Update on the TAMBO Monte Carlo Simulation

Jeffrey Lazar

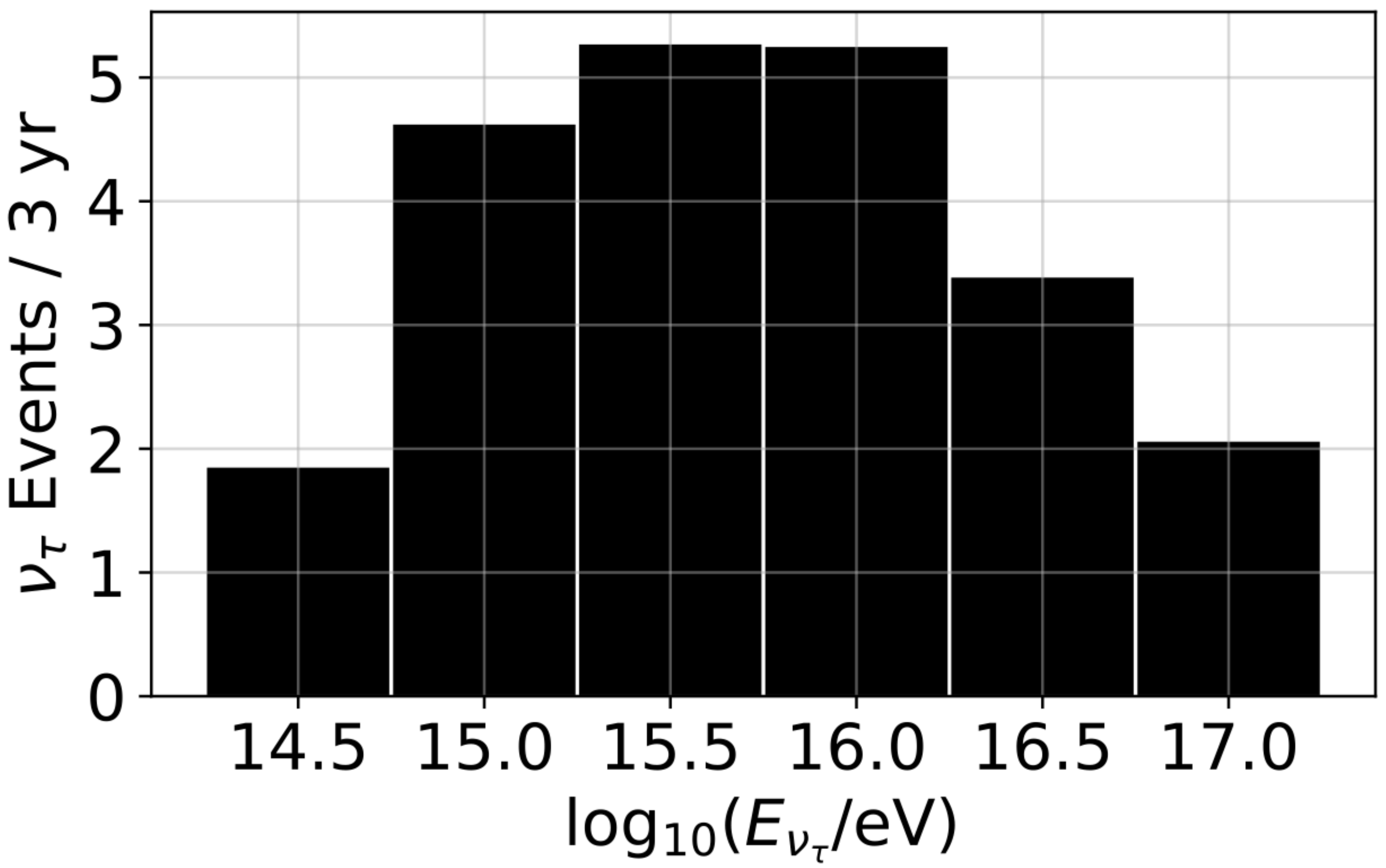
TAMBO Touch Base

Universidad Nacional de San Agustín de Arequipa

18 Oct. 2022

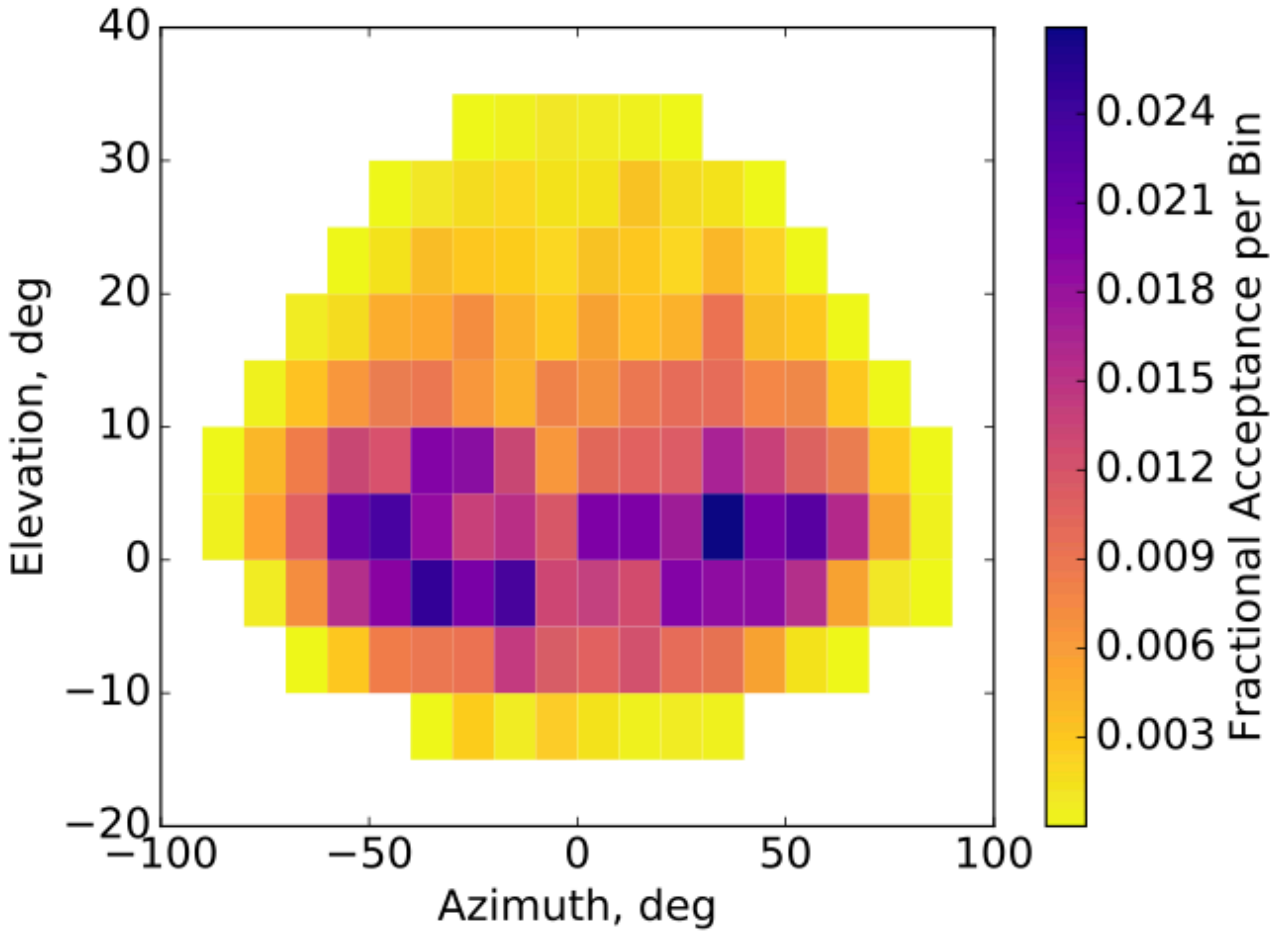
