Exercise: source routine

First practice the “new” source routine

Starting Flair project

Based on the basic template

Geometry

- Target is removed
- Everything is in vacuum
- An ideal sphere (R=10cm) enclose the beam origin located

Scoring

- **`USRBIN`** scoring of **`ALL-PART`** fluence – To see where the beam goes
- **`USRBDX`** scoring of **`ALL-PART`** crossing the sphere – To see the beam spectrum

Exercise 1 – Adding the source routine

In this exercise we will add the default “new” source routine to a Flair project and test if it is working correctly.

1. Start with the provided template
2. Copy the “source_newgen.f” user routine to the project
3. Compile the custom executable
4. Add an empty **`SOURCE`** card to the input
5. Verify that the custom executable is selected on the [Run] tab for the *run/source_routine* simulation
6. Run the simulation
7. Verify that the source routine is loaded:

   Search for “**`=== User written source: ===`**“ in the output file

Exercise 2 – Simple beam

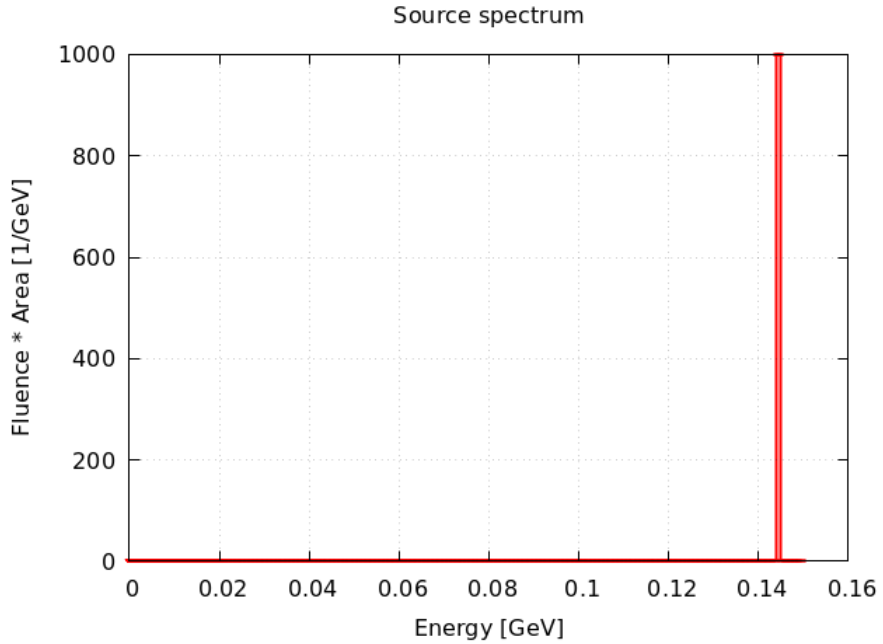
Proton beam with:

