Event Reconstruction at the LHC II

FERMILAB-CERN
HADRON COLLIDER PHYSICS
SUMMER SCHOOL
2024

Richard Cavanaugh, Fermilab/UIC, HCPSS 31 July 2024

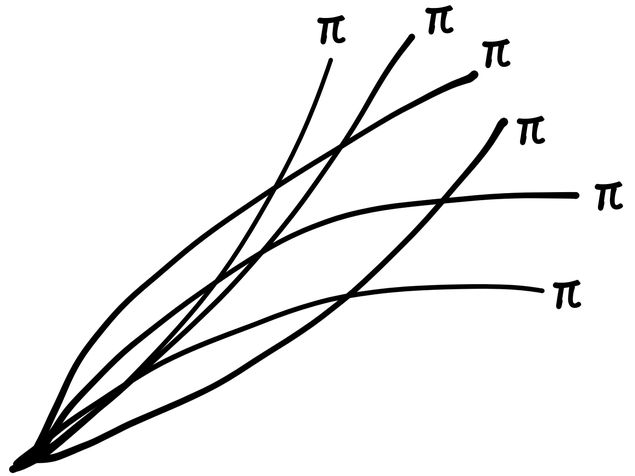
Recall: Jet Composition

Recall: Jet Composition

- Charged Particles: ~60%

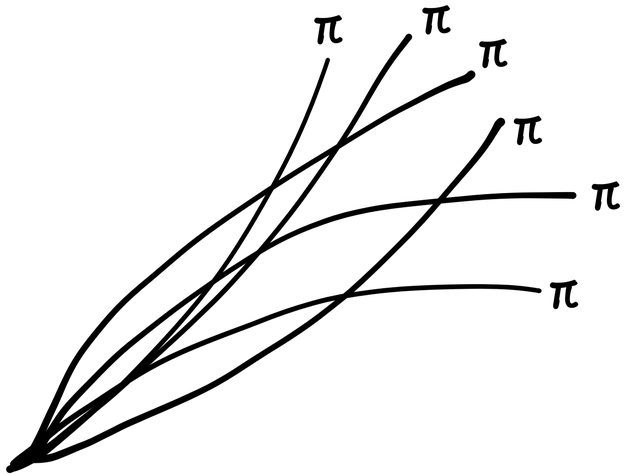
Recall: Jet Composition

- Charged Particles: ~60%



Recall: Jet Composition

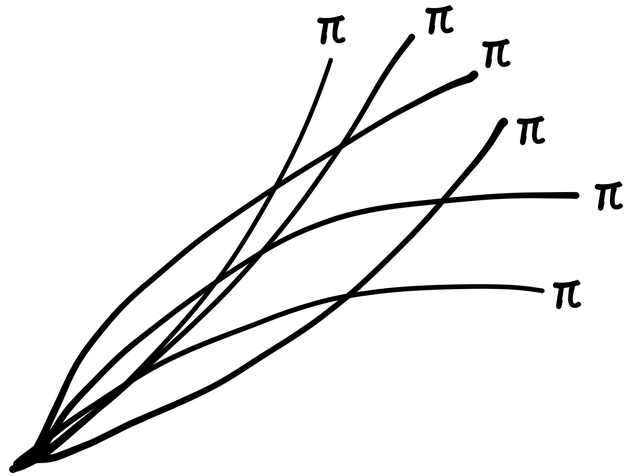
- Charged Particles: ~60% **Tracking Detectors**



Recall: Jet Composition

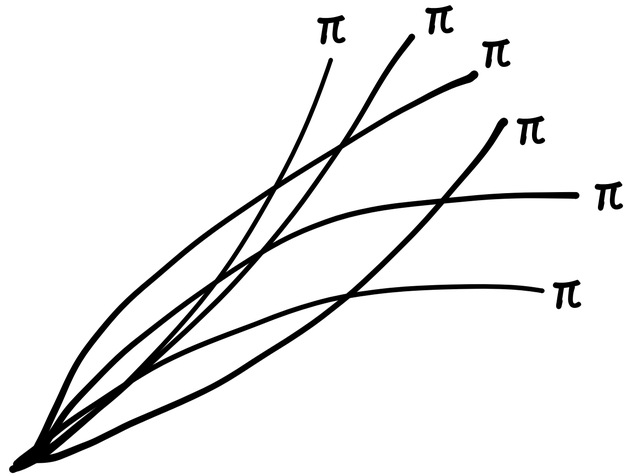
- Charged Particles: ~60% **Tracking Detectors**

- Photons: ~25%



Recall: Jet Composition

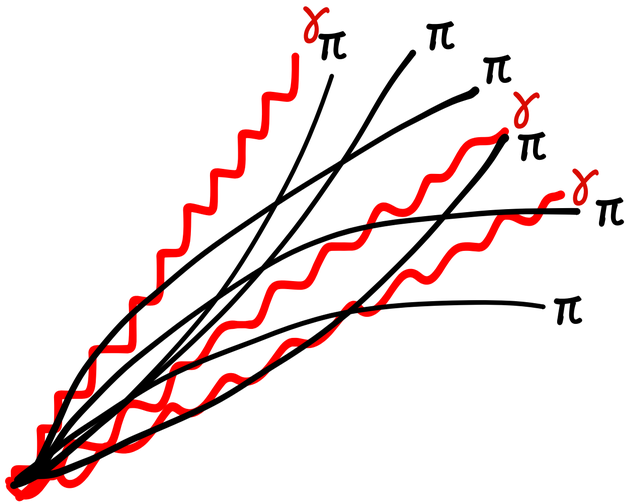
- Charged Particles: ~60% **Tracking Detectors**
- Photons: ~25% **EM Calorimeters**



Recall: Jet Composition

- Charged Particles: ~60% **Tracking Detectors**

- Photons: ~25% **EM Calorimeters**



Recall: Jet Composition

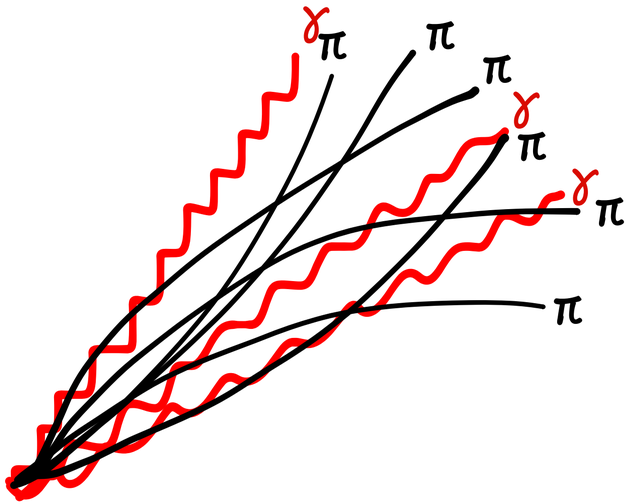
- Charged Particles: ~60%

Tracking Detectors

- Photons: ~25%

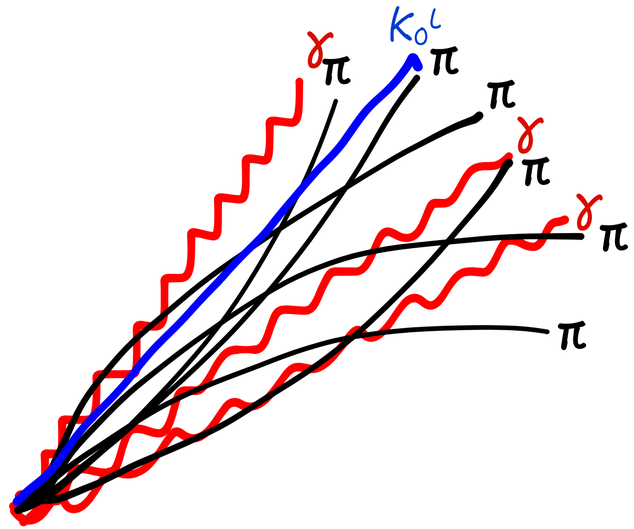
EM Calorimeters

- Long-lived Neutral Hadrons: ~10%



Recall: Jet Composition

- Charged Particles: ~60% **Tracking Detectors**
- Photons: ~25% **EM Calorimeters**
- Long-lived Neutral Hadrons: ~10%



Recall: Jet Composition

- Charged Particles: ~60%

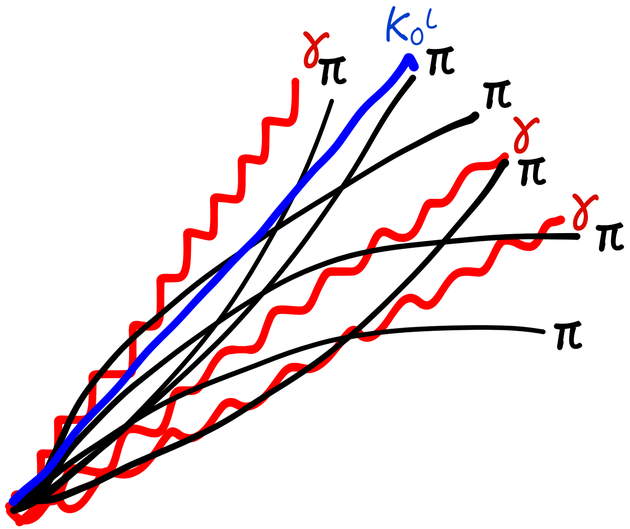
Tracking Detectors

- Photons: ~25%

EM Calorimeters

- Long-lived Neutral Hadrons: ~10%

HAD Calorimeters



Recall: Jet Composition

- Charged Particles: ~60%

Tracking Detectors

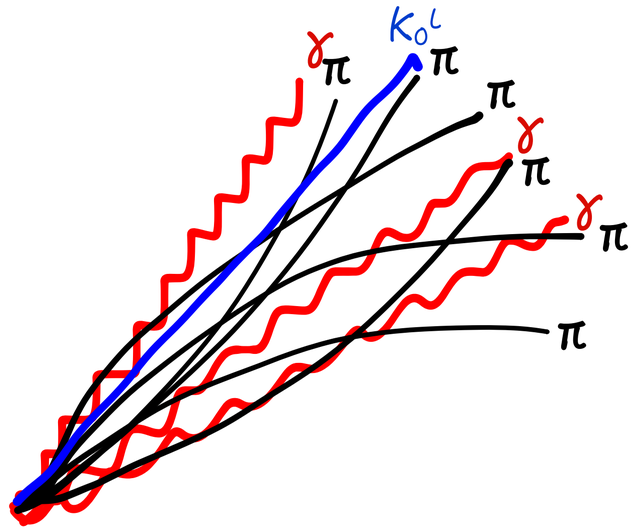
- Photons: ~25%

EM Calorimeters

- Long-lived Neutral Hadrons: ~10%

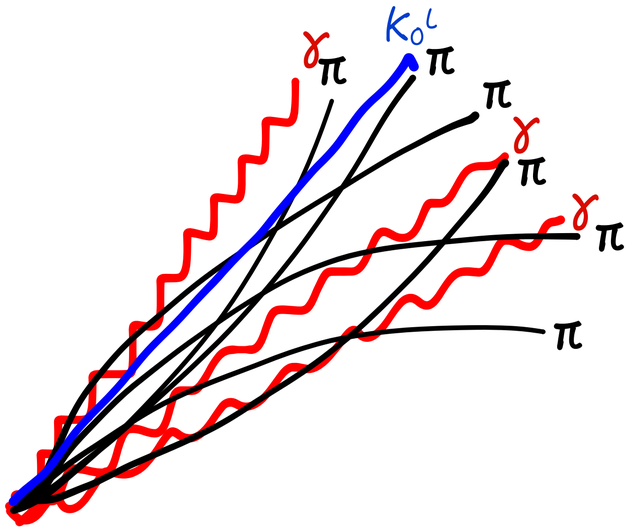
HAD Calorimeters

- Short-lived Neutral Hadrons: ~5%



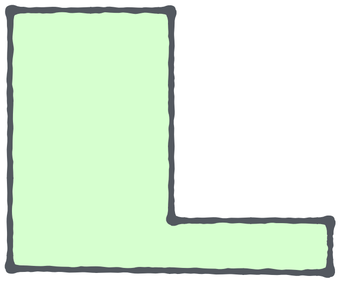
Recall: Jet Composition

- Charged Particles: ~60% **Tracking Detectors**
- Photons: ~25% **EM Calorimeters**
- Long-lived Neutral Hadrons: ~10% **HAD Calorimeters**
- Short-lived Neutral Hadrons: ~5% **Tracking Detectors**



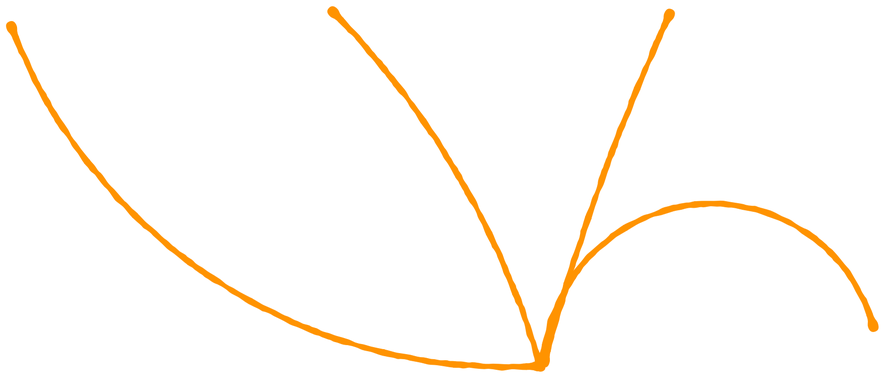
Recall: Associate hits within each detector

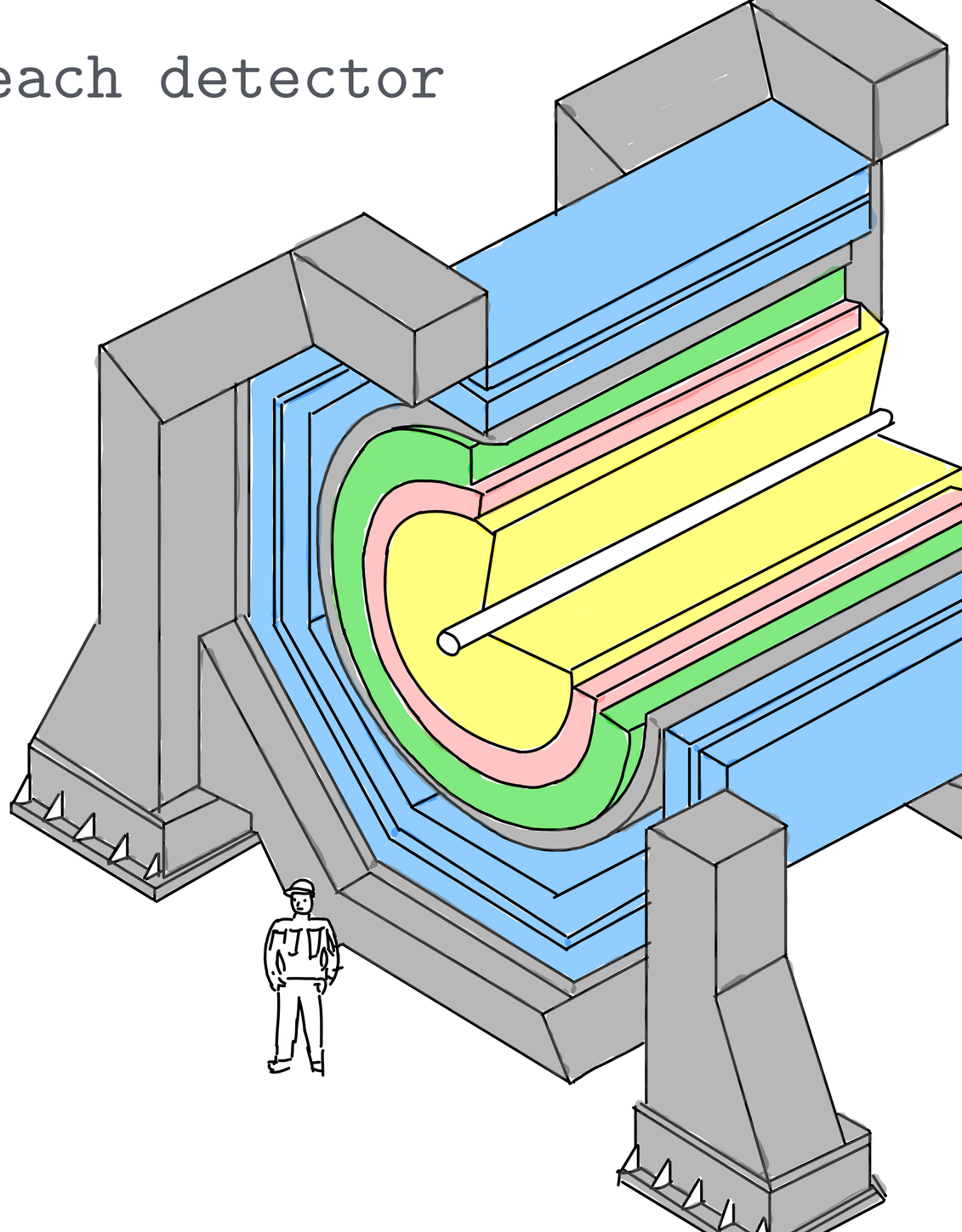
HCAL Clusters



ECAL Clusters

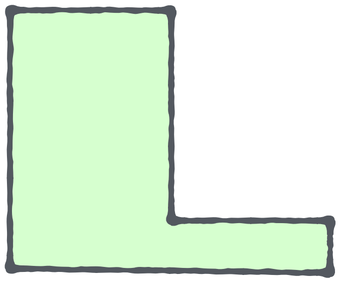
Tracks





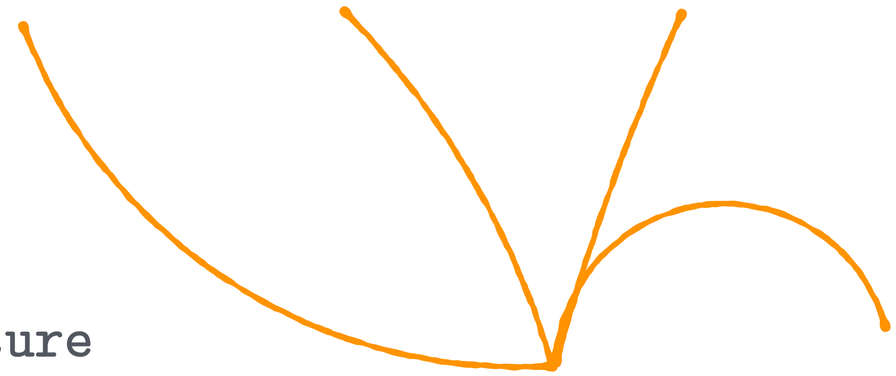
Recall: Associate hits within each detector

HCAL Clusters

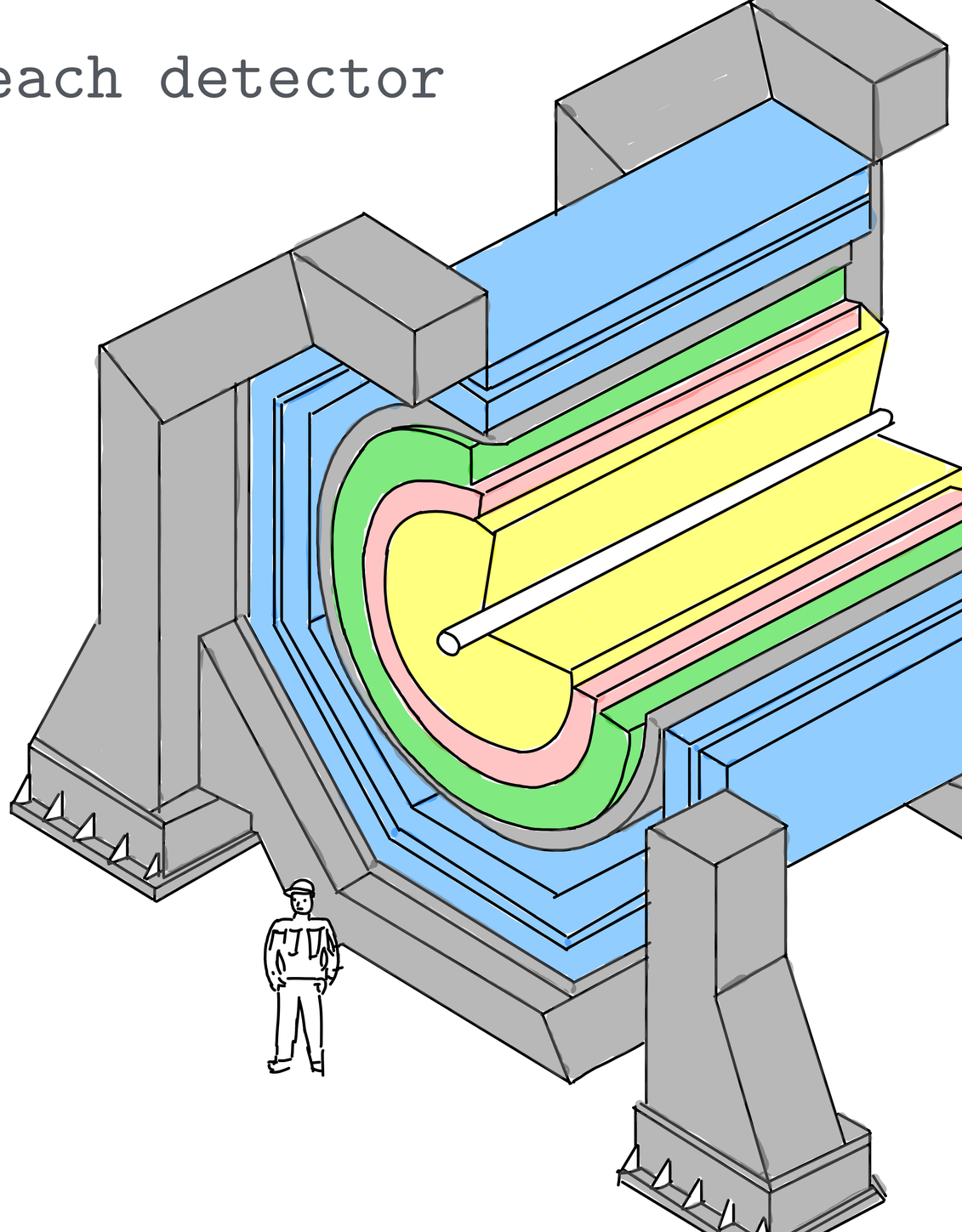


ECAL Clusters

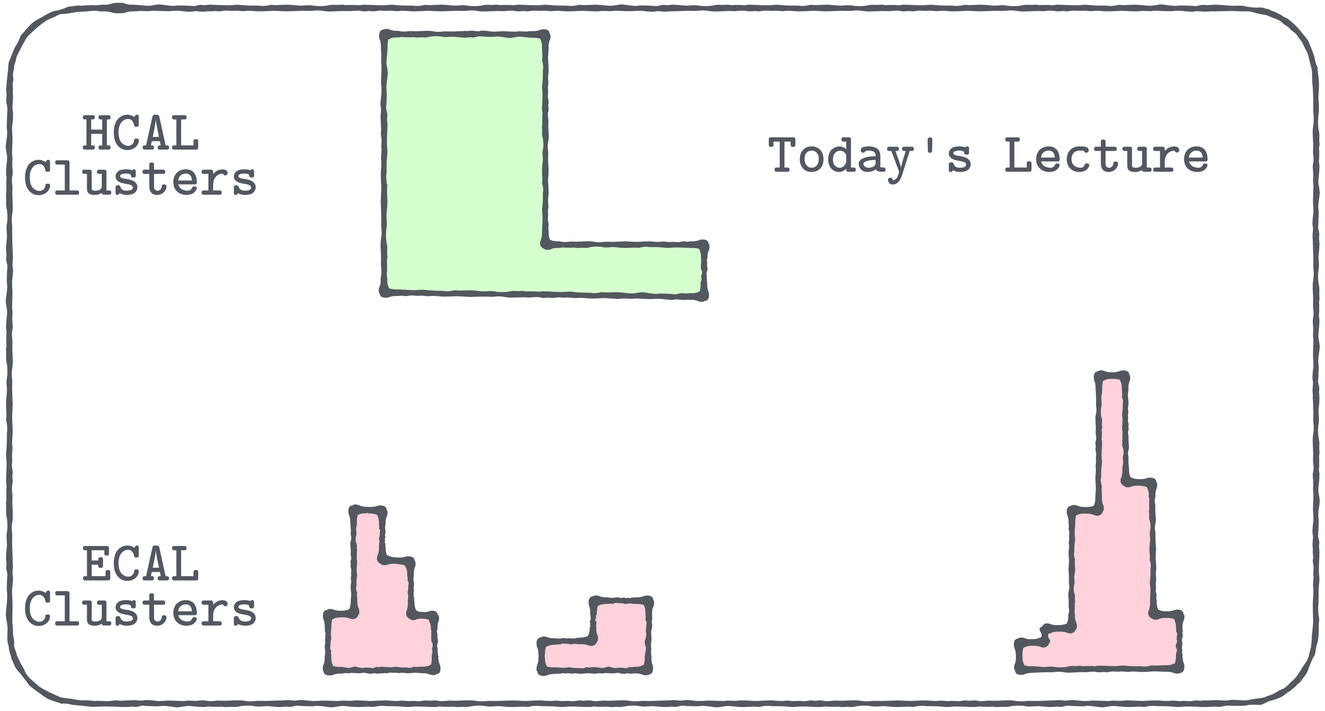
Tracks

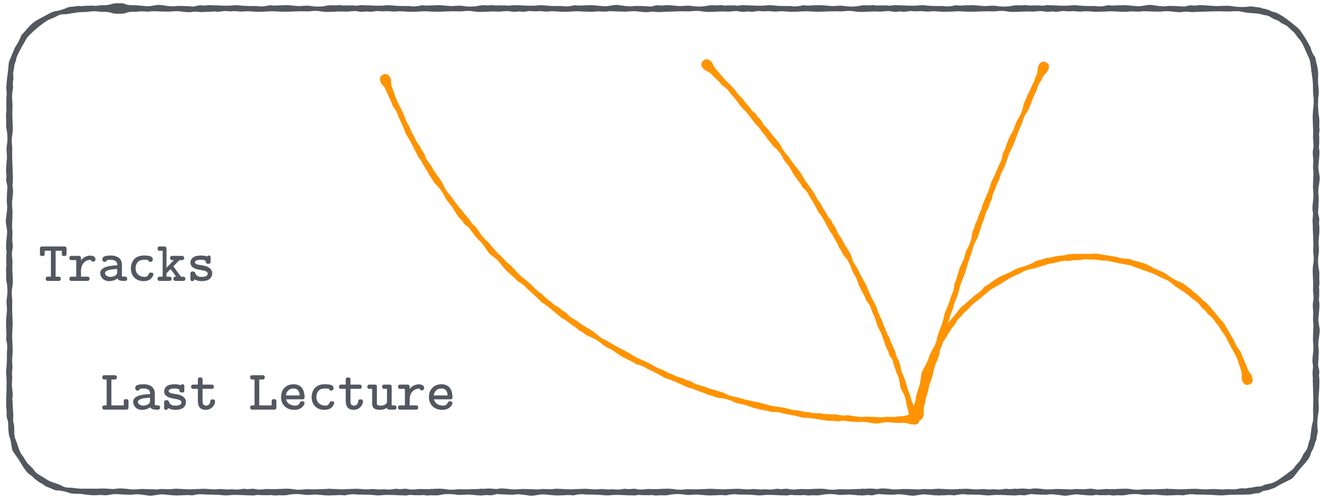


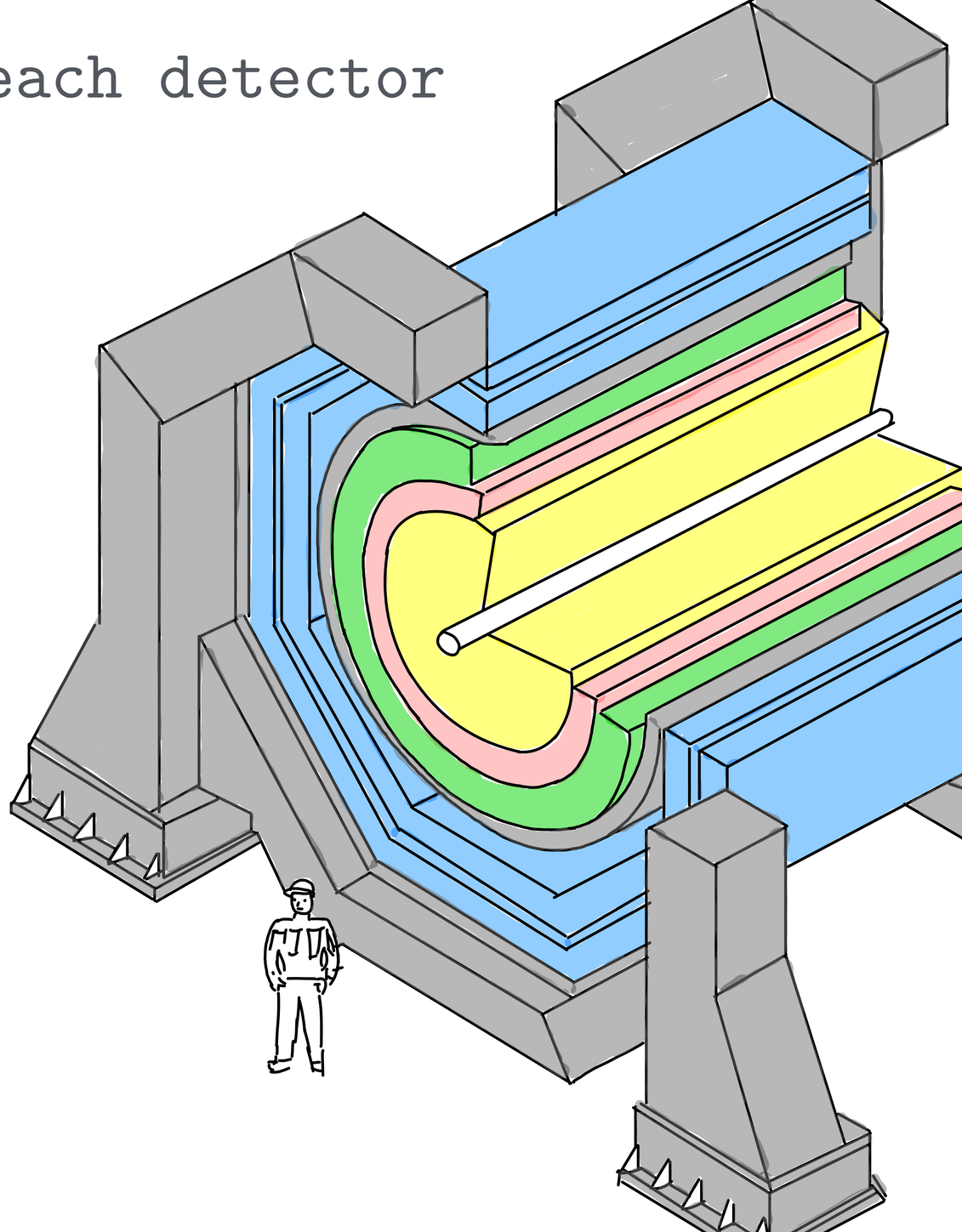
Last Lecture



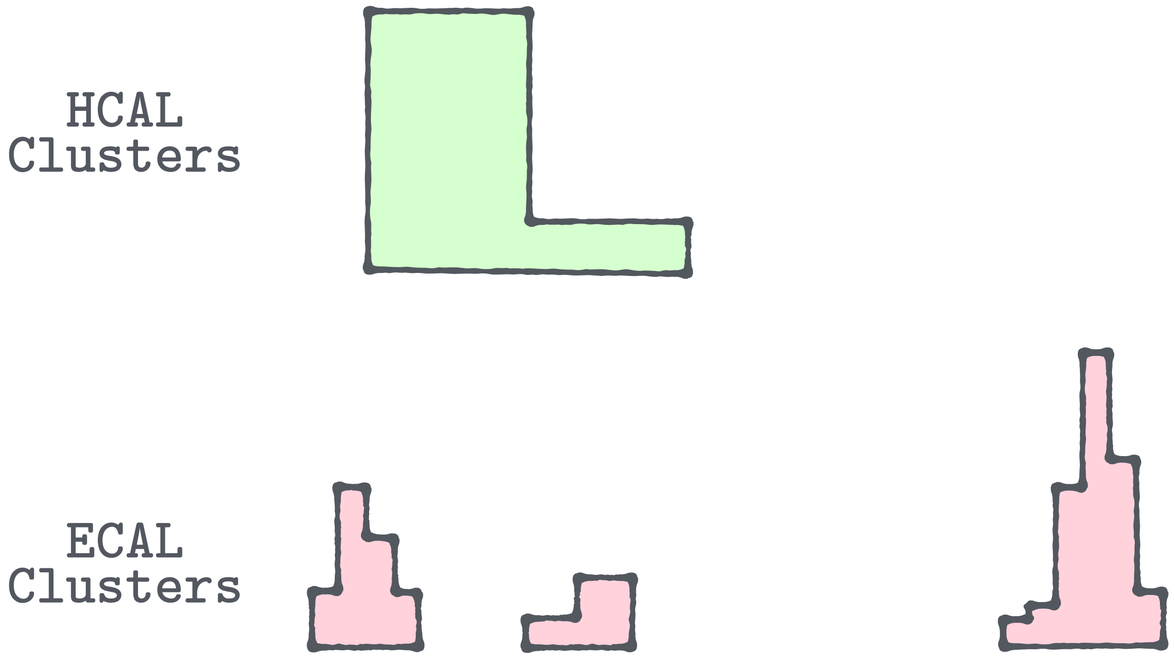
Recall: Associate hits within each detector





