LLNL's nuclear data infrastructure

Presented to CSEWG:

3 Nov 2015

Bret Beck

Lawrence Livermore National Laboratory

LLNL-PRES-??????

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

LLNL nuclear data infrastructure

- FUDGE
  - Package that handles all LLNL nuclear data management, updating and processing
  - Started around 2002
  - For Updating Data and Generating ENDL
  - For Updating Data and Generating Evaluation
  - Top level in Python – allows for scripting and interactive running

- Have added support for GND
  - In fact, this is now the main focus for FUDGE

What can FUDGE do

- Read/write ENDL/ENDF/GND data
  - Two-way translations between ENDF and GND
  - Internally, FUDGE handles GND (and ENDL)
- Print
- Plot
- Manipulation
- Check
  - ENDF to GND does a rigorous ENDF format check
  - Lots of other checks of GND data
- Process data
  - The rest of this talk will mainly be about processing

Progress in the last year

- We have a nuclear data crisis at LLNL that consumed much time
  - Trouble comes from converting ENDF to ENDL. Basically, our ENDL format is too limiting.
- GND
  - Updating for comments from SG38 and our ideas
  - FUDGE is updated to track GND
- In my opinion, both GND and FUDGE are becoming more object oriented
  - In part, because we are getting advice from computer scientist
  - I view this as a good thing – it leads to less code

What is processing of nuclear data?

- I use the definition of: “any manipulation of evaluated nuclear reaction data”

- This includes
  - cross section reconstruction, heating of cross sections, calculating average product outgoing energy, transfer matrix calculation ...

$$\left\langle E'(E)\right\rangle = m(E)\int dE'\, E' \int d\mu\, P(E \rightarrow E', \mu)$$

This quantity is stored in ENDL and GND.

Comment about particles

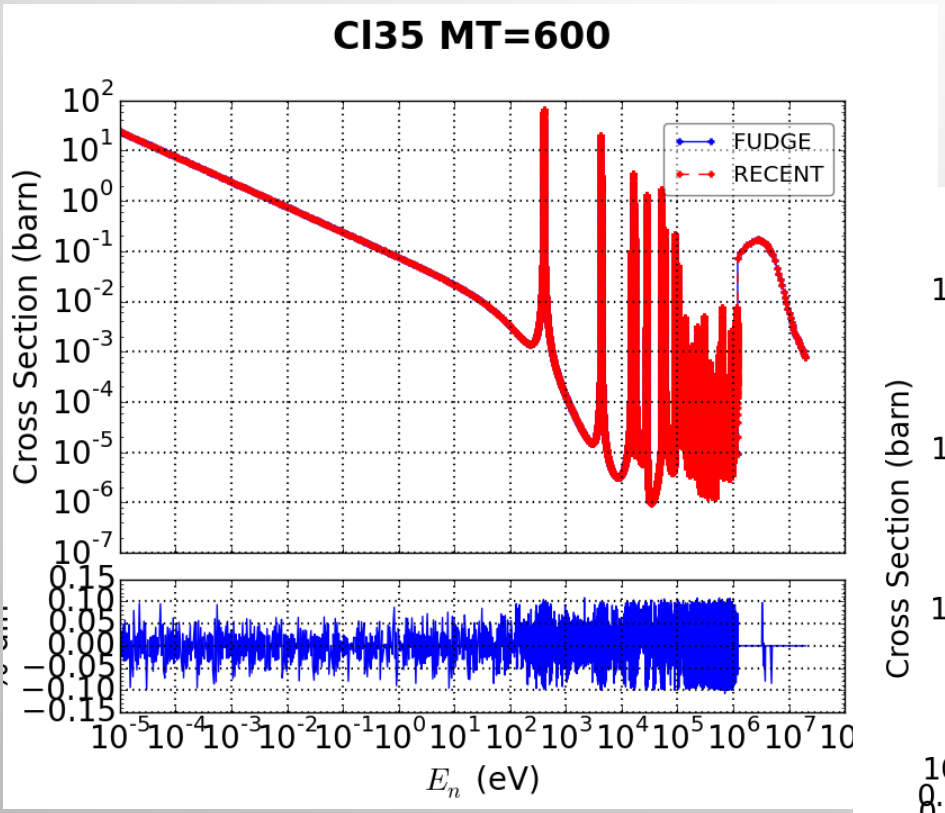
- Historically, LLNL via ENDL supported the transport of 7 light particles
  - neutron, proton, deuteron, triton, helium-3, alpha and gamma
  - In ENDL, if one of these 7 products is in a reaction, its multiplicity/ distribution must be given
- This includes
  - Processing (deterministic, Monte Carlo and thermal nuclear)
    - Except Coulomb scattering as it is not a localized collision
  - Transport in production codes
- With GND, FUDGE supports processing of “any” localized collision with “any” input and output particles
  - For a reaction, evaluator must provide cross section and product data (i.e., multiplicity and distribution)

