalState *finalState = channel->getFinalState();`
4. So now we must give this information to the nucleus:
   `theNucleus->applyFinalState(finalState);`
5. If stopping time has not yet been reached, return back to step 1.

Conclusion and outlook

The new code is validated against the original FORTRAN by checking the values of the internal INCL variables

- Particle positions
- "Avatar" times
- Found collisions

The avatar search routine shown before (the core of INCL) works.

Next milestones:

- Adding NN cross sections
- Realistic elastic NN collisions

Produce a simplified cascade model prototype with the new code early 2010. The code should be able to do more realistic physics by next summer.