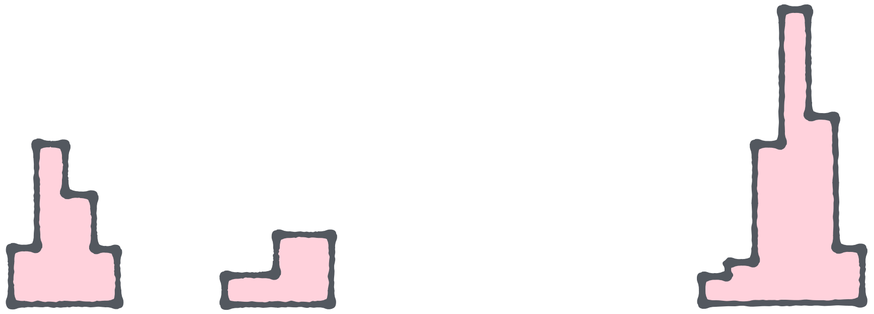


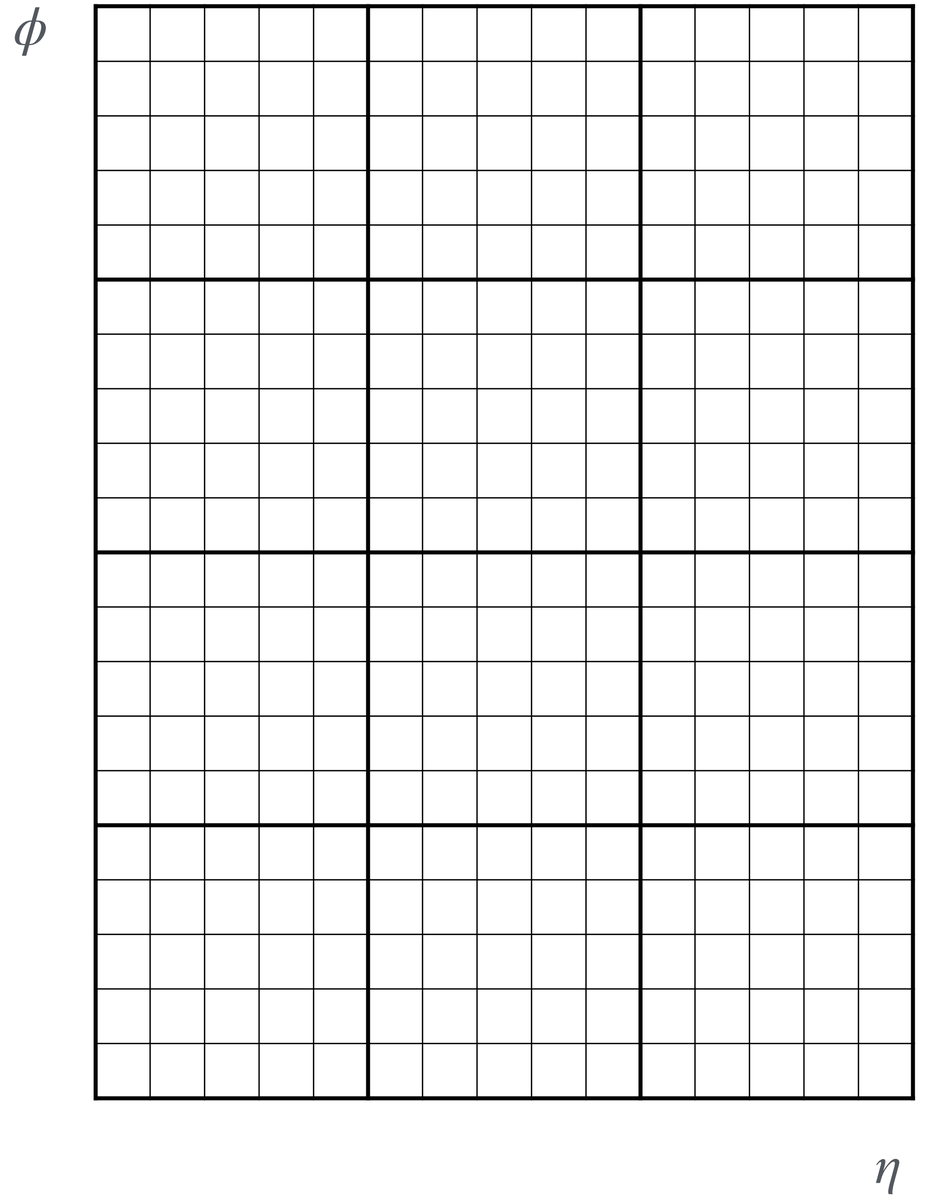
Calorimeter Clustering



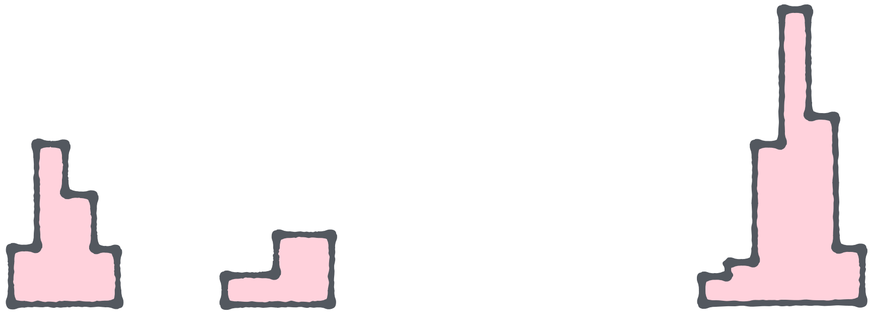
Calorimeter Clustering

ECAL Clusters

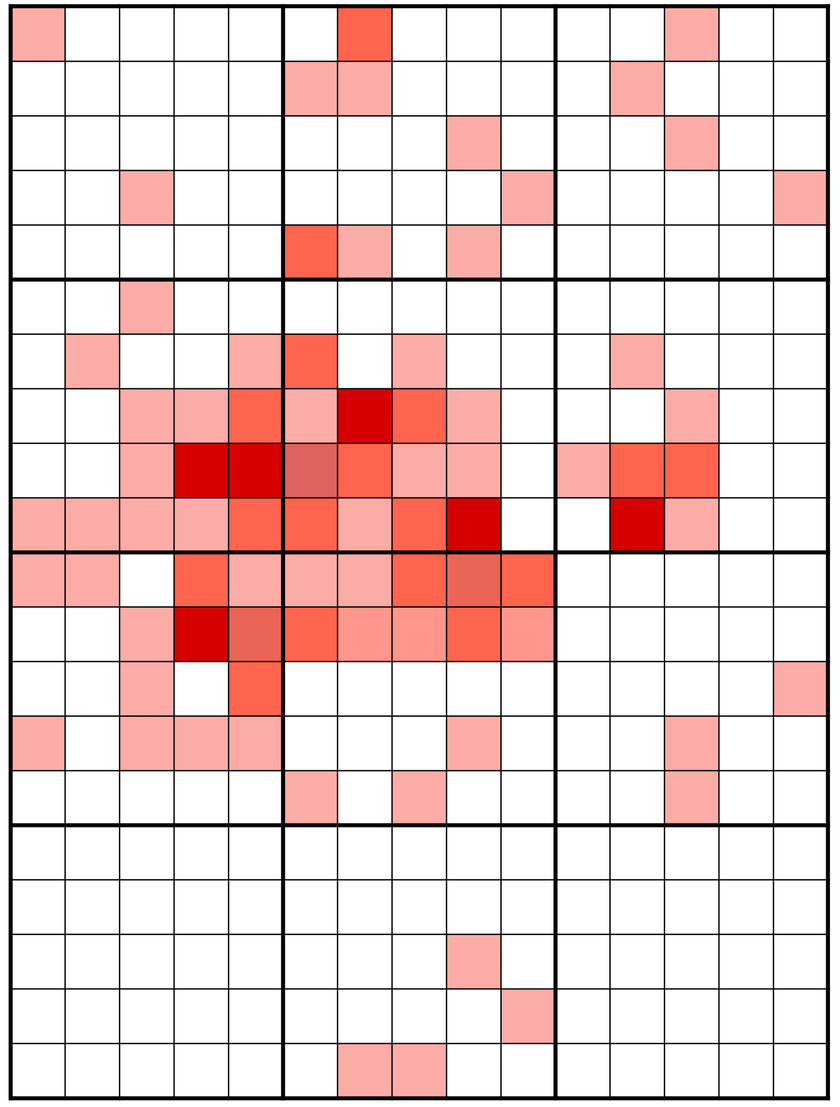




ECAL Clusters



Calorimeter Clustering

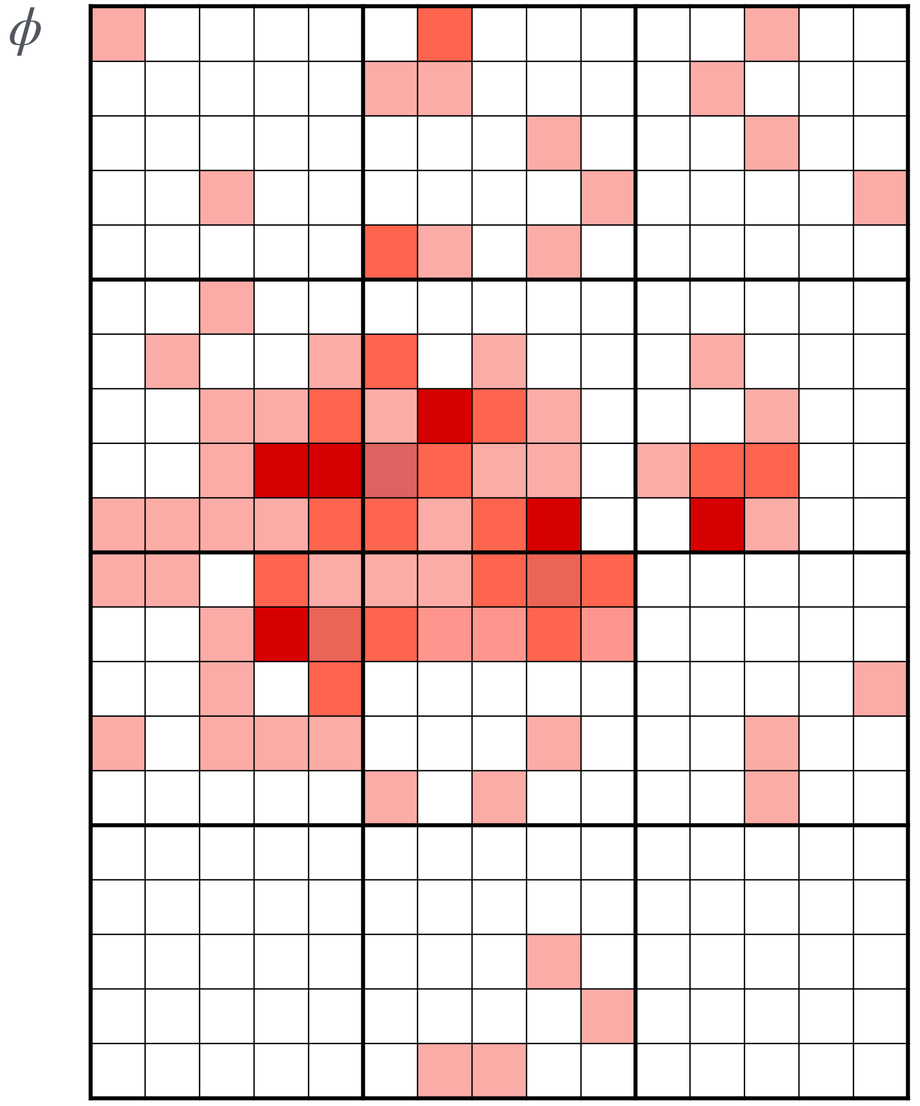


Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

ECAL Clusters

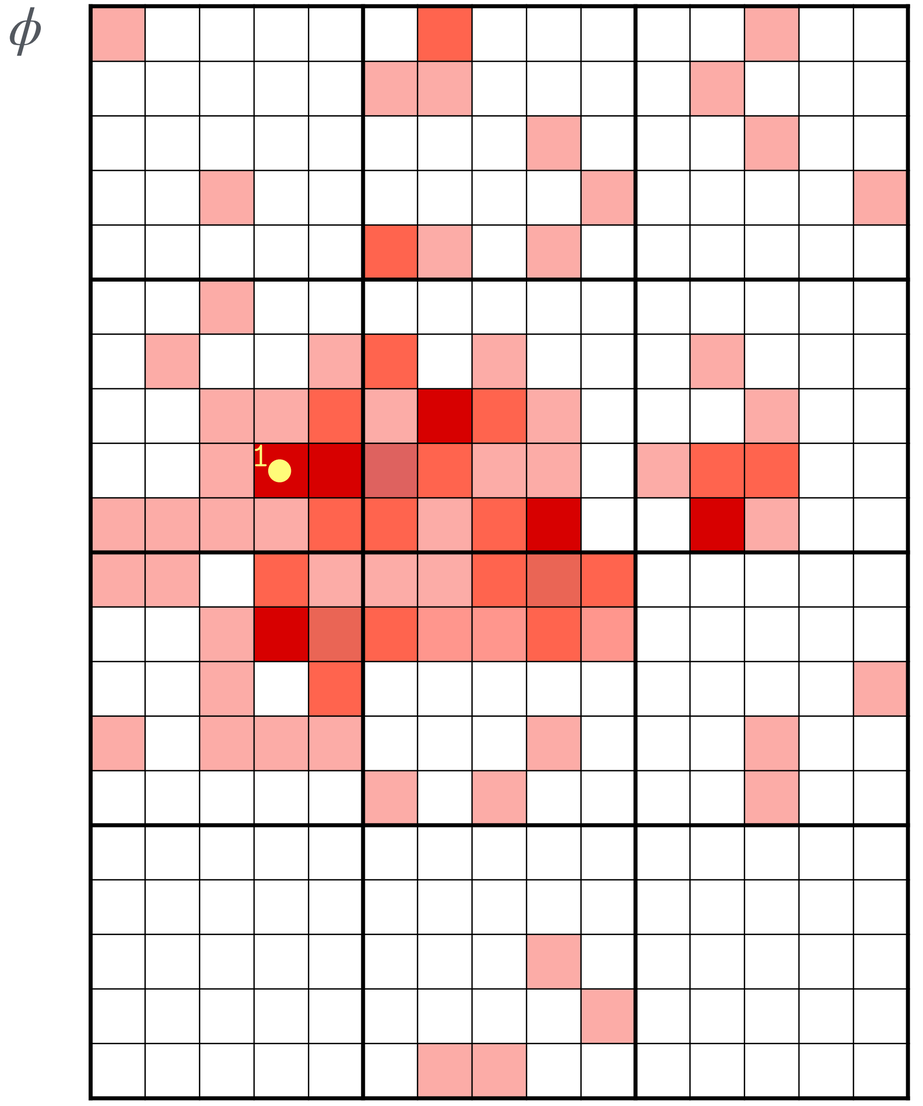


Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

ECAL Clusters

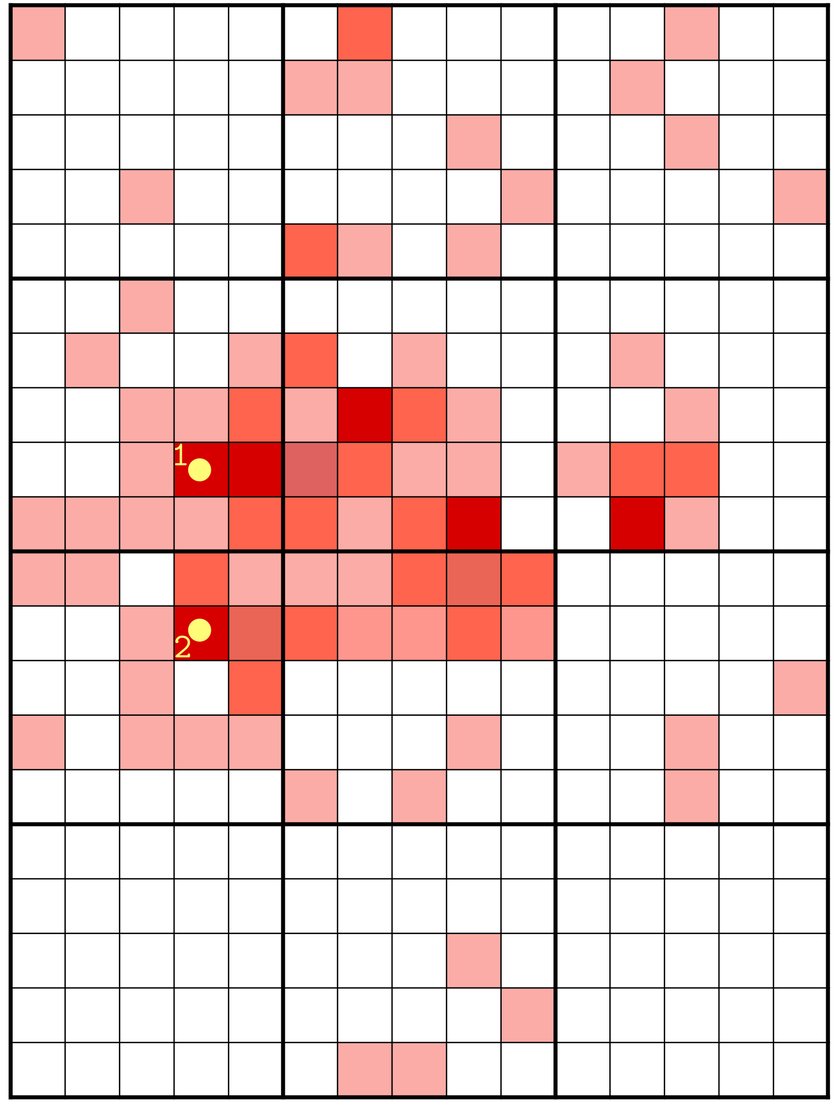


Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

ECAL Clusters

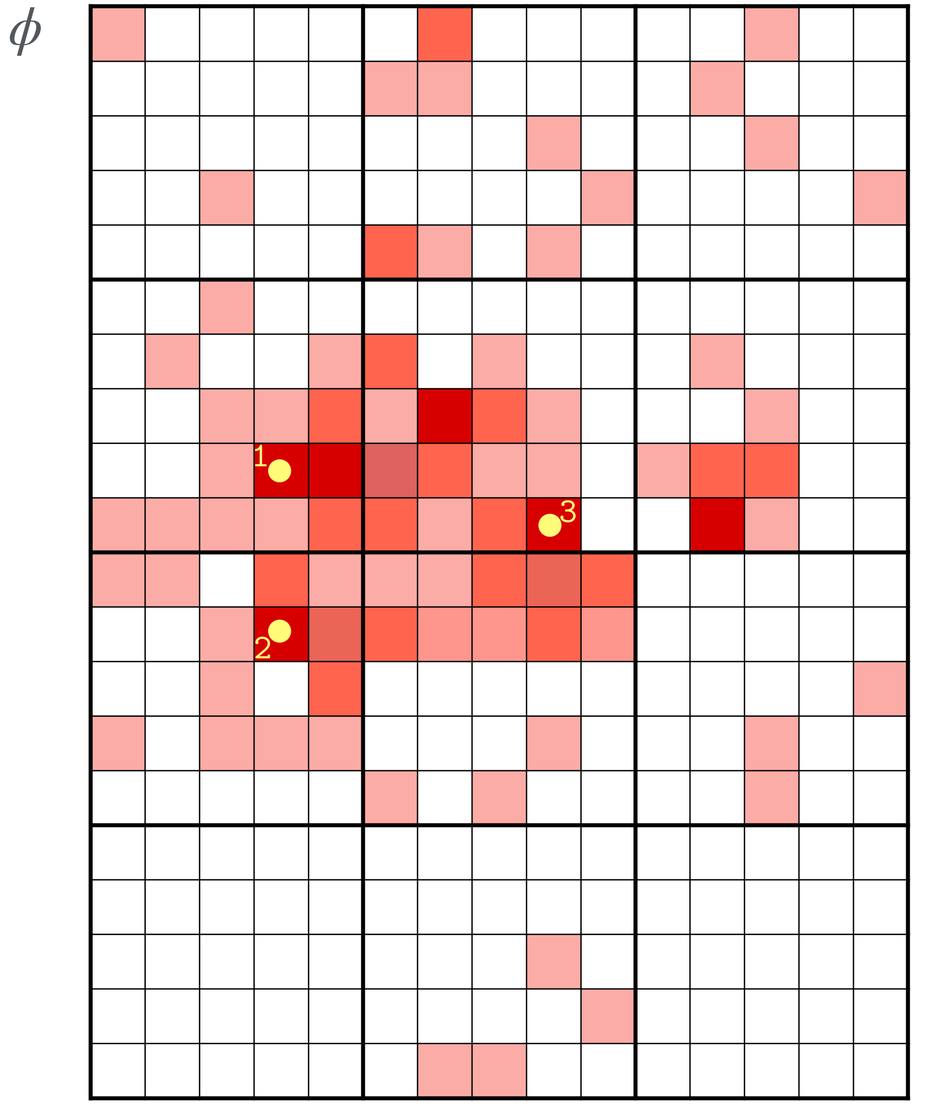


Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

ECAL Clusters

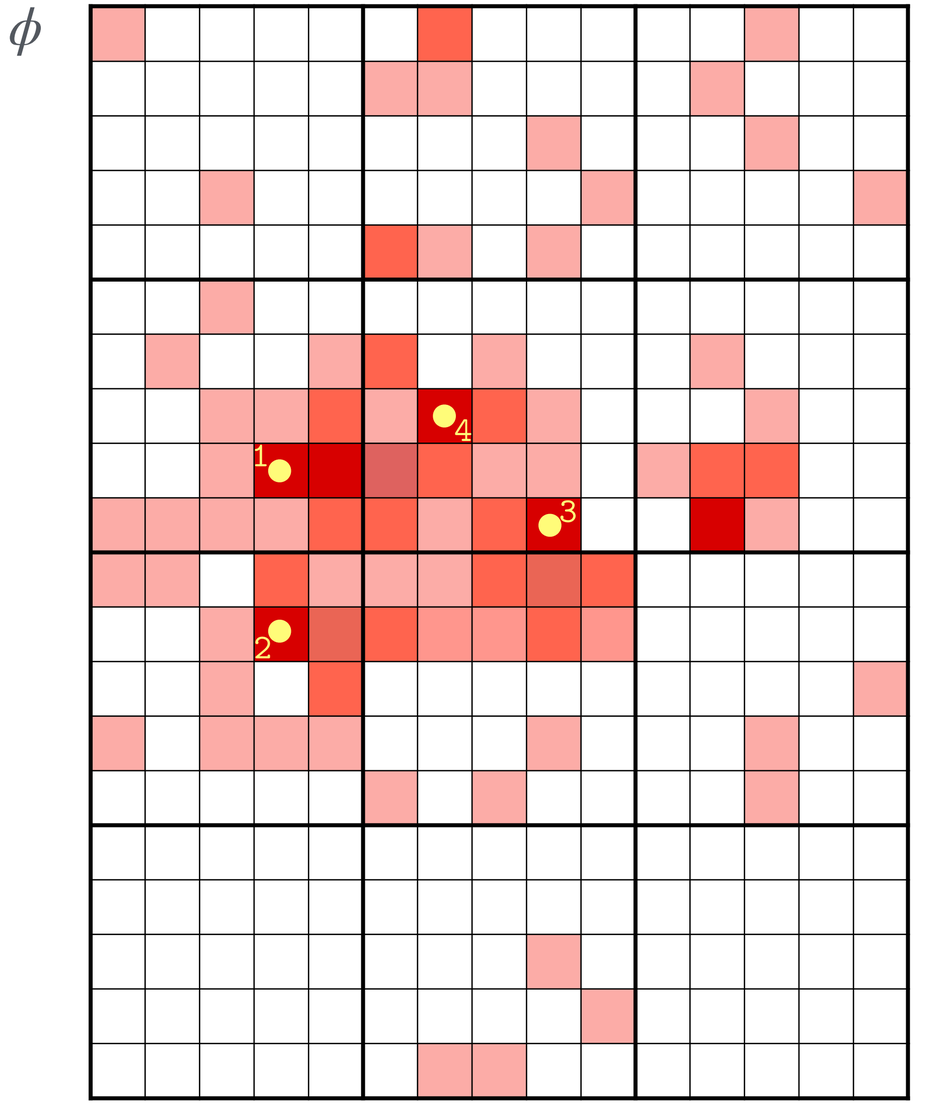


Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

ECAL Clusters

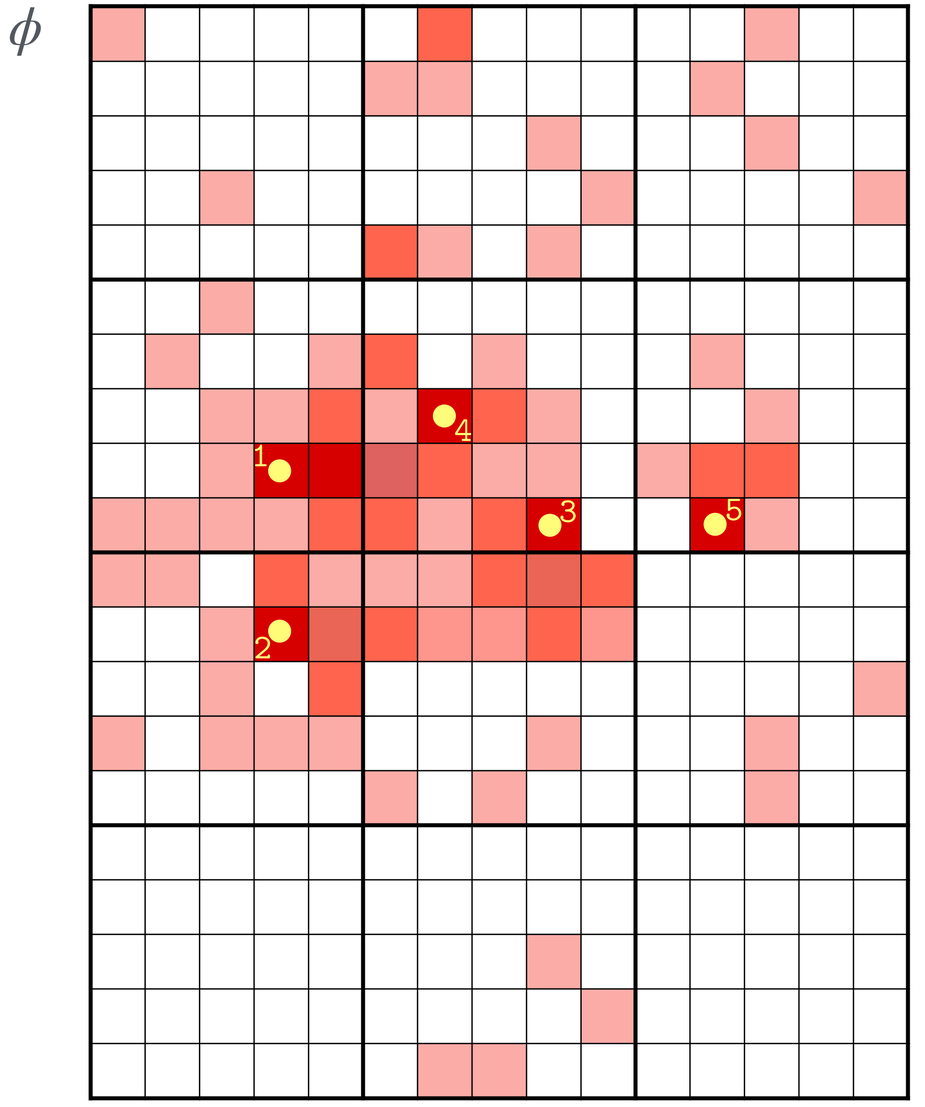


Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

ECAL Clusters



Calorimeter Clustering

Step 1: Identify Seeds

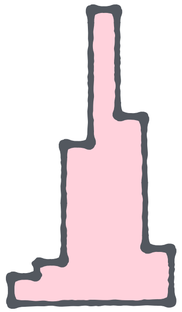
- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