What we're after

True rates



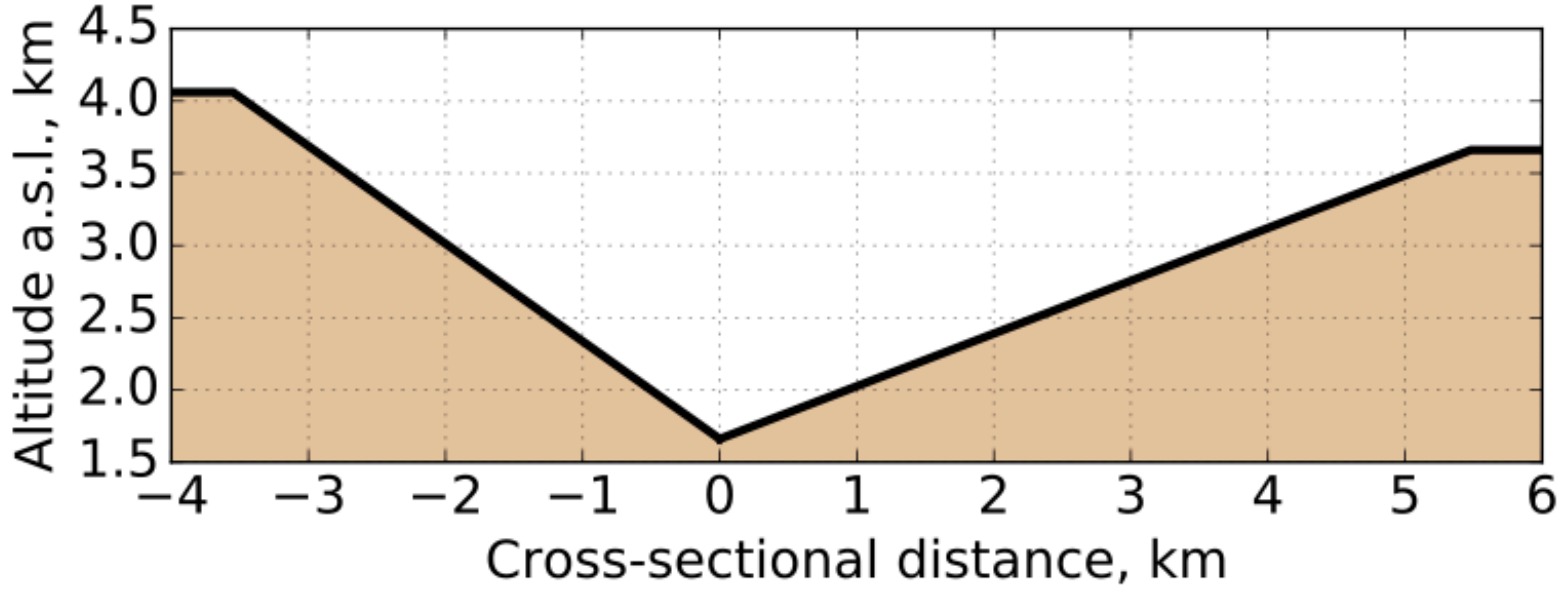
Total number of events in 3 years: Total number of events expected by extrapolating the high-energy flux measured by IceCube. Reproduced from DOI:10.1086/423124

