INSS21 **Calculating Core-Collapse Neutrino Event Rates in Realistic Detectors**

K. Scholberg

1 Introduction

After a few warm-ups, we will use the SNOwGLoBES (`http://phy.duke.edu/~schol/snowglobes`) code to compute realistic event rates and observed spectra in a liquid argon detector. The first part makes use of neutrino flux models available as part of the SNOwGLoBES package. If time permits, you can try creating your own flux files and exploring time-dependent neutrino flux models.

2 Warm-ups

Use information in the lecture slides (or Google) to make back-of-the-envelope estimates for the following:

- **Warm-up #1:** Estimate the number of solar neutrino interactions occurring in a typical human body over a human lifetime. Consider neutrino-electron elastic scattering only. Assume the human body is made of water and that it has a mass of 80 kg. Assume the solar neutrino flux is $\phi \sim 2 \times 10^6$ cm^{-2}s^{-1}.

- **Warm-up #2:** Estimate the number of $\nu_e - {}^{40}$Ar charged-current neutrino interactions in a 40-kton liquid-argon detector for a core collapse at the center of the Milky Way. Estimate also the number for a core collapse in M31. (Do not cheat by reading event rates directly off any plot!)

- **Warm-up #3:** Assume that on average 7000 neutrinos will be recorded by Super-K (22.5 ktons of water) for a core-collapse supernova at 10 kpc. Plot the probability of recording at least one event as a function of distance, out to 100 Mpc. Indicate on your plot the far edge of the Galaxy, the LMC and Andromeda. Superimpose on the plot a similar sensitivity curve for Hyper-K (374 ktons of water).

3 Requirements for this Exercise

- You need to have basic working knowledge of the Linux/Unix shell and be able to navigate and execute commands on the command line in a terminal.
- You need to have Linux or MacOS with GLoBES [1, 2] installed (and therefore with all of GLoBES's dependencies), and Perl.
- You should have some kind of plotting tool. Root [3] will enable you to reproduce exactly the plots in the SNOwGLoBES manual, for which there are example scripts in the `plots` subdirectory, but other tools should work fine. There are some example Python notebooks as well.
- For the last part, you will need some familiarity with some kind of programming language or other tool that will enable you to write formatted function values into a text file. Fortran, C, C++, Perl, Python, Root, Mathematica, and many other things will work for this.

4 Installing SNOwGLoBES on Your Laptop

Follow the instructions in the SNOwGLoBES manual, section 2.2. You will need to set the `GLB_DIR` environment variable to where GLoBES is installed, and you will need to set the `SNOWGLOBES` environment variable to the SNOwGLoBES installation directory. Hints: check the INSTALL file for hints for installing GLoBES for your operating system. Also, it is strongly suggested to use `configure --prefix=GLB_DIR` to avoid installing GLoBES in a system area.

5 Calculating Event Rates in a Liquid Argon Detector

5.1 Rates for Fluxes in SNOwGLoBES

First, try out SNOwGLoBES using the included example fluxes and the `ar40kt` detector configuration (40 kton of liquid argon, using Icarus-like smearing).

- In your $SNOWGLOBES directory, do

```
./supernova.pl livermore argon ar40kt
```

- Plot the *interaction* rates versus neutrino energy, using your favorite plotting tool. These will be found in the out subdirectory, labeled by flux, channel and detector configuration, *e.g.*, livermore_nue_Ar40_ar40kt_events.dat.
- Plot also the *smeared* (detected) event rates versus observed energy. These will be found also in the out subdirectory, with the "_smeared" label.
- Compare your plots to the plots in the SNOwGLoBES manual.
- The total event rates can be dumped out using the make_event_table.pl script, *e.g.*, do

```
./make_event_table.pl livermore argon ar40kt
```

- Compare your answers to your back-of-the-envelope estimate.
- How far away could a core-collapse burst be observed (at, say, 90% C.L., assuming no background) in a liquid argon detector for the "Livermore" model? How about for the "GVKM" model?

5.2 "Pinched" Fluxes

Next, try computing event rates for flux files corresponding to the "Garching" parameterization [4]:

$$\phi(E_\nu) = \mathcal{N} \left(\frac{E_\nu}{\langle E_\nu \rangle} \right)^\alpha \exp\left[-\left(\alpha + 1\right) \frac{E_\nu}{\langle E_\nu \rangle} \right] , \quad (1)$$

where E_ν is the neutrino energy, $\langle E_\nu \rangle$ is the mean neutrino energy, α is the "pinching parameter", and $\mathcal{N}$ is a normalization factor.

These can be created using the pinched tool in the fluxes subdirectory.

- Try a few different parameter values for the different flavors: $\langle E_\nu \rangle$ ranging from 5 to 20 MeV, α from 0 to 7. Normalize for luminosity of 1.5×10^{51} ergs/s integrated over 10 seconds. Plot the fluxes.

- Put the flux files in the `fluxes` subdirectory and run `supernova.pl` on them. Plot the resulting observed spectra.
- For the indicated range of parameters for ν_e fluxes, plot number of events in liquid argon as a function of α and $\langle E_\nu \rangle$.

5.3 More Explorations

If time permits, you can look at:

- Try calculating events for a stopped-pion flux (`stpi.dat` is included). How large a detector at the Spallation Neutron Source would be needed to meausre the cross section in argon to 10% in a year?
- Time-dependent fluxes, which can be described by evolution of pinching parameters as a function of time.
- Signals in different detector types.

You can explore also how well it would be possible to determine supernova parameters given a signal expectation for some particular assumptions.

References

[1] `http://www.mpi-hd.mpg.de/lin/globes/`.

[2] Patrick Huber, M. Lindner, and W. Winter. Simulation of long-baseline neutrino oscillation experiments with GLoBES. *Comput. Phys. Commun.*, 167:195, 2005.

[3] `http://root.cern.ch/drupal/`.

[4] Irene Tamborra, Bernhard Muller, Lorenz Hudepohl, Hans-Thomas Janka, and Georg Raffelt. High-resolution supernova neutrino spectra represented by a simple fit. *Phys.Rev.*, D86:125031, 2012.