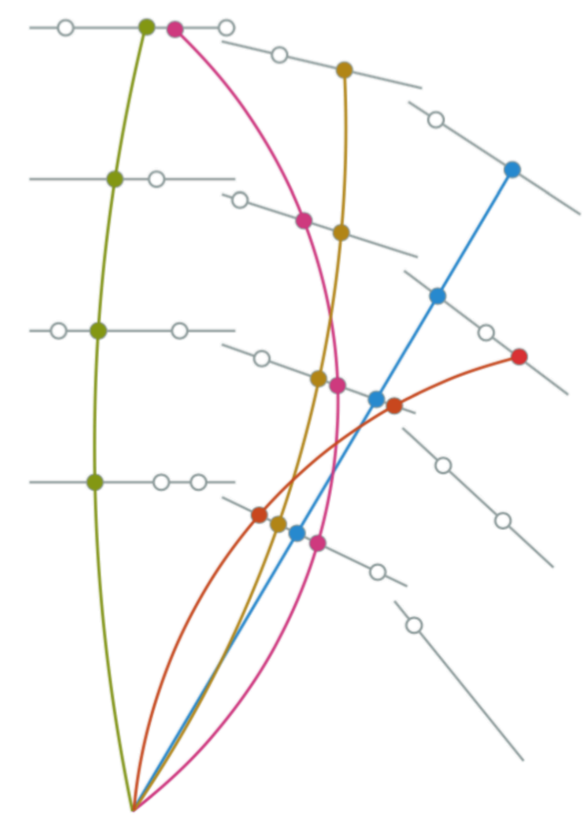
Accelerating Graph-Based Tracking with Symbolic Regression

F. di Bello, E. Gross, C. Schiavi, **N. Soybelman**

IML Workshop, 30.01.2024

Motivation

- Tracking information is fundamental for triggers
- New ML methods for tracking are promising, but slow
- Quantization and implementations on FPGAs complicated, especially for advanced network structures like GNNs
- The lower the execution time of our neural nets, the higher rates we can handle
- How can we accelerate them?



Symbolic Regression on FPGAs for Fast Machine Learning Inference

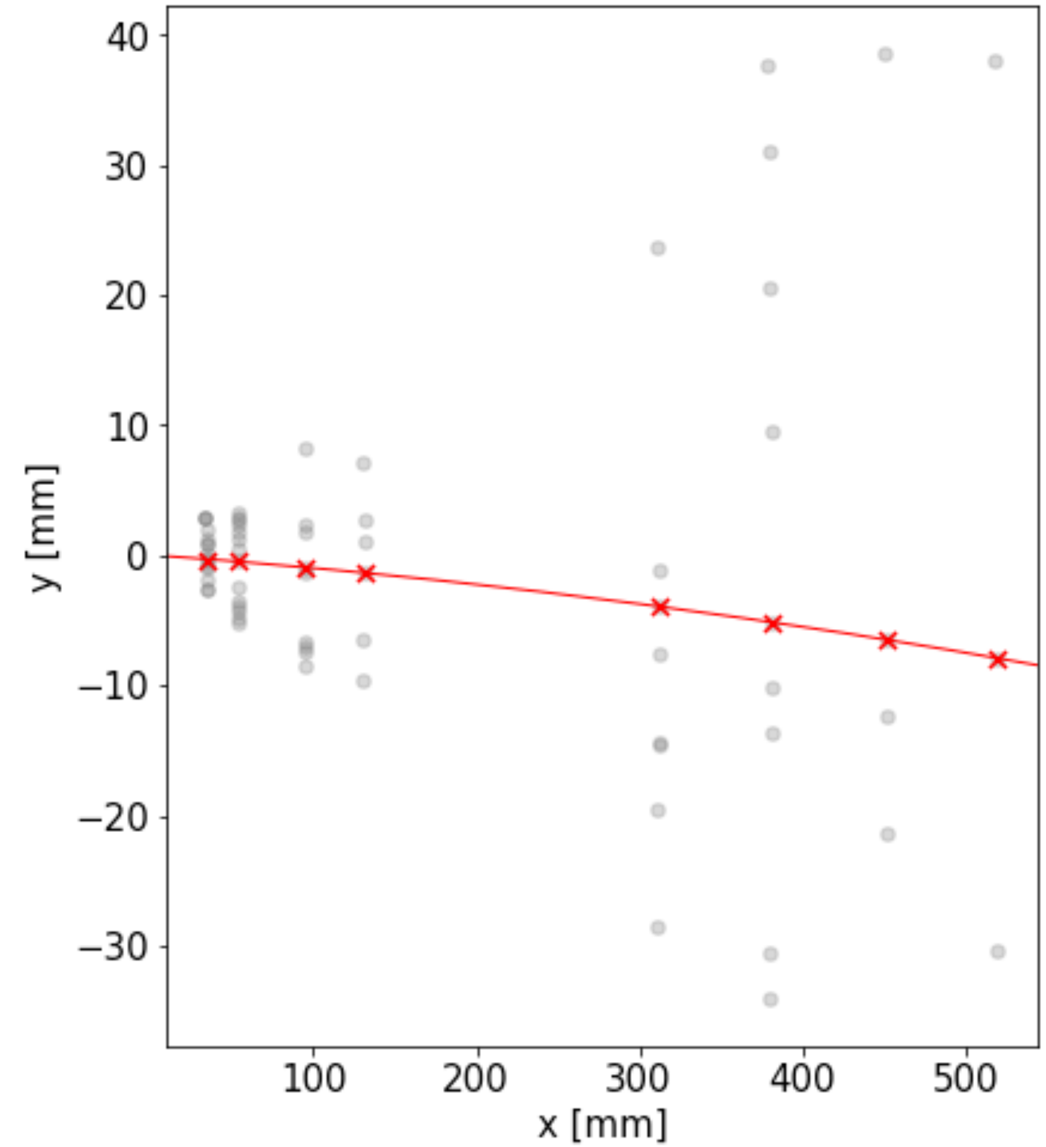
arXiv: 2305.04099

Ho Fung Tsoi[1*], *Adrian Alan* Pol[2], *Vladimir* Loncar[3,4], *Ekaterina* Govorkova[3], *Miles* Cranmer[2,5], *Sridhara* Dasu[1], *Peter* Elmer[2], *Philip* Harris[3], *Isobel* Ojalvo[2], and *Maurizio* Pierini[6]