Detector response

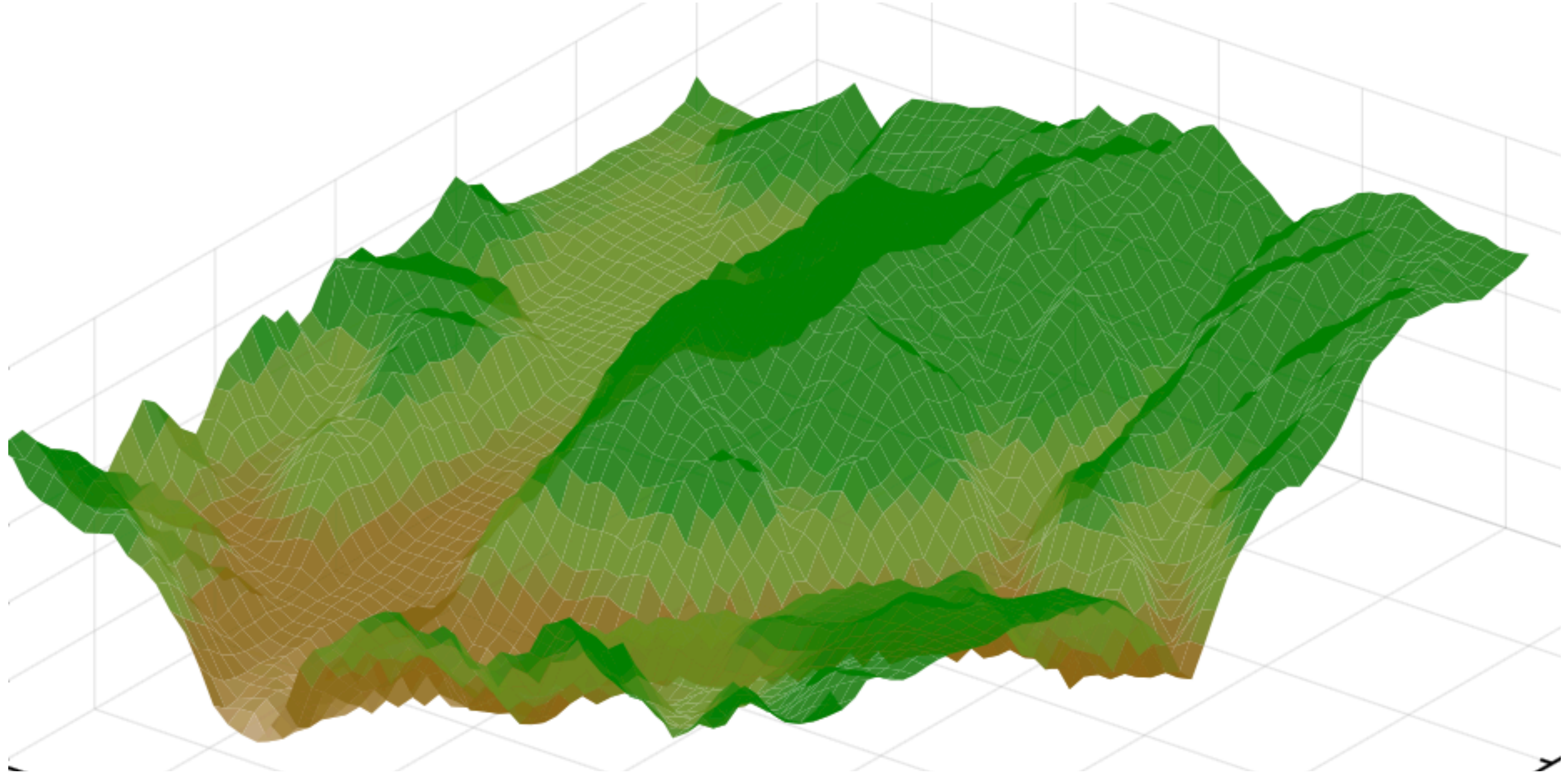


Fractional acceptance of events as a function of arrival direction: Reproduced from DOI:10.1086/423124

Moving past the first pass

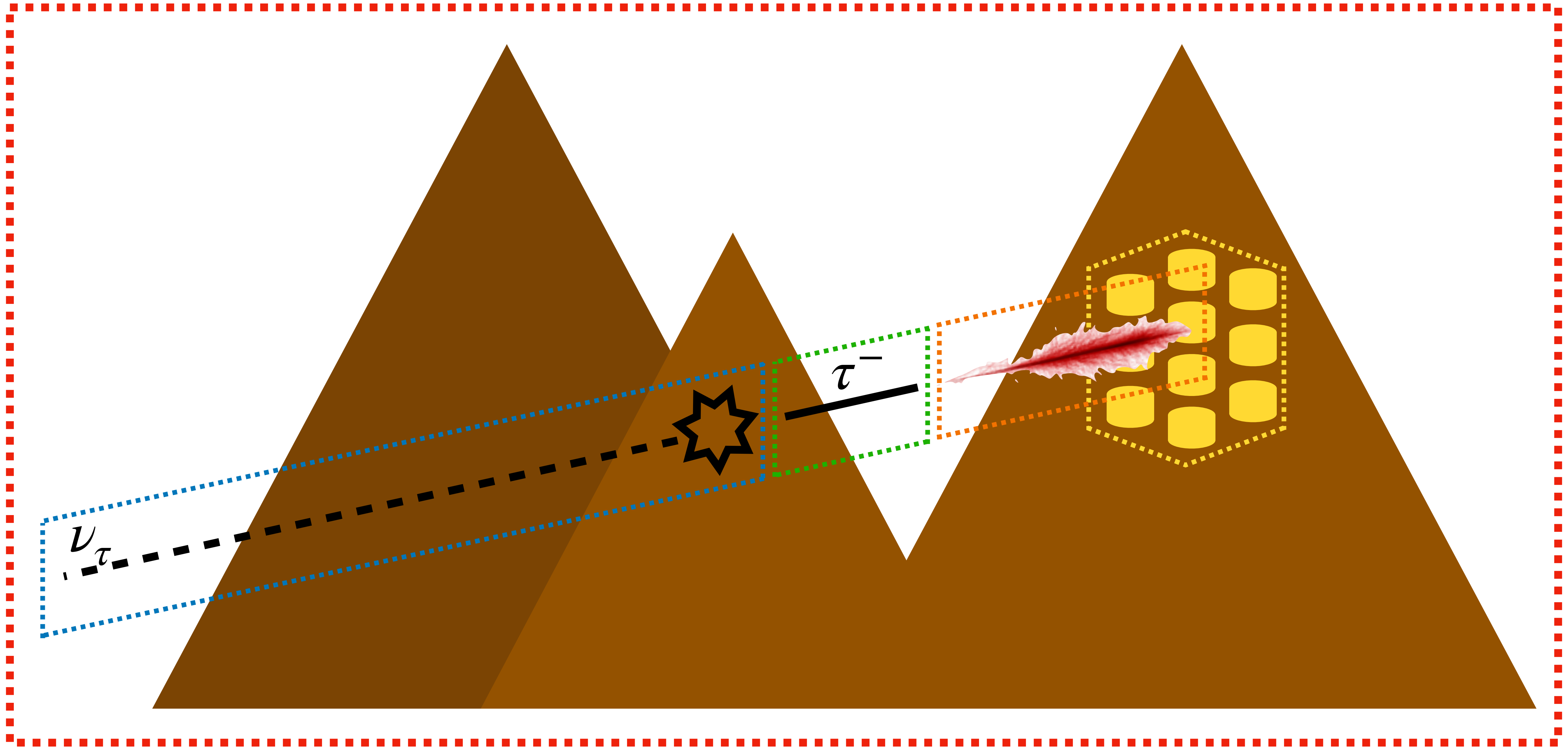


- Simplified geometry
- No treatment of $\tau^{\pm}$ energy losses
- Approximation of air-shower physics



- Realistic
- Full treatment of $\tau^{\pm}$ energy losses
- (Less) approximation of air-shower physics. Eventually few simulation

Monte Carlo Simulation: an overview



Initial neutrino injection: Select initial neutrino properties, *i.e.* energy, direction, interaction vertex, *etc.*

