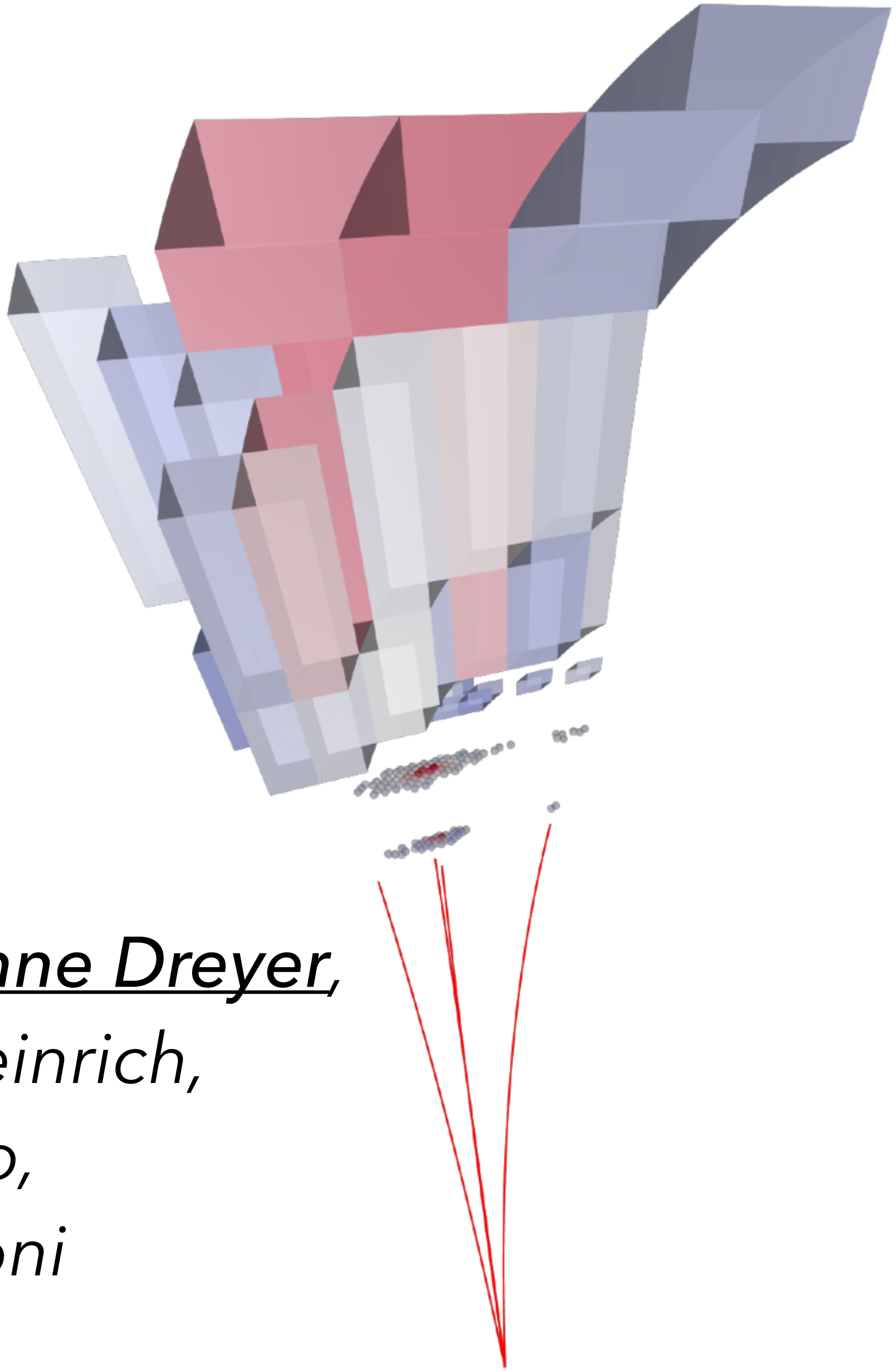
מכון ויצמן למדע
WEIZMANN INSTITUTE OF SCIENCE

MORTIMER B.
ZUCKERMAN
STEM LEADERSHIP
PROGRAM

Reconstructing particles in jets with set transformer and hypergraph models



ML4Jets 2022
Rutgers University

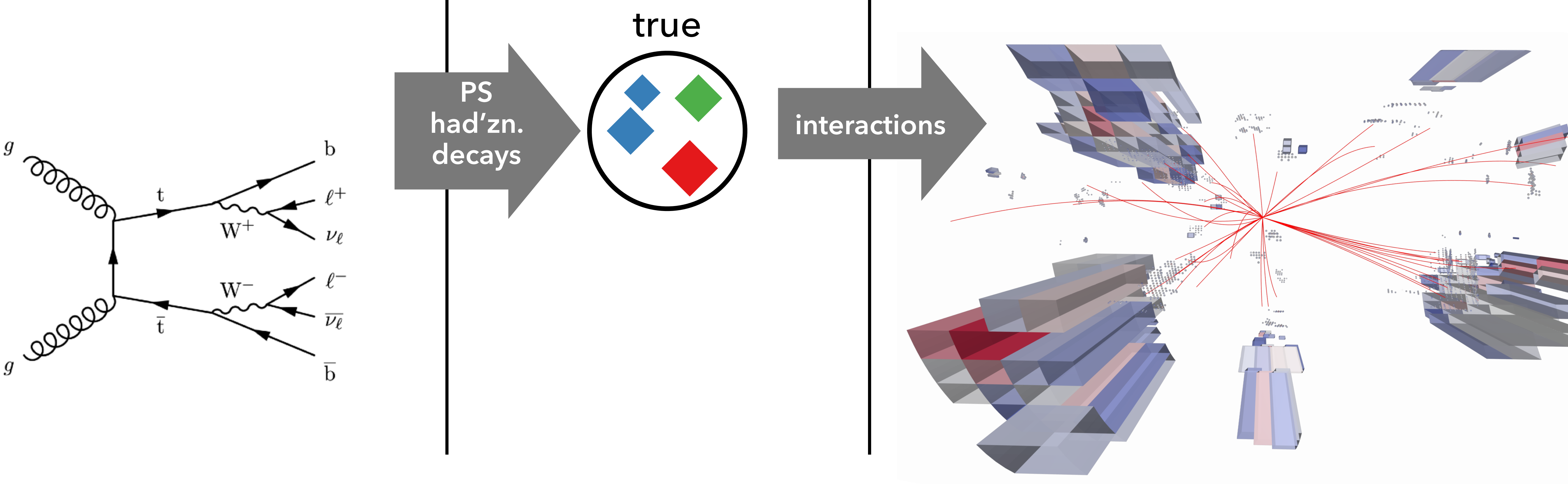
*F. Armando Di Bello, Etienne Dreyer,
S. Ganguly, E. Gross, L. Heinrich,
A. Ivina, N. Kakati, M. Kado,
L. Santi, J. Shlomi, M. Tusoni*

Particle reconstruction

Hard scatter process

Final state particles

Energy deposits in detector

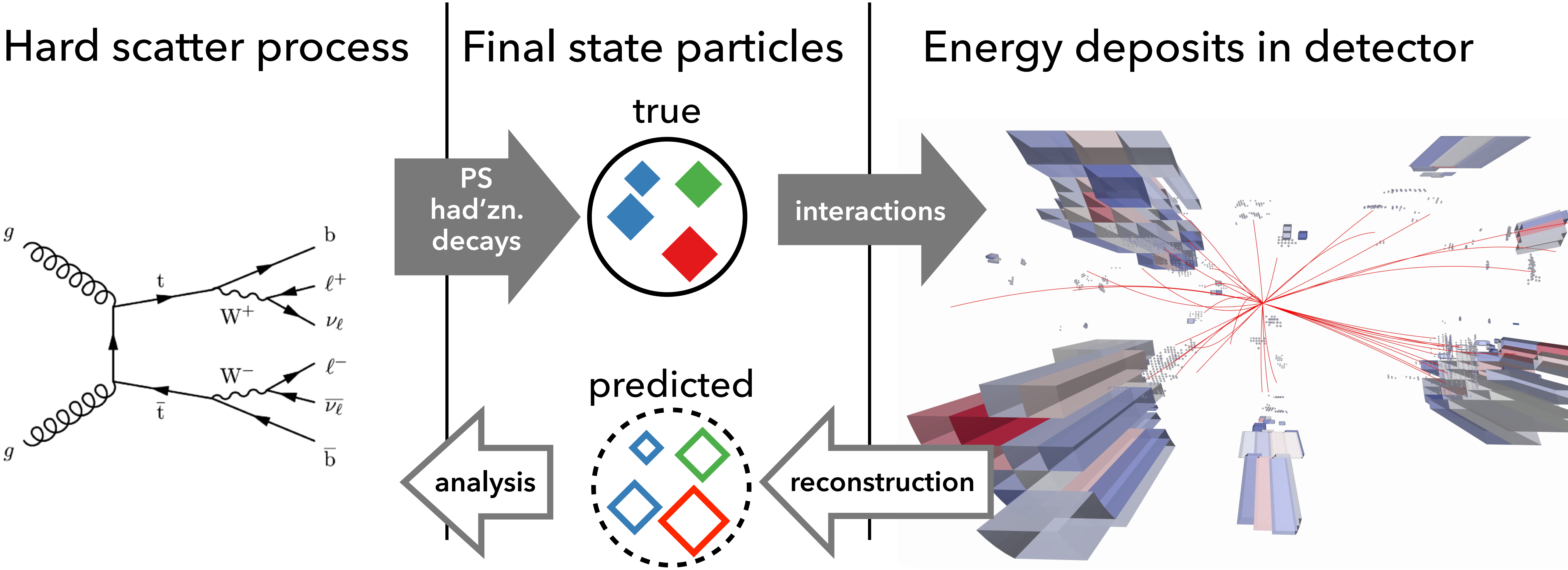


PS had'zn. decays

true

interactions

Particle reconstruction



Hard scatter process | Final state particles | Energy deposits in detector

PS had'zn. decays

true

interactions

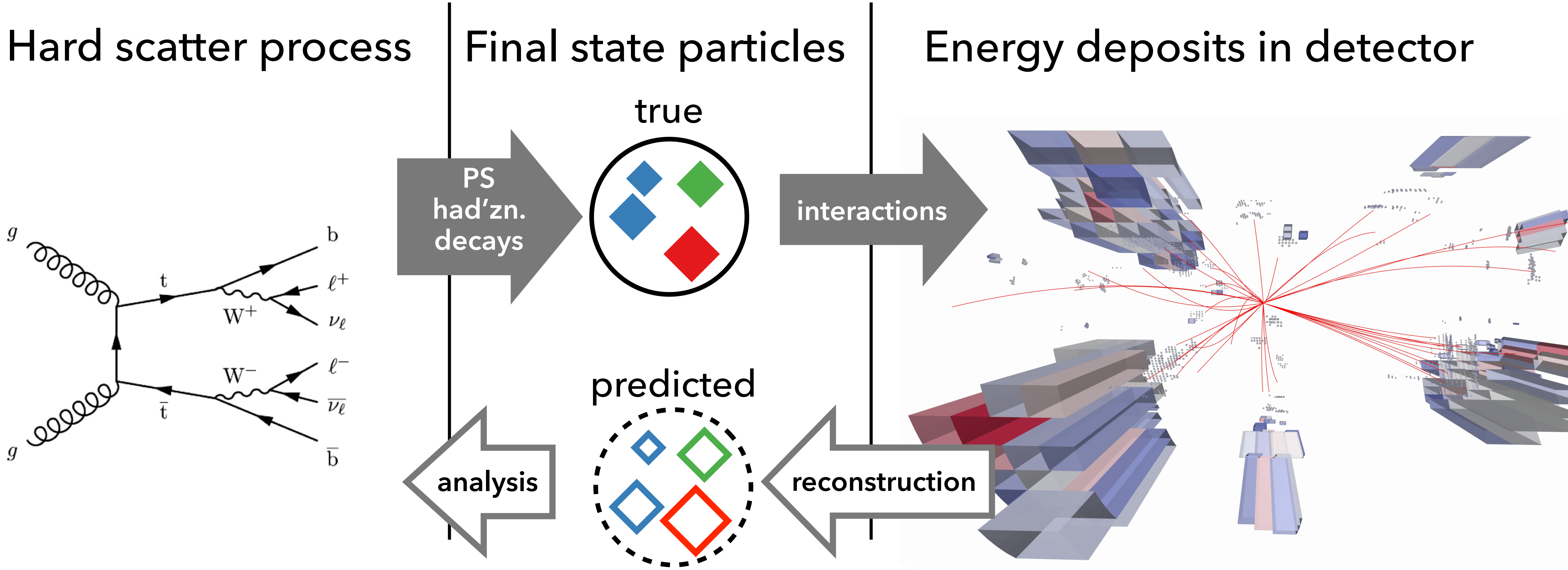
predicted

analysis

reconstruction

Infer the **set** of particles which produced
the **set** of energy deposits in detector

Particle reconstruction



Hard scatter process

Final state particles

Energy deposits in detector

PS had'zn. decays

true

interactions

predicted

reconstruction

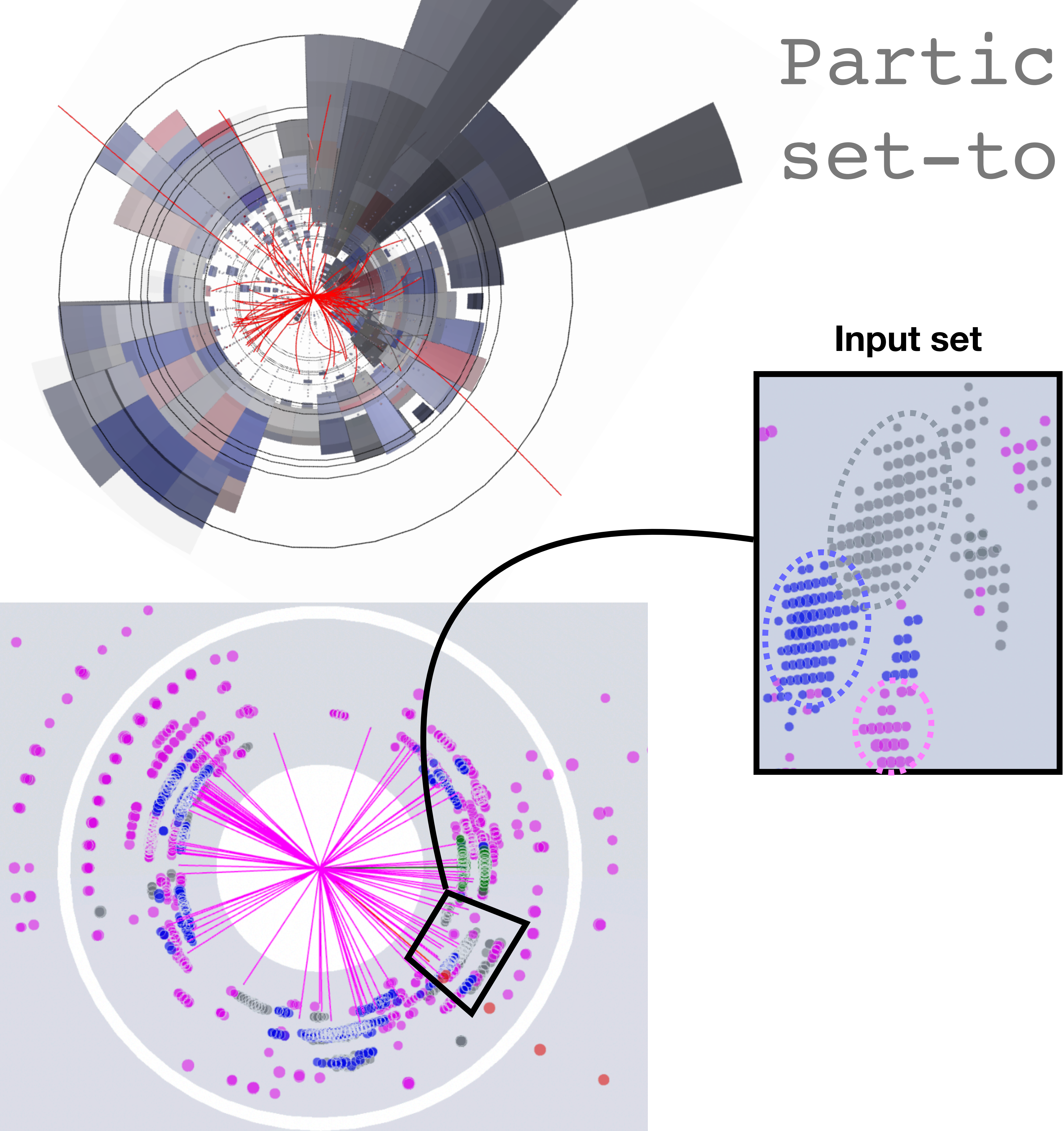
analysis

Infer the **set** of particles which produced the **set** of energy deposits in detector

Challenges:

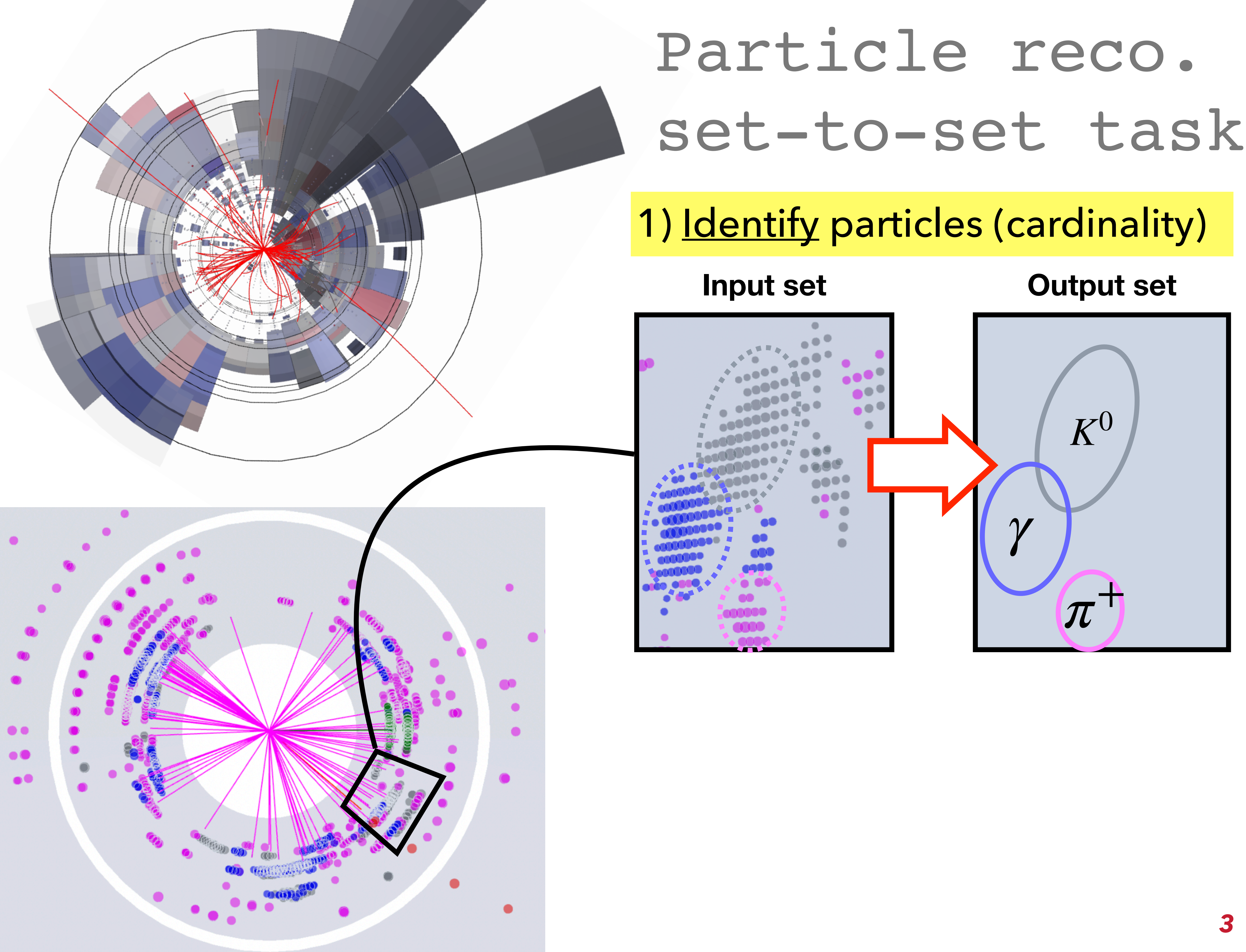
- Physical overlap (due to collimated particles and pileup)
- Feature overlap between different particle signatures
- Dimensionality of data and complexity of spatial correlations

Particle reco.
set-to-set task

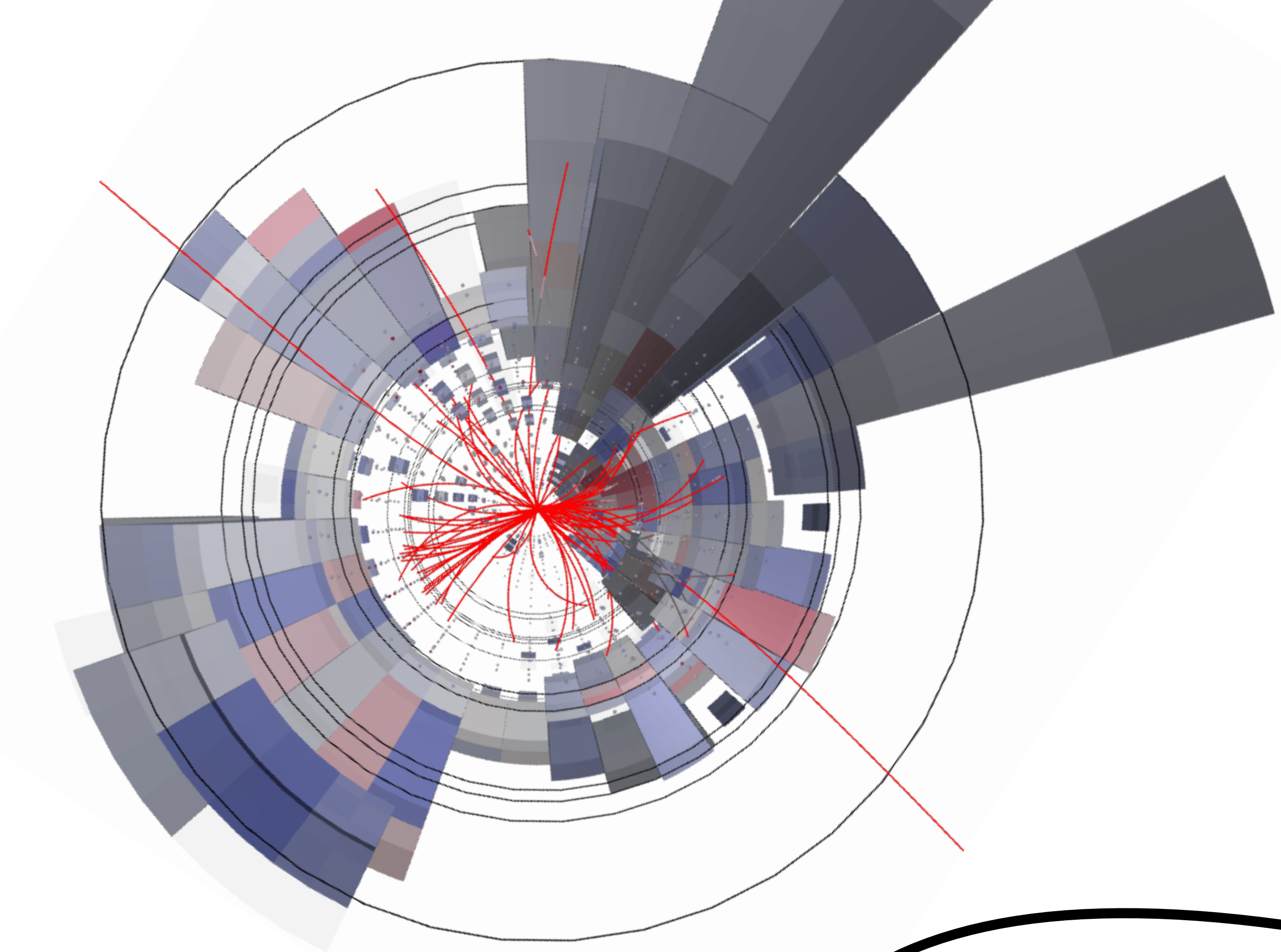


Particle reco. set-to-set task

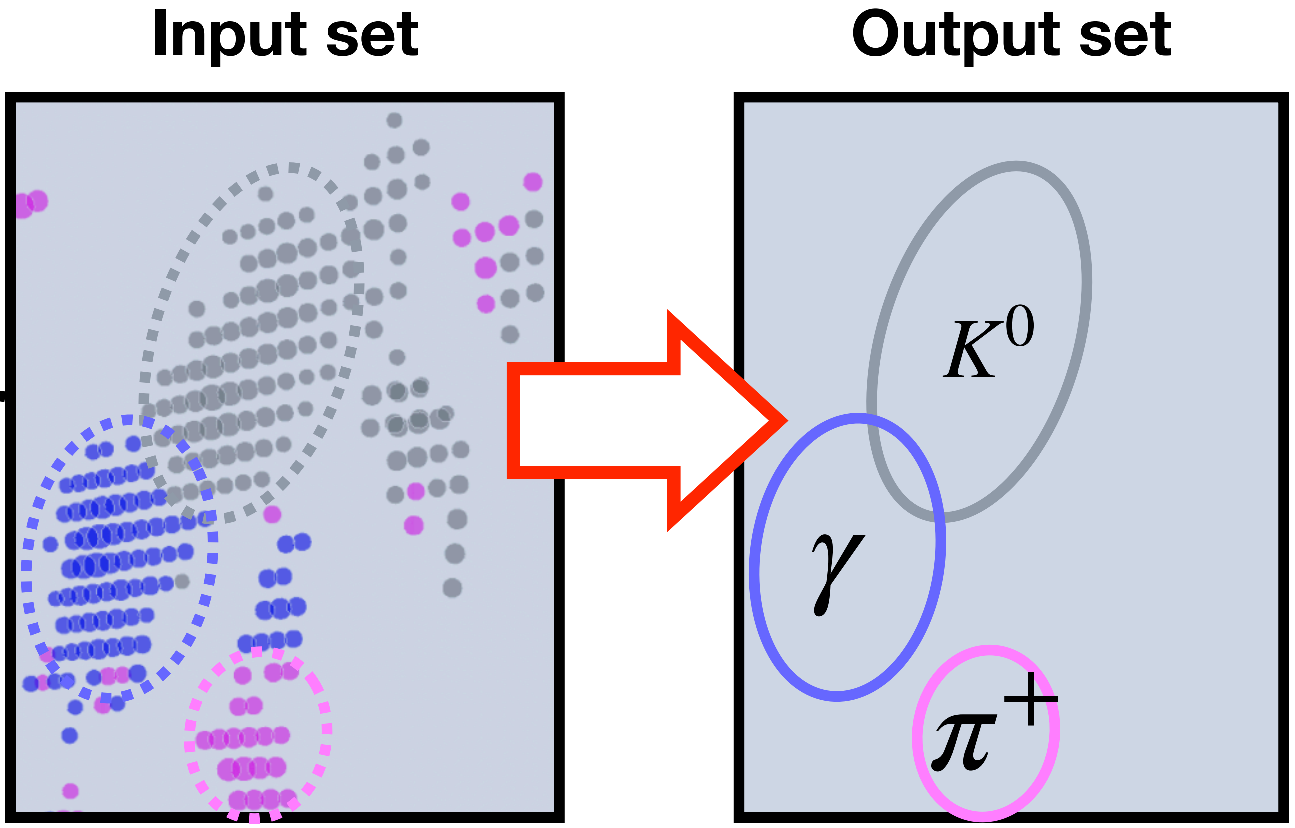
1) Identify particles (cardinality)



Particle reco. set-to-set task

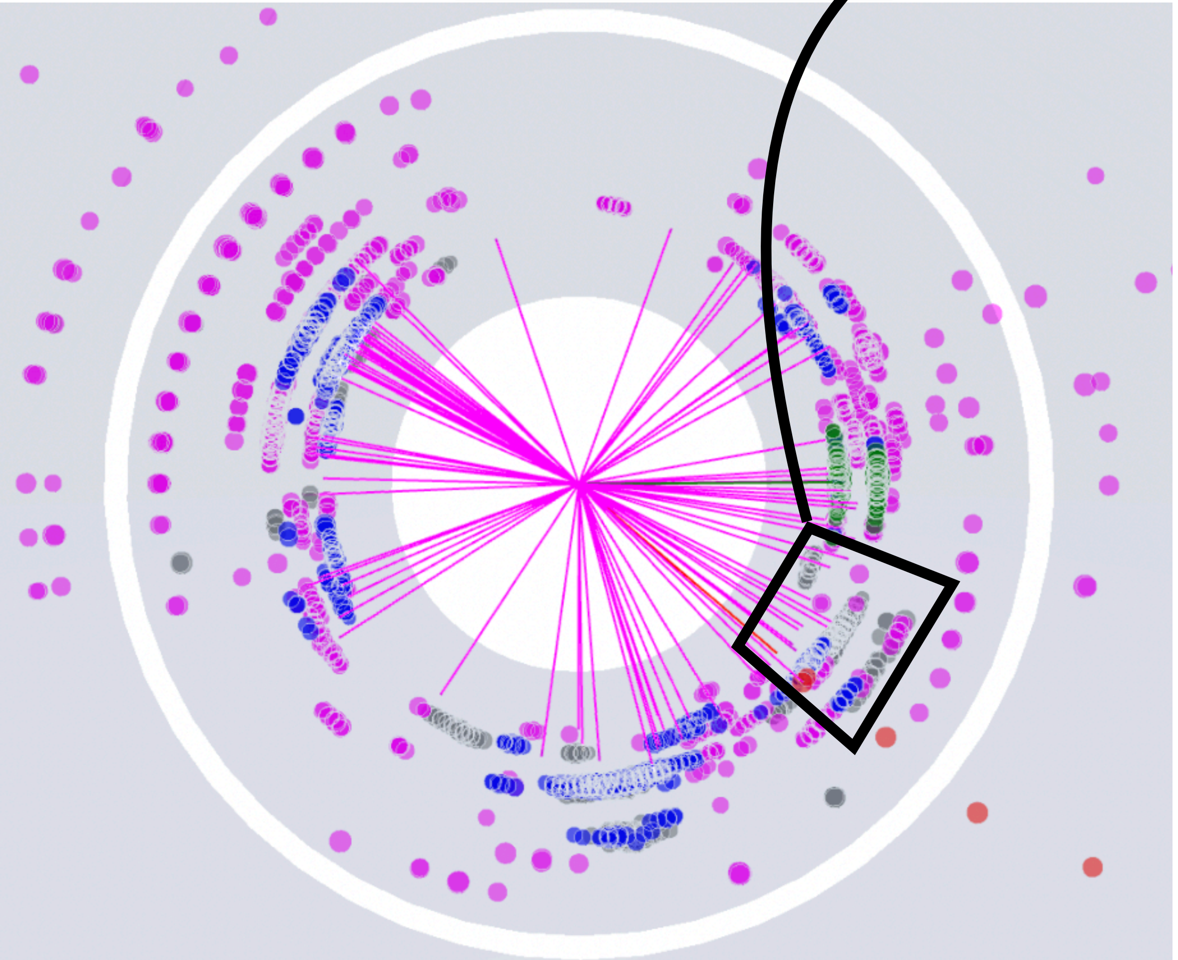


1) Identify particles (cardinality)