- Approximate MLP with symbolic expression
- Used for jet classification
- Easy to implement on FPGA and fast inference

Can be generalized to more complex problems

Dataset



- Single track events $p_T > 20$ GeV
- Pile-up $\mu = 25$ (LHC Run 3)
- Simplified cylindrical detector (barrel)
- Detector radii matching ATLAS ID
- No r-smearing
- ϕ-, z-smearing based on ATLAS Pixel and SCT
- Preselect hits in $0.1 \times 0.1 \; \eta - \phi$ wedge, ± 5 mm around PV, fully contains track

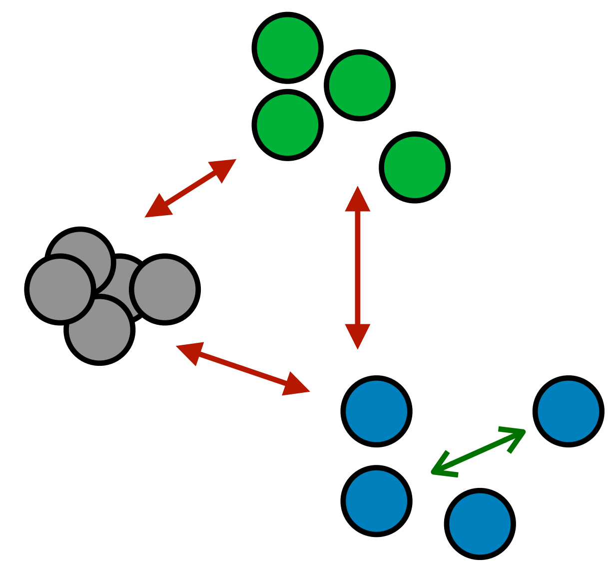
Modified Object Condensation

Object condensation: one-stage grid-free multi-object reconstruction in physics detectors, graph, and image data

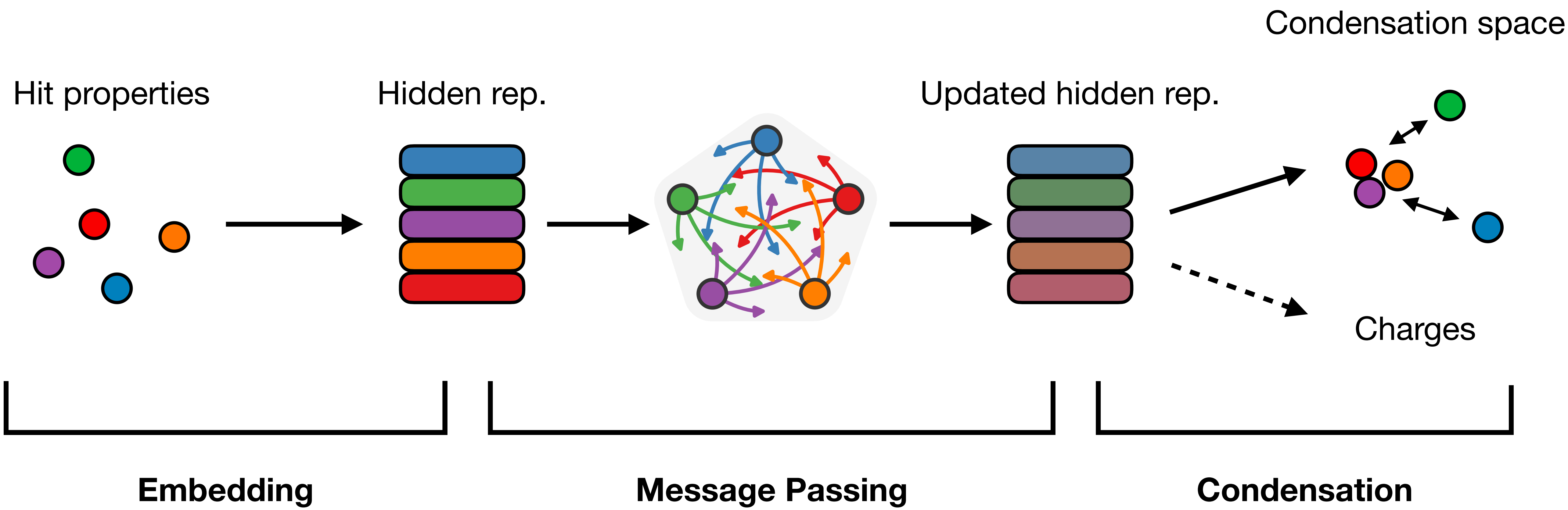
Jan Kieseler[1]
(jan.kieseler@cern.ch)

arXiv: 2002.03605

- Learn condensation space
- Hits of the same track are mapped in one cluster
- Predict charges for potential — BCE loss signal vs. bkg.
- Attractive potential: Pull hits from same track together
- Repulsive potential: Push hits not from signal track away
- Condense noise: Push all noise/pileup hits to origin



