Monte Carlo Pulse Shape Discrimination model and fitter for liquid Argon dark matter detectors

Sarthak Choudhary, Manish Gupta, Łukasz Pawela, Marcin Kuźniak, Tina Pollmann, Jan Ruhland, Piotr Gawron

On behalf of
DEAP-3600 Collaboration

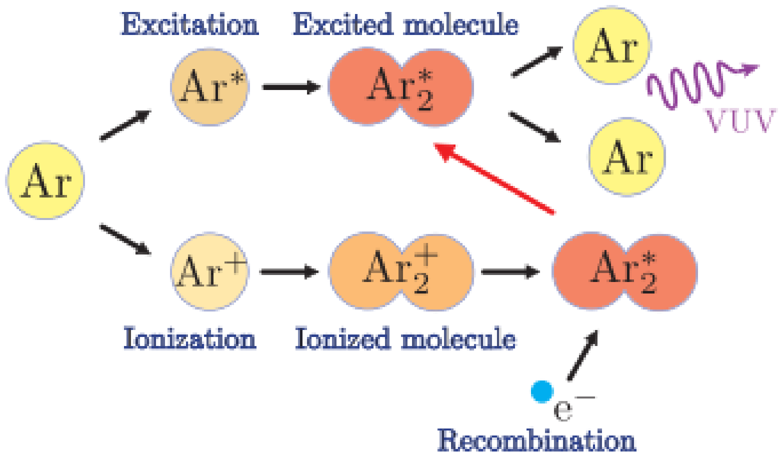
European Funds Smart Growth
Republic of Poland
FNP Foundation for Polish Science
European Union European Regional Development Fund
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952480
Dark Wave

Outline

I. Introduction
II. Toy Monte Carlo Model
III. PyTorch implementation

Argon Scintillation

- Particle colliding with Argon can lead to creation of excited dimers (excimers) of Argon via different mechanisms.
- Since singlet/triplet ratio depends on the charge density of the incident particle, it can be used to identify the incident particle.
- β^- particles released in decay of ^{39}Ar are main source of background in the energy range of interest.



The two mechanisms leading to the emission of 128 nm photon.
Ref: Amsler et. al., JINST 3:P02001,2008

Pulse Shape Discrimination (PSD) Parameter

Time constants for singlet and triplet:

$\tau_1 \sim 6$ ns

$\tau_3 \sim 1.5\ \mu s$

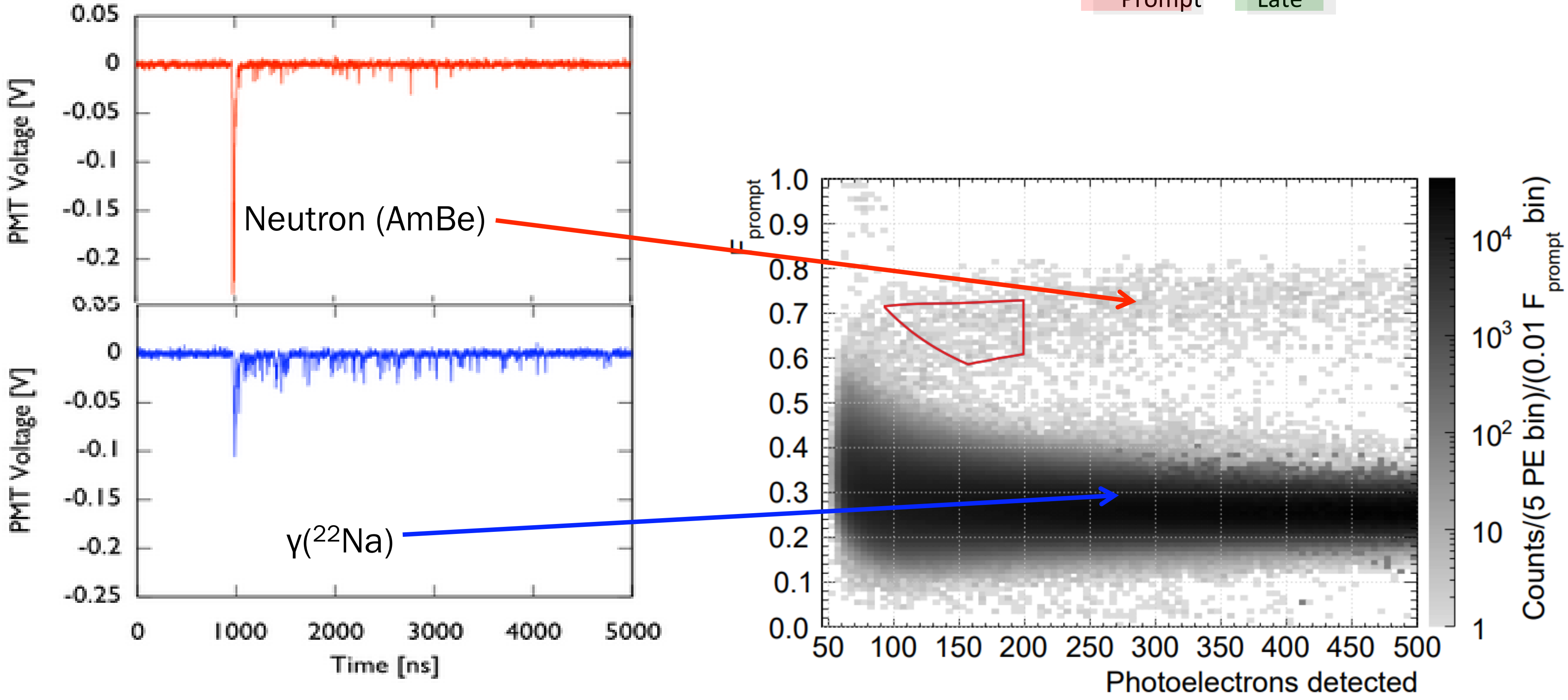
Prompt : 0-60ns

Late: 60ns-10μs

$$F_{prompt} = \frac{N_{Prompt}}{N_{Prompt} + N_{Late}}$$

Signal like (Nuclear recoil)

Background like (Electron recoil)



PMT signal for different sources. Ref: Phys. Rev. D 100, 022004 (2019)

PSD in DEAP-3600

- For WIMP search using PSD, we have to estimate the number of background events in the WIMP search region. Any events above expected number of background events would be a potential WIMP event. This necessitates a very good model for PSD.
- In a study[1], fit to DEAP-3600 science data were carried out with an **effective** PSD model.