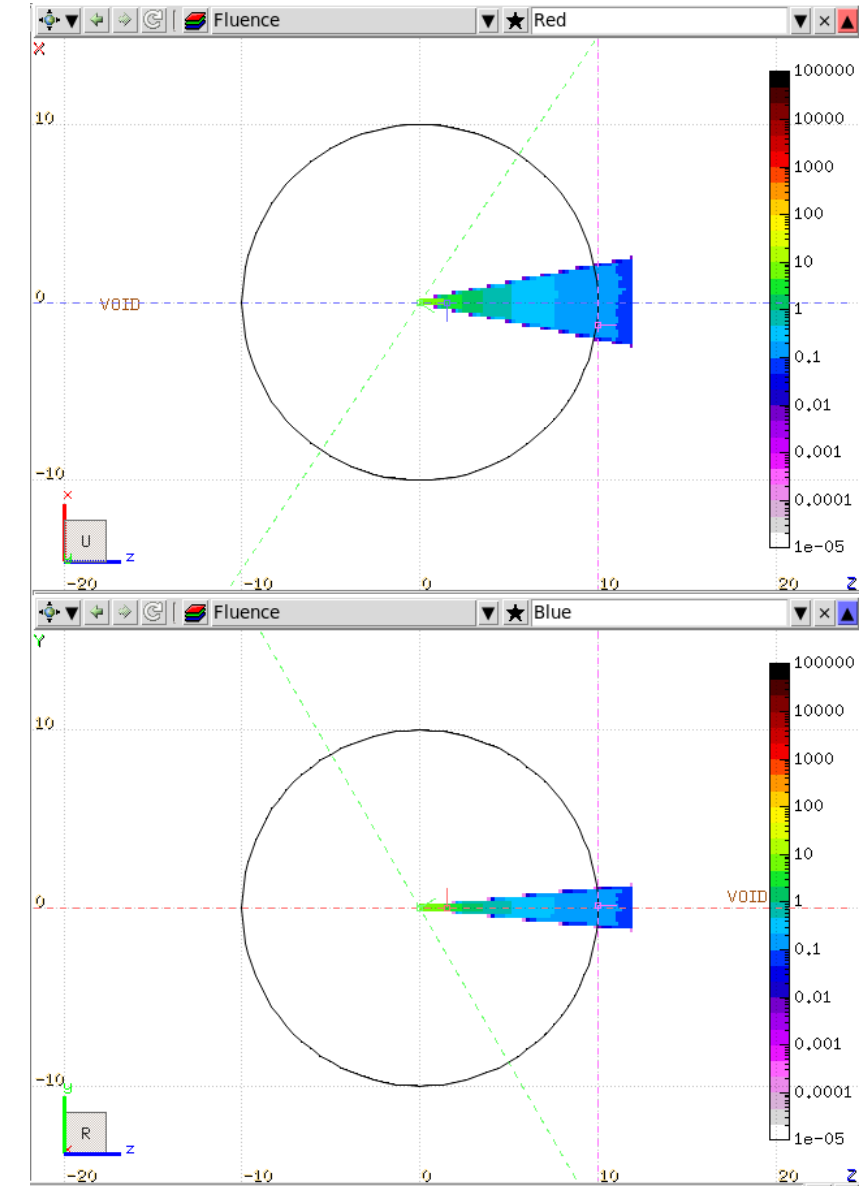
- 145 MeV energy
- Flat divergence:
  - X (X-Z plane): 400 mrad
  - Y (Y-Z plane): 200 mrad

1. Set the particle energy and type on the **BEAM** card
2. Notice that only one divergence value can be set on the **BEAM** card
   For different divergences, the source routine must be used
3. Using any text editor set *divergence_x* and *divergence_y* variables in the source routine. Note:
   a. Use double precision formatting for numbers
   b. The unit is [radians] in the source routine, while [mrad] on the **BEAM** card

Exercise 2 – Simple beam

4. Recompile your custom executable
5. Rerun the simulation (use 1e5 primaries / cycle)
6. Process the data
7. Verify the divergences on the [Geometry] tab
8. Plot the spectrum of the beam on the [Plot] tab





Exercise 3 – Two simultaneous beam

A proton beam with 100 MeV, and an electron beam with 50 MeV energy

Intensity ratio: 1:3 (25% - 75%)

1. (Remove / comment out the divergences)
2. Declare a double precision variable and assign a random number to it.
   Use: … **`= FLRNDM (xdummy)`**
3. Use "*if*" / "*else*" / "*end if*" logic to select between the two beams
   1. If my random variable is less than 0.25, then it is proton beam
   2. Else it is an electron beam
4. Set the following parameters for both beams:
   a. *`particle_code`* (See FLUKA manual for particle codes)
   b. *`momentum_energy`*