GNN with Symbolic Regression

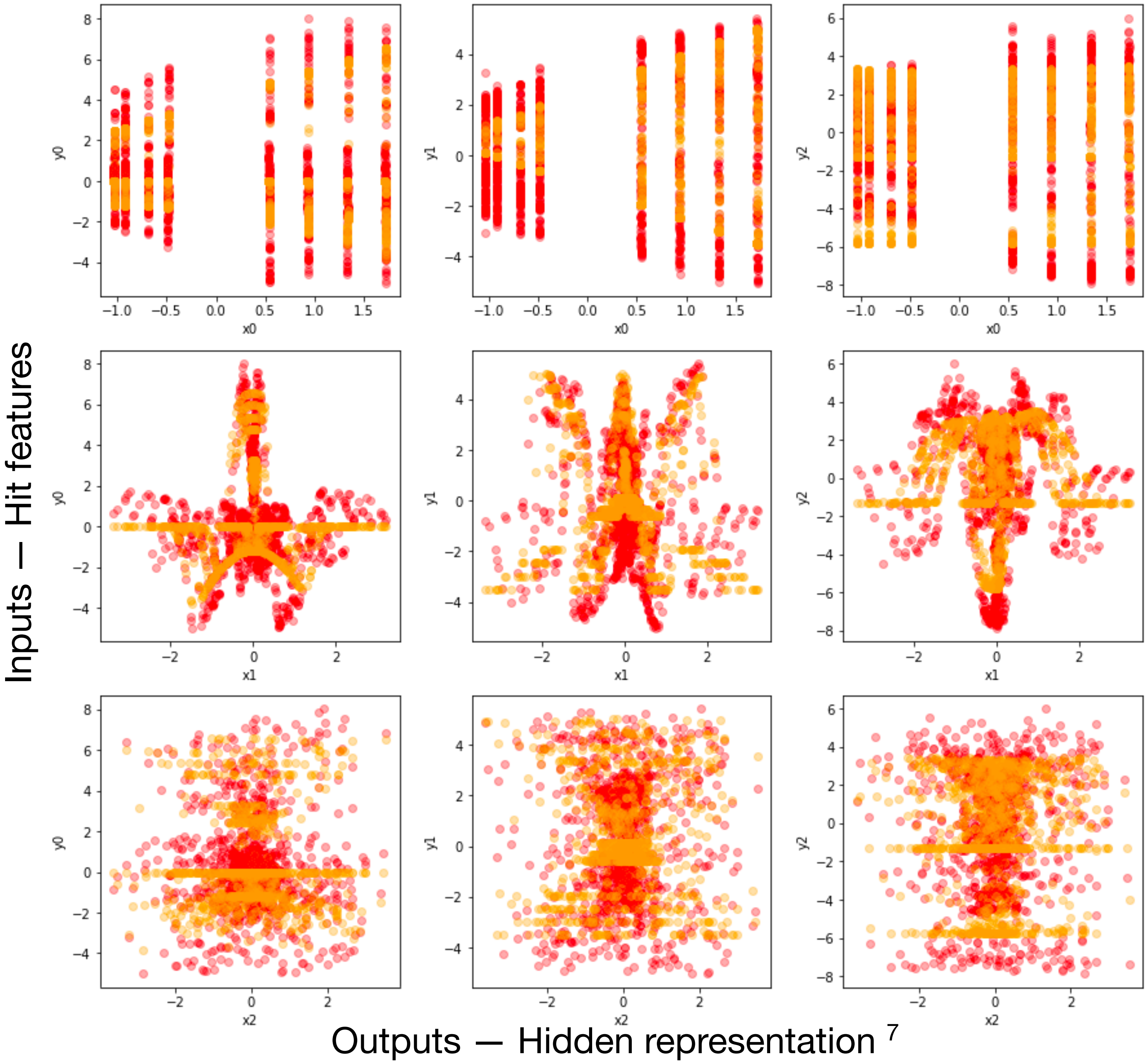


- In total we have 3 MLP to be replaced with SR
- Preserving graph structure
- After each replacement retrain rest of the network
- Small network, hidden rep. and cond. space have dimension 5