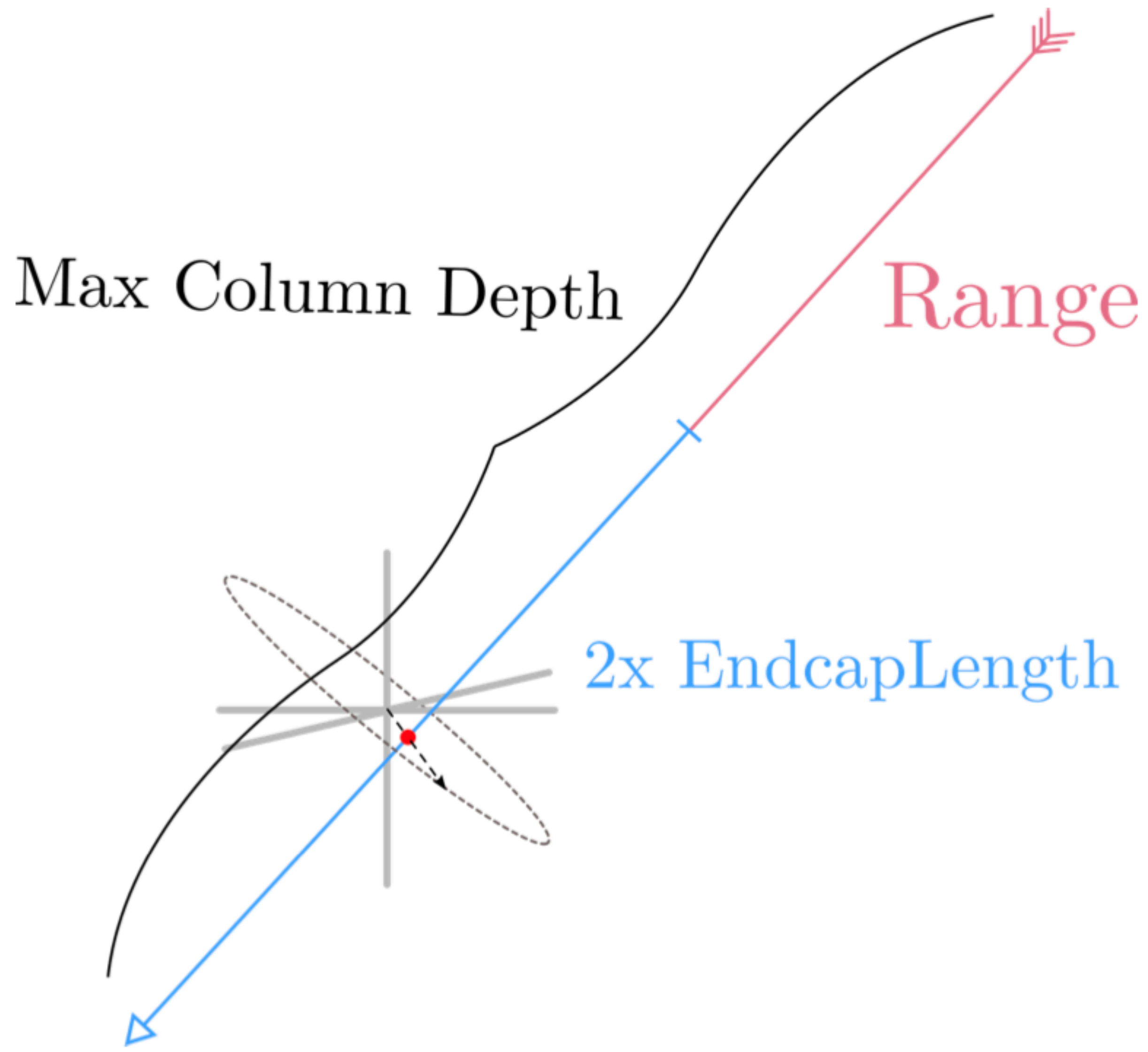
Charged lepton propagation: Propagate outgoing charged lepton, accounting for energy losses and decay, to find decay point

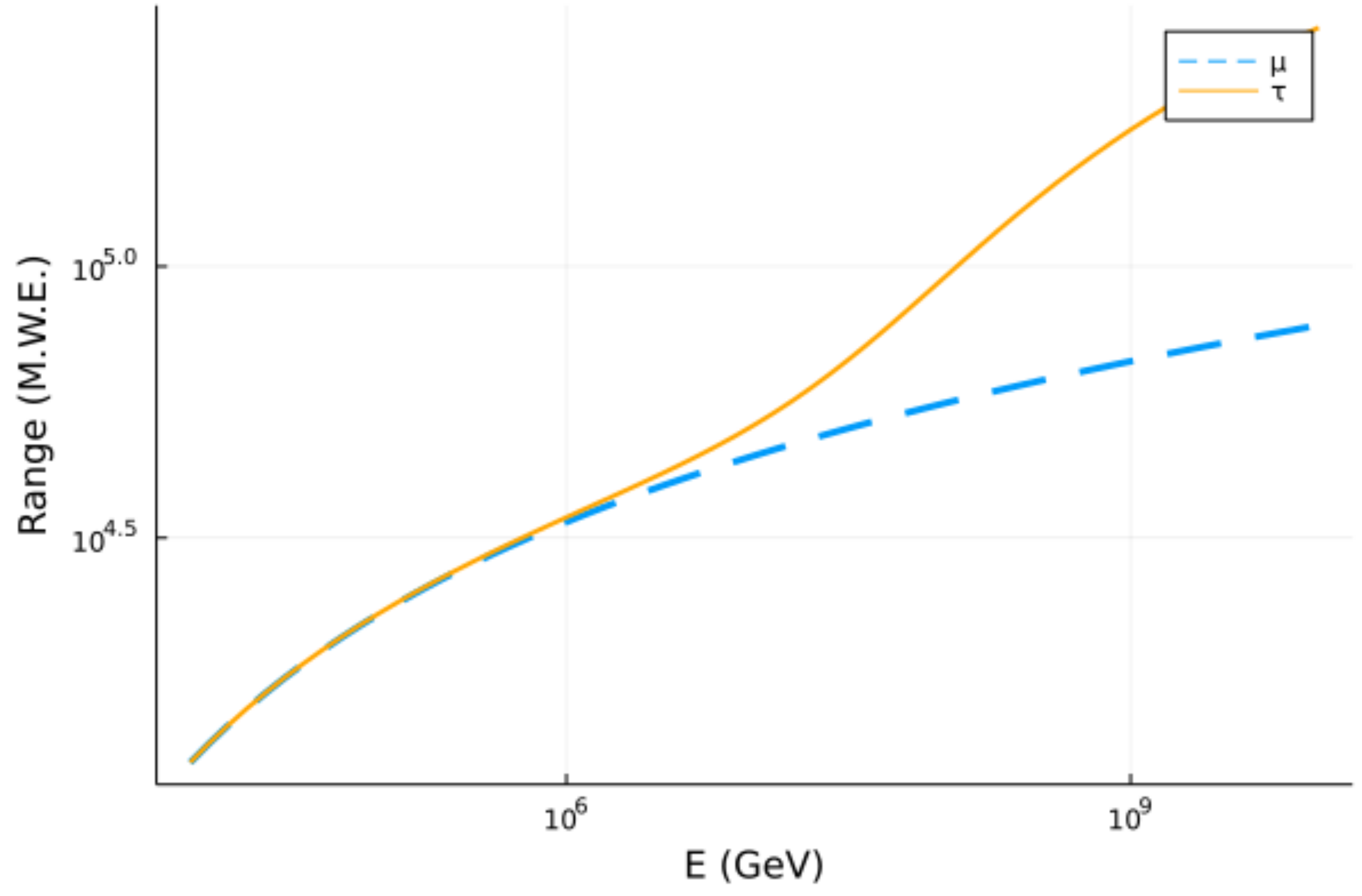
Air-shower simulation: Model shower development from lepton decay