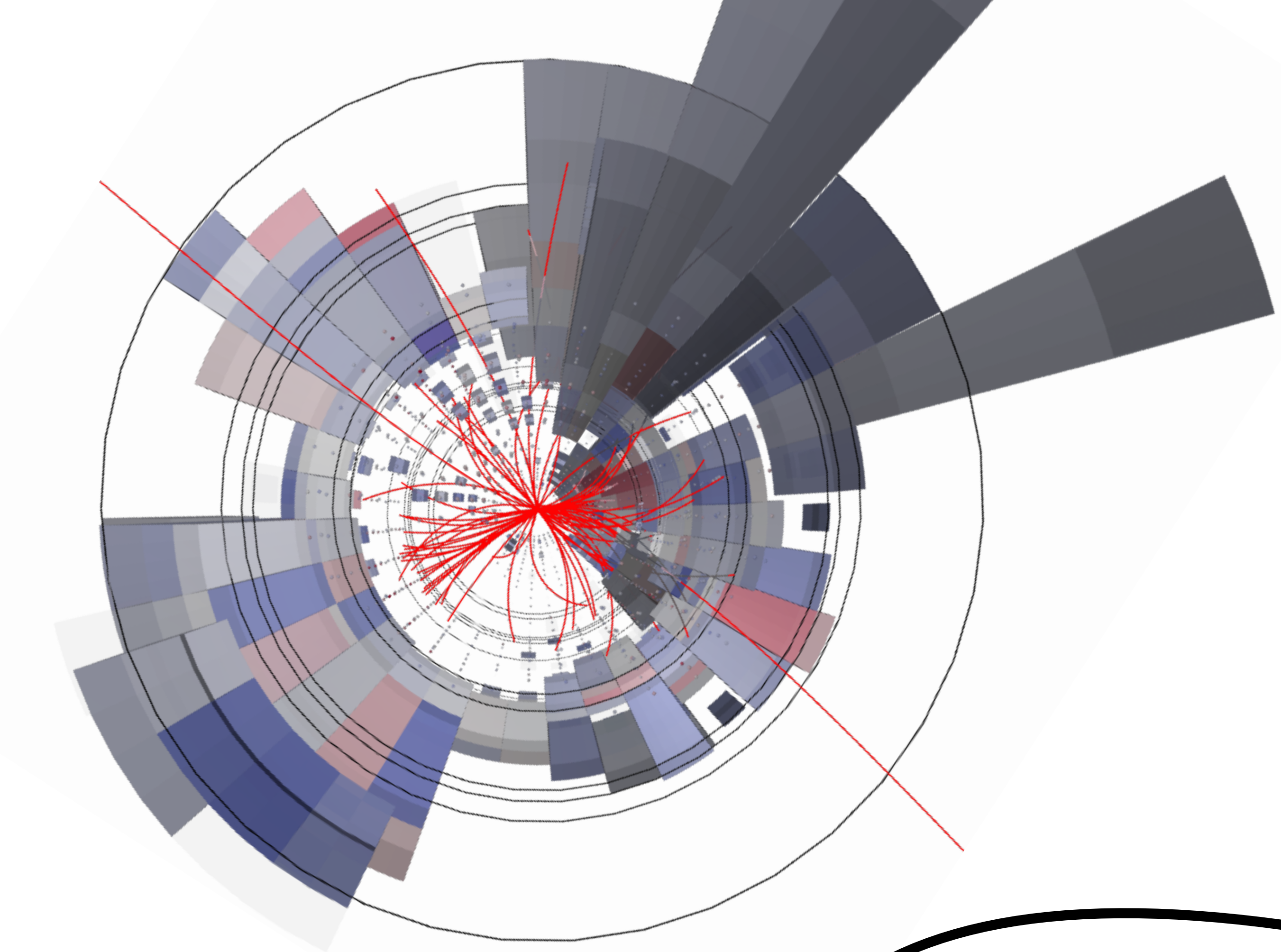


2) Classify them

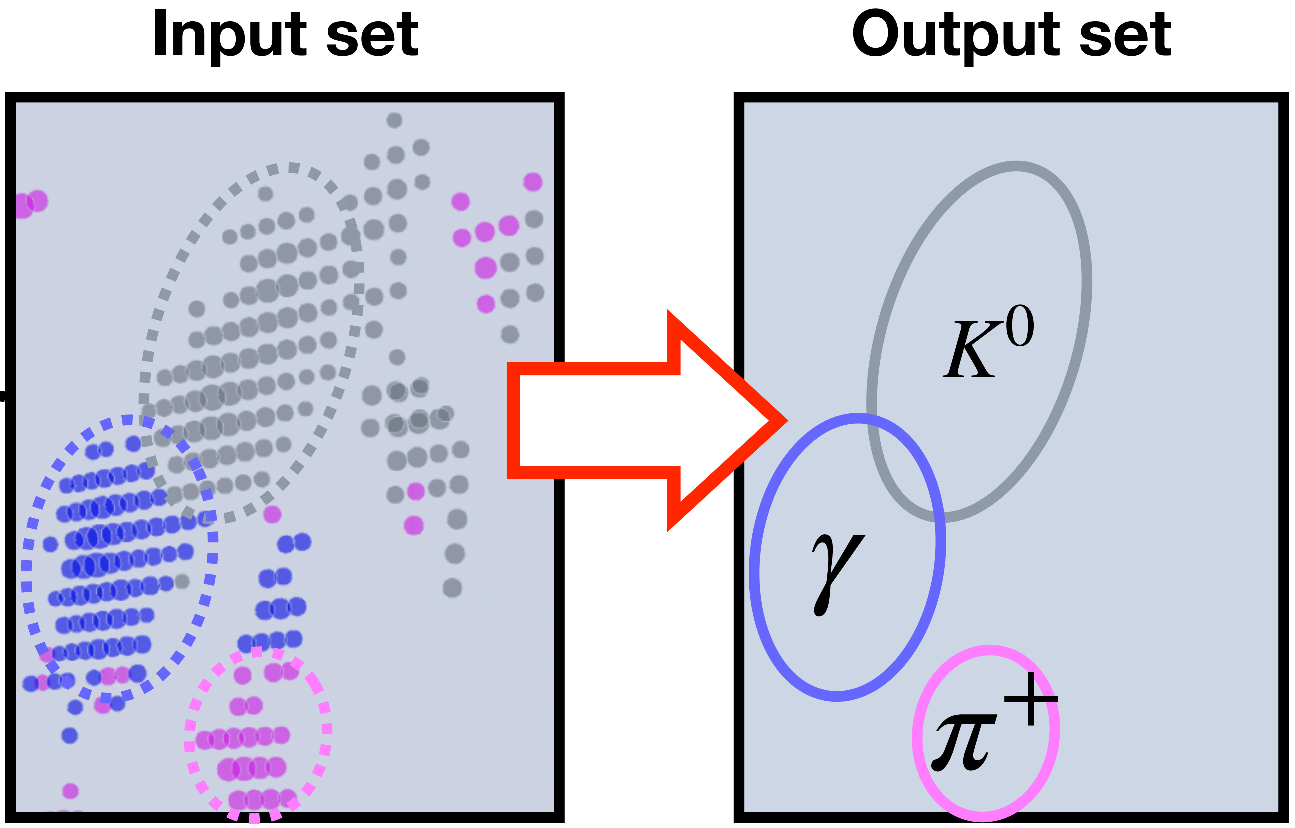
- ● neutral had.,
- ● charged had.,
- ● photon,

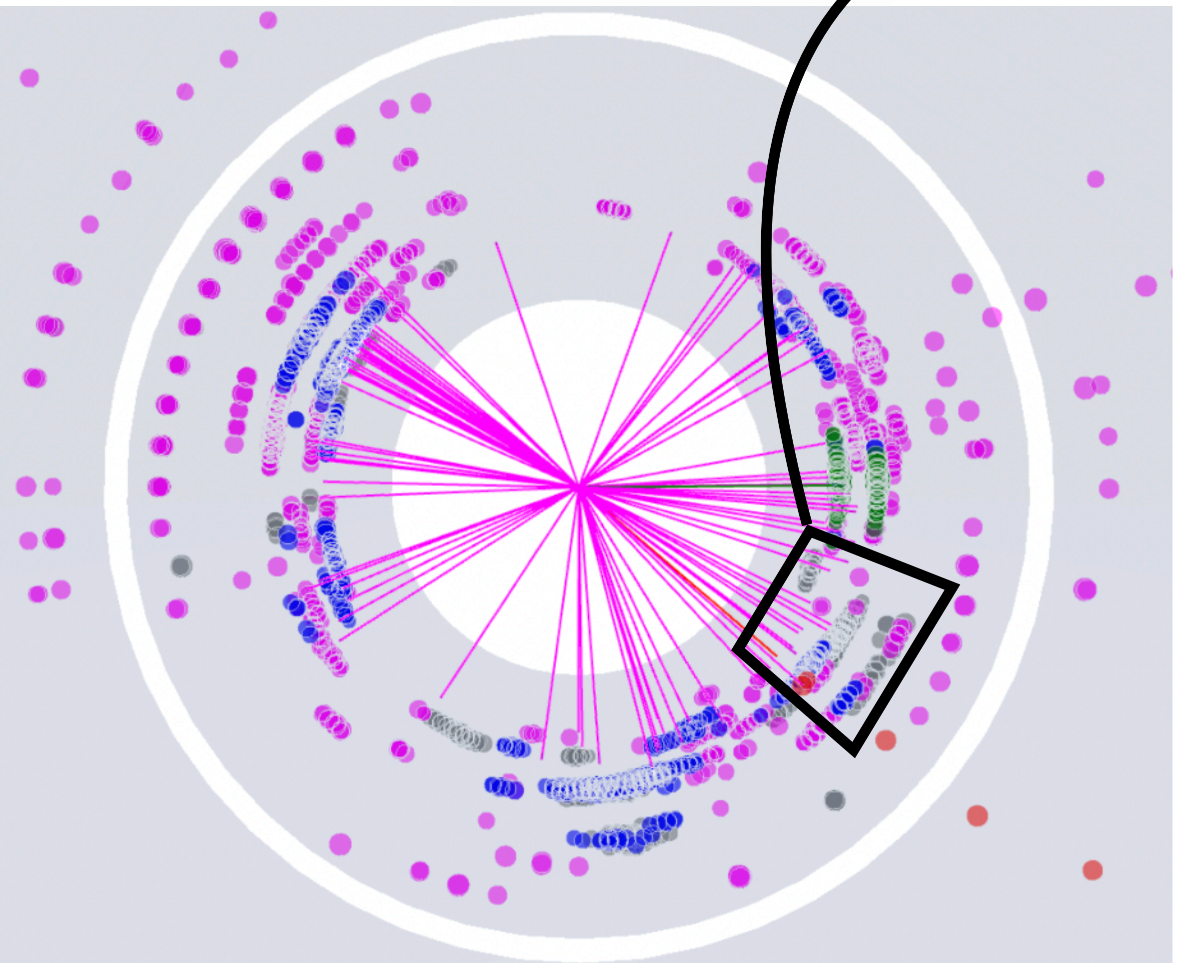


Particle reco. set-to-set task



1) <u>Identify</u> particles (cardinality)



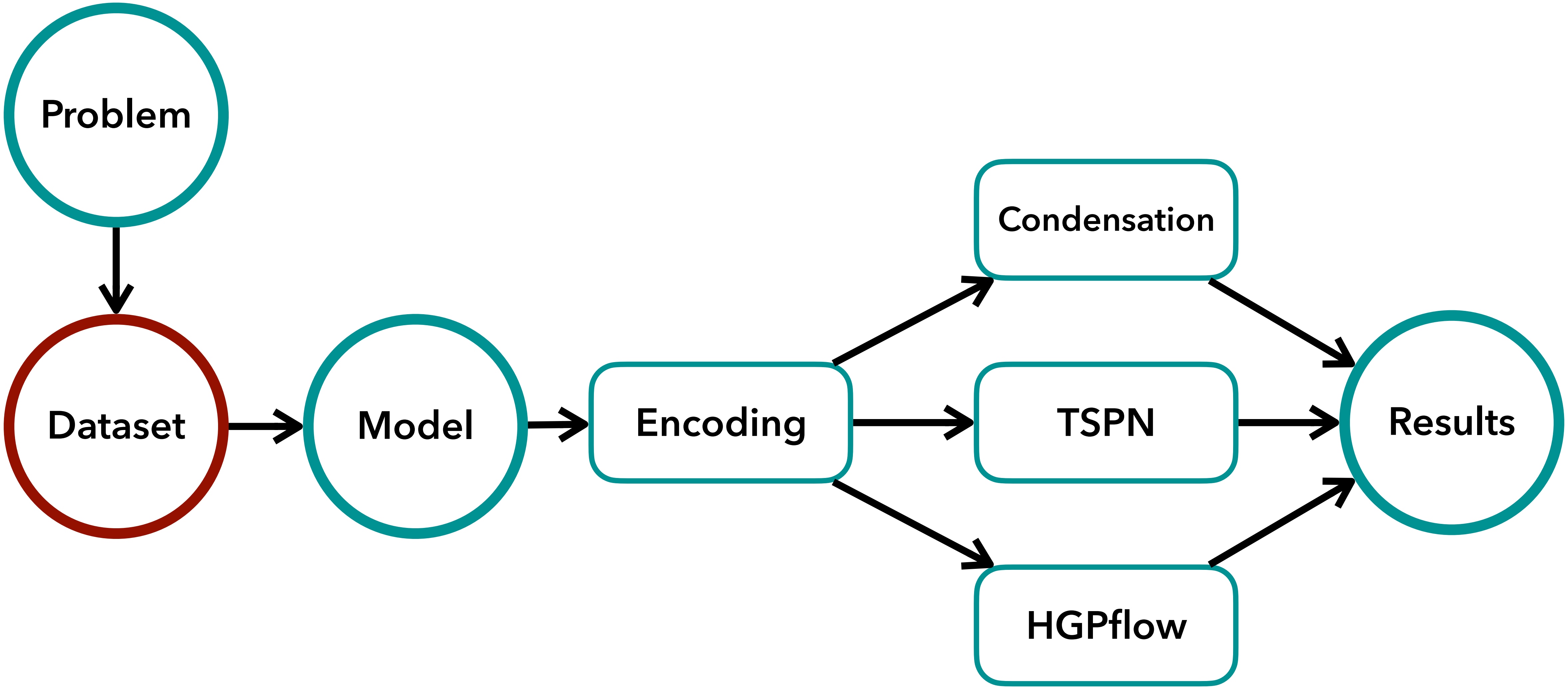


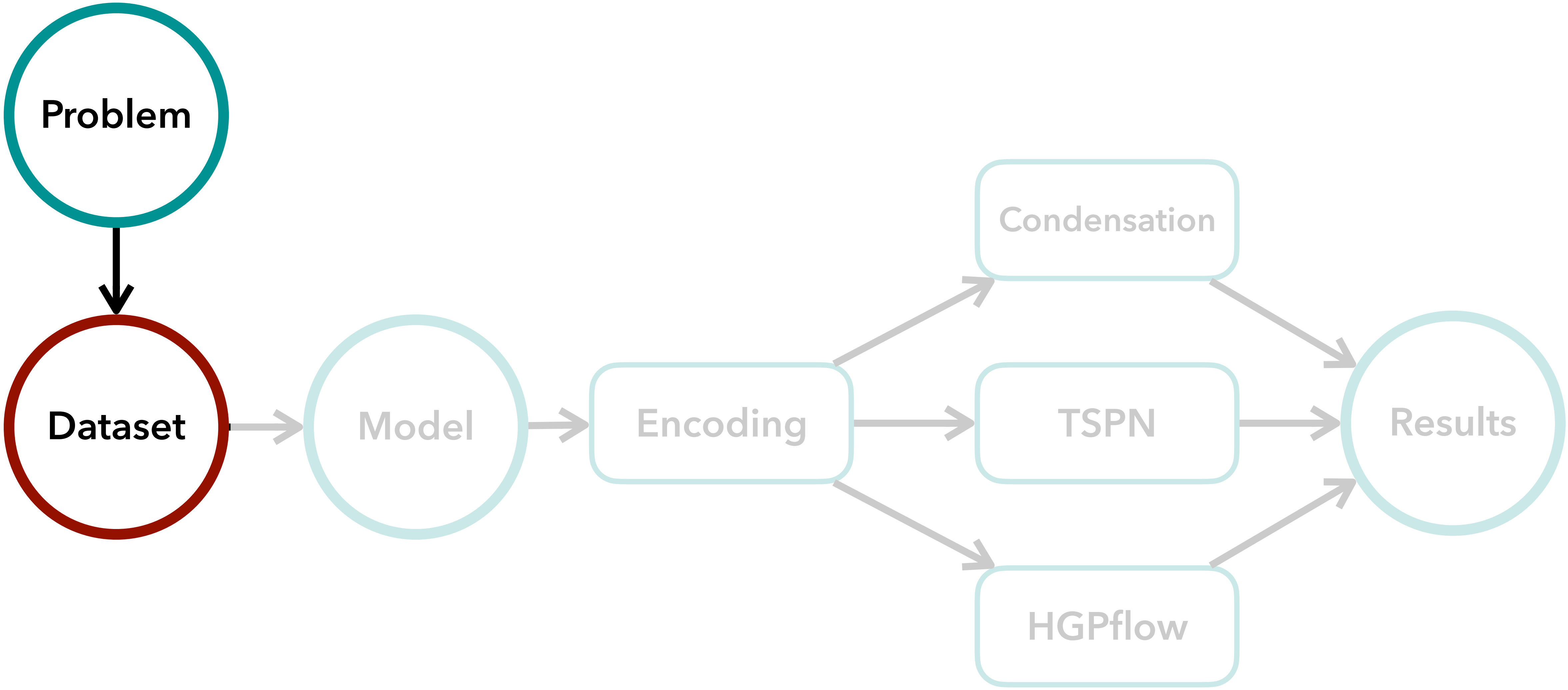
2) <u>Classify</u> them

- ● neutral had.,
- ● charged had.,
- ● photon,

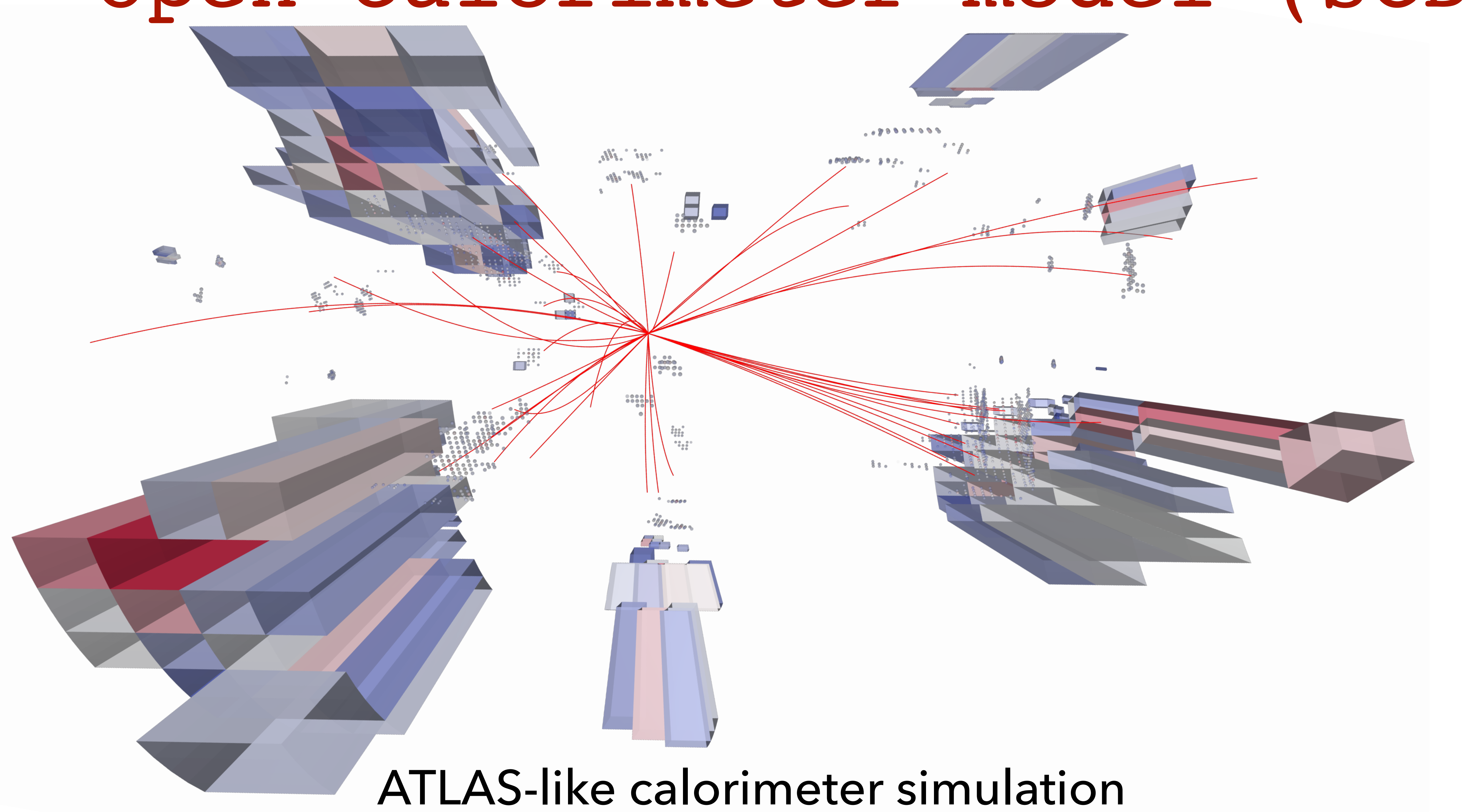
3) <u>Regress</u> their properties

- ✓ Direction (η, ϕ)
- ✓ Momentum (p_T)


Problem
Dataset
Model
Encoding
Condensation
TSPN
HGPflow
Results


Problem
Dataset
Model
Encoding
Condensation
TSPN
HGPflow
Results

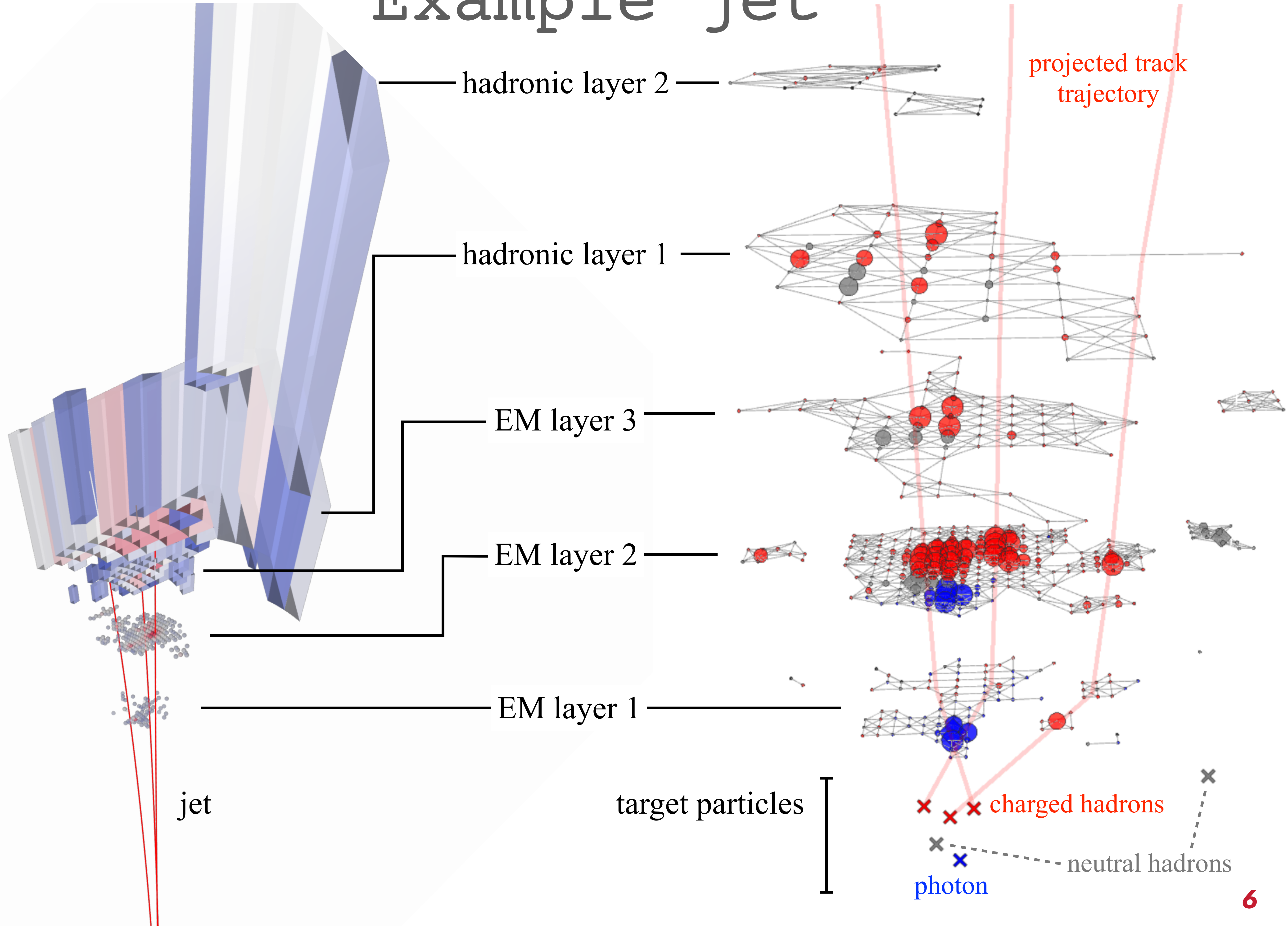
Open calorimeter model (SCD)

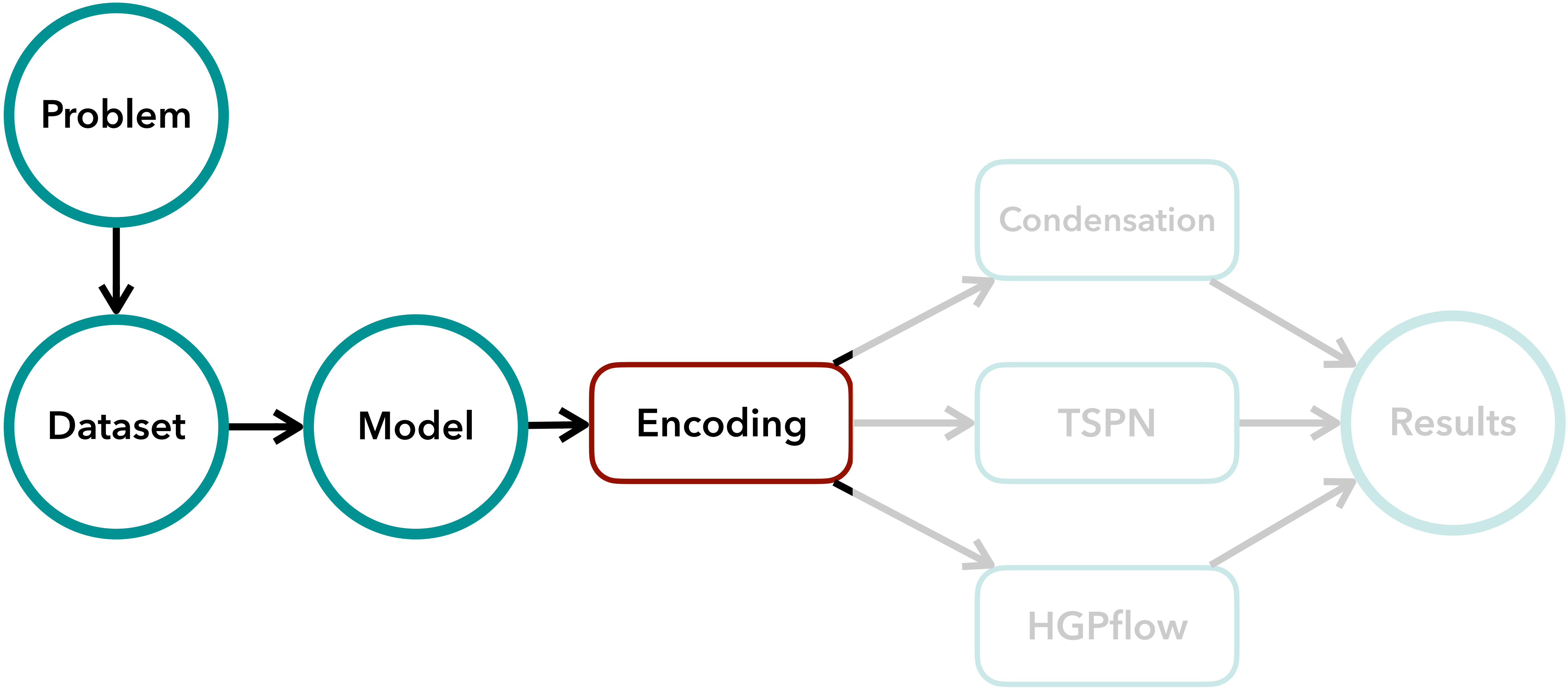


ATLAS-like calorimeter simulation

- Interfaced to Pythia8 event generator
- Tracking emulation in 3.8T magnetic field
- 3 ECAL + 3 HCAL concentric GEANT4 calorimeter layers
- To be released in forthcoming paper

Example jet


hadronic layer 2
projected track trajectory
hadronic layer 1
EM layer 3
EM layer 2
EM layer 1
jet
target particles
charged hadrons
neutral hadrons
photon



Problem

Dataset

Model

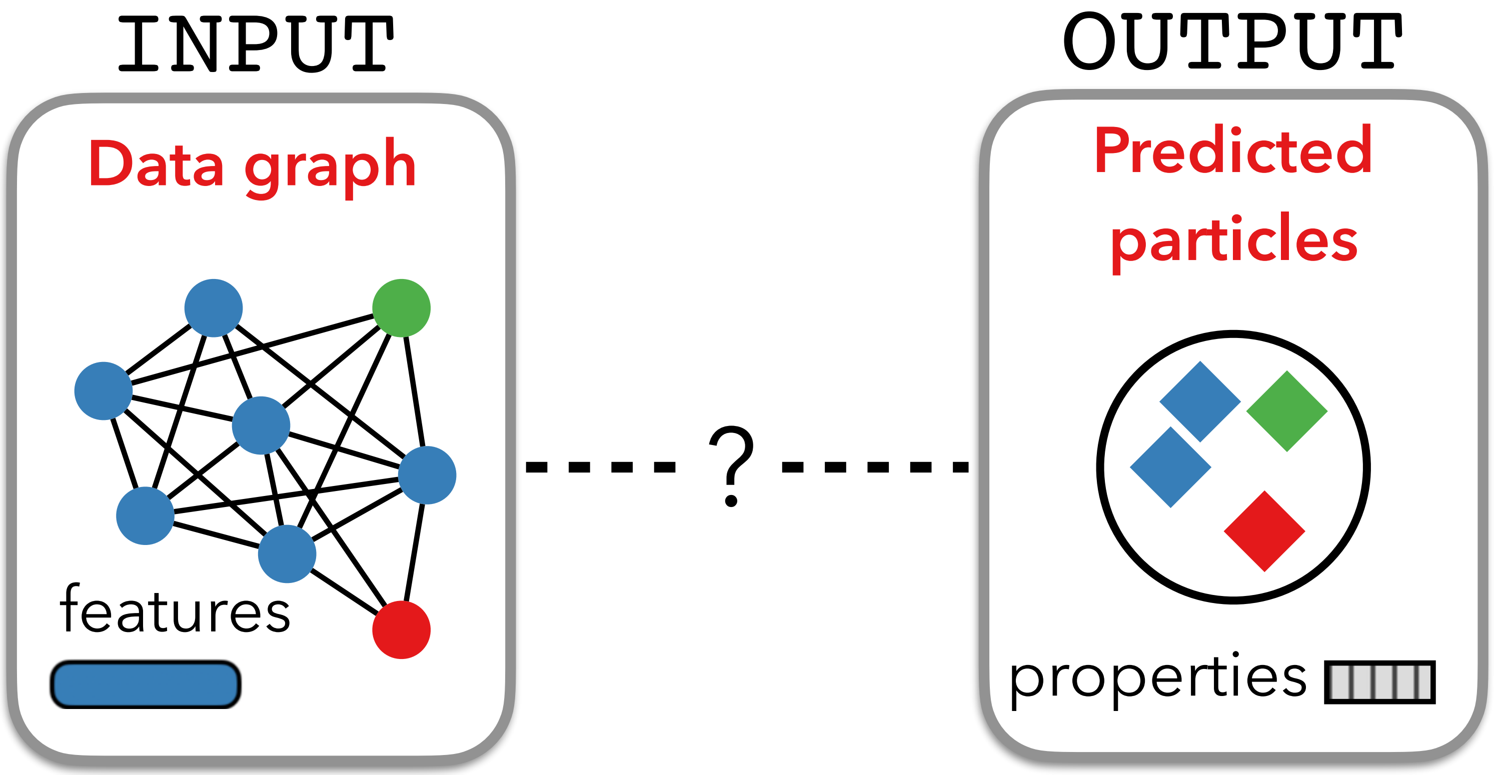
Encoding

Condensation

TSPN

HGPflow

Results



INPUT

Data graph

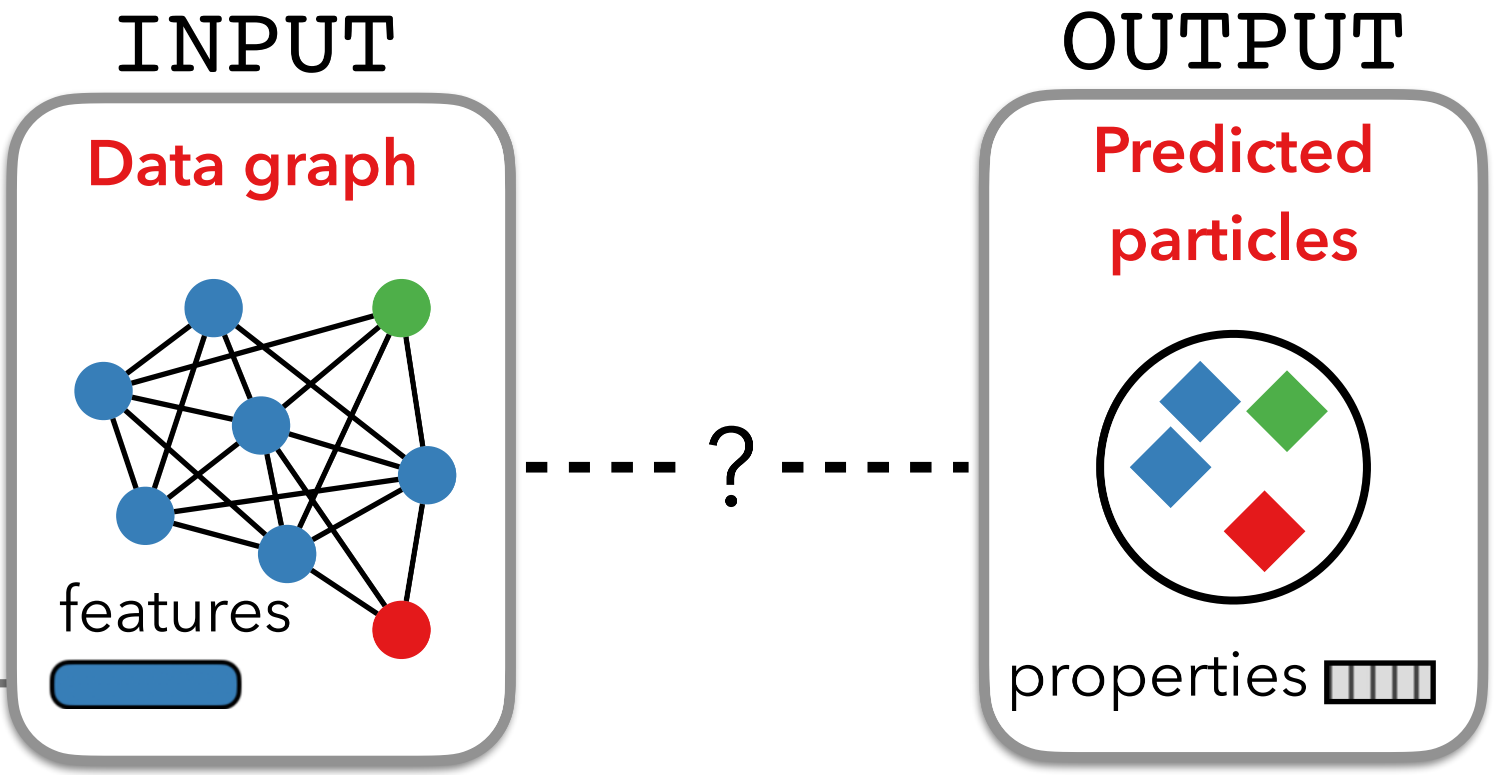
features

?