Symbolic Regression

Using PySR package

Look inside first MLP — example



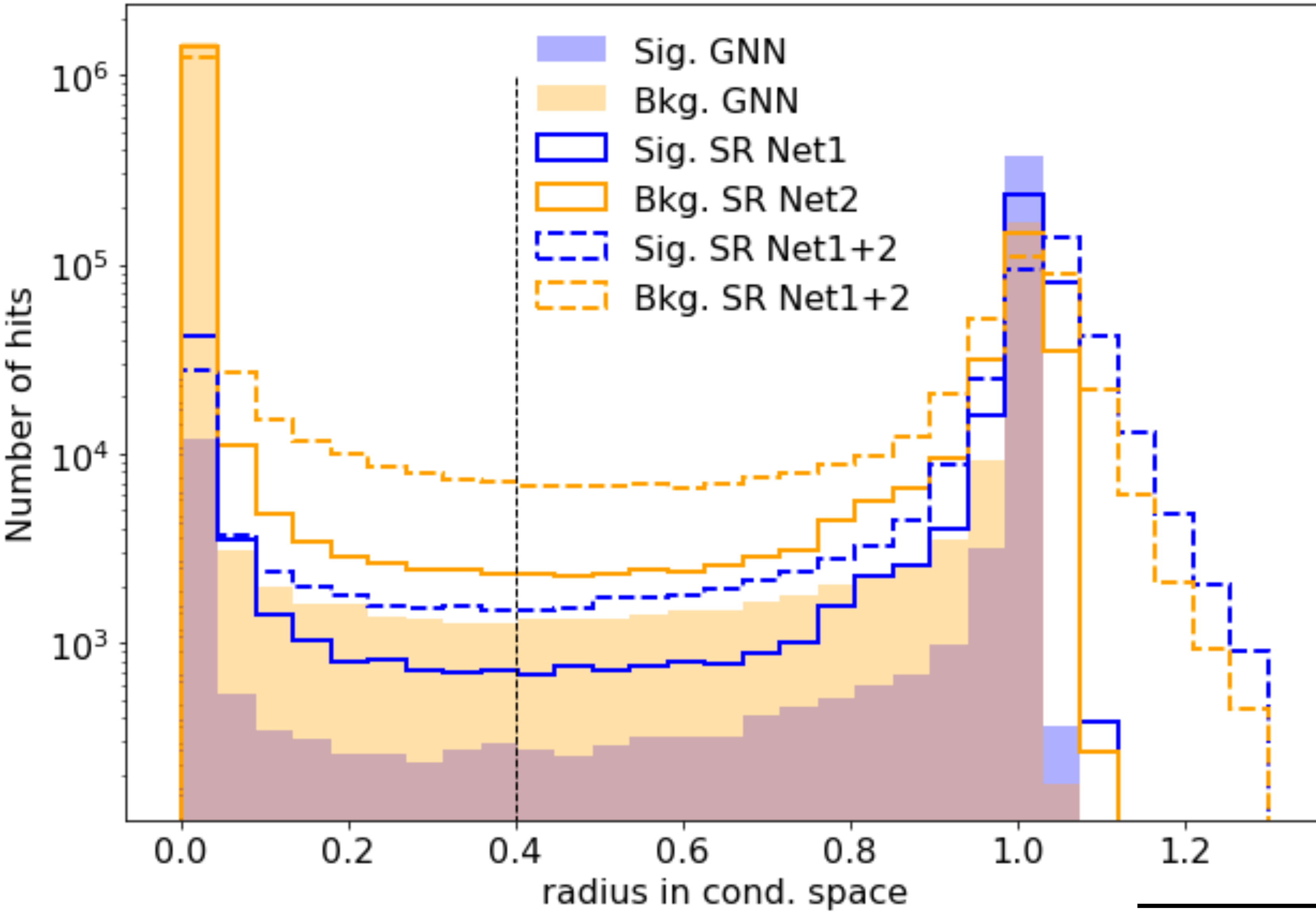
NN target

SR prediction

- SR learns main structures
- Some performance loss expected

N. Soybelman — Weizmann Institute

Hit efficiency — Cond. space

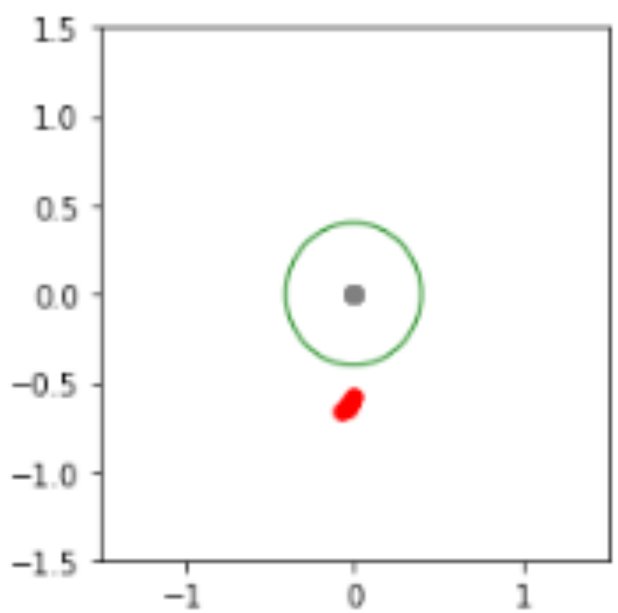


- Compare results of:
  - baseline GNN
  - 1st MLP with SR
  - 1st + 2nd with SR

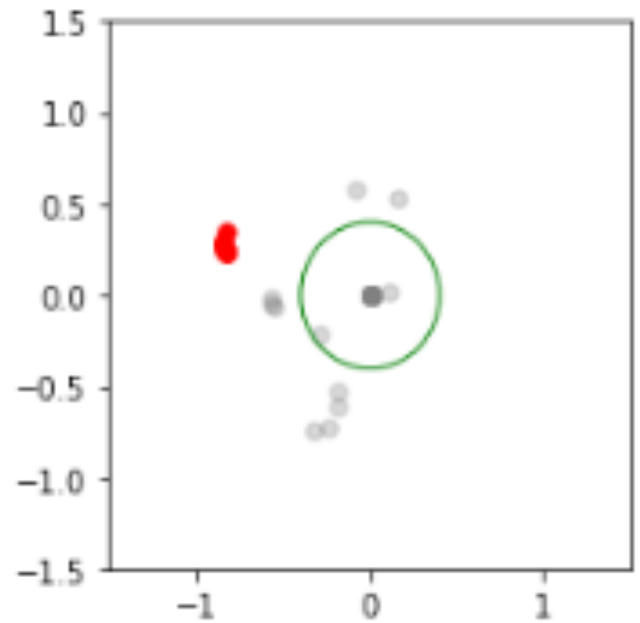
- Keep all hits with $r > 0.4$

	GNN	SR 1	SR 1+2
Sig. efficiency	96.4%	87.0%	89.1%
Bkg. rejection	88.3%	85.0%	77.4%

