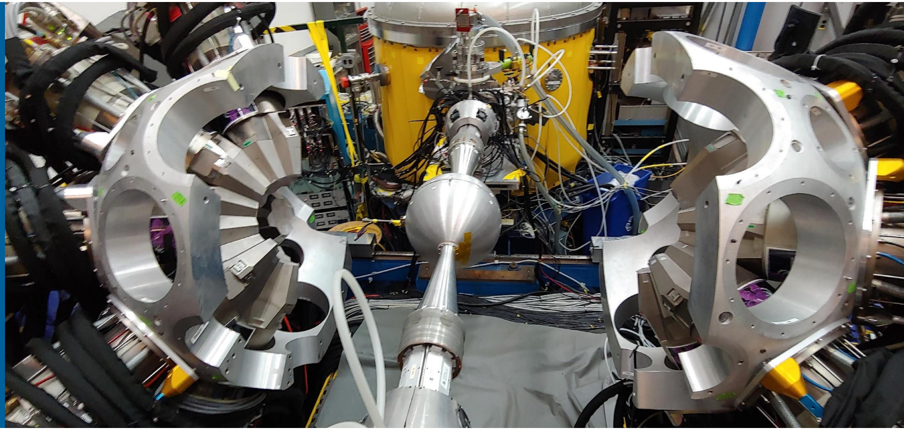
4th AGATA-GRETINA/GRETA Tracking Arrays Collaboration Meeting -- 11/21/2024

Tracking Developments with AI/ML

Thomas F Lynn



Mike Carpenter (FOA PI)
Torben Lauritsen

Sven Leyffer
Thomas Lynn

Dominic Yang

Amel Korichi

Physics Division
Argonne National Laboratory
Laboratory

Mathematics and Computing Science Division
Argonne National Laboratory

UCLA

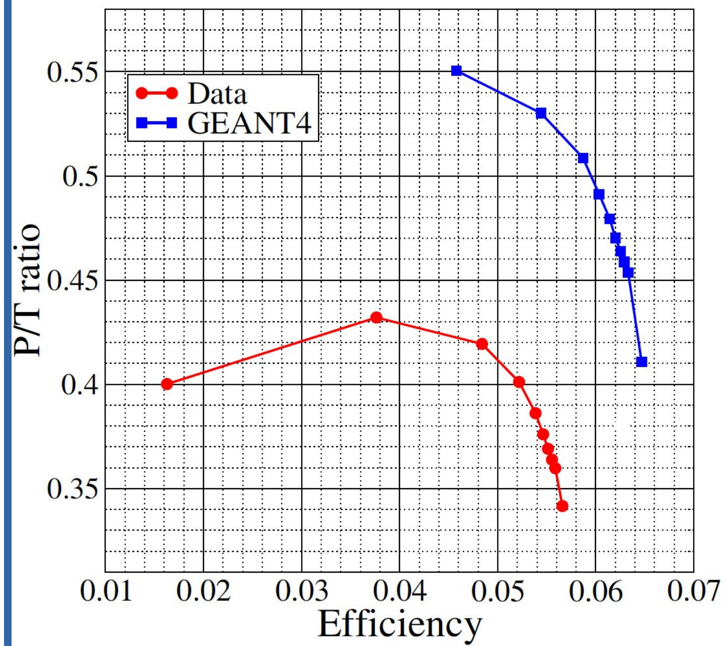
IJCLab IN2P3/CNRS
Irene Joliot Curie

ENERGY
U.S. Department of Energy laboratory managed by UChicago Argonne, LLC.

Project Goals

Machine-Learning (ML) tools for Gamma-Ray Tracking

Current tracking arrays (AGATA & GRETINA) do not meet the required performance



A. Korichi and T. Lauritsen, Eur. Phys. J. A (2019) 55: 121
AGATA-GRETINA Review paper

- Develop new techniques to enhance existing γ-ray tracking algorithms, boosting photopeak efficiency and improving the signal-to-background ratio (P/T).
- Adapt these techniques to accurately perform Doppler correction with the first interaction point (ordering!)
- Expand these methods to handle pair production events.
- Incorporate these tools into tracking codes used by the community.

Outline

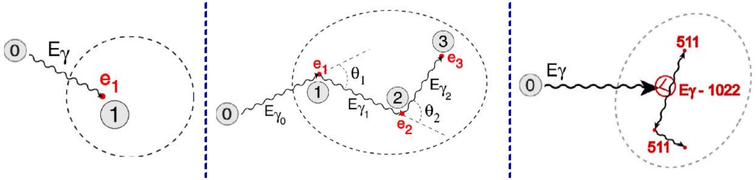
1. Available information for tracking
2. The tracking optimization problem
   a. The full problem
   b. Tracking in practice: Cluster, Order, Suppress
3. The choice of Figure of Merit(s)
4. Picking the best Figure of Merit (for simulated data)
5. Picking the best Figure of Merit (for experimental data)

Trackable γ-ray Interactions

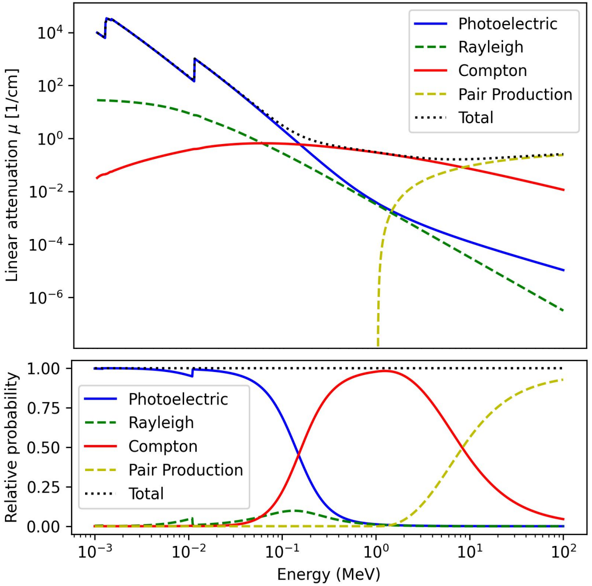
Three interaction types of interest

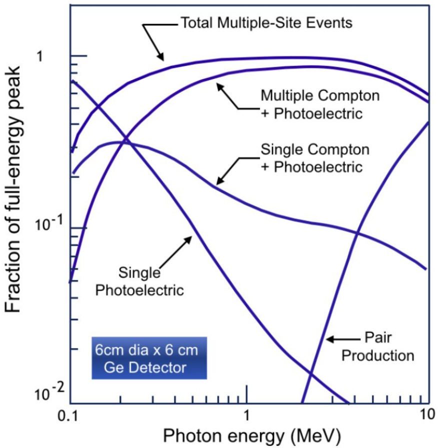
~ 100 keV | ~1 MeV | ~10 MeV | γ-ray energy

Photoelectric	Compton Scattering	Pair Production



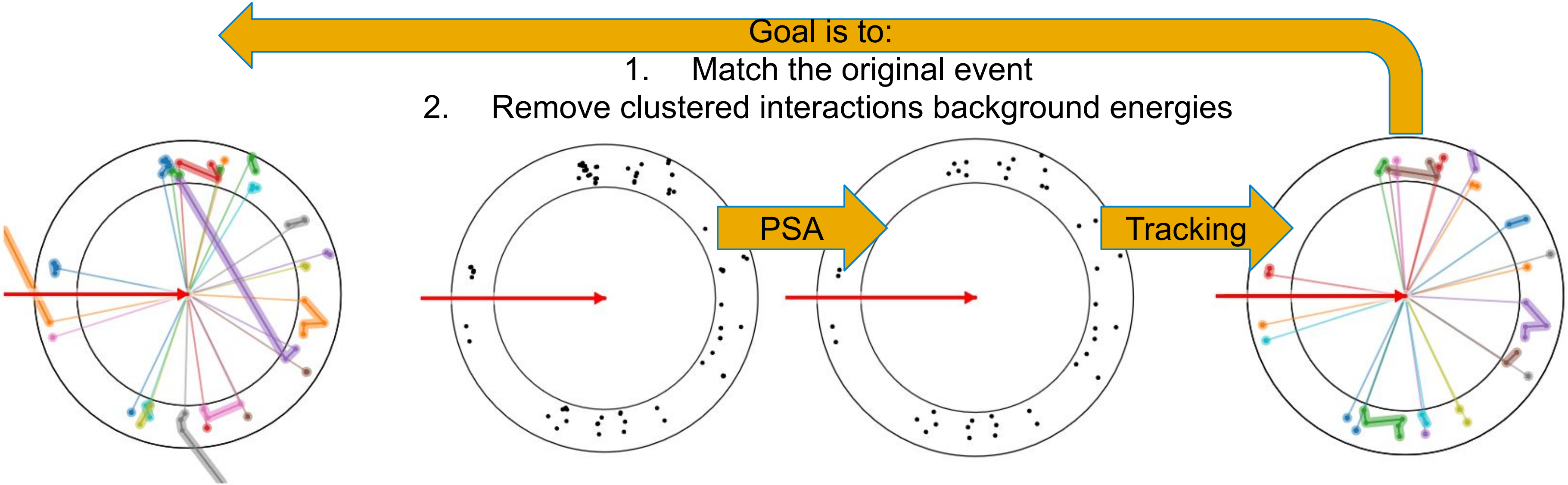
Isolated hits	Angle/Energy	Pattern of hits
Probability of Interaction depth	$E_{\gamma}^{'} = \dfrac{E_{\gamma}}{1 + \frac{E_{\gamma}}{m_0c^2}(1 - cos\theta)}$	$E_{1st} = E_{\gamma} - 2m_0c^2$





Goal of Tracking

Simulated event



Actual event: clustering/ordering of interactions

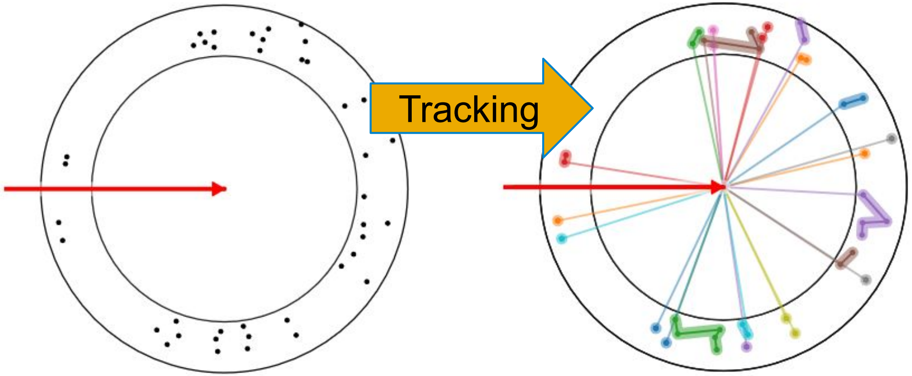
Actual interactions

Interactions recorded by PSA: “Packed and smeared”

Tracked event: interactions are clustered and ordered

The Full Tracking Problem

Organize interactions to recover the experimental event as best as possible



DATA: interaction positions and energies

GOAL: Find the ordered clusters of interaction that optimize a *Figure of Merit (FOM)*

What FOM recovers the event?