Detector response: Simulate internal hardware to model what we will see

Event weighting: Remove unphysical remnants from selection of initial neutrino properties

Initial Neutrino Injection





Charged Lepton Propagation

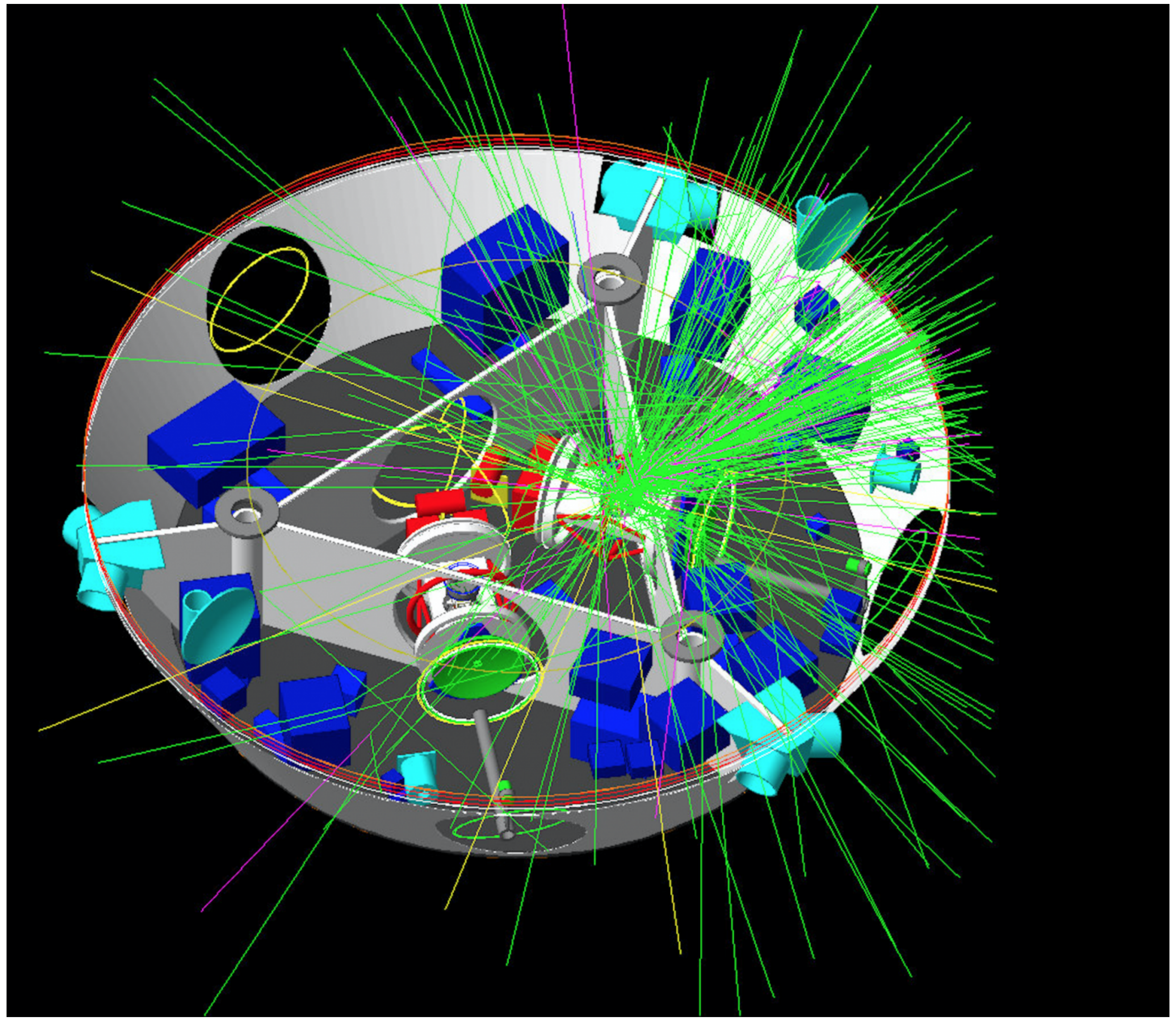
- PROPOSAL is a Monte Carlo framework for propagating charged leptons and gamma rays
- Most up-to-date cross sections and for interaction and decay
- C++ and Python interfaces. Sometimes can be installed with `pip`

Air-Shower Simulation

- CORSIKA is a MC-based software for calculating air-shower
- FORTRAN-based code, c++ hopefully coming soon
- Currently working on an approximation that treats the mountain as a plane
- We will need to simulate a large CORSIKA library for a fully accurate simulation

Detector Response

- Geant4 is a toolkit for simulating the passage of particles through matter
- Detailed treatment of internal detector geometries
- Reference for LHC experiments
- C++ interface



Event Weighting

$$\frac{1}{w_{\text{gen}}} = N_{\text{gen}} \frac{1}{\Omega_{\text{gen}} A_{\text{gen}}} \times \frac{\rho_{\text{gen}}(\ell)}{X_{\text{gen}}} \times \frac{1}{\sigma_{\text{tot}}(E)} \frac{\partial \sigma}{\partial y}(E, y) \times \frac{\Phi_{\text{gen}}(E)}{\int \Phi_{\text{gen}}\, \mathrm{d}E}$$

$$w_{\text{phys}} = \sum_{\{\text{gen}\}} \underbrace{\left(\frac{X_{\text{phys}} N_A}{M_{\text{target}}} \times \frac{\rho_{\text{phys}}(\ell)}{X^{\text{col}}_{\text{phys}}} \times \frac{\partial \sigma}{\partial y} \times \boxed{\Phi_{\text{phys}}(E, \theta)} \right)}_{\text{physical distribution}} \times w_{\text{gen}}$$

- We generate according to a distribution which may not be physical
- For example, we may sample event uniformly on a sphere but attenuation makes the unlikely
- Remove "unphysicalness" reweighting by removing generation dependence
- Also allows simulation to be applied to different fluxes, cross section models, *etc*.