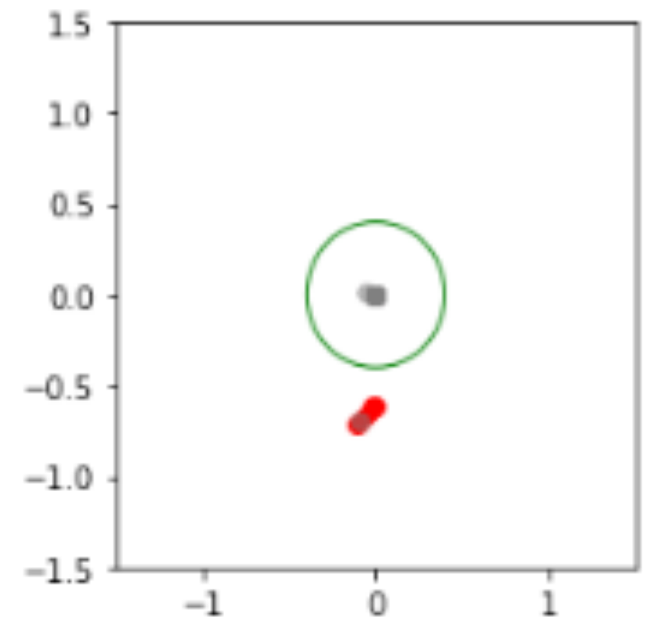
Cluster selection



Ideal case



Needs clustering

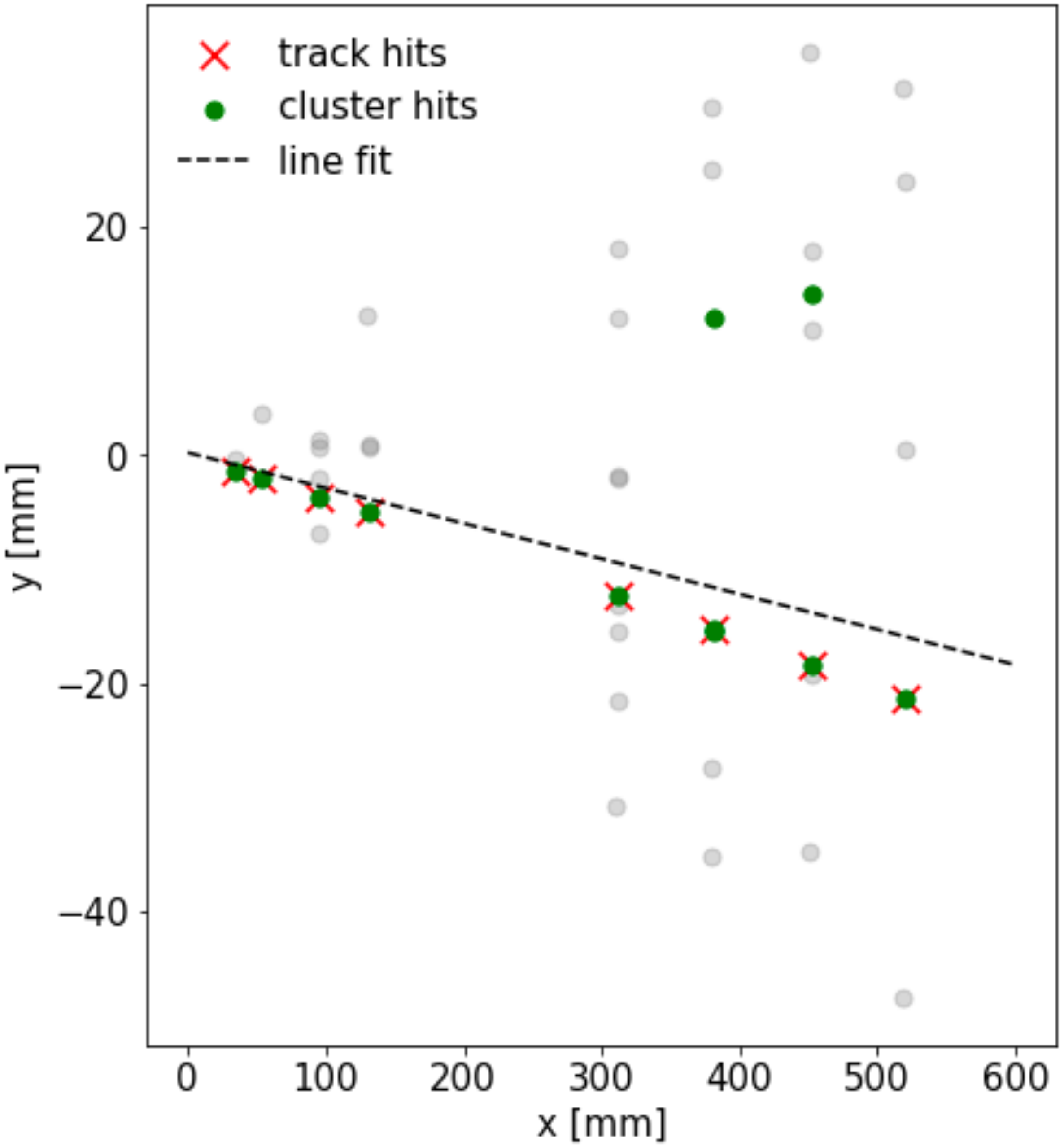


Contamination

- MeanShift Clustering on points with hits with $r > 0.4$
- Keep largest cluster

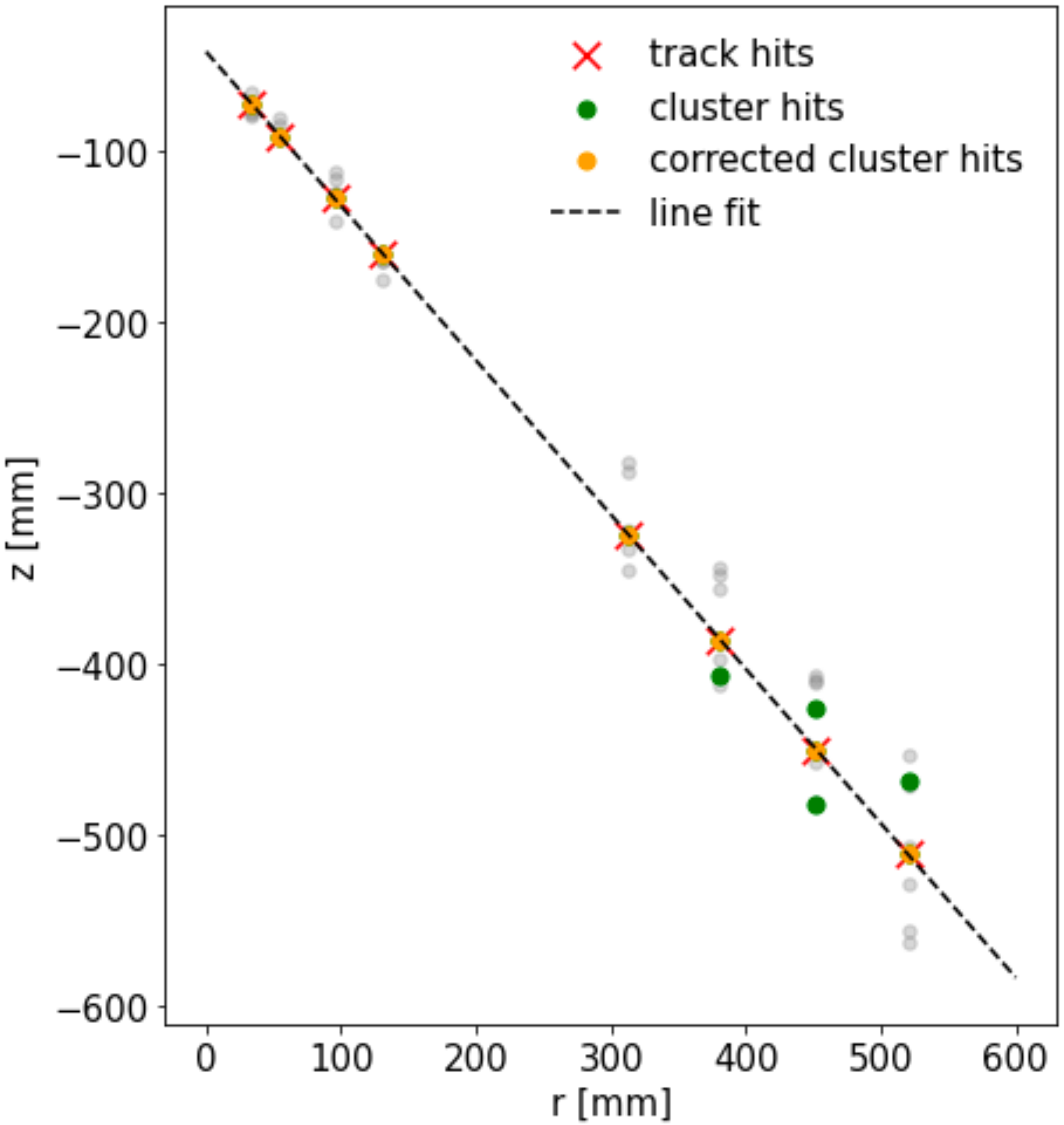
	GNN	SR 1	SR 1+2
Sig. efficiency	91.7%	79.5%	76.3%
Bkg. rejection	95.7%	96.2%	90.0%