PROBLEM: Too many possible ordered clusters of interactions!

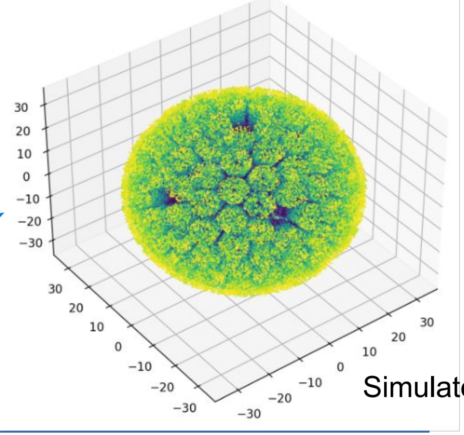
10 interactions → 58,941,091 possible ordered clusters
60 interactions → as many possibilities as atoms in the universe

In Practice: Cluster then Order

Detector
Local level

True hits

PSA/Decomposition hits

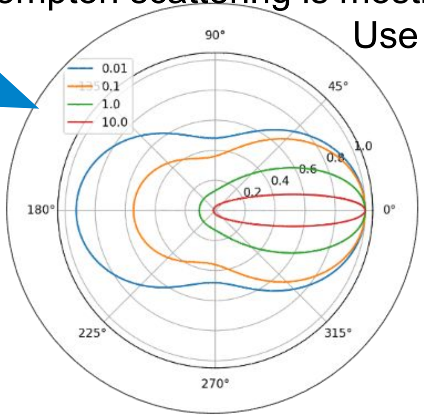


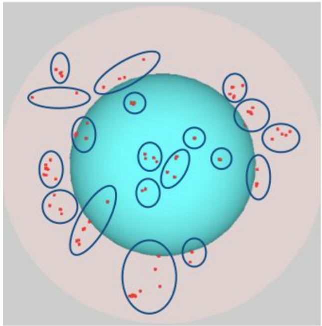
Simulated AGATA data

Global level

Hit Clusterization

Compton scattering is mostly forward (Klein-Nishina)
Use a cone clustering (alpha)





AFT & OFT

Order cluster interactions (use FOM)

FOM for ordered cluster

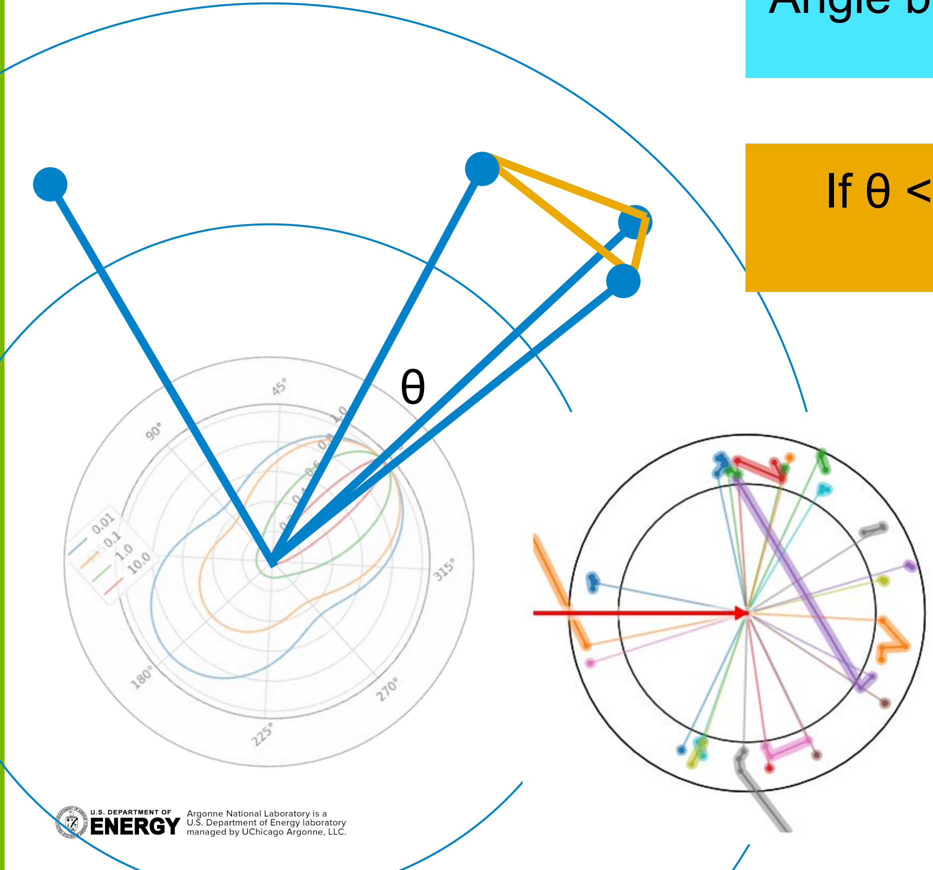
Accept cluster

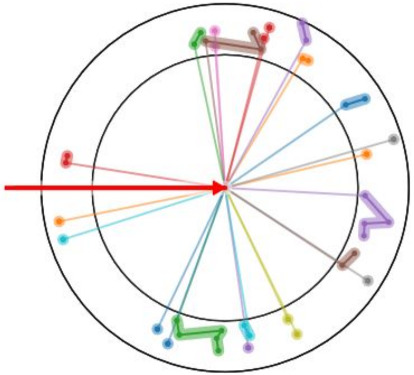
Reject cluster/mark

Cone Clustering

Angle between interactions θ

If θ < threshold angle α:
Same cluster





Great with:

- Low multiplicity

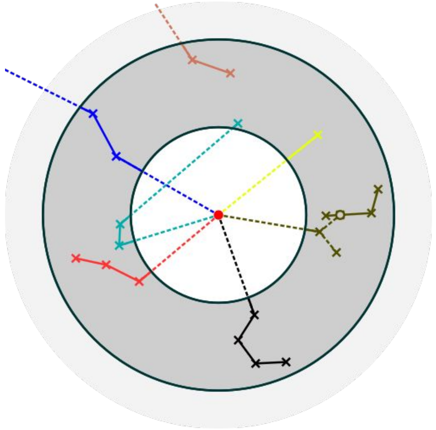
Bad with:

- γ-rays close together
- γ-rays that cross the detector

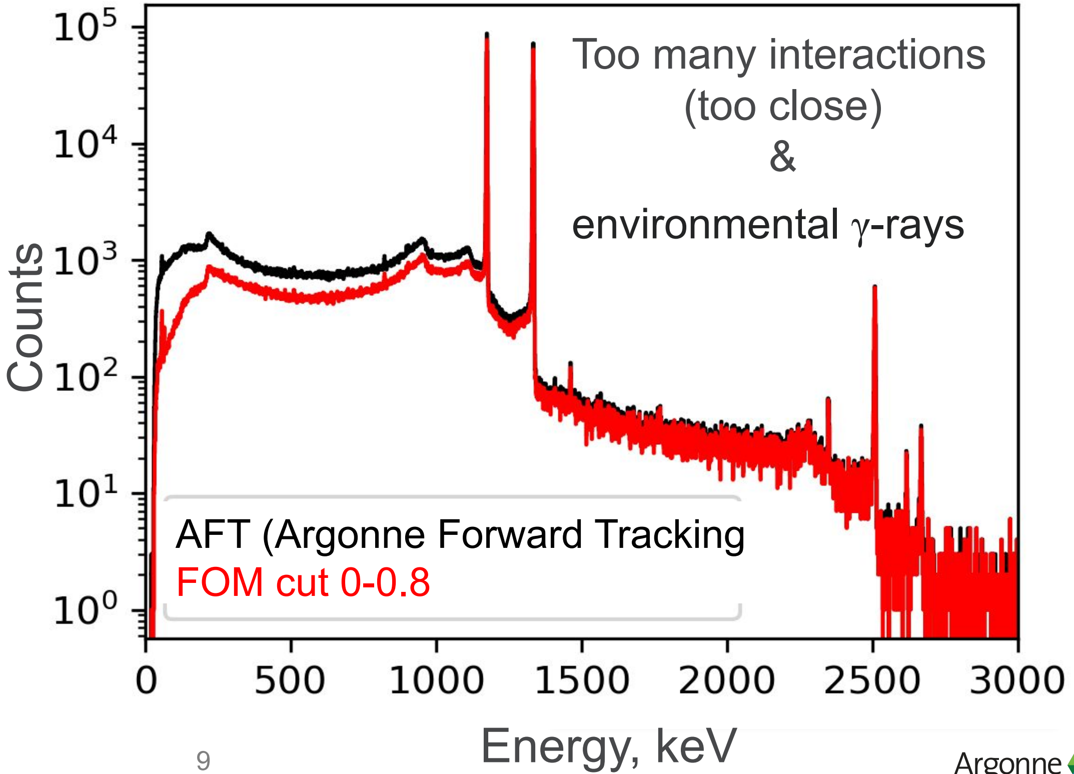
Attempts to use ML to improve clustering have not yet worked for experimental data