OUTPUT

Predicted particles

properties



INPUT

Data graph

features

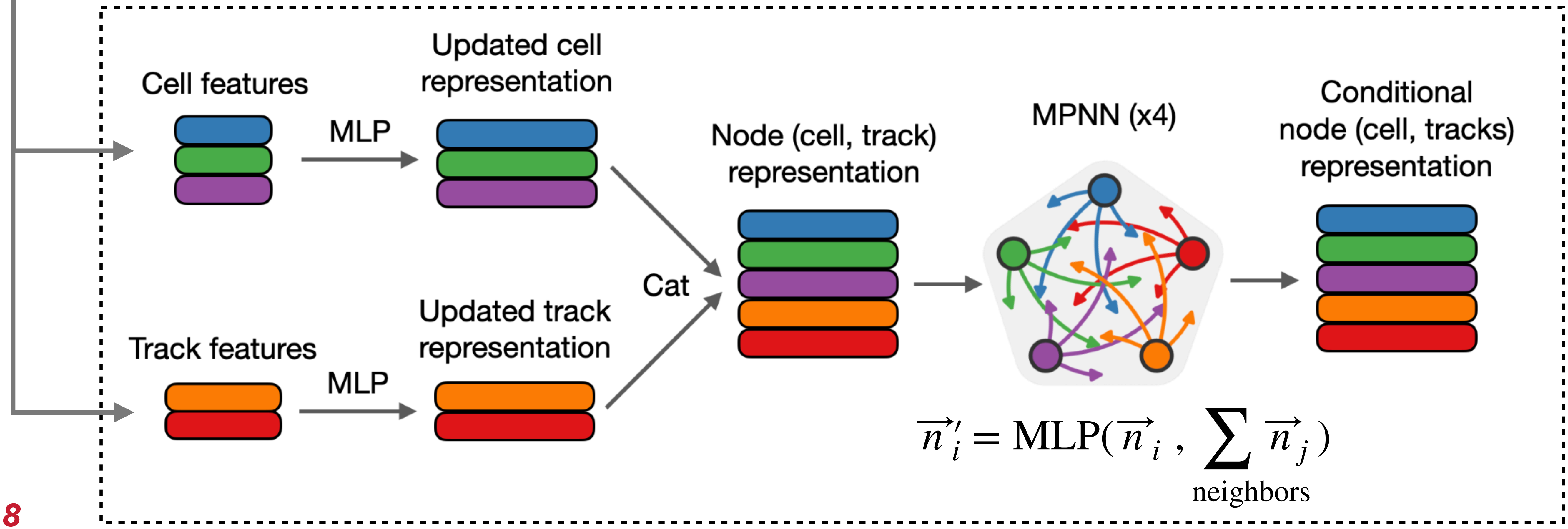
OUTPUT

Predicted particles

properties

?

Node encoding



Cell features

MLP

Updated cell representation

Track features

MLP

Updated track representation

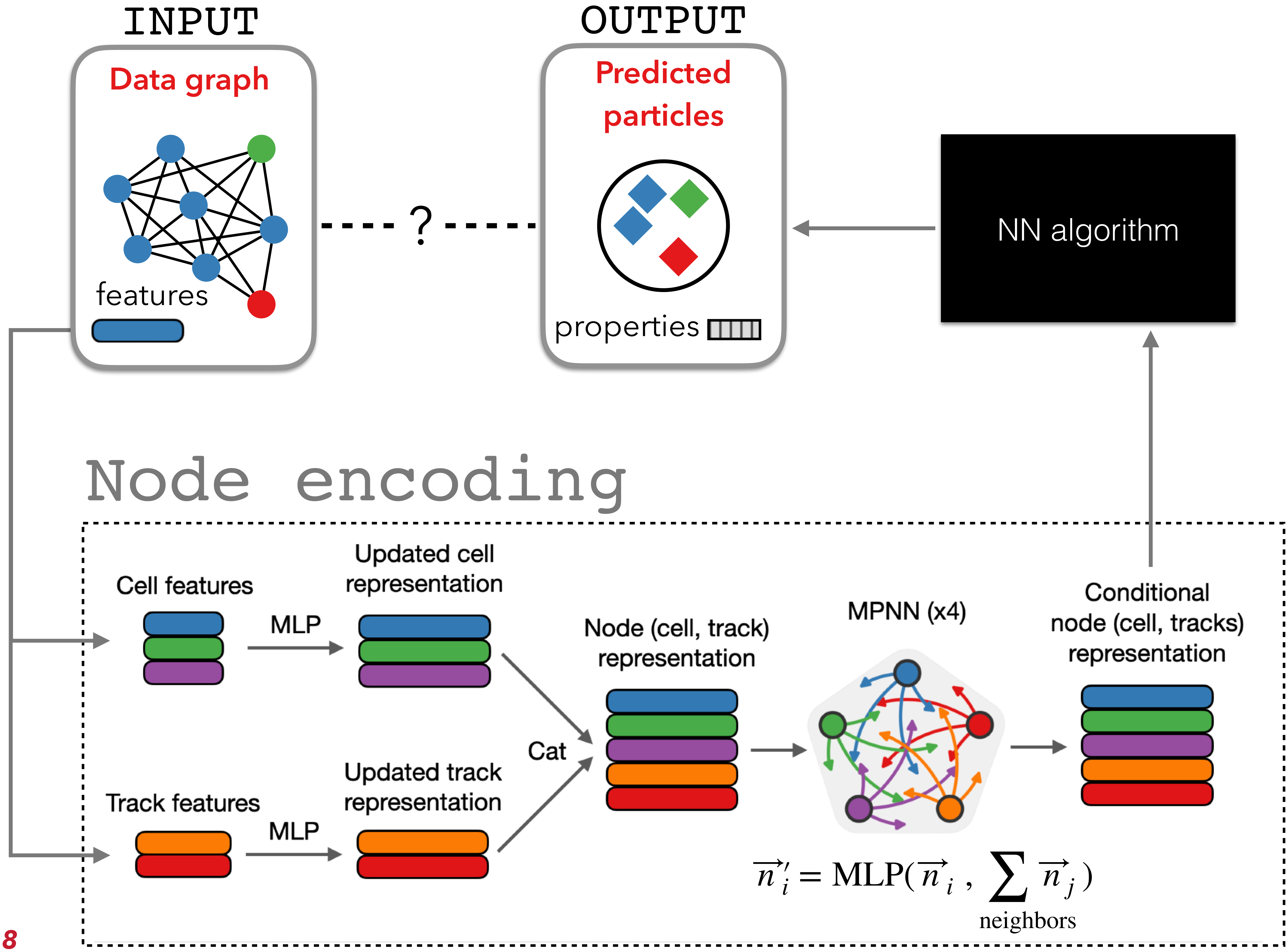
Cat

Node (cell, track) representation

MPNN (x4)

Conditional node (cell, tracks) representation

$$\vec{n}'_i = \text{MLP}(\vec{n}_i\,,\ \sum_{\text{neighbors}} \vec{n}_j)$$



INPUT

Data graph

features

OUTPUT

Predicted particles

properties

?

NN algorithm

Node encoding

Cell features

MLP

Updated cell representation

Track features

MLP

Updated track representation

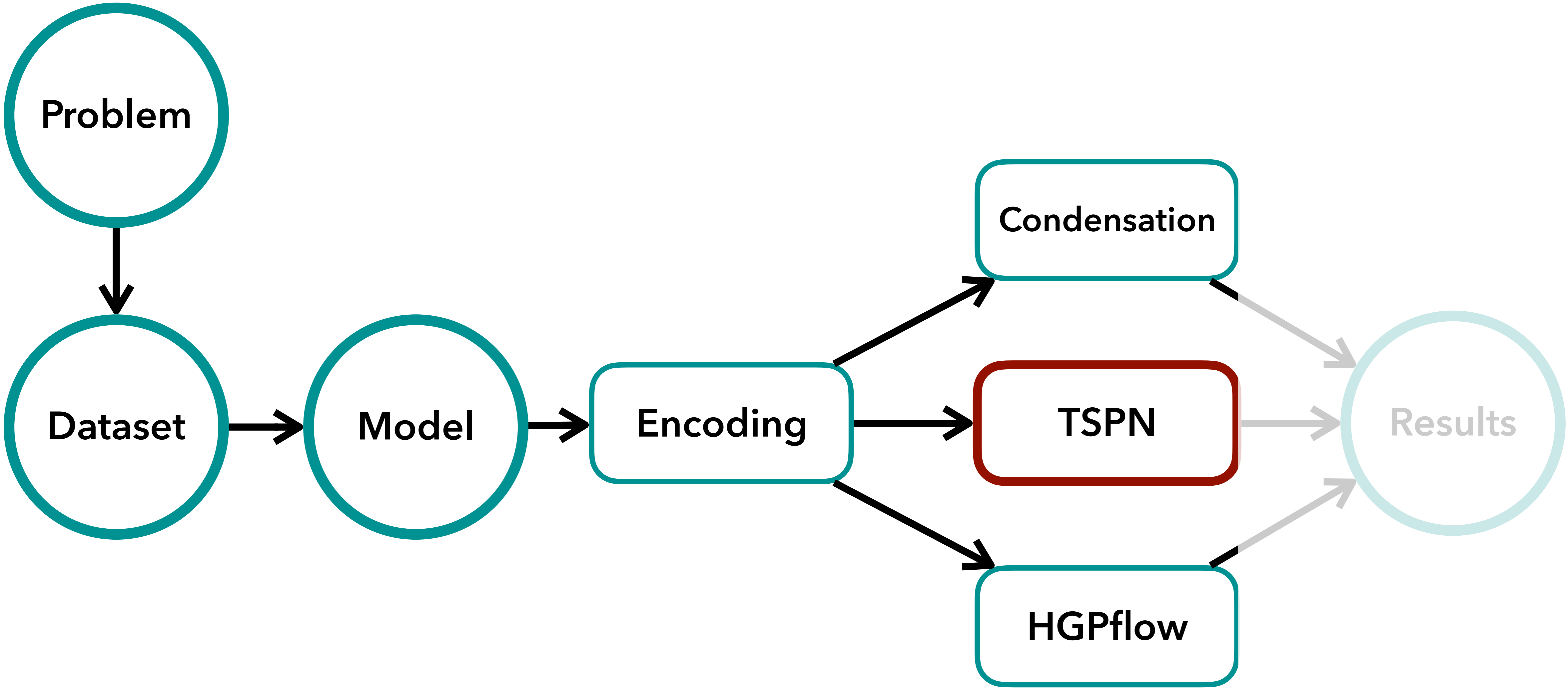
Cat

Node (cell, track) representation

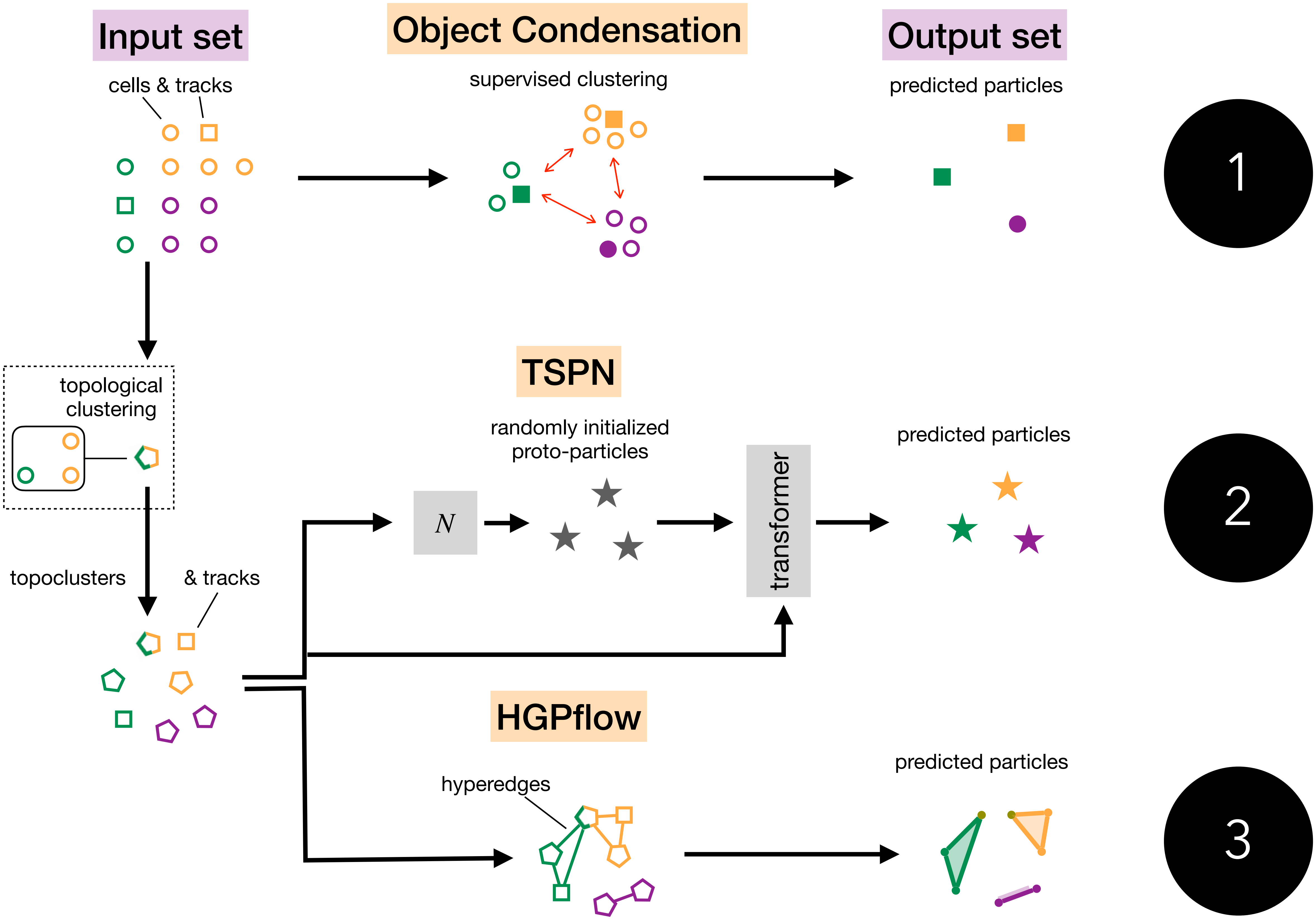
MPNN (x4)

Conditional node (cell, tracks) representation

$$\vec{n}'_i = \mathrm{MLP}(\vec{n}_i \, , \sum_{\text{neighbors}} \vec{n}_j)$$


Problem
Dataset
Model
Encoding
Condensation
TSPN
HGPflow
Results

Going from (many) nodes to (few) particles



Input set

cells & tracks

Object Condensation

supervised clustering

Output set

predicted particles

1

topological clustering

TSPN

randomly initialized proto-particles

N

transformer

predicted particles

2

topoclusters

& tracks

HGPflow

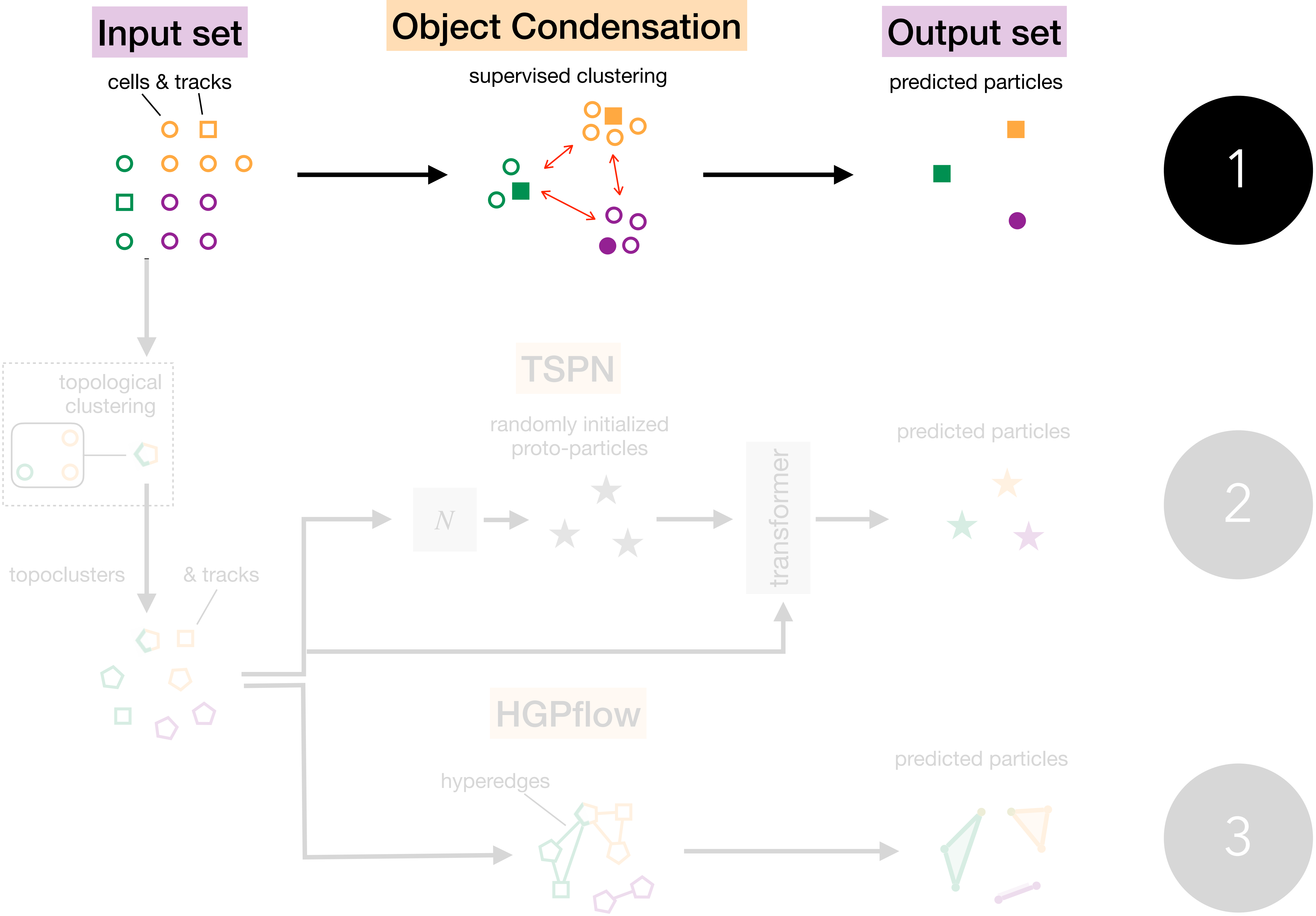
hyperedges

predicted particles

3

Going from (many) nodes to (few) particles

J. Kieseler arXiv:2002.03605 *(won't have time to present today)*



Input set

cells & tracks

Object Condensation

supervised clustering

Output set

predicted particles

1

topological clustering

topoclusters

& tracks

TSPN

randomly initialized proto-particles

N

transformer

predicted particles

2

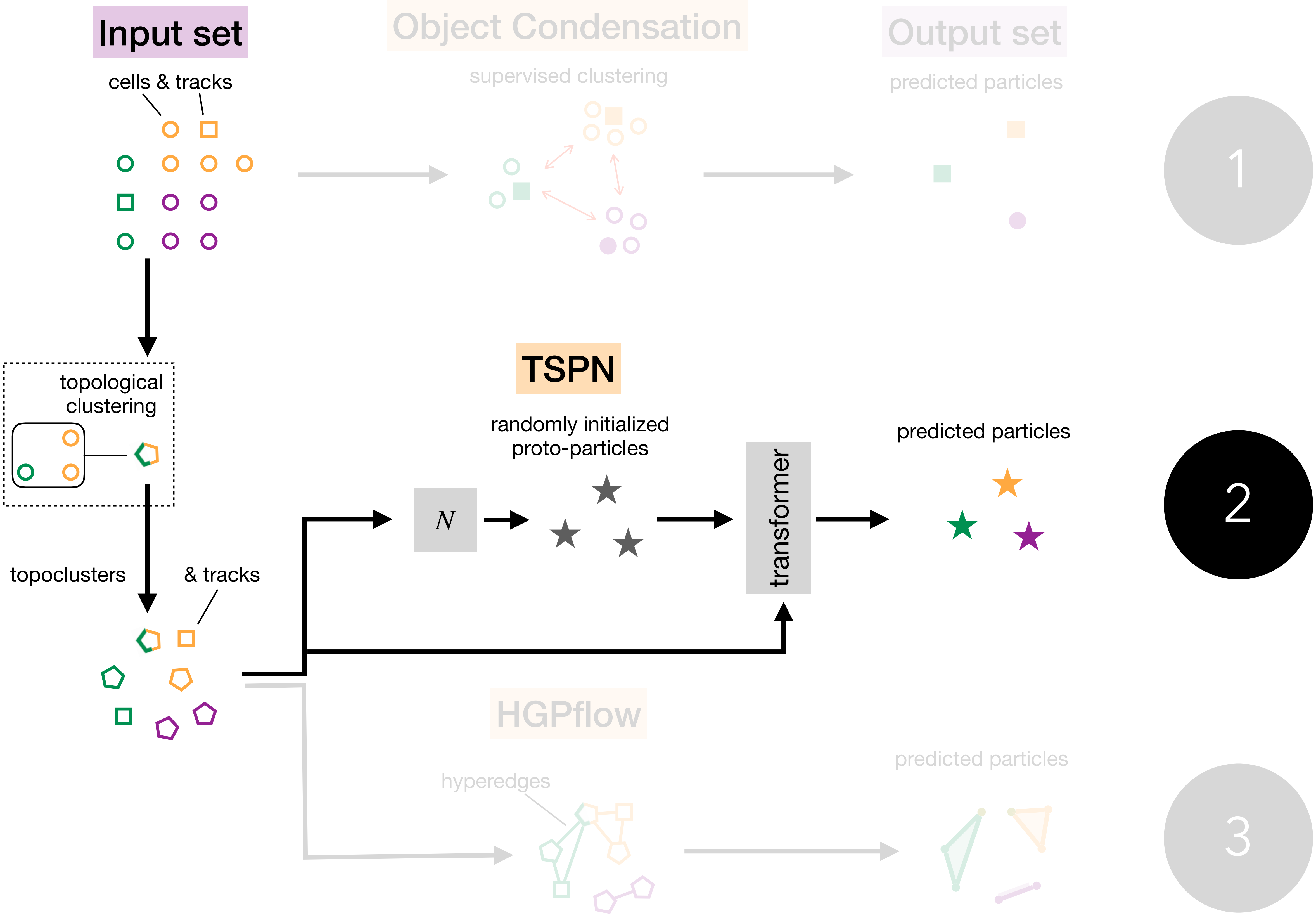
HGPflow

hyperedges

predicted particles

3

Going from (many) nodes to (few) particles



Input set

cells & tracks

topological clustering

topoclusters

& tracks

Object Condensation

supervised clustering

Output set

predicted particles

1

TSPN

randomly initialized proto-particles

predicted particles

2

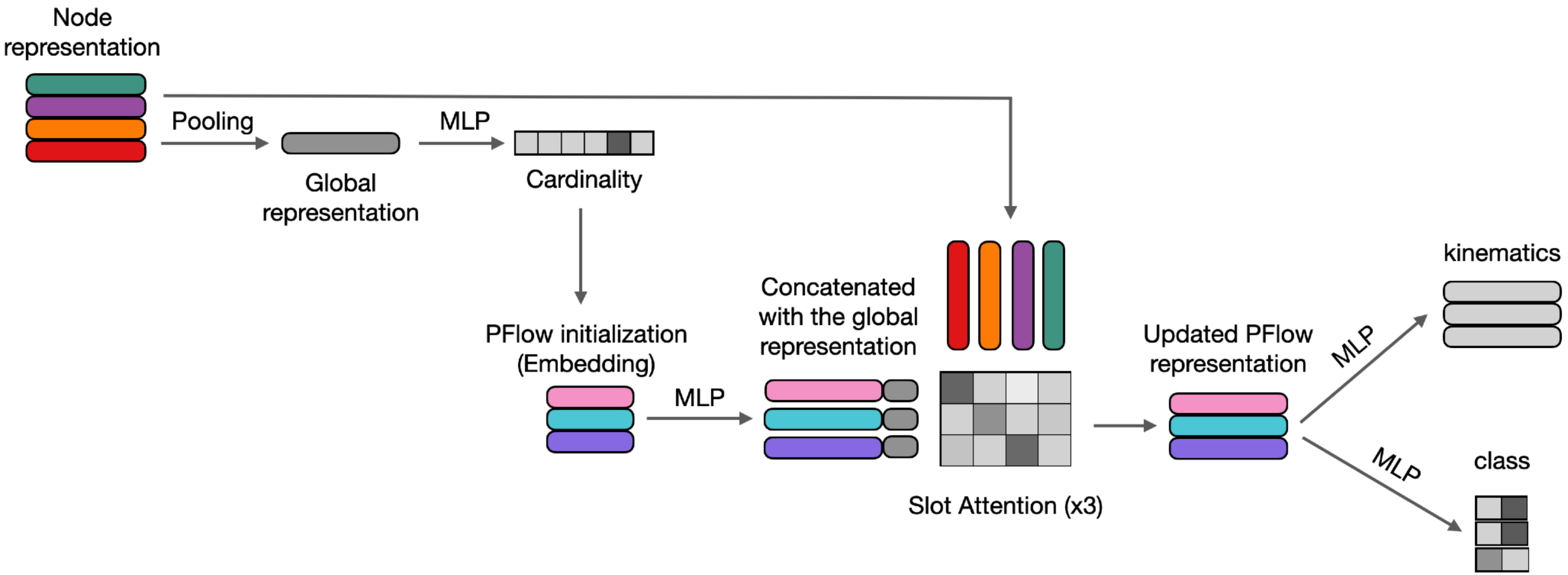
HGPflow

hyperedges

predicted particles

3

Transformer set prediction with slot attention

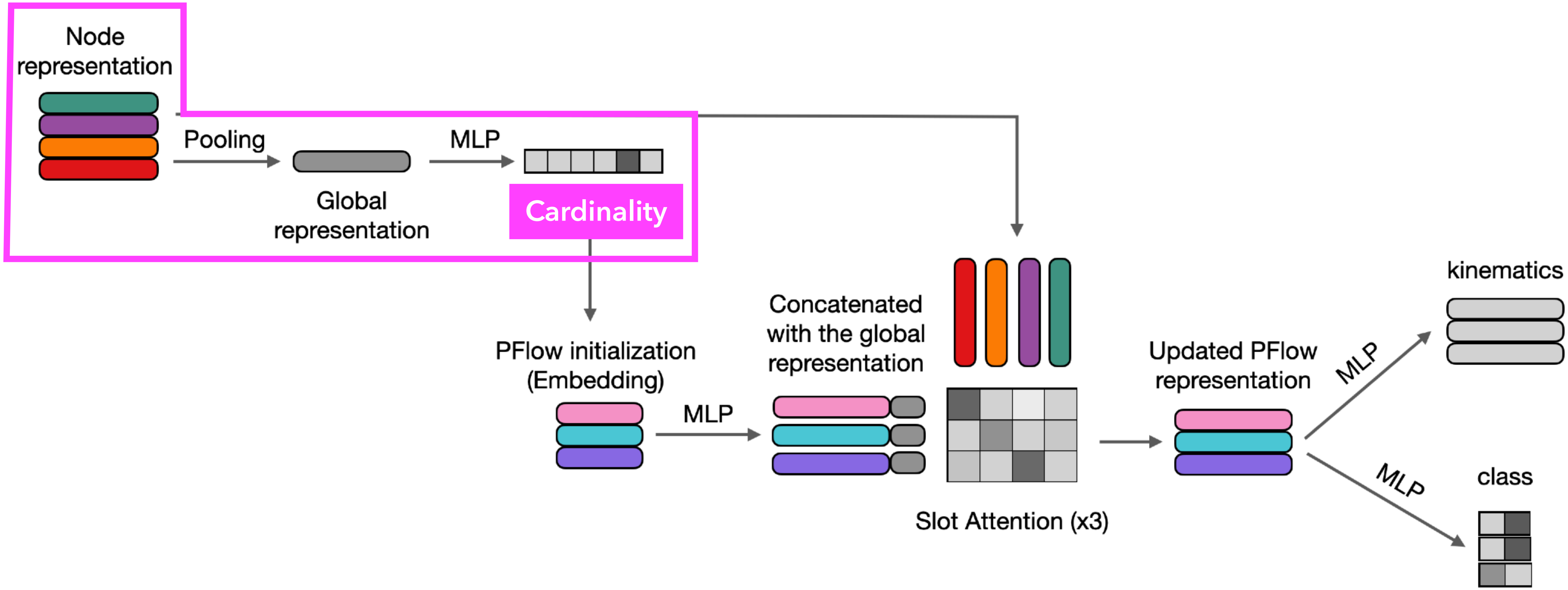


Based on:

A. R. Kosiorek, H. Kim, D. J. Rezende
arXiv:2006.16841

F. Locatello, D. Weissenborn et al.
arXiv:2006.15055

Transformer set prediction with slot attention

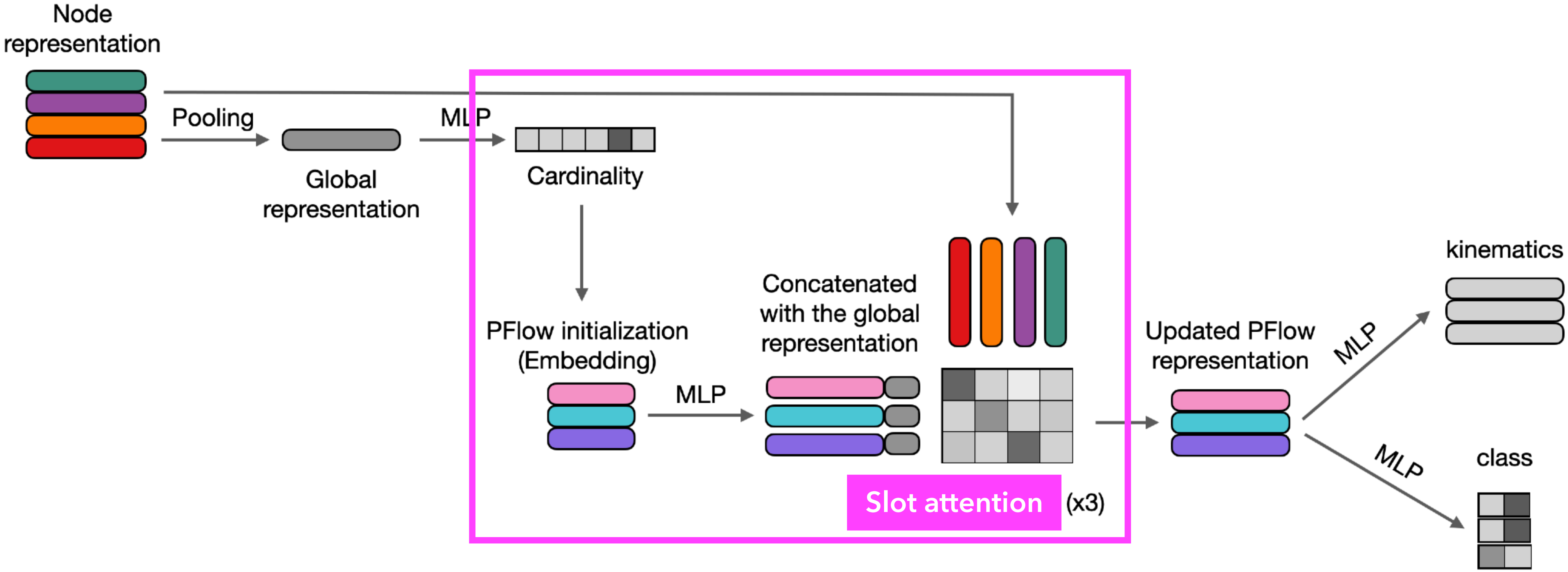


<u>Based on:</u>

A. R. Kosiorek, H. Kim, D. J. Rezende
arXiv:2006.16841

F. Locatello, D. Weissenborn et al.
arXiv:2006.15055

Transformer set prediction with slot attention



attention weights

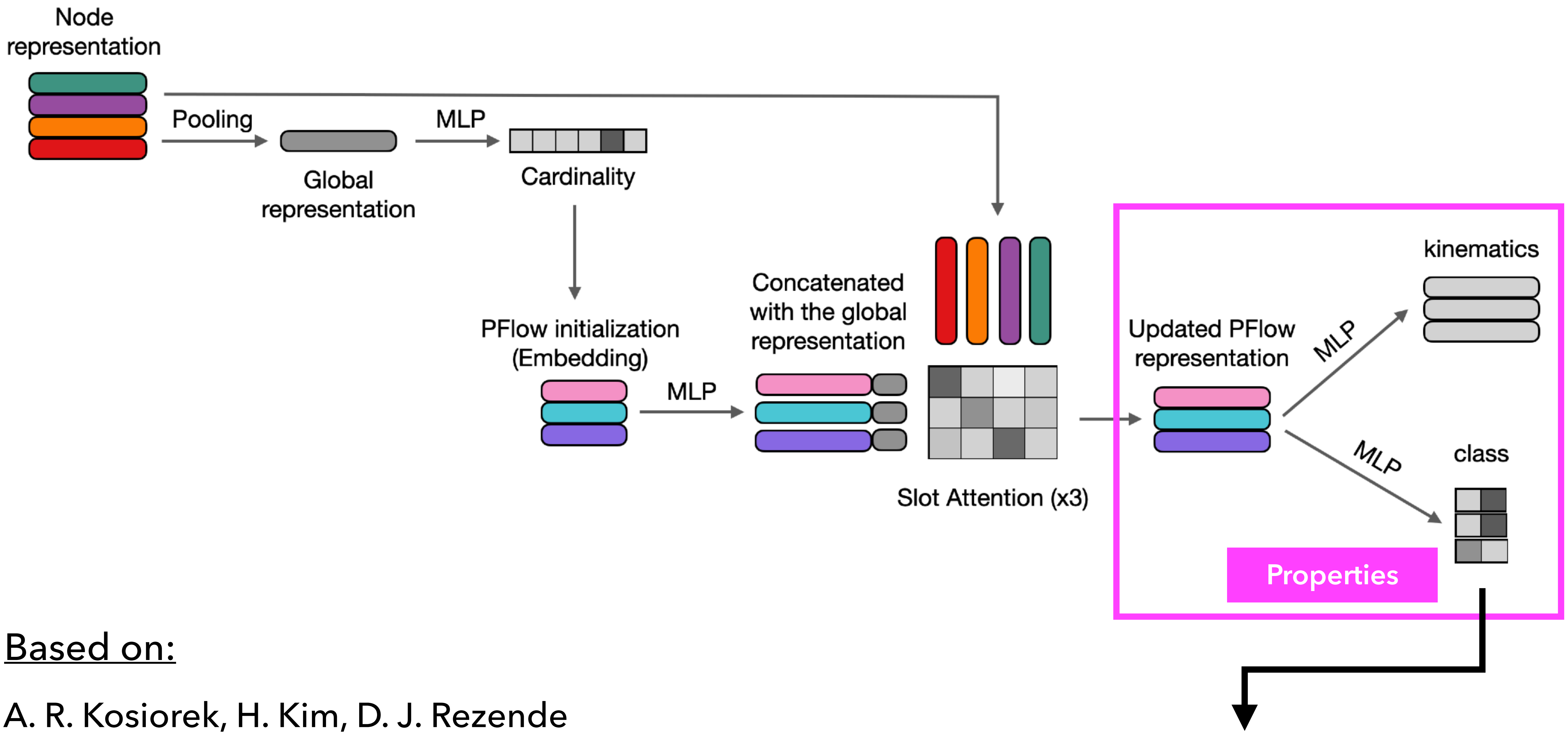
~ particle-node affinities

Based on:

A. R. Kosiorek, H. Kim, D. J. Rezende
arXiv:2006.16841

F. Locatello, D. Weissenborn et al.
arXiv:2006.15055

Transformer set prediction with slot attention



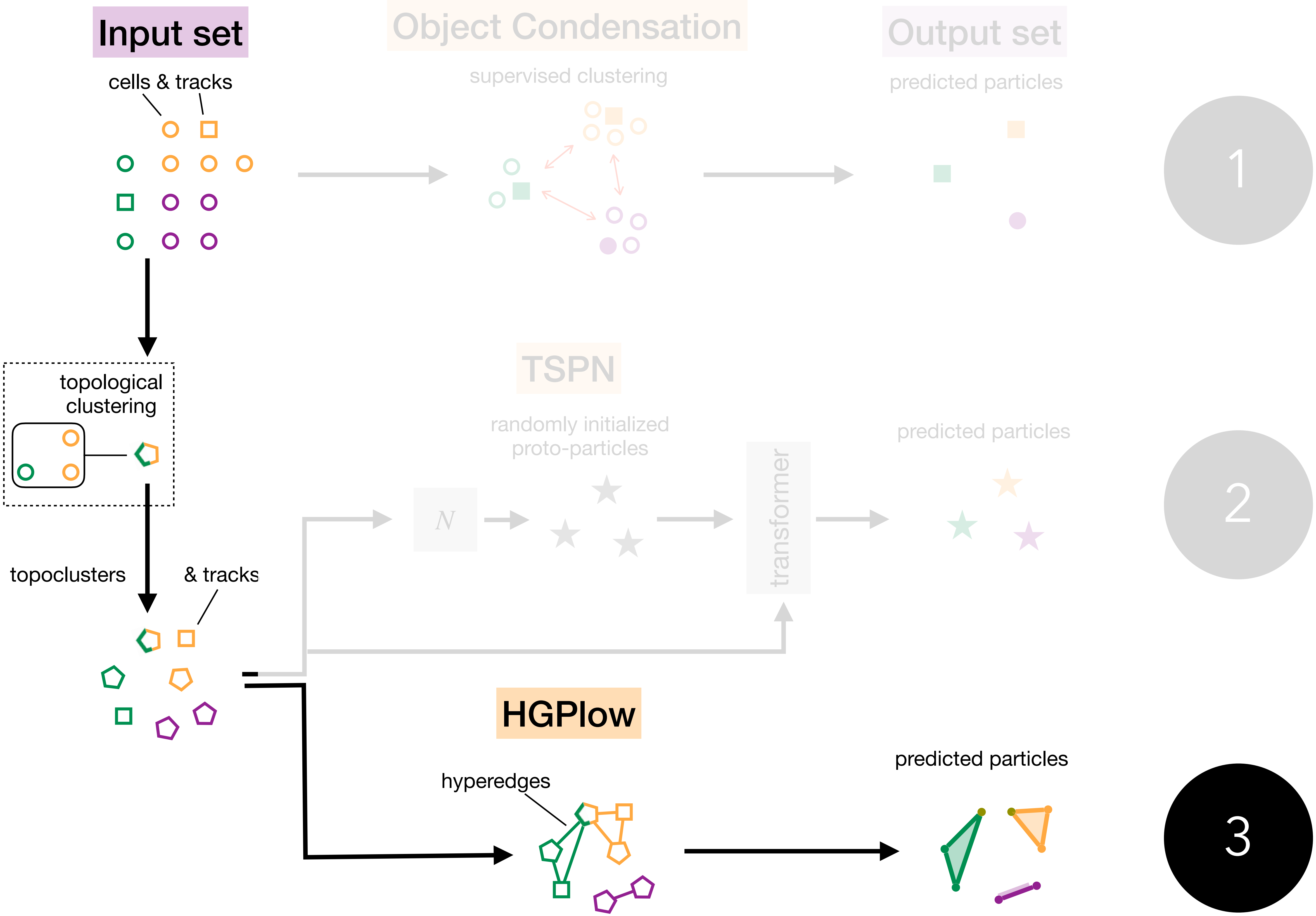
Based on:

A. R. Kosiorek, H. Kim, D. J. Rezende
arXiv:2006.16841

F. Locatello, D. Weissenborn et al.
arXiv:2006.15055

compared with target particles
via permutation-invariant matching
(Hungarian algorithm)

Going from (many) nodes to (few) particles



Input set

cells & tracks

Object Condensation

supervised clustering

Output set

predicted particles

1

topological clustering

TSPN

randomly initialized proto-particles

N

transformer

predicted particles

2

topoclusters

& tracks

HGPlow

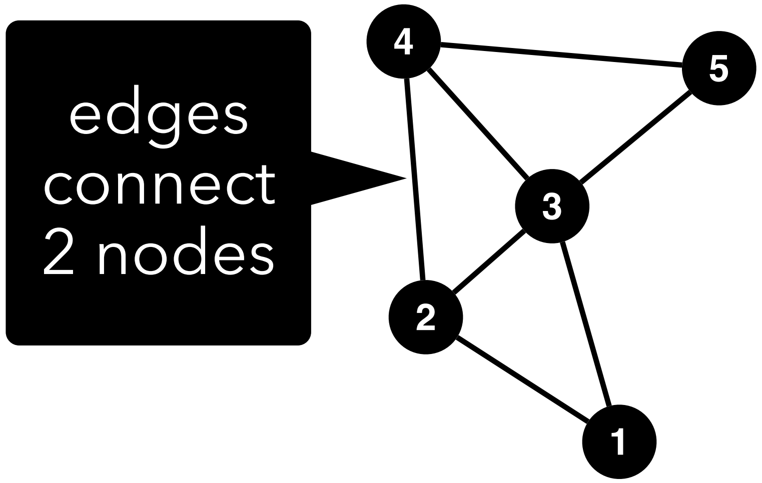
hyperedges

predicted particles

3

What is a hypergraph?

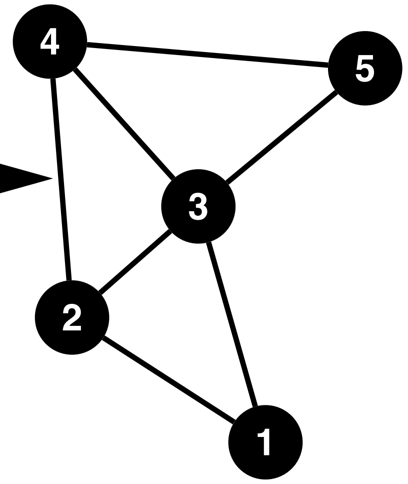
Graph



edges connect 2 nodes

What is a hypergraph?

Graph



edges connect 2 nodes

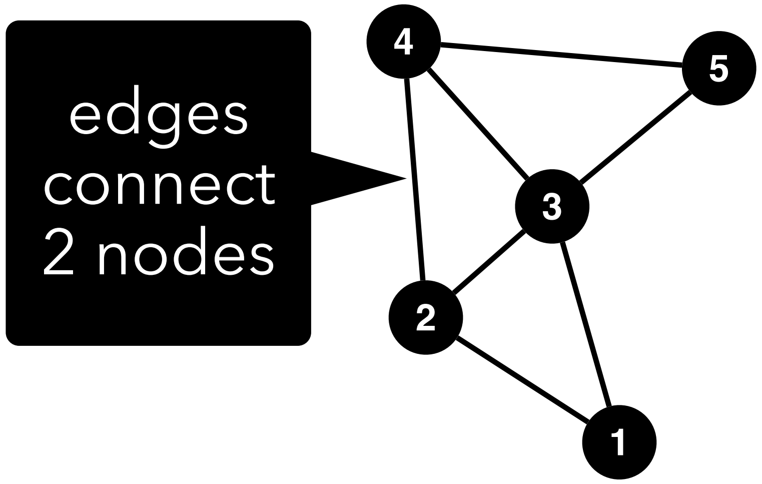
Adjacency matrix

$$\begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 1 & 1 & 1 & 0 & 0 \\ 2 & 1 & 1 & 1 & 1 & 0 \\ 3 & 1 & 1 & 1 & 1 & 1 \\ 4 & 0 & 1 & 1 & 1 & 1 \\ 5 & 0 & 0 & 1 & 1 & 1 \end{array}$$

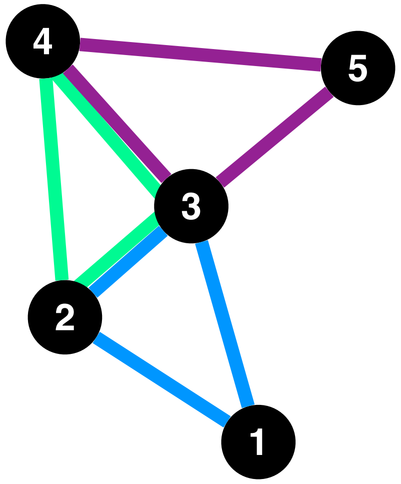
$$(N \times N)$$

What is a hypergraph?

Graph


edges connect 2 nodes

Hypergraph

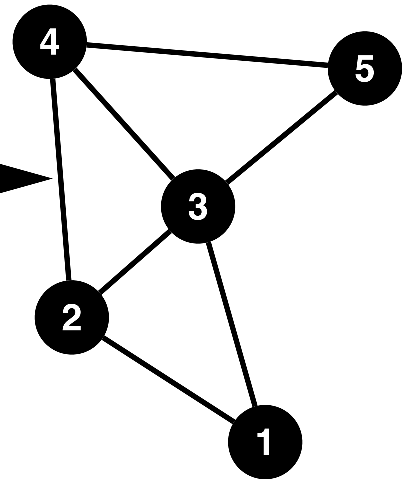


Adjacency matrix

$$\begin{array}{c|ccccc} & ① & ② & ③ & ④ & ⑤ \\ \hline ① & 1 & 1 & 1 & 0 & 0 \\ ② & 1 & 1 & 1 & 1 & 0 \\ ③ & 1 & 1 & 1 & 1 & 1 \\ ④ & 0 & 1 & 1 & 1 & 1 \\ ⑤ & 0 & 0 & 1 & 1 & 1 \end{array} \quad (N \times N)$$

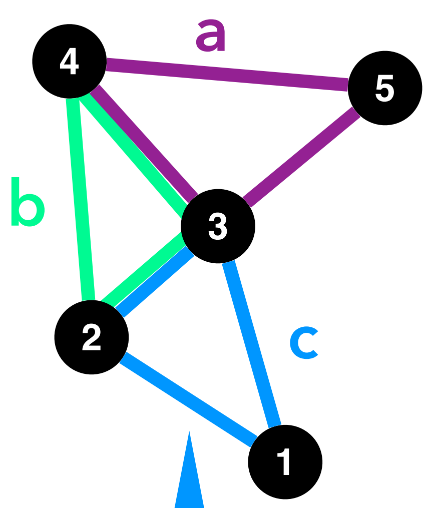
What is a hypergraph?

Graph



edges connect 2 nodes

Hypergraph



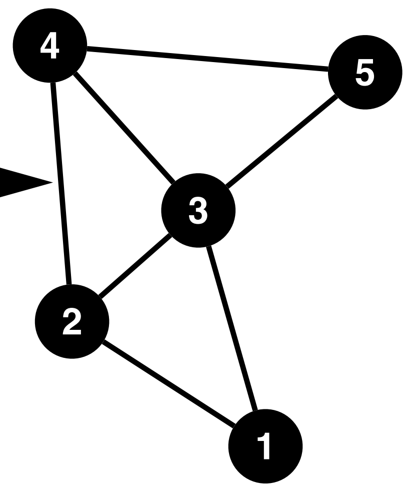
hyperedges connect ≥ 1 nodes

Adjacency matrix

$$\begin{array}{c|ccccc|} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 1 & 1 & 1 & 0 & 0 \\ 2 & 1 & 1 & 1 & 1 & 0 \\ 3 & 1 & 1 & 1 & 1 & 1 \\ 4 & 0 & 1 & 1 & 1 & 1 \\ 5 & 0 & 0 & 1 & 1 & 1 \end{array} \qquad (N \times N)$$

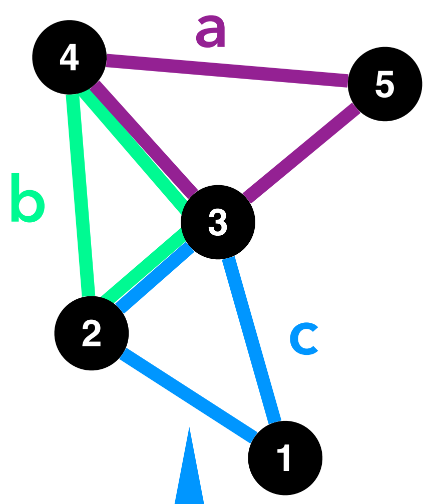
What is a hypergraph?

Graph



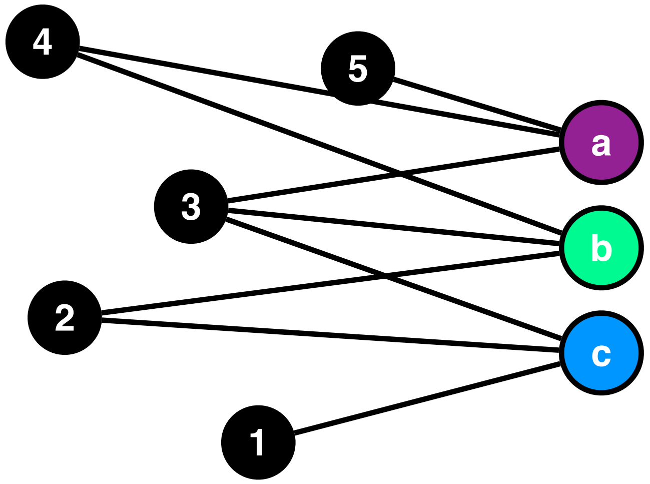
edges connect 2 nodes

Hypergraph



hyperedges connect ≥ 1 nodes

Bipartite graph



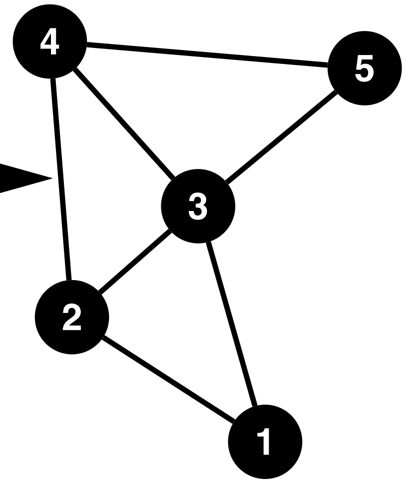
Adjacency matrix

$$\begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 1 & 1 & 1 & 0 & 0 \\ 2 & 1 & 1 & 1 & 1 & 0 \\ 3 & 1 & 1 & 1 & 1 & 1 \\ 4 & 0 & 1 & 1 & 1 & 1 \\ 5 & 0 & 0 & 1 & 1 & 1 \end{array}$$

$(N \times N)$

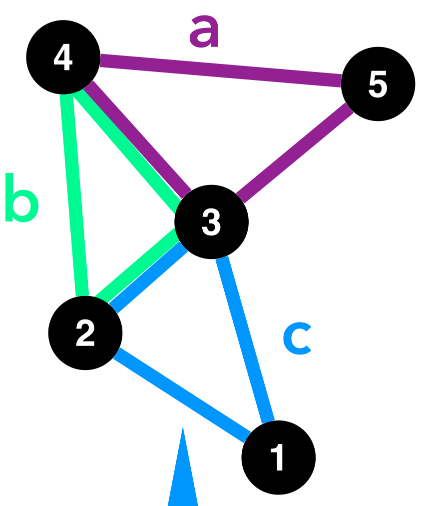
What is a hypergraph?

Graph



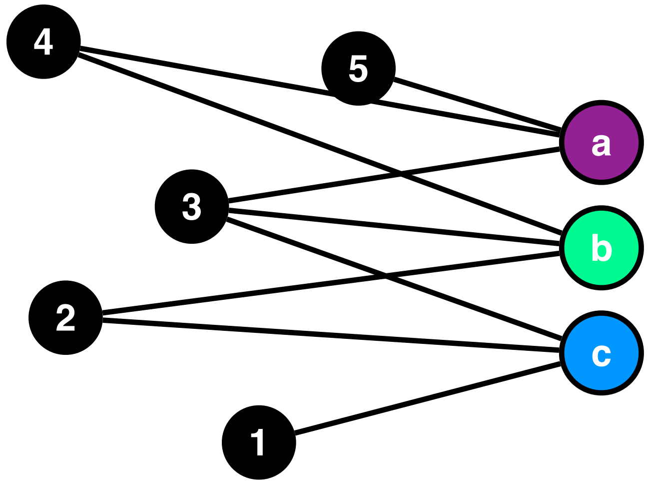
edges connect 2 nodes

Hypergraph



hyperedges connect ≥ 1 nodes

Bipartite graph



Adjacency matrix

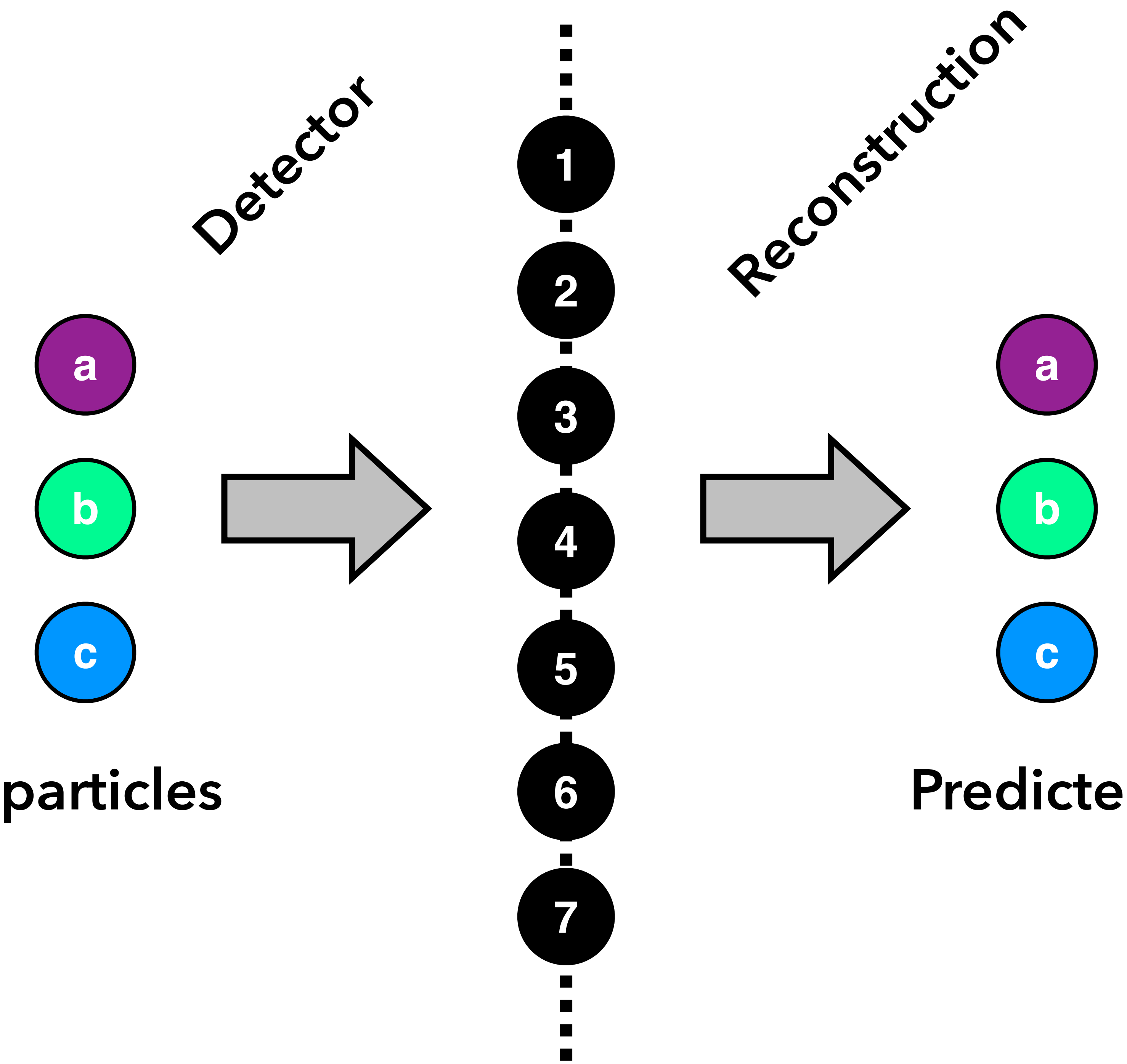
$$\begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 1 & 1 & 1 & 0 & 0 \\ 2 & 1 & 1 & 1 & 1 & 0 \\ 3 & 1 & 1 & 1 & 1 & 1 \\ 4 & 0 & 1 & 1 & 1 & 1 \\ 5 & 0 & 0 & 1 & 1 & 1 \end{array} \quad (N \times N)$$

Incidence matrix

$$\begin{array}{c|ccc} & a & b & c \\ \hline 1 & 0 & 0 & 1 \\ 2 & 0 & 1 & 1 \\ 3 & 1 & 1 & 1 \\ 4 & 1 & 1 & 0 \\ 5 & 1 & 0 & 0 \end{array} \quad (N \times H)$$

Why use a hypergraph?

cartoon from Nilotpal Kakati



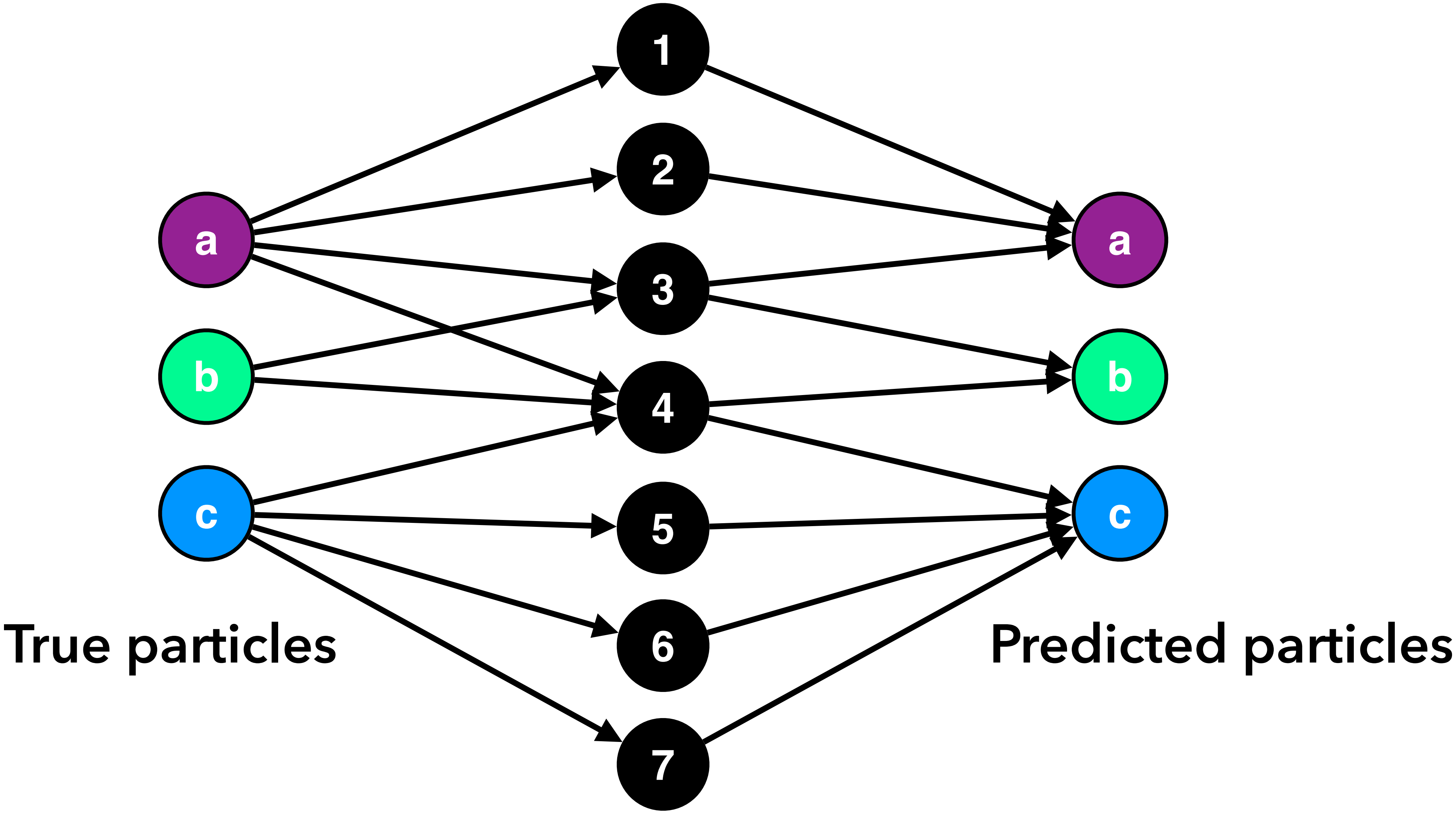
True particles

Predicted particles

Topoclusters and tracks

Why use a hypergraph?

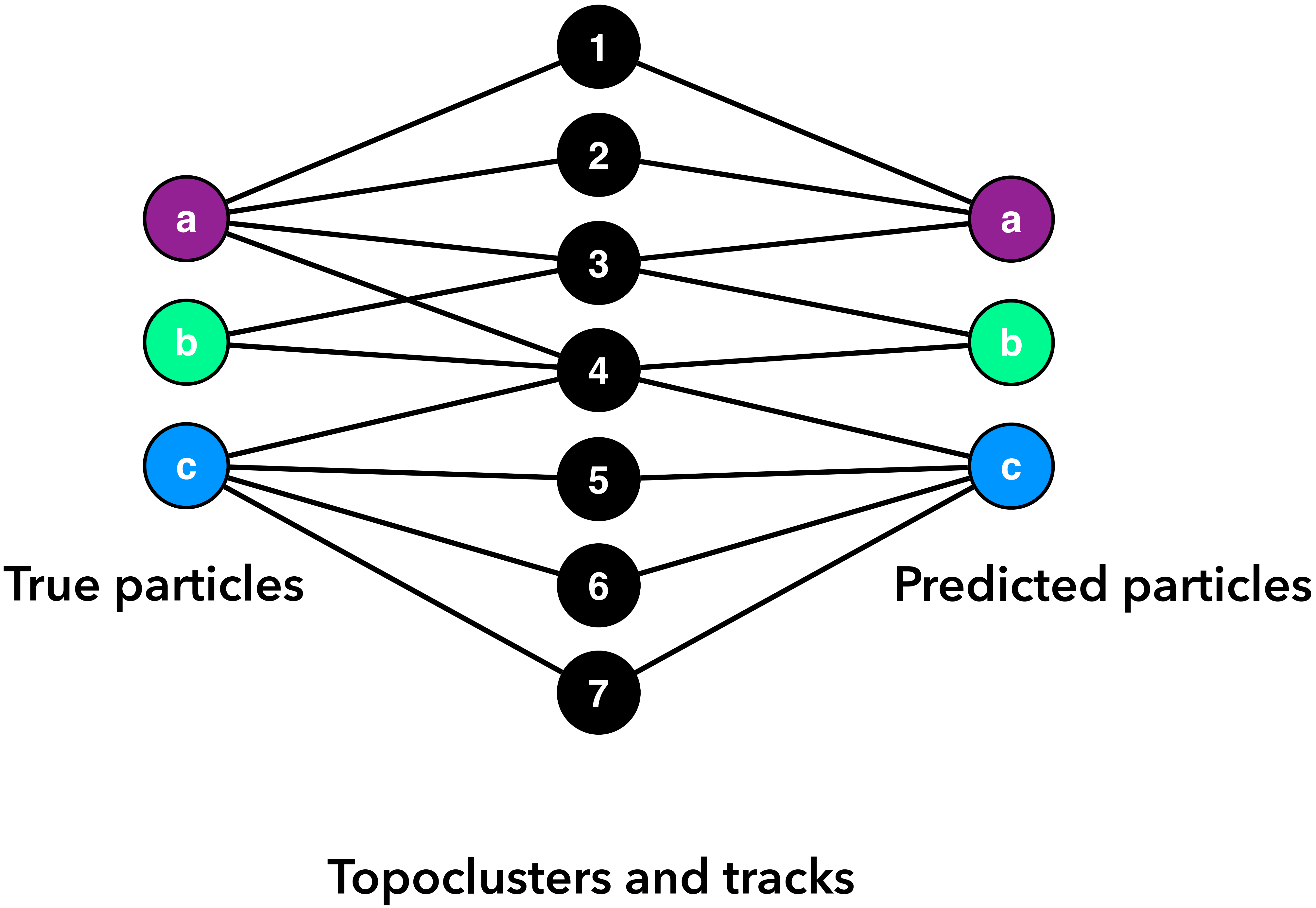
cartoon from Nilotpal Kakati



Topoclusters and tracks

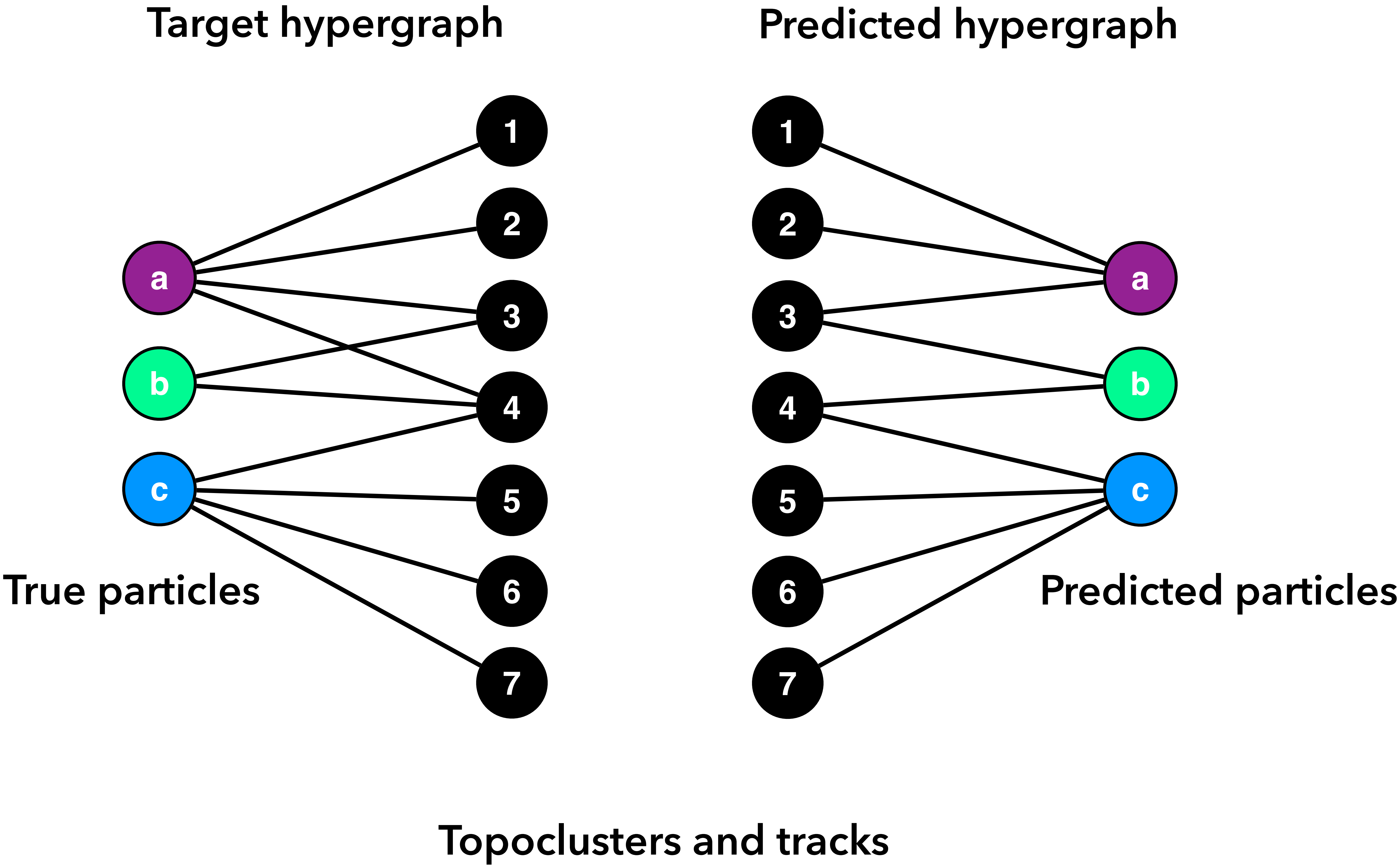
Why use a hypergraph?

cartoon from Nilotpal Kakati

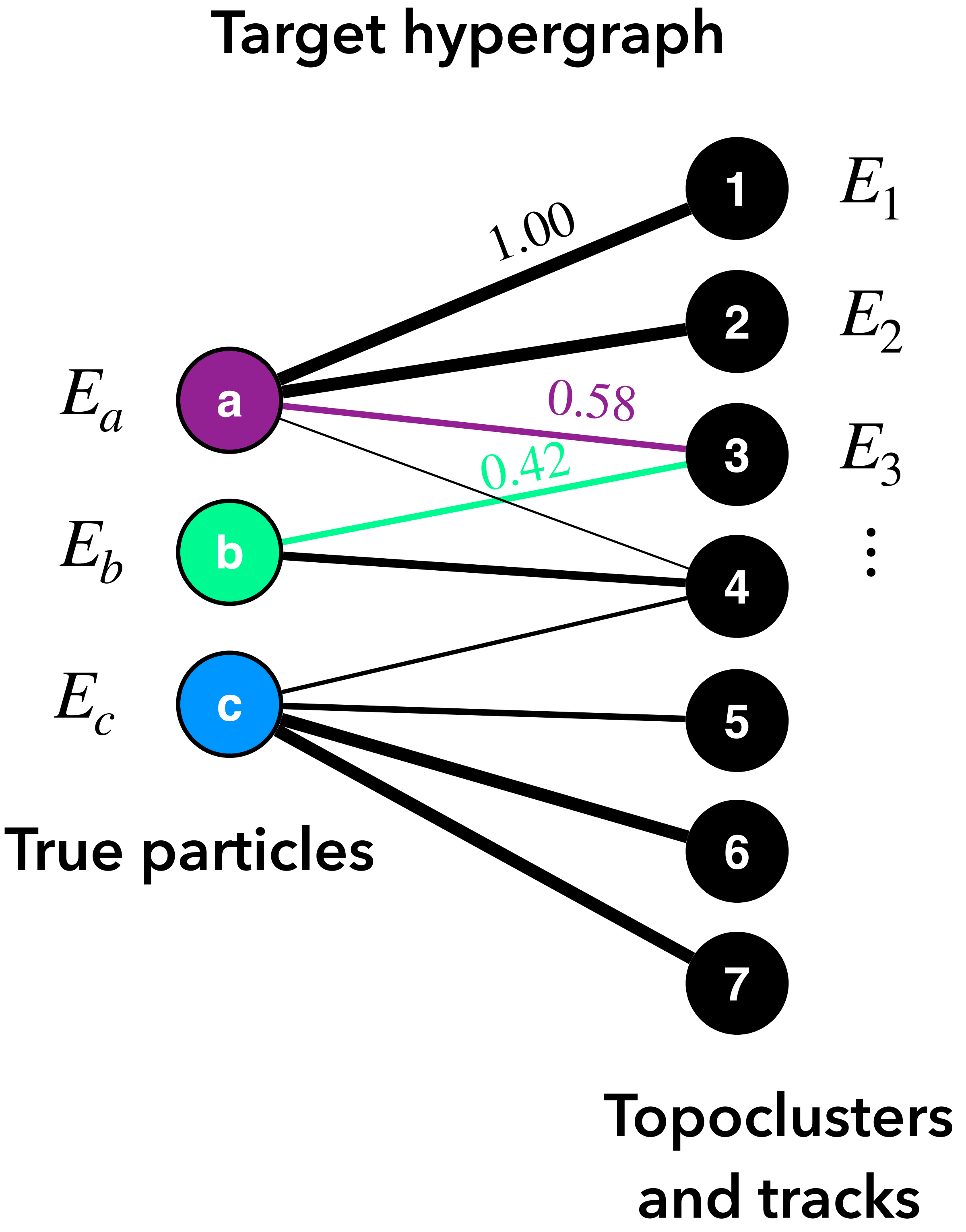


Why use a hypergraph?