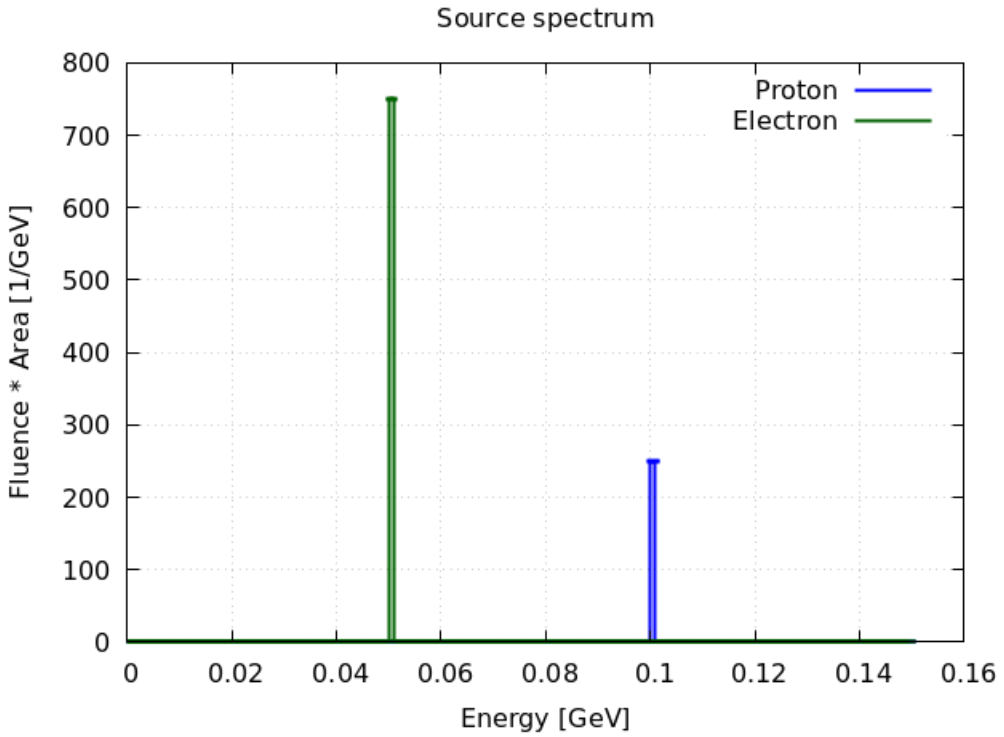
Exercise 3 – Two simultaneous beam

5. Set *energy_logical_flag* to ***.true.***
   To make sure we sample energy and not momentum
6. Enable debugging by setting the *debug_logical_flag* to ***.true.***
   (at the end of the file)
7. Compile your custom executable
8. Clone the **URSBDX** scoring twice, and set the new ones to score PROTON and ELECTRON fluence
9. Run the simulation
10. Process the data
11. Check the .log file for the debug output
12. Plot the results using the [Plot] tab (clone or edit the existing plot as needed)

Exercise 3 – Two simultaneous beam

```
source_newgen.f - Debug output
------------------------------
Particle -E [GeV]        X [cm]          Y [cm]          Z [cm]          Cosx            Cosy            Cosz            Weight
       1 -1.0000000E-01  0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   1.0000000E+00   1.0000000E+00
       3 -5.0000000E-02  0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   1.0000000E+00   1.0000000E+00
       3 -5.0000000E-02  0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   1.0000000E+00   1.0000000E+00
       3 -5.0000000E-02  0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   1.0000000E+00   1.0000000E+00
       1 -1.0000000E-01  0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   1.0000000E+00   1.0000000E+00
       3 -5.0000000E-02  0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   1.0000000E+00   1.0000000E+00
       3 -5.0000000E-02  0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   1.0000000E+00   1.0000000E+00
       3 -5.0000000E-02  0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   0.0000000E+00   1.0000000E+00   1.0000000E+00
```

…



Exercise 4 – Using a sampling subroutine

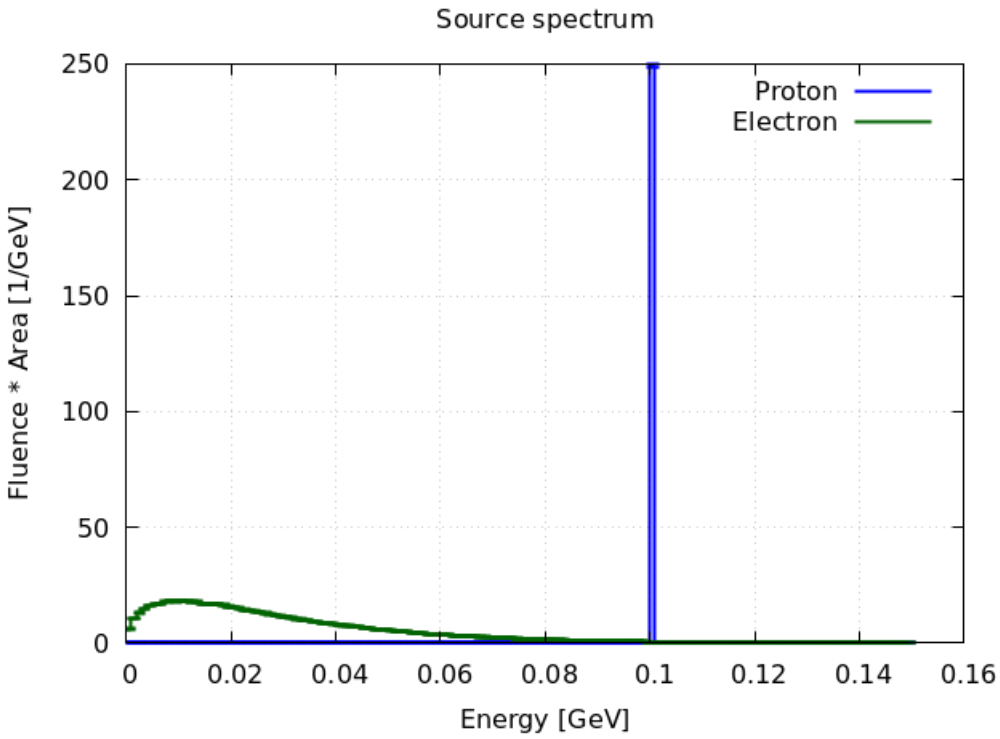
We keep the proton beam as is, but we will sample the energy of the electrons according the Maxwell-Boltzmann distribution