γ-ray Clustering Challenges

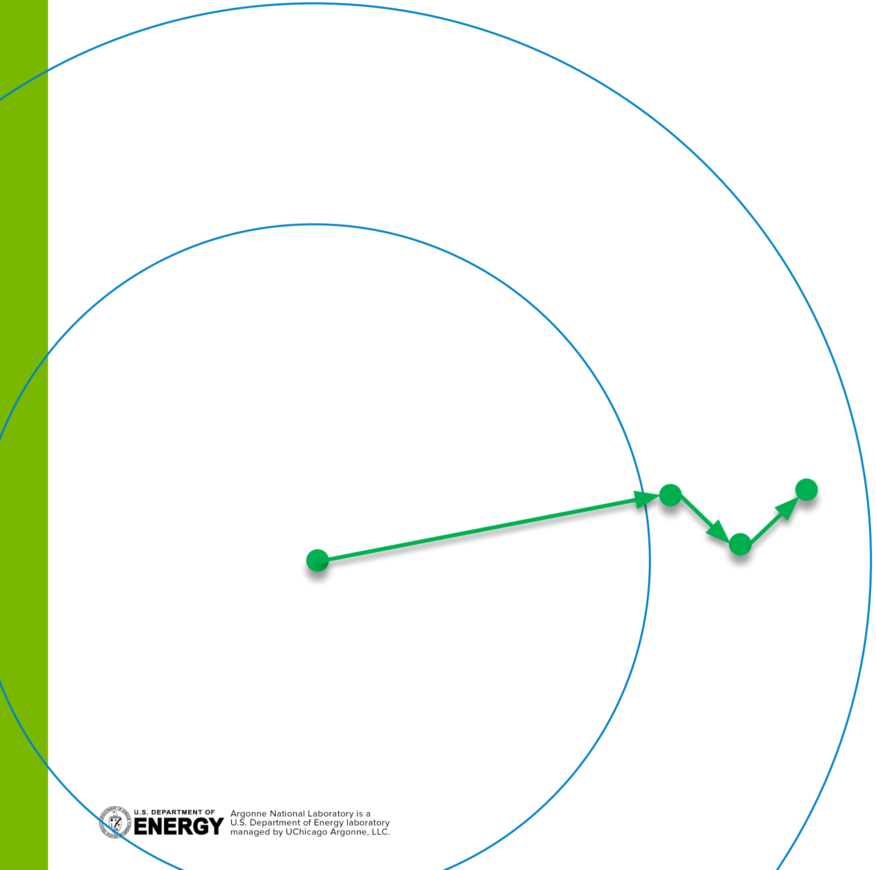
- γ-rays too close
- γ-rays escape
- γ-rays crossing the detector



^{60}Co Spectrum

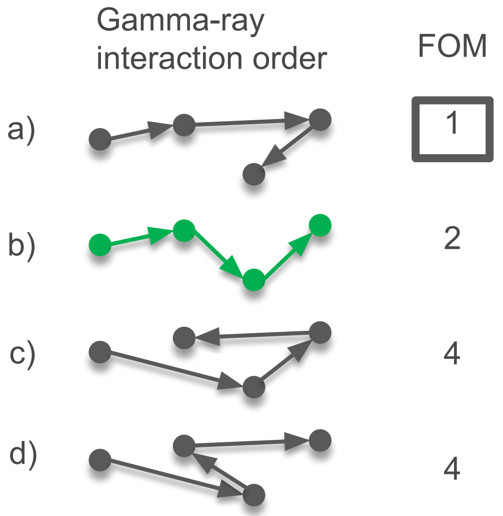


Ordering



Hypothesize cluster represents a complete γ-ray originating from the central target

Evaluate a FOM for all possible interaction orders



Select the order with the best FOM, whether correct or not

Assumption: If some energy is missing, even the best FOM would still be bad

Existing possible FOMs

$$E_{\text{out}} = \frac{E_{\text{in}}}{1 + \frac{E_{\text{in}}}{m_e c^2}(1 - \cos\theta)}$$

- Derive existing FOMs from Compton scattering formula, conservation of energy, and probabilities

$$E_{i-1} = \sum_{j=i}^{N} e_j$$

- Argonne Forward Tracking (AFT):

$$\left(\theta^{\text{theo}} - \theta^{\text{obs}}\right)^2$$

- Orsay Forward Tracking (OFT):

$$\left(E_{\text{out}}(e_{i+1:}) - E_{\text{out}}(\theta^{\text{obs}}, e_{i:})\right)^2 - \log(P)$$

- Mars Gamma-ray Tracking (MGT)

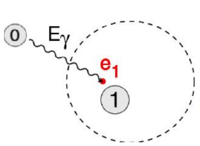
$$\left(\frac{E_{\text{in}}(e_{i:}) - E_{\text{in}}(\theta^{\text{obs}}, e_{i+1:})}{E_{\text{out}}}\right)^2$$

For perfect measurements, all squared error terms are *zero* for correctly ordered, complete energy γ-ray

With noise, all FOMS act differently. Ordering by any FOM may create errors.

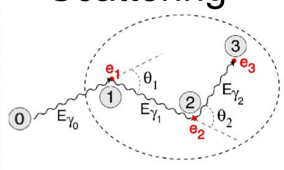
Where do current FOMs apply?