Track fitting



1. **Approximate line fit in x-y plane**
   Remove extreme outliers that ruin circle fit

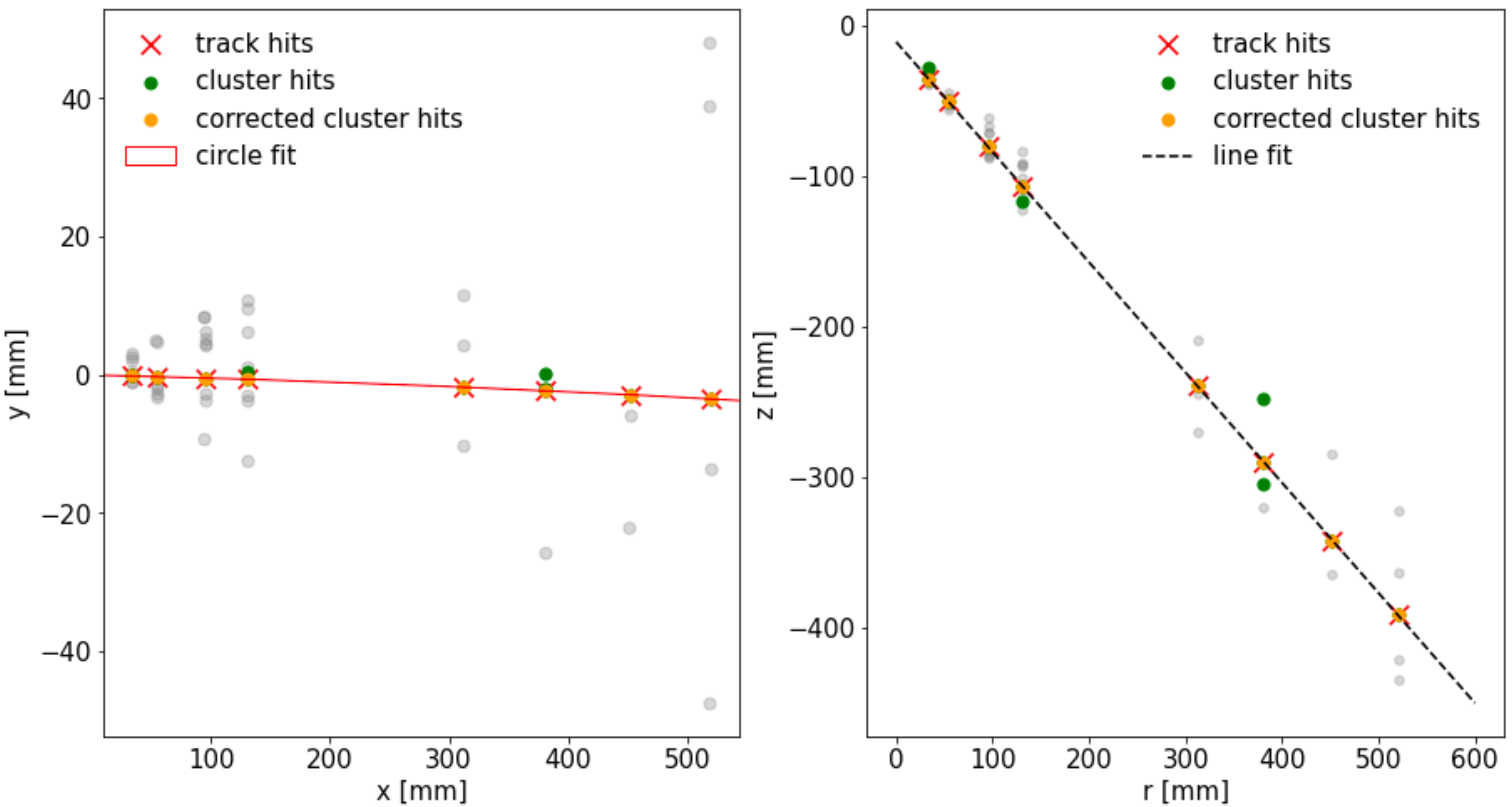
Track fitting



2. **Line fit in r-z plane**
   - Physically there is only one hit per layer
   - Fit line, remove point with highest χ^2
   - Repeat, until 1 hit per layer left

