Optimisation of Fast Likelihood Functions for Dark Matter and Rare Event Searches

Joshua Robert Green

Flamedisx can evaluate the **likelihood** of data with **more observables** and **shape varying parameters** faster than **Monte Carlo (MC)** can [1].
The author solved **performance problems** unique to **Flamedisx** from the addition of the **Noble Element Simulation Technique (NEST)** models, but **rate estimation** requires work [2].

1: Searching for Dark Matter [3][4]

Dual phase time projection chambers are used to search for anomalous recoils.
Mostly search for nuclear recoils (NRs):

- Less noise than electron recoils (ERs)
- Weakly Interacting Massive Particles

NRs are much more compact and produce more prompt light (S1) than electrons (S2), due to recombination.

As such usually models consider **S1/S2,** correcting for any effects in the other spaces, e.g. drift time, X-Y position, and event time. (S1c/S2c)

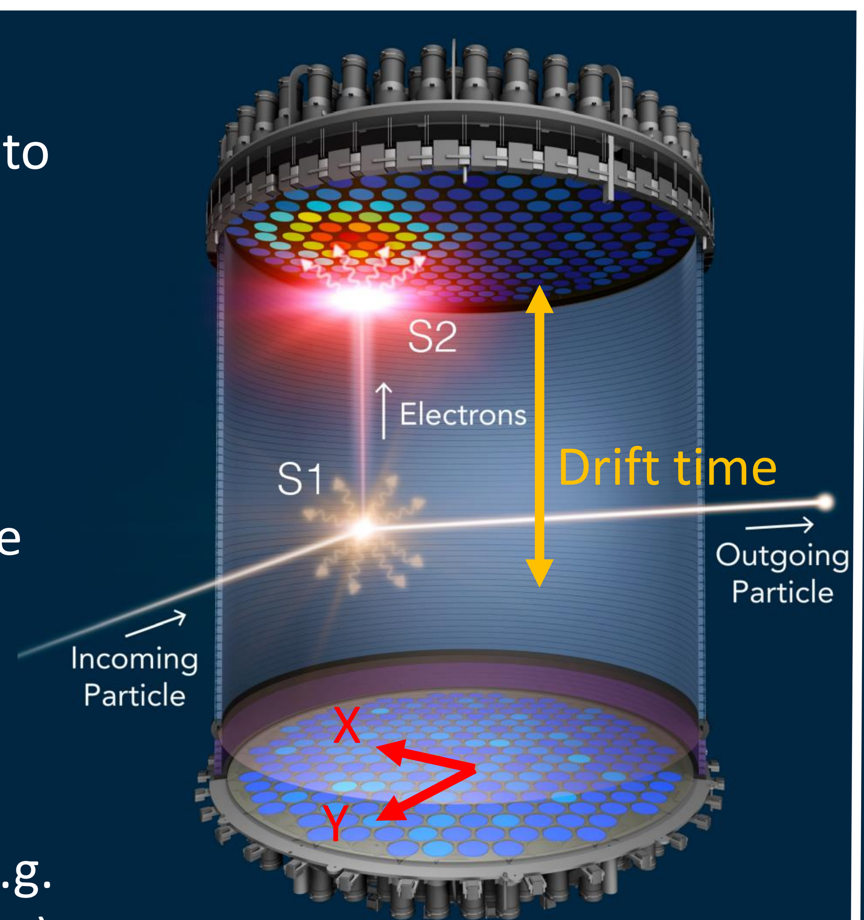


Diagram of the LUX-ZEPLIN dual phase time projection chamber [3].