1. Use the `sample_maxwell_boltzmann_energy()` function to sample the electron energy
2. Set the temperature to 20 MeV
3. Compile / Run / Process the results
4. Plot the results

For extra points:

What happens, if you don't use double precision format for the temperature?

(Check the debug output in the log file)



Exercise 5 – Passing values to the source routine

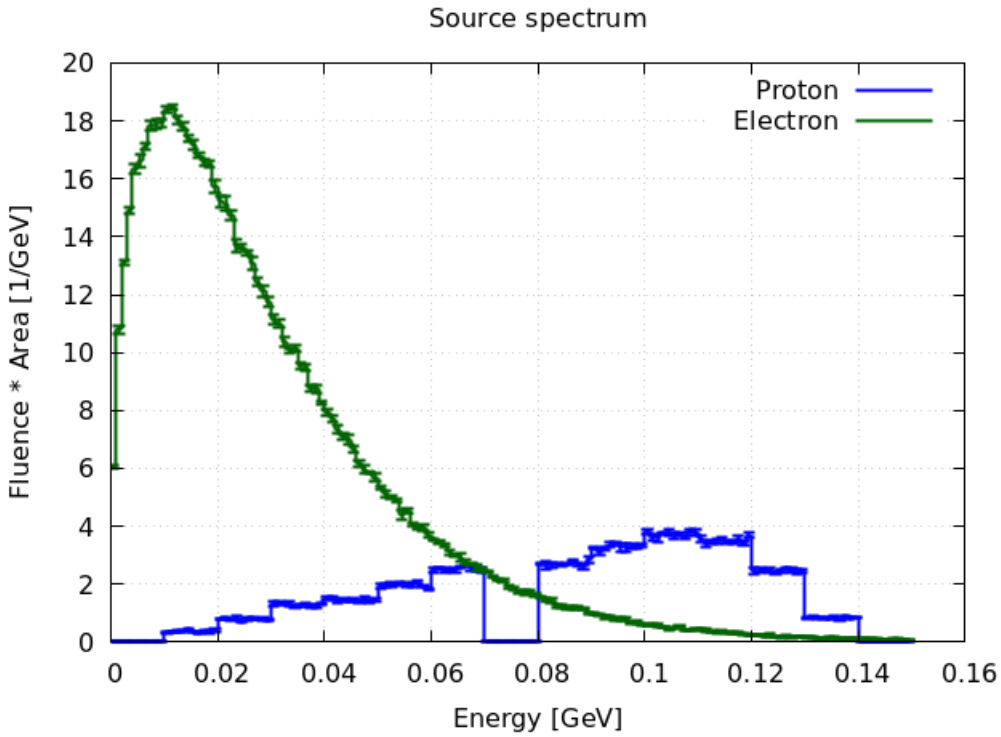
If we want to change a parameter of the source routine often, it is better to pass a value to it, instead of constantly recompiling

1. Use the first value on the **`SOURCE`** card as the electron temperature.
2. Set it to 20 MeV
   a. You can use a **`#define`** directive
   b. You may add a comment to the card to remember what it is used for
3. In the source routine use **`WHASOU(1)`** variable as the temperature for the sampling function
   a. You may pass the value to a custom variable first
4. On the [Run] tab click “Default Defines” to update the simulation
5. Compile / Run / Process the results

Exercise 6 – Sampling from a histogram

Sometimes we want to model a measured beam spectrum. To do this we can read from a histogram file.

1. Use the *`sample_histogram_momentum_energy()`* function to set the proton beam's energy
   a. Filename: "histogram.txt"
   b. Unit: "MeV"
2. Compile / Run / Process the results
3. Check the output file for the read histogram
4. Plot the results




FLUKA