Single interactions

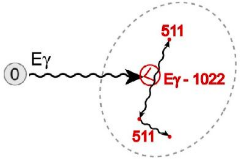


No order

Compton Scattering



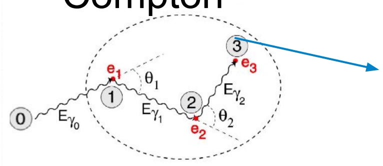
Pair production



No single order

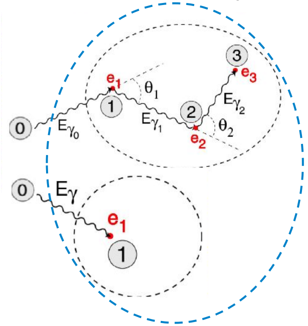
Previous FOMs don’t work

Escaped Compton



Previous FOMs don’t work

Cluster combining multiple rays

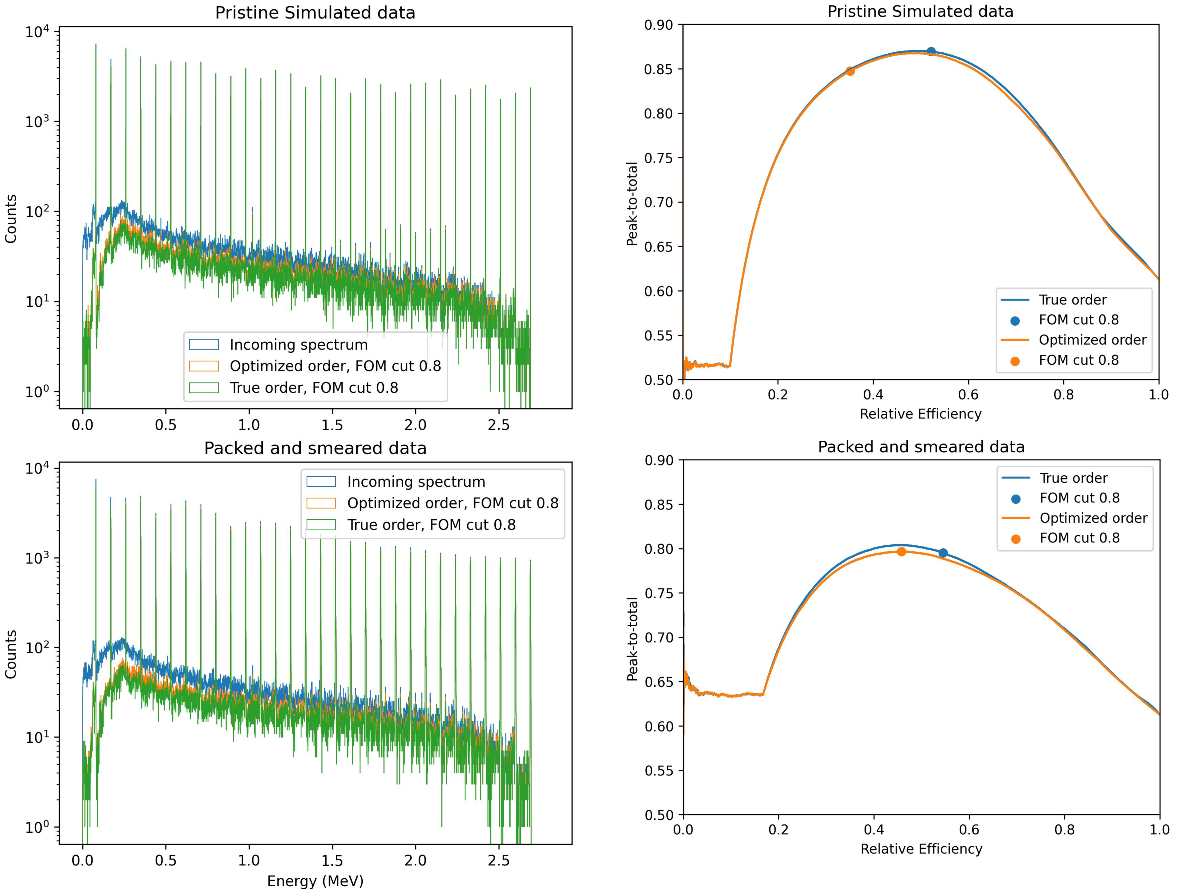


Or split into multiple clusters

No FOMs work

No order

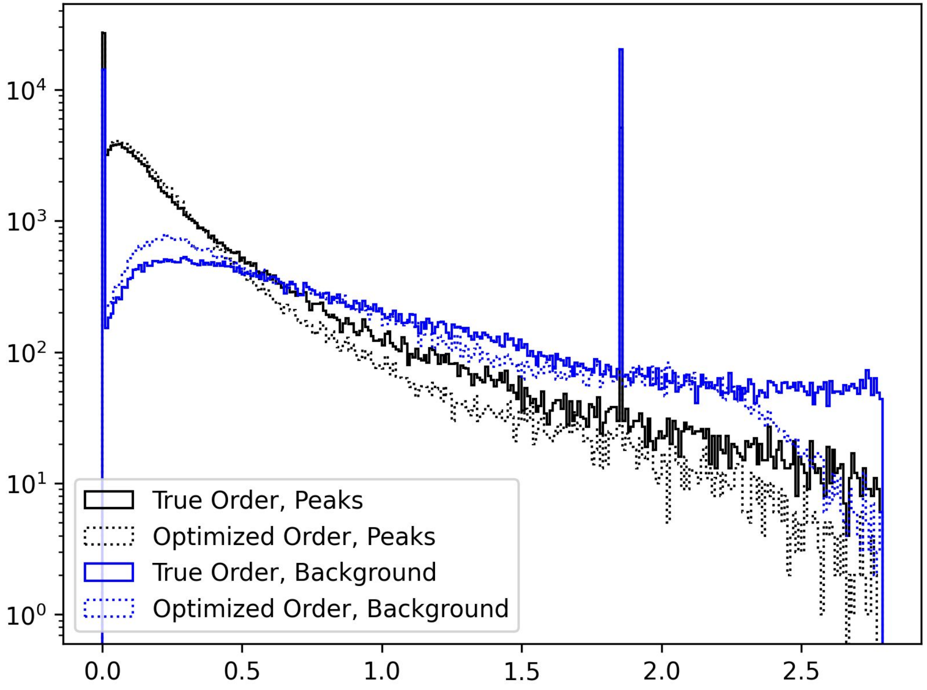
Simulated data using AFT



- 30 separate energies
- Provided correct clusters
- Ordered with pristine simulated data
- Ordered with packed-and-smeared values

Ordering simulated data using AFT

- Ordering process decreases FOM values. Selects:
  - True order
  - or False order with a better FOM
- Decreasing the FOM value for background counts makes suppression harder

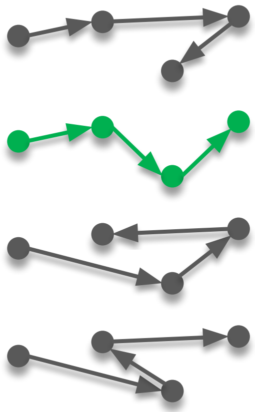


Optimizing interaction order for Doppler correction and linear polarization measurements

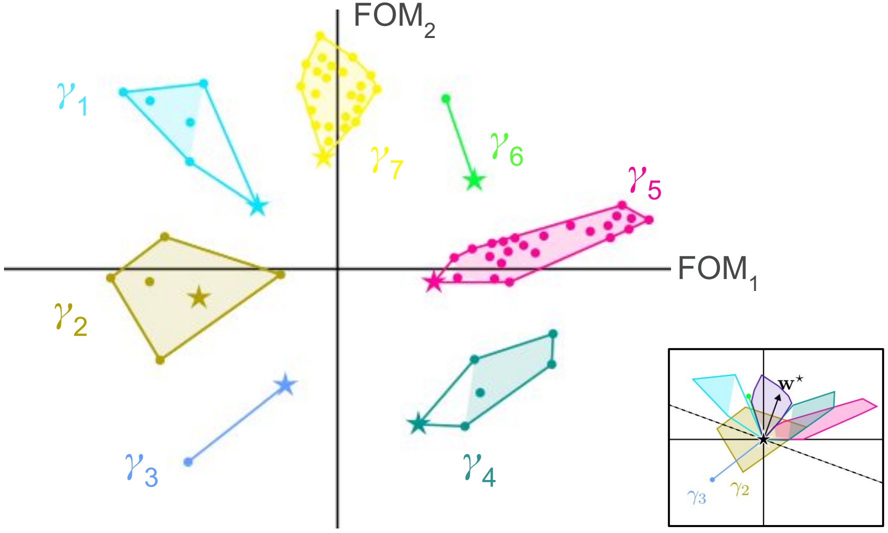
- Interaction order is needed for Doppler correction
  - Common with high v/c data that will be produced at ATLAS and FRIB
- Chosen by Figure-Of-Merit (FOM) value

Formally a Learning-to-rank (LTR) problem (e.g., search engine optimization)

FOMs and other features are combined to get the right order as often as possible



	Gamma-ray interaction order	FOM_1	FOM_2	$FOM_1 + FOM_2$
a)		$\boxed{1}$	4	5
b)		2	2	$\boxed{4}$
c)		4	4	8
d)		4	$\boxed{1}$	5



Hyperplane classification of relative cluster

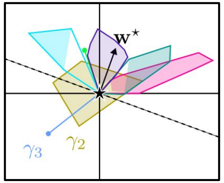
ML Approach for Learning-to-rank

- When ordering, we want
  FOM(best incorrect order) > FOM(true order)
- We don't care about the FOM value, only the difference between desired and undesired orders

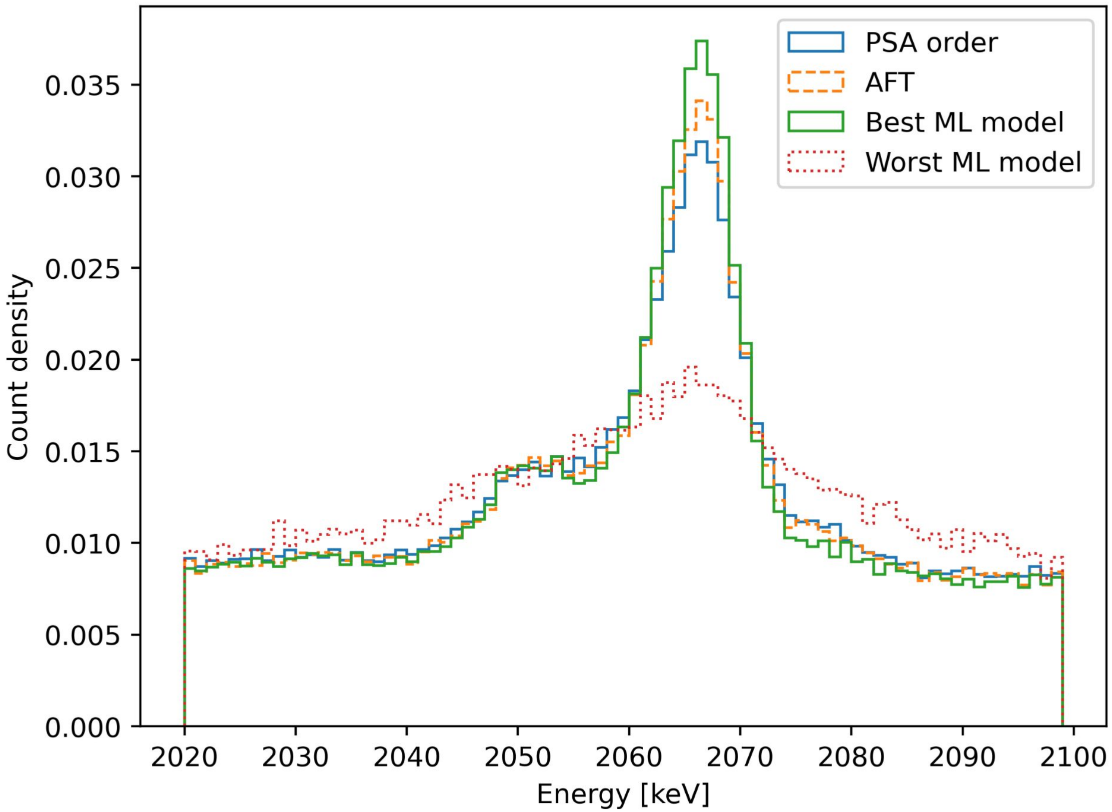
- The **best incorrect order** requires ordering with the FOM

- Let FOM be weighted sum of physics derived objectives (e.g. existing FOMs), a simple, interpretable model, that prevents overfitting (*maximizes likelihood that the model can survive the translation from simulated data to experimental data)*
  $$\mathbf{FOM(order) = w^T \mathit{f}(order)}$$
- Allows simplification
  $$\mathbf{w^T(\mathit{f}(incorrect) - \mathit{f}(true)) > 0}$$
- If all features/FOMs are quantities that we want to minimize, constrain **w** positive, protect against overfitting

- Use linear classification (introduce mirrored data as second class → off the shelf solvers)



Test ML models on ^{92}Mo in-beam data



Fusion-evaporation reaction

^{12}C(^{84}Kr,xn)

Beam Energy = 394 MeV

Recoil velocity ~8 %

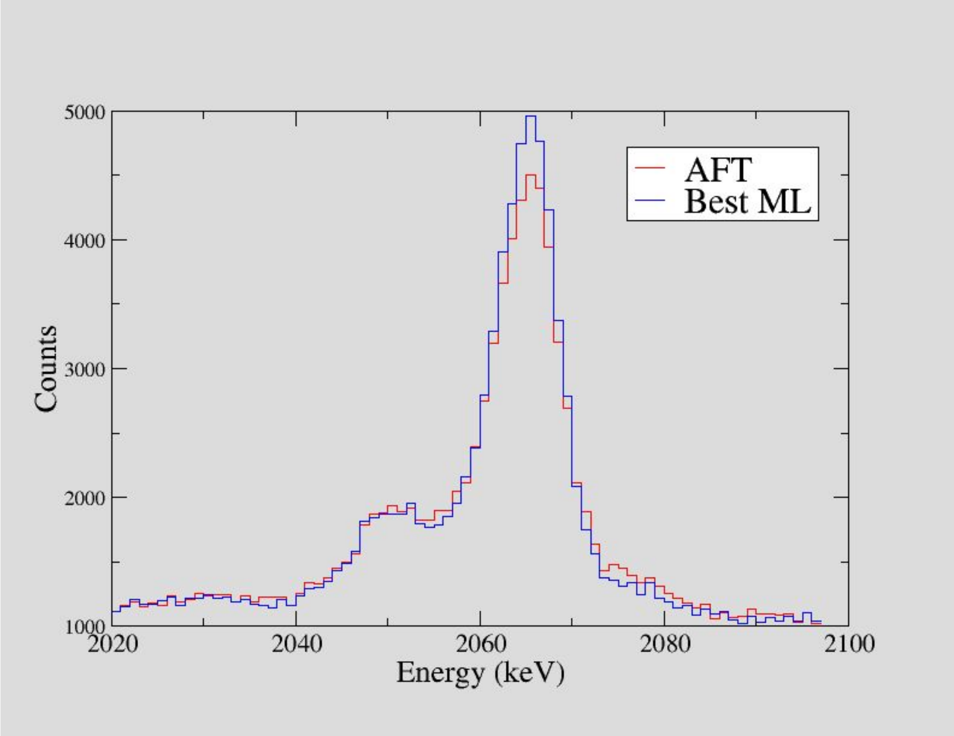
No FOM cut/supression. Only Doppler correction