TAURUNNER

Python-based package for simulating ν_τ propagation: Uses MC approach to propagate tau neutrinos including previously neglected

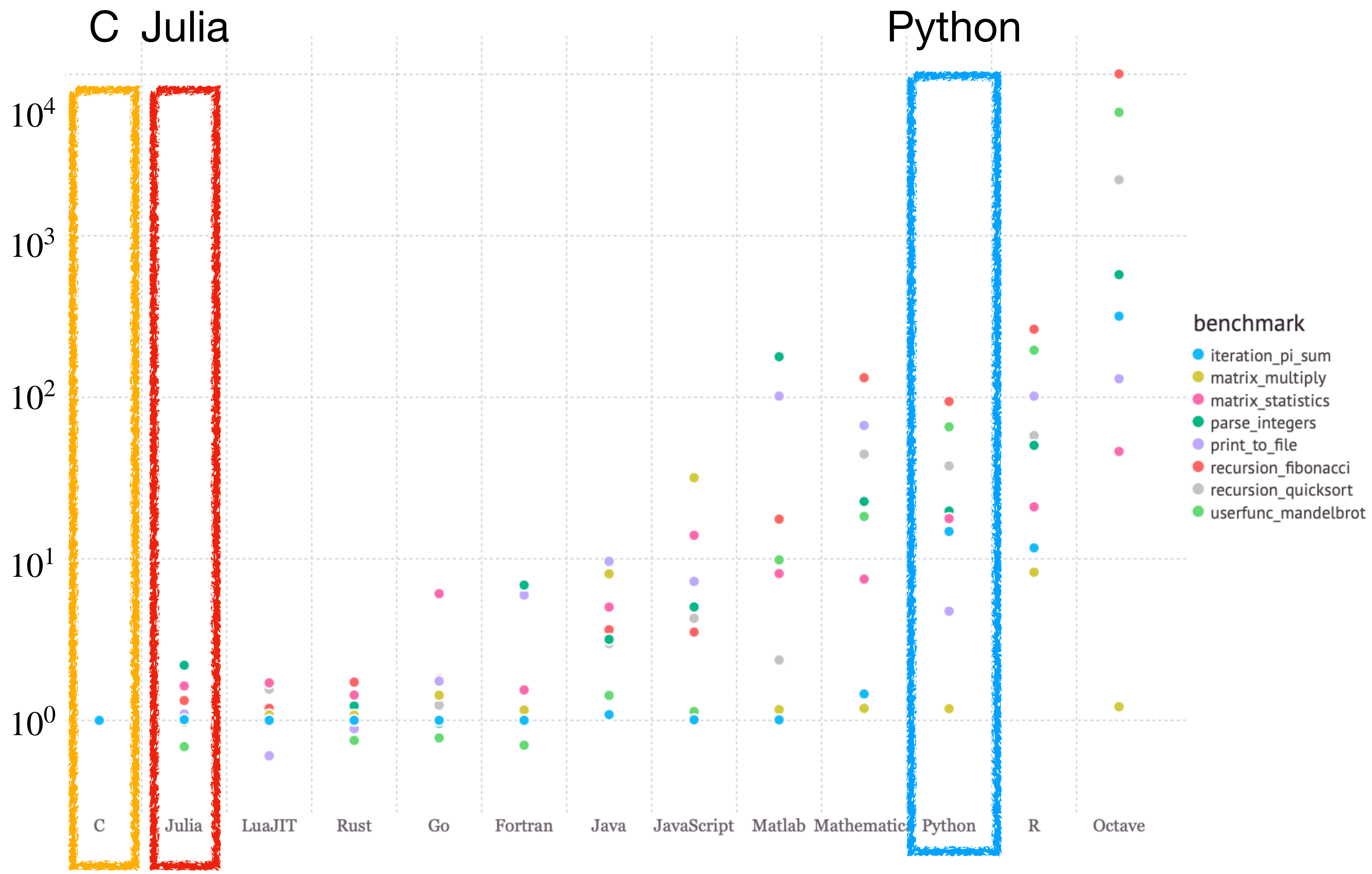
Python, C++, FORTRAN

Which one should we use ?

- None of them :-)
- Currently being designed in the Julia language
- Initial injection is purely in Julia, and most other things rely on Julia-based interface to other codes