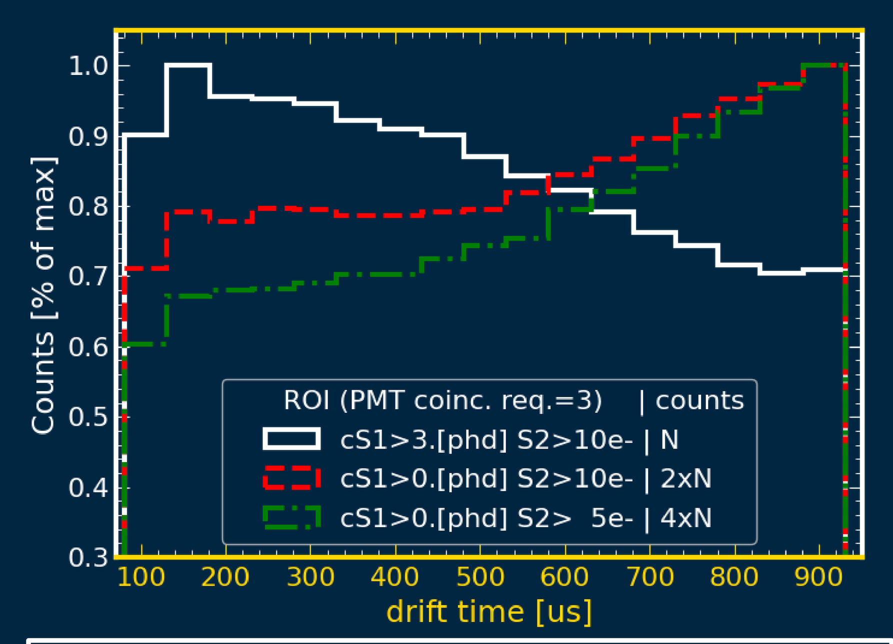
2: What else could a model want?

More Dimensions:

- The behaviour of threshold signals has spatial effects.
- Some key backgrounds are spatially distributed.

More parameters

- Uncertainty on NEST's NR models impacts near-threshold signals.
- Incorporating shape-varying systematic uncertainties negates the need for conservative modelling choices.
- Band-fitting procedures can have degeneracy difficulties.



Simulated Drift time distribution of coherent nuclear scatters from 8-boron solar neutrino in a dark matter detector using different thresholds for a detector with LUX-ZEPLIN first science run conditions [4].

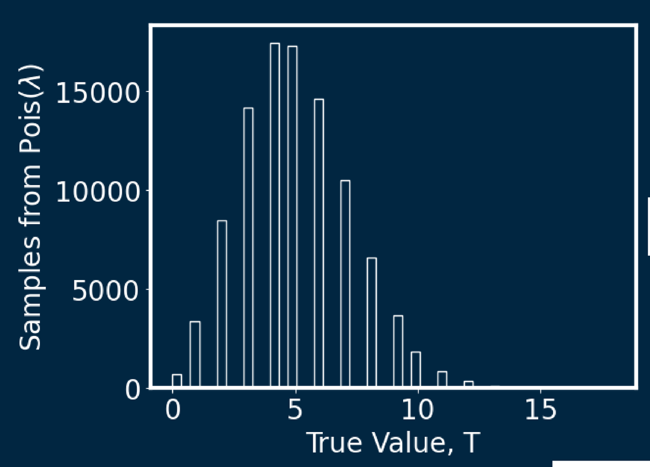
3: Flamedisx vs MC Example[1]

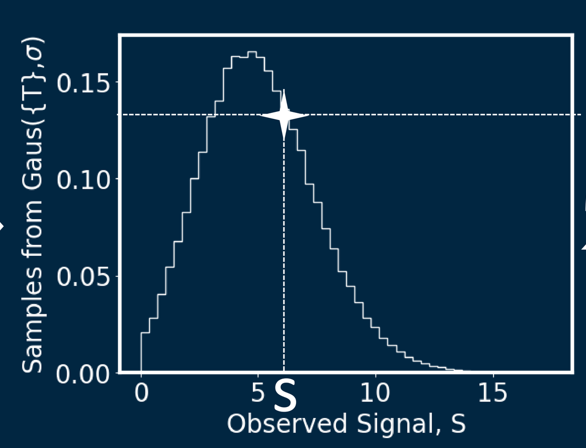
One-observable (S) problem of a model that depends on two parameters, λ and σ, that map a "true" spectrum T to a signal S.