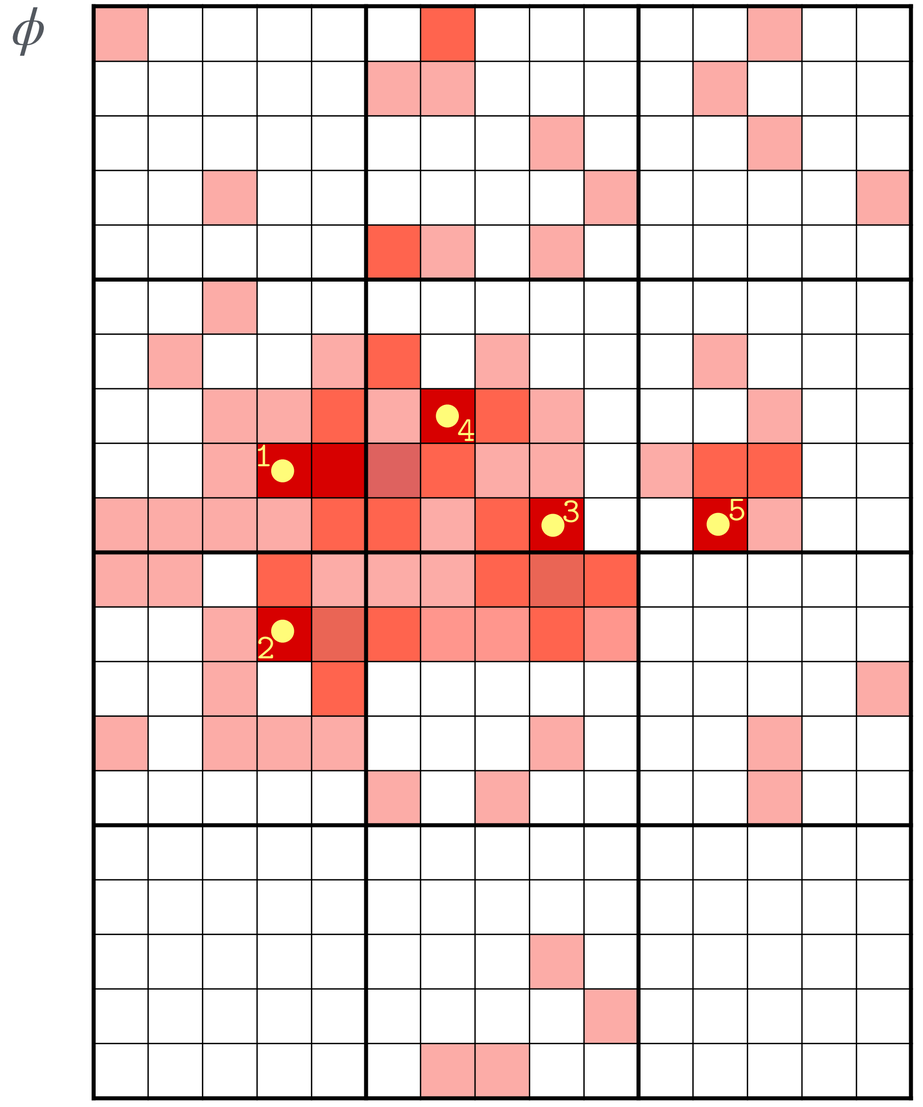
Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at least a corner in common with cell already in cluster





Calorimeter Clustering

Step 1: Identify Seeds

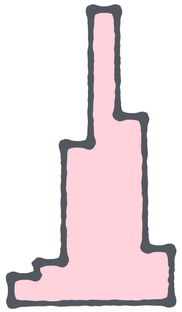
- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

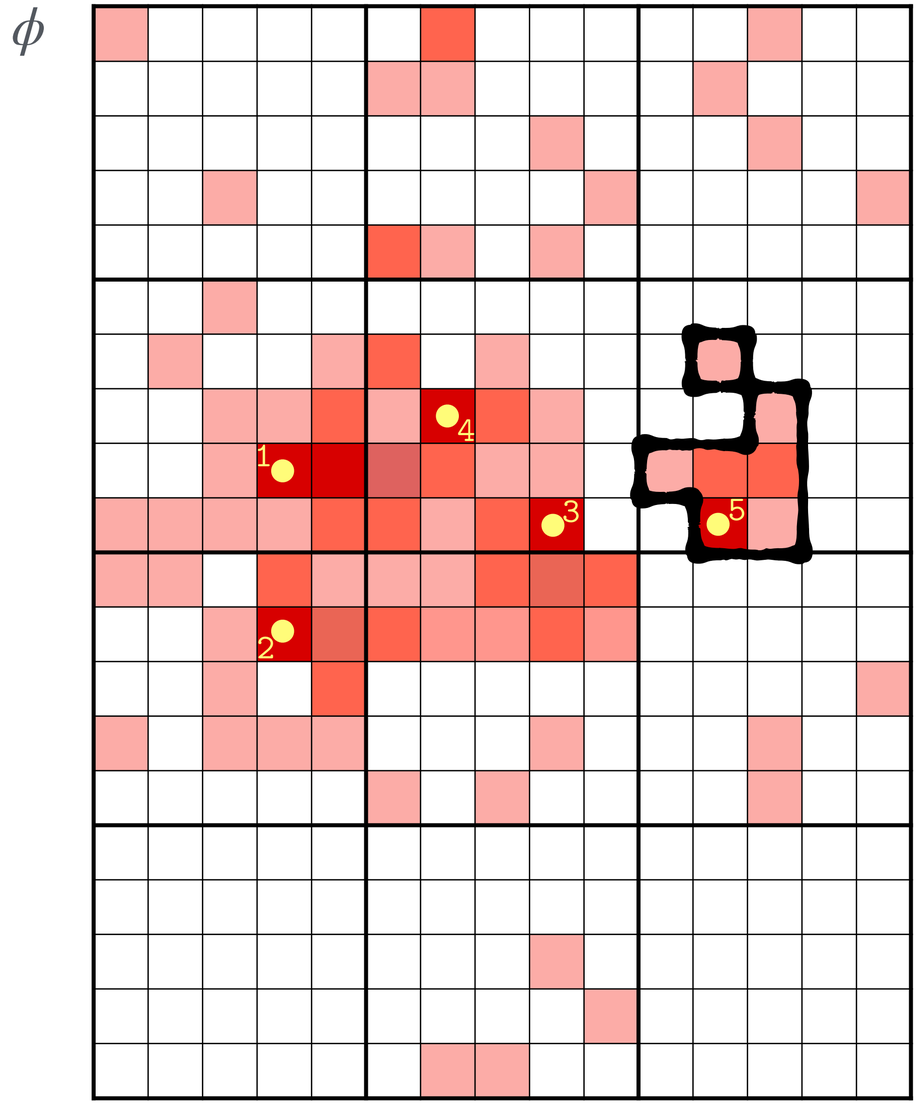
Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at least a corner in common with cell already in cluster





Calorimeter Clustering

Step 1: Identify Seeds

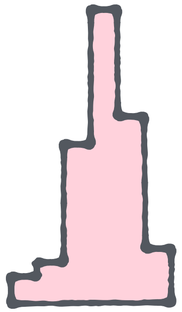
- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

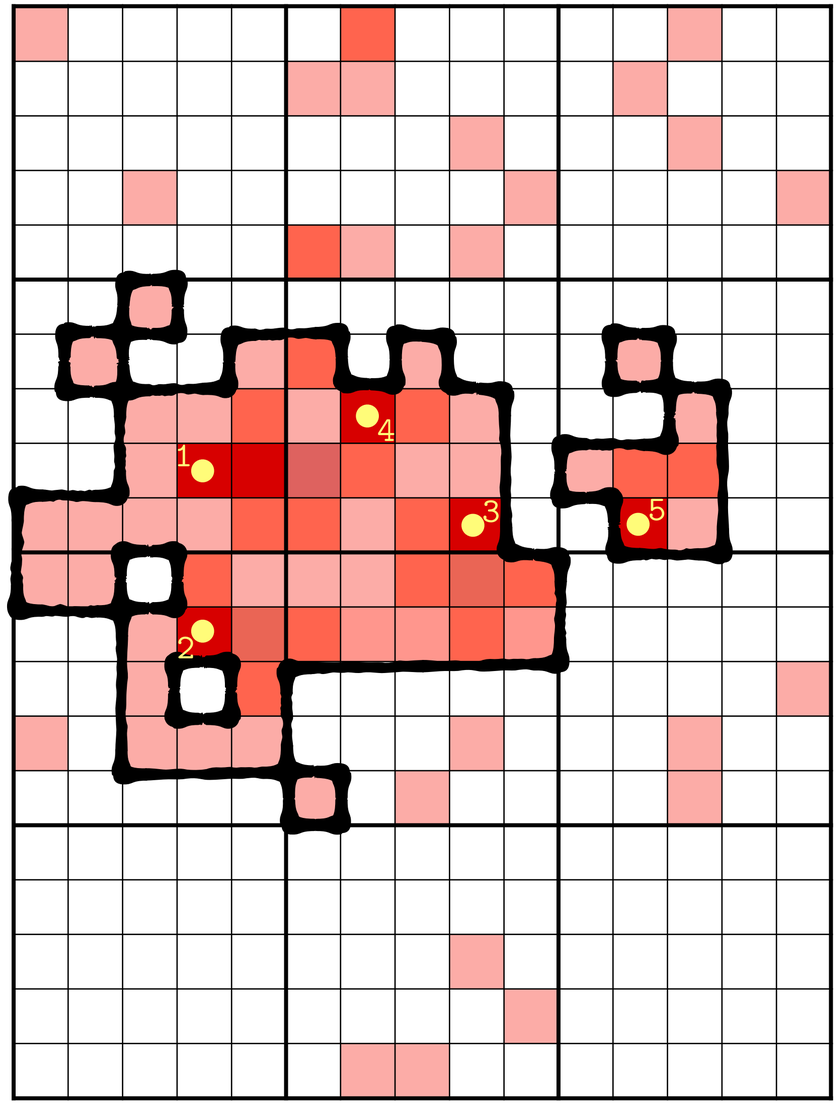
Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at least a corner in common with cell already in cluster





Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at least a corner in common with cell already in cluster

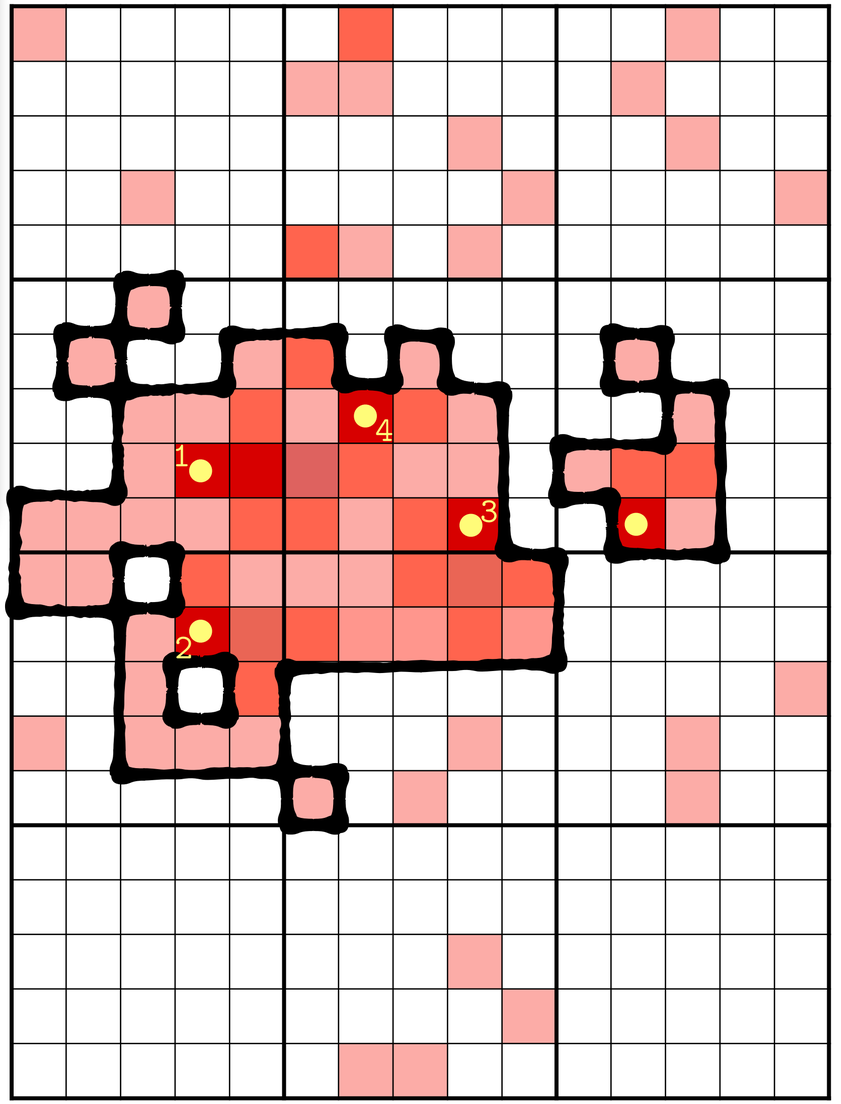
Step 3: Share energy in Topological Cluster

- Single seed clusters with N cells:
  - Energy is sum over all cells:

$$E = \sum_{i}^{N} E_i$$

  - Position is energy weighted sum:

$$\vec{\mu} = \frac{1}{E_{tot}} \sum_{i}^{N} E_I \vec{x}_i$$



Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at least a corner in common with cell already in cluster

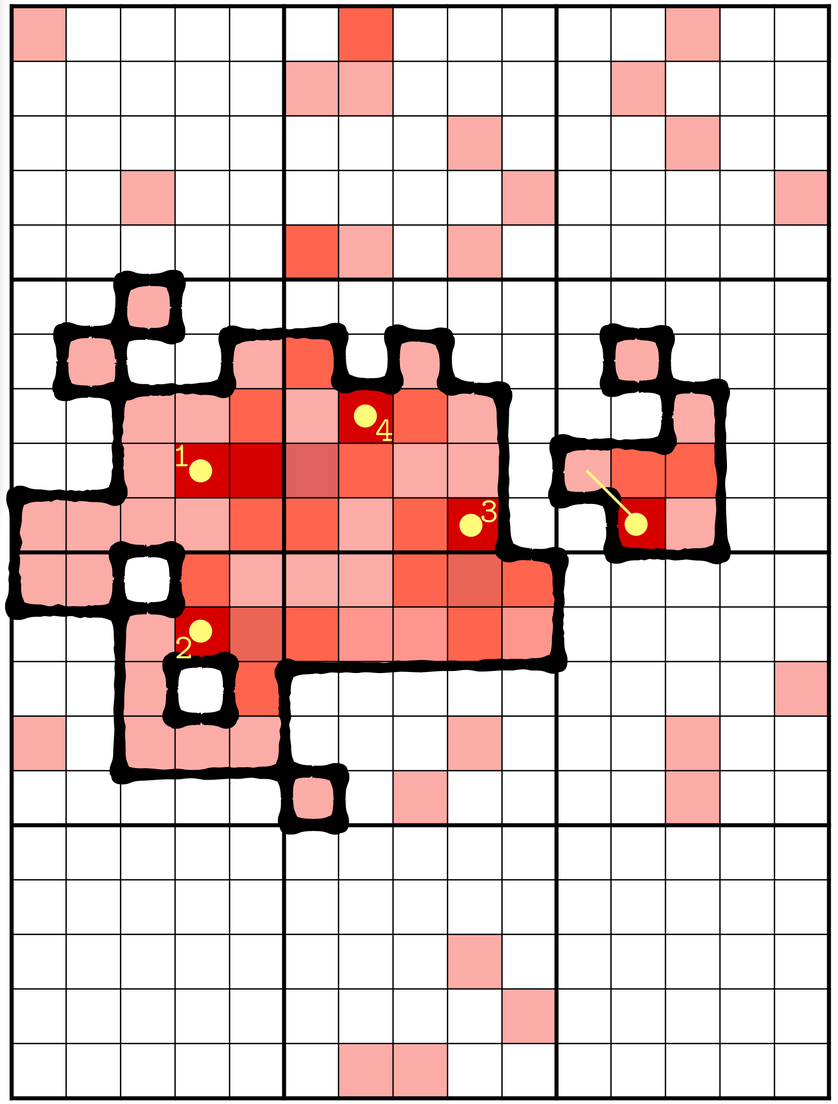
Step 3: Share energy in Topological Cluster

- Single seed clusters with N cells:
  - Energy is sum over all cells:

$$E = \sum_{i}^{N} E_i$$

  - Position is energy weighted sum:

$$\vec{\mu} = \frac{1}{E_{tot}} \sum_{i}^{N} E_I \vec{x}_i$$



Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at least a corner in common with cell already in cluster

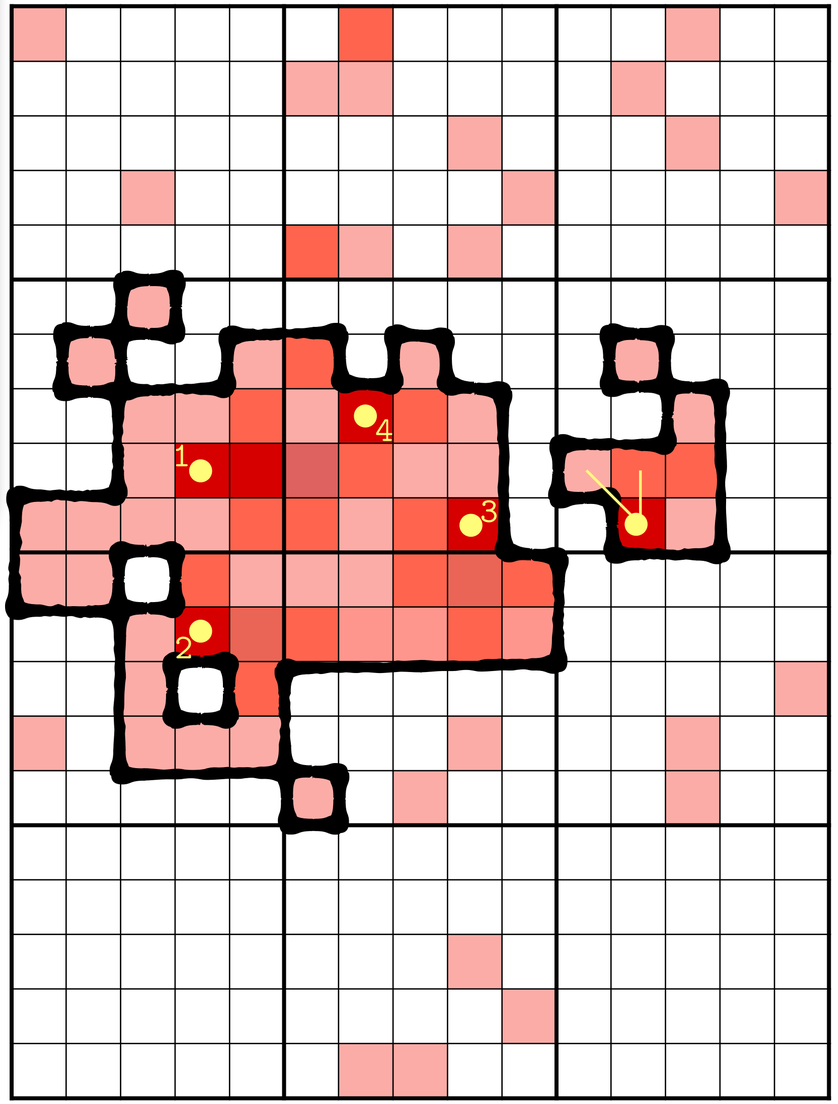
Step 3: Share energy in Topological Cluster

- Single seed clusters with N cells:
  - Energy is sum over all cells:

$$E = \sum_{i}^{N} E_i$$

  - Position is energy weighted sum:

$$\vec{\mu} = \frac{1}{E_{tot}} \sum_{i}^{N} E_I \vec{x}_i$$



Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at least a corner in common with cell already in cluster

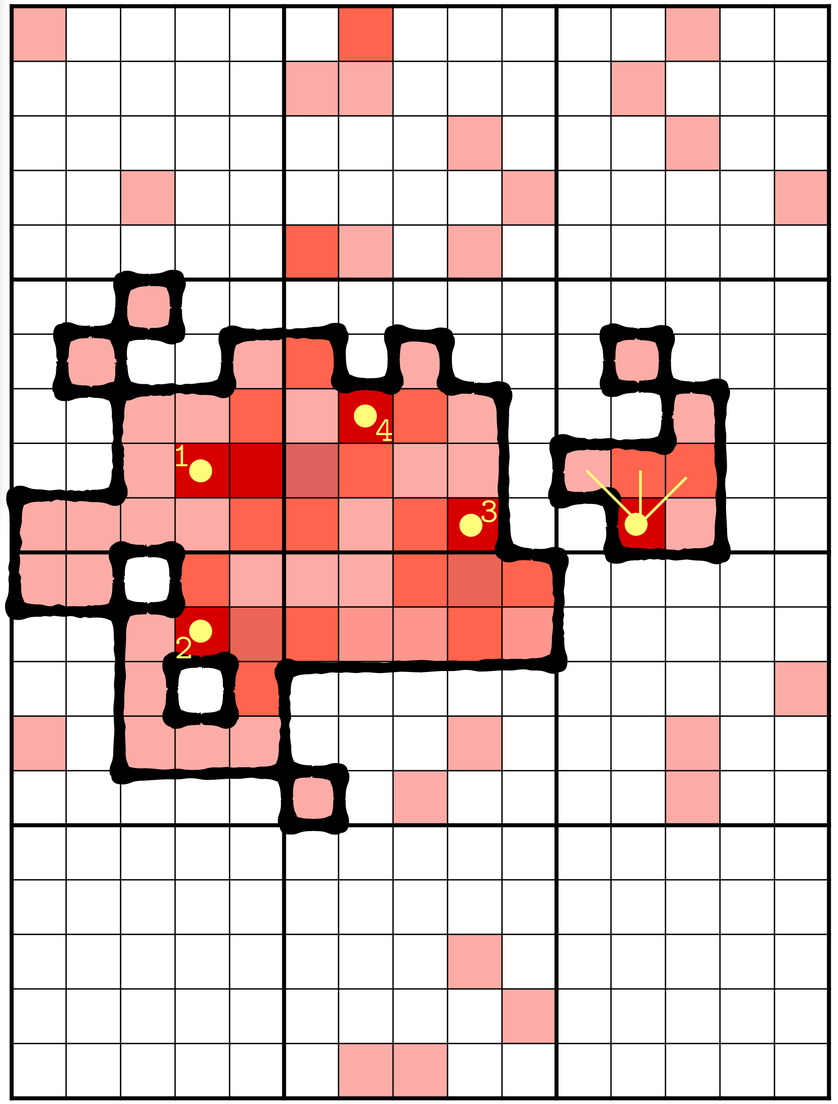
Step 3: Share energy in Topological Cluster

- Single seed clusters with N cells:
  - Energy is sum over all cells:

$$E = \sum_{i}^{N} E_i$$

  - Position is energy weighted sum:

$$\vec{\mu} = \frac{1}{E_{tot}} \sum_{i}^{N} E_I \vec{x}_i$$



Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at least a corner in common with cell already in cluster

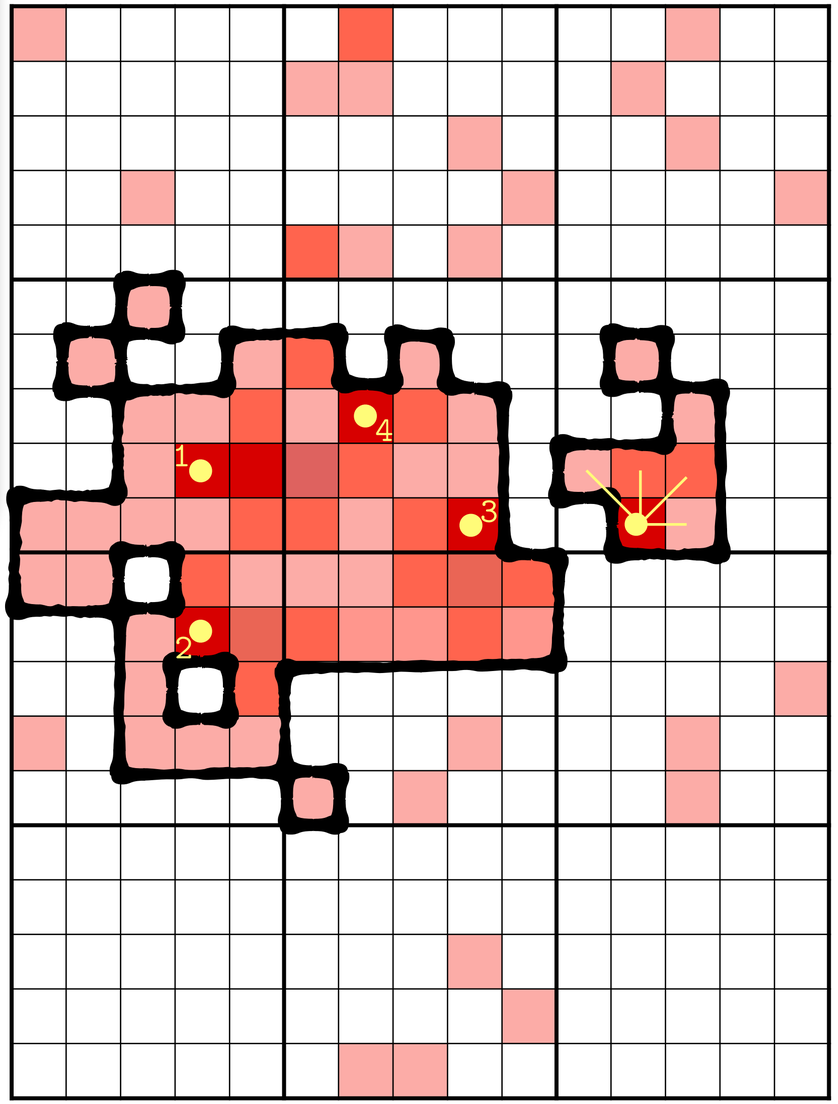
Step 3: Share energy in Topological Cluster

- Single seed clusters with N cells:
  - Energy is sum over all cells:

$$E = \sum_{i}^{N} E_i$$

  - Position is energy weighted sum:

$$\vec{\mu} = \frac{1}{E_{tot}} \sum_{i}^{N} E_I \vec{x}_i$$



Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at least a corner in common with cell already in cluster

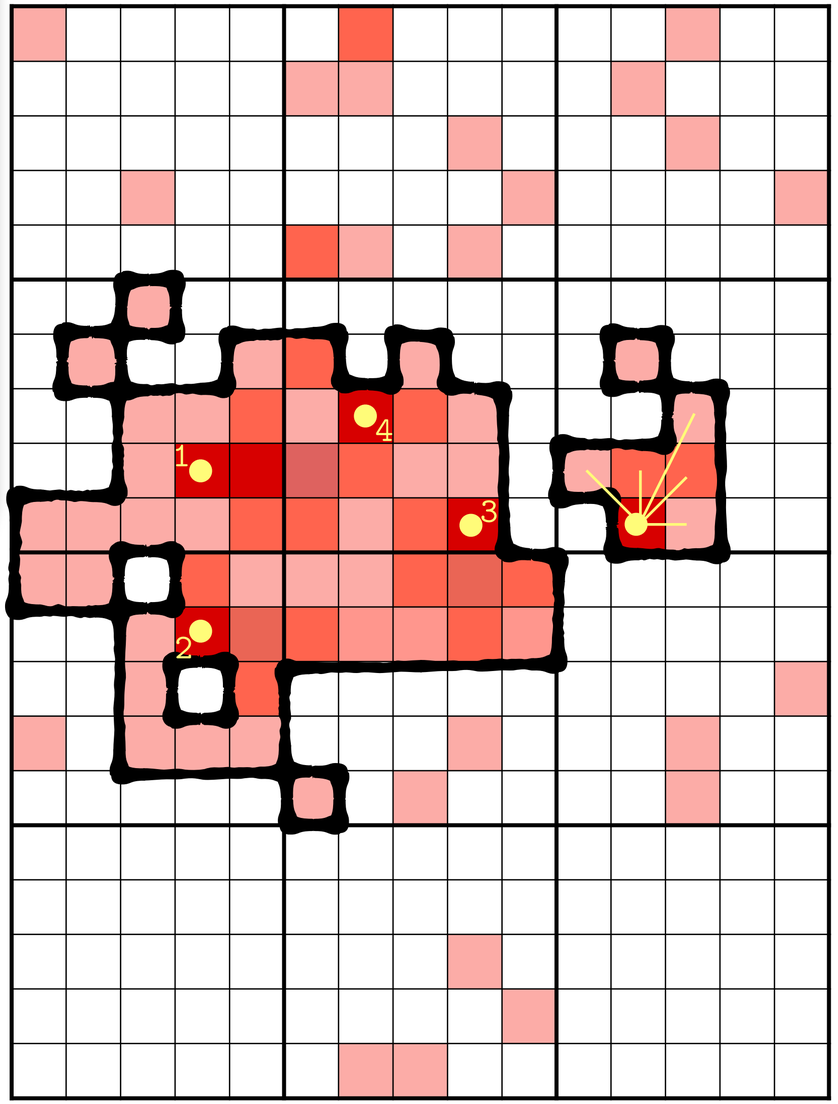
Step 3: Share energy in Topological Cluster

- Single seed clusters with N cells:
  - Energy is sum over all cells:

$$E = \sum_{i}^{N} E_i$$

  - Position is energy weighted sum:

$$\vec{\mu} = \frac{1}{E_{tot}} \sum_{i}^{N} E_I \vec{x}_i$$



Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at least a corner in common with cell already in cluster

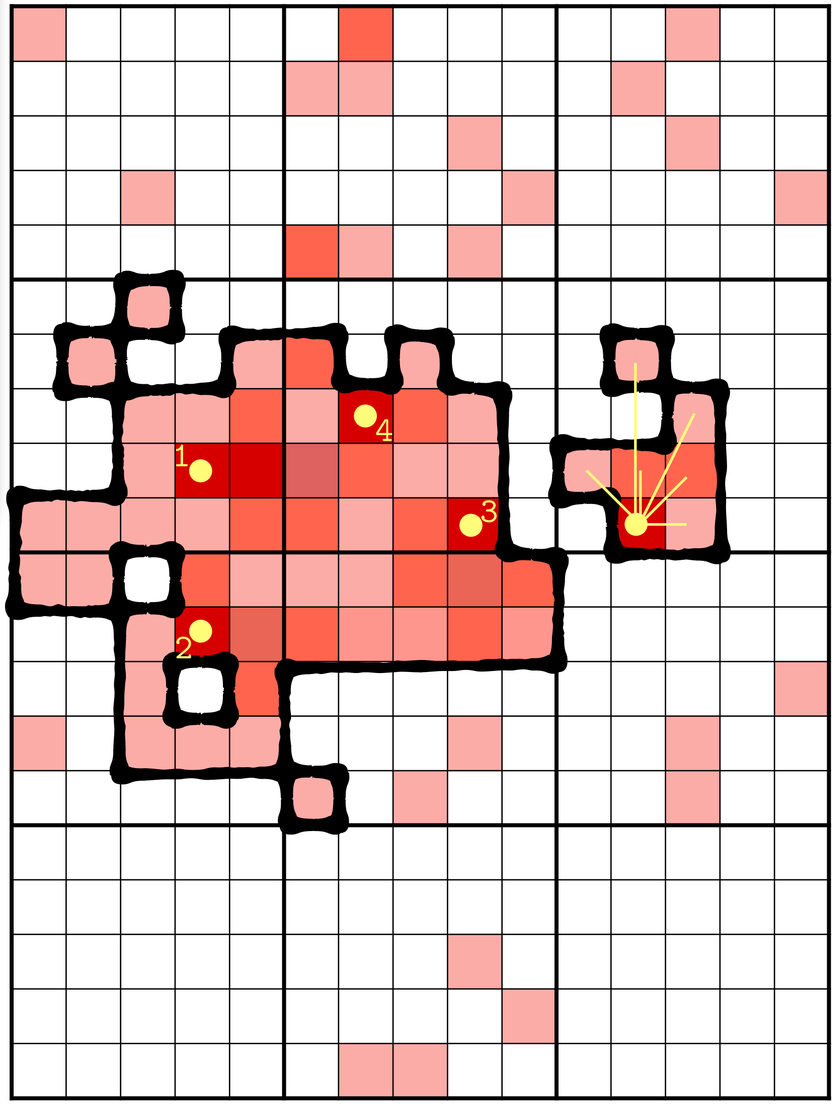
Step 3: Share energy in Topological Cluster

- Single seed clusters with N cells:
  - Energy is sum over all cells:

$$E = \sum_{i}^{N} E_i$$

  - Position is energy weighted sum:

$$\vec{\mu} = \frac{1}{E_{tot}} \sum_{i}^{N} E_I \vec{x}_i$$



Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at least a corner in common with cell already in cluster