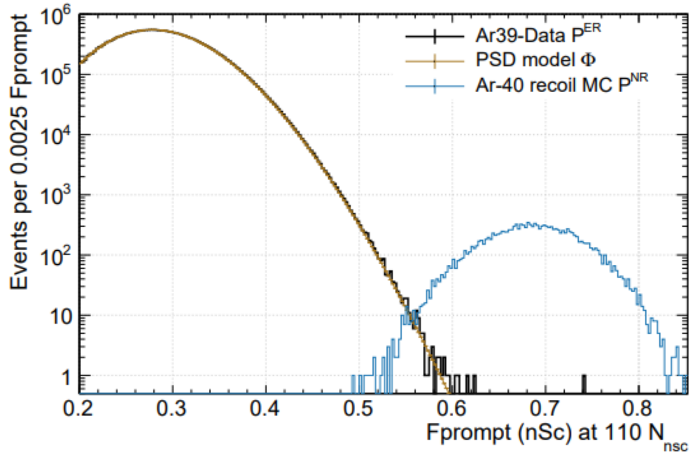
$$\Phi(x) = I \cdot \Gamma(x; \mu, b) * \mathrm{Gauss}(x; \mu = 0, \sigma)$$

$$b(N_{\mathrm{nsc}}) = a_0 + \frac{a_1}{N_{\mathrm{nsc}}} + \frac{a_2}{N_{\mathrm{nsc}}^2}$$

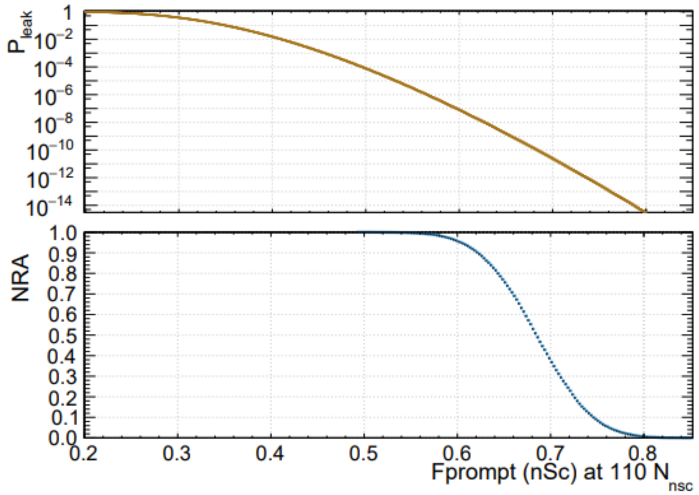
$$\sigma(N_{\mathrm{nsc}}) = a_3 + \frac{a_4}{N_{\mathrm{nsc}}} + \frac{a_5}{N_{\mathrm{nsc}}^2}$$

$$\mu(N_{\mathrm{nsc}}) = a_6 + \frac{a_7}{N_{\mathrm{nsc}}} + \frac{a_8}{N_{\mathrm{nsc}}^2} + \frac{a_9}{N_{\mathrm{nsc}}^3}$$

1 Ref: Eur. Phys. J. C (2021) **81**: 823



(a)



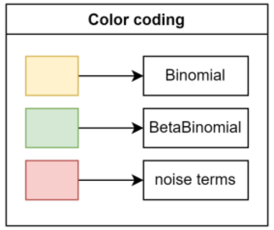
(b)

(a) The F_{nsc} prompt distributions at 110 N_{nsc} are shown for ^{39}Ar β events (background), together with the model fit (b) The background leakage probability (based on the fit model to 39Ar data) and signal acceptance (based on signal MC) as a function of the PSD parameter is shown. Ref: Eur. Phys. J. C (2021) **81**: 823

PSD model limitations

- In **effective** PSD model.
  - Parameters are given by the best fit to the detector data. The model parameters have no physical meaning and hence can not be used for reliable extrapolation.
- Therefore, we need a **physical** PSD model, such as the one developed for DEAP-1 [Astroparticle Physics 85 (2016) 1-23].
  - A simplified **analytic** model was developed which couldn't include all the scintillation physics.
  - The toy **Monte Carlo** model overcomes that but the ROOT implementation proved too slow for fitting data.
- To overcome this, fast toy MC has now been implemented in python using **PyTorch**.