$$P(S|\sigma,\lambda) = \sum_{T=-\infty}^{+\infty} Gaus(S|T,\sigma)Pois(T|\lambda)$$

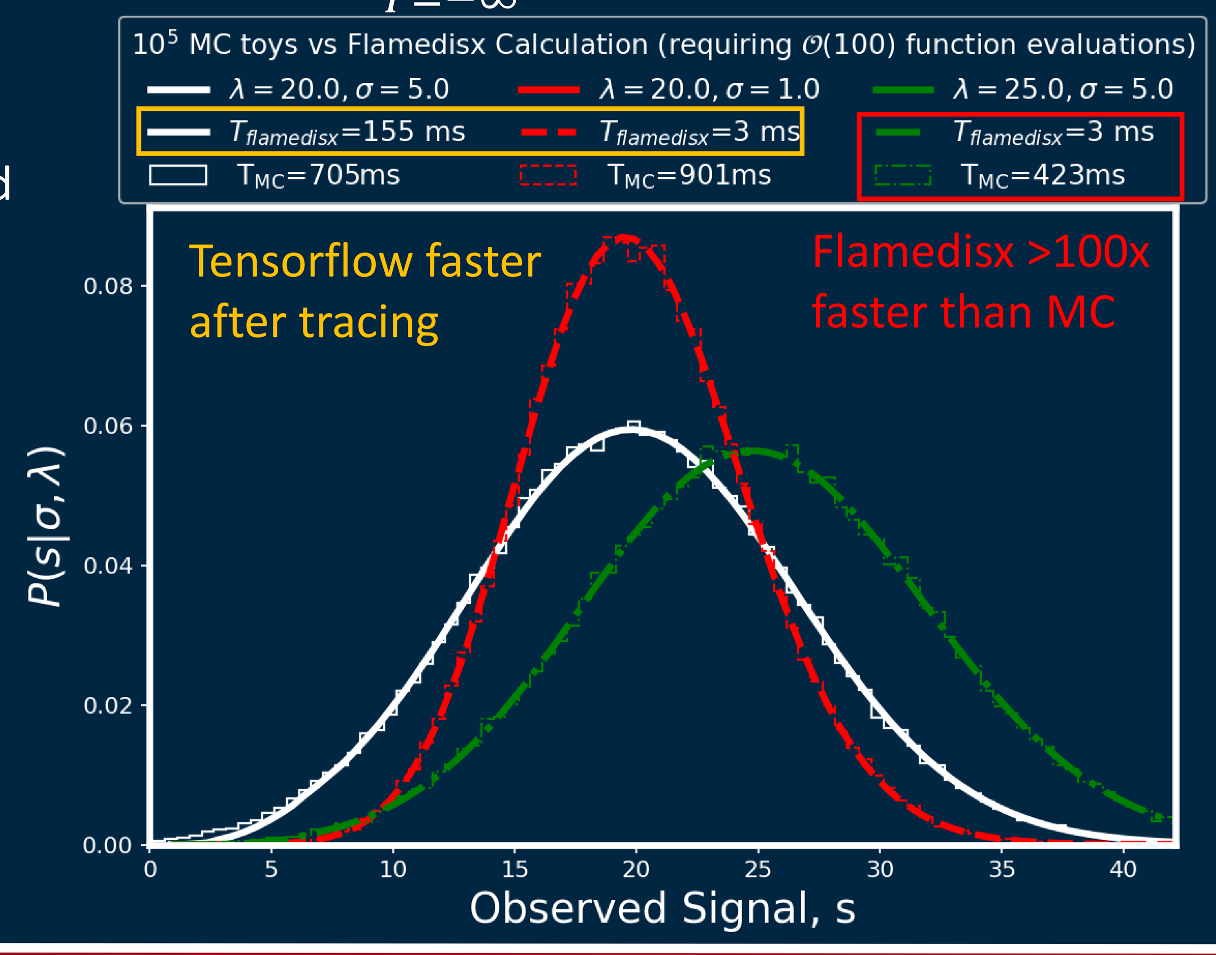
MC:

1. Fixed λ and σ
2. Propagate enough events through model to fill histogram
3. Normalise





Repeat process with different parameters and interpolate



Flamedisx:

Infer **approximate** range of parameters to sum over
Far fewer values to evaluate than for MC

$$\sum_{T\in\{T\}_{guess}} Gaus(s|T,\sigma)Pois(T|\lambda) = P'(s|\sigma,\lambda)$$

Evaluate the sum in explicitly **differentiable code**

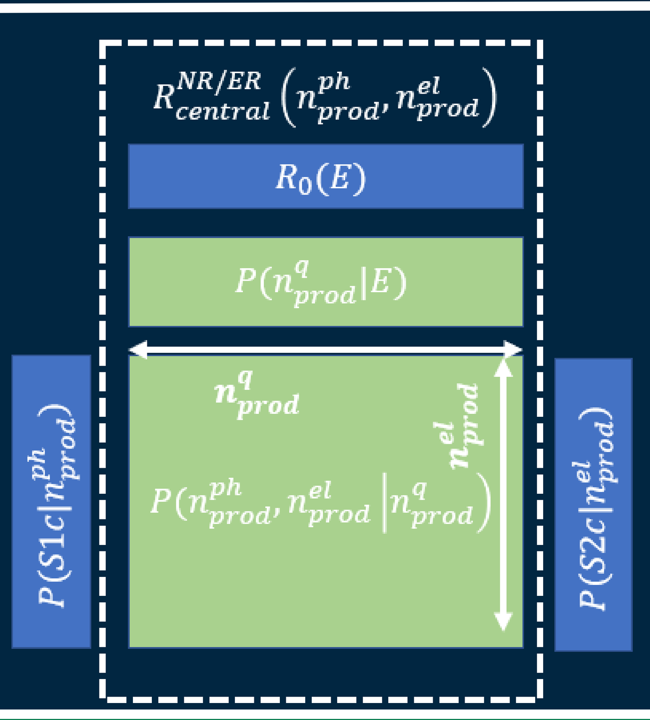
Note: if this calculation were visualised as below it would be

$$\frac{Gaus(S|T,\sigma)}{Pois(T|\lambda)}\ T$$

Differentiable Code:
Explicit graph of approx. probability function:

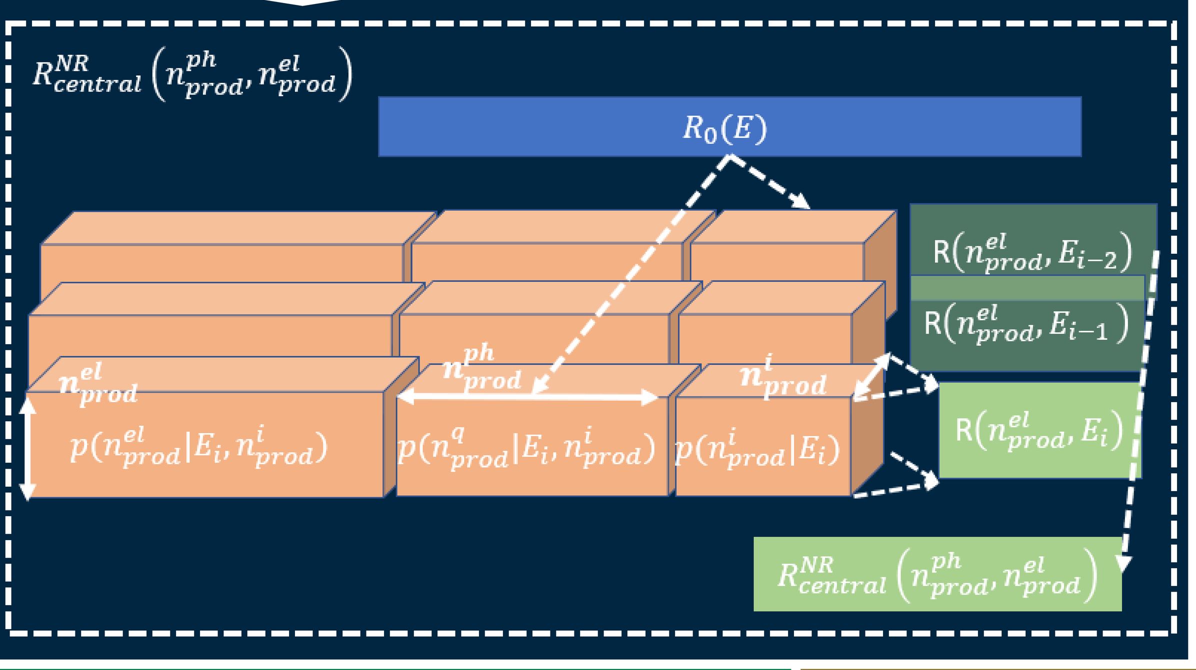
- Fast re-evaluation for new values
- **Auto-differentiation** for optimisation

4: FlameNEST [2]



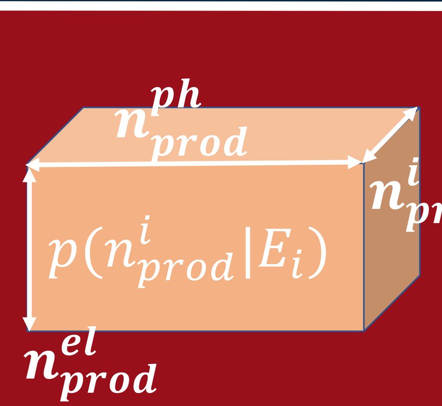
NEST modifies the xenon scintillation/ionisation yield model:

New internal dimension and more energy dependence to **model recombination.**



5: The Problem

In differentiable programming each evaluation of a function represents a graph of primitive functions.



New models are evaluated on excess dimensions increasing memory usage:

- Reduced performance for fitting all parameters
- **Impossible to fit yield model parameters**

6: The Solution:

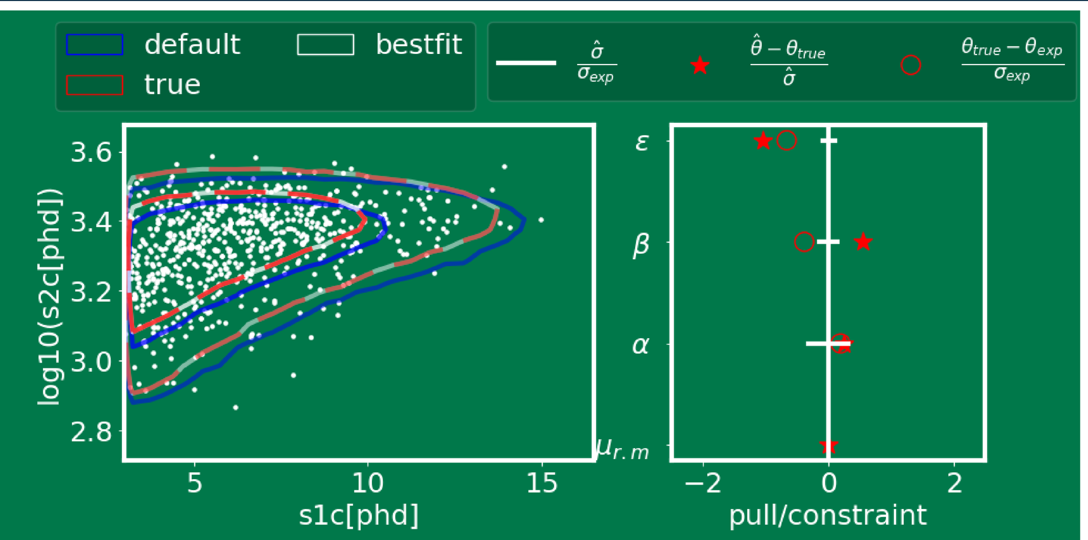
Lift the degeneracy!