Example of parameters, FOMs and models that have been used in this work

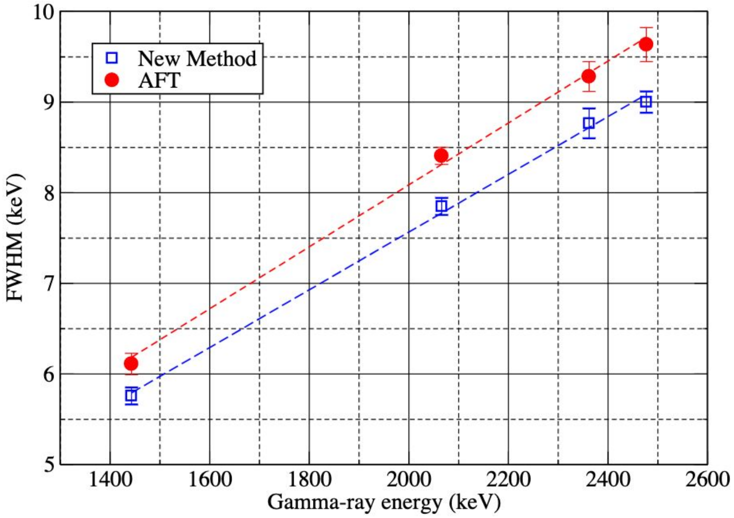
A	B	Simulated data						Experimental data	
		all_accuracy_correlation	all_accuracy_R	complete_accuracy_correlation	complete_accuracy_R	incomplete_accuracy_correlation	incomplete_accuracy_R	validation_accuracy_	validation_accuracy_R
C	C_1000	-0.058193674	0.058193674	-0.053454752	0.053454752	-0.052224147	0.052224147	0.20516106	0.00045849
	C_10000	0.058193674	0.058193674	0.053454752	0.053454752	0.052224147	0.052224147	-0.20516106	0.00045849
Columns	**cols_aft**	-0.076325647	0.076325647	-0.005300437	0.005300437	-0.204519661	0.204519661	-0.01204583	0.8387107
	cols_aft-fast	0.0888634	0.0888634	0.107414741	0.107414741	0.025623966	0.025623966	0.0385706	0.5144265
	cols_aft-fast-tango	0.128330901	0.128330901	0.109293607	0.109293607	0.133188852	0.133188852	0.07734063	0.19061326
	cols_aft-fastest	0.021426865	0.021426865	0.052850234	0.052850234	-0.050385041	0.050385041	-0.14379215	0.01459295
	cols_aft-fastest-tango	0.069065148	0.069065148	0.063813769	0.063813769	0.061197885	0.061197885	-0.07738052	0.19038397
	cols_aft-tango	-0.006229761	0.006229761	-0.003607784	0.003607784	-0.010028997	0.010028997	-0.07953441	0.1783003
	cols_aft-true	-0.432470377	0.432470377	-0.203709027	0.203709027	-0.794319516	0.794319516	0.08811009	0.13578374
	cols_all	0.157322643	0.157322643	0.126755755	0.126755755	0.178449978	0.178449978	0.36398759	0
	cols_fast	0.089000176	0.089000176	0.107563293	0.107563293	0.025698618	0.025698618	-0.06284176	0.28783962
	cols_fast-tango	0.128222102	0.128222102	0.109299883	0.109299883	0.132868287	0.132868287	0.05580173	0.3453722
	cols_fastest	-0.520524263	0.520524263	-0.620266848	0.620266848	-0.168822785	0.168822785	-0.13266372	0.02435088
	cols_oft	0.113075771	0.113075771	0.104667581	0.104667581	0.099797409	0.099797409	0.09618718	0.10330652
	cols_oft-fast	0.153309525	0.153309525	0.137876211	0.137876211	0.143771884	0.143771884	0.00680071	0.90851543
	cols_oft-fast-tango	0.16630786	0.16630786	0.13366864	0.13366864	0.189327228	0.189327228	-0.038431	0.51595111
	cols_oft-fastest	0.130196021	0.130196021	0.11340284	0.11340284	0.129833899	0.129833899	-0.06679056	0.25855802
	cols_oft-fastest-tango	0.140003446	0.140003446	0.10277407	0.10277407	0.179850803	0.179850803	-0.05422619	0.35918367
	cols_oft-tango	0.129167097	0.129167097	0.093509465	0.093509465	0.168679336	0.168679336	-0.05350823	0.36559044
	cols_oft-true	-0.478740905	0.478740905	-0.530005993	0.530005993	-0.240212145	0.240212145	-0.00558416	0.92482719
Model type	model_type_lp	0.043636361	0.043636361	0.038356837	0.038356837	0.042782798	0.042782798	0.06523047	0.26987133
	model_type_lr	-0.077810651	0.077810651	-0.06480834	0.06480834	-0.0838193	0.0838193	0.12590334	0.03269045
	model_type_milp	0.099322254	0.099322254	0.08493778	0.08493778	0.102348504	0.102348504	-0.23708948	0.00004821
	model_type_svm	-0.065147964	0.065147964	-0.058486277	0.058486277	-0.061312002	0.061312002	0.04595567	0.43721044
Non-negative	nonneg_False	0.00252598	0.00252598	-0.032721003	0.032721003	0.075811971	0.075811971	-0.30128662	0.00000019
	nonneg_True	-0.00252598	0.00252598	0.032721003	0.032721003	-0.075811971	0.075811971	0.30128662	0.00000019

C: Controls the sparsity of the model; a smaller C means a simpler model
Columns: Groups of FOM features
Model type: The approach for training the ML model
LP: Linear program (more precise than SVM), LR: Logistic regression (simplest, but least accurate)
MILP: Mixed integer linear program (most accurate), SVM: Support-vector machine (basic linear model)
Non-negative: If "noneg = True," all weights in the FOM are non-negative, focusing on minimizing values.
If "noneg = False," some weights can be negative, allowing for maximization.

Results for ^{92}Mo in-beam data



FWHM	Peak Area	Energy
8.02(6)	31763(266)	2065.63(4)
8.75(7)	30169(277)	2065.65(5)

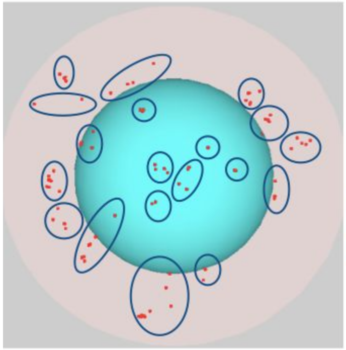


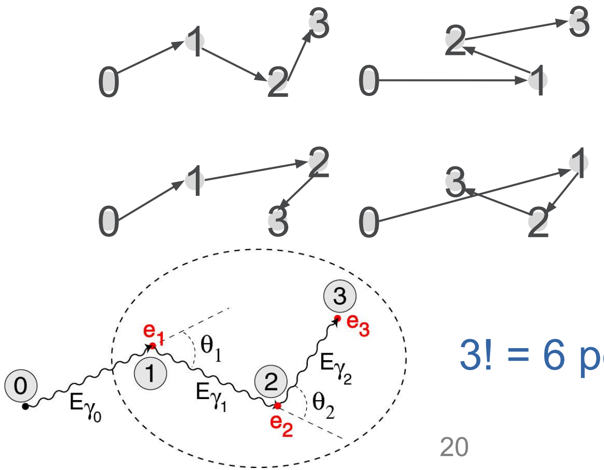
Clear improvement in the energy resolution & efficiency

ML TOOLS FOR GAMMA-RAY TRACKING

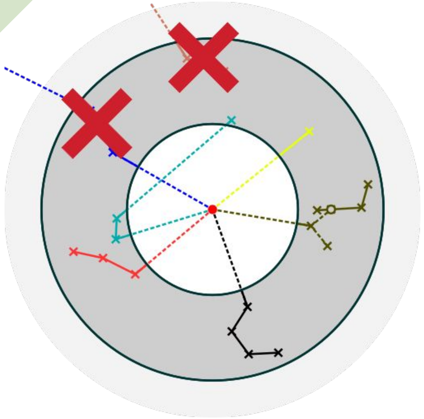
Three complex operations

- Cluster interactions into separate γ-rays
- Order interactions for individual γ-rays
- Suppress γ-rays scattering out of the detector