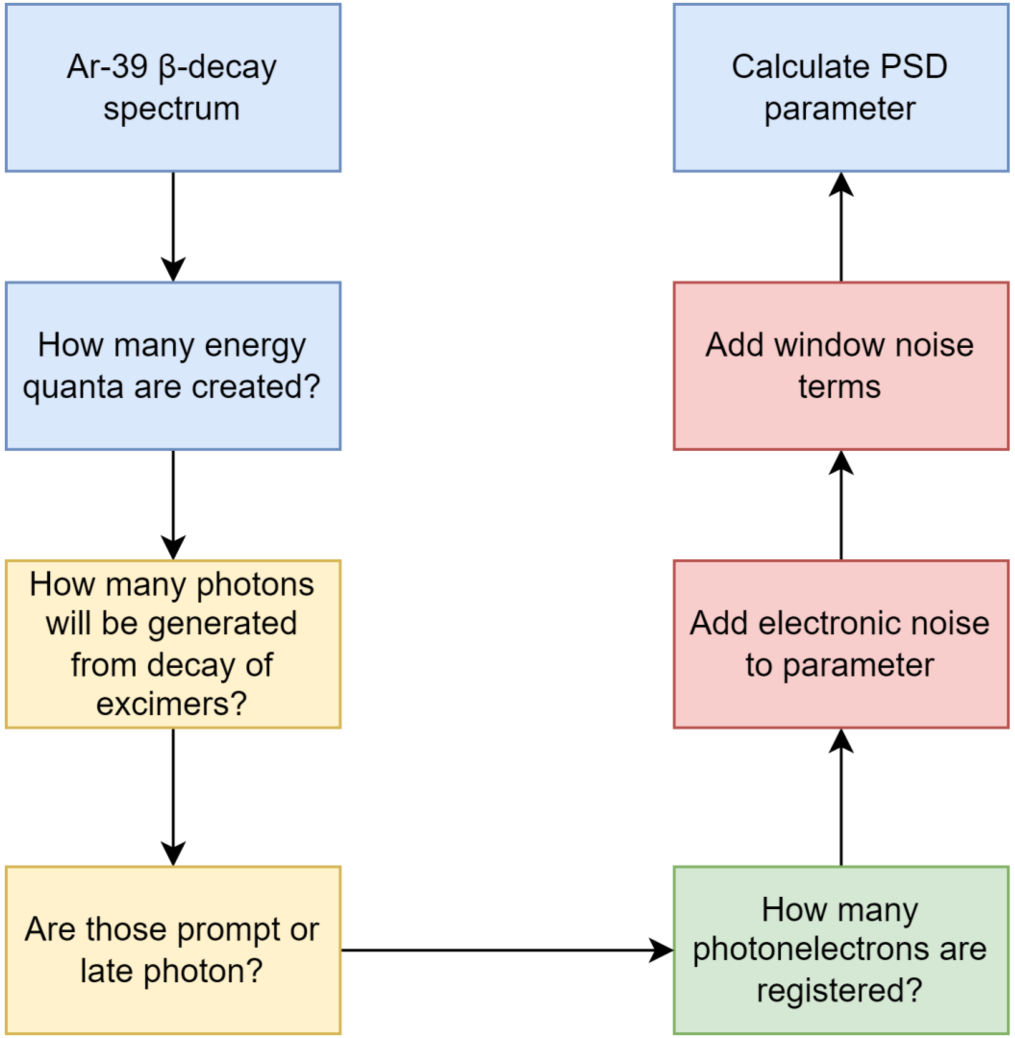
toy MC model



$$n_q = \mathrm{int}\left(\mathrm{Gauss}\left(E \cdot W, \sqrt{E \cdot W \cdot F_{Fano}}\right)\right)$$

$$n_{exc} = \mathrm{Binom}(n_q, \frac{ER}{1+ER})$$
$$n_{ph} = n_{exc} + \mathrm{Binom}(n_q - n_{exc}, RP)$$

$$n_{ph,p} = \mathrm{Binom}\left(n_{ph}, \quad \overline{pp}(E)\right)$$
$$n_{ph,l} = n_{ph} - n_{ph,p}$$



$$pp = \frac{n''_{pe,p}}{n''_{pe,p} + n''_{pe,l}}$$

$$\sigma_{w,pl} = \mathrm{Gauss}(0, \epsilon_w n'_{pe,l})$$
$$n''_{pe,l} = n'_{pe,l} + \sigma_{w,pl} + \delta_l$$
$$n''_{pe,p} = n'_{pe,p} - \sigma_{w,pl} + \delta_p$$

$$\sigma^2_{spe,p} = \epsilon^2_{spe} n_{pe,p}$$
$$\sigma^2_{spe,l} = \epsilon^2_{spe} n_{pe,l}$$
$$\sigma^2_p = \sigma^2_{spe,p} + \sigma^2_{elec,p}$$
$$\sigma^2_l = \sigma^2_{spe,l} + \sigma^2_{elec,l}$$
$$n'_{pe,p} = \mathrm{Gauss}(n_{pe,p}, \sigma_p)$$
$$n'_{pe,l} = \mathrm{Gauss}(npe_{,l,} \sigma_l)$$

$$n_{pe,p} = \mathrm{Binom}\left(n_{ph,p}, \mathrm{Beta}(\alpha, \beta)\right)$$
$$n_{pe,l} = \mathrm{Binom}\left(n_{ph,l}, \mathrm{Beta}(\alpha, \beta)\right)$$

Ref: M Szydagis *et al* 2011 *JINST* **6** P10002

Ref: *Astroparticle Physics* **85** (2016) 1-23

PyTorch implementation

- The MC model is implemented in python and heavily relies on PyTorch.
- PyTorch, developed by Meta AI, powers a variety of softwares, e.g., Tesla Autopilot, Uber pyro etc.