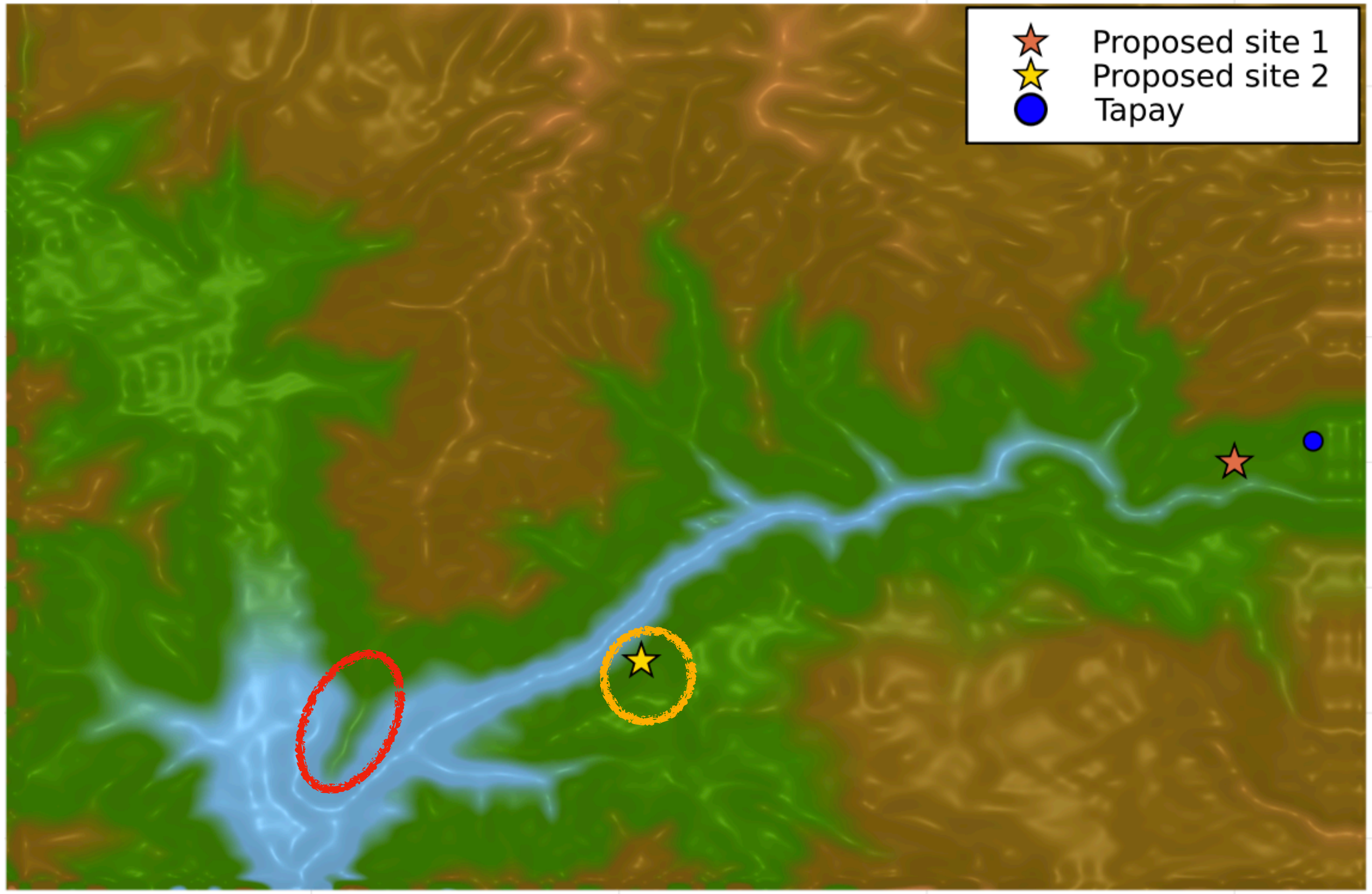
Julia Language

- Relatively new language optimized for data analysis
- JIT compilation allows for a nice combination of interactive coding and speed
- Can be used as scripts, in a REPL, or in Jupyter notebooks

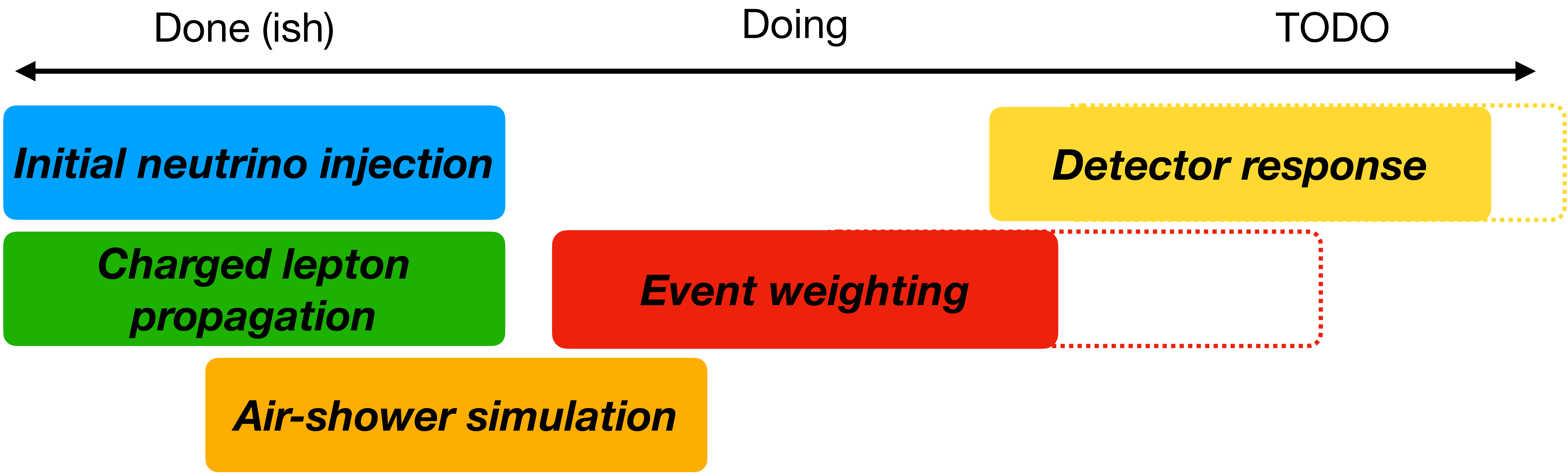


Timing comparison for various languages: Timing comparisons on a variety of benchmarks normalized to their time in C. Note, compile time is not included for Julia

Where do we stand ?