Track fitting

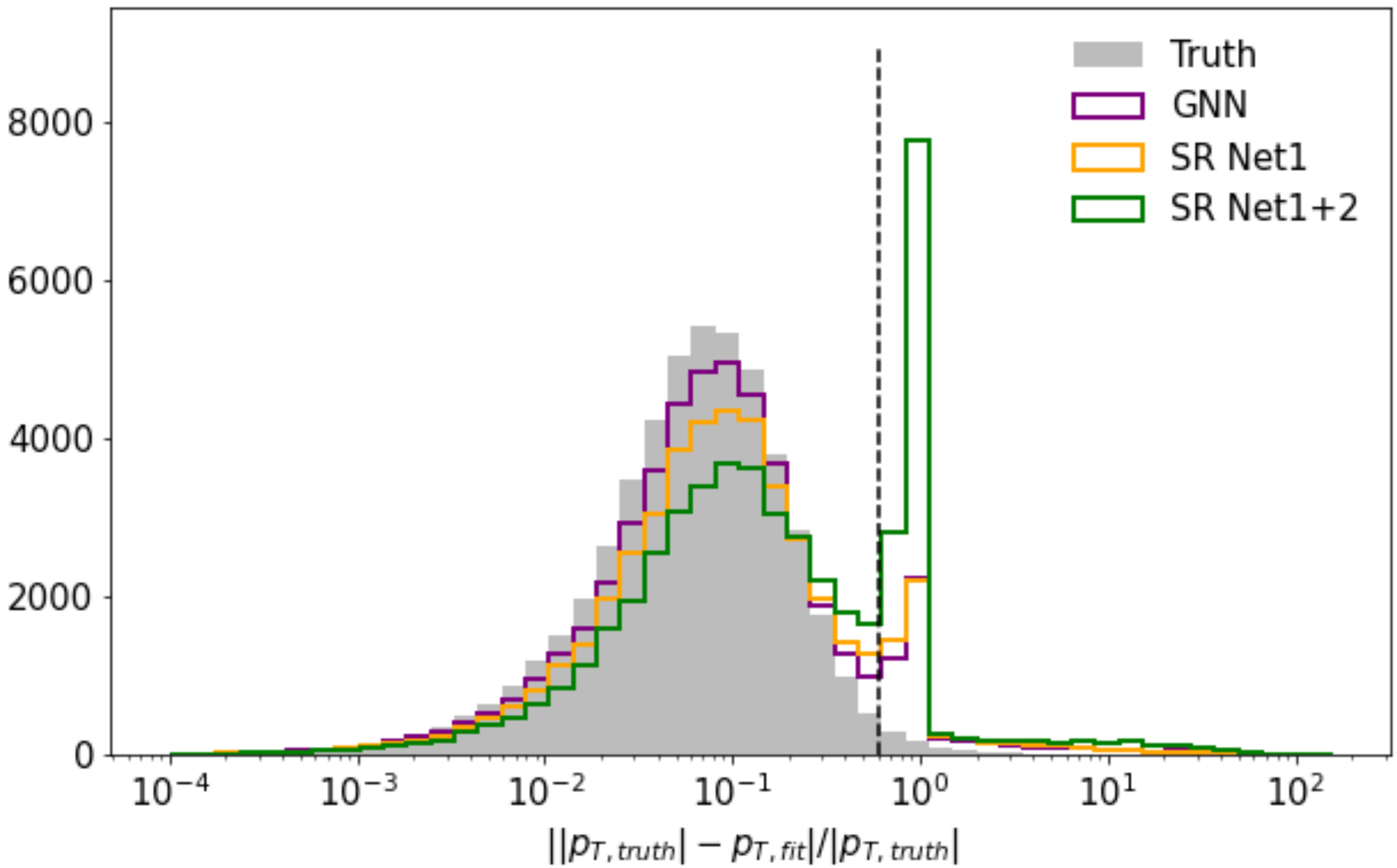
3. Circle fit in x-y with remaining hits