PyTorch

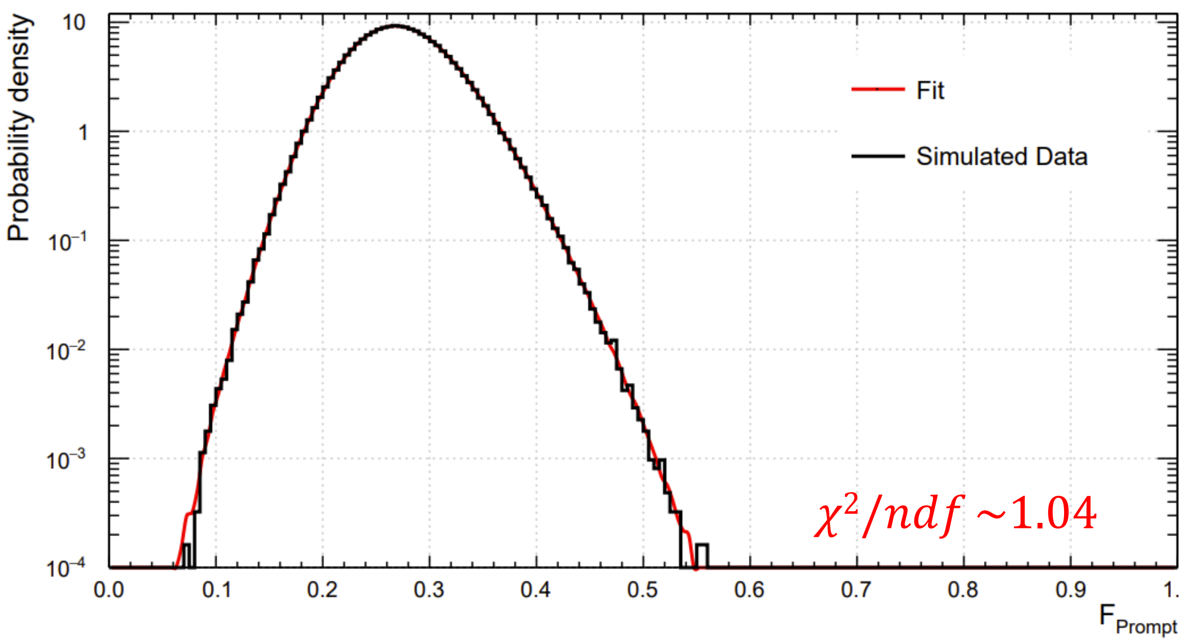
- PyTorch is an open source deep learning framework that provides a Python package for high-level features like multidimensional arrays (like NumPy) with strong GPU acceleration.
- PyTorch is a python and C++/java framework and has fast sampling procedures. Supports parallel and distributed computing.
- Python packages such as NumPy, SciPy, and Cython can be used to extend PyTorch functionalities and services.
- Widely supported and used in both industry and research.

PyTorch Build	Stable (1.7.1)	Preview (Nightly)			
Your OS	Linux	Mac	Windows		
Package	Conda	Pip	LibTorch	Source	
Language	Python	C++ / Java			
CUDA	9.2	10.1	10.2	11.0	None

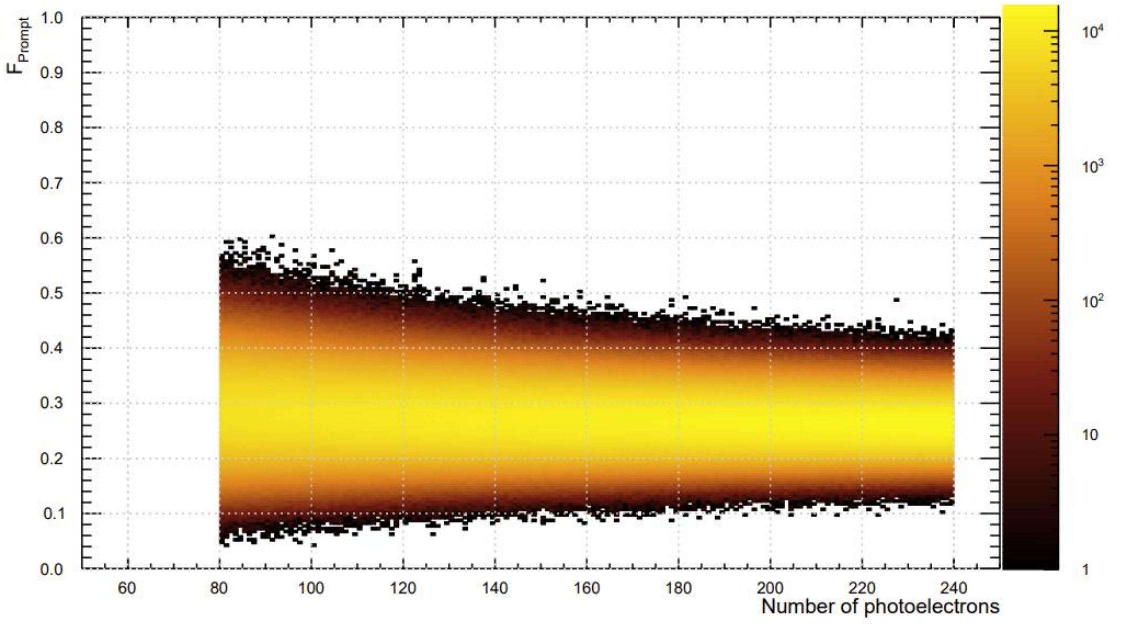
- With PyTorch’s inbuilt binomial generator and tensor operations, processing speed are increased by, at least, two orders on a single GPU for 10^8 samples.

Validation

- PyTorch code can perform 2 dimensional multiparameter fit.
- The plots below show fit to simulated data generated with the MC model itself with $\chi^2/ndf \sim 1.04$
- 3 x 10^6 events in fake data (~ 1 day of detector data)
- fitting time on a single GPU~ 12 minutes
- Fitter found to be consistent.