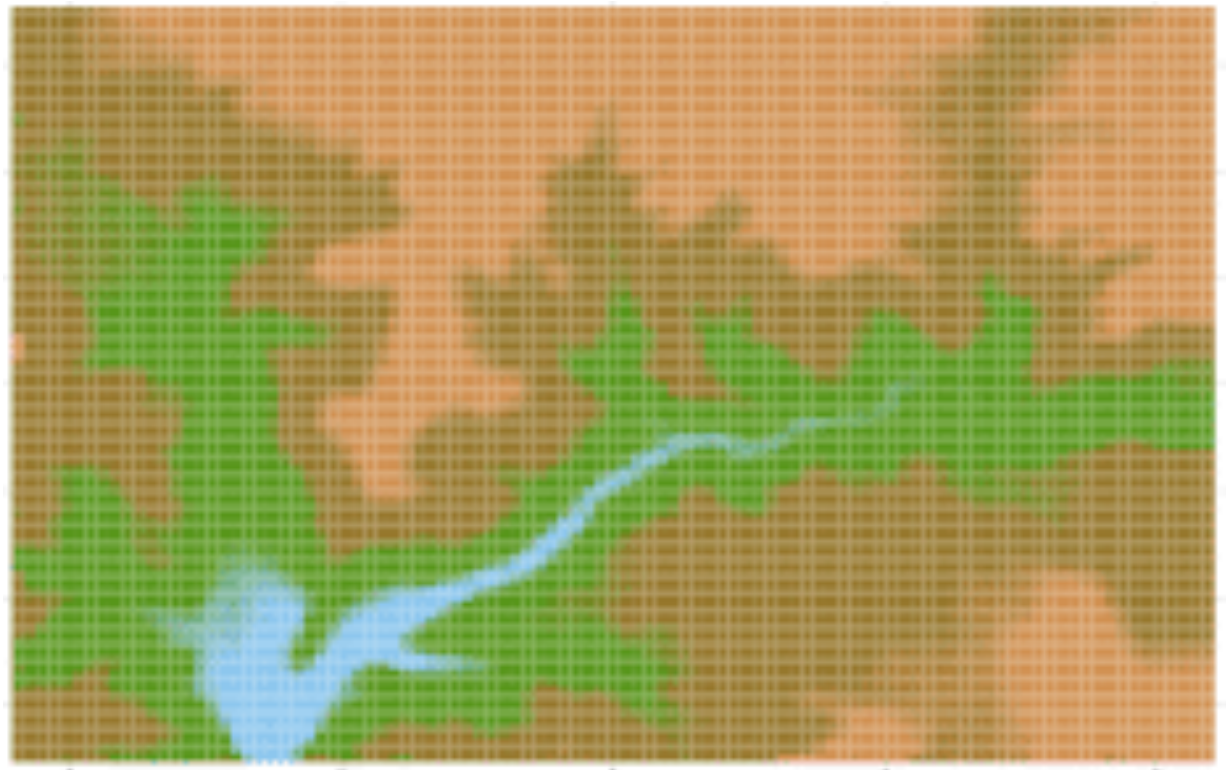


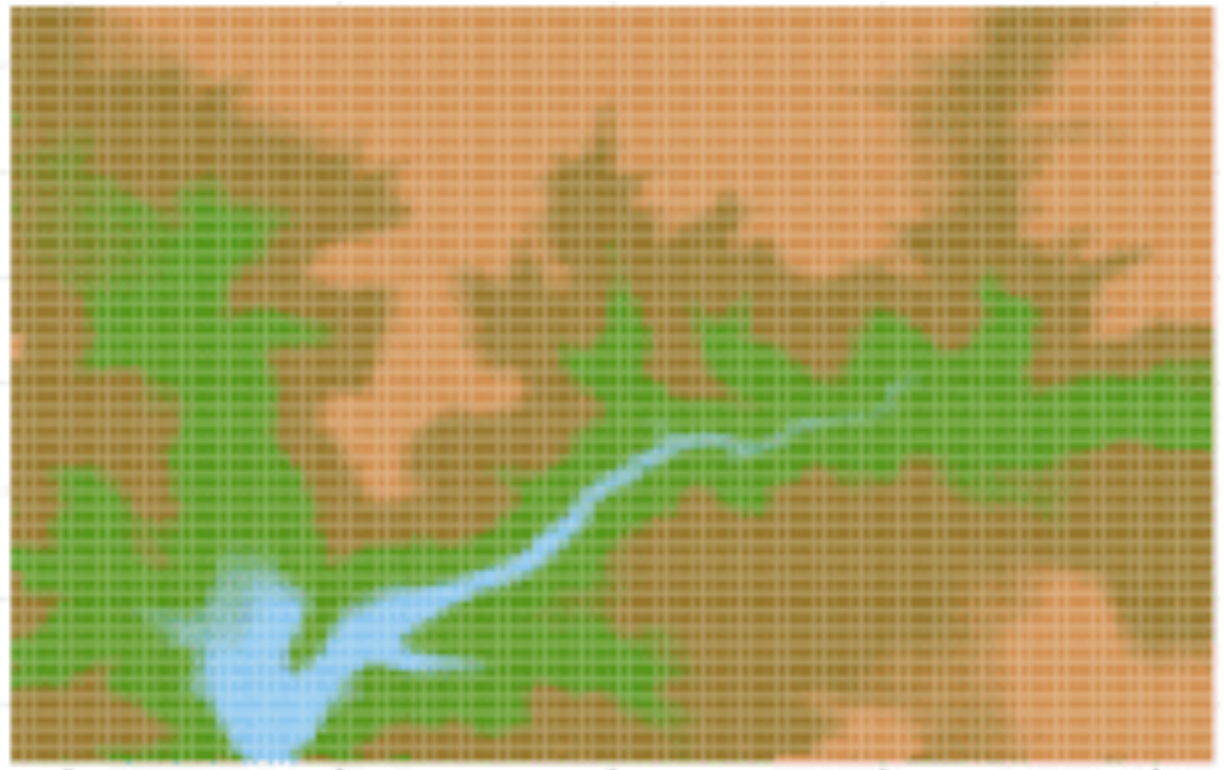
Where do we stand ?

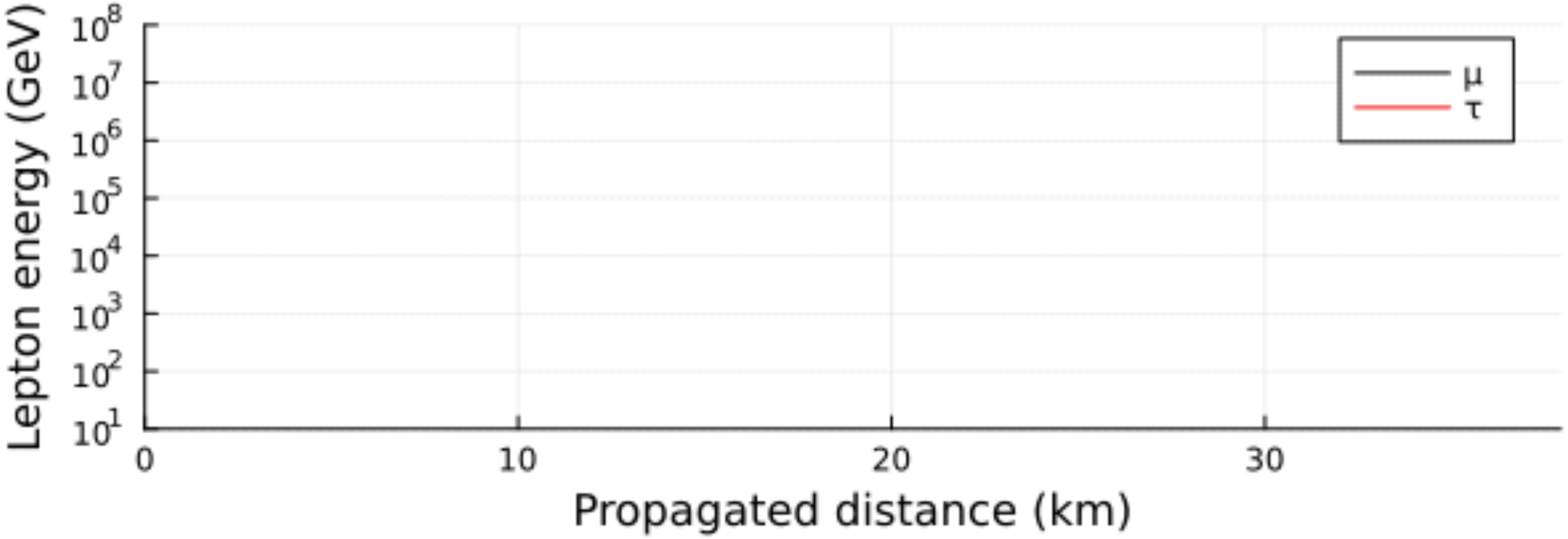


- If we can finish the air-shower simulation, we can do a first round of simulation
- This will allow us to assess detector needs, optimize configurations, and determine feasibility of sites

A GIF for the End :-)







We would love your help !

- We are close to pushing past a major milestone and producing realistic simulation at a site of interest
- Beyond this, we will need to hone and perfect the particle physics and detector simulations
- Plenty to do. Code needs to be tested, bugs needs to be found, and large simulation needs to be run
- All skill levels welcome and no familiarity with Julia necessary :-)