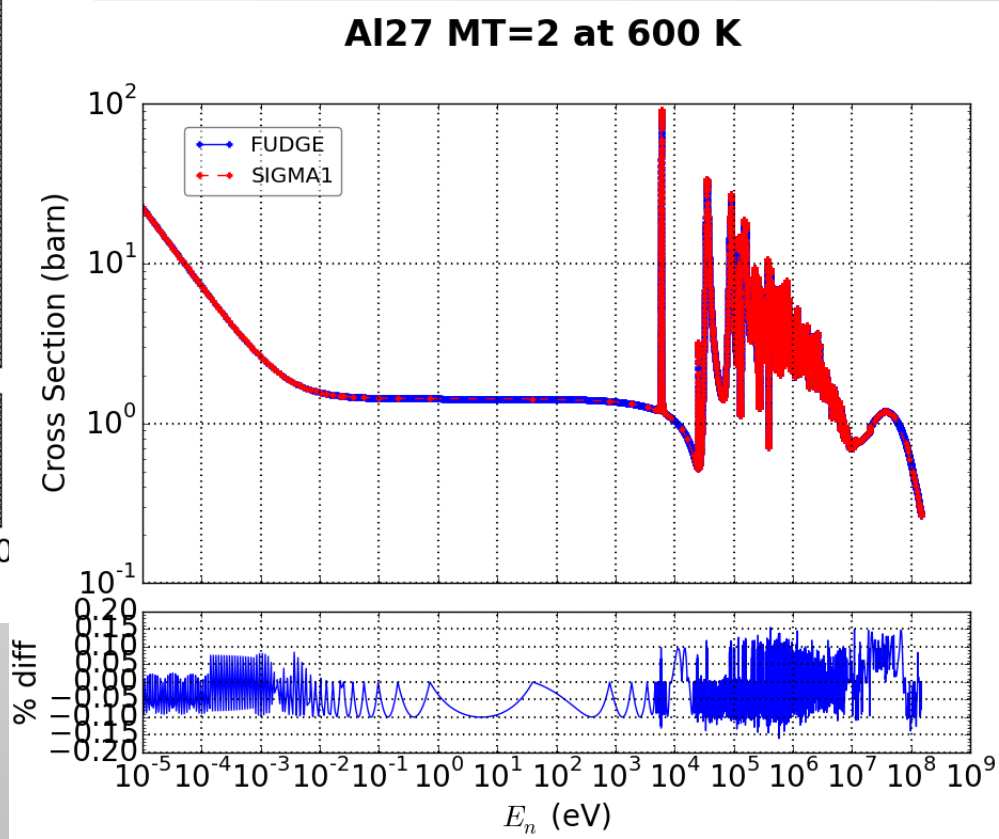
Processing FUDGE will support

- Cross section reconstruction from resonances
  - Completed and tested (except for Adler-Adler)
- Cross section heating
  - Completed and tested
- Product average energy
  - In testing.
- Deterministic (transfer matrices)
- Monte Carlo
- Thermo-nuclear – averaging over projectile temperature
- Realization (mean + uncertainty) via Kiwi

All of these can be stored in GND simultaneously.

Cross section reconstruction and heating





Investigating one small difference in heating.

FUDGE product average energy processing

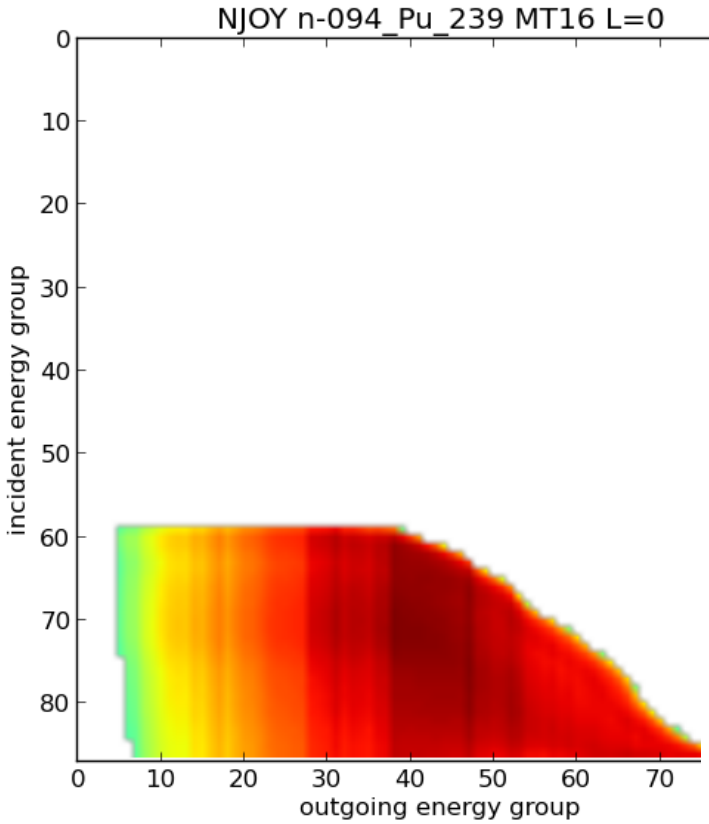
- For any product with a multiplicity and distribution (except Coulomb scattering) the average energy to the product as a function of projectile energy is calculated and stored.
- At LLNL, deposition energy is calculated at user code run time
  - That is, a KERMA-like quantity is not stored but is calculated at run time based on the particles being transported
  - User codes tell us what particles they are transporting and we calculate total incoming energy (projectile kinetic energy minus Q(E)) and subtract transported product energies.