- Evaluate unique inputs for models and map out to correct dimensions.
- This also uses time and memory so can only do when degeneracy lifted.

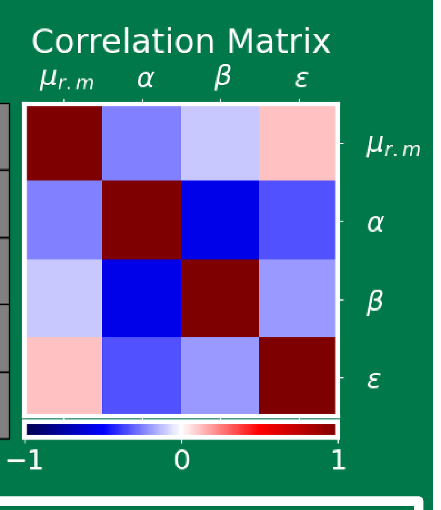
Resulting in:

- **28 x reduction in memory for fitting yield model parameters**
- 6 x reduction in other model parameters
  - **Other fits sped up 6x** (parallel processing).

Auto-differentiation gives **immediate uncertainty quantification** using the covariance matrix.



	predicted	True	bestfit
$\mu_{r.m}$	n/a	2.0	2.00 ± 0.09
α	11.0 ± 0.5	11.1	11.0 ± 0.2
β	1.10 ± 0.05	1.08	1.10 ± 0.01
ε	12.6 ± 2.9	10.7	10.4 ± 0.03



Three NR yield model parameters fit to a simulated data set, using a flat NR source from 0-10keV (true recoil energy) to emulate a near-threshold source. Using same cuts as in ref.[4].
Fit wall time 11minutes and 30mins to build total rate estimator.

7: Rate estimation Challenge

$$\ln\left(L\left(\vec{\theta}\,\middle|\{o_i\}\right)\right) = -\mu\left(\vec{\theta}\right) + \sum_i^{events} \ln\left(\sum_j^{sources} R^j\left(\vec{\theta},\{o_i\}\right)\right) + c$$

Here we've focussed on the **differential rate**, $R^j\left(\vec{\theta},\{o_i\}\right)$ term.

Need the **total rate** $\mu\left(\vec{\theta}\right)$ as a function to the parameters:

- Explicit integration would be too expensive.
- **Requires MC** to estimate, but far fewer events than evaluating R!
- For the test case, it was 3x the fit time to build the total rate estimator and that was for only 3 out of the 12 possible NEST NR parameters.

Future solutions may include implementation of Markov Chain Monte Carlo, more efficient MC estimation, or approximate integration tools.

Bibliography:
[1] Finding Dark Matter Faster with Explicit Profile Likelihoods 10.1103/PhysRevD.102.072010
[2] FlameNEST: explicit profile likelihoods with the Noble Element Simulation Technique 10.1088/1748-0221/17/08/P08012
[3] LUX-ZEPLIN Technical Design report Arxiv:1703.0914
[4] First Dark Matter Search Results from the LUX-ZEPLIN (LZ) Experiment 10.1103/PhysRevLett.131.041002

FCT Fundação para a Ciência e a Tecnologia
UKRI Science and Technology Facilities Council
Sanford Underground Research Facility
Swiss National Science Foundation
ibs Institute for Basic Science