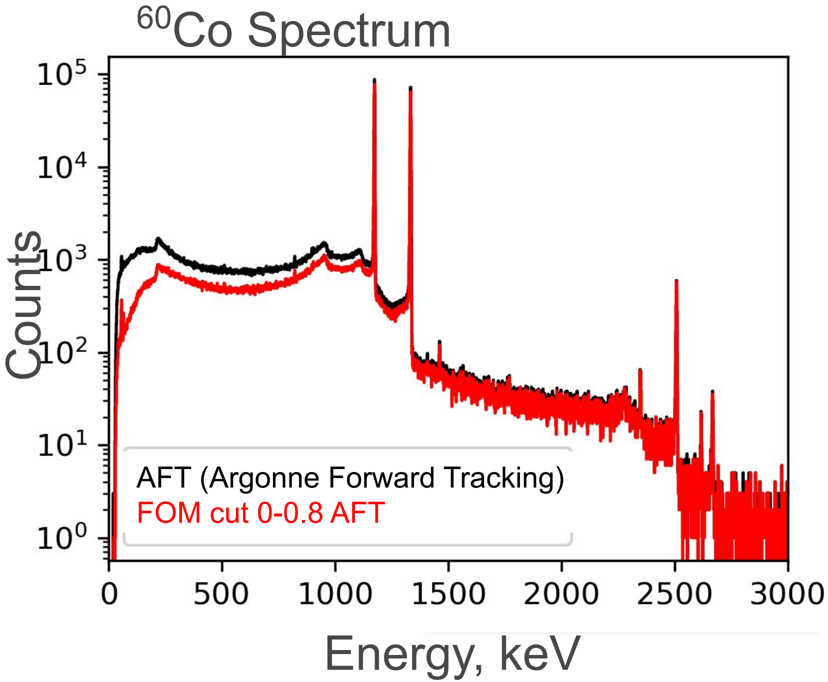


3! = 6 permutations



Supression of ordered clusters

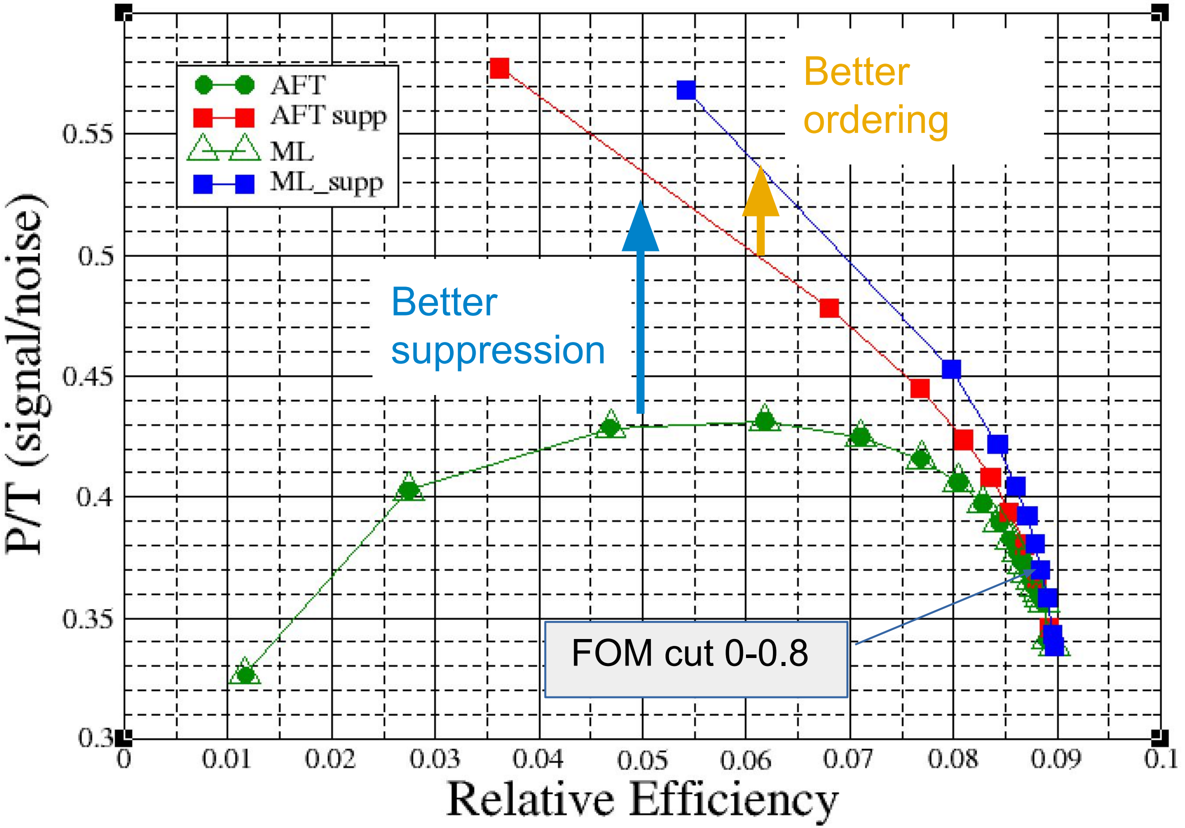
- Final FOM check to remove background from energy spectrum
- ML classification problem:
    - Use linear model to help interpretability, protect against overfitting, help transition to experimental data



Results for ^{60}Co source data

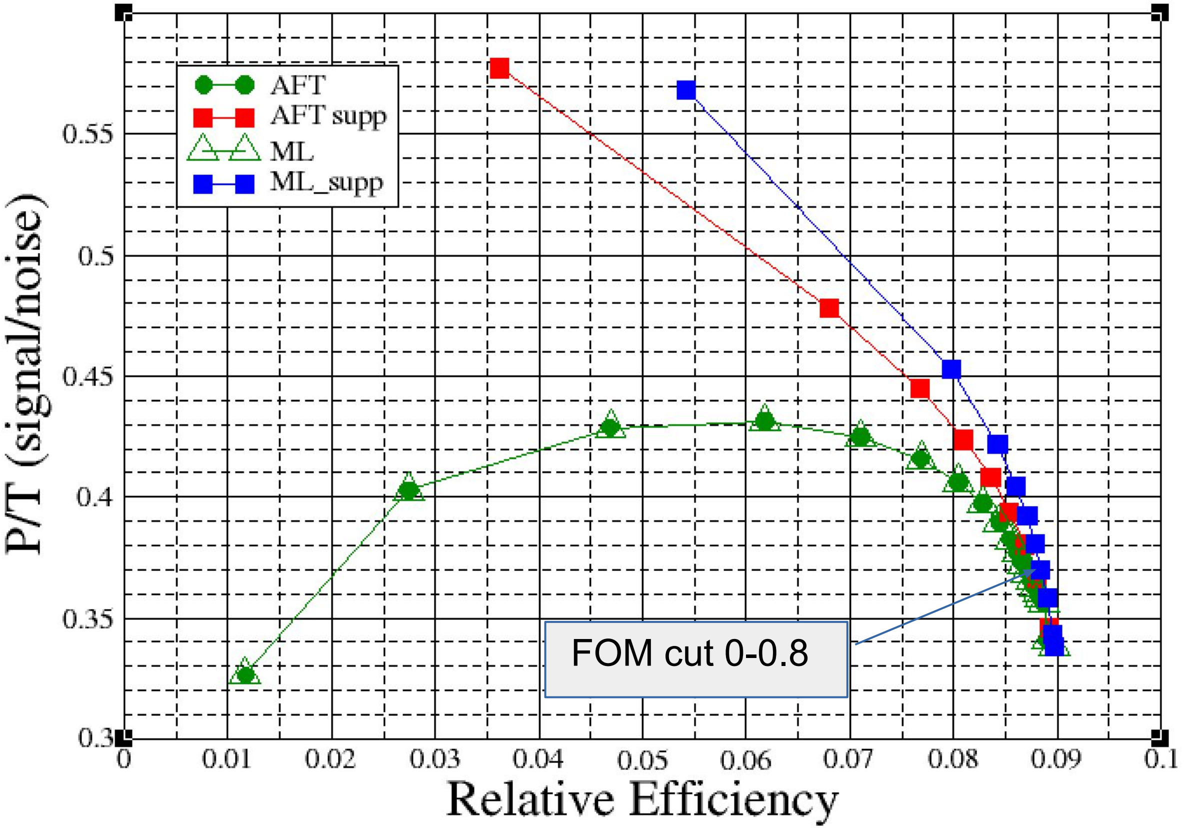
GRETINA (AFT) FOM

$$\frac{1}{N-1}\sum_{i=1}^{N-1}\left(\theta^{\text{geo}} - \theta^{\text{theo}}(E_{i-1}, E_i)\right)^2$$



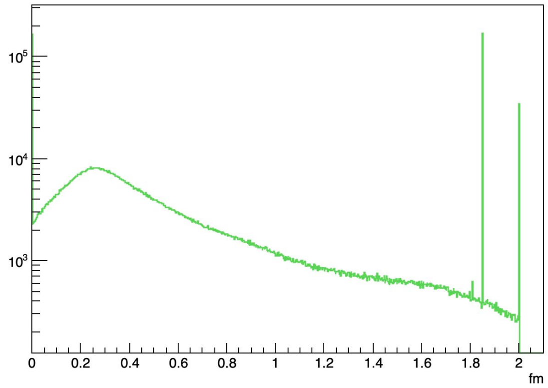
- Good ordering, especially for incomplete gamma-rays, helps clean up the spectrum
- Biggest gain: More complex handling of single interactions
- Single interactions previously had FOM values of 0 or 1.85 (AFT), now continuous

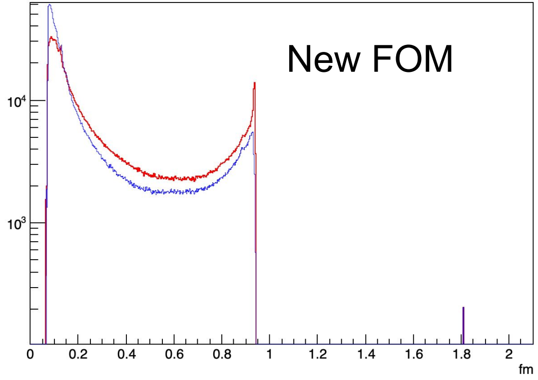
Results for ^{60}Co source data



GRETINA (AFT) FOM

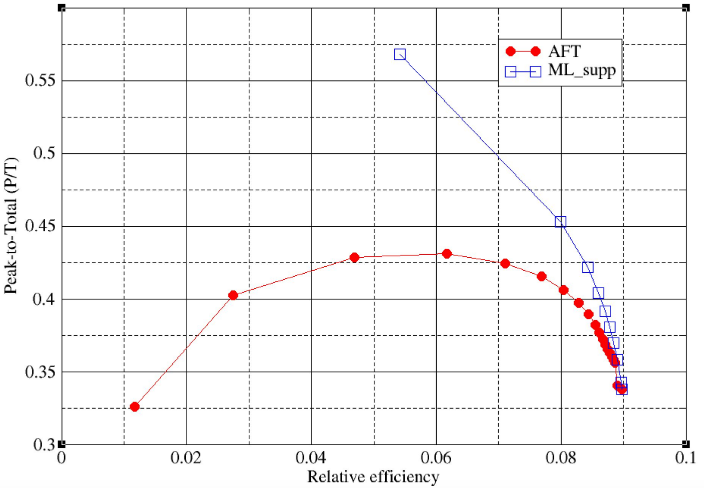
$$\frac{1}{N-1}\sum_{i=1}^{N-1}\left(\theta^{\mathrm{geo}}-\theta^{\mathrm{theo}}(E_{i-1},E_i)\right)^2$$