N. Soybelman — Weizmann Institute

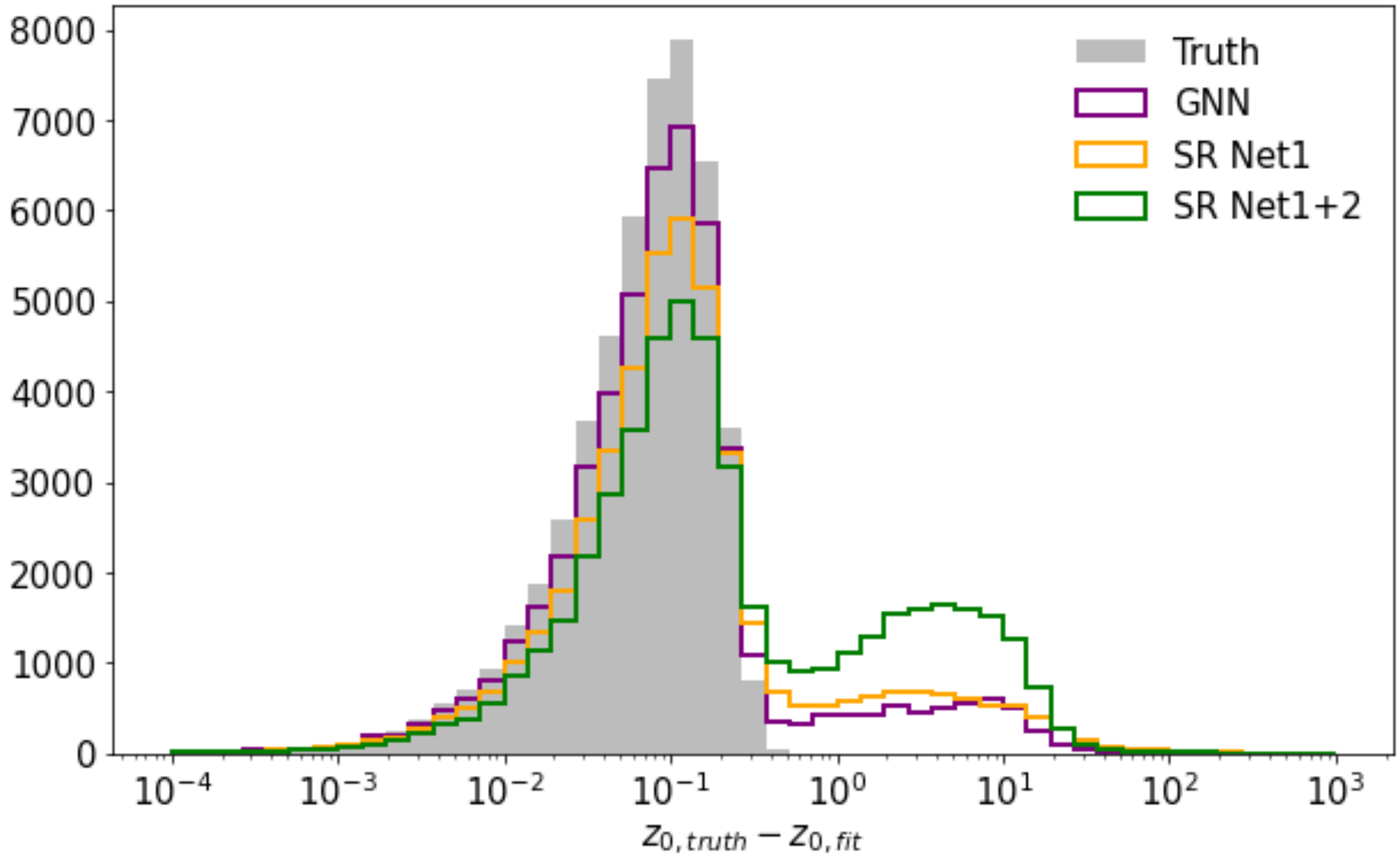
Track efficiency

Residual of track pT fit



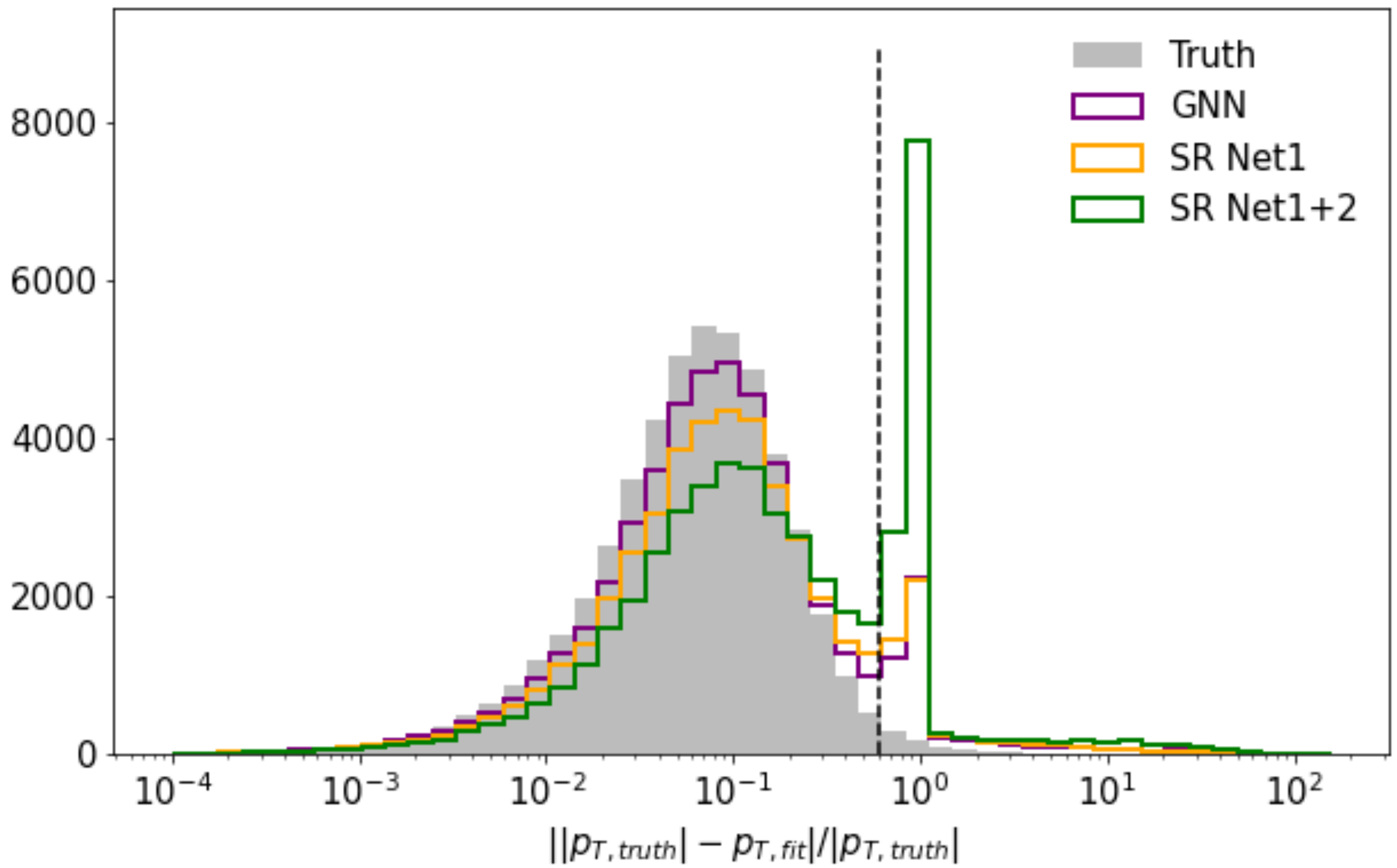
Track efficiency

Residual of z0 fit



N. Soybelman — Weizmann Institute

Track efficiency — summary



Very preliminary, not optimized

Some fits fail because …

- < 3 points remain after clustering
- < 3 points remain after correcting

Fit fail	GNN	SR 1	SR 1+2
Clustering	2.1%	7.2%	5.5%
Correction	4.9%	4.5%	1.4%
pT res > 0.6	9.6%	10.5%	25.9%
Total	**16.6%**	**22.2%**	**32.8%**

Timing studies

	Timing	CPU load
GNN	105 μs	80%
Full SR	45 μs	10%

- Time difference will be more significant with bigger GNN
- Reduction in CPU load ⟶ more processes in parallel

N. Soybelman — Weizmann Institute

Summary & Outlook

- Simple GNN for tracking with object condensation
- Preliminary work in partially replacing GNN with SR
- Further studies to optimize GNN, clustering, fitting, …
- Can be used for high level trigger HLT
- Possible candidate for L1 trigger if implemented on FPGA