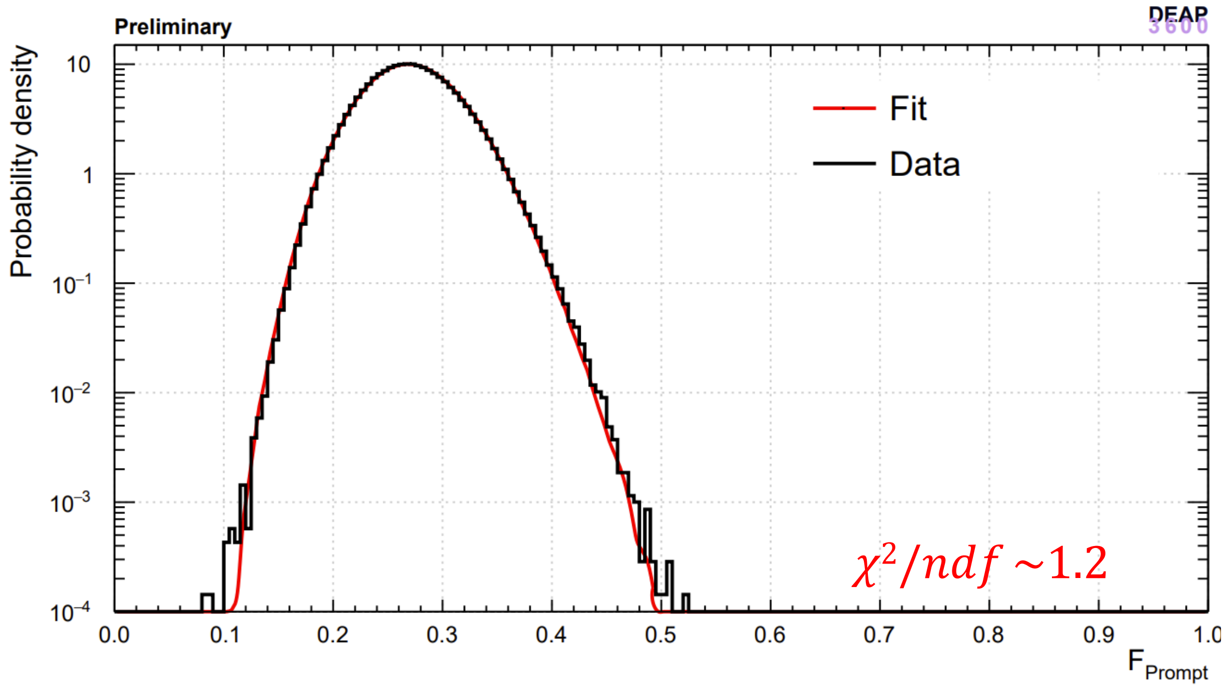
Model fit to simulated data.



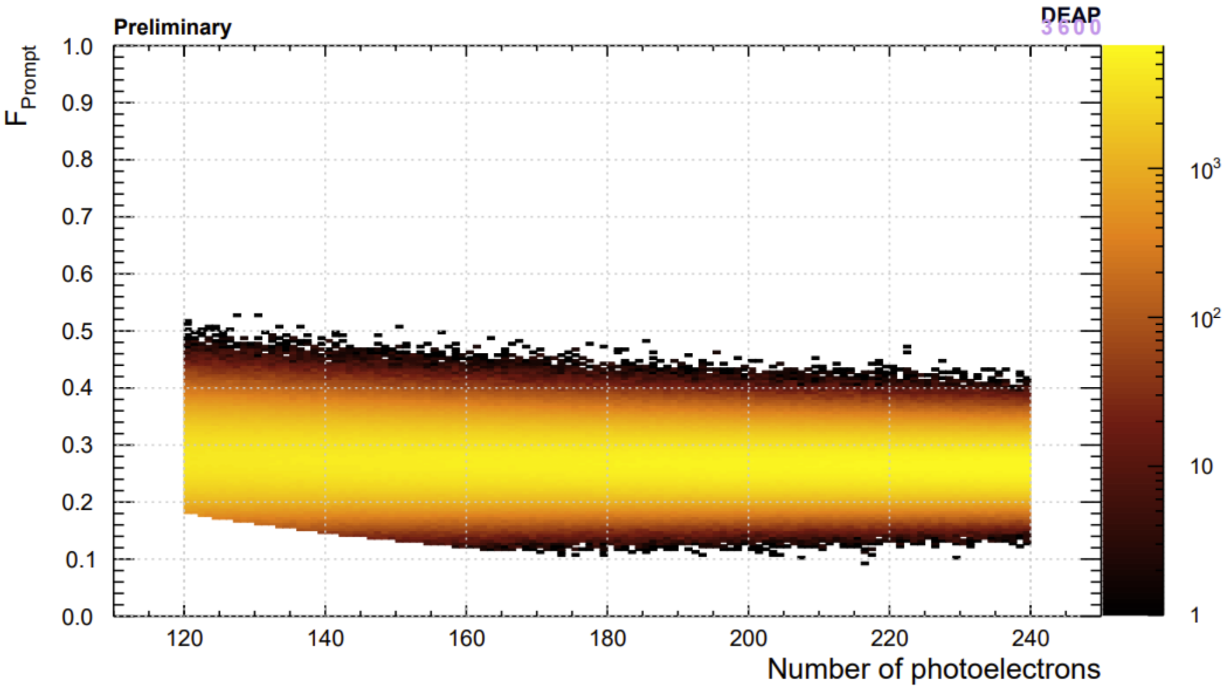
Model distribution for the best parameter.

Fit to DEAP-3600 data

- Fits to Science data in range 120 to 240 photoelectrons are shown in the plots below.
- Currently, tuning noise model to data.
- $\chi^2/ndf \sim 1.2$



Model fit to DEAP-3600 detector data.



Parameter distribution for the best fit.

Summary

- A general overview of Pulse Shape Discrimination in Argon dark matter detectors is presented.
- A physics based PSD model is presented.
- We have implemented the toy MC PSD model using **PyTorch for faster processing**.
- Model details, such as, **noise terms** can be updated with relative ease.
- First **fits to DEAP-3600** data seem promising.

Future Plan

- **distributed computing** on a farm of GPU’s.
- Apply the **background model** for DEAP-3600 data.

Thank You

Performance boost with PyTorch

PSD MC Sampling Time

Key Points:

✓ PyTorch based PSD MC is ~60 times faster than root based sampling on CPU for 10^8 samples.

✓ Pytorch based PSD MC is ~6 times faster on GPU than on CPU for 10^8 samples.