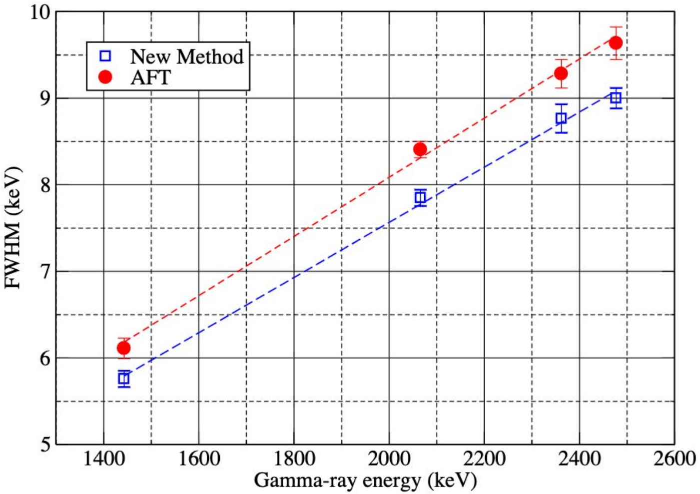


Results summary

P/T improved by ~10 %
Efficiency ~ 6 %



FWHM improved by 9 %



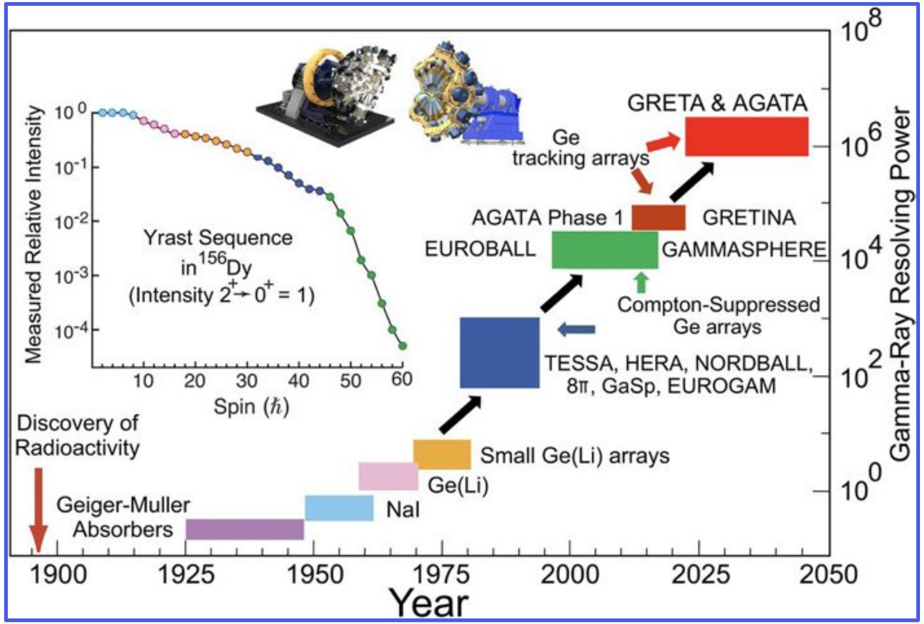
These numbers look small BUT !

FIGURE OF MERIT FOR THE EVALUATION OF A SPECTROMETER PERFORMANCE COMPOSITE PARAMETER WITH:

Total photopeak efficiency ε
Energy resolution FWHM
photopeak-to-total ratio P/T

$$R = \frac{\delta E}{FWHM} \; P/T$$

δE Average spacing between consecutive transitions in a typical cascade



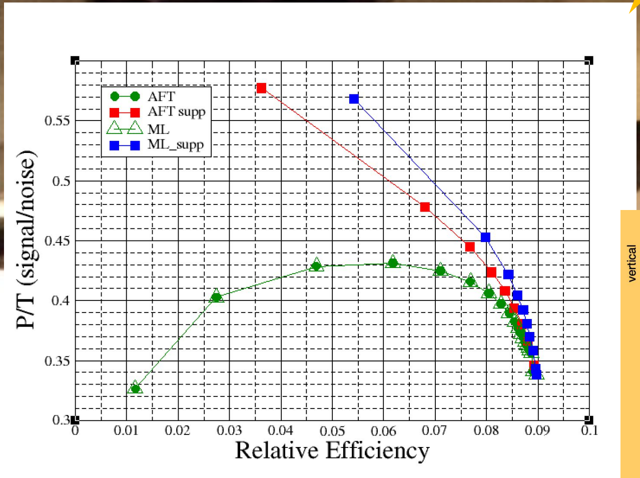
Resolving Power(RP) ~ R^{Fold}

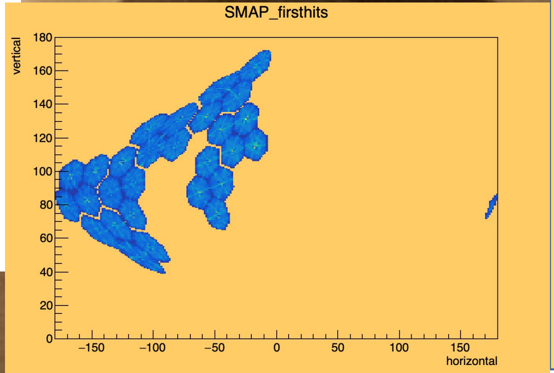
For a 5-fold γ-ray event
(typical for high-spin Gammasphere exp.)

10 %P/T better → increase RP by 60%

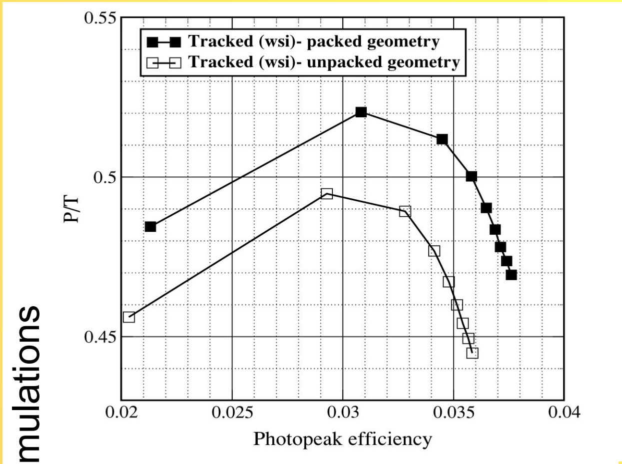
8 % fwhm better → increase RP by 52%

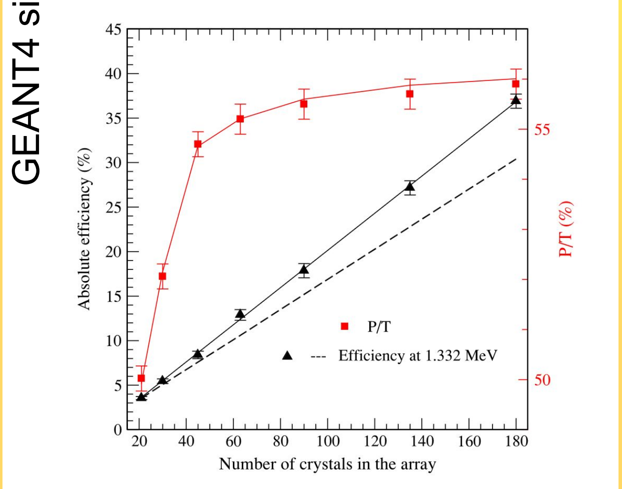
This results in more than a factor 2.5 gain in the Resolving Power





GEANT4 simulations





A more populated array towards GRETA (with new PSA?)will do much better !

ADOPTED METHODOLOGY

GEANT4 Simulated data → Radioactive source data with GRETINA → In-beam GRETINA data

High and low multiplicity data: clusterization, escape suppression
Efficiency and P/T evaluation

High and low recoil velocity: ordering the interactions
1st interaction for Doppler correction
1st and 2nd interactions for Linear polarization

In all cases the results were compared to those obtained using conventional tracking codes AFT (Argonne Forward Tracking) and OFT (Orsay Forward Tracking)

GAMMA-RAY TRACKING SUMMARY

Where ML and data science techniques apply to the problem

Cluster interactions into separate γ-rays

- Angle clustering
- ML clustering
- GNN clustering

Order interactions for individual γ-rays

- Choice of FOM
- Combined clustering/ordering

Suppress γ-rays scattering out of the detector

- Choice of FOM
- ML classification
- Recover γ-ray energies

We designed the ML features to be minimized to prevent overfitting.

While maximizing certain features might be beneficial, simpler and sparser models generally perform better for experimental data. Complex models excel with simulated data but risk overfitting.