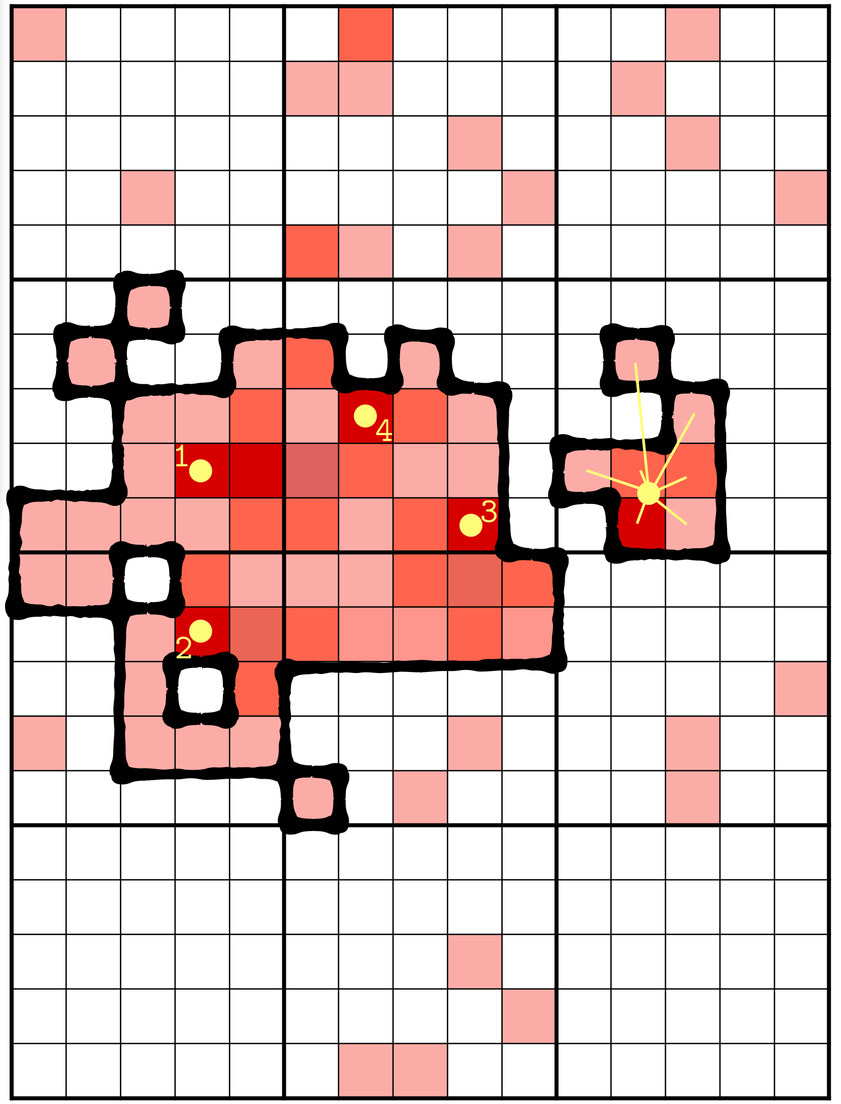
Step 3: Share energy in Topological Cluster

- Single seed clusters with N cells:
  - Energy is sum over all cells:

$$E = \sum_{i}^{N} E_i$$

  - Position is energy weighted sum:

$$\vec{\mu} = \frac{1}{E_{tot}} \sum_{i}^{N} E_I \vec{x}_i$$



Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

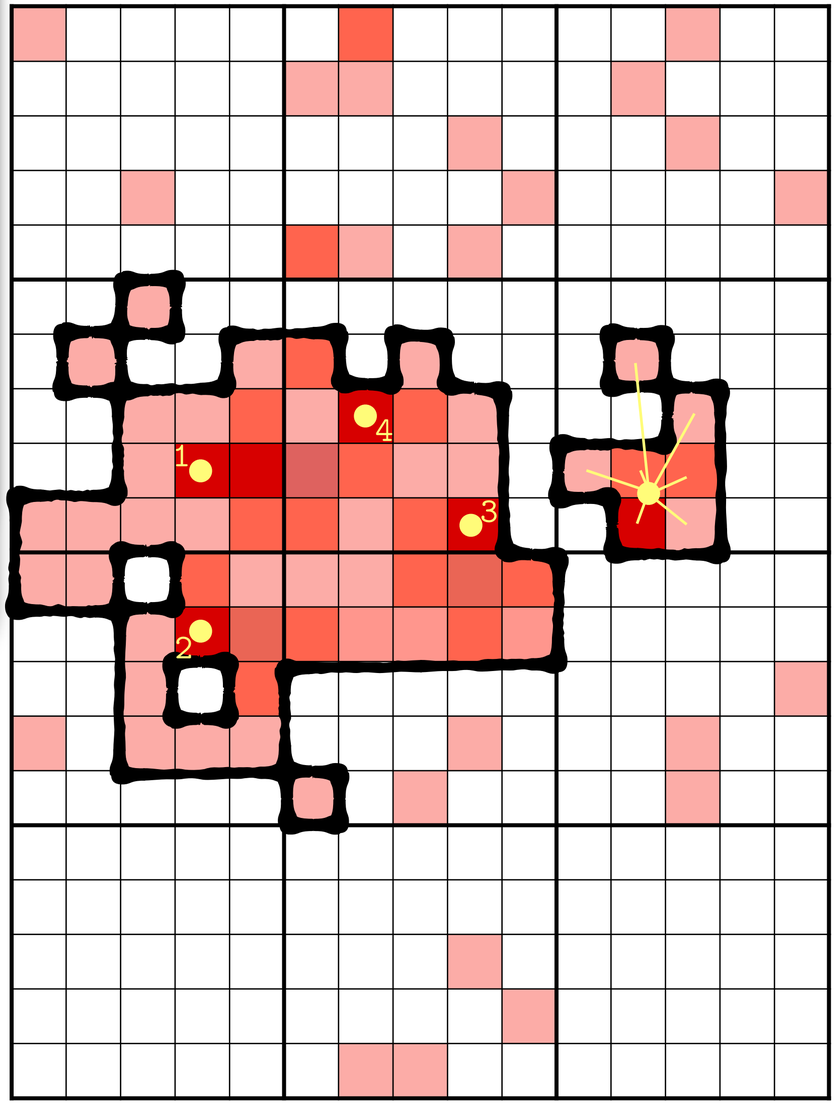
- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at least a corner in common with cell already in cluster

Step 3: Share energy in Topological Cluster

- Multi M seed clusters with N cells:
- Rule: Seed never shares its energy
- Gaussian-mixture Model
  - Start: cluster = seed
  - calc energy sharing for each cell
  - Recompute energy
  - Recompute position
  - Iterate until cluster energies & positions converge: change by
    - less than 10^{-8} or 50 steps



Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at l… corner in …

Step 3: Share energy in Topological Cluster

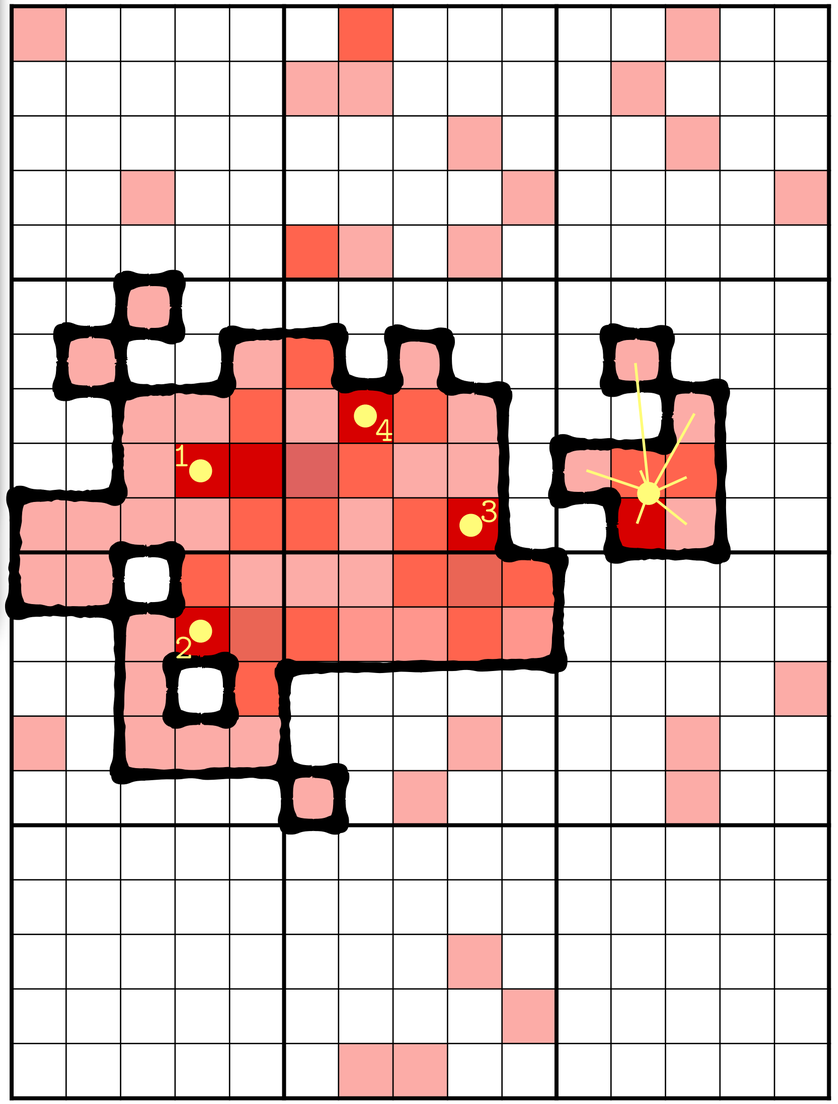
- Share energy of i-th cell with j-th seed according to the fraction:

$$f_{ij} = \frac{A_i \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}{\sum_k^N A_k \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}$$

- Where the amplitude A_i is a fit parameter

Step 3: Share energy in Topological Cluster

- Multi M seed clusters with N cells:
- Rule: Seed never shares its energy
- Gaussian-mixture Model
  - Start: cluster = seed
  - calc energy sharing for each cell
  - Recompute energy
  - Recompute position
  - Iterate until cluster energies & positions converge: change by
    - less than 10^{-8} or 50 steps



Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at l… corner in …

Step 3: Share energy in Topological Cluster

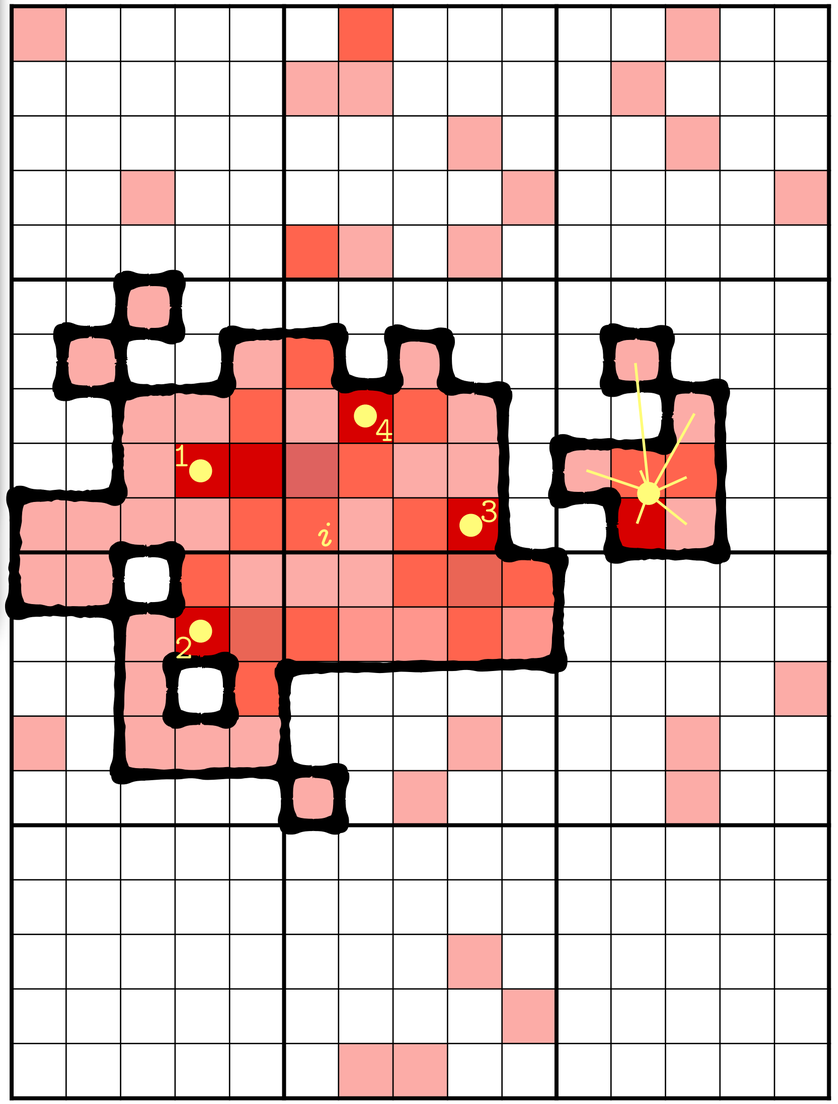
- Multi M seed clusters with N cells:
- Rule: Seed never shares its energy
- Gaussian-mixture Model
  - Start: cluster = seed
  - calc energy sharing for each cell
  - Recompute energy
  - Recompute position
  - Iterate until cluster energies & positions converge: change by
    - less than 10^{-8} or 50 steps

Step 3: Share energy in Topological Cluster

- Share energy of i-th cell with j-th seed according to the fraction:

$$f_{ij} = \frac{A_i \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}{\sum_k^N A_k \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}$$

- Where the amplitude A_i is a fit parameter



Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at l… corner in …

Step 3: Share energy in Topological Cluster

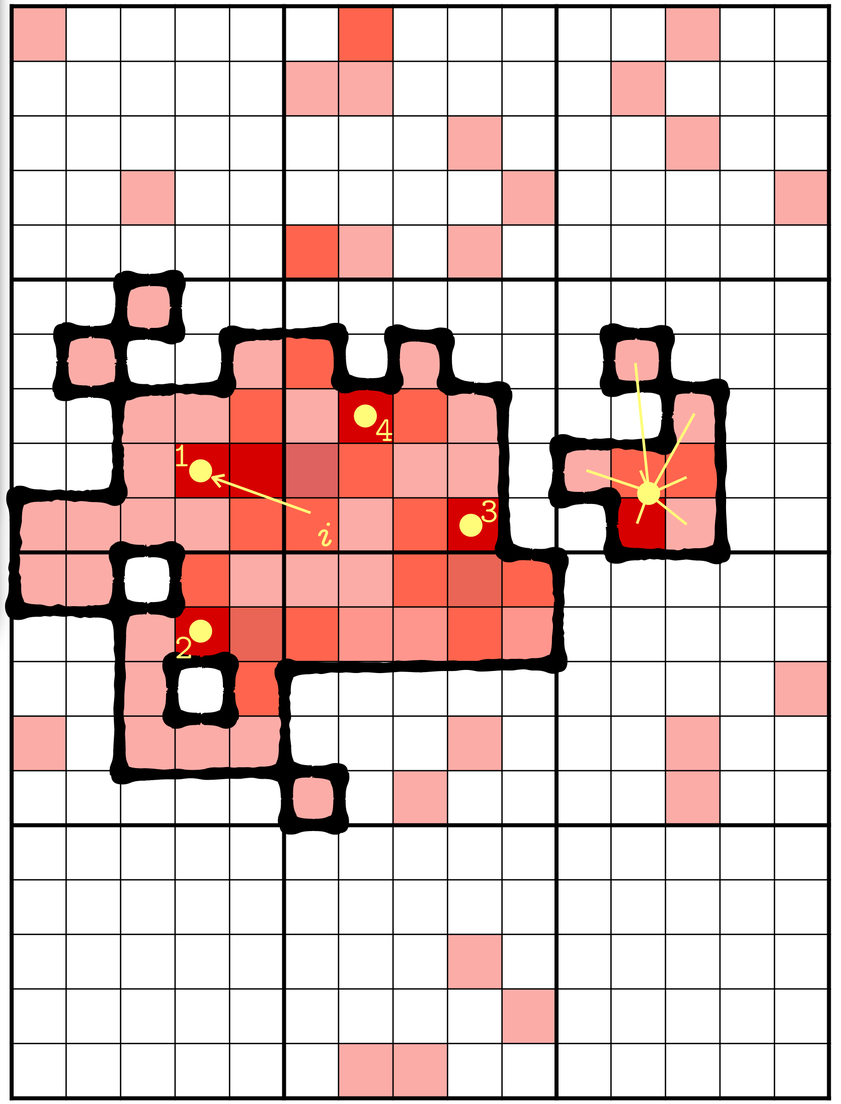
- Multi M seed clusters with N cells:
- Rule: Seed never shares its energy
- Gaussian-mixture Model
  - Start: cluster = seed
  - calc energy sharing for each cell
  - Recompute energy
  - Recompute position
  - Iterate until cluster energies & positions converge: change by
    - less than 10^{-8} or 50 steps

Step 3: Share energy in Topological Cluster

- Share energy of i-th cell with j-th seed according to the fraction:

$$f_{ij} = \frac{A_i \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}{\sum_k^N A_k \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}$$

- Where the amplitude A_i is a fit parameter



Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at l… corner in …

Step 3: Share energy in Topological Cluster

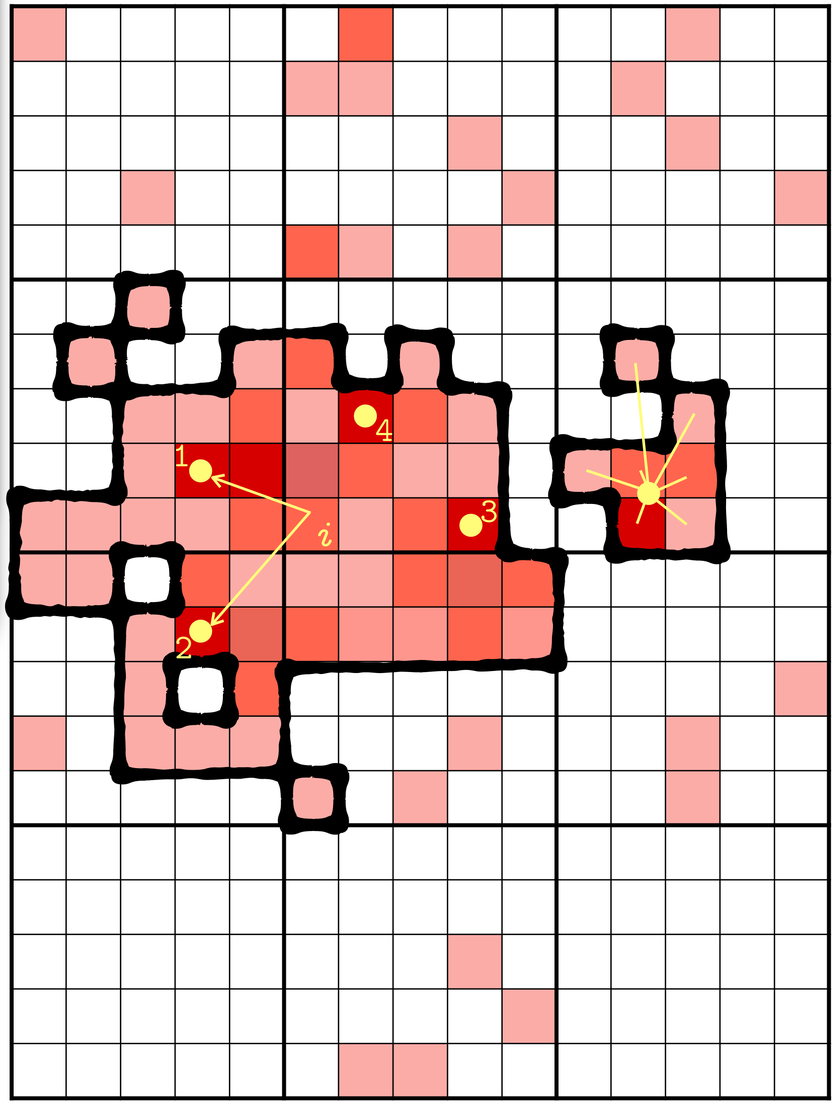
- Multi M seed clusters with N cells:
- Rule: Seed never shares its energy
- Gaussian-mixture Model
  - Start: cluster = seed
  - calc energy sharing for each cell
  - Recompute energy
  - Recompute position
  - Iterate until cluster energies & positions converge: change by
    - less than 10^{-8} or 50 steps

Step 3: Share energy in Topological Cluster

- Share energy of i-th cell with j-th seed according to the fraction:

$$f_{ij} = \frac{A_i \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}{\sum_k^N A_k \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}$$

- Where the amplitude A_i is a fit parameter



Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at l… corner in …

Step 3: Share energy in Topological Cluster

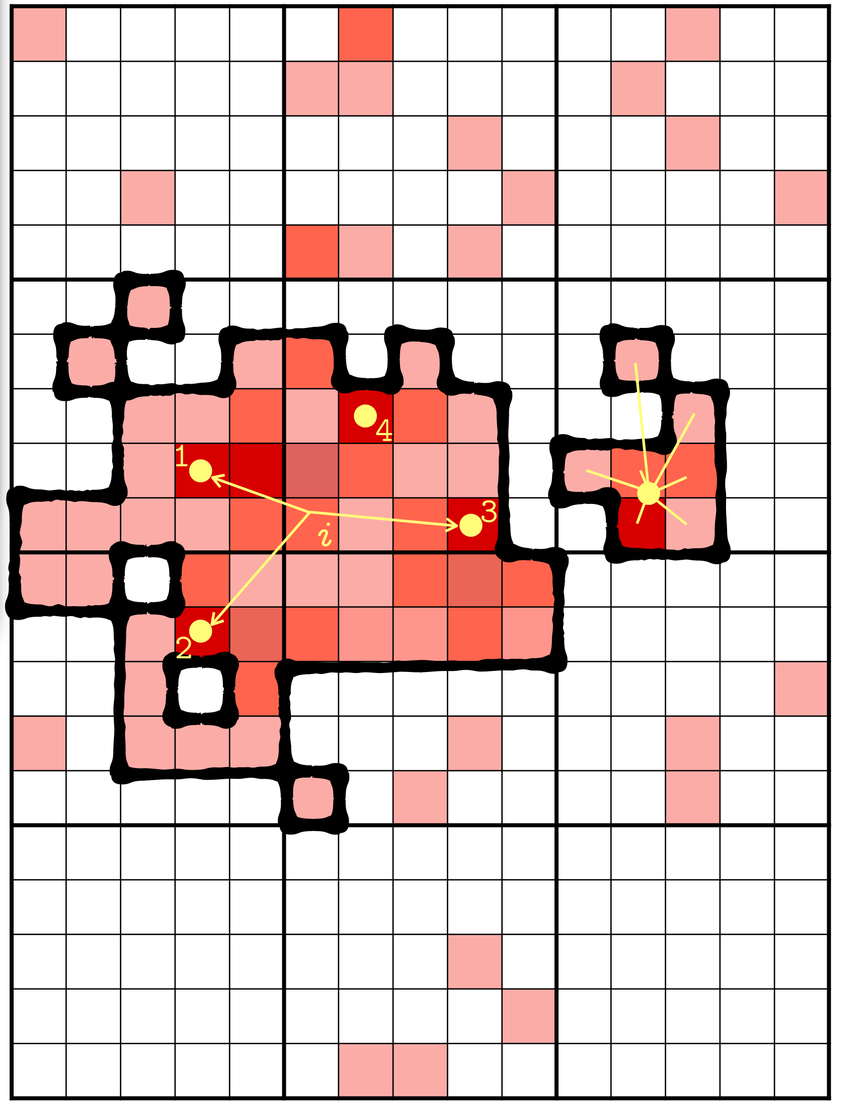
- Multi M seed clusters with N cells:
- Rule: Seed never shares its energy
- Gaussian-mixture Model
  - Start: cluster = seed
  - calc energy sharing for each cell
  - Recompute energy
  - Recompute position
  - Iterate until cluster energies & positions converge: change by
    - less than 10^{-8} or 50 steps

Step 3: Share energy in Topological Cluster

- Share energy of i-th cell with j-th seed according to the fraction:

$$f_{ij} = \frac{A_i \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}{\sum_k^N A_k \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}$$

- Where the amplitude A_i is a fit parameter



Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at l… corner in …

Step 3: Share energy in Topological Cluster

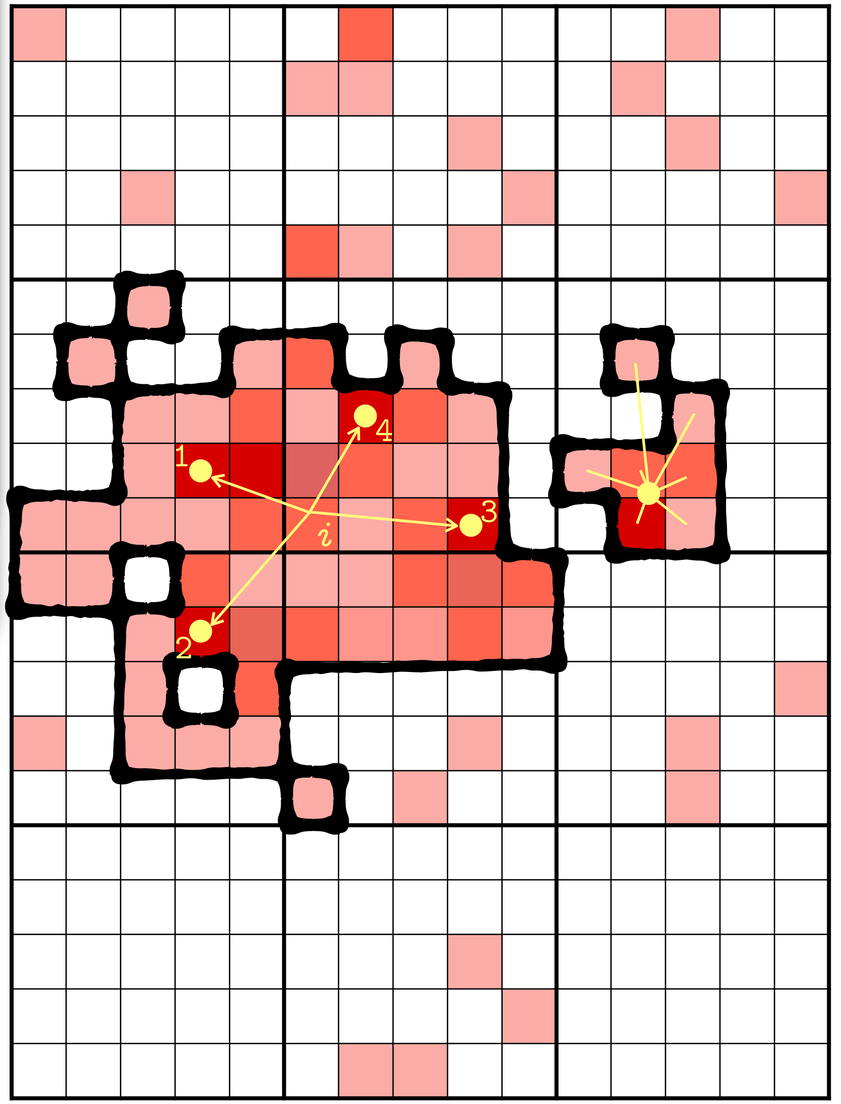
- Multi M seed clusters with N cells:
- Rule: Seed never shares its energy
- Gaussian-mixture Model
  - Start: cluster = seed
  - calc energy sharing for each cell
  - Recompute energy
  - Recompute position
  - Iterate until cluster energies & positions converge: change by
    - less than 10^{-8} or 50 steps

Step 3: Share energy in Topological Cluster

- Share energy of i-th cell with j-th seed according to the fraction:

$$f_{ij} = \frac{A_i \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}{\sum_k^N A_k \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}$$

- Where the amplitude A_i is a fit parameter



Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at l… corner in …

Step 3: Share energy in Topological Cluster

- Multi M seed clusters with N cells:
- Rule: Seed never shares its energy
- Gaussian-mixture Model
  - Start: cluster = seed
  - calc energy sharing for each cell
  - Recompute energy
  - Recompute position
  - Iterate until cluster energies & positions converge: change by
    - less than 10^{-8} or 50 steps

Step 3: Share energy in Topological Cluster

- Share energy of i-th cell with j-th seed according to the fraction:

$$f_{ij} = \frac{A_i \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}{\sum_k^N A_k \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}$$

- Where the amplitude A_i is a fit parameter

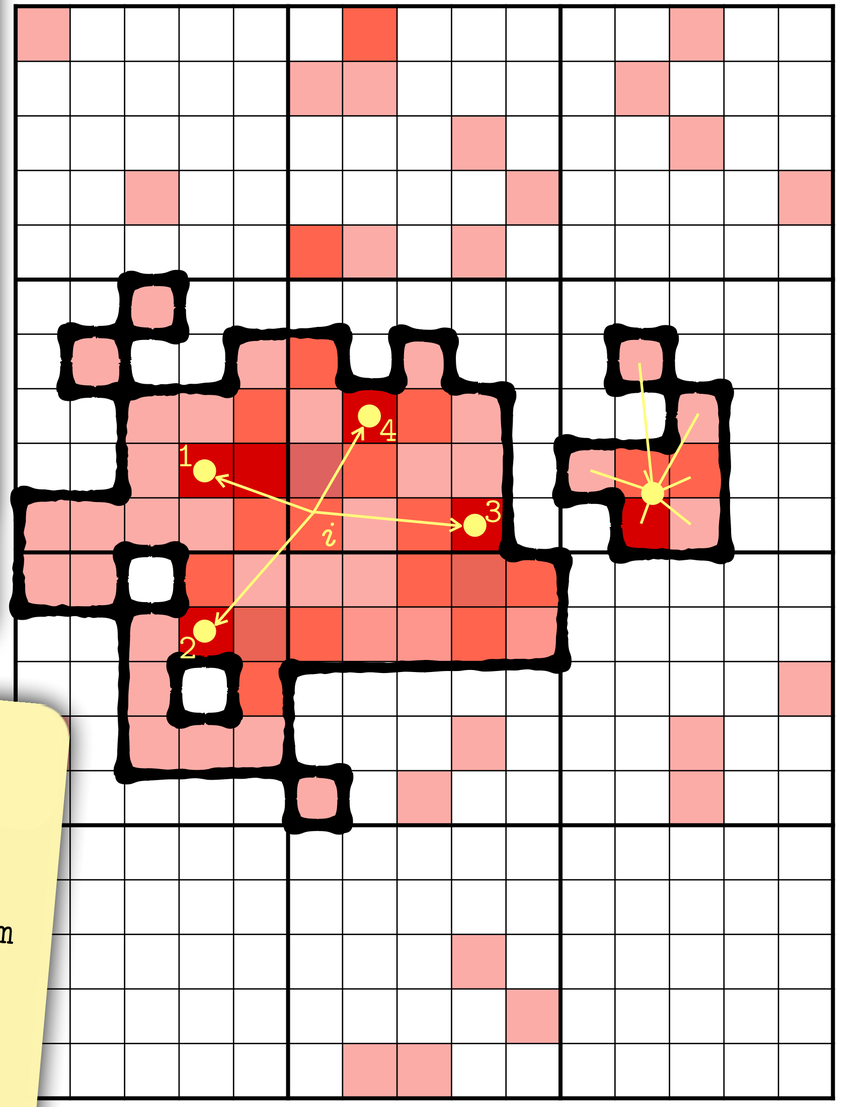
Step 3: Share energy in Topological Cluster

- Recompute amplitude of i-th cell as energy weighted shared fraction from the M seed clusters:

$$A_i = \sum_j^M f_{ij} E_j$$

- Recompute position as weighted fraction

$$\vec{d}_i^{\,new} = \sum_k^M f_{ik} \vec{d}_k^{\,old}$$



Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at l… corner in …

Step 3: Share energy in Topological Cluster

- Multi M seed clusters with N cells:
- Rule: Seed never shares its energy
- Gaussian-mixture Model
  - Start: cluster = seed
  - calc energy sharing for each cell
  - Recompute energy
  - Recompute position
  - Iterate until cluster energies & positions converge: change by
    - less than 10^{-8} or 50 steps

Step 3: Share energy in Topological Cluster

- Share energy of i-th cell with j-th seed according to the fraction:

$$f_{ij} = \frac{A_i \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}{\sum_k^N A_k \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}$$

- Where the amplitude A_i is a fit parameter

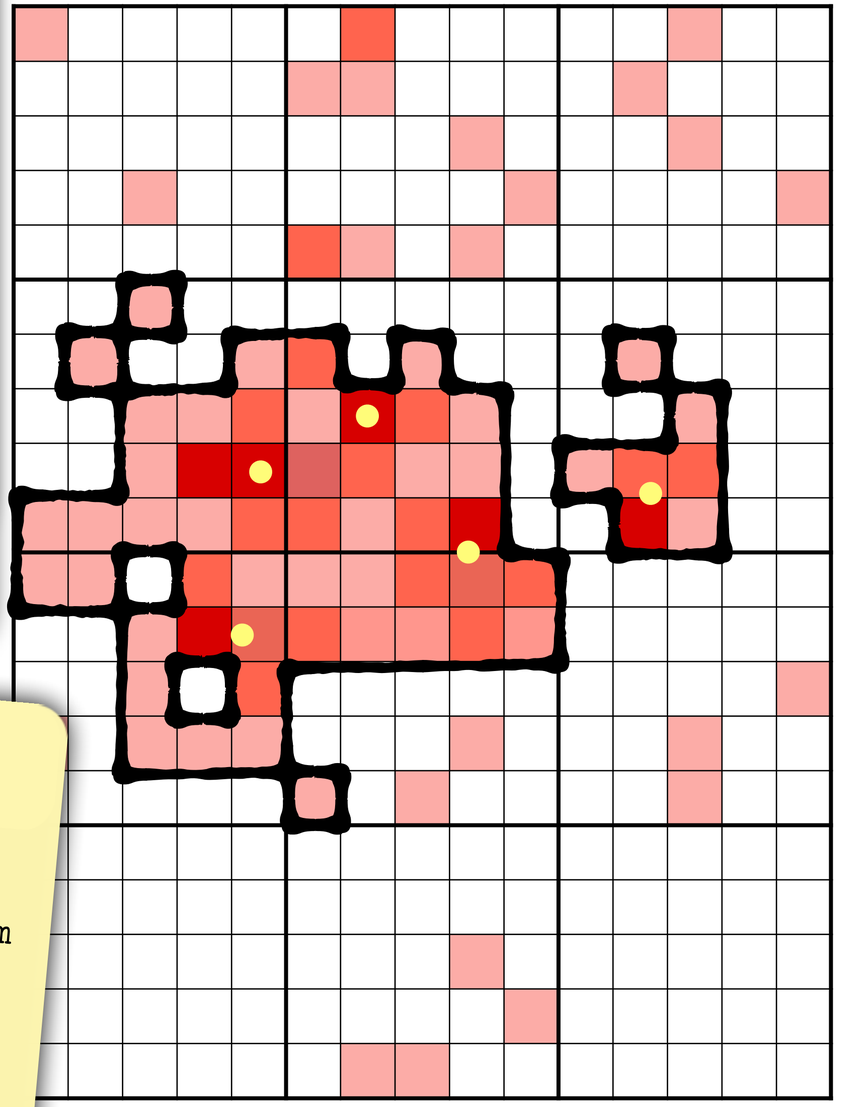
Step 3: Share energy in Topological Cluster

- Recompute amplitude of i-th cell as energy weighted shared fraction from the M seed clusters:

$$A_i = \sum_j^M f_{ij} E_j$$

- Recompute position as weighted fraction

$$\vec{d}_i^{\,new} = \sum_k^M f_{ik} \vec{d}_k^{\,old}$$



Calorimeter Clustering

Step 1: Identify Seeds

- Any cell that is above some energy threshold
- Local maximum: energy larger than all neighbouring cells

Step 2: Form Topological Clusters

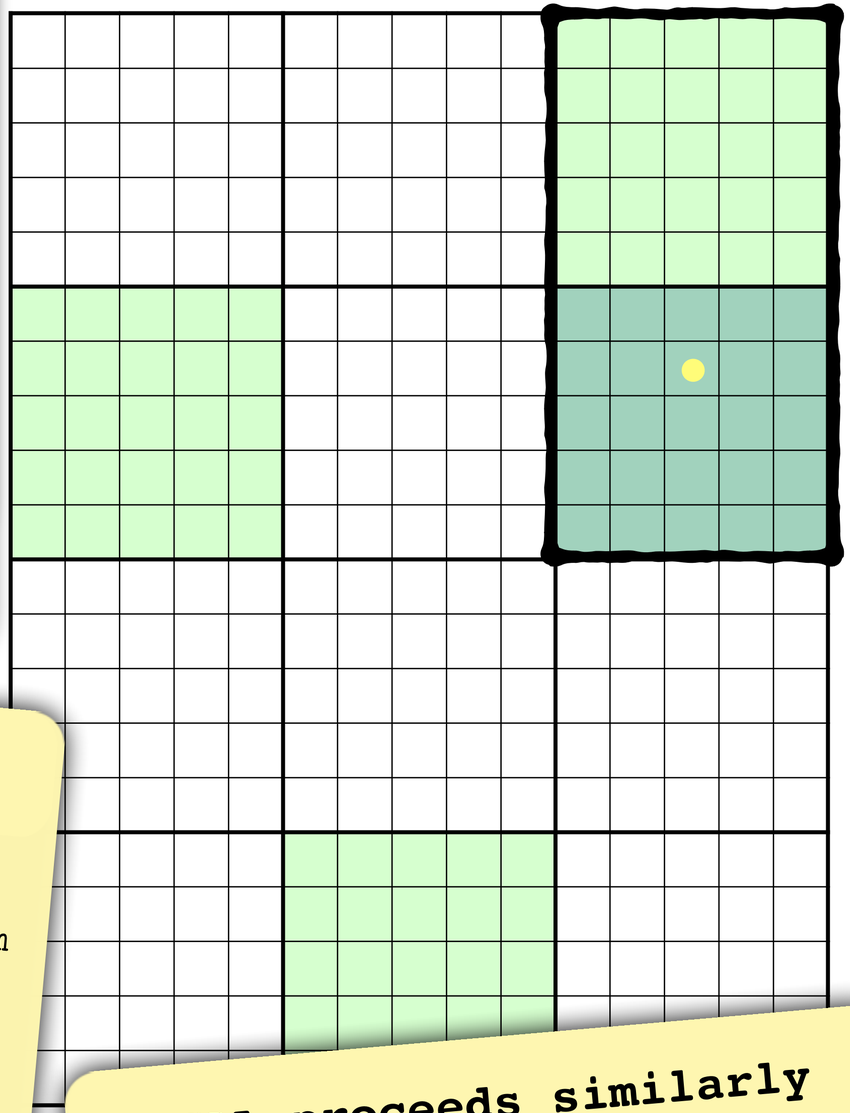
- Grown from seeds
- Consider only cells with energy above:

$$E + 2\,\sigma_{noise}(E)$$

- Aggregate all cells with at l… corner in …

Step 3: Share energy in Topological Cluster

- Multi M seed clusters with N cells:
- Rule: Seed never shares its energy
- Gaussian-mixture Model
  - Start: cluster = seed
  - calc energy sharing for each cell
  - Recompute energy
  - Recompute position
  - Iterate until cluster energies & positions converge: change by
    - less than 10^{-8} or 50 steps



Step 3: Share energy in Topological Cluster

- Share energy of i-th cell with j-th seed according to the fraction:

$$f_{ij} = \frac{A_i \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}{\sum_k^N A_k \exp\left[-\frac{|\vec{d}_{ij}|^2}{\sigma^2}\right]}$$

- Where the amplitude A_i is a fit parameter

Step 3: Share energy in Topological Cluster

- Recompute amplitude of i-th cell as energy weighted shared fraction from the M seed clusters:

$$A_i = \sum_j^M f_{ij} E_j$$

- Recompute position as weighted fraction

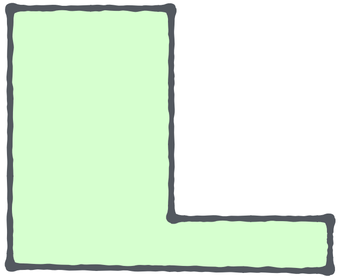
$$\vec{d}_i^{\,new} = \sum_k^M f_{ik} \vec{d}_k^{\,old}$$

HCAL proceeds similarly

Just less granularity and different parameters

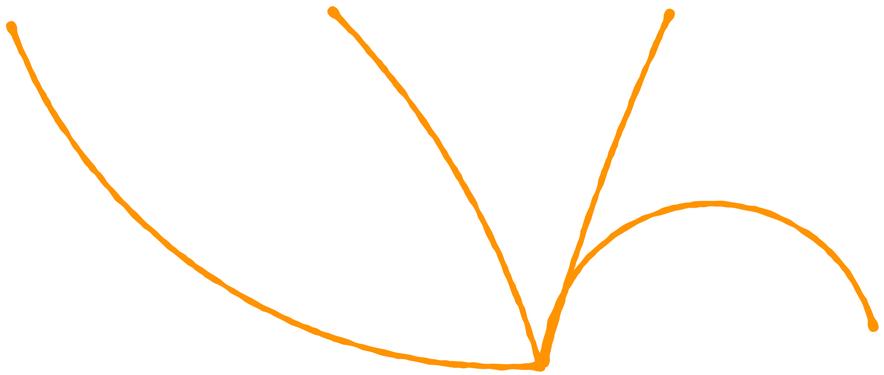
Next: Link across detectors

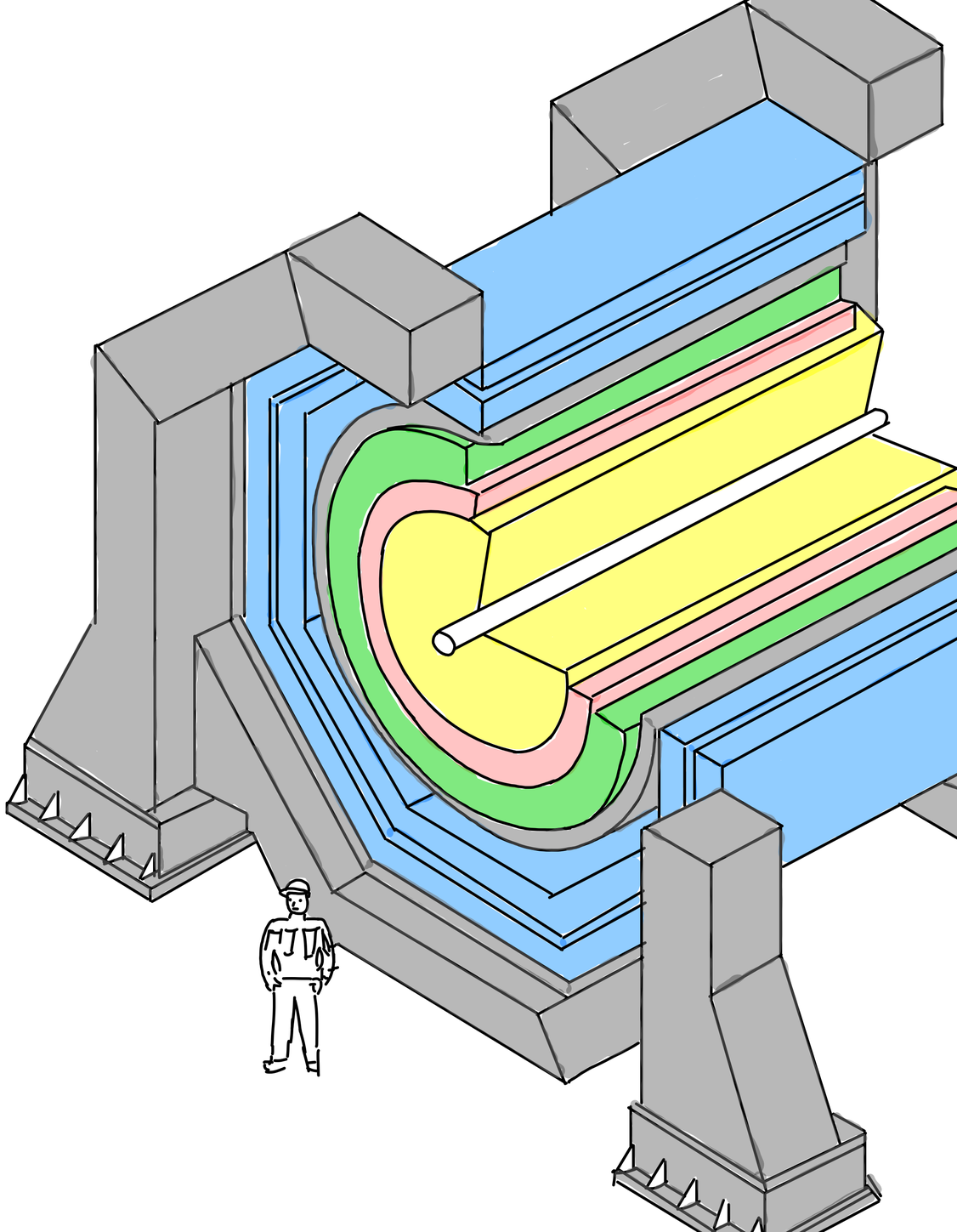
HCAL Clusters



ECAL Clusters

Tracks



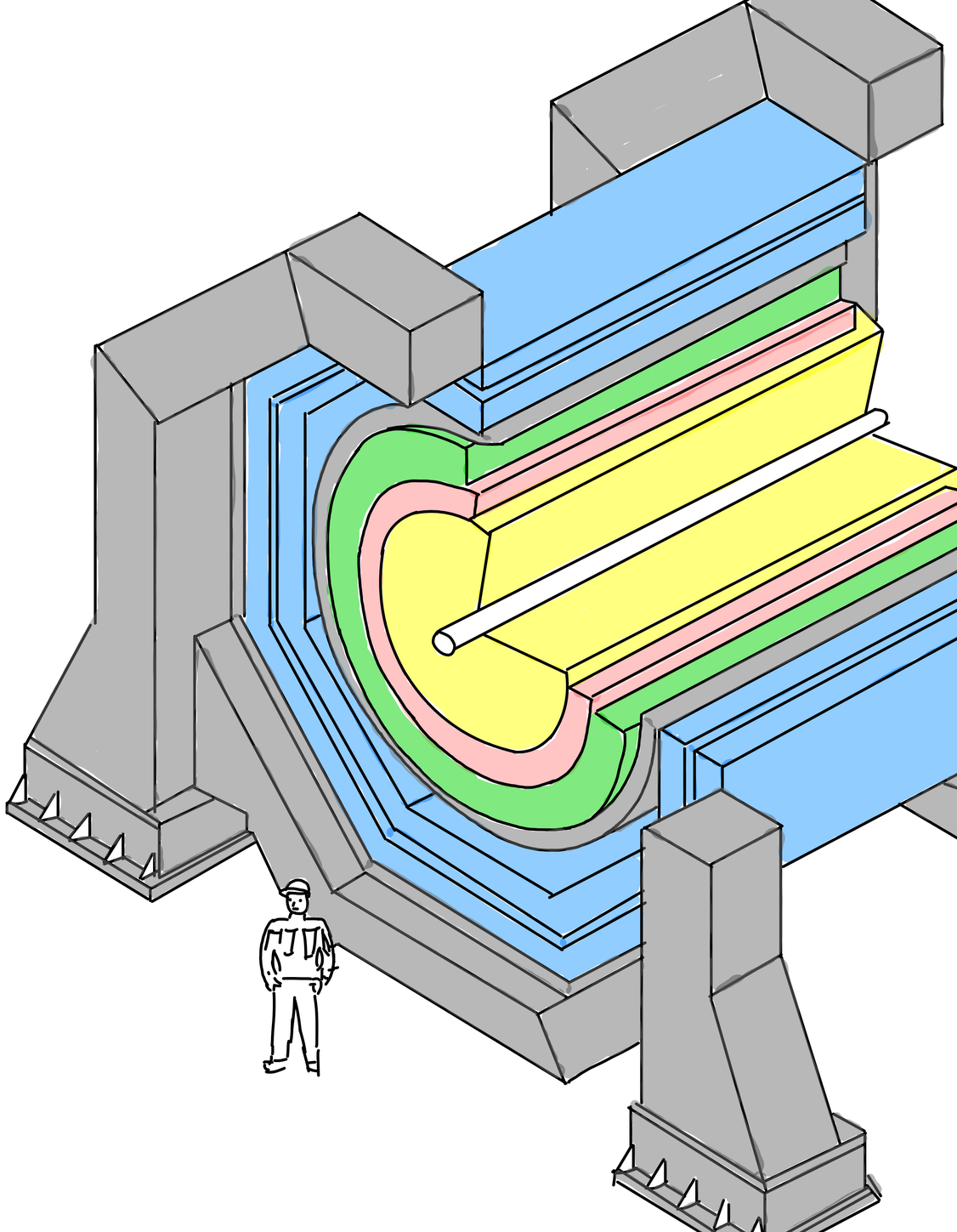


Next: Link across detectors

HCAL Clusters

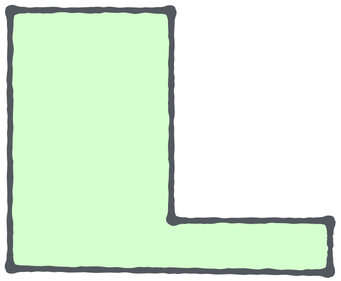
ECAL Clusters

Tracks



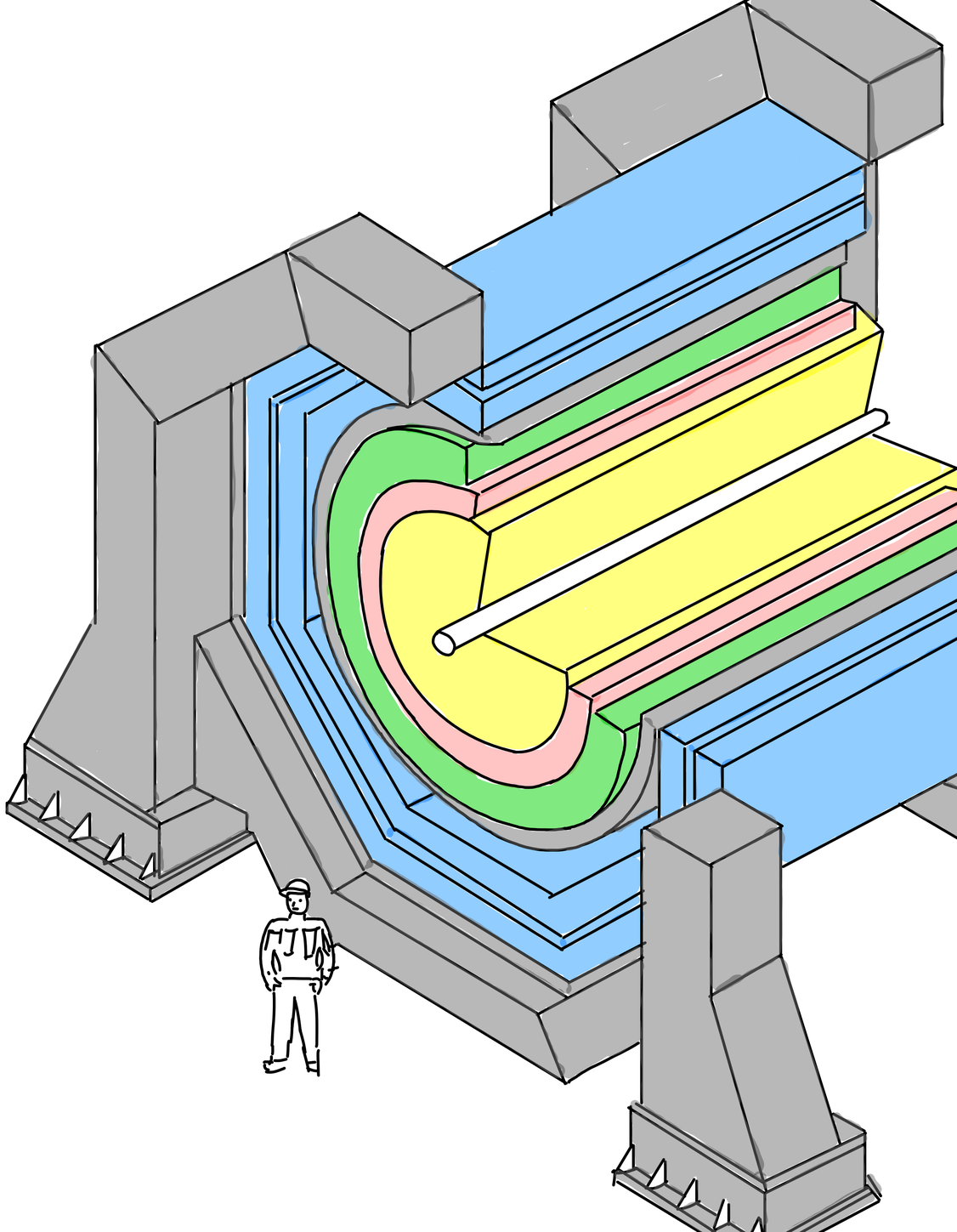
Next: Link across detectors

HCAL Clusters



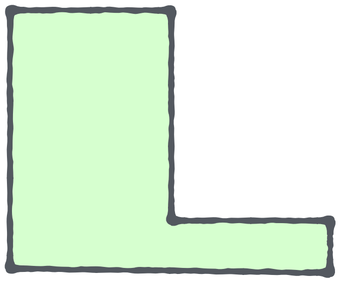
ECAL Clusters

Tracks



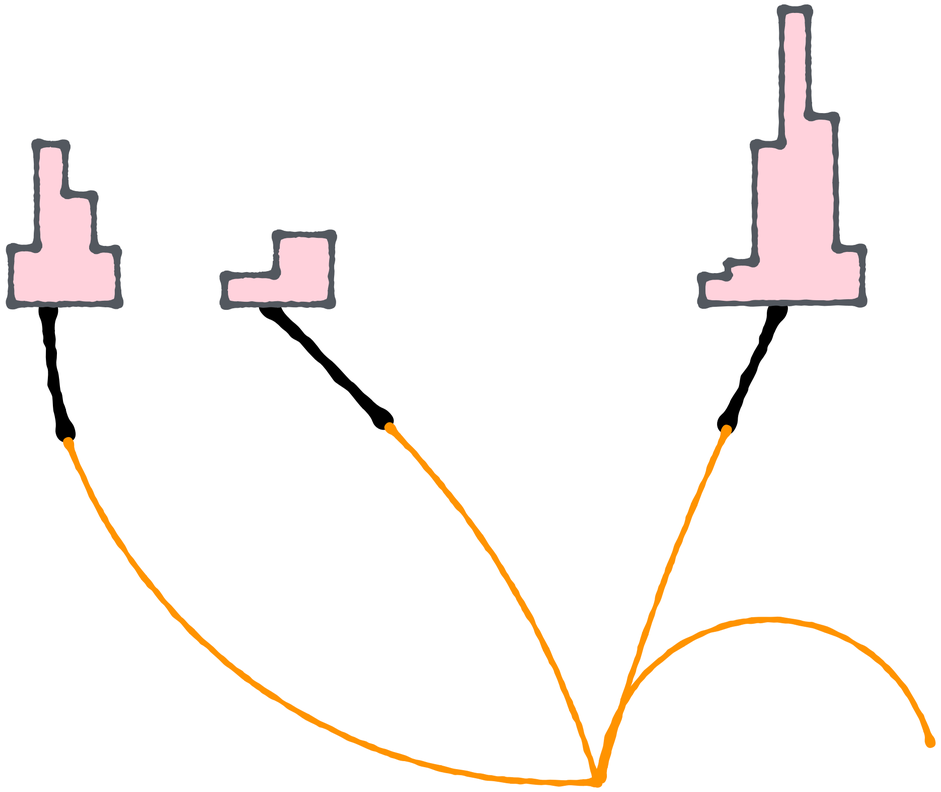
Next: Link across detectors

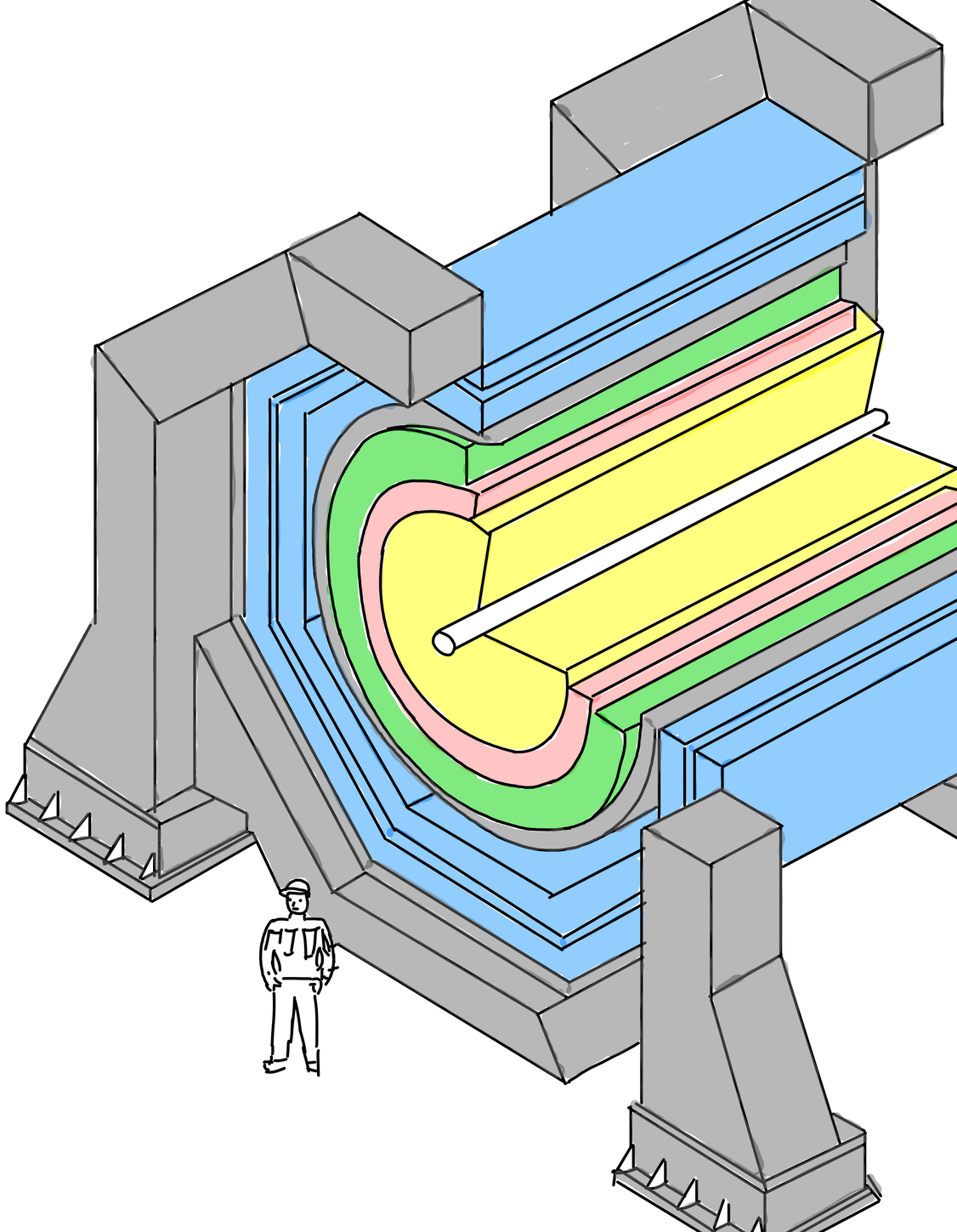
HCAL Clusters



ECAL Clusters

Tracks



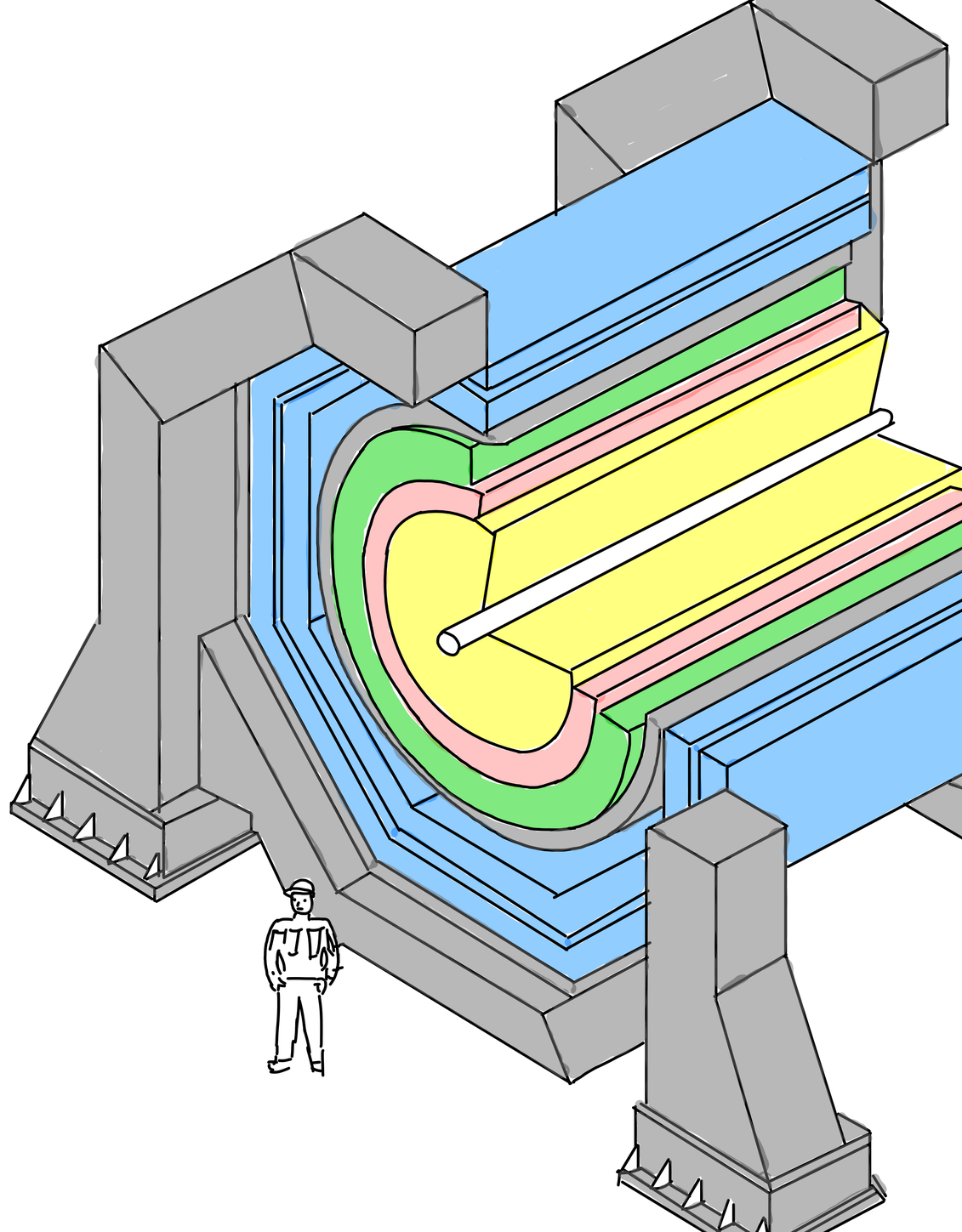


Next: Link across detectors

HCAL Clusters

ECAL Clusters

Tracks

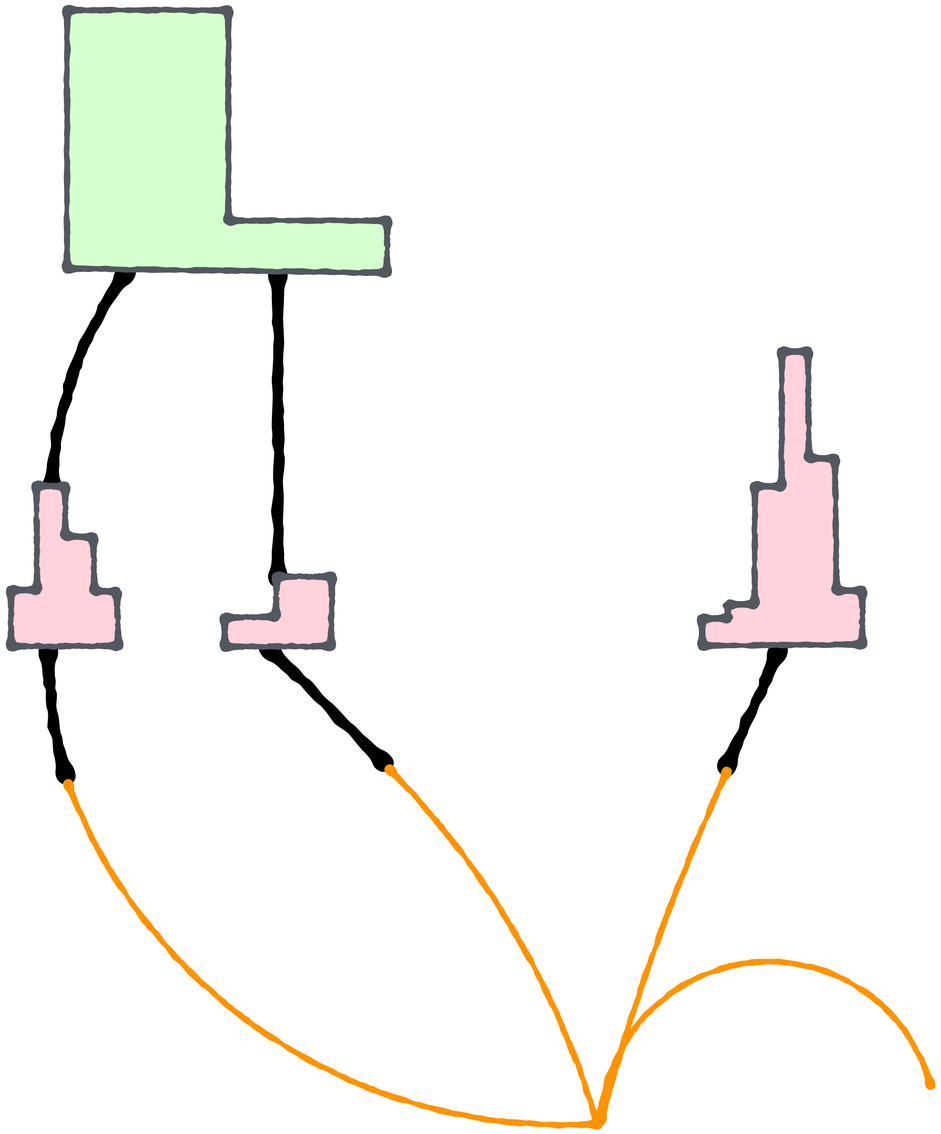


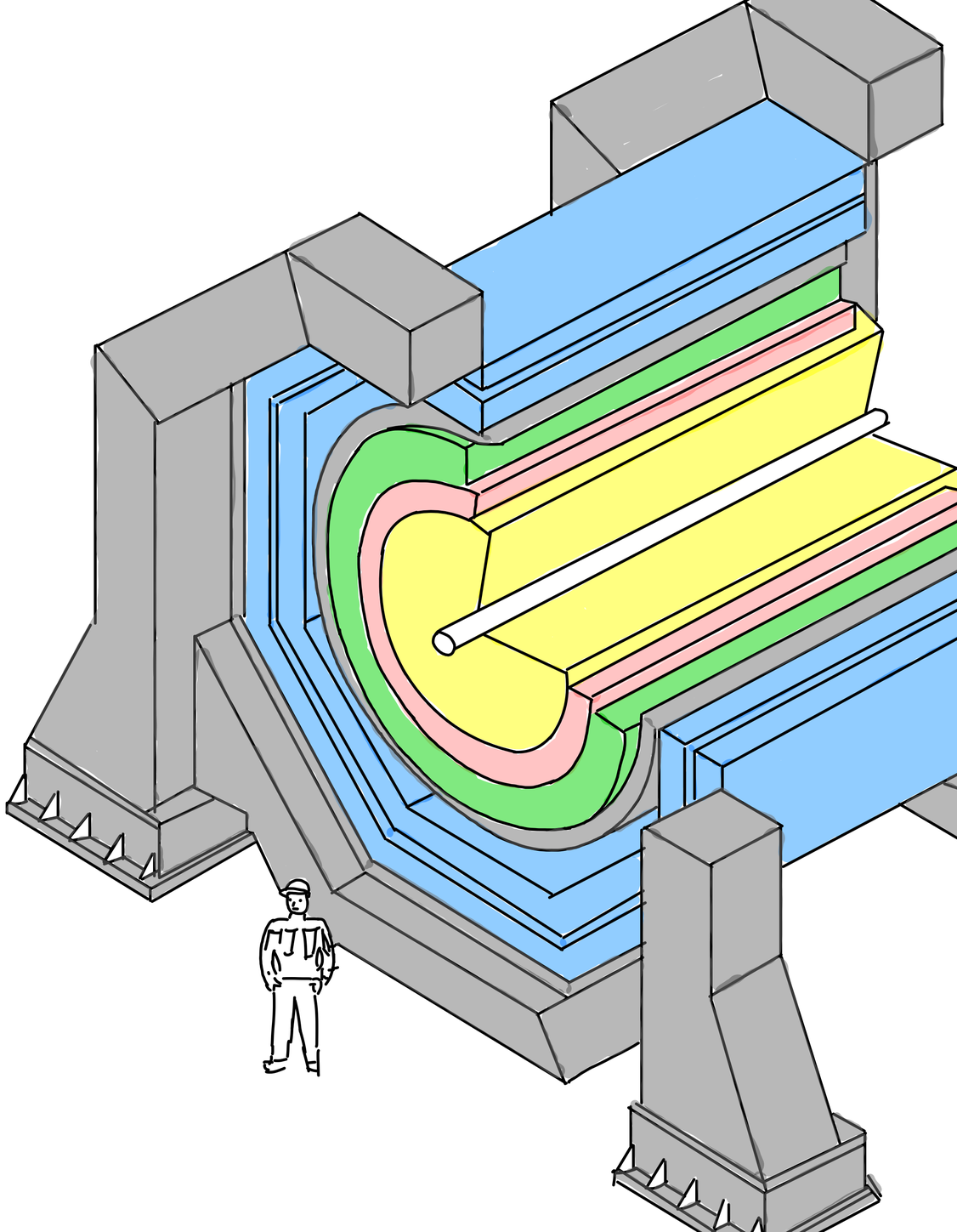
Next: Link across detectors

HCAL Clusters

ECAL Clusters

Tracks



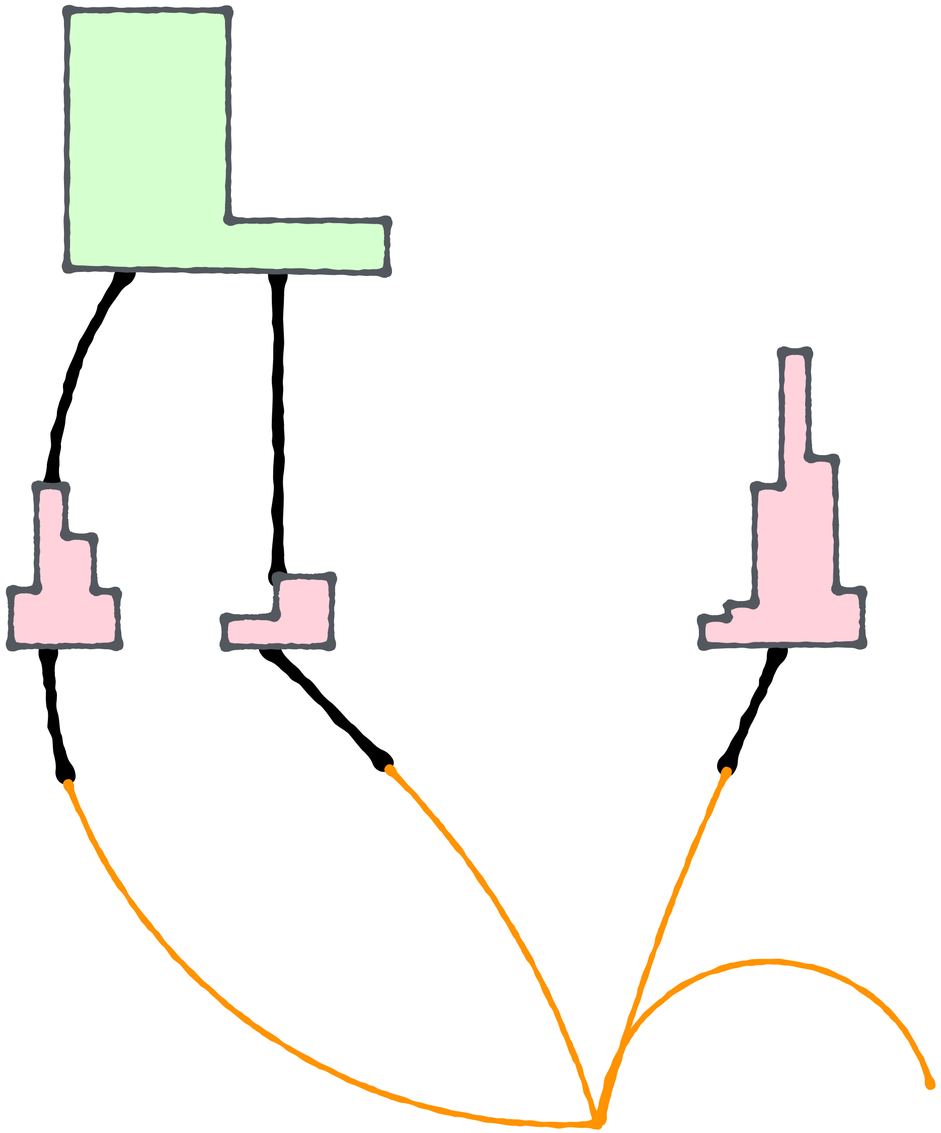


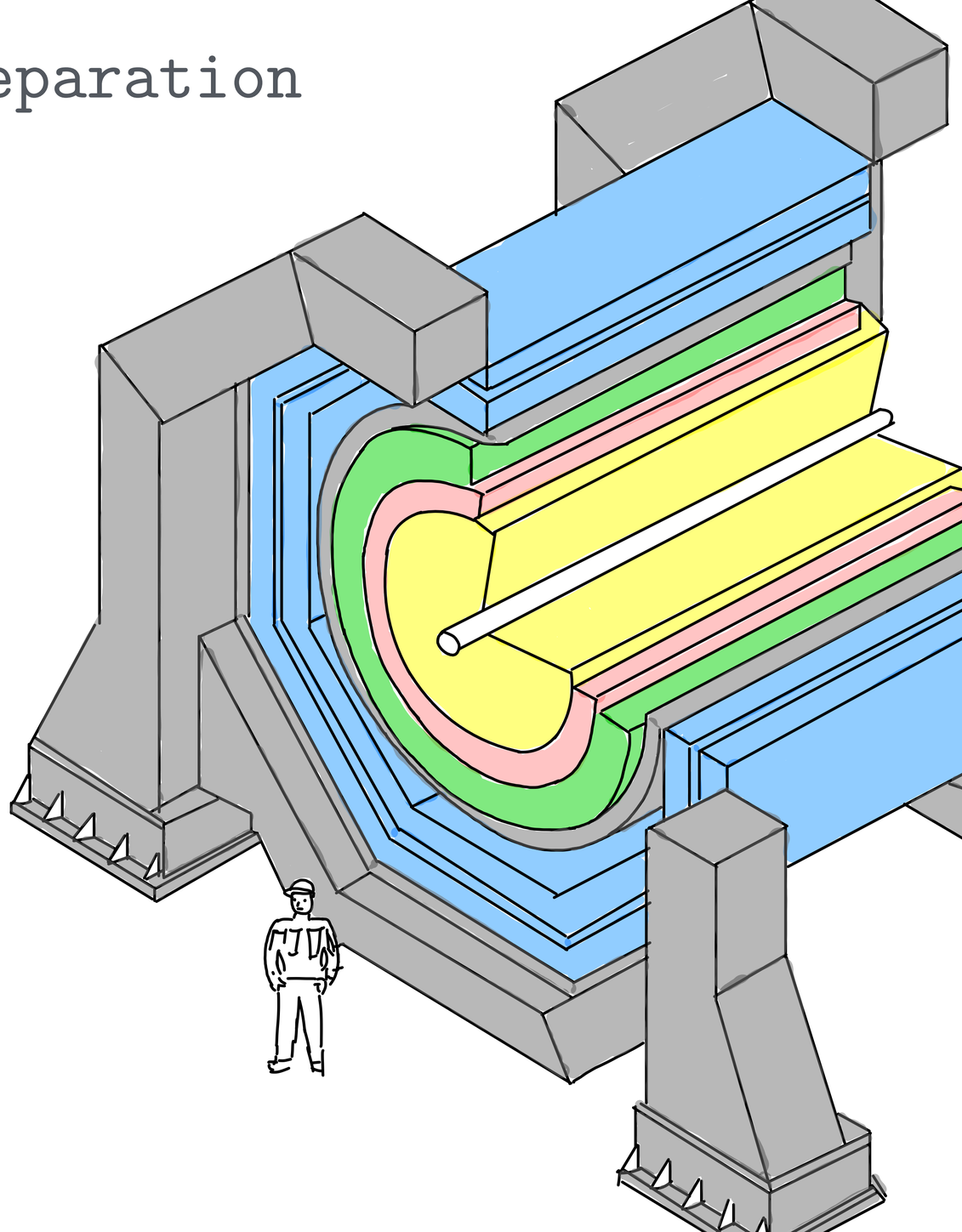
Finally: Apply Particle ID & Separation

HCAL Clusters

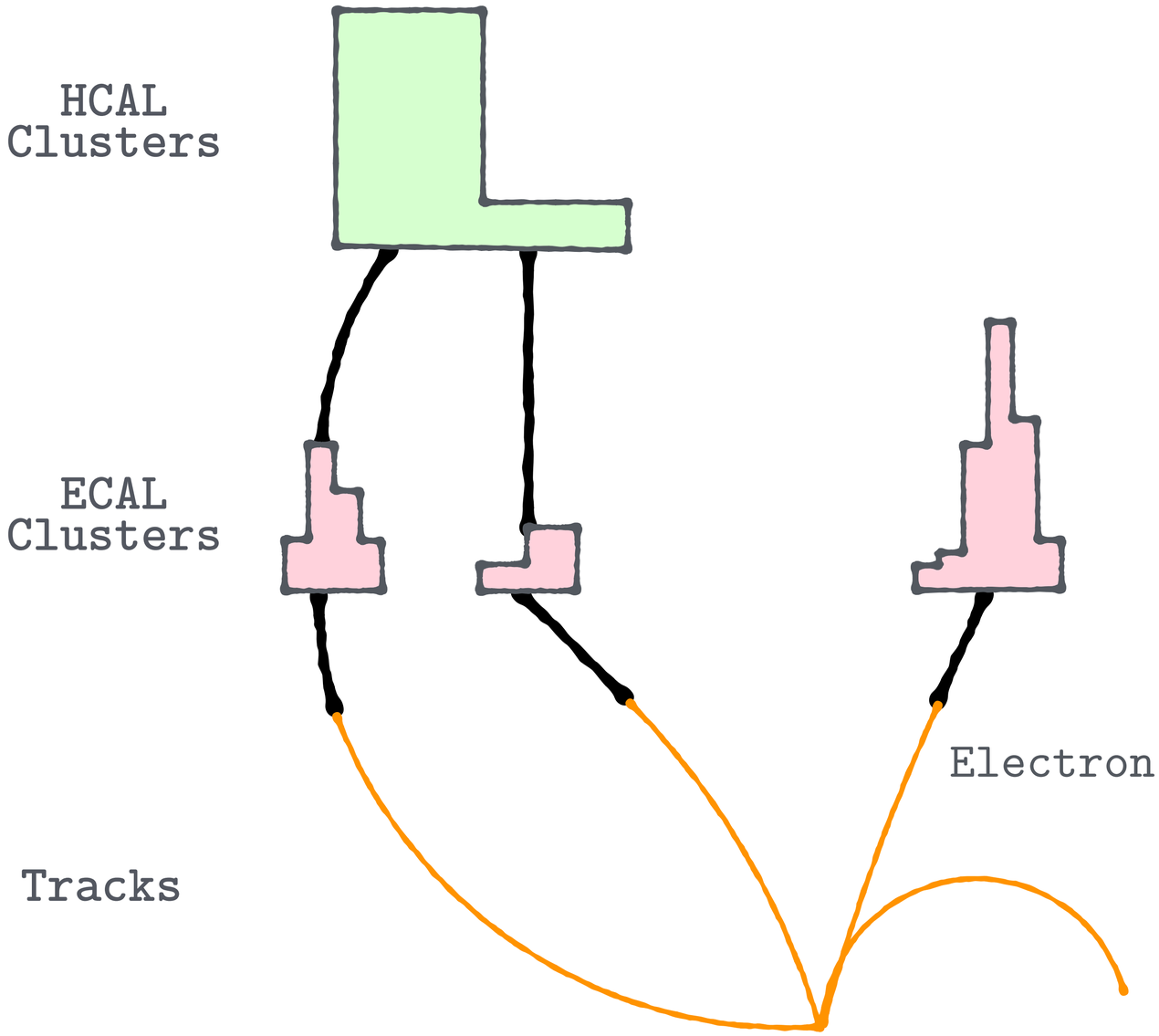
ECAL Clusters

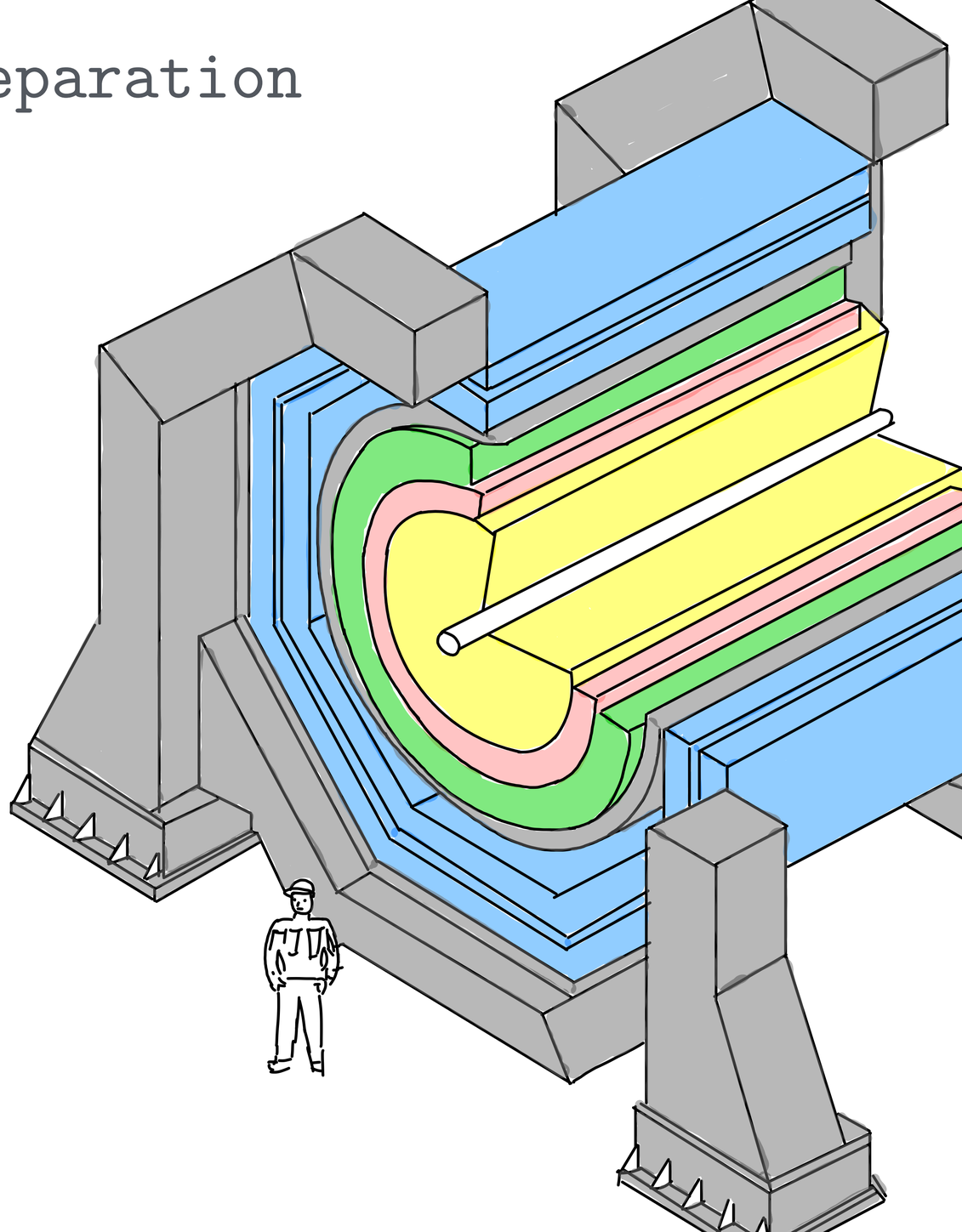
Tracks



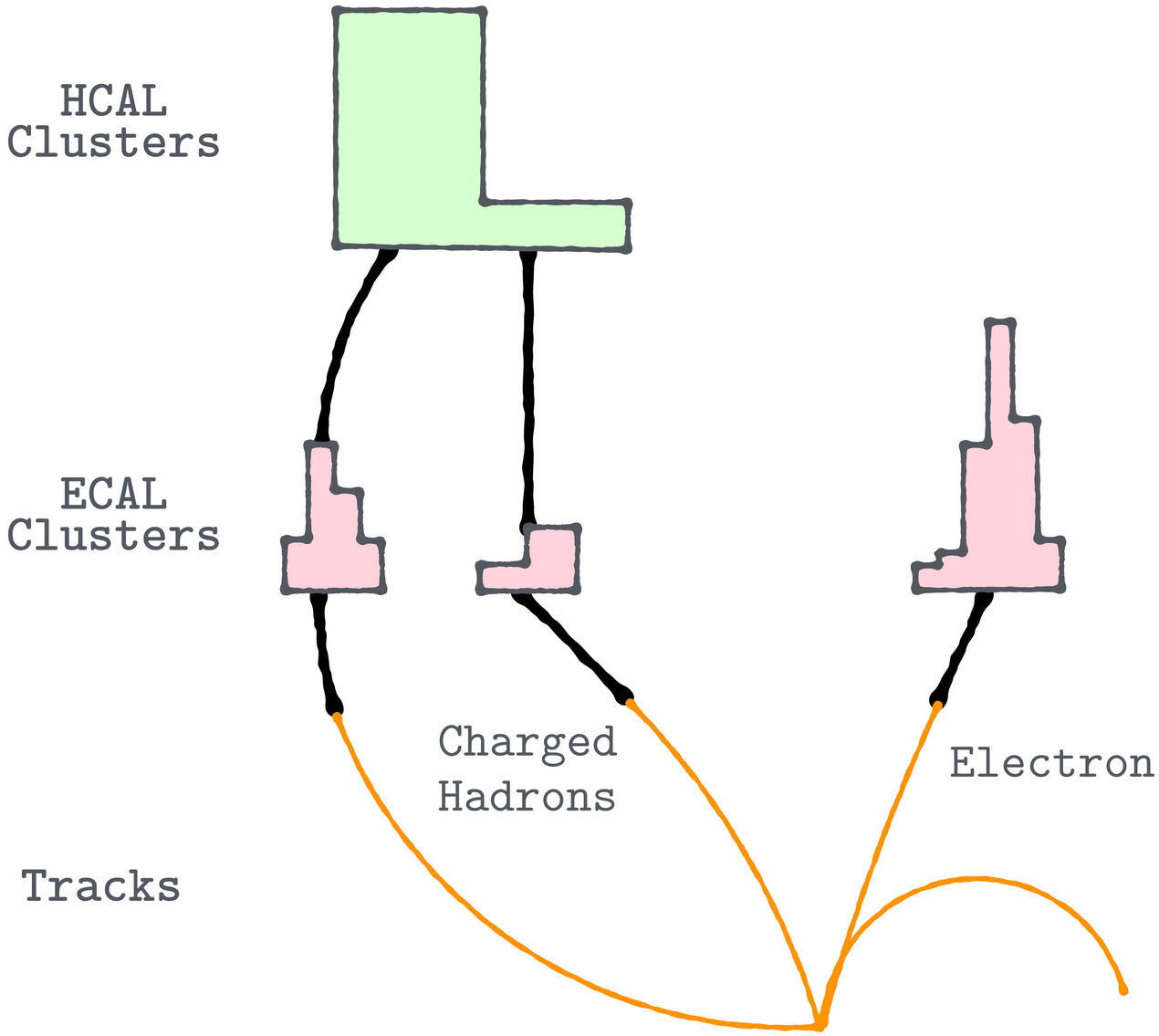


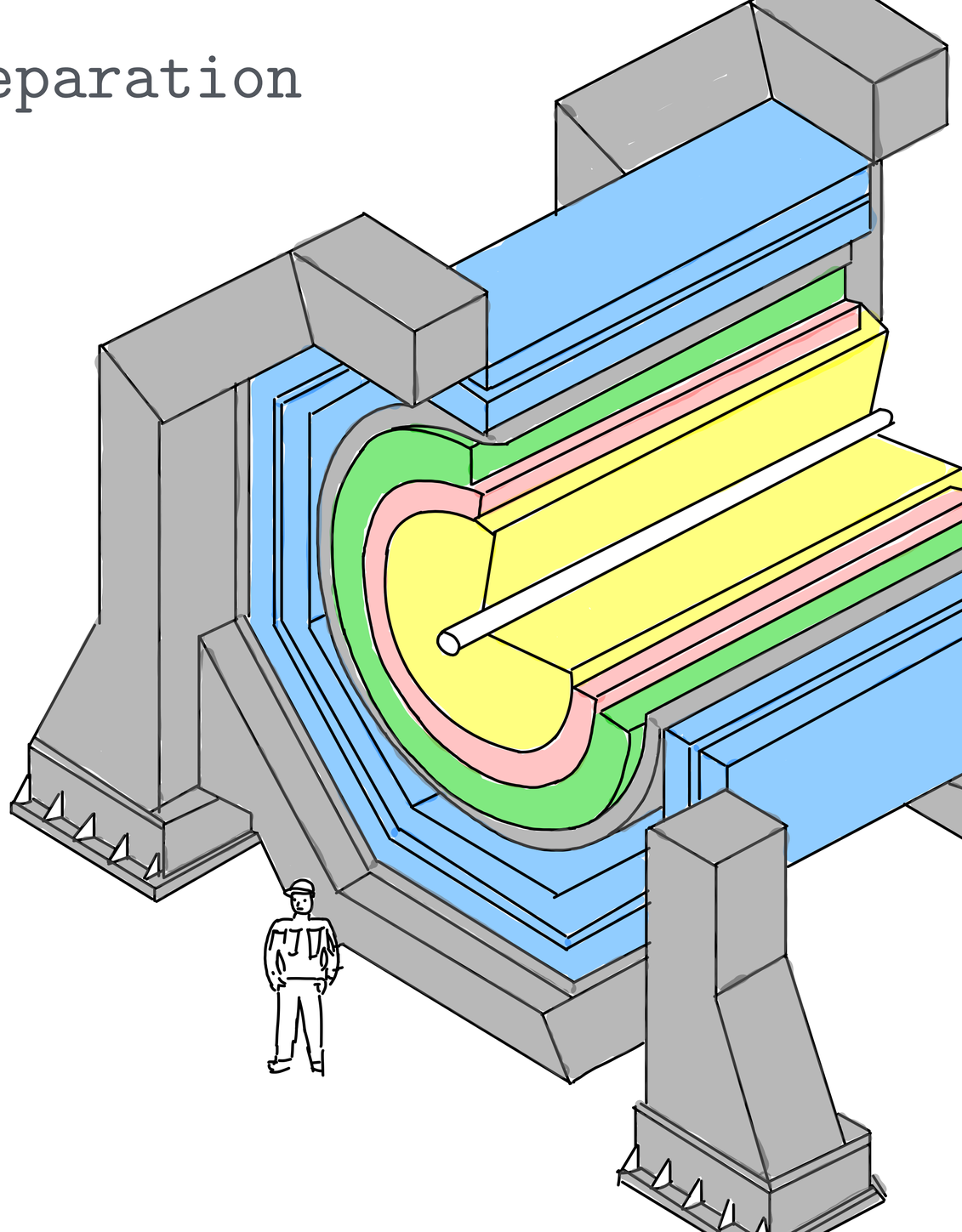
Finally: Apply Particle ID & Separation



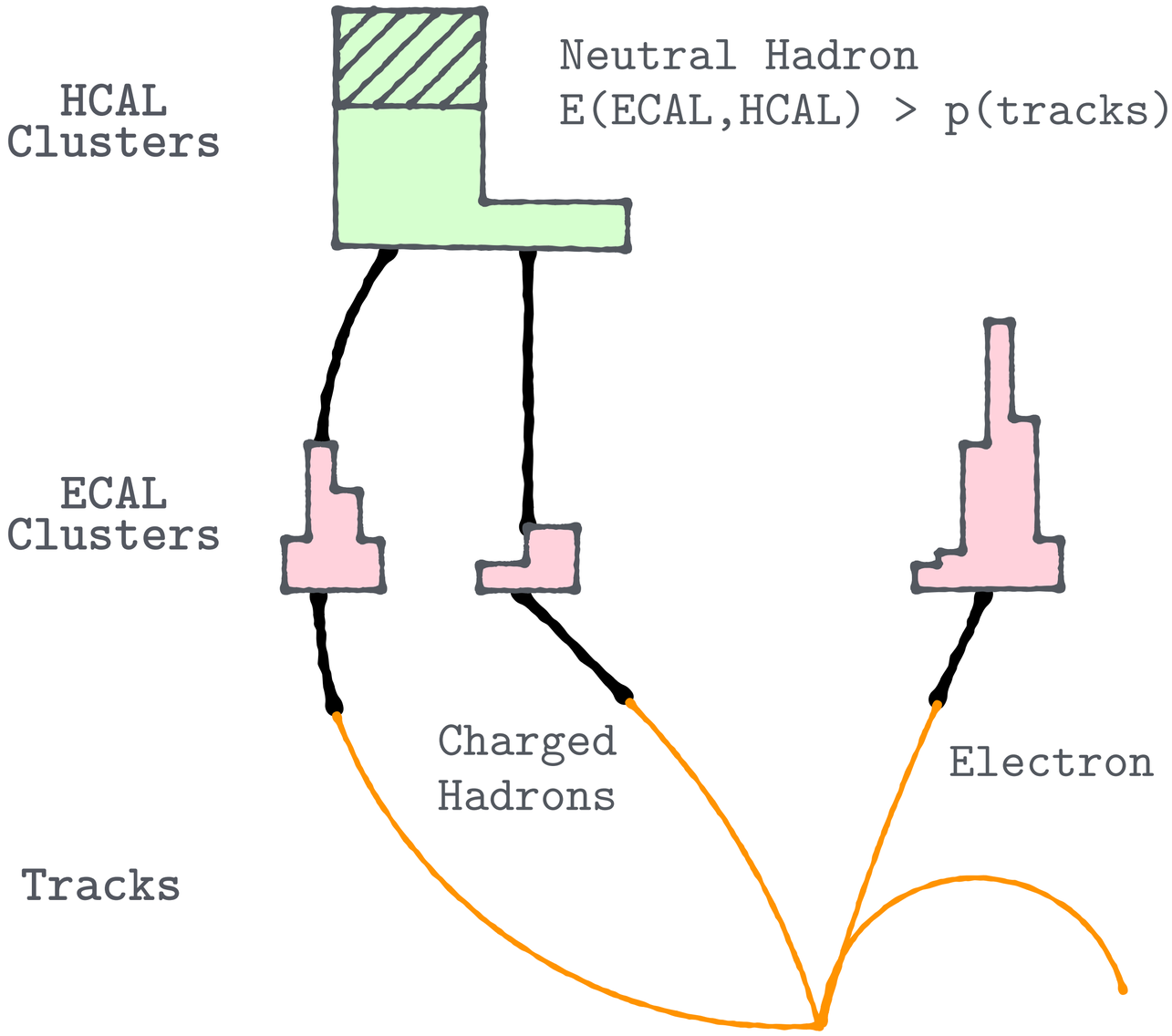


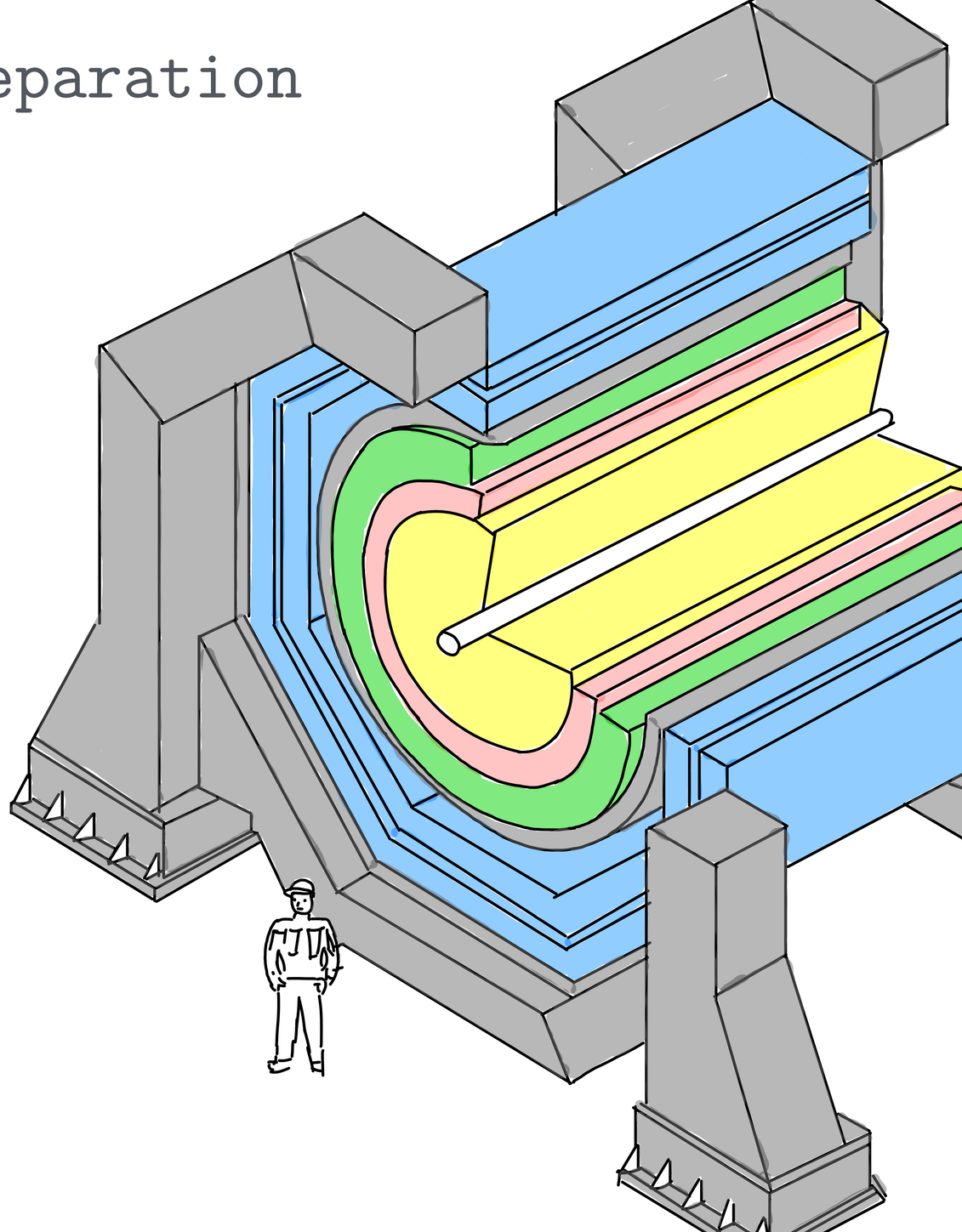
Finally: Apply Particle ID & Separation





Finally: Apply Particle ID & Separation





Clean the Event During Reconstruction

Clean the Event During Reconstruction

- Find and “remove” muons

Clean the Event During Reconstruction

- Find and “remove” muons

σ_{track}

Clean the Event During Reconstruction

- Find and “remove” muons
- Find and “remove” electrons

σ_{track}

Clean the Event During Reconstruction

- Find and “remove” muons σ_{track}
- Find and “remove” electrons $\min[\ \sigma_{\text{track}}, \sigma_{\text{ECAL}}\ ]$

Clean the Event During Reconstruction

- Find and “remove” muons
- Find and “remove” electrons
- Find and “remove” charged hadrons

σ_{track}

$\min[\ \sigma_{\text{track}}, \sigma_{\text{ECAL}}\ ]$

Clean the Event During Reconstruction

- Find and “remove” muons σ_{track}
- Find and “remove” electrons $\min[\ \sigma_{\text{track}}, \sigma_{\text{ECAL}}\ ]$
- Find and “remove” charged hadrons $\min[\ \sigma_{\text{track}}, \sigma_{\text{HCAL}}\ ]$

Clean the Event During Reconstruction

- Find and “remove” muons σ_{track}
- Find and “remove” electrons $\min[\ \sigma_{\text{track}}, \sigma_{\text{ECAL}}\ ]$
- Find and “remove” charged hadrons $\min[\ \sigma_{\text{track}}, \sigma_{\text{HCAL}}\ ]$
- Find and “remove” converted photons

Clean the Event During Reconstruction

- Find and “remove” muons σ_{track}
- Find and “remove” electrons $\min[\ \sigma_{\text{track}}, \sigma_{\text{ECAL}}\ ]$
- Find and “remove” charged hadrons $\min[\ \sigma_{\text{track}}, \sigma_{\text{HCAL}}\ ]$
- Find and “remove” converted photons σ_{track}

Clean the Event During Reconstruction

- Find and “remove” muons — σ_{track}
- Find and “remove” electrons — $\min[\ \sigma_{\text{track}}, \sigma_{\text{ECAL}}\ ]$
- Find and “remove” charged hadrons — $\min[\ \sigma_{\text{track}}, \sigma_{\text{HCAL}}\ ]$
- Find and “remove” converted photons — σ_{track}
- Find and “remove” V0’s

Clean the Event During Reconstruction

- Find and “remove” muons σ_{track}
- Find and “remove” electrons $\min[\ \sigma_{\text{track}}, \sigma_{\text{ECAL}}\ ]$
- Find and “remove” charged hadrons $\min[\ \sigma_{\text{track}}, \sigma_{\text{HCAL}}\ ]$
- Find and “remove” converted photons σ_{track}
- Find and “remove” V0’s σ_{track}

Clean the Event During Reconstruction

- Find and “remove” muons — σ_{track}
- Find and “remove” electrons — $\min[\ \sigma_{\text{track}}, \sigma_{\text{ECAL}}\ ]$
- Find and “remove” charged hadrons — $\min[\ \sigma_{\text{track}}, \sigma_{\text{HCAL}}\ ]$
- Find and “remove” converted photons — σ_{track}
- Find and “remove” V0’s — σ_{track}
- Find and “remove” photons

Clean the Event During Reconstruction

- Find and “remove” muons — σ_{track}
- Find and “remove” electrons — $\min[\ \sigma_{\text{track}}, \sigma_{\text{ECAL}}\ ]$
- Find and “remove” charged hadrons — $\min[\ \sigma_{\text{track}}, \sigma_{\text{HCAL}}\ ]$
- Find and “remove” converted photons — σ_{track}
- Find and “remove” V0’s — σ_{track}
- Find and “remove” photons — σ_{ECAL}

Clean the Event During Reconstruction

- Find and “remove” muons — σ_{track}
- Find and “remove” electrons — $\min[\ \sigma_{\text{track}}, \sigma_{\text{ECAL}}\ ]$
- Find and “remove” charged hadrons — $\min[\ \sigma_{\text{track}}, \sigma_{\text{HCAL}}\ ]$
- Find and “remove” converted photons — σ_{track}
- Find and “remove” V0’s — σ_{track}
- Find and “remove” photons — σ_{ECAL}
- Left with neutral hadrons (10%)

Clean the Event During Reconstruction

- Find and “remove” muons — σ_{track}
- Find and “remove” electrons — $\min[\ \sigma_{\text{track}}, \sigma_{\text{ECAL}}\ ]$
- Find and “remove” charged hadrons — $\min[\ \sigma_{\text{track}}, \sigma_{\text{HCAL}}\ ]$
- Find and “remove” converted photons — σ_{track}
- Find and “remove” V0’s — σ_{track}
- Find and “remove” photons — σ_{ECAL}
- Left with neutral hadrons (10%) — σ_{HCAL} + fakes

Clean the Event During Reconstruction

- Find and “remove” muons — σ_{track}
- Find and “remove” electrons — $\min[\ \sigma_{\text{track}}, \sigma_{\text{ECAL}}\ ]$
- Find and “remove” charged hadrons — $\min[\ \sigma_{\text{track}}, \sigma_{\text{HCAL}}\ ]$
- Find and “remove” converted photons — σ_{track}
- Find and “remove” V0’s — σ_{track}
- Find and “remove” photons — σ_{ECAL}
- Left with neutral hadrons (10%) — σ_{HCAL} **+ fakes**

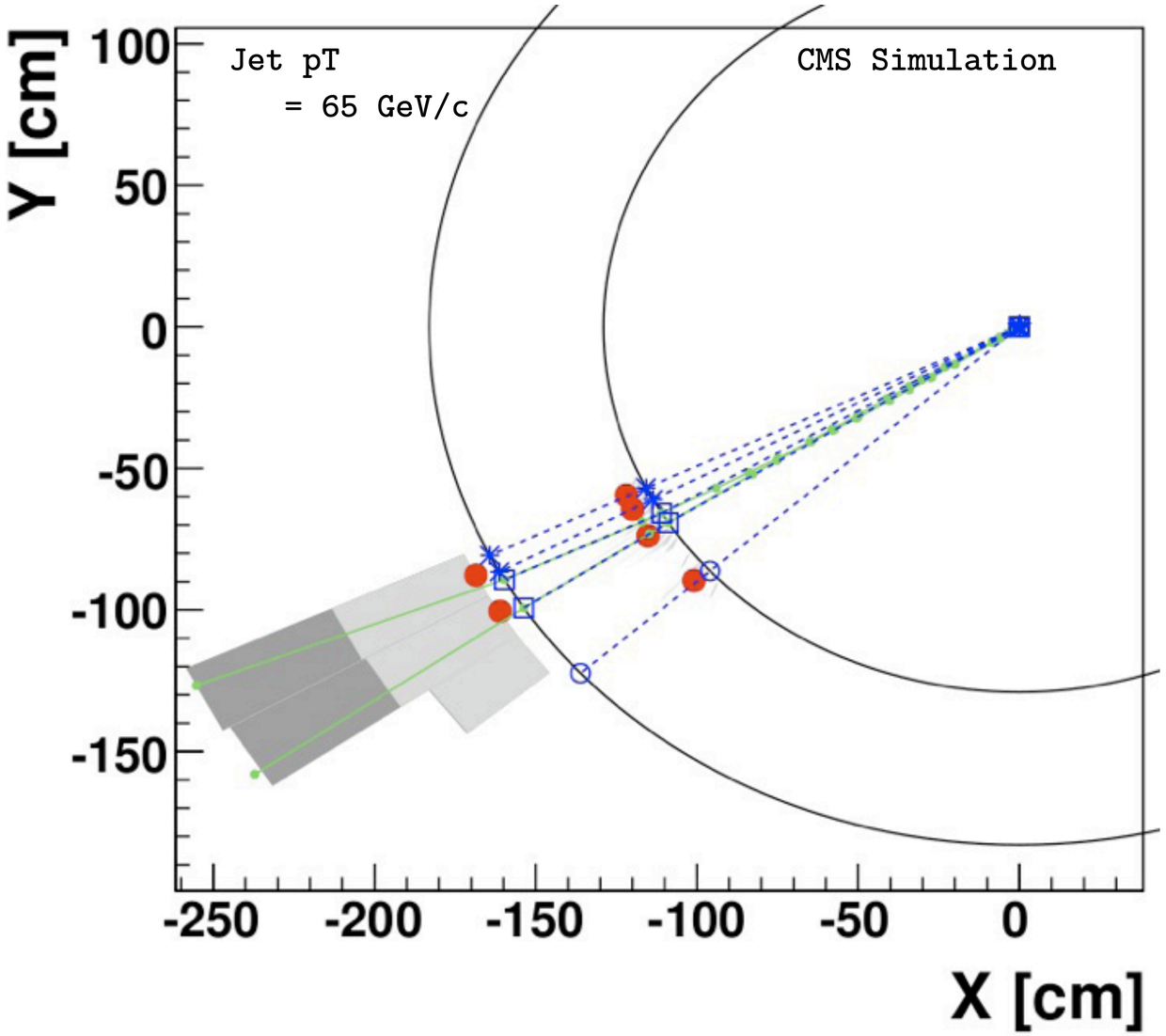
- Use above list of Particles to describe entire event!

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

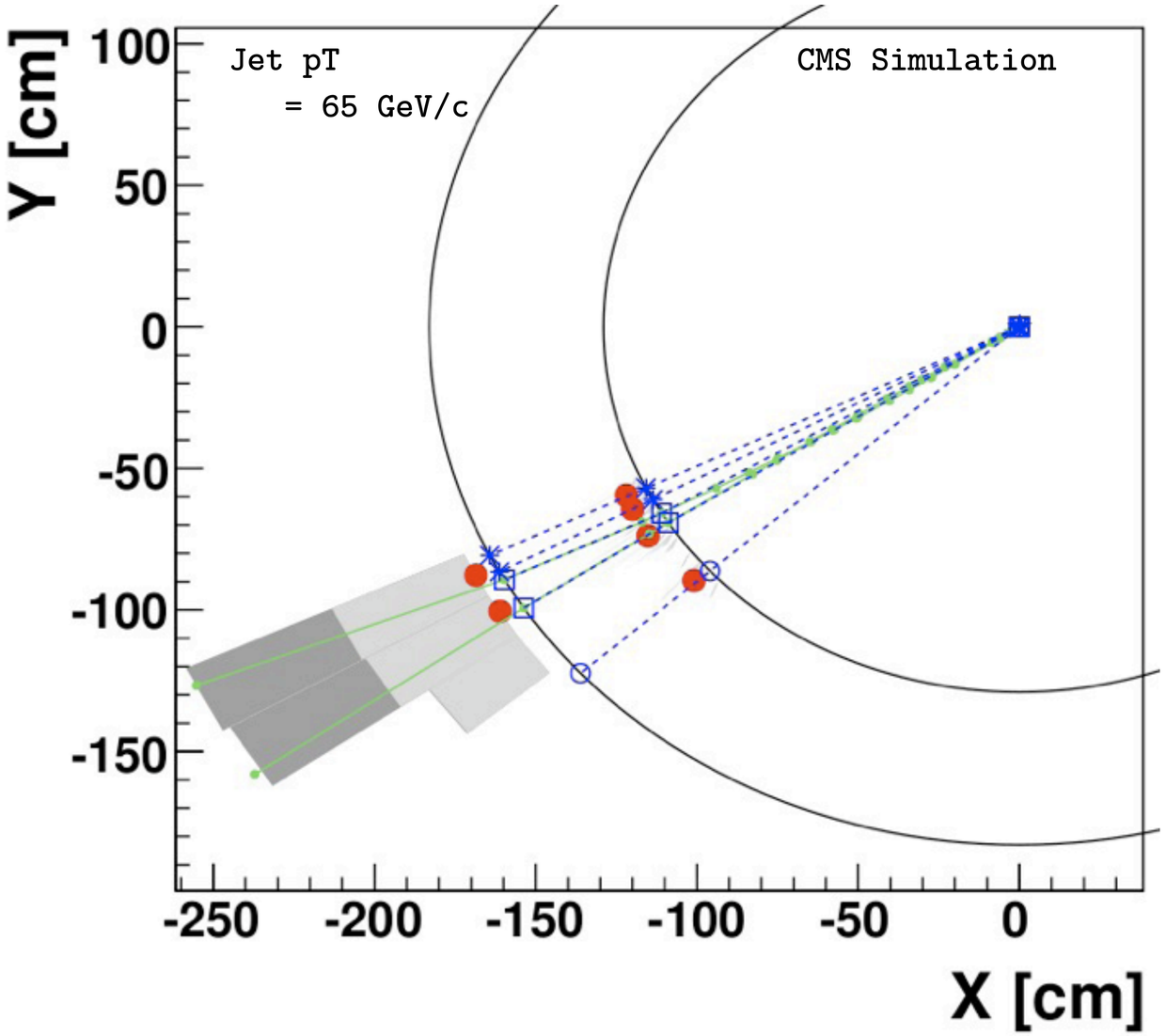
Four true particles:

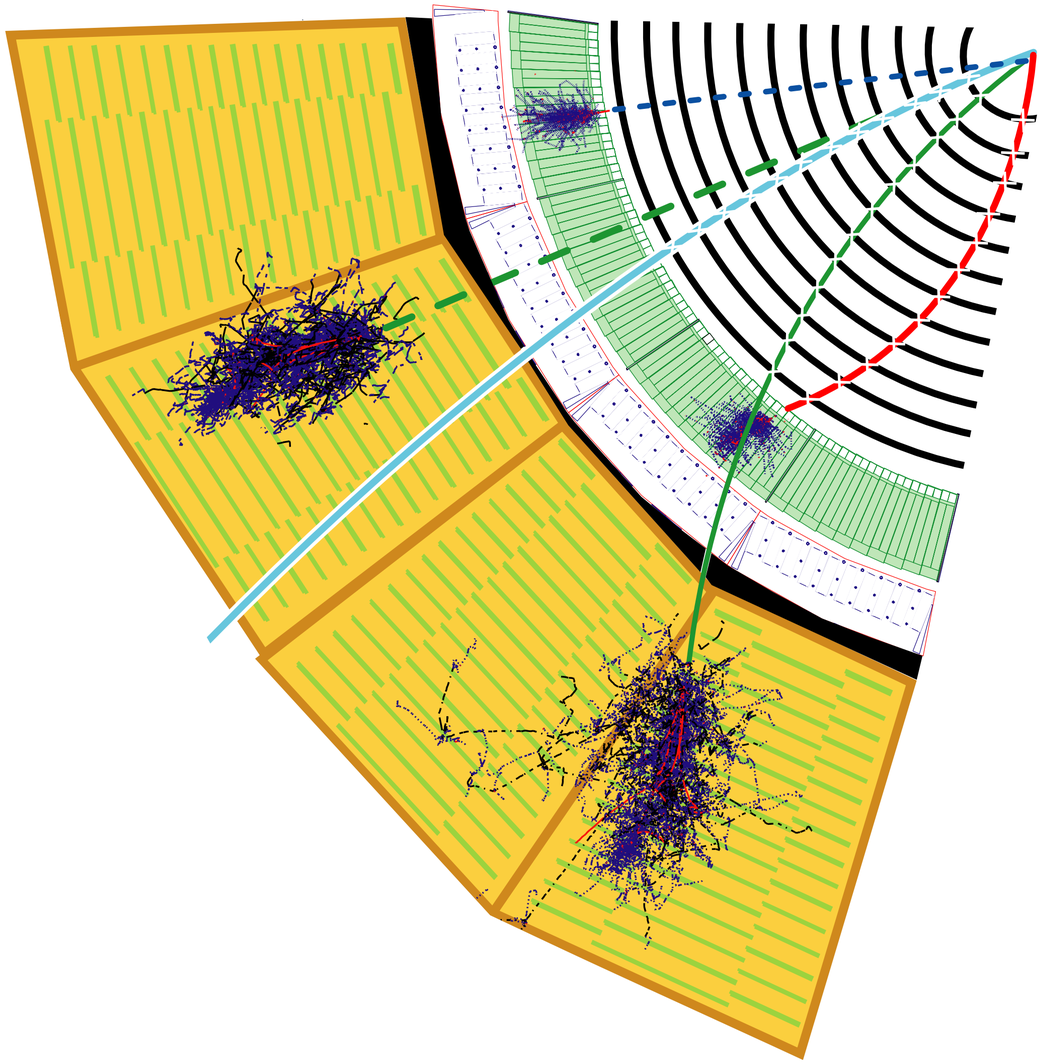
$\pi^+, \pi^-, \pi^0, K_L^0$



Four true particles:

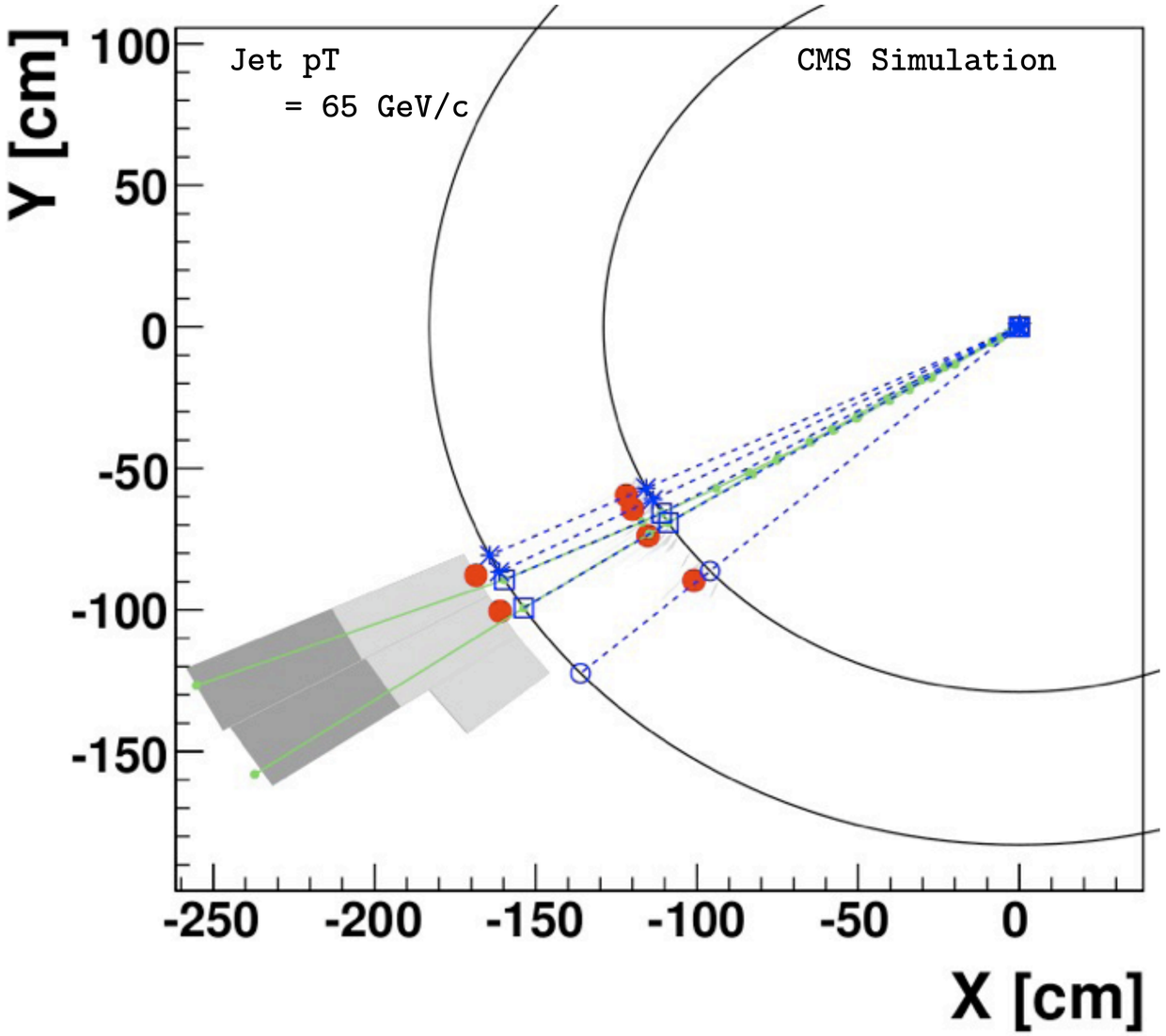
$\pi^+, \pi^-, \pi^0, K_L^0$





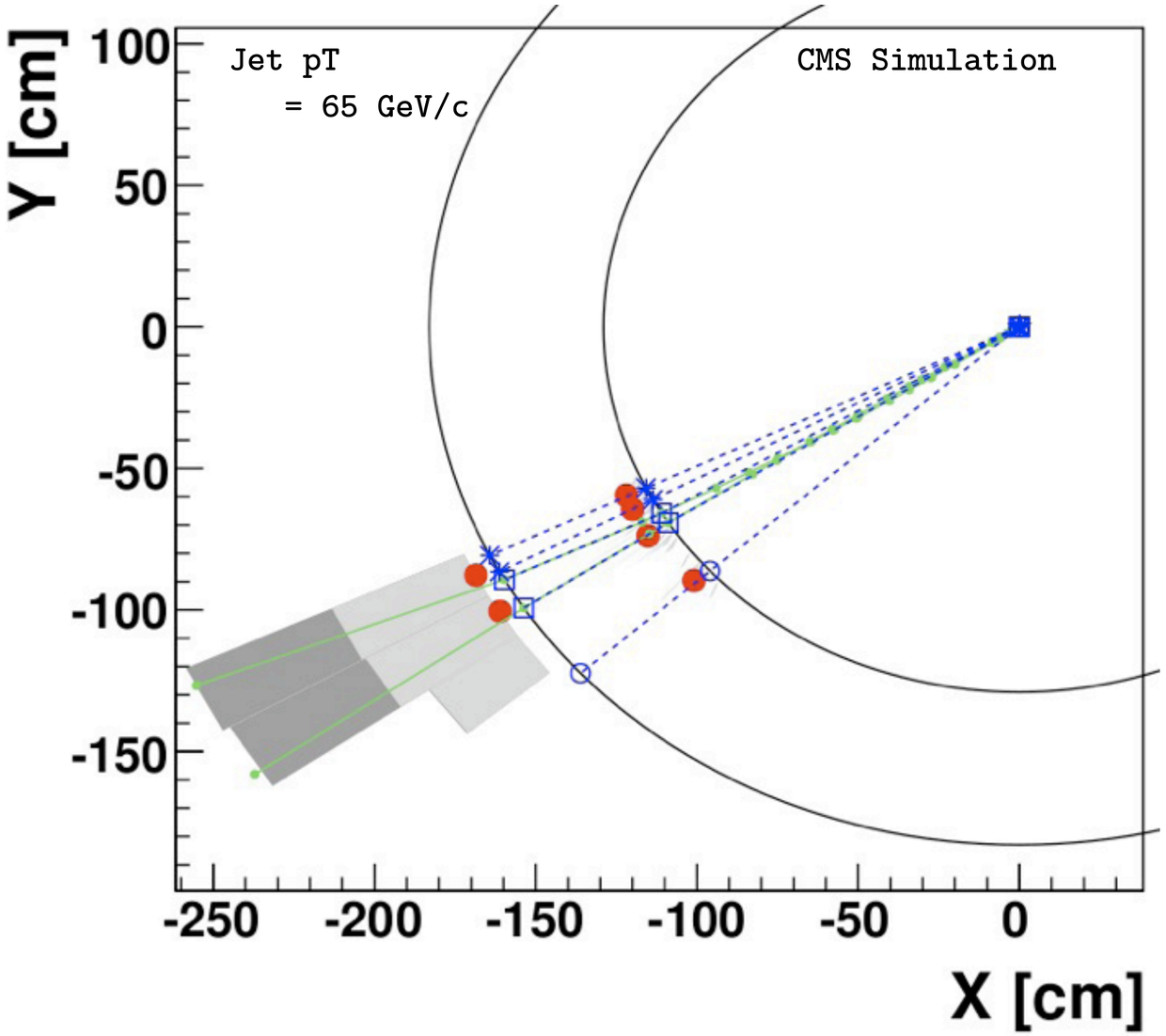
Four true particles:

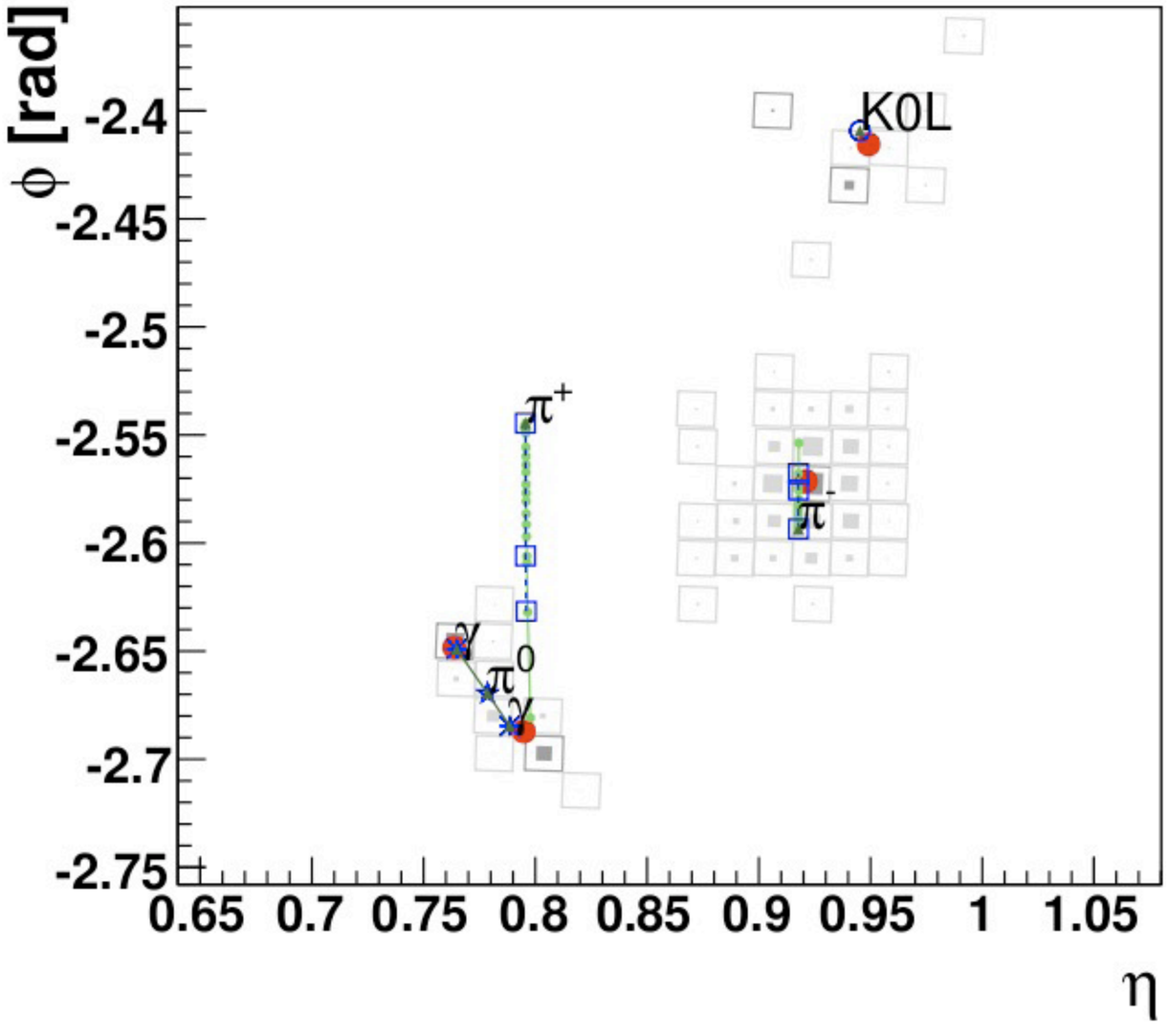
$\pi^+, \pi^-, \pi^0, K_L^0$



Four true particles:

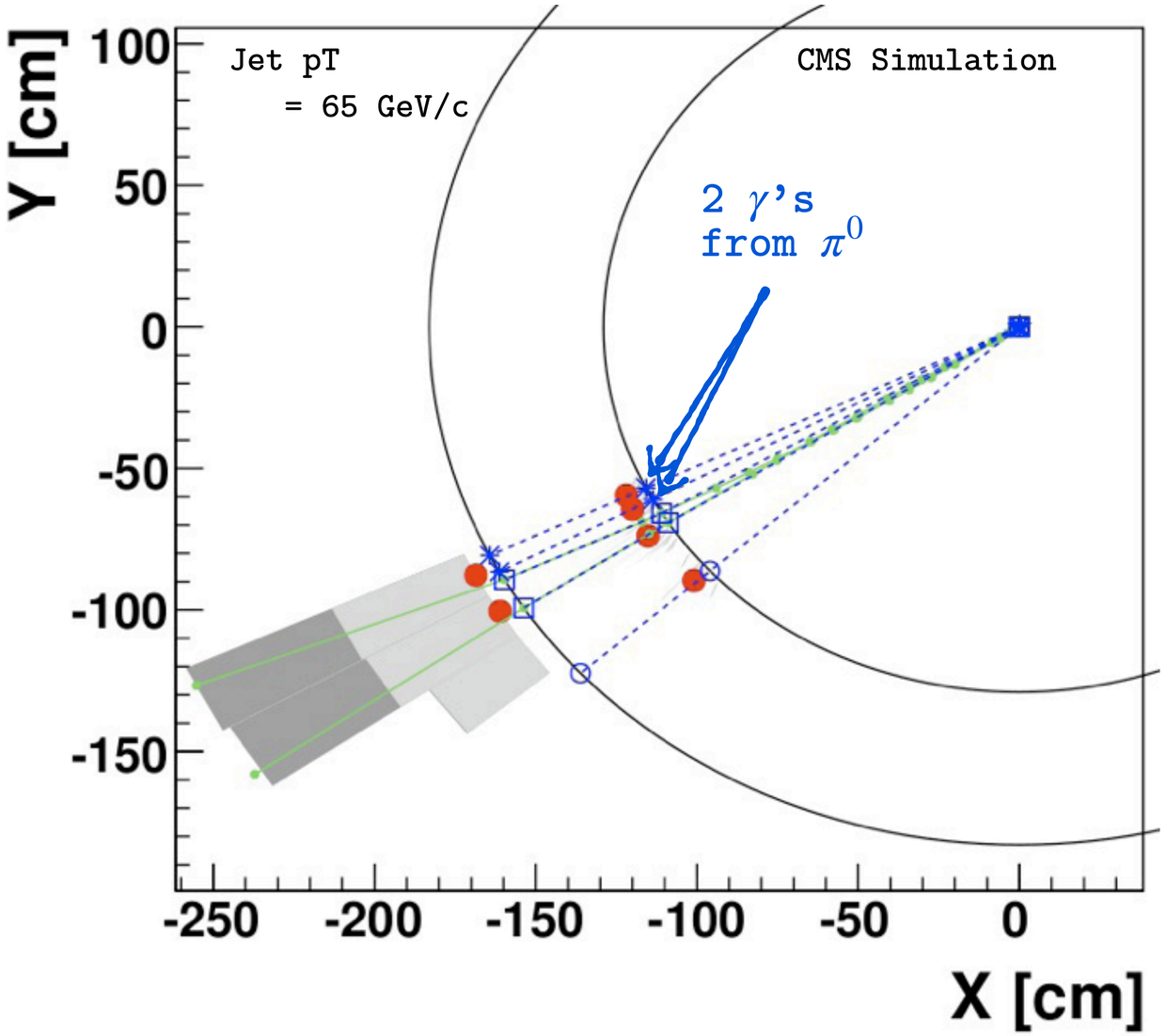
$\pi^+, \pi^-, \pi^0, K_L^0$

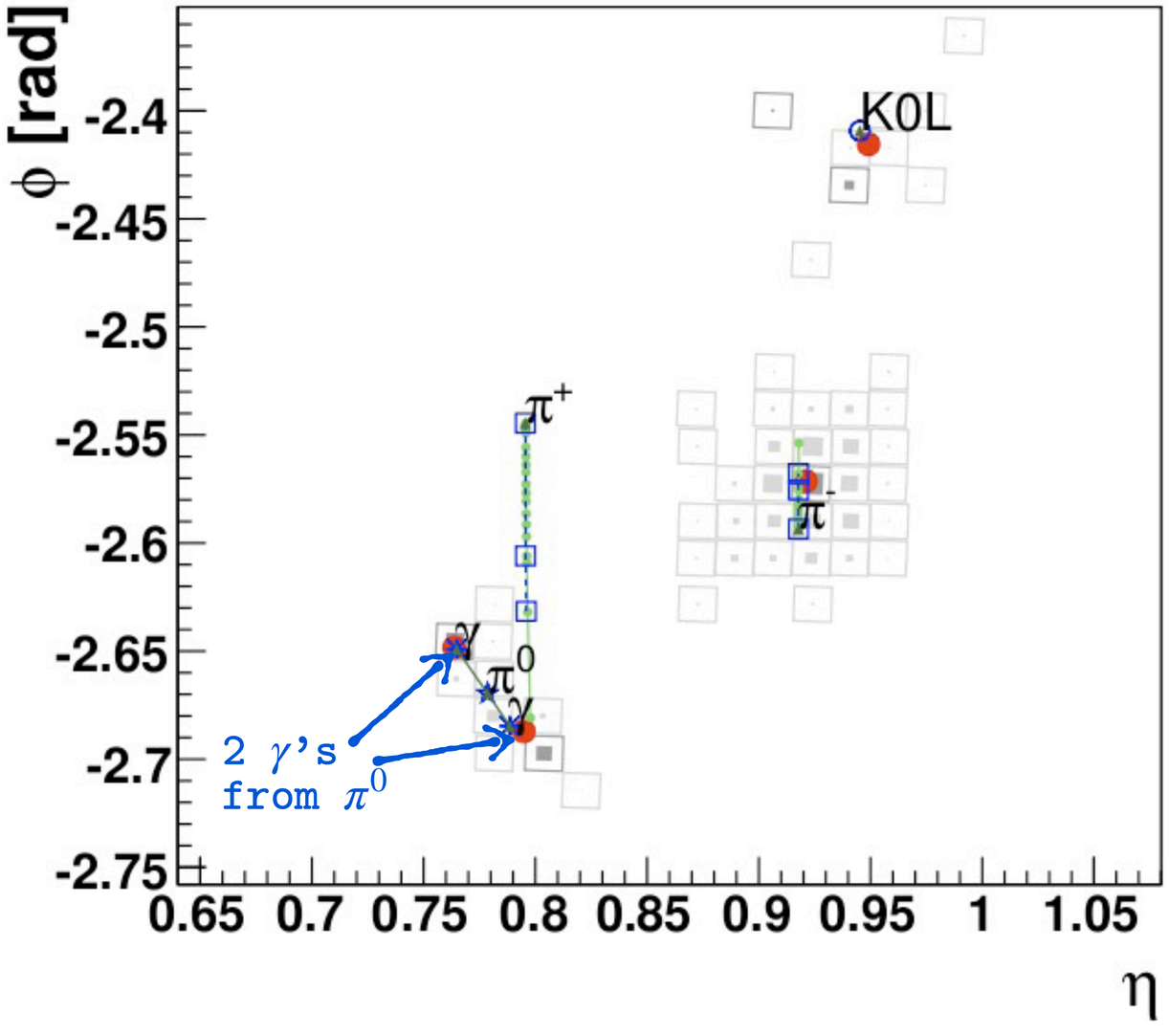




Four true particles:

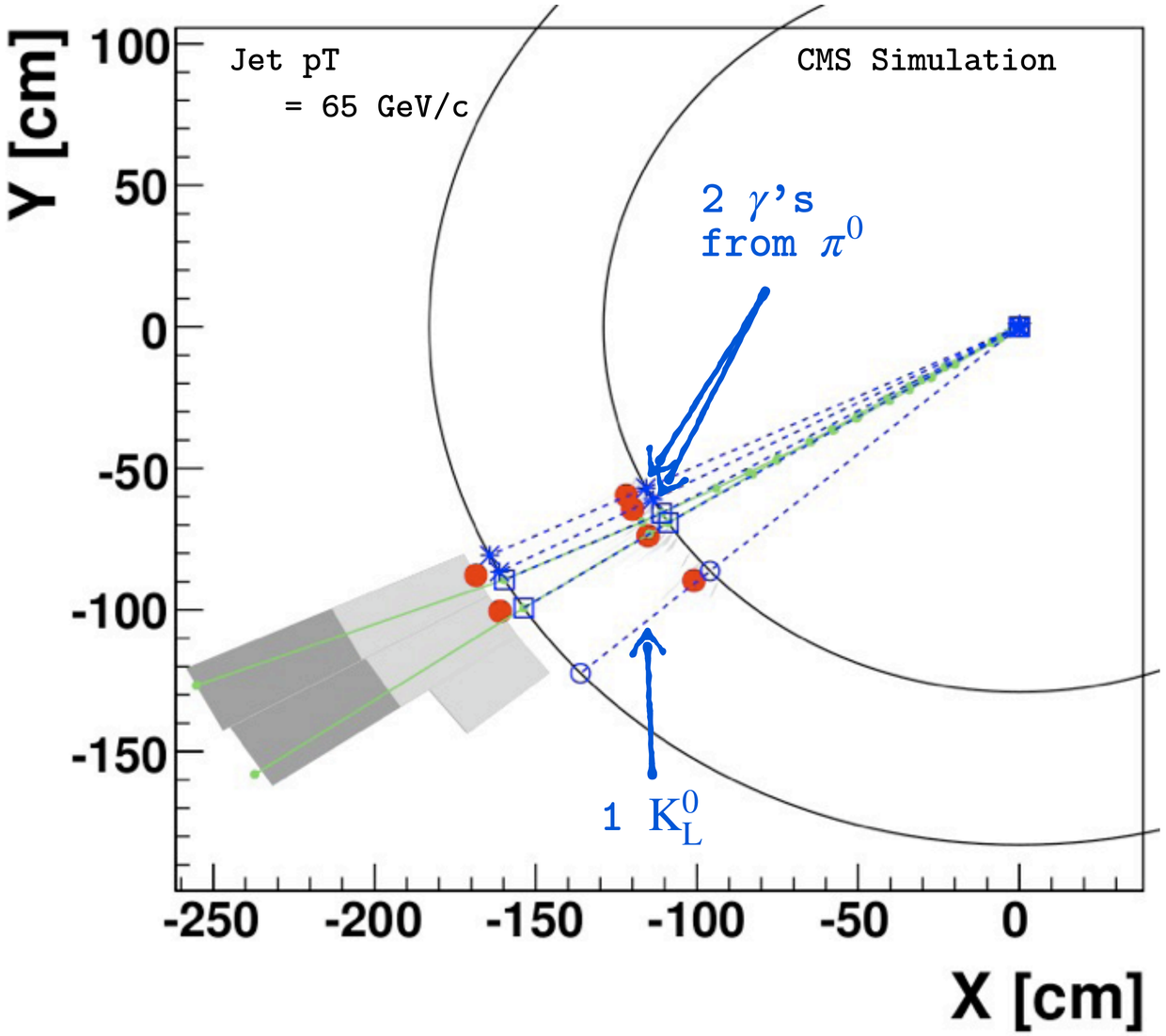
$\pi^+, \pi^-, \pi^0, K_L^0$

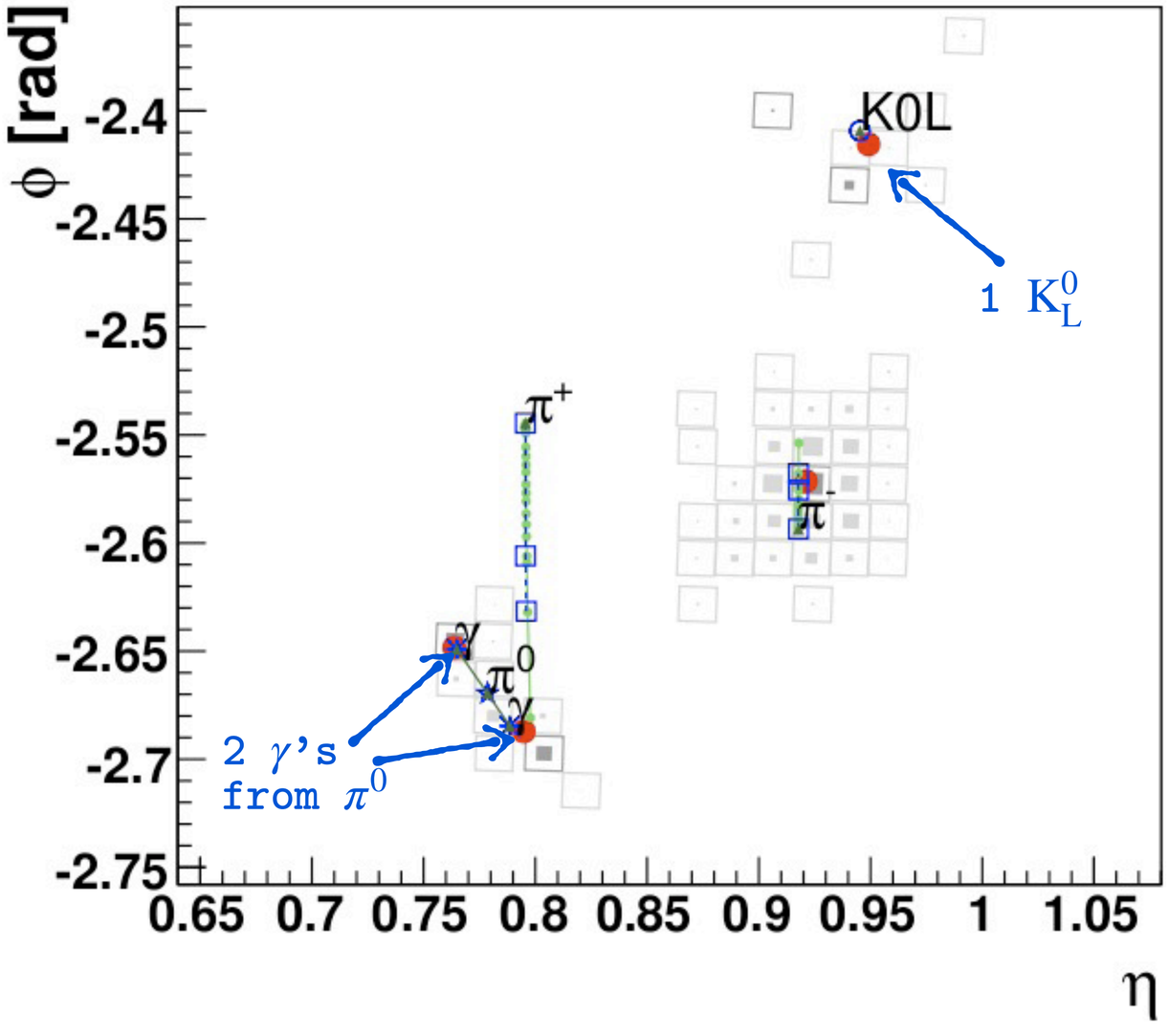




Four true particles:

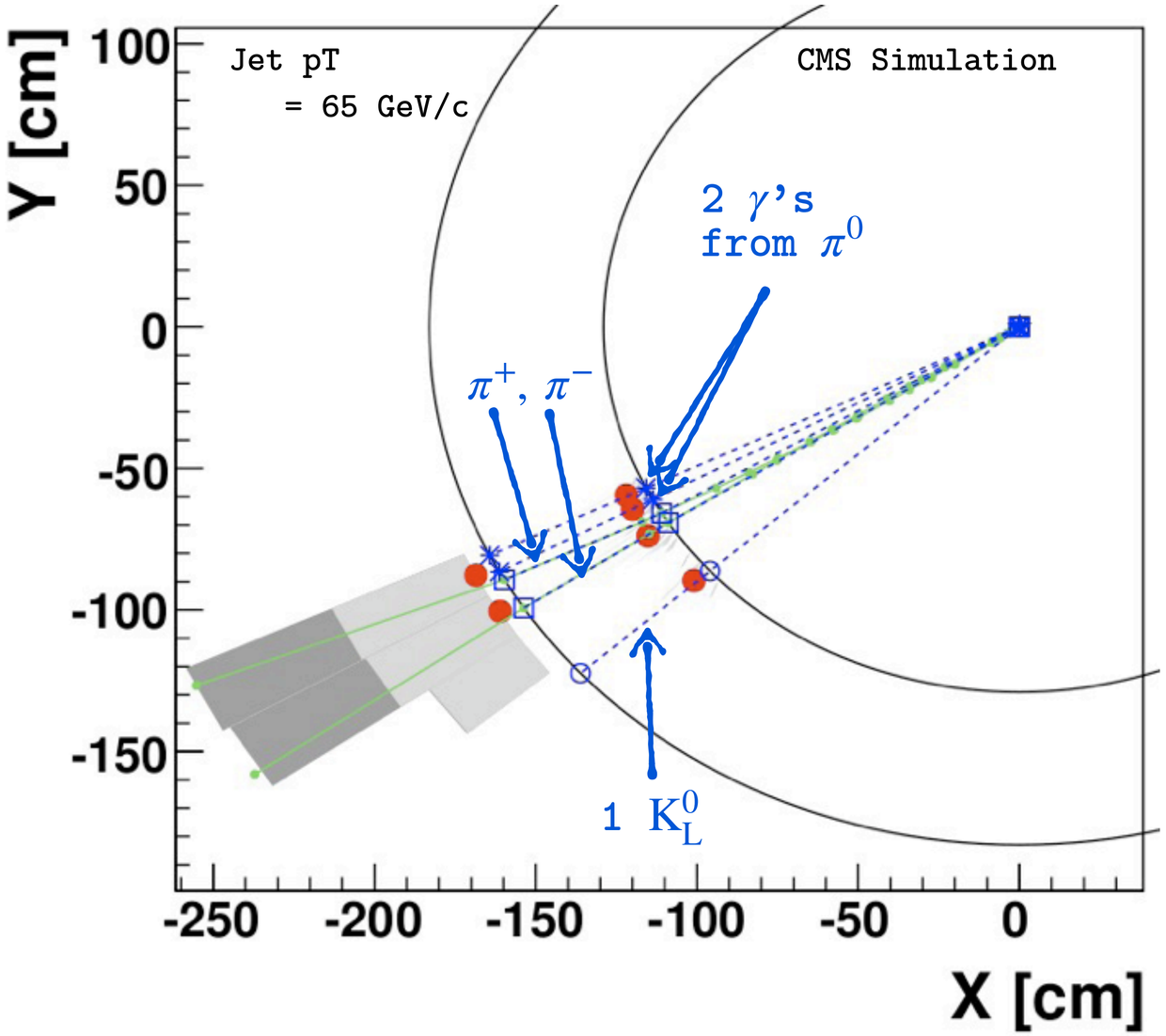
$\pi^+, \pi^-, \pi^0, K_L^0$

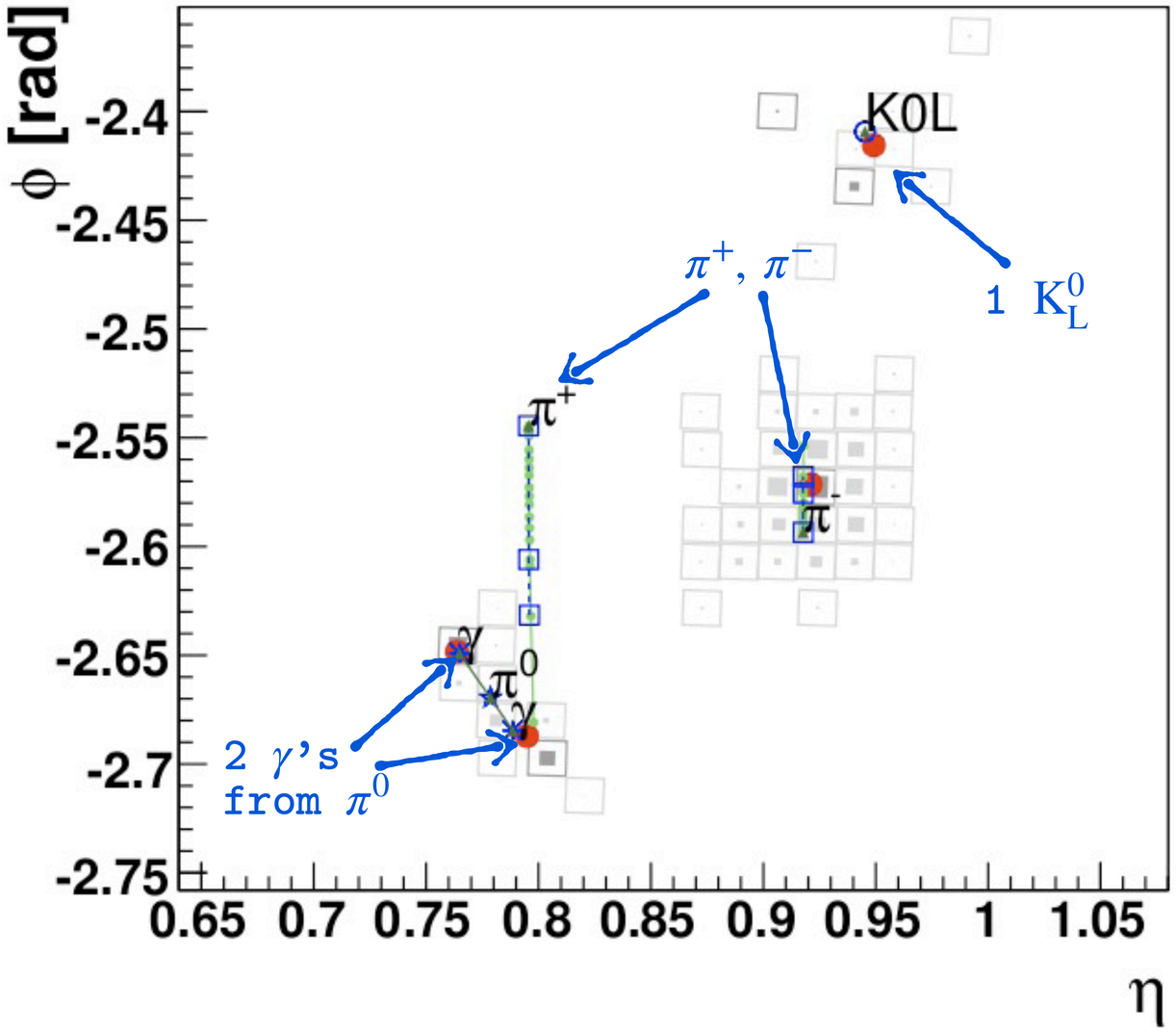




Four true particles:

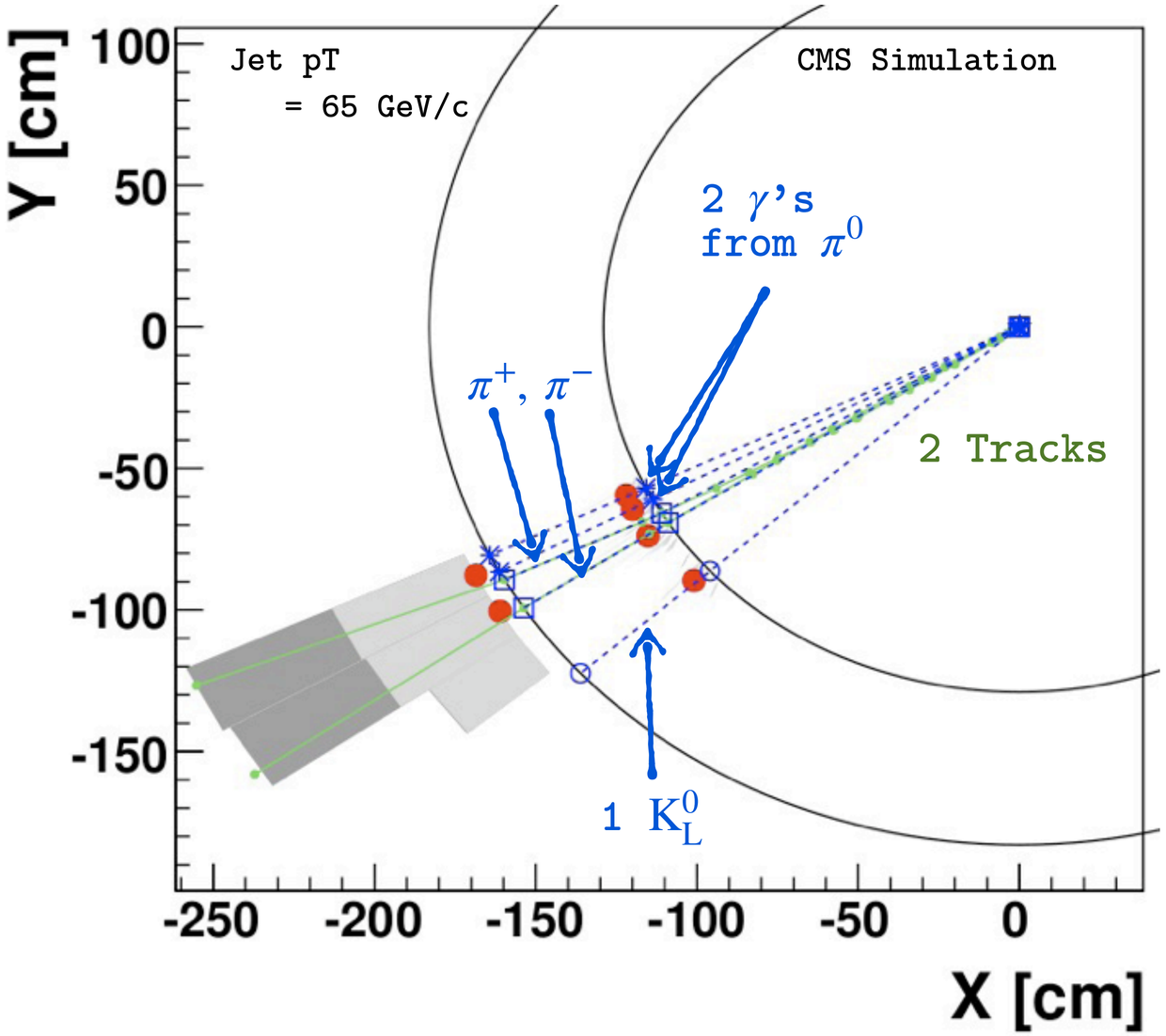
$\pi^+, \pi^-, \pi^0, K_L^0$

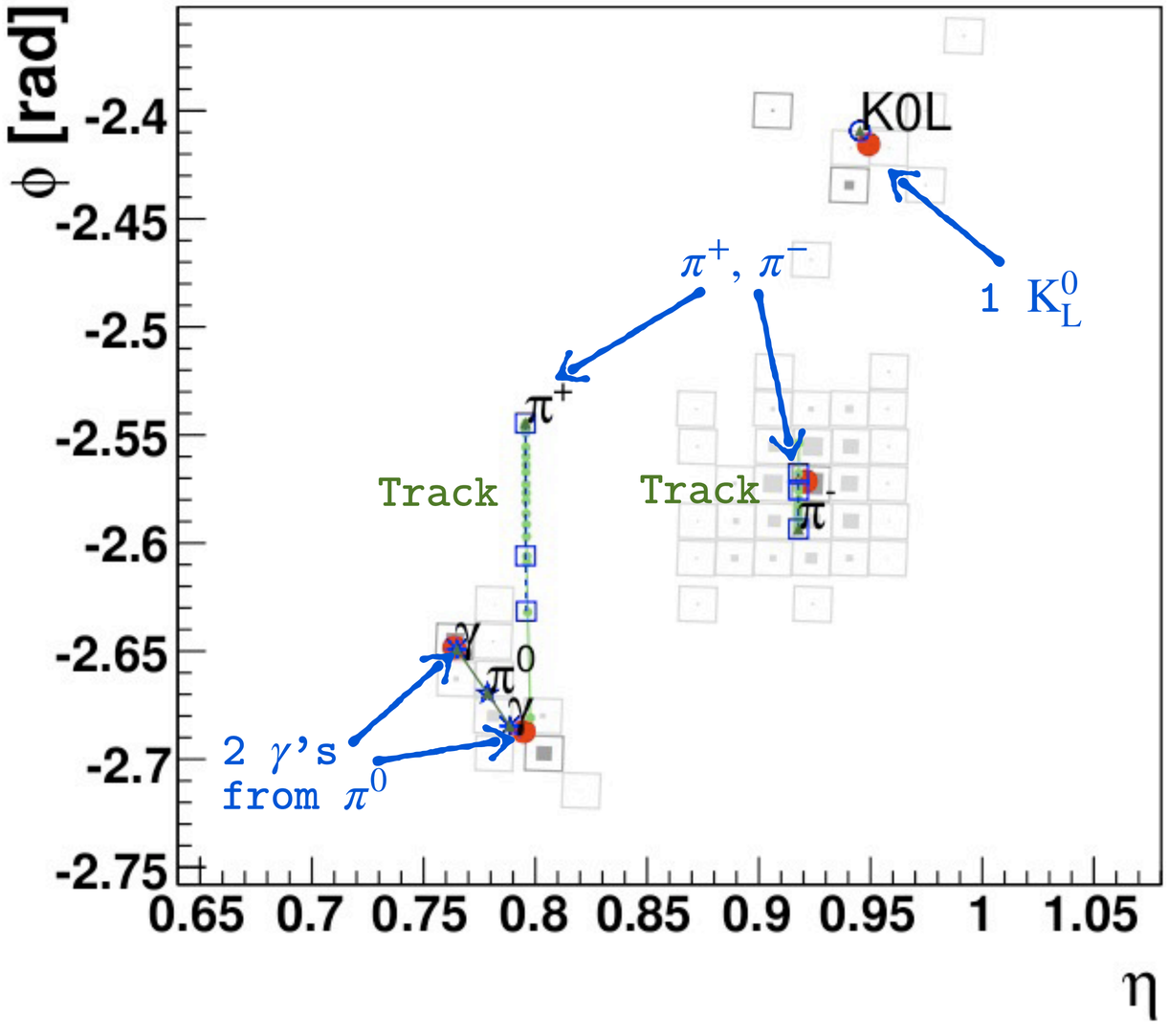




Four true particles:

