Fast Likelihood Analysis in More Dimensions for Xenon TPCs

The Flamedisx package

Bart Pelssers

in collaboration with

Jelle Aalbers and Cristian Antochi

Stockholm University

PyHEP, 18 October 2019

Dark Matter

- Makes up 26.4% (Planck 2018) of the total energy–density in the universe.
- Its presence is inferred from cosmological and astronomical observations at both small and large scales.
  - Rotation curves of galaxies.
  - Gravitational lensing of clusters.
  - Cosmic Microwave Background measurements.
- Direct detection experiments aim to detect the low–energy, $\mathcal{O}(\text{keV})$, recoil of dark matter on ordinary matter.

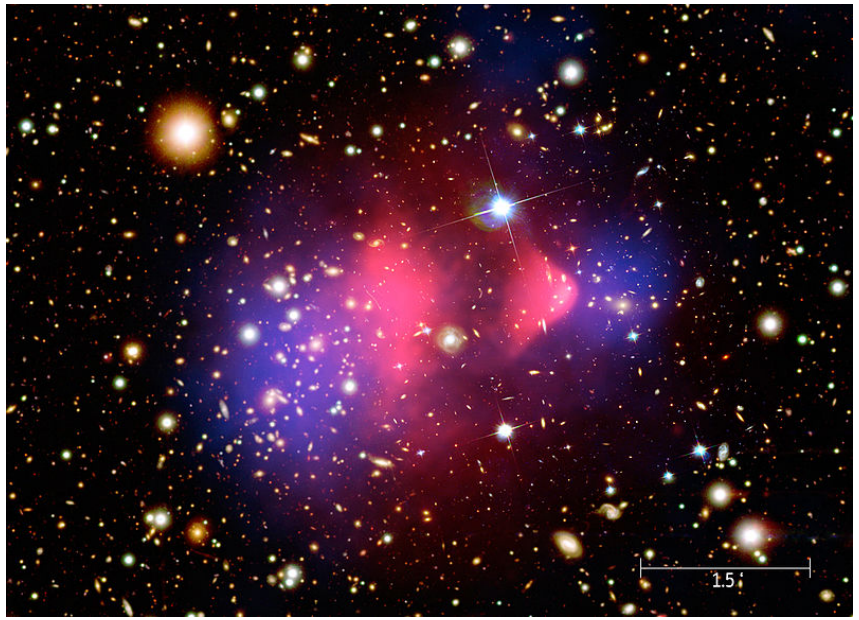
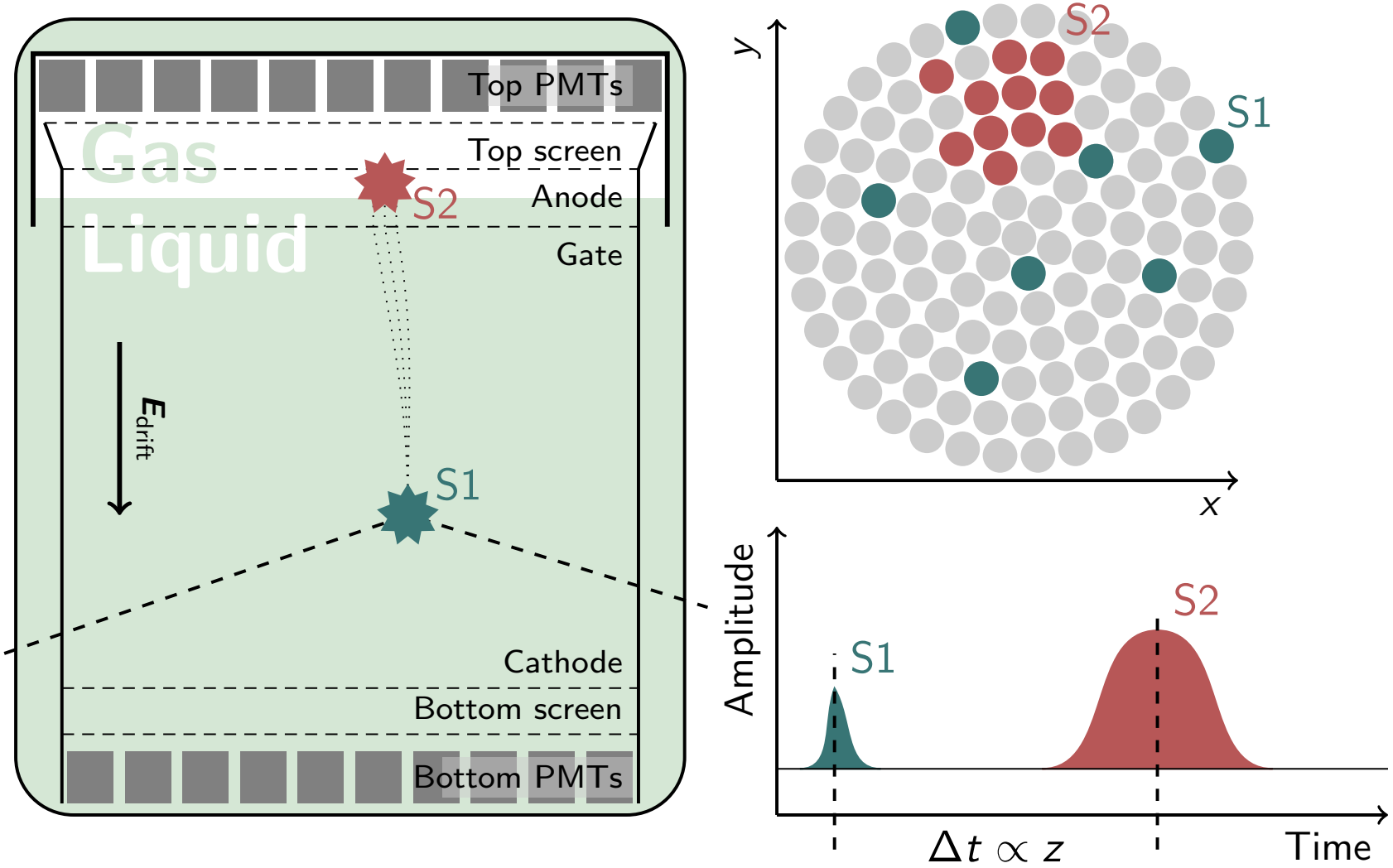


Figure: Composite image (optical, x-ray, computed dark-matter) of mass distribution in the bullet-cluster of galaxies. Chandra X-ray Observatory Center

The XENON Dark Matter Experiment

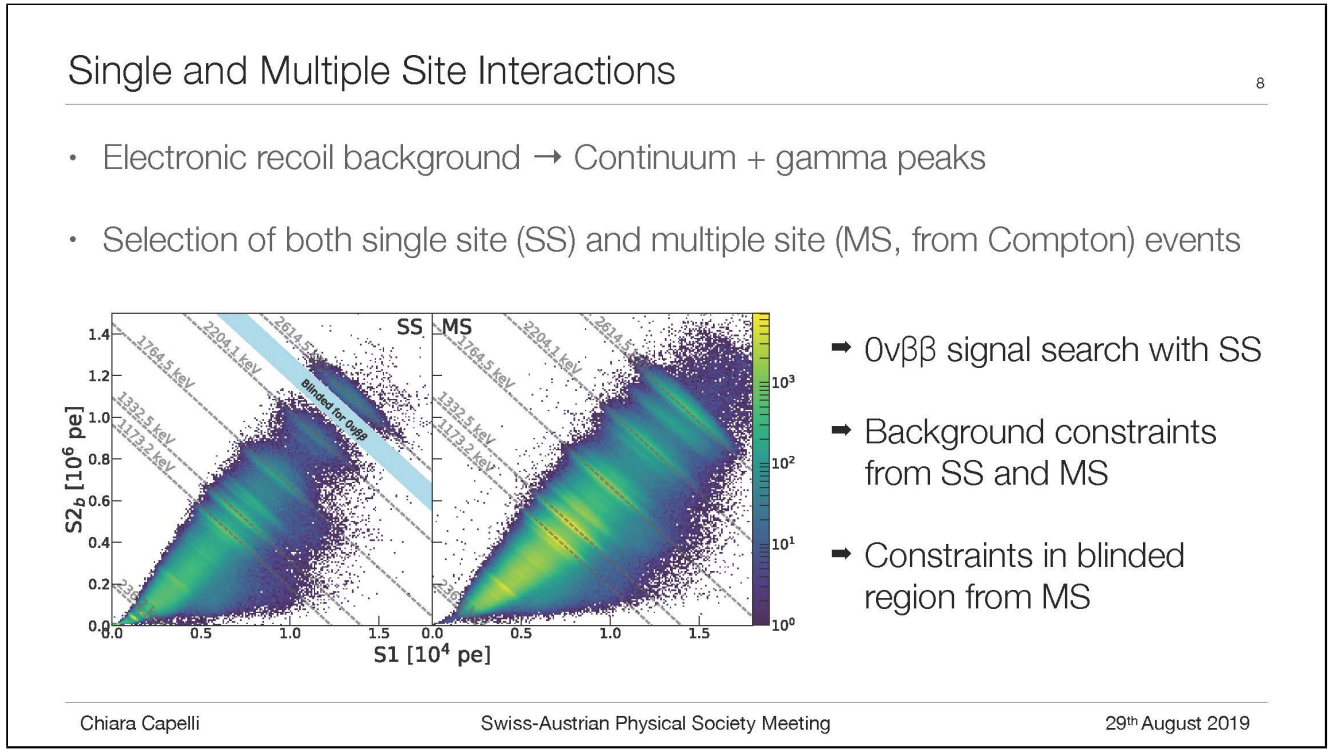
The Dual–phase Time Projection Chamber



The Liquid Xenon Emission Model

The LXe emission model is complex:

- S1 and S2 are anti–correlated
  $\rightarrow$
  pdf convolution
- ER recombination fluctuation

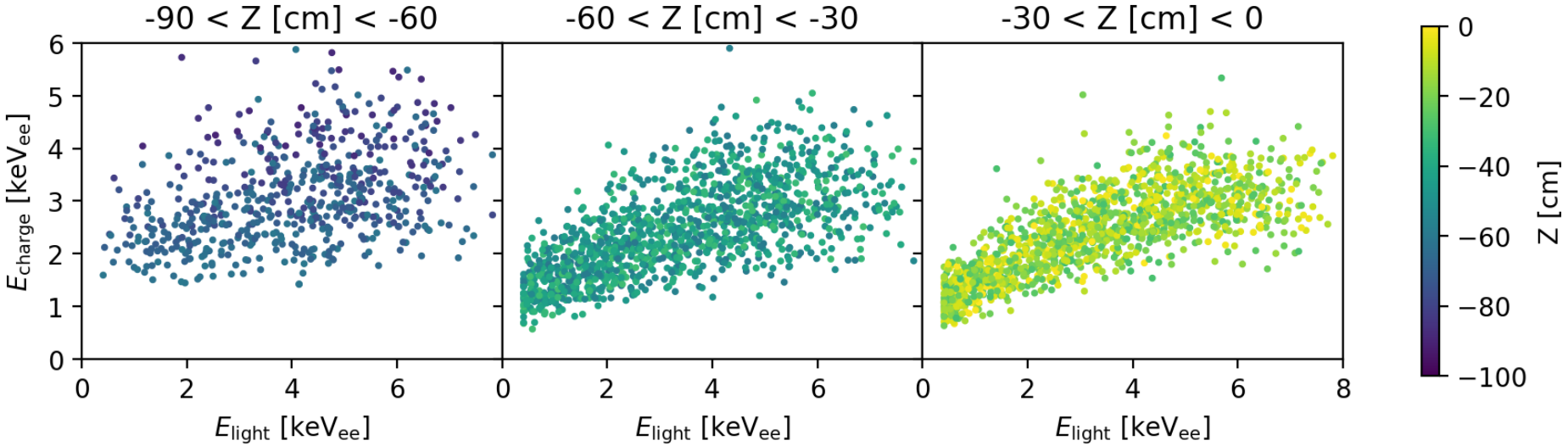


The Liquid Xenon Emission Model

The LXe model has several observable dimensions; S1, S2, x, y, z, t

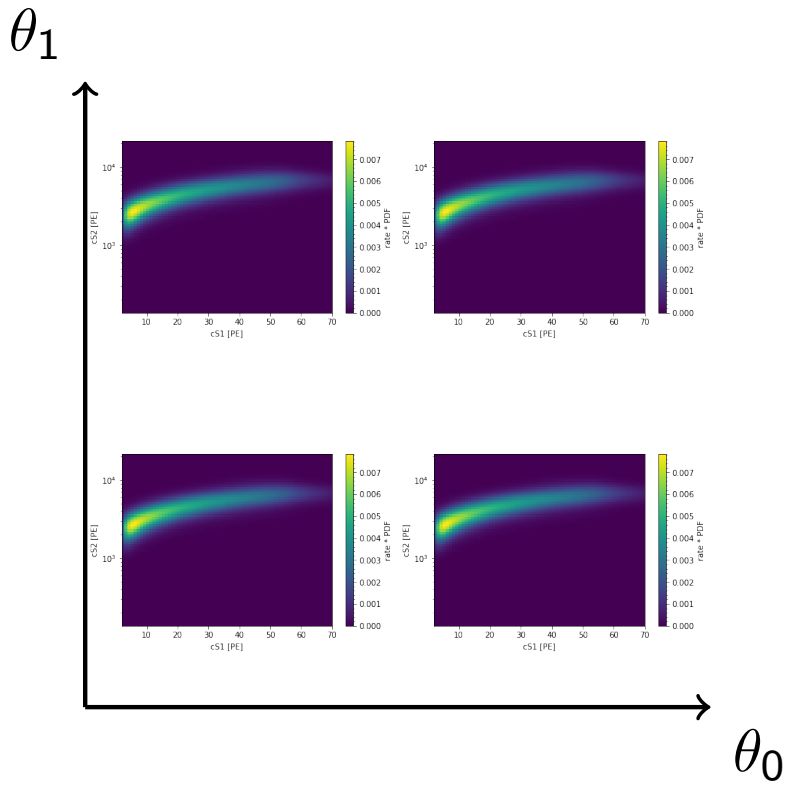
Example

Event discrimination is function of depth, including z as dimension in the likelihood will improve event discrimination and sensitivity.

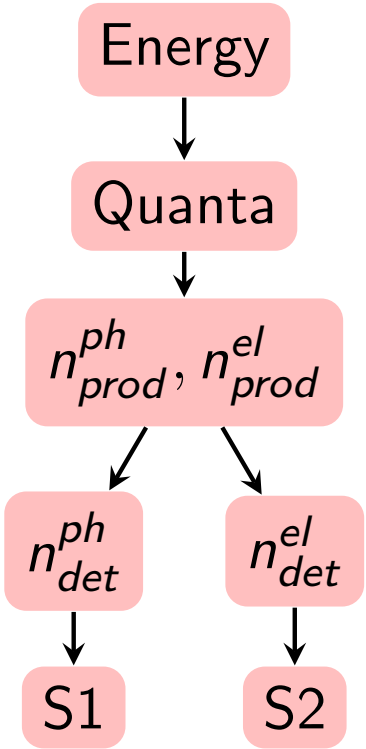


Inference using Monte–Carlo Simulations

- Traditionally cS1, cS2 templates are computed by running many MC simulations. (cS1, cS2 being detector independent w.r.t. S1, S2)
- Templates are made for different parameter values and interpolated.
- These methods scale poorly
- Computing templates takes: $\mathcal{O}(\mathrm{e}^{\mathrm{n_{obs}}+\mathrm{n_{params}}})$



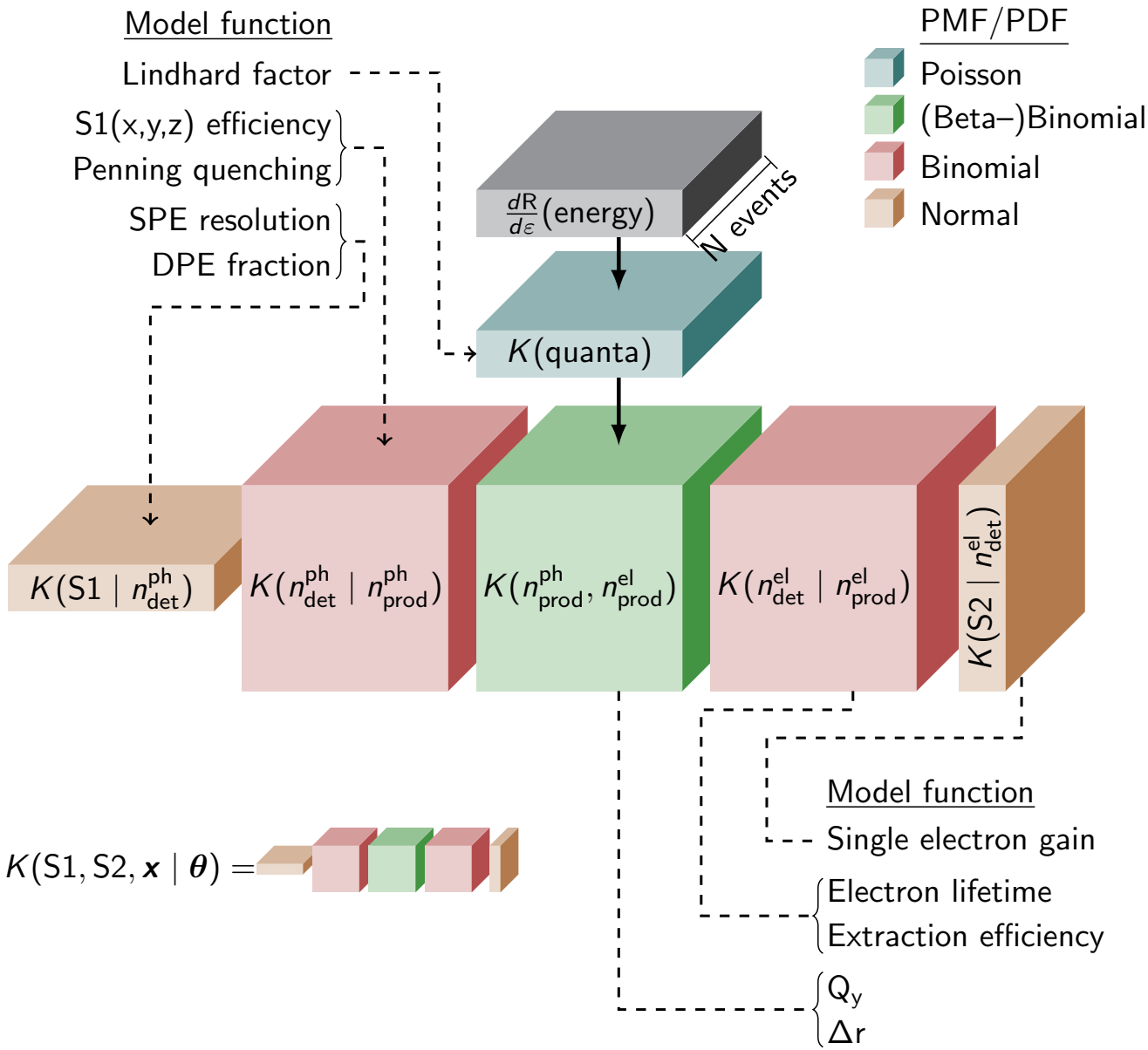
Inference by Explicit Summation



Write the LXe emission model as one analytic expression returning the differential rate $K(\mathrm{S1}, \mathrm{S2}, \boldsymbol{x} \mid \boldsymbol{\theta})$ for an event with energy S1 and S2 at position and time $\boldsymbol{x} = (x, y, z, t)$, depending on parameters $\boldsymbol{\theta}$.

- Implemented using several matrix multiplications
- Batched evaluation
- Analytic $\rightarrow$ differentiable

Implementation of LXe Emission Model



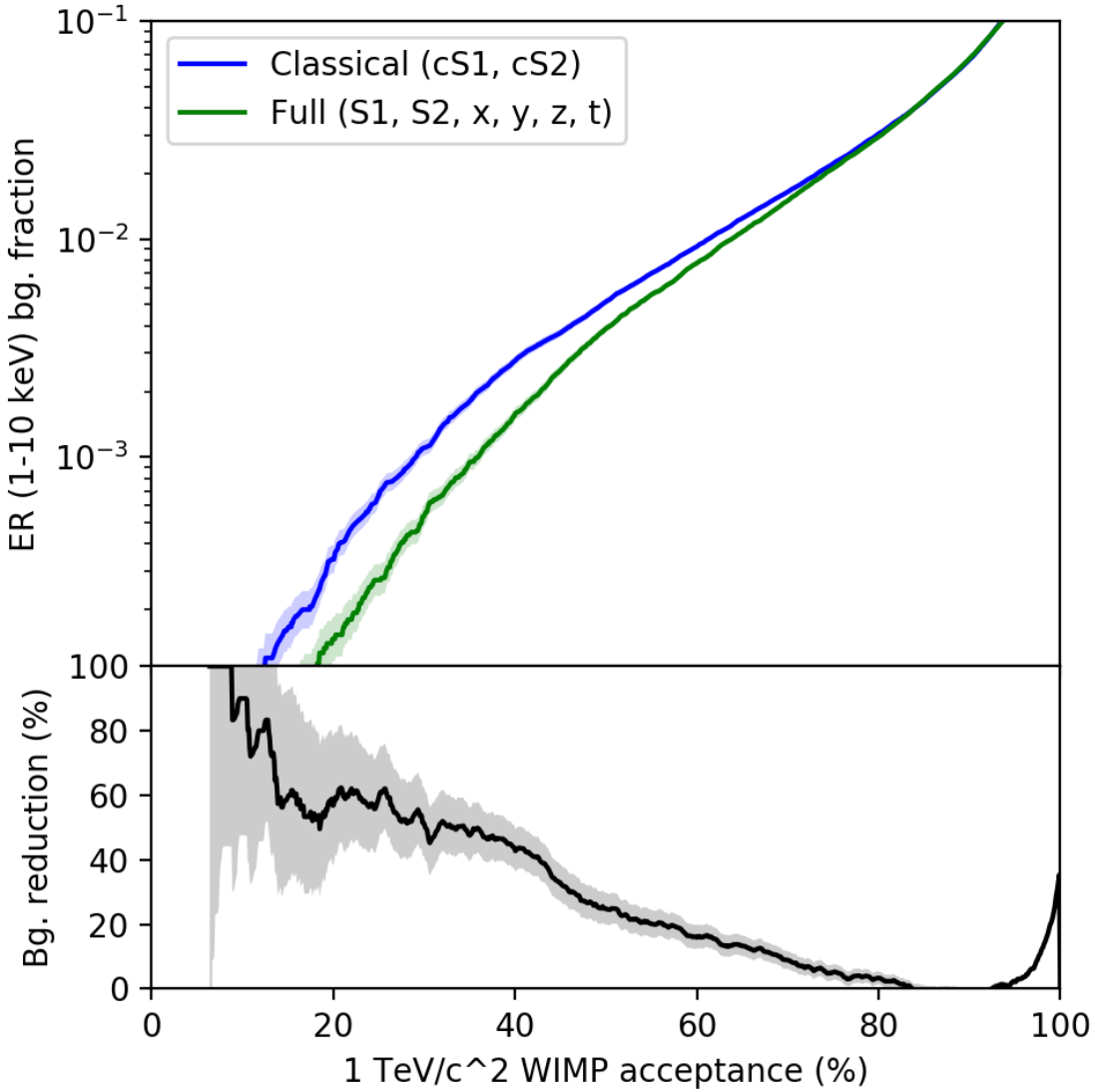
Likelihood

Flamedisx uses an extended (unbinned) likelihood. It can contain:

- Multiple sources derived from base implementation of electronic recoil (ER), nuclear recoil (NR) or interactions of Weakly Interacting Massive Particles (WIMPs).
- Different datasets for different sources (Calibration and Science data)
- Template sources (i.e. to model accidental coincidence events)
- Constraints

$$\ln \mathcal{L} = -\mu(\boldsymbol{\theta}) + \sum_{\text{events}} \ln \sum_{\text{sources}} K(\text{S1}, \text{S2}, \boldsymbol{x} \mid \boldsymbol{\theta}) \tag{1}$$

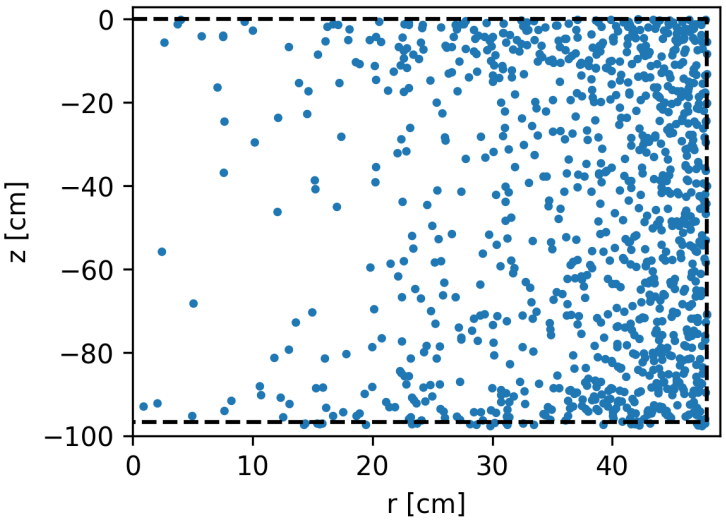
Event Discrimination



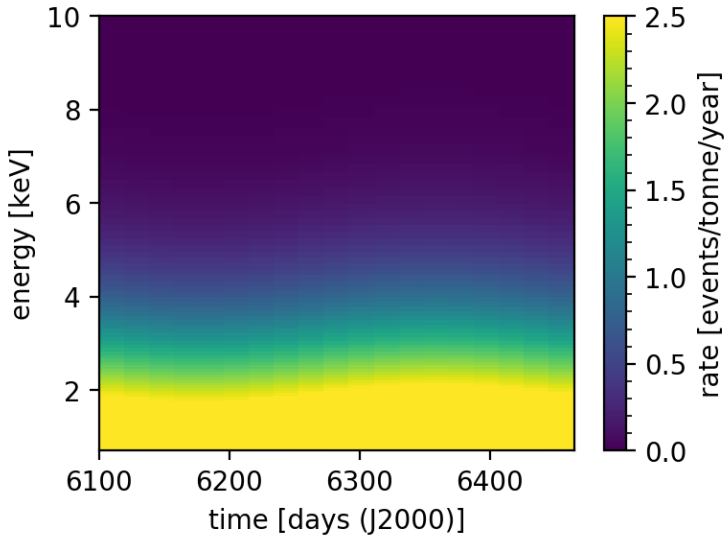
- Comparing template based cS1, cS2 modelling to full 6D likelihood.
- Single event discrimination for a XENON1T SR0 setup
- Full model has variable electron lifetime between 200 μs and 500 μs

Extending Models

- Add spatial rate multiplier
- For modelling of external backgrounds (from surface or materials)



- Time dependent energy spectrum
- Include WIMP annual modulation signature



Running Flamedisx

Try out the Flamedisx Tutorial notebook directly on Google Colab here

- Flamedisx is written in Python 3
- Few dependencies, most importantly:
  - Tensorflow 2
  - Tensorflow probability
  - wimprates
- Code runs on CPU and GPU without additional configuration.
- Factor 10 to 20 speedup going from consumer laptop to Tesla K80 GPU

Summary

- Extra dimensions $\rightarrow$ better sensitivity
- More parameters $\rightarrow$ better modeling
- Speed $\rightarrow$ multiple fits, toyMC studies, non–asymptotic inference.

Flamedisx can be found on GitHub and installed via PyPI:
github.com/FlamTeam/flamedisx

```
pip install flamedisx
```