To avoid this, we intentionally restricted the ML model's capabilities (sparse, linear models, minimization only)

Project status

- Python Code has been published on GitHub
- New ordering approaches enhance existing techniques, improving the resolving power by up to 2.4 for Doppler-corrected data
- Learning To Rank (LTR) methods enable expanded tracking optimizations
- New suppression approaches further enhance the resolving power and are nearly ready for experiments
- Journal paper manuscript is in preparation

github.com/lynntf/GRETO

Gamma Ray Energy Tracking Optimization

Should be moved off of my personal github

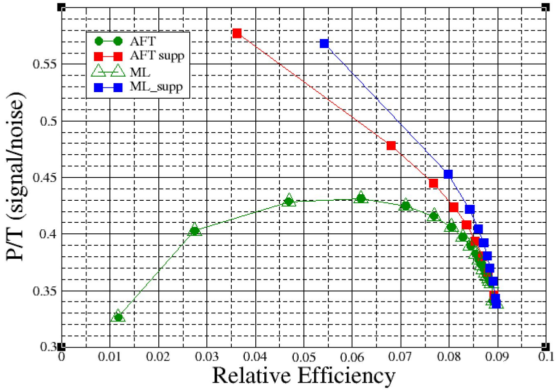
What could make this better

- Alignment of simulated data and PSA output
  - Better alignment means better transfer of ML models from simulation to experiment
  - Better ML models can be applied: RNN, transformers, etc.
- Different training data (GEANT4)
  - Prefer completely unbiased data
- Somehow training with experimental data
  - Pencil beam data?
- Integration with other/new metrics
  - Graph neural network output can be *added* to existing FOM features
- ML models implemented in something faster than python
- We still need better algorithms/optimization to use with new FOM

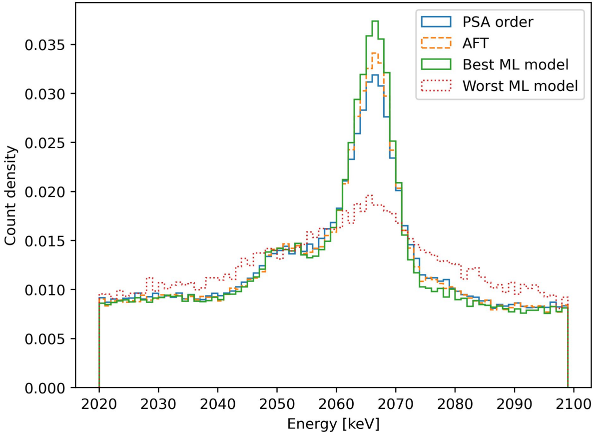
Conclusions

- Transferring ML models from simulated data to experimental data is tricky; easy to overfit
- Previously used FOMs are well motivated but ill suited for ordering in cases where they don’t make sense (incomplete gamma-rays)
- Learning-to-rank allowed us to construct (from physics based objectives) a FOM for ordering that improved Doppler correction, P/T, and efficiency
- Assigning single interactions more descriptive FOM values creates huge improvements in P/T and efficiency

^{60}Co



^{92}Mo - Doppler



Thank you !

About me

Postdoc at Argonne in Math and Computer Science from 2022-2024
Focused on γ-ray tracking and optimization

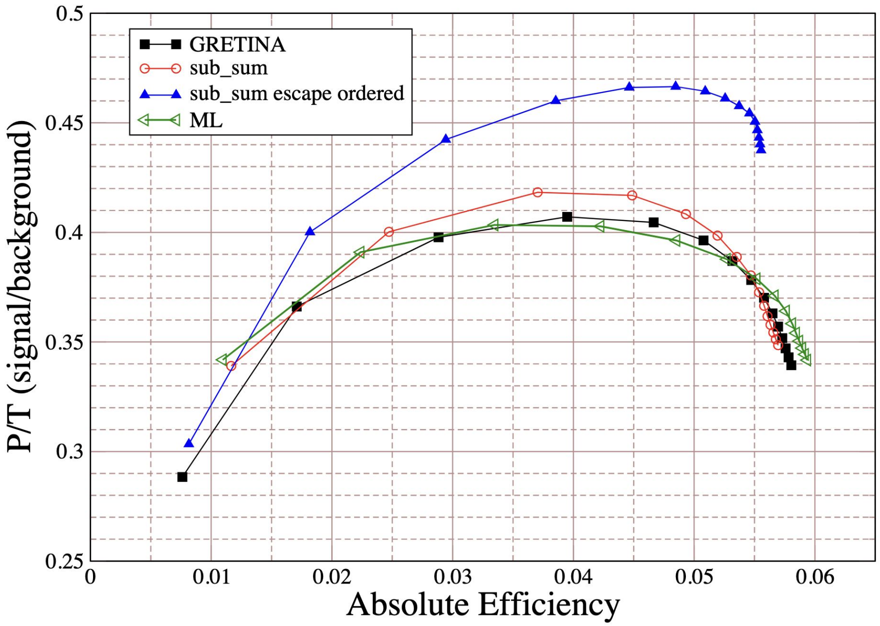
Worked closely with Amel and Torben

Curently at Johns Hopkins University Applied Physics Lab since August 2024

FUTURE WORK AND EXTENSIONS

Improving the resolving power of GRETINA for further analysis

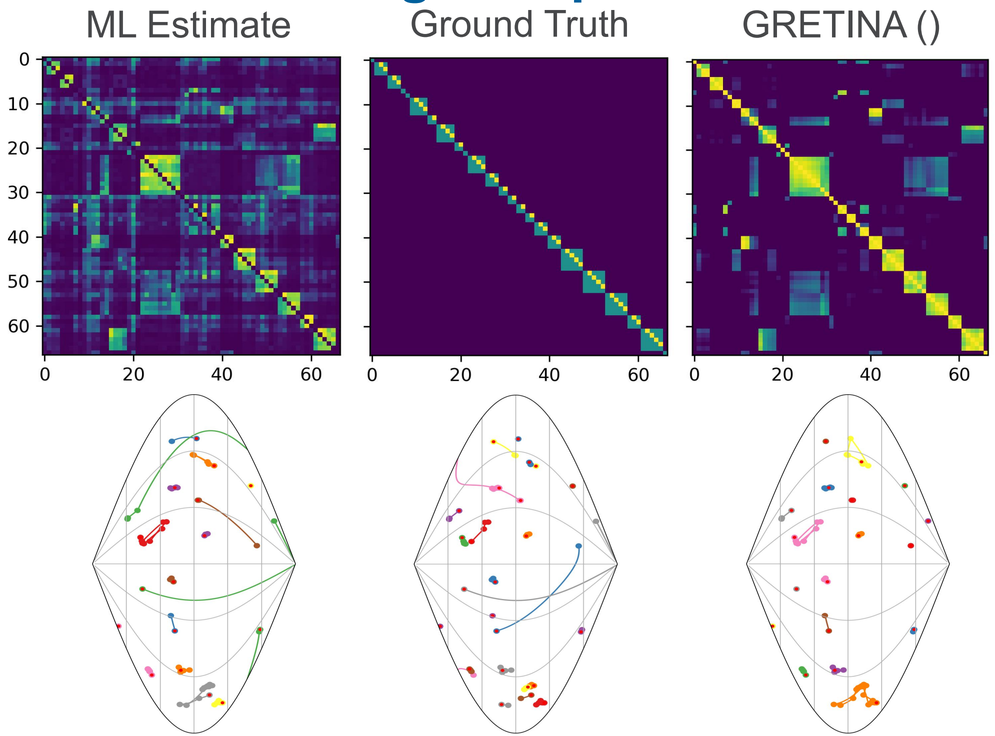
- Improved recovery of escape energies instead of suppression
- ML tools for fast tracking
- ML training using experimental data from sources
- ML tools for on-line learning
- Optimization based approaches for better clustering
- Apply techniques to the problem of pair production



ML CLUSTERING

Clustering beyond GRETINA without knowledge of spectrum

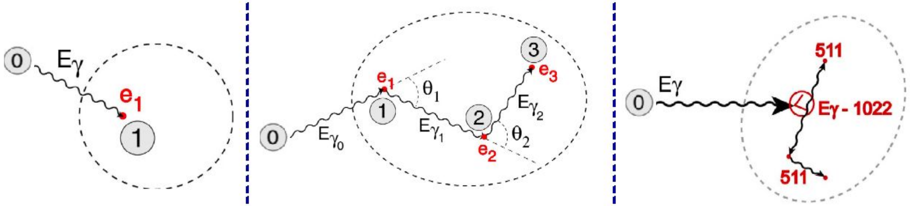
- GRETINA clustering is done spatially with respect to cluster spread (scattering forward)
- Use ML to create an alternate distance metric by which to cluster
  - Learned from data
  - Include additional clustering steps beyond singles
  - Include cluster order



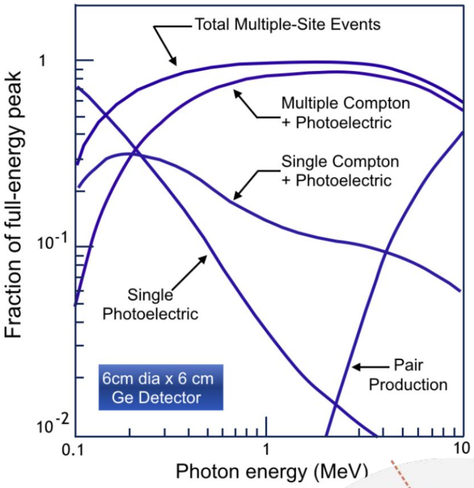
γ-ray Interaction Data

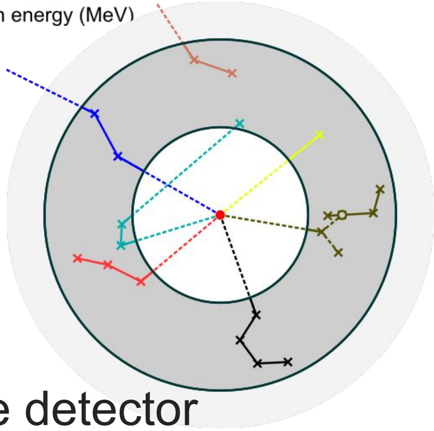
Overview of the principle

~ 100 keV	~1 MeV	~10 MeV γ-ray energy
Photoelectric	**Compton Scattering**	**Pair Production**



Isolated hits	Angle/Energy	Pattern of hits
Probability of Interaction depth	$E'_{\gamma} = \dfrac{E_{\gamma}}{1 + \frac{E_{\gamma}}{m_0 c^2}(1 - cos\theta)}$	$E_{1st} = E_{\gamma} - 2m_0c^2$





Challenges:

- γ-rays too close
- γ-rays escape
- γ-rays crossing the detector
- Suppress environmental γ-rays

RECREATING COMPTON SUPPRESSION

Correctly ordering escaped γ-rays improves suppression

- Previously done with BGO absorber
- FOM correctly orders < 50% of escapes
  - Wrong order favorable over truth
  - Suppression suffers
- Using escape energy estimate improves suppression (Tashenov & Gerl 2010)
  - Order for escapes is essential for suppression
- ML can further improve ordering & suppression