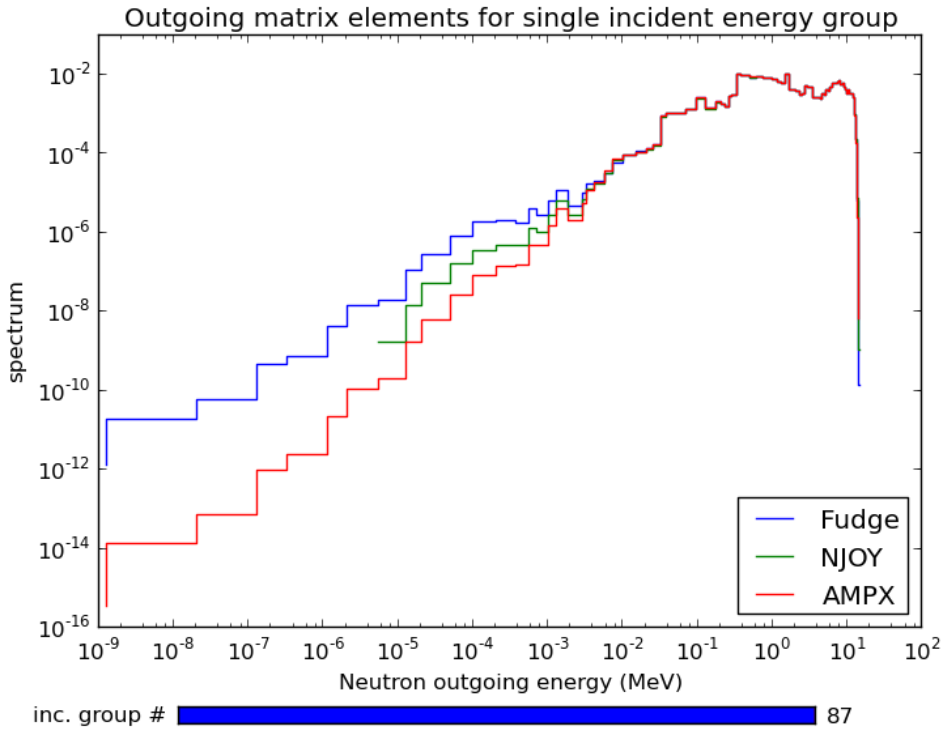
FUDGE calculates each product's average energy and not a KERMA-like quantity. This is stored in the GND file.

FUDGE deterministic processing

- Grouping one-d quantities
  - cross section, multiplicity, product average energy
  - Completed
- Computing transfer matrices
  - All ENDF product distributions supported
  - Relativistic effects can be enabled for 2-body reactions
  - Completed
  - Thermal upscattering model for constant elastic scattering of a neutron is in FUDGE/ENDL, still to be implemented in FUDGE/GND
- Still to implement
  - Self-shielding
    - Needs to be defined
  - Thermal neutron scattering

Comparison to NJOY and AMPX





FUDGE Monte Carlo processing

- For MC, LLNL philosophy is to limit pre-processing to computationally intensive tasks, save faster computations to be done at load time by the access code GIDI to not limit users
- Pre-processing
  - Reconstruct resonances, heat cross section
  - Probability tables and multi-band still to be implemented
- At load time processing
  - Grouping, cdf from pdf, equal probable binning, etc.
- In the future, at run time processing
  - Heating cross sections, maybe others?
- We will implement these in FUDGE and store in GND later. Allows for viewing and testing.

FUDGE Thermal nuclear processing

- Averages data over projectile energy assuming a Maxwellian projectile and target distributions
  - Used in NIF simulations
- Stored at various temperatures
- Have code for ENDL, need to implement into FUDGE/GND

What is FUDGE missing

- Support for neutron thermal scattering
  - This needs to be better defined in GND. Working with others including D. Roubtsov.
- Self-shielding
  - Unresolved resonance region probability table
  - multi-band support
- Others?

Summary

- FUDGE will be released within a month
  - Will not include transfer matrix processing
  - Released under BSD licensing
- Another version of FUDGE will be released early next year
  - Will include transfer matrix processing
- GIDI
  - GND access library for transport codes
  - Already being tested in Mercury and GEANT4 (MC codes)
  - Deterministic version being designed