✓ Pytorch based PSD MC is ~360 times faster than root based implementation on GPU for 10^8 Samples

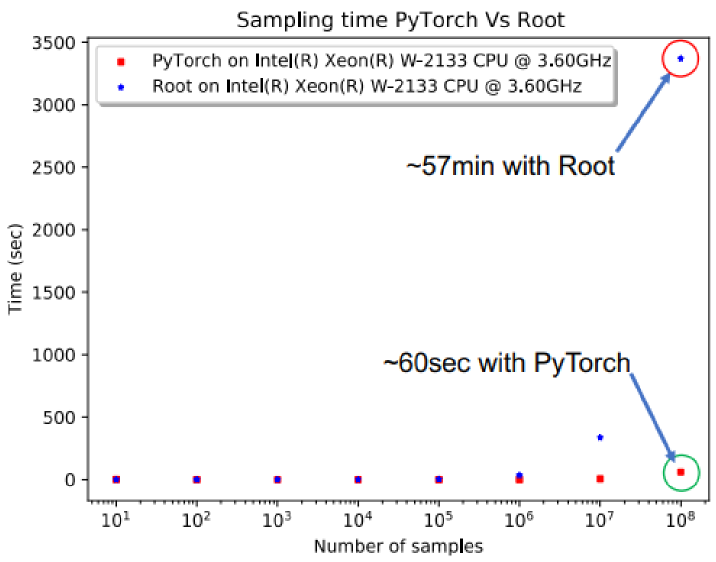


Fig. a

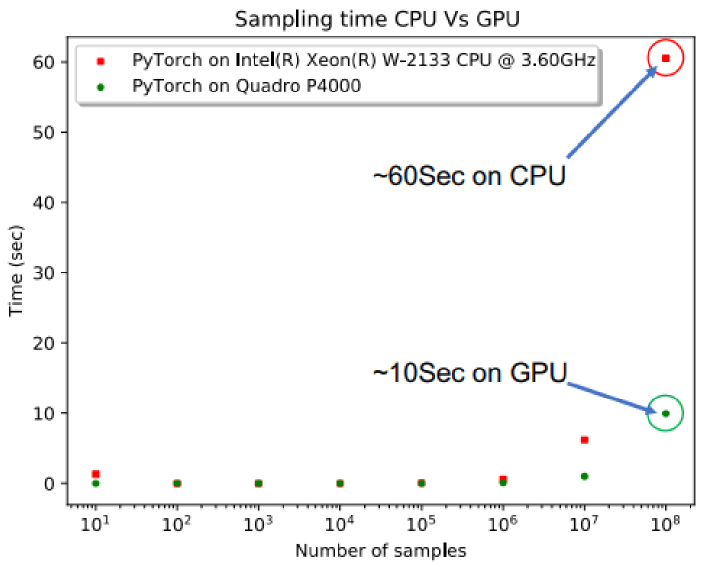
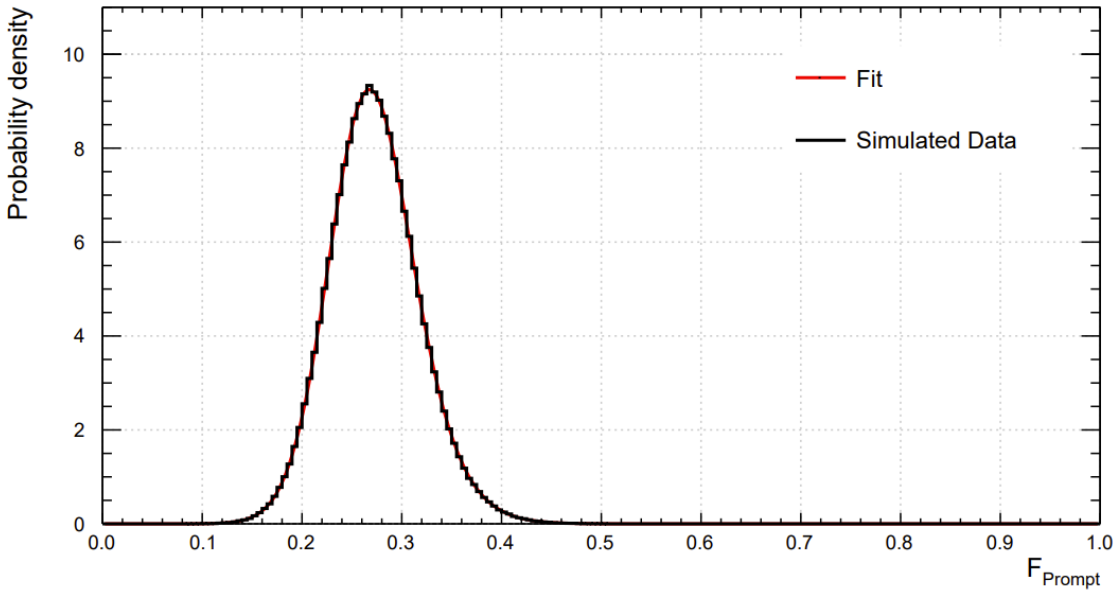


Fig. b

Ref: Manish K. Gupta

Fitter

- Cost function: negative log likelihood
- tMinuit called via pyROOT
- Simplex minimization algorithm



Model fit to simulated data.