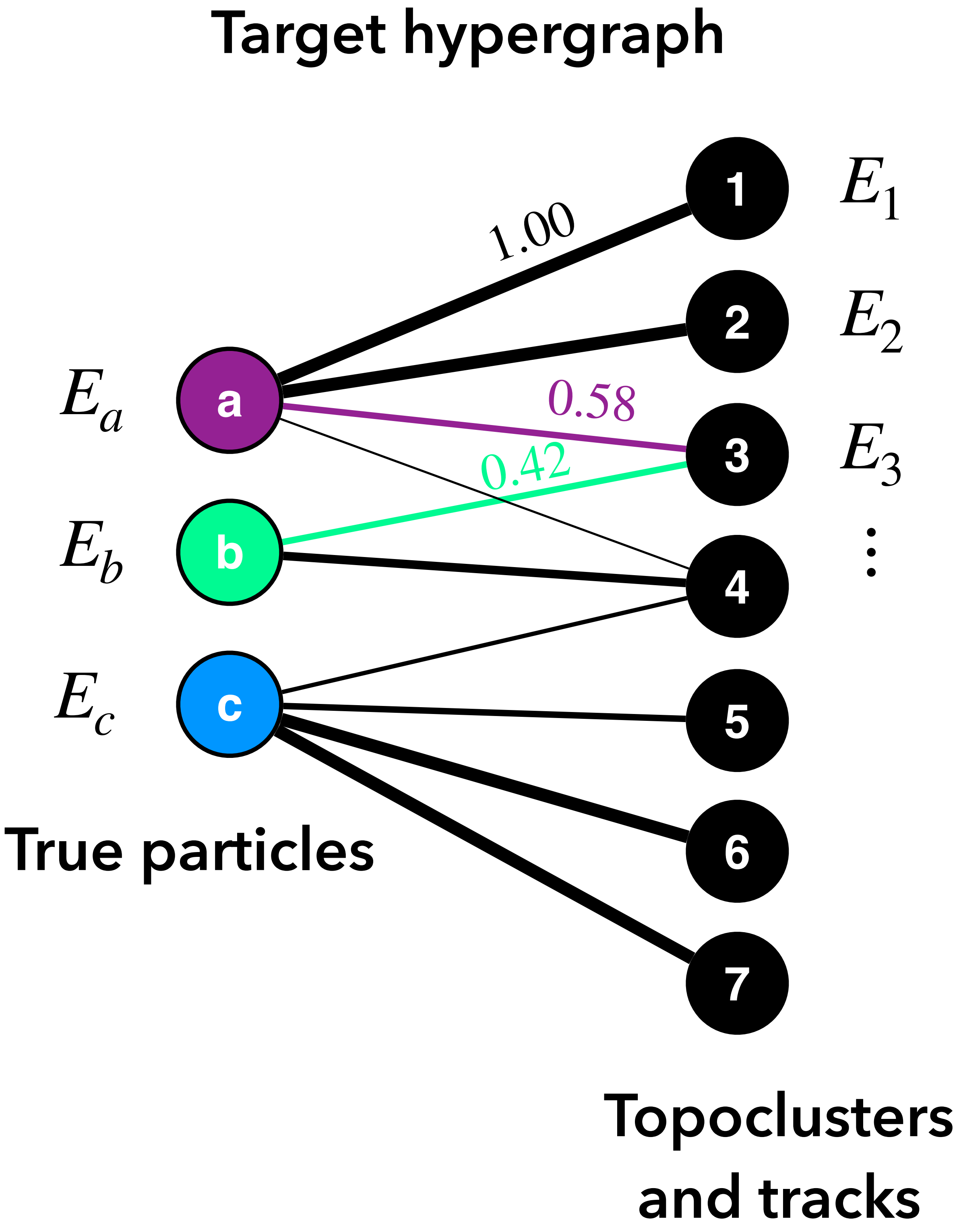
cartoon from Nilotpal Kakati



Why use a hypergraph?



Why use a hypergraph?

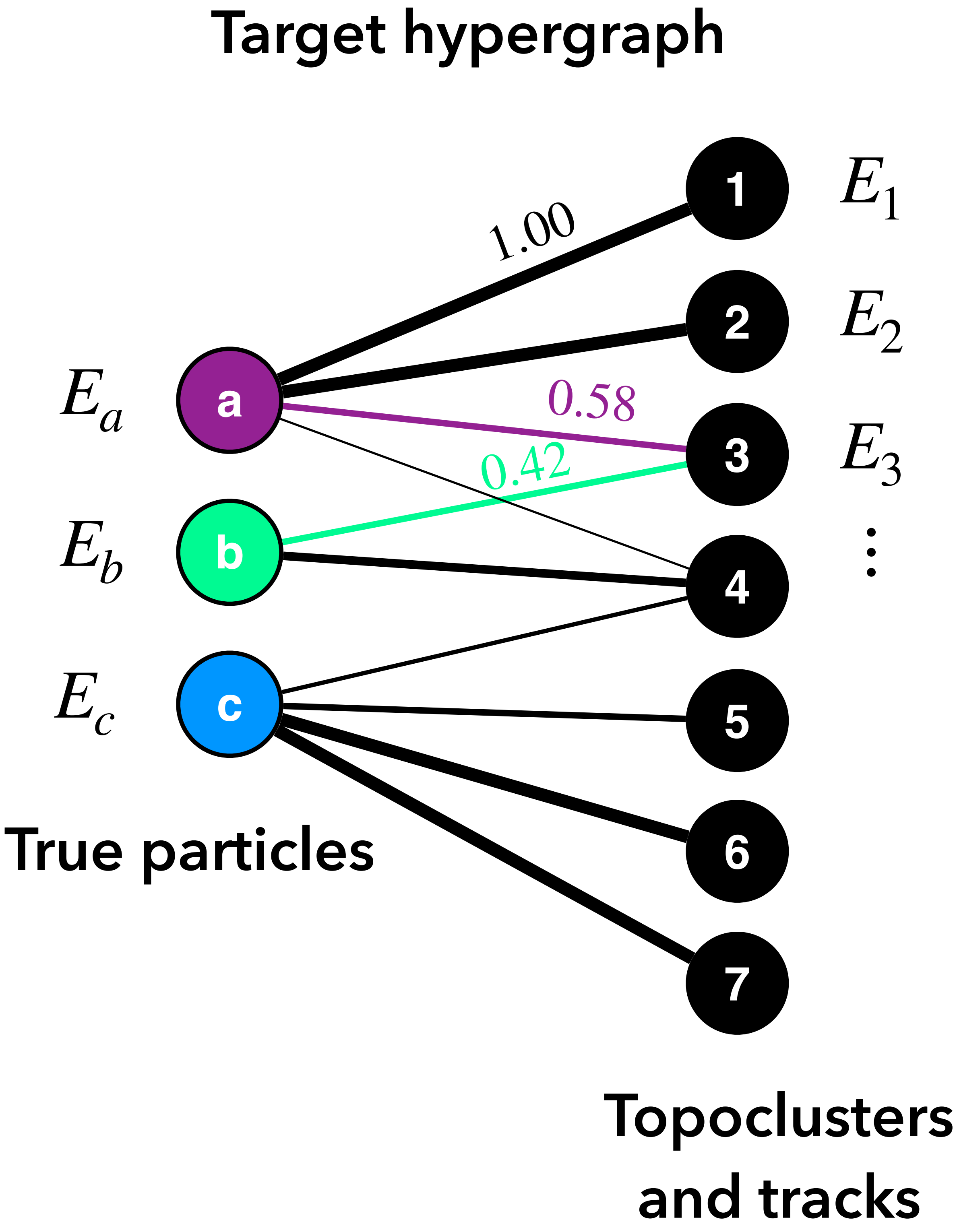


Physically-interpretable incidence matrix

$$[I]_{ia} = \frac{E_{ia}}{E_i}$$

⇒ *fraction of topocluster i energy contributed by particle a*

Why use a hypergraph?



Physically-interpretable incidence matrix

$$[I]_{ia} = \frac{E_{ia}}{E_i}$$

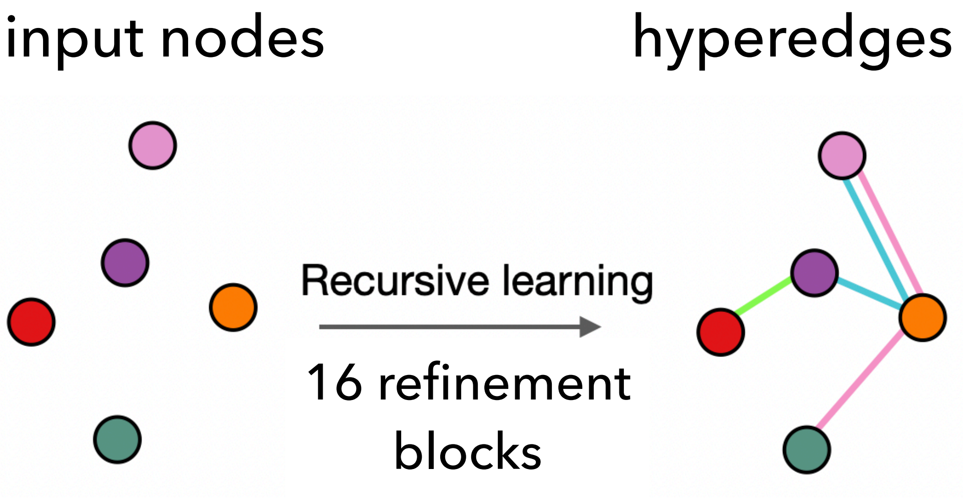
⇒ *fraction of topocluster i energy contributed by particle a*

Advantages

- ✳ Interpretability
- ✳ Biased toward E cons.
- ✳ Can approx. particle energy as incidence-weighted sum of node energies

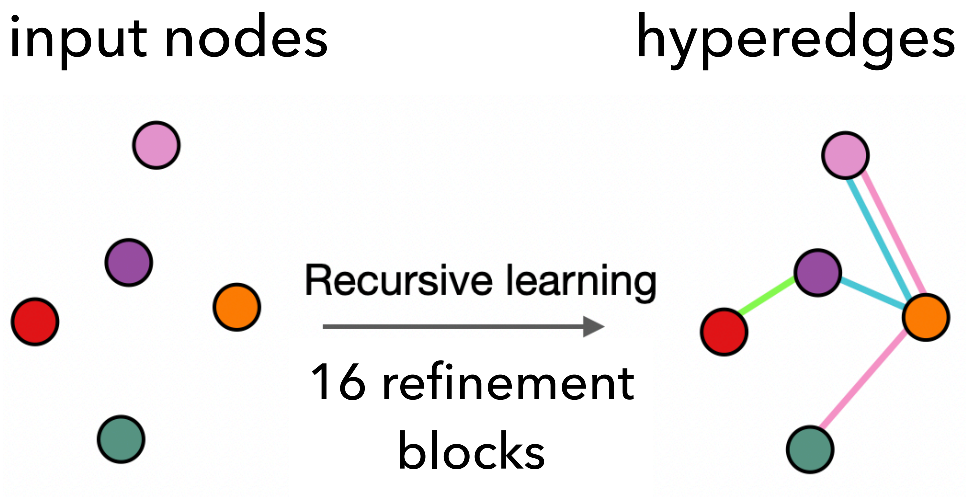
HGPflow algorithm

1) predict incidence matrix



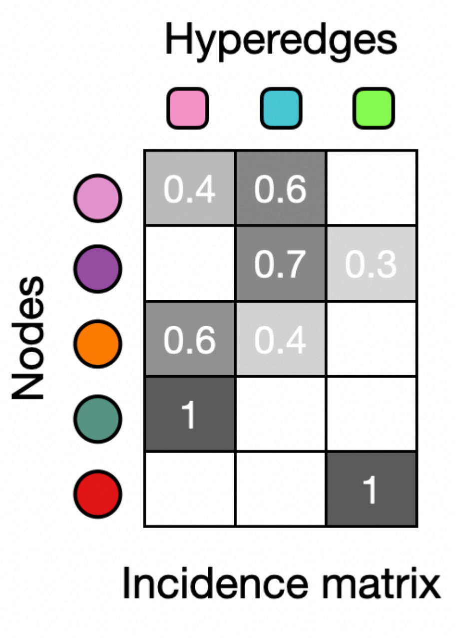
HGPflow algorithm

1) predict incidence matrix



Energy-fraction incidence matrix

$$I_{ia} =$$

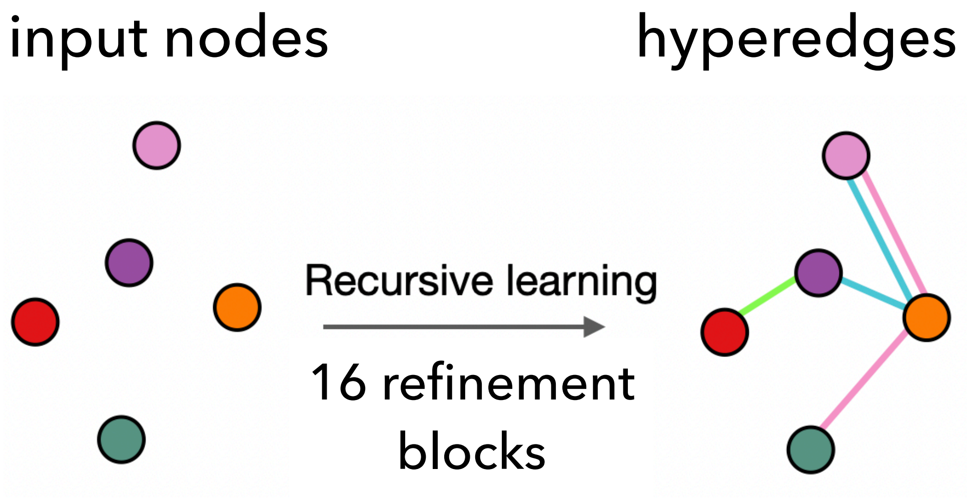


	■	■	■
●	0.4	0.6	
●		0.7	0.3
●	0.6	0.4	
●	1		
●			1

Recurrently predicting hypergraphs
arXiv:2106.13919

HGPflow algorithm

1) predict incidence matrix

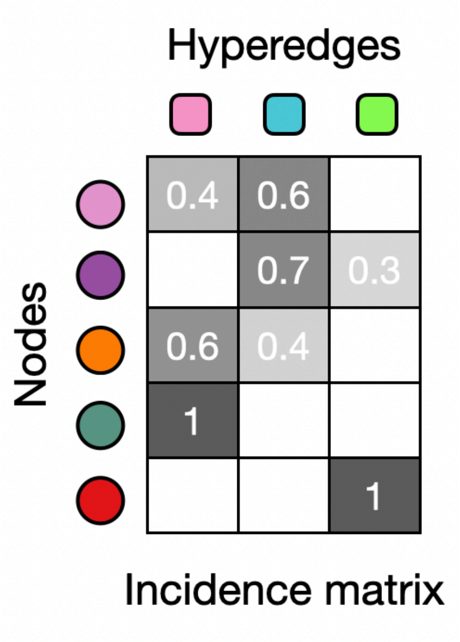


Recursive learning

16 refinement blocks

Energy-fraction incidence matrix

$$I_{ia} =$$



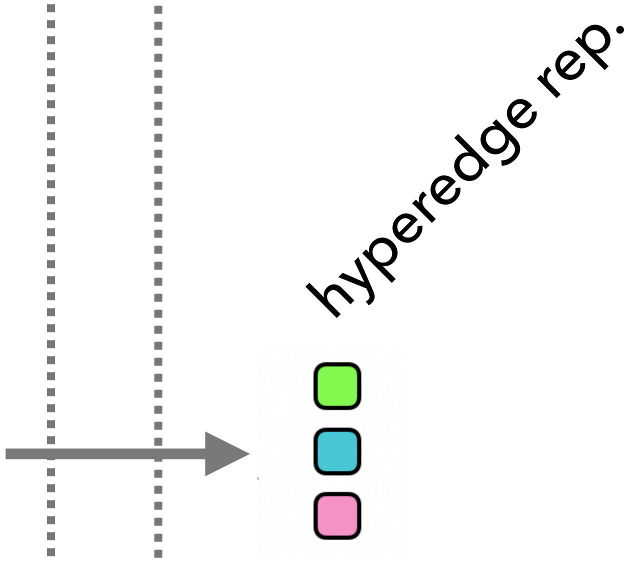
Hyperedges

	🟪	🟦	🟩
🟣	0.4	0.6	
🟣		0.7	0.3
🟠	0.6	0.4	
🟢	1		
🔴			1

Incidence matrix

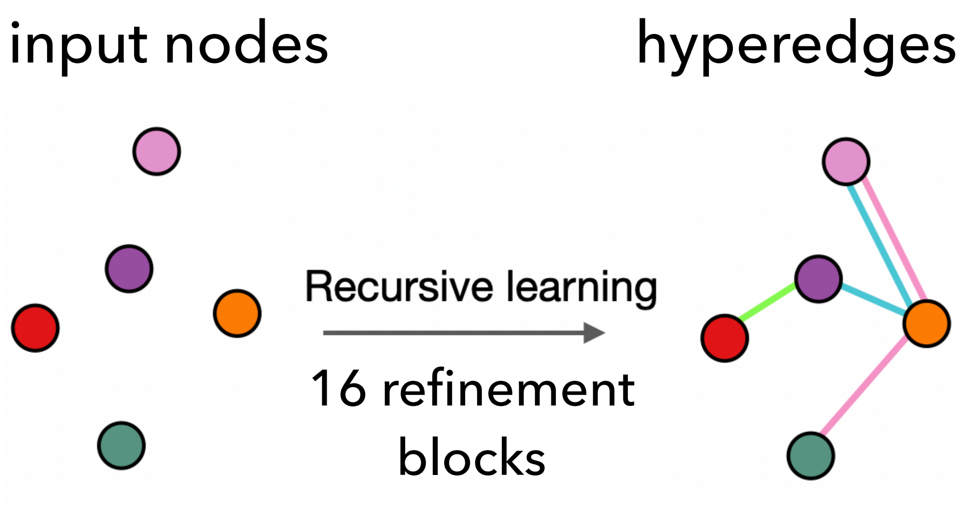
Recurrently predicting hypergraphs
arXiv:2106.13919

2) predict particle properties



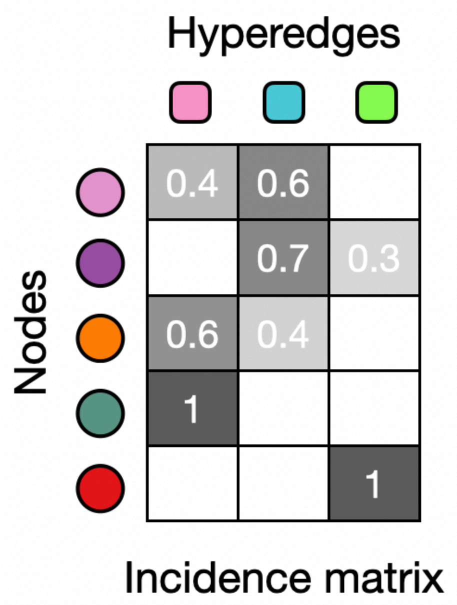
HGPflow algorithm

1) predict incidence matrix



Energy-fraction incidence matrix

$$I_{ia} =$$

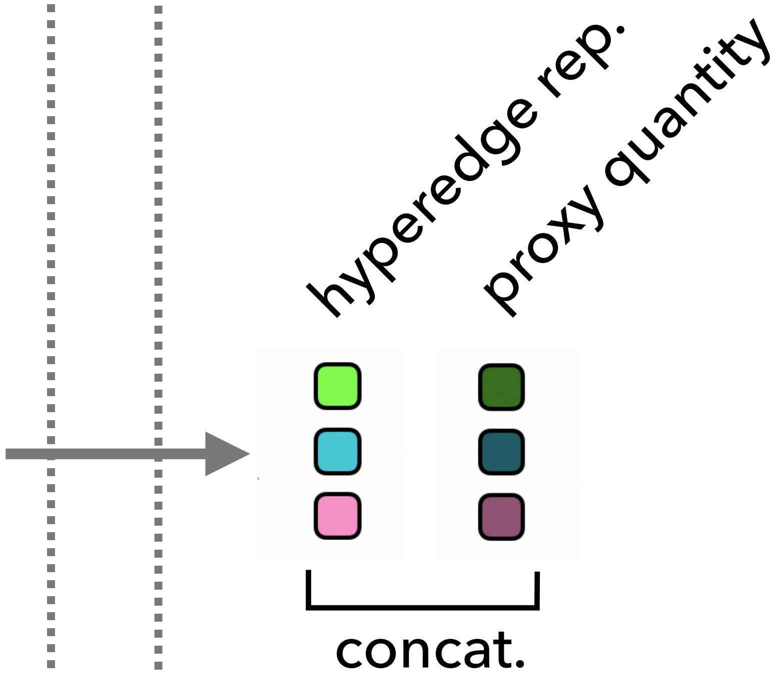


0.4	0.6	
	0.7	0.3
0.6	0.4	
1		
		1

Recurrently predicting hypergraphs

arXiv:2106.13919

2) predict particle properties



Energy-weighted proxy quantities:

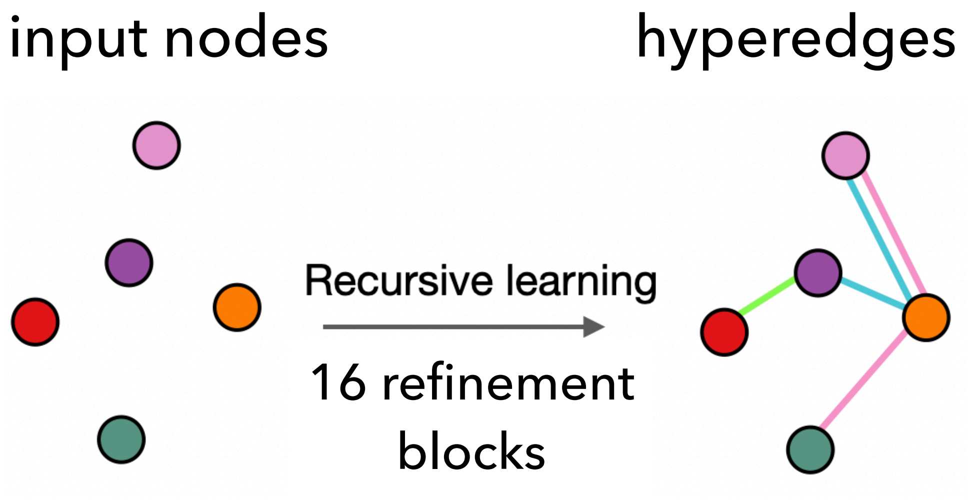
$$\hat{E}_a = \sum_{\text{nodes } i} E_i \cdot I_{ia}$$

$$\hat{\eta}_a = \sum_{\text{nodes } i} \eta_i \cdot \frac{E_{ia}}{E_a}$$

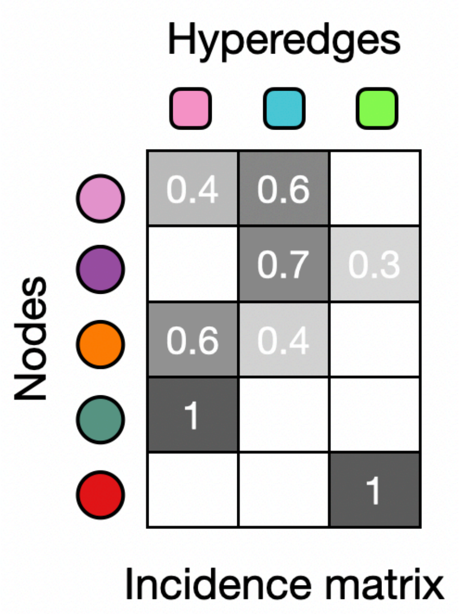
$$\hat{\phi}_a = \sum_{\text{nodes } i} \phi_i \cdot \frac{E_{ia}}{E_a}$$

HGPflow algorithm

1) predict incidence matrix



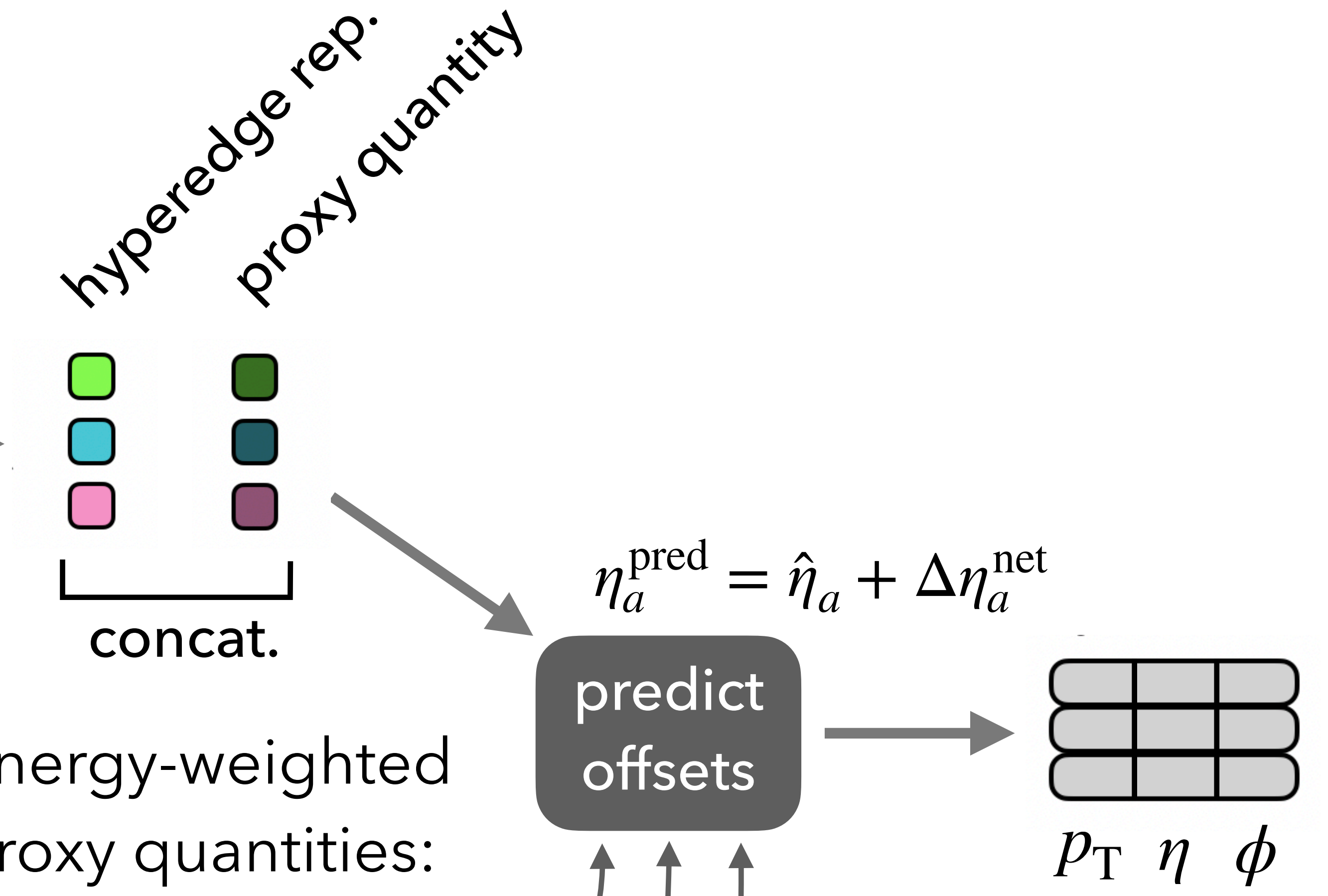
Energy-fraction incidence matrix



$$I_{ia} = \begin{array}{|c|c|c|} \hline 0.4 & 0.6 & \\ \hline & 0.7 & 0.3 \\ \hline 0.6 & 0.4 & \\ \hline 1 & & \\ \hline & & 1 \\ \hline \end{array}$$

Recurrently predicting hypergraphs
arXiv:2106.13919

2) predict particle properties



$$\eta_a^{\text{pred}} = \hat{\eta}_a + \Delta\eta_a^{\text{net}}$$

Energy-weighted proxy quantities:

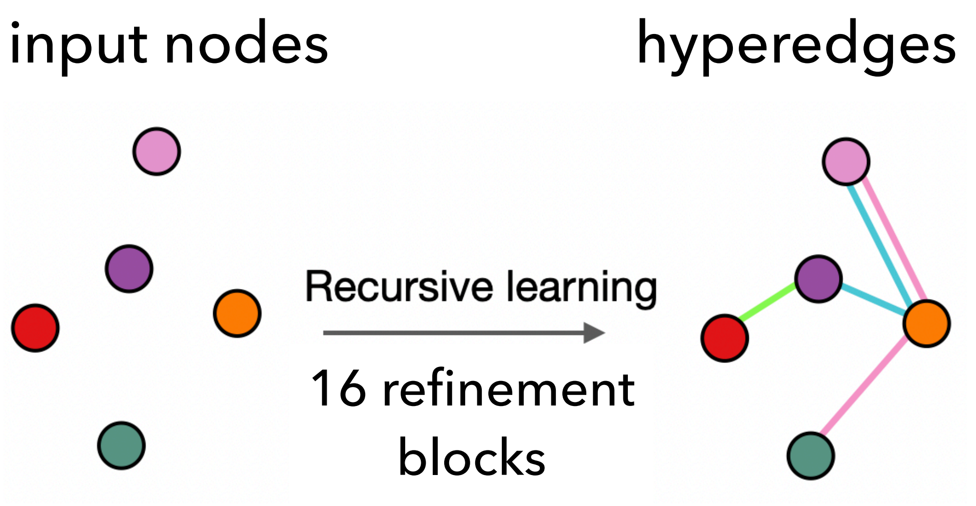
$$\hat{E}_a = \sum_{\text{nodes } i} E_i \cdot I_{ia}$$

$$\hat{\eta}_a = \sum_{\text{nodes } i} \eta_i \cdot \frac{E_{ia}}{E_a}$$

$$\hat{\phi}_a = \sum_{\text{nodes } i} \phi_i \cdot \frac{E_{ia}}{E_a}$$

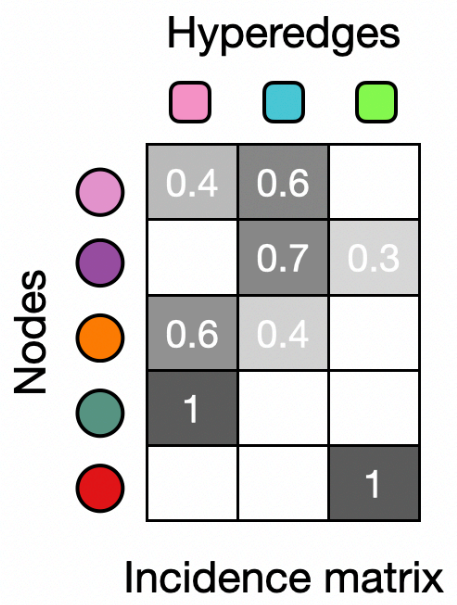
HGPflow algorithm

1) predict incidence matrix



Energy-fraction incidence matrix

$$I_{ia} =$$

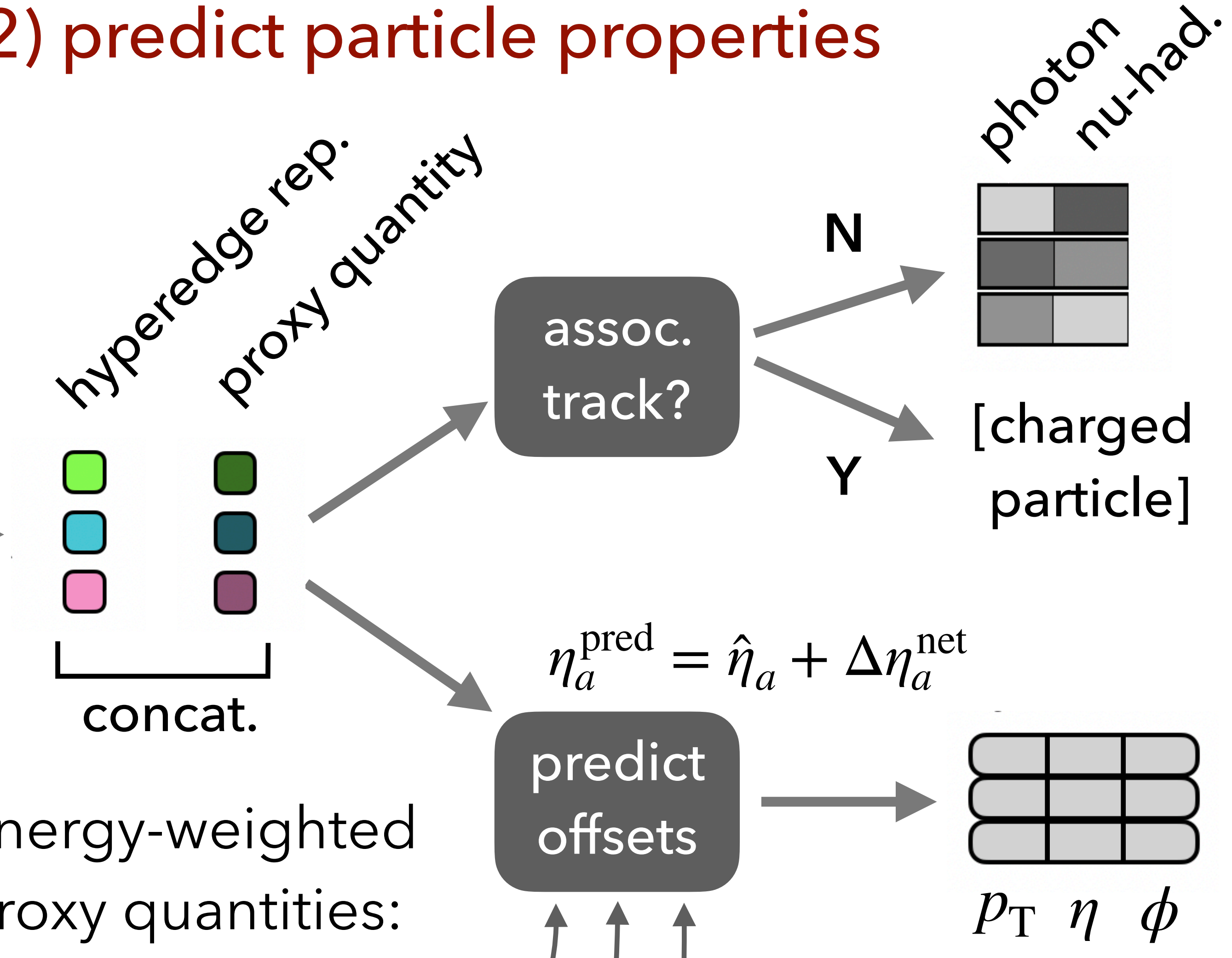


0.4	0.6	
	0.7	0.3
0.6	0.4	
1		
		1

Recurrently predicting hypergraphs

arXiv:2106.13919

2) predict particle properties



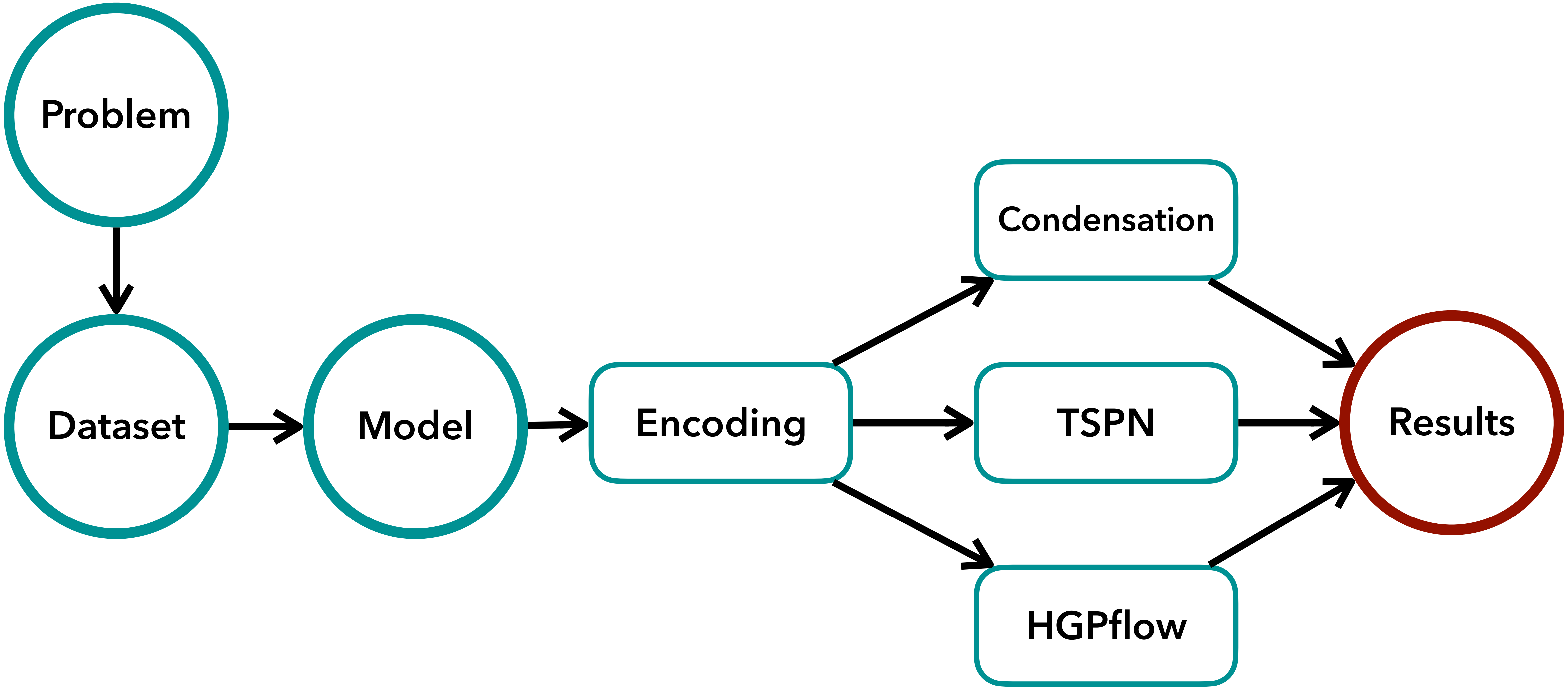
$$\eta_a^{\text{pred}} = \hat{\eta}_a + \Delta\eta_a^{\text{net}}$$

Energy-weighted proxy quantities:

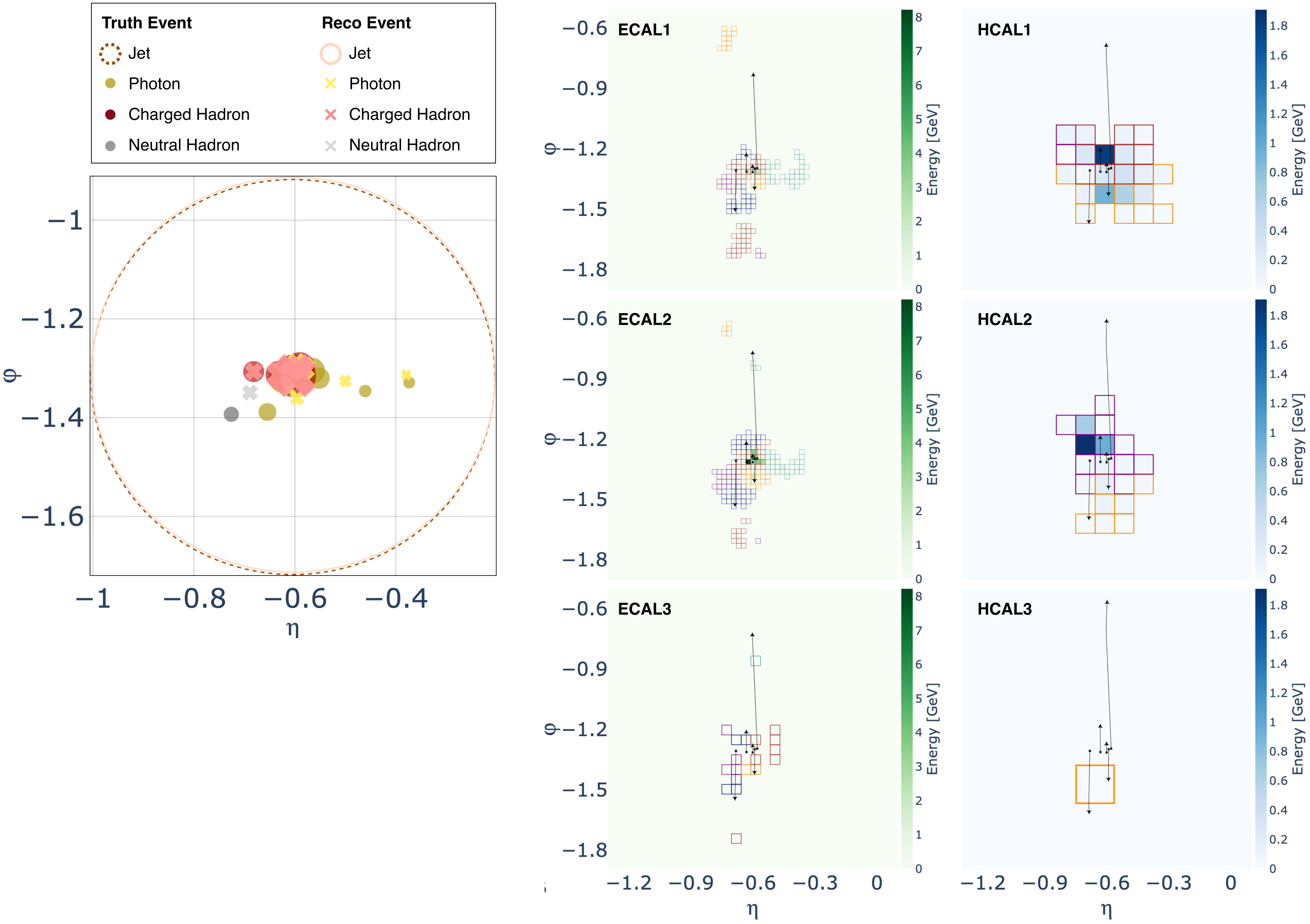
$$\hat{E}_a = \sum_{\text{nodes } i} E_i \cdot I_{ia}$$

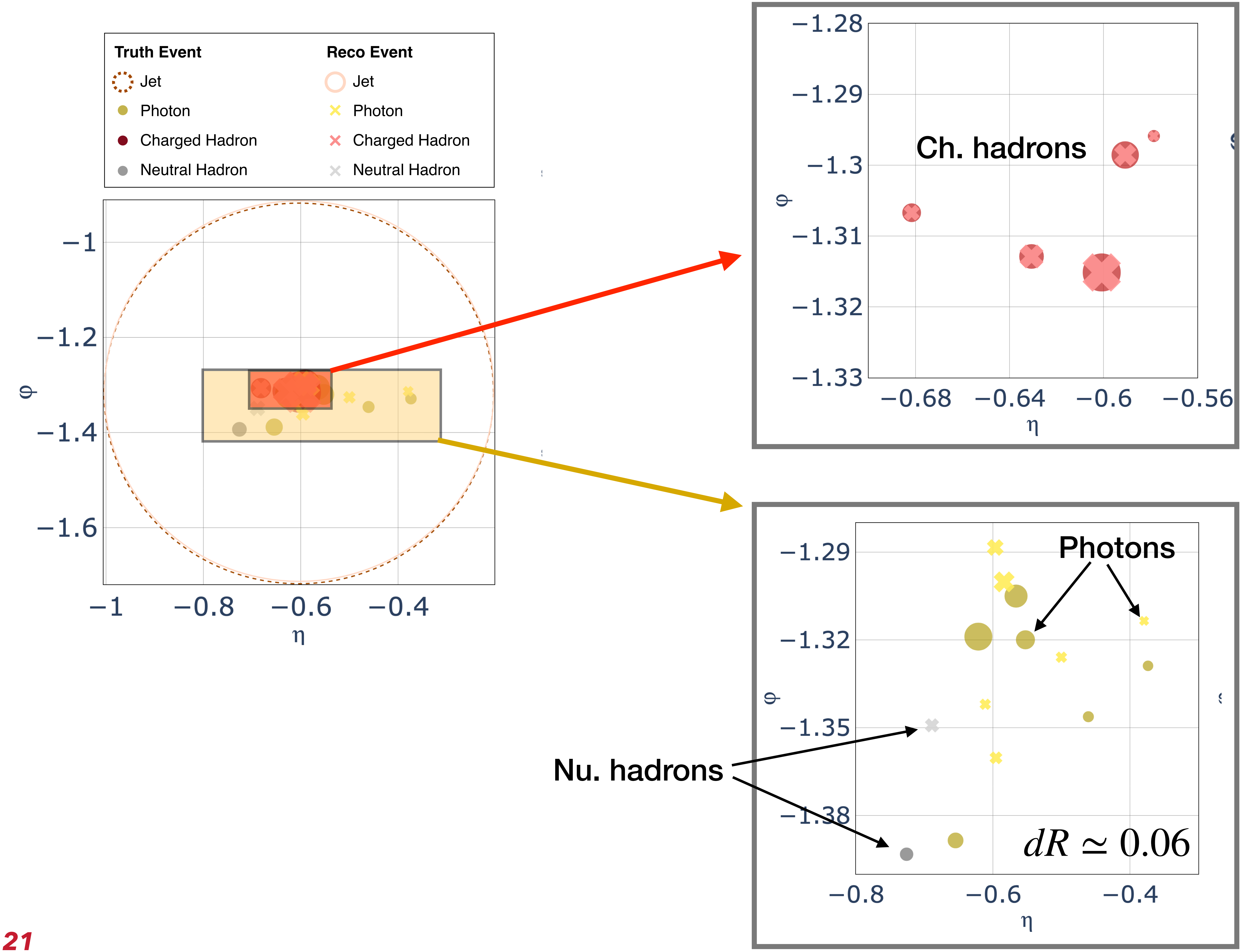
$$\hat{\eta}_a = \sum_{\text{nodes } i} \eta_i \cdot \frac{E_{ia}}{E_a}$$

$$\hat{\phi}_a = \sum_{\text{nodes } i} \phi_i \cdot \frac{E_{ia}}{E_a}$$



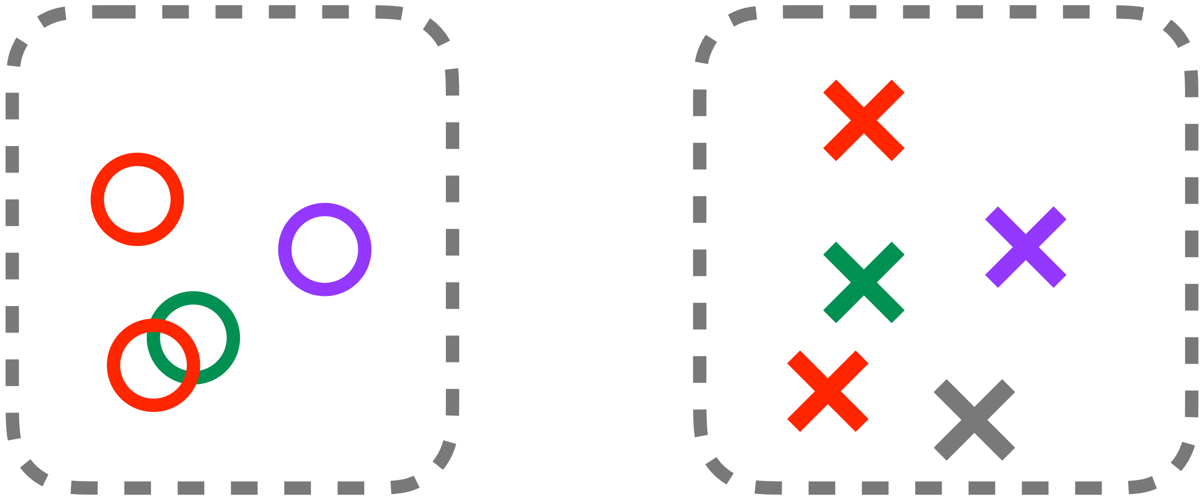
Problem
Dataset
Model
Encoding
Condensation
TSPN
HGPflow
Results


Truth Event
Jet
Photon
Charged Hadron
Neutral Hadron
Reco Event
Jet
Photon
Charged Hadron
Neutral Hadron
φ
η
−1
−1.2
−1.4
−1.6
−1
−0.8
−0.6
−0.4
ECAL1
ECAL2
ECAL3
HCAL1
HCAL2
HCAL3
−0.6
−0.9
−1.2
−1.5
−1.8
−1.2
−0.9
−0.6
−0.3
0
Energy [GeV]
0
1
2
3
4
5
6
7
8
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8


Truth Event
Jet
Photon
Charged Hadron
Neutral Hadron
Reco Event
Jet
Photon
Charged Hadron
Neutral Hadron
−1
−1.2
−1.4
−1.6
φ
−1
−0.8
−0.6
−0.4
η
−1.28
−1.29
−1.3
−1.31
−1.32
−1.33
φ
−0.68
−0.64
−0.6
−0.56
η
Ch. hadrons
−1.29
−1.32
−1.35
−1.38
φ
−0.8
−0.6
−0.4
η
Photons
Nu. hadrons
$dR \simeq 0.06$

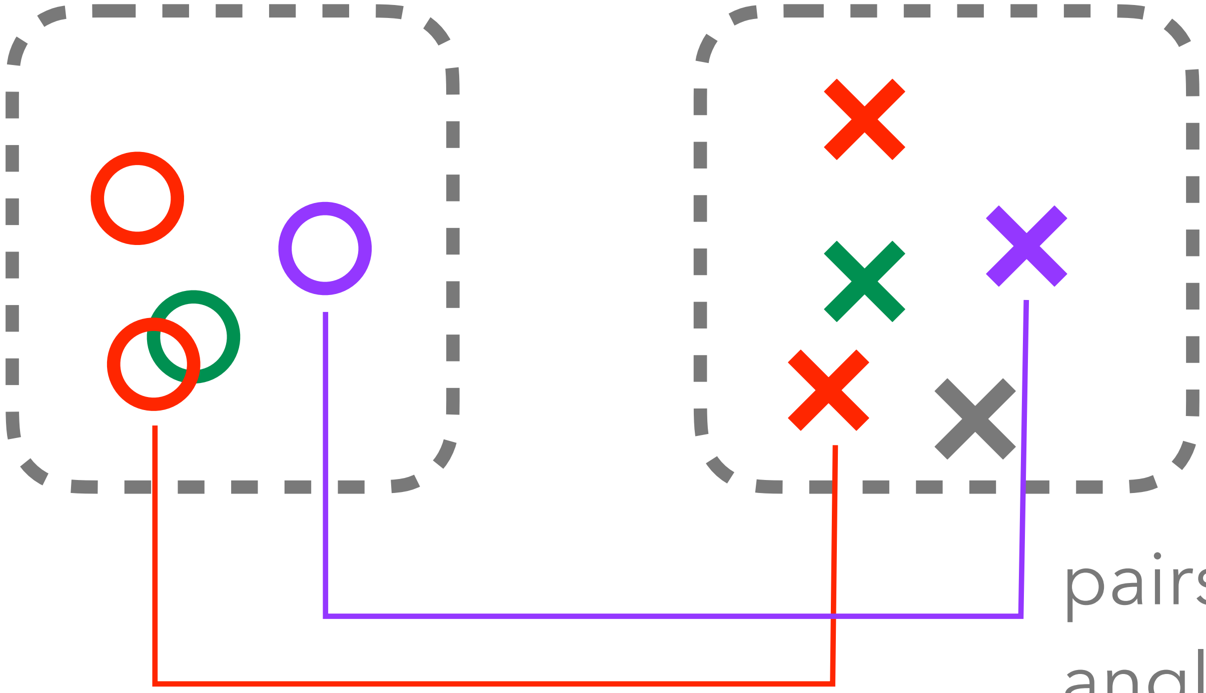
Comparing pred. vs target particles

1) Hungarian matching:



Comparing pred. vs target particles

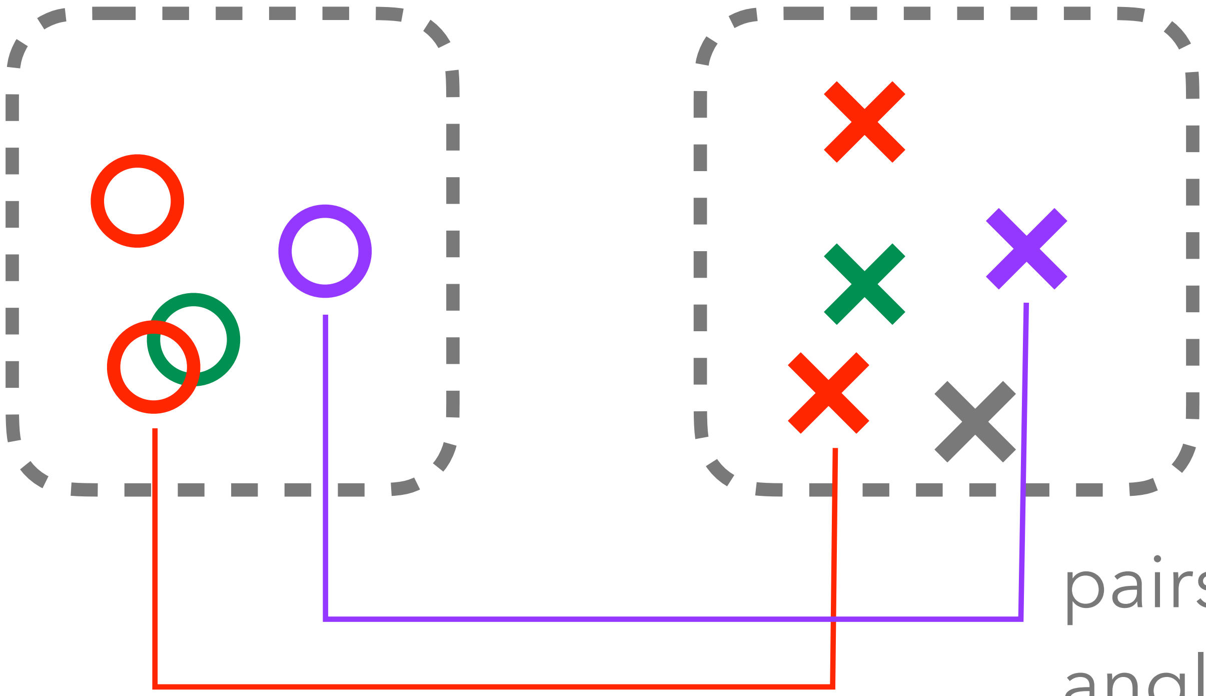
1) Hungarian matching:



pairs with best match in angle & momentum

Comparing pred. vs target particles

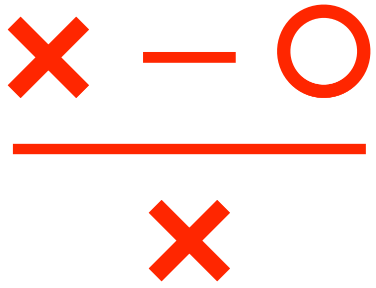
1) Hungarian matching:



pairs with best match in angle & momentum

2) Performance metrics

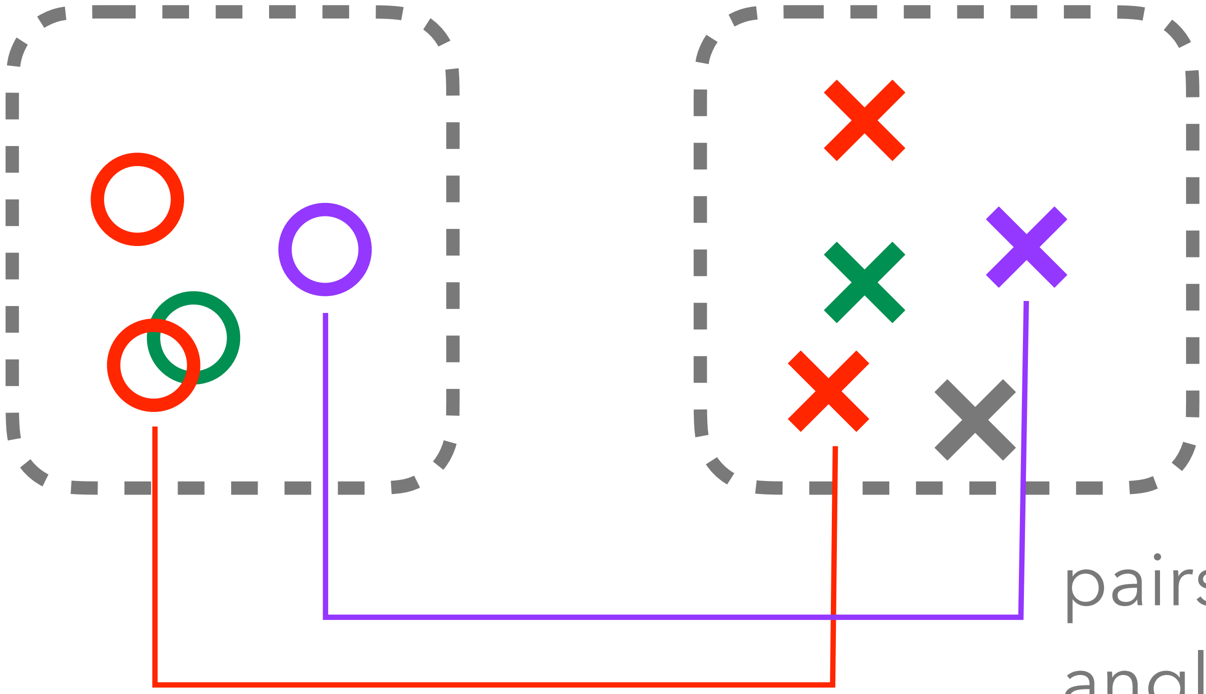
- Efficiency and fake rate
- Particle angular, momentum residuals



- Classification purity

Comparing pred. vs target particles

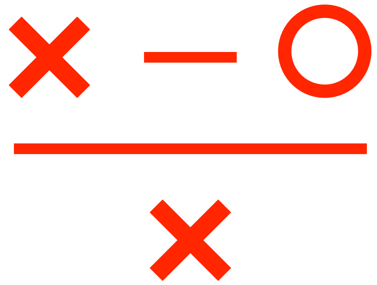
1) Hungarian matching:



pairs with best match in angle & momentum

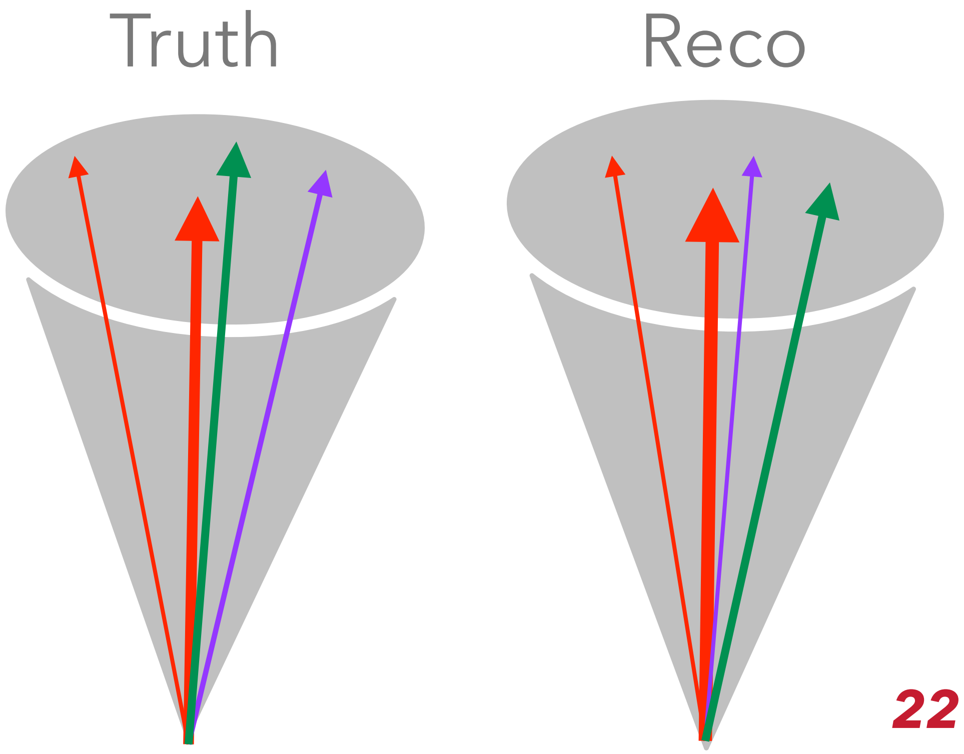
2) Performance metrics

- Efficiency and fake rate
- Particle angular, momentum residuals

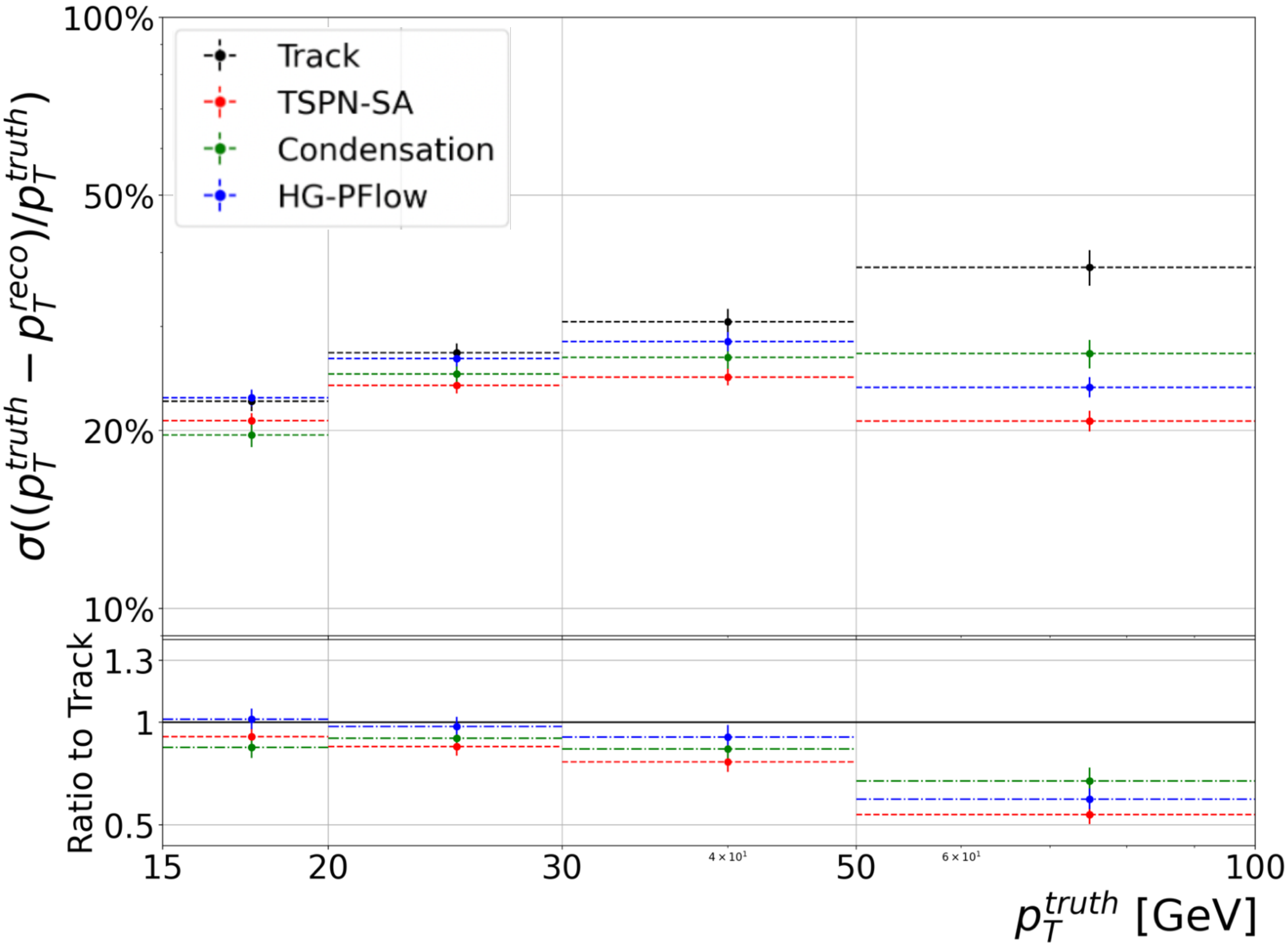


- Classification purity

- Jet-level quantities



Charged particle momentum

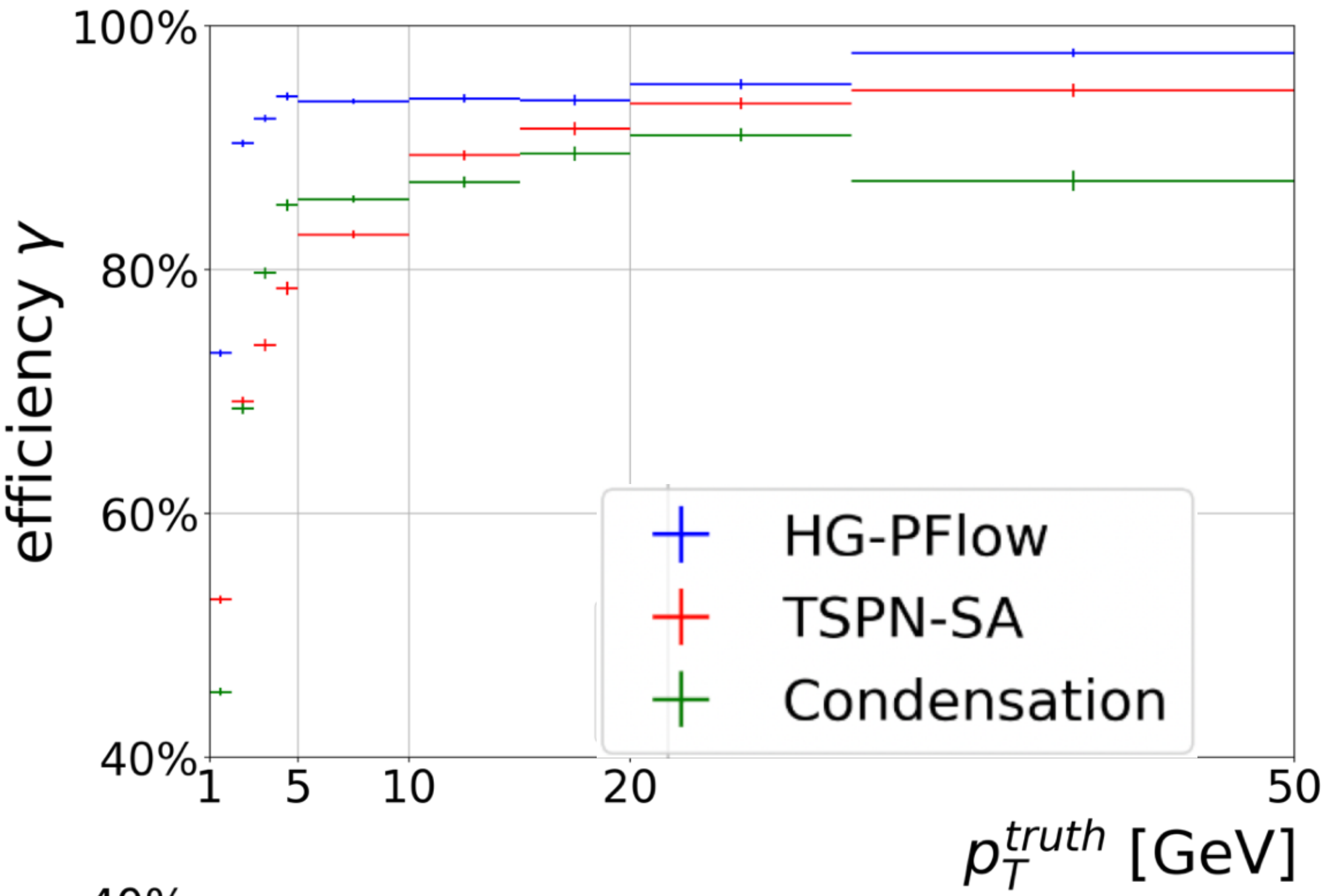


⇒ ML models exploit complementary information from calorimeter activity

Photon efficiency:

$$\epsilon = \frac{\#\text{matched targets}}{\#\text{total targets}}$$

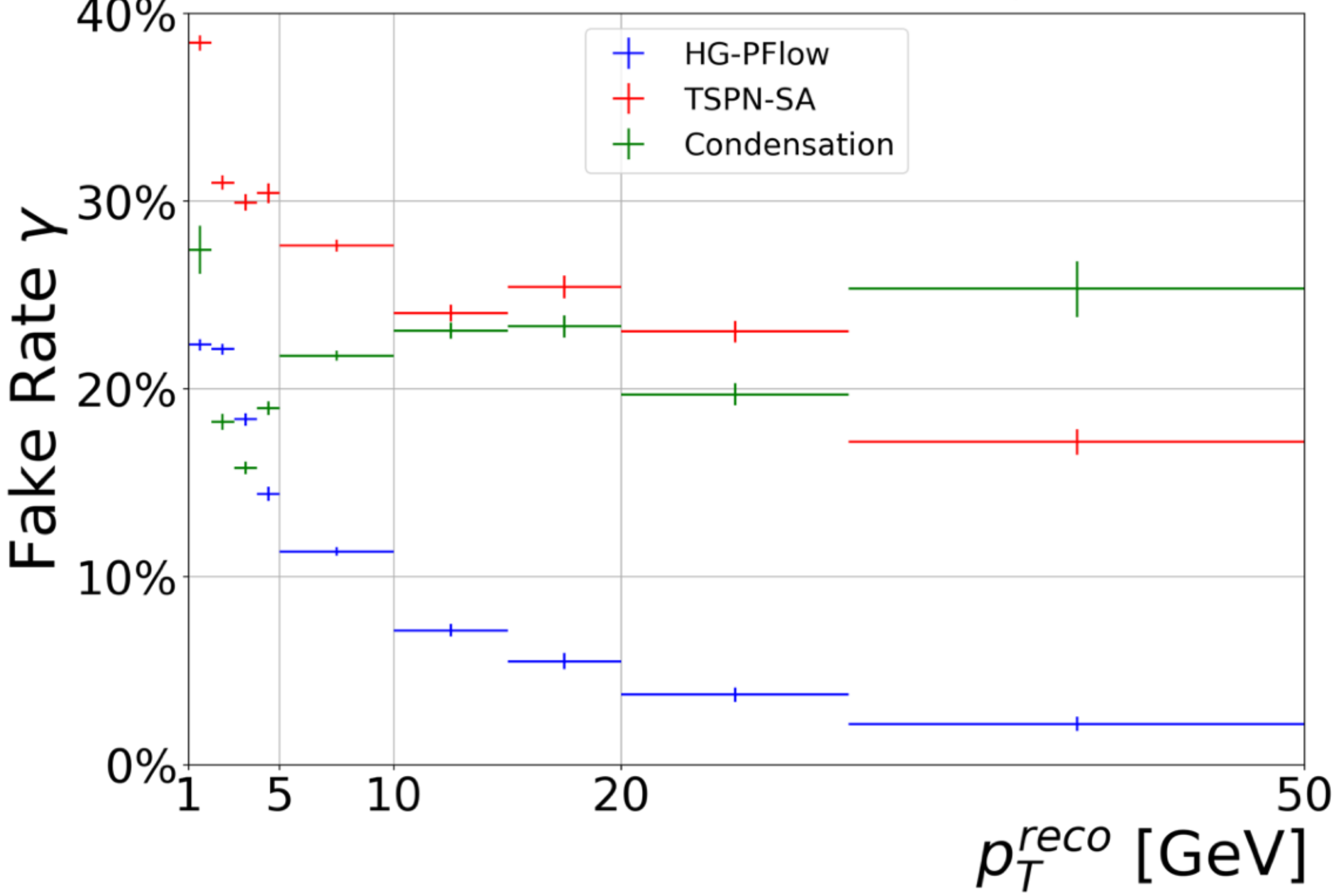
>90% above 2 GeV



Photon fake rate:

$$\epsilon = \frac{\#\text{unmatched predictions}}{\#\text{total predictions}}$$

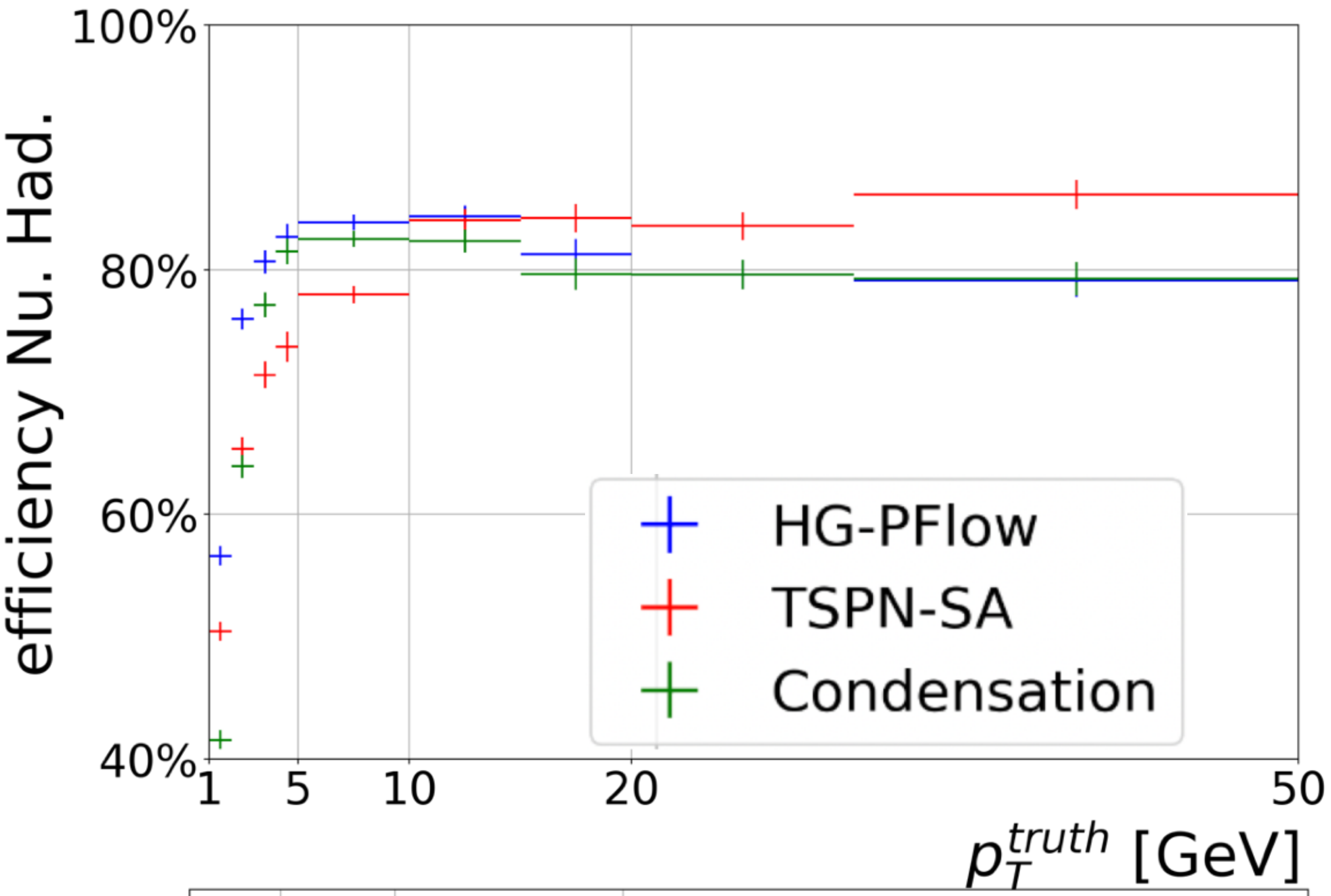
<5% above 20 GeV



Nu. Had. efficiency:

$$\epsilon = \frac{\#\text{matched targets}}{\#\text{total targets}}$$

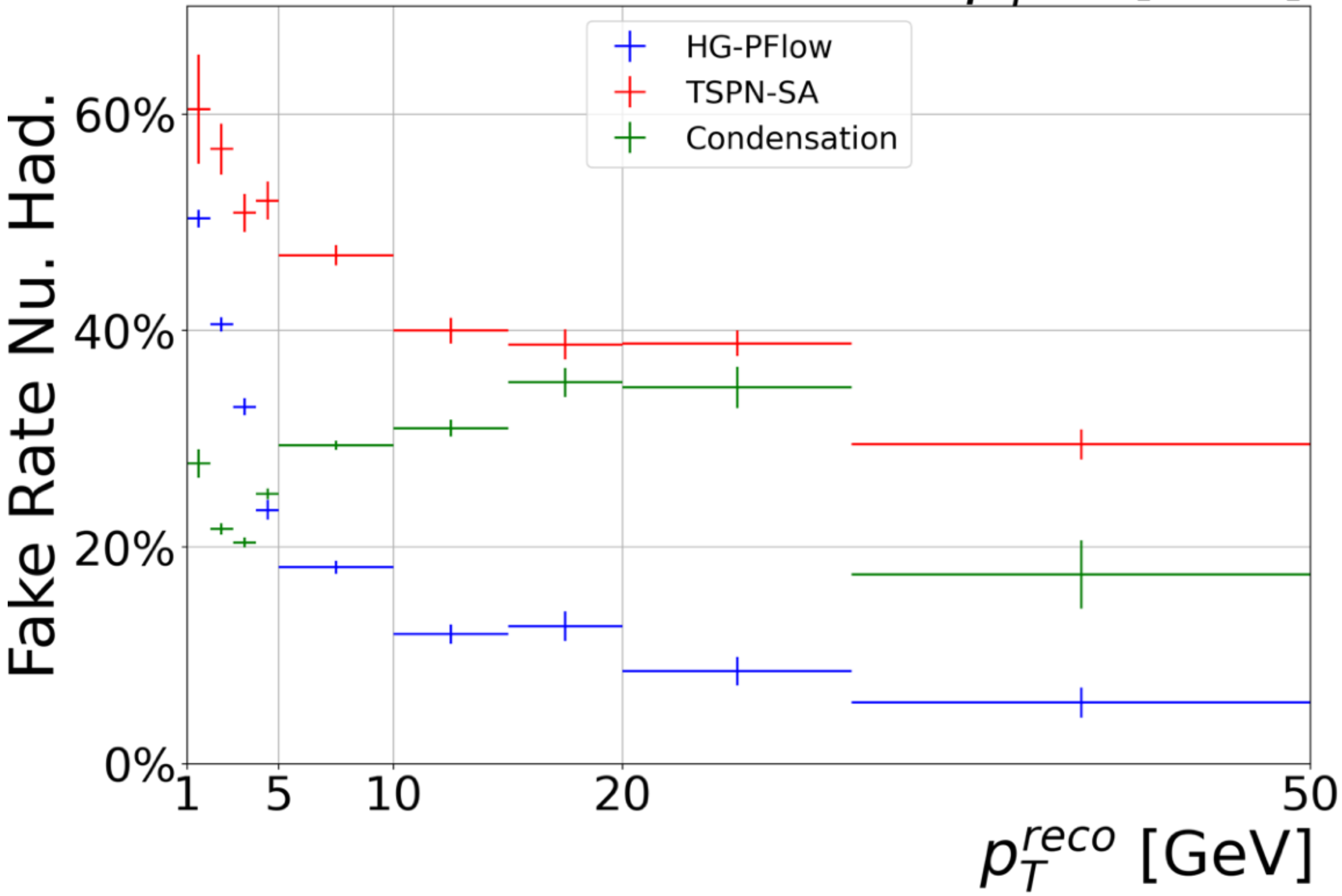
$\simeq$ 80% above 3 GeV



Nu. Had. fake rate:

$$\epsilon = \frac{\#\text{unmatched predictions}}{\#\text{total predictions}}$$

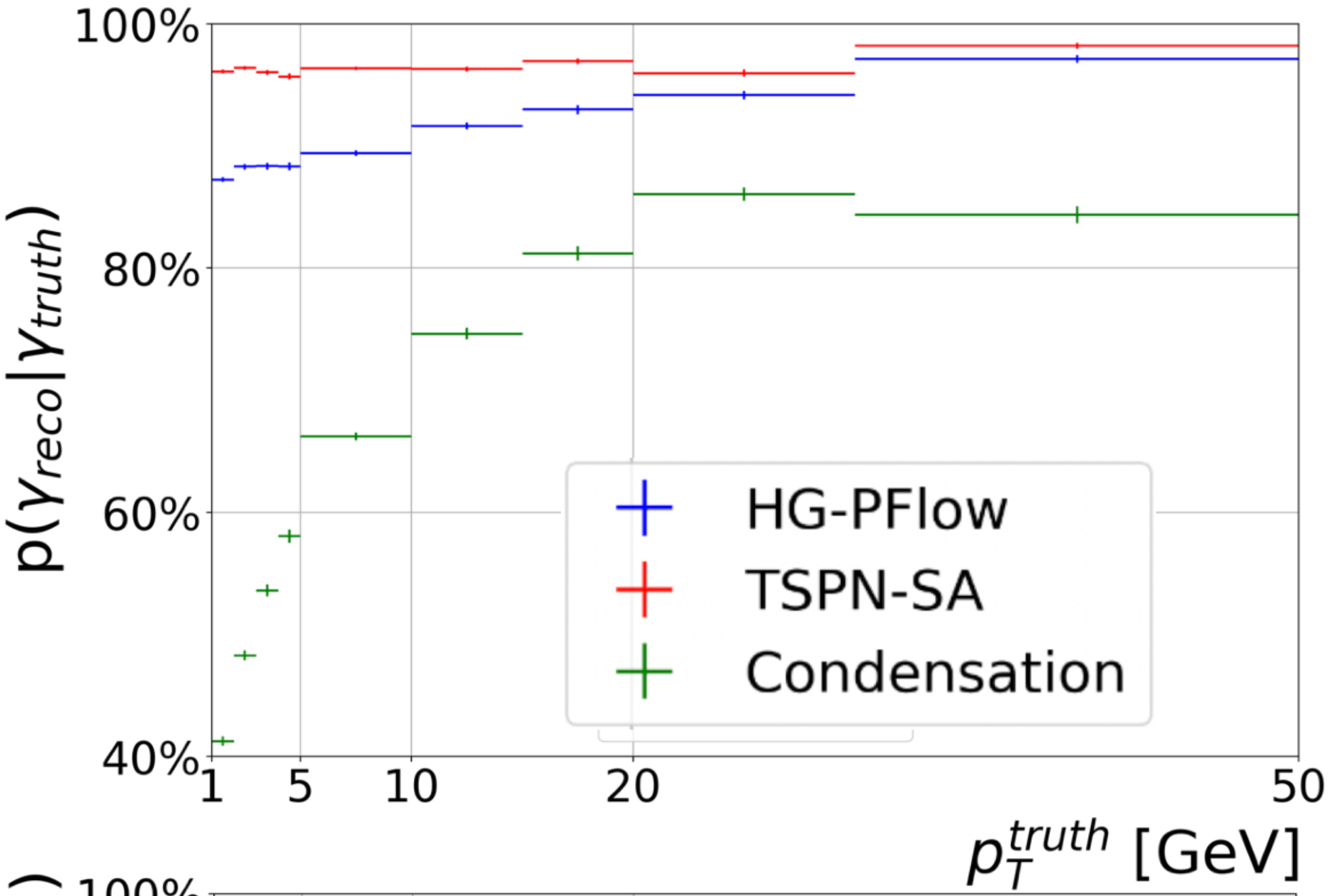
<10% above 20 GeV



Classification accuracy

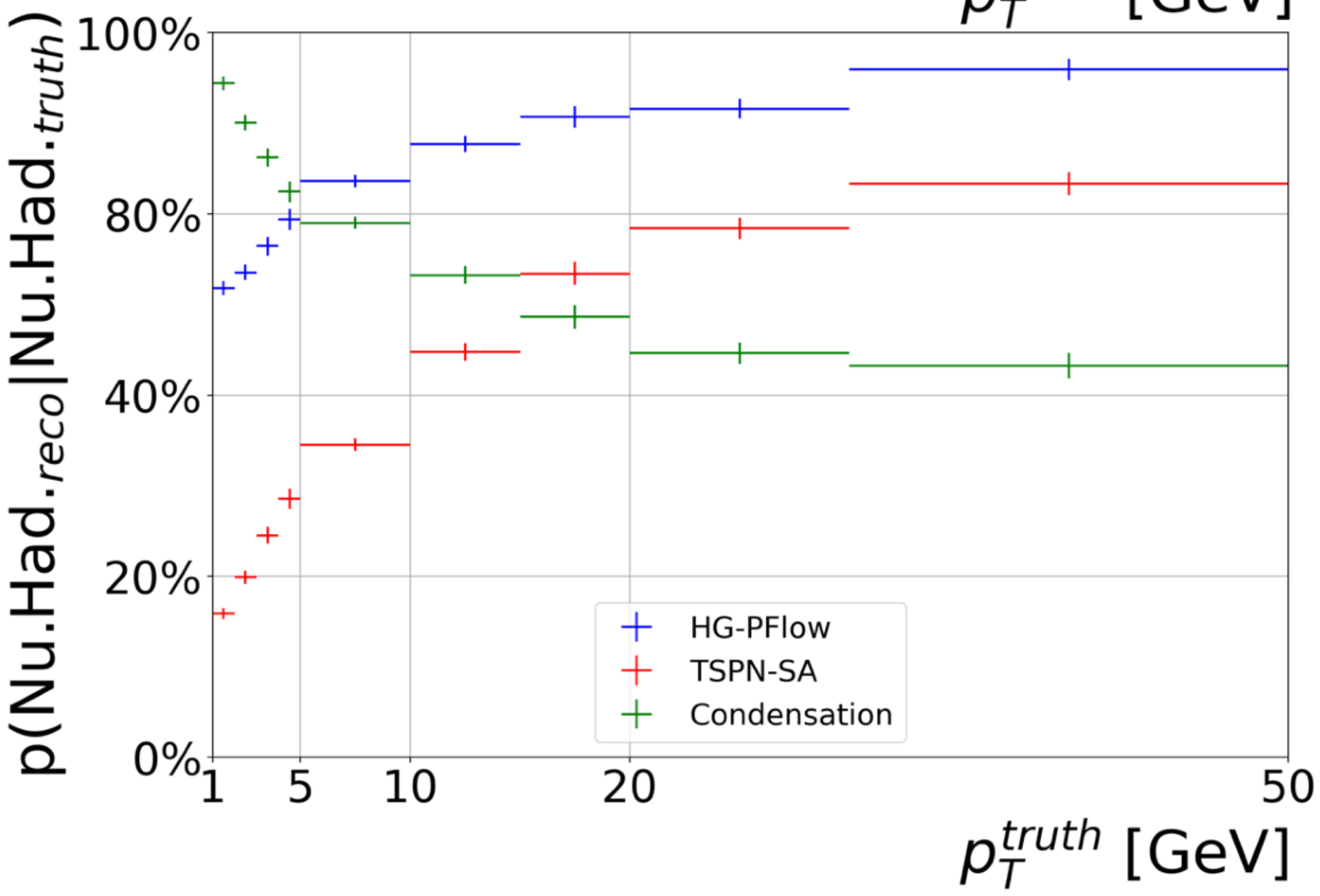
Photons

>90% above 5 GeV



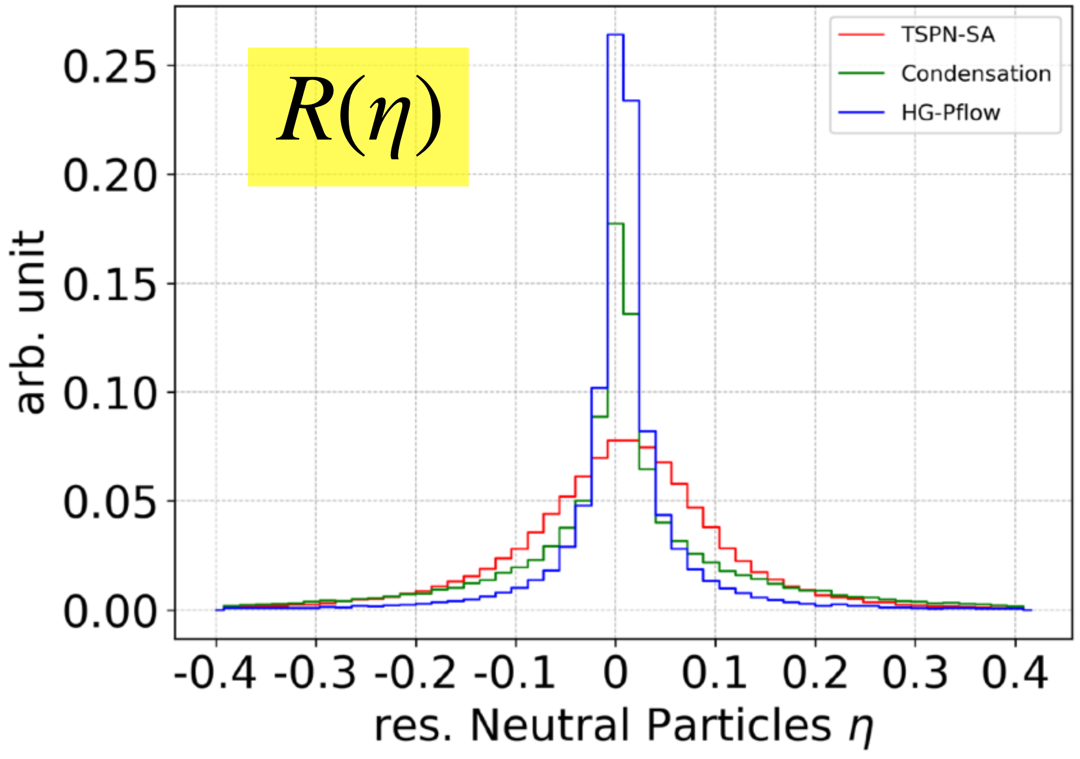
Neutral Hadrons

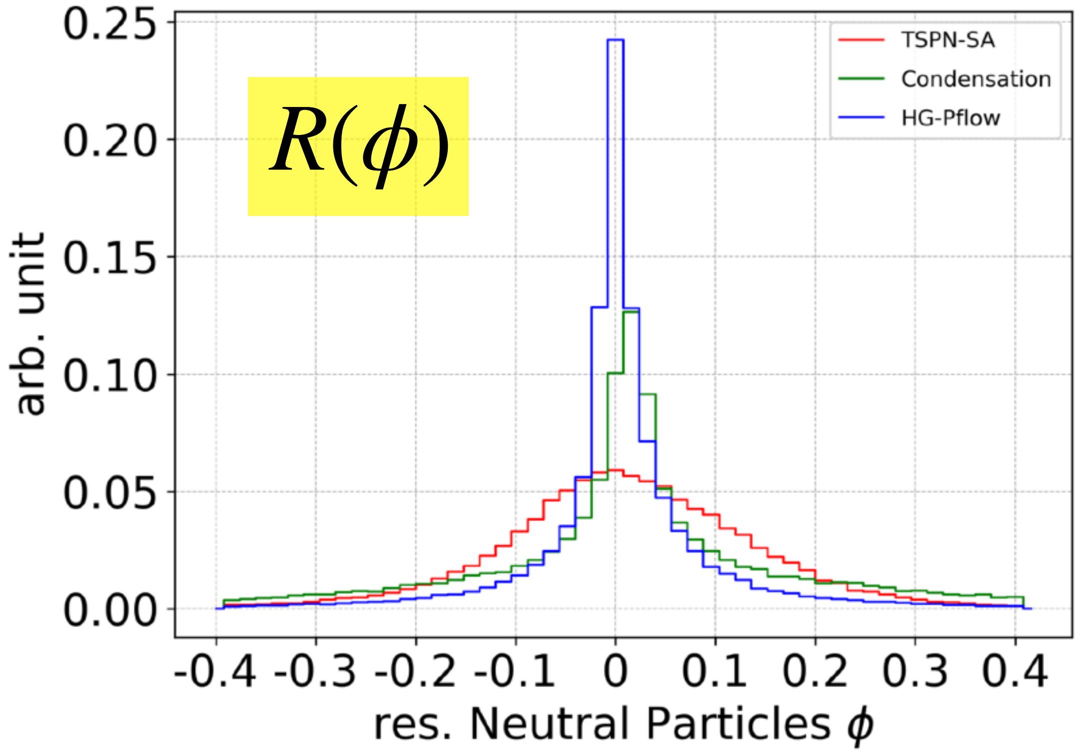
>90% above 15 GeV

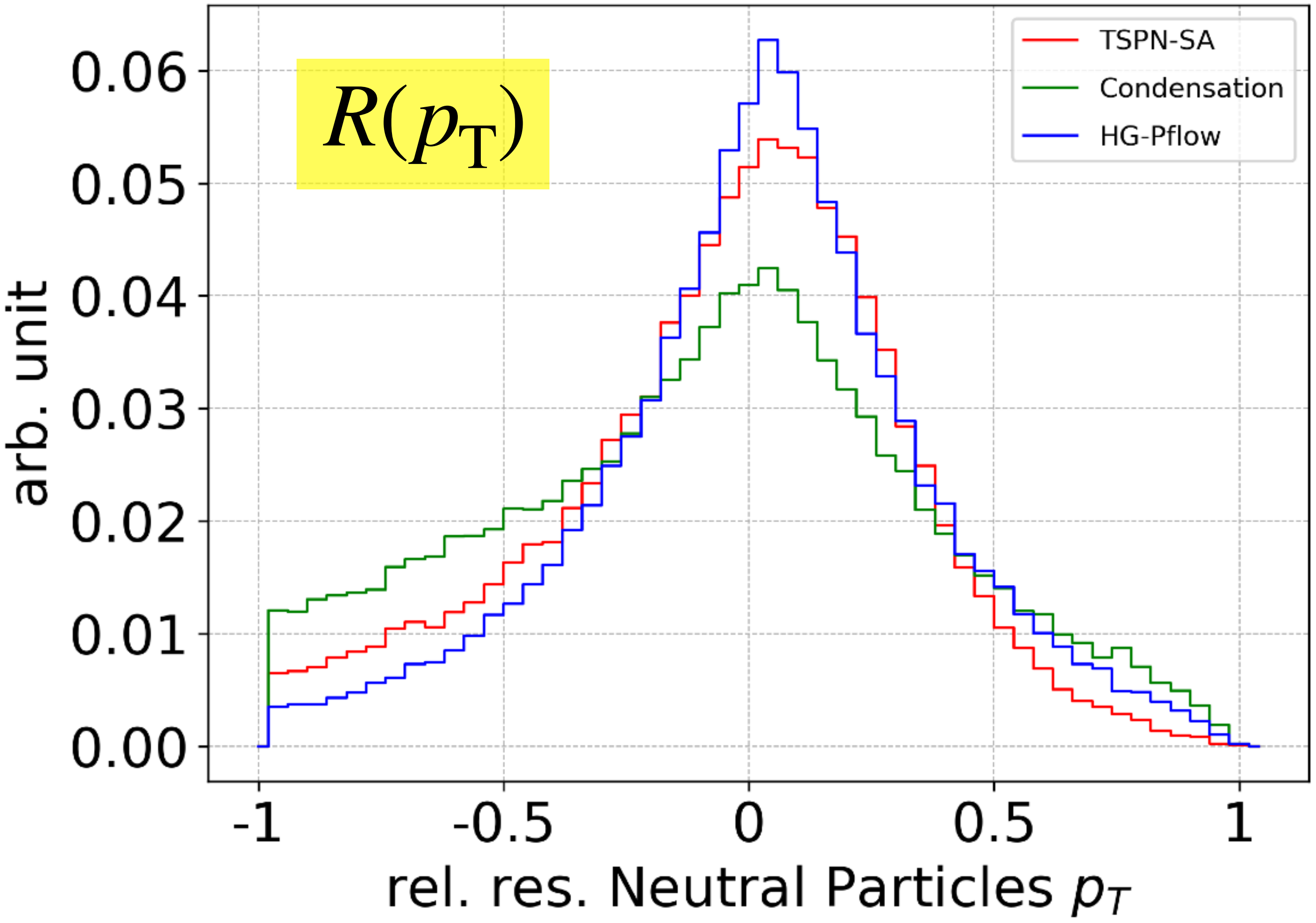


Neutral particle properties

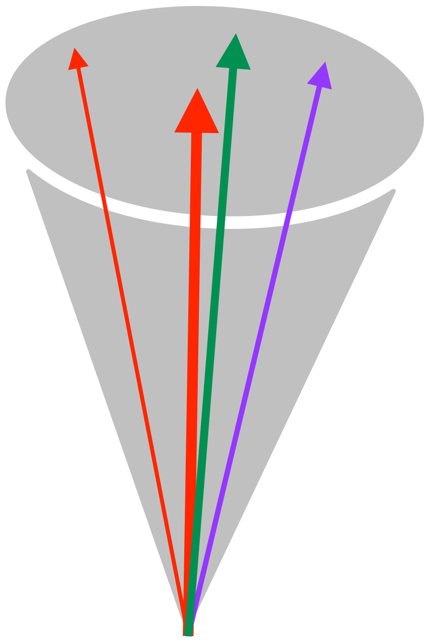
$$R \equiv \frac{\text{target} \; - \; \text{predicted}}{\text{target}}$$







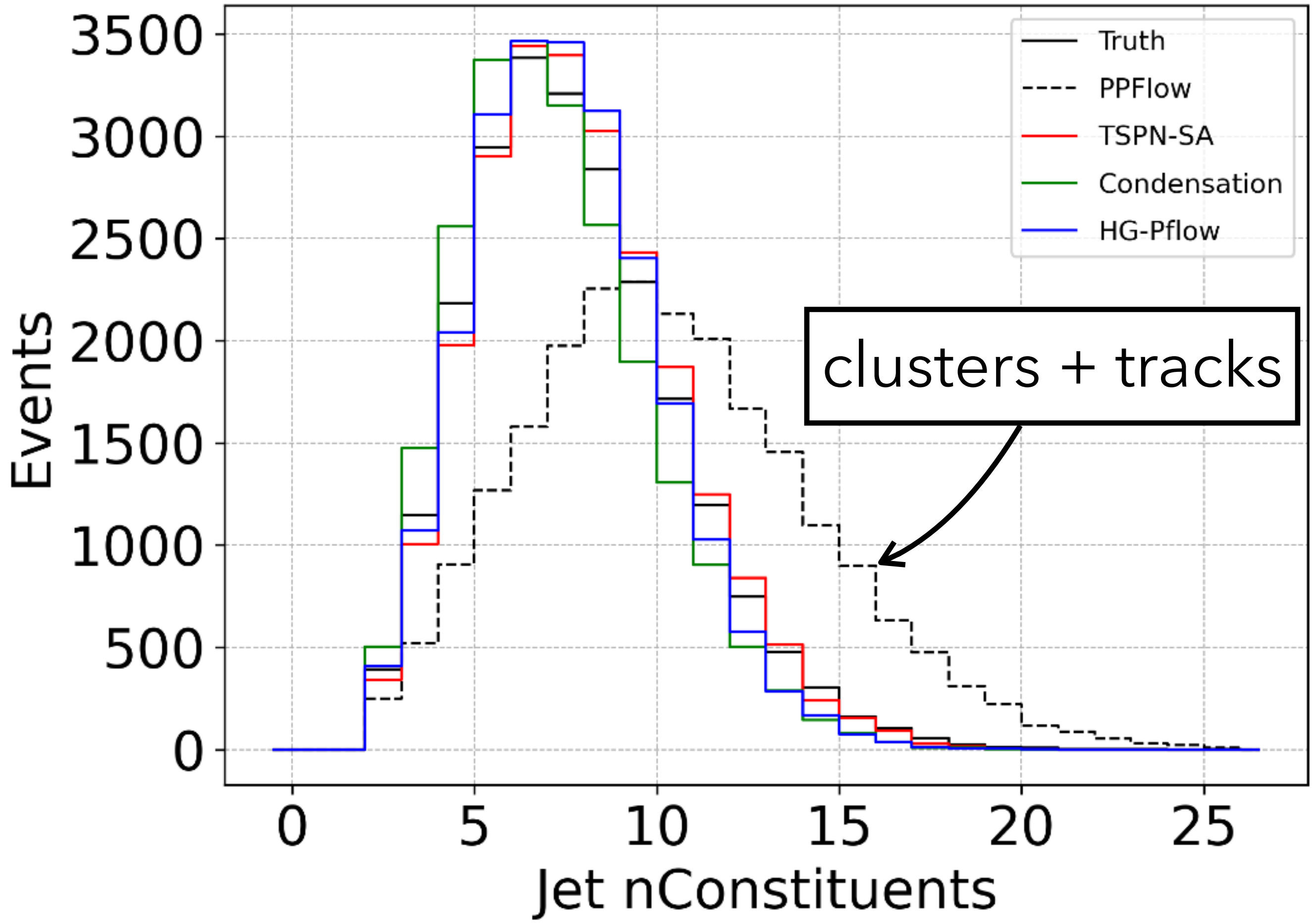
Jet-level performance



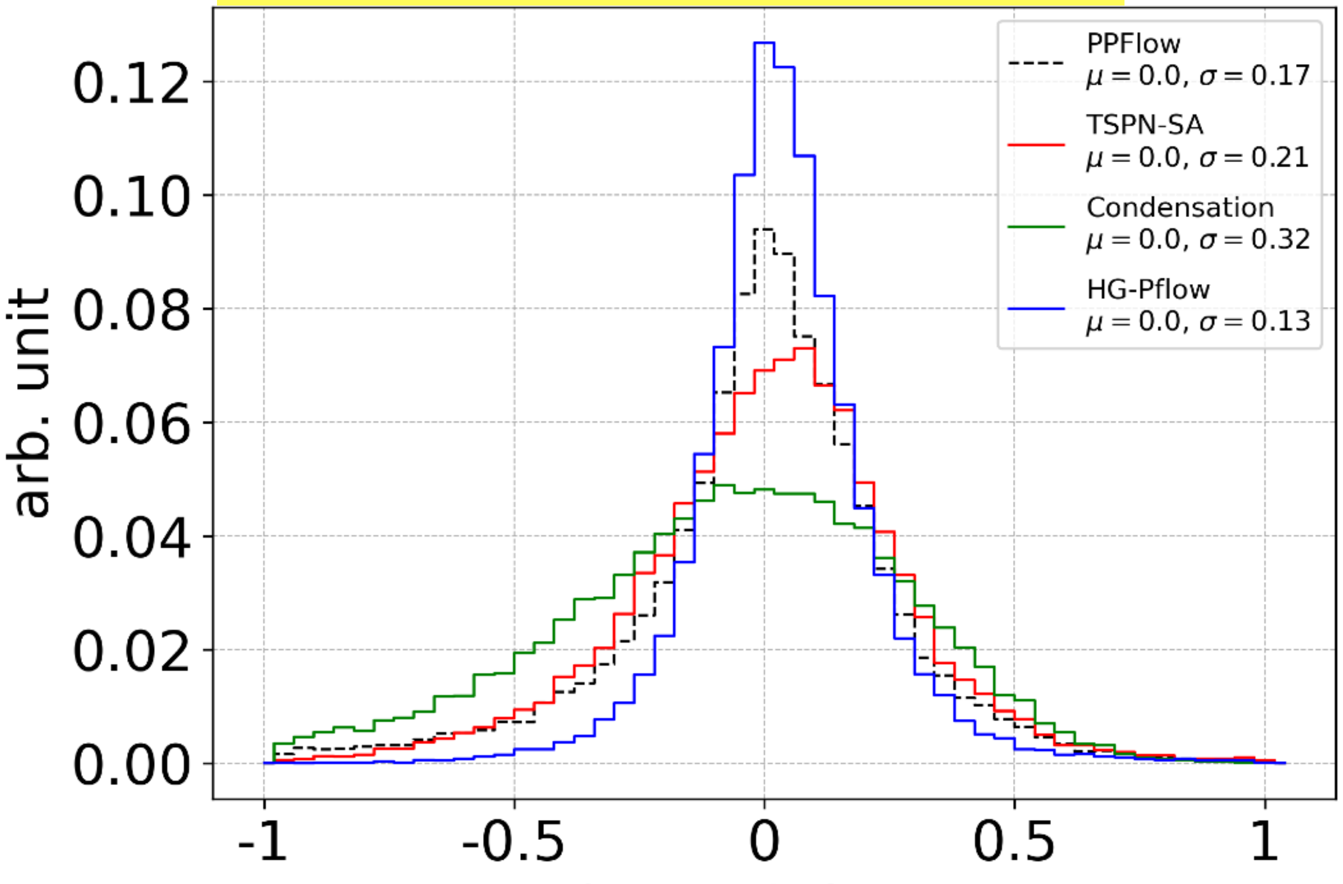
- ---- Truth
- ---- PPFlow
- ━━ TSPN-SA
- ━━ Condensation
- ━━ HG-Pflow

ATLAS-like parameterized particle-flow

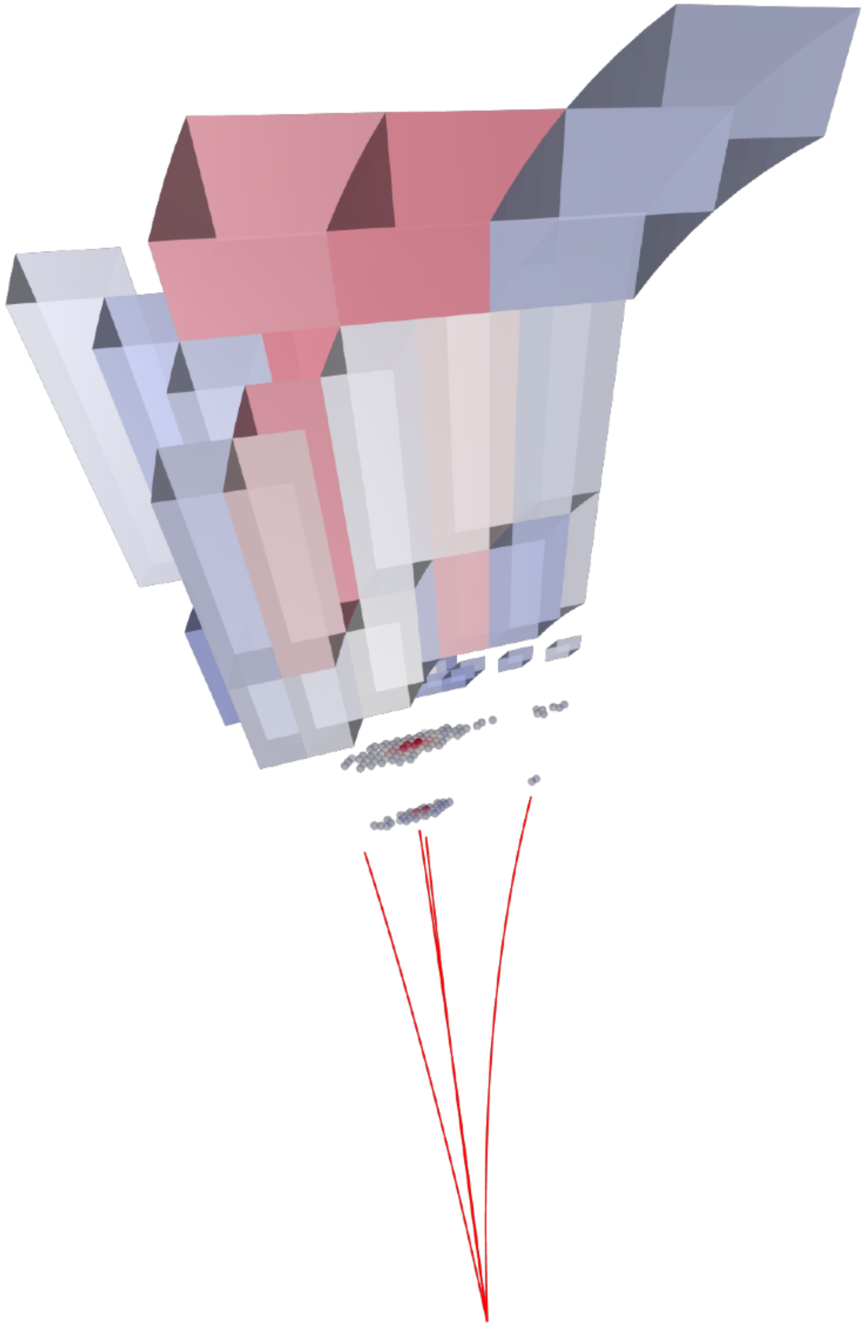
 # constituents / jet



Jet momentum resolution

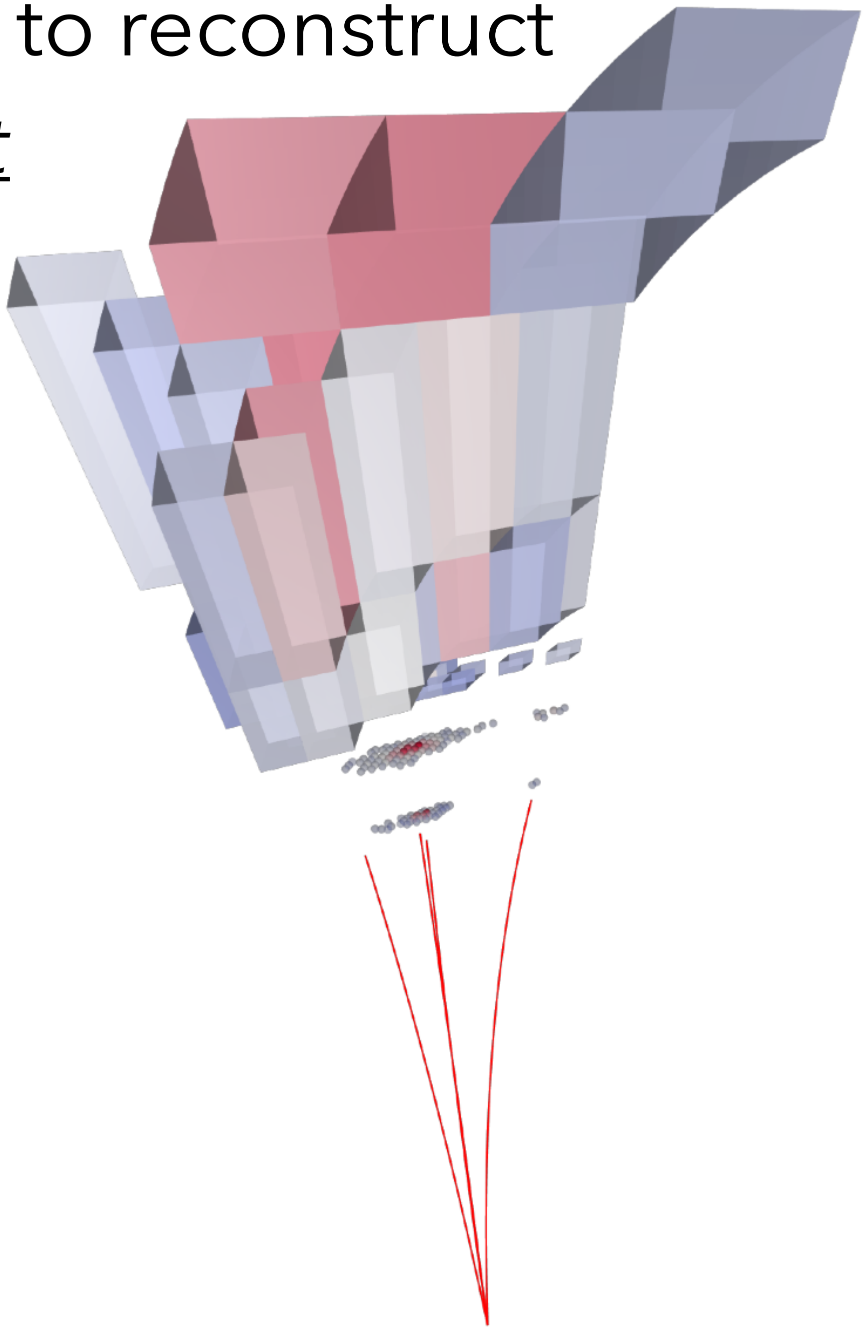


Take-home messages & outlook

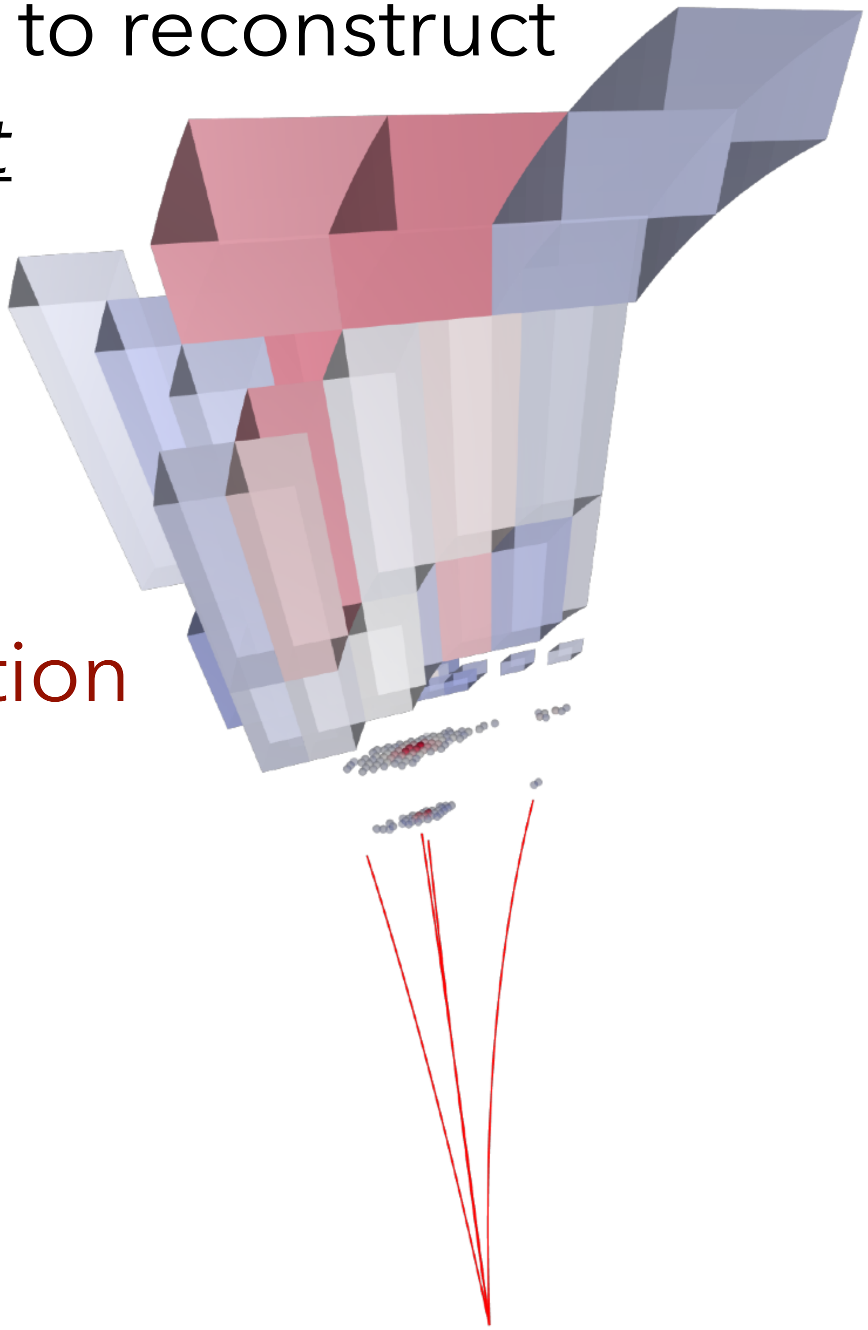


Take-home messages & outlook

- Graph-based set-to-set models show ability to reconstruct individual neutral constituents *inside of a jet*



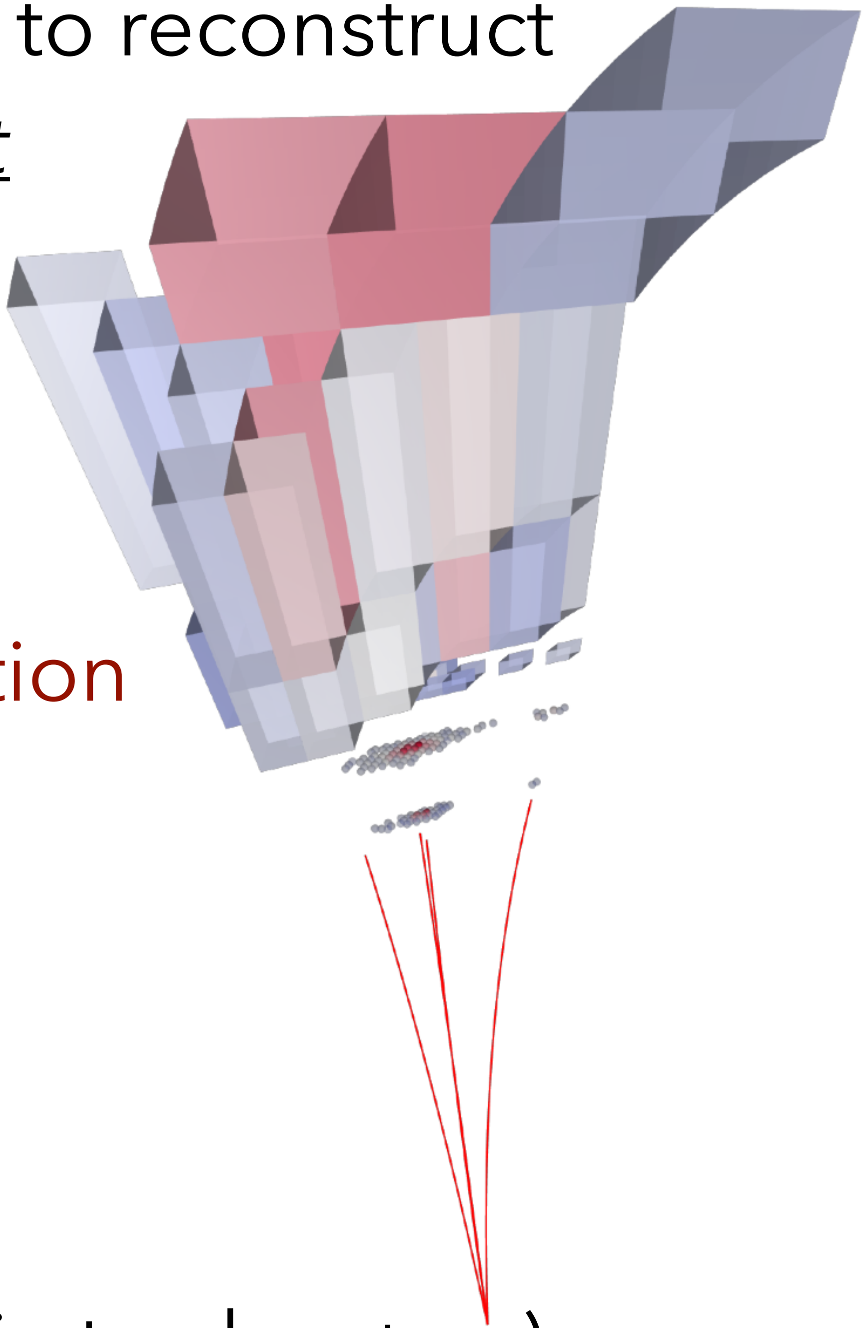
Take-home messages & outlook



- Graph-based set-to-set models show ability to reconstruct individual neutral constituents *inside of a jet*

- *HGPflow*: particles as hyperedges
  - ✓ Enables physically-interpretable results
  - ✓ Introduces bias towards energy conservation
  - ✓ Shows best performance in our study

Take-home messages & outlook



- Graph-based set-to-set models show ability to reconstruct individual neutral constituents *inside of a jet*

- *HGPflow*: particles as hyperedges
  - ✓ Enables physically-interpretable results
  - ✓ Introduces bias towards energy conservation
  - ✓ Shows best performance in our study

Stay tuned for arXiv!

Next steps

- Full-event dataset (+ pileup + γ conversions in tracker + …)
- Improvements to HGPflow (one-shot training, cell-level input, …)

Bonus

"Particle flow" paradigm

Main idea: use full detector information in complimentary way

e.g. track-cluster association

→ problem: potential overcounting

Traditional recipe [1]

1. Identify groups of cells (topological clustering)
2. Find associated tracks
3. Decide whether to merge with additional clusters
4. Subtract expected E from track to infer contribution from neutral particles

- Each step relies on discrete rules with few tuned parameters
- Does not predict cardinality or properties of the neutral particles

Can we approach this as a machine learning task?

---

[1] ATLAS Collaboration. Jet reconstruction and performance using particle flow with the ATLAS Detector. Eur. Phys. J. C 77 (2017)

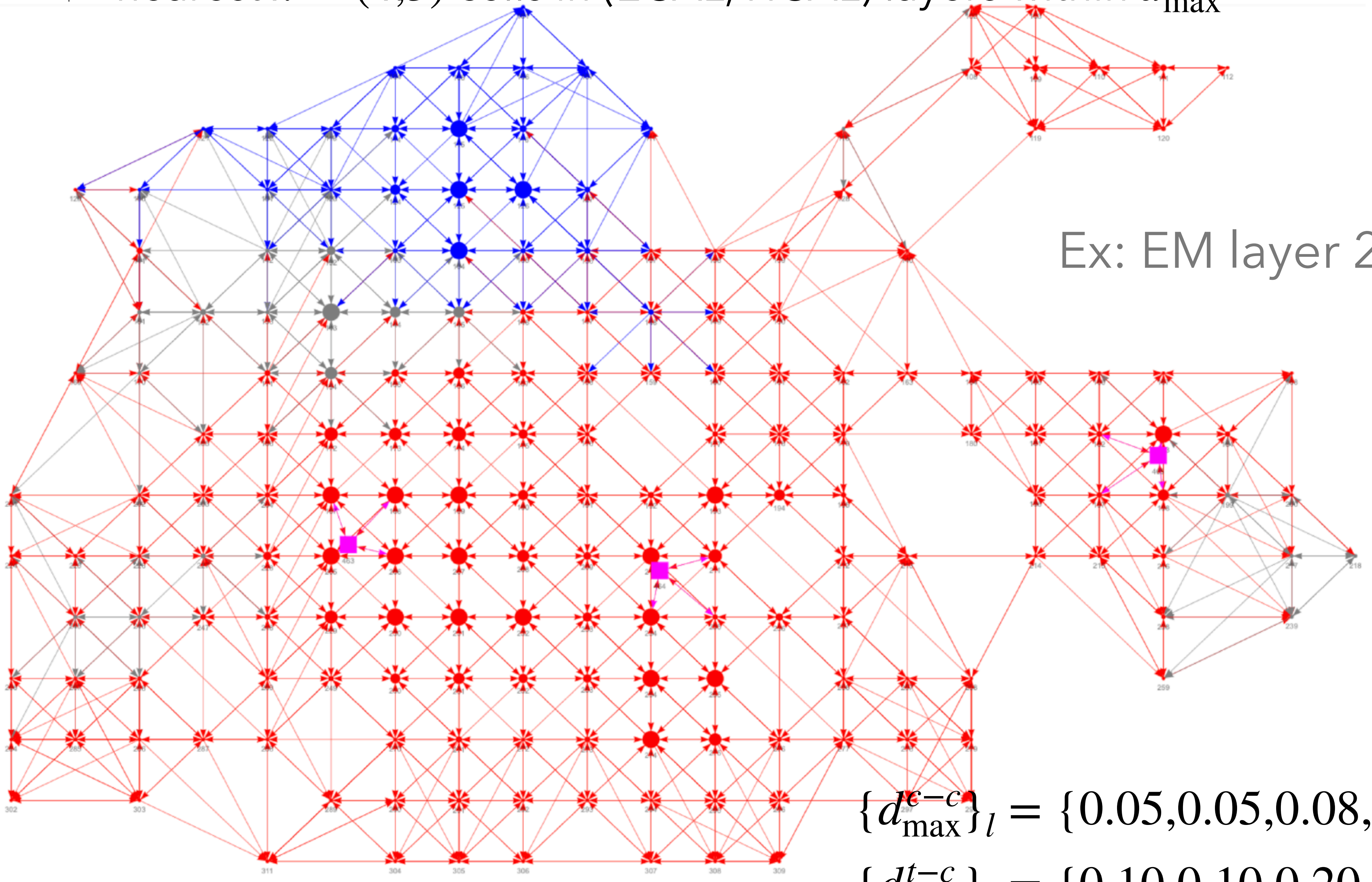
Graph construction

● Cells connected to:

- ✳ nearest $k = (8,6)$ cells in same (ECAL, HCAL) layer within $d_{\max}^{c-c}$
- ✳ single nearest cell in neighboring layers within $d_{\max}^{c-c}$

■ Tracks connected to:

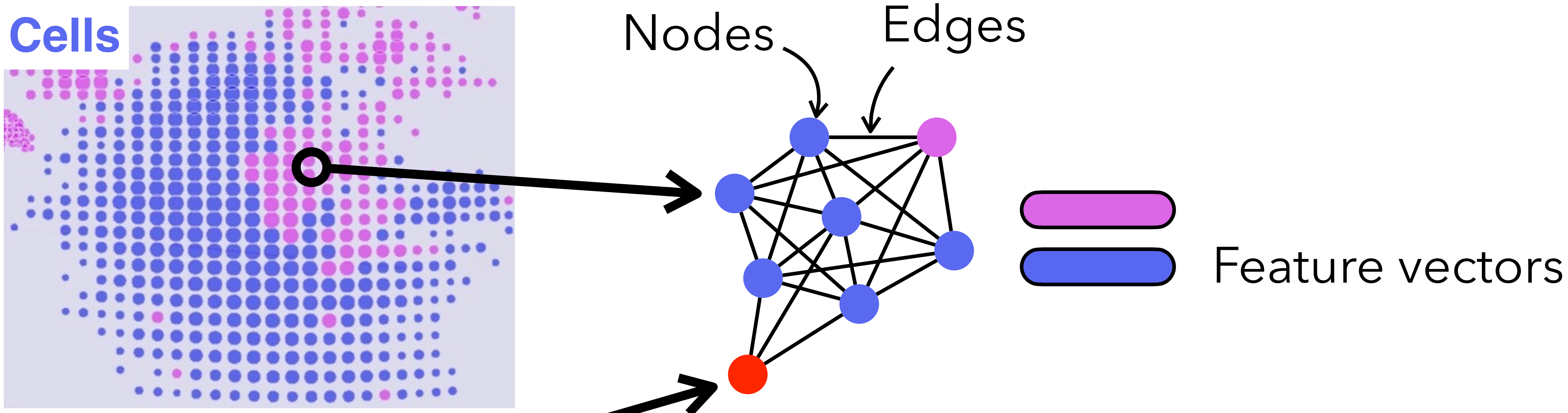
- ✳ nearest $k = (4,3)$ cells in (ECAL, HCAL) layers within $d_{\max}^{t-c}$

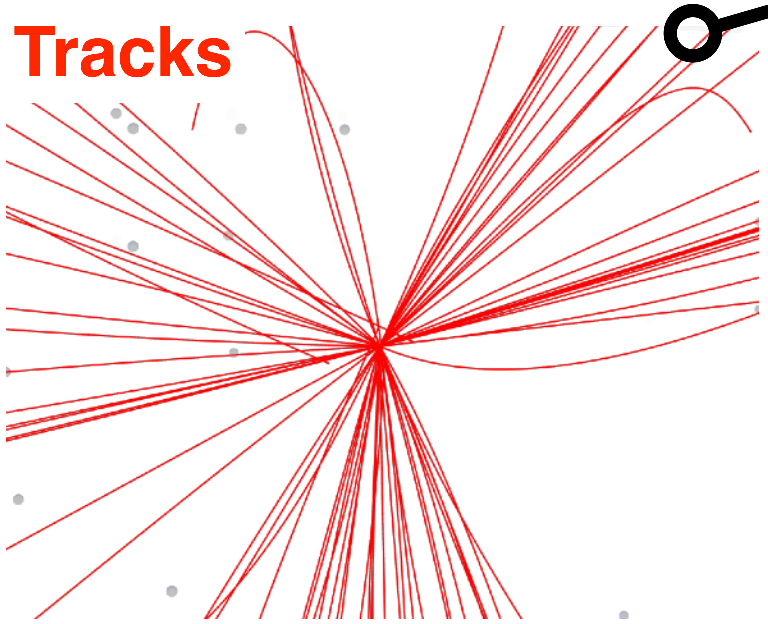


Ex: EM layer 2

$$\{d_{\max}^{c-c}\}_l = \{0.05, 0.05, 0.08, 0.15, 0.15, 0.30\}$$

$$\{d_{\max}^{t-c}\}_l = \{0.10, 0.10, 0.20, 0.50, 0.50, 1.00\}$$





Type	Spatial	Kinematic	Auxiliary	Total
cell	x, y, z, η, ϕ, layer	$E, E/\sigma_{\text{noise}}$	isTrack, …	11
track	$d_0, z_0, \{\eta, \phi\}_{0,\text{proj}}$	$p_{\text{T}}, q/p$	isTrack, isMuon, …	26

Why use graphs?

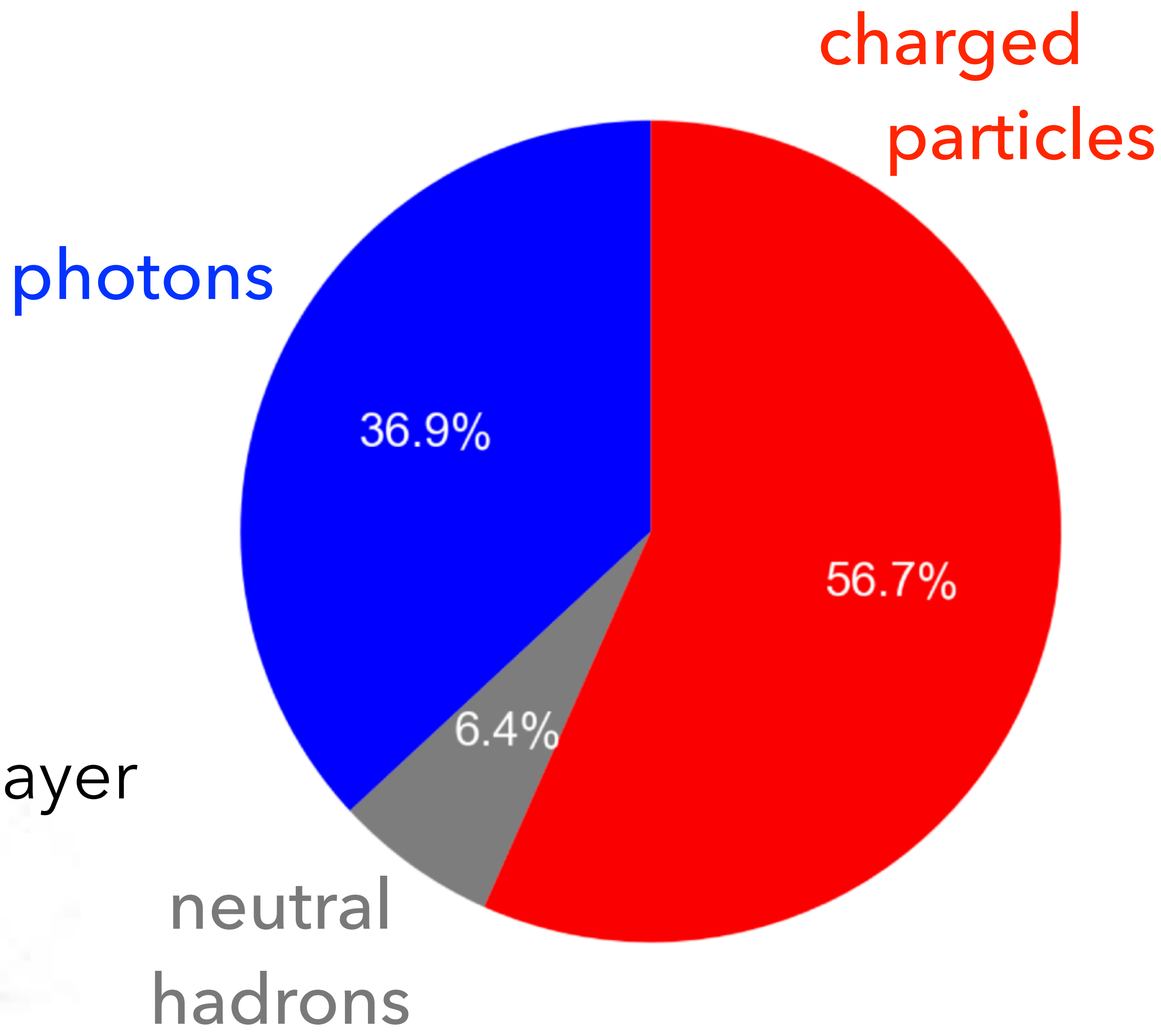
⇒ Well-suited to detector data:

- No ordering
- Set-to-set problem with variable cardinality
- Sparse (most cells not activated)
- Irregular (multiple detector geometries)
- Key info encoded in spatial relationships

Plan of attack

Dataset

- 50k single-jet events (30k test)
- SCD calo simulation
- Track momentum smearing
- Neglect pileup
- No γ conversions upstream from iron layer



Target

Particles $\geq 1\text{GeV}$ entering calorimeter

ML reconstruction algorithms

1. Object condensation
2. TSPN + slot attention
3. HGPflow