Validation: Agreement between input and best fit

Parameter	Input Value	Best fit value	Hesse approx. error
epsspec	0.673876	0.674953	4.11982e-04
epsw	0	2.90833e-03	1.22308e-03
selecprompt	0	Fixed	
seleclate	0.8	0.801280	7.92436e-01
b	335	335.000	3.24245e+01
pe_scale	0.95	Fixed	
deltal	1.3	Fixed	
deltap	0	Fixed	
p2	0.0771841	0.0771841	1.23894e-05
kq	8.12647e-08	8.12647e-08	7.77881e-11
tau3	1989.32	1989.32	3.63609e-01

Correlation Matrix

PARAMETER CORRELATION COEFFICIENTS

NO.	GLOBAL	1	2	4	5	9	10	11
1	0.52463	1.000	-0.520	-0.524	0.523	-0.121	0.480	-0.492
2	0.99566	-0.520	1.000	0.993	-0.996	0.235	-0.912	0.938
4	0.99833	-0.524	0.993	1.000	-0.998	0.235	-0.914	0.941
5	0.99890	0.523	-0.996	-0.998	1.000	-0.236	0.916	-0.942
9	0.24719	-0.121	0.235	0.235	-0.236	1.000	-0.235	0.204
10	0.91664	0.480	-0.912	-0.914	0.916	-0.235	1.000	-0.869
11	0.94217	-0.492	0.938	0.941	-0.942	0.204	-0.869	1.000

Fit to DEAP-3600 data

Fits to Science data are shown in the plots below:

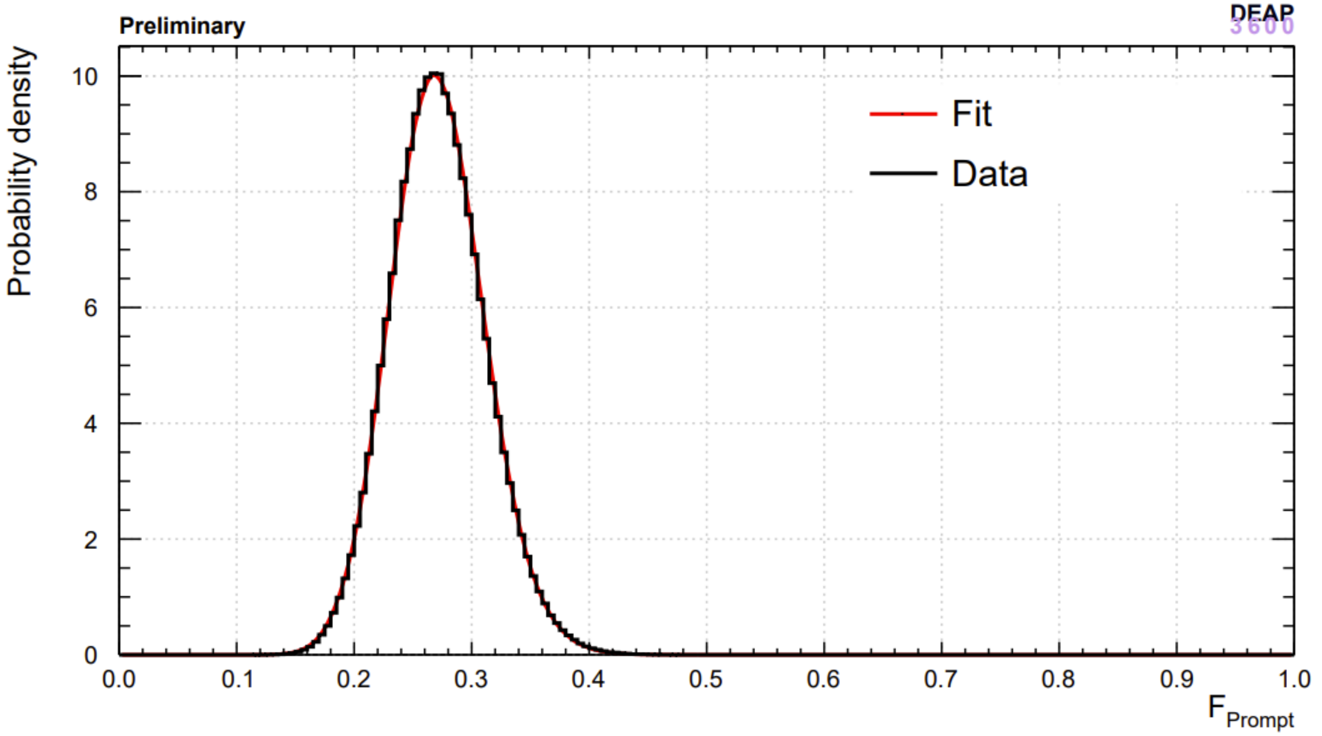


Fig: Model fit to DEAP-3600 detector data (linear Y-axis).

PSD models and limitations

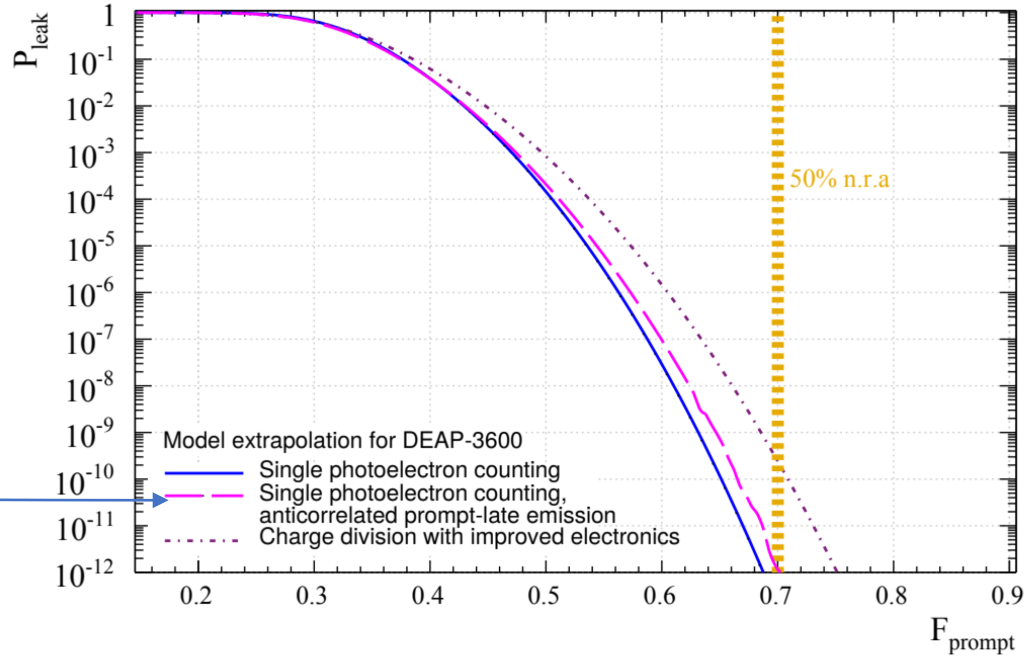
- For WIMP search using PSD, we have to estimate the number of background events in the WIMP search region. Any events above expected number of background events would be a potential WIMP event. This necessitates a very good model for PSD.
- In a study, fit to DEAP-3600 science data were carried out with an **effective** PSD model[1].
  - Parameters are given by the best fit to the detector data. The model parameters have no physical meaning and hence can not be used for reliable extrapolation.
- Therefore, we need a **physical** PSD model.
  - A simplified analytic model was developed which couldn't include all the detection physics.
  - The **Monte Carlo** model overcomes that but the ROOT implementation proved too slow for fitting data.

$$\Phi(x) = I \cdot \Gamma(x; \mu, b) * \text{Gauss}(x; \mu = 0, \sigma)$$

$$b(N_{\text{nsc}}) = a_0 + \frac{a_1}{N_{\text{nsc}}} + \frac{a_2}{N_{\text{nsc}}^2}$$

$$\sigma(N_{\text{nsc}}) = a_3 + \frac{a_4}{N_{\text{nsc}}} + \frac{a_5}{N_{\text{nsc}}^2}$$

$$\mu(N_{\text{nsc}}) = a_6 + \frac{a_7}{N_{\text{nsc}}} + \frac{a_8}{N_{\text{nsc}}^2} + \frac{a_9}{N_{\text{nsc}}^3}$$



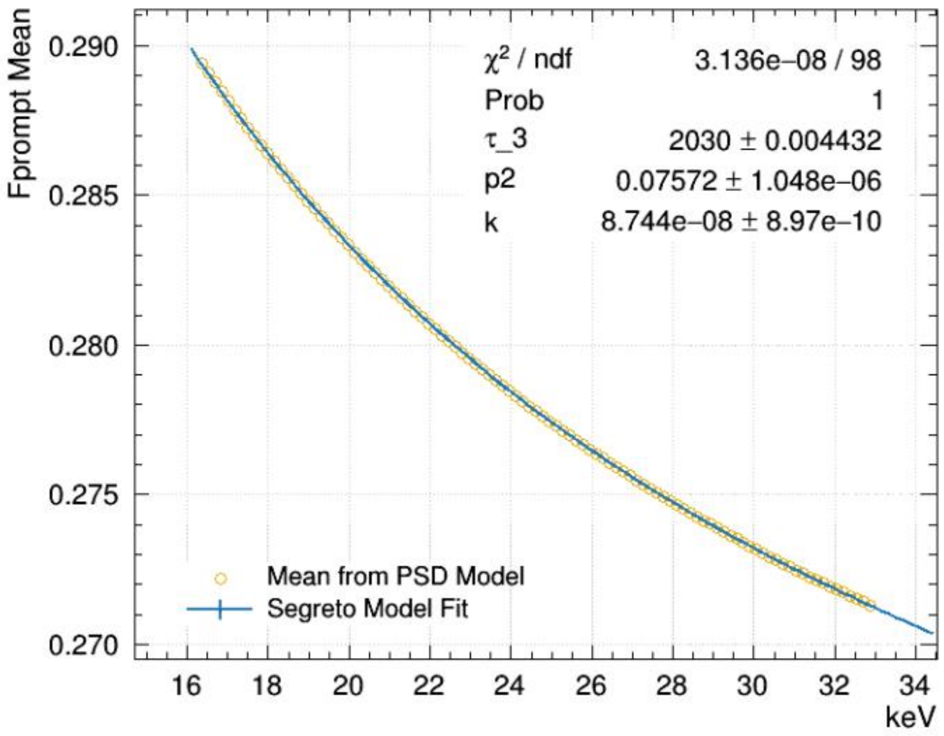
The dashed line was generated with a toy simulation following the logic of the analytic model. Ref: Astroparticle Physics 85 (2016) 1-23

1 Ref: Eur. Phys. J. C (2021) **81**: 823

Updated Analytic model

- Physics model for mean Fprompt vs. energy [E. Segreto, Phys. Rev. D 103, 043001 (2021)]. Reduces the number of free parameters compared to the effective model.
- Corrected variance model, to allow for a quadratic noise term

<u>New physical parametrisation of Fprompt Mean</u>



Ref: Jan Ruhland

Old parametrisation:

$$Fprompt_{mean}\left(N_{nsc}\right) = p_0 + \frac{p_1}{x - p_2} + \frac{p_3}{\left(x - p_4\right)^2}$$

New physicsal parametrisation:

$$Fprompt_{mean}\left(N_{nsc}\right) = \frac{L_1 + p_2 \cdot L_3}{L_1 + L_3}$$

L_1 : scintillation photons from singlet decay

L_3 : scintillation photons from triplet decay

p_2 : fraction of triplet light in prompt window

τ_3 : unquenched decay time of triplet component [ns]

k : characteristic constant $\left[\frac{\mu m^3}{ns \cdot MeV}\right]$

Terminology

- W, scintillation and ionization Yield
- Excitation Ratio, ER = E/I
- Probability(E) = ER/(1+ER)
- Recombination Probability, RP
- $\alpha = n_{ph} * b * \text{detection\_probability}$
- $\beta = n_{ph} * b * (1 - \text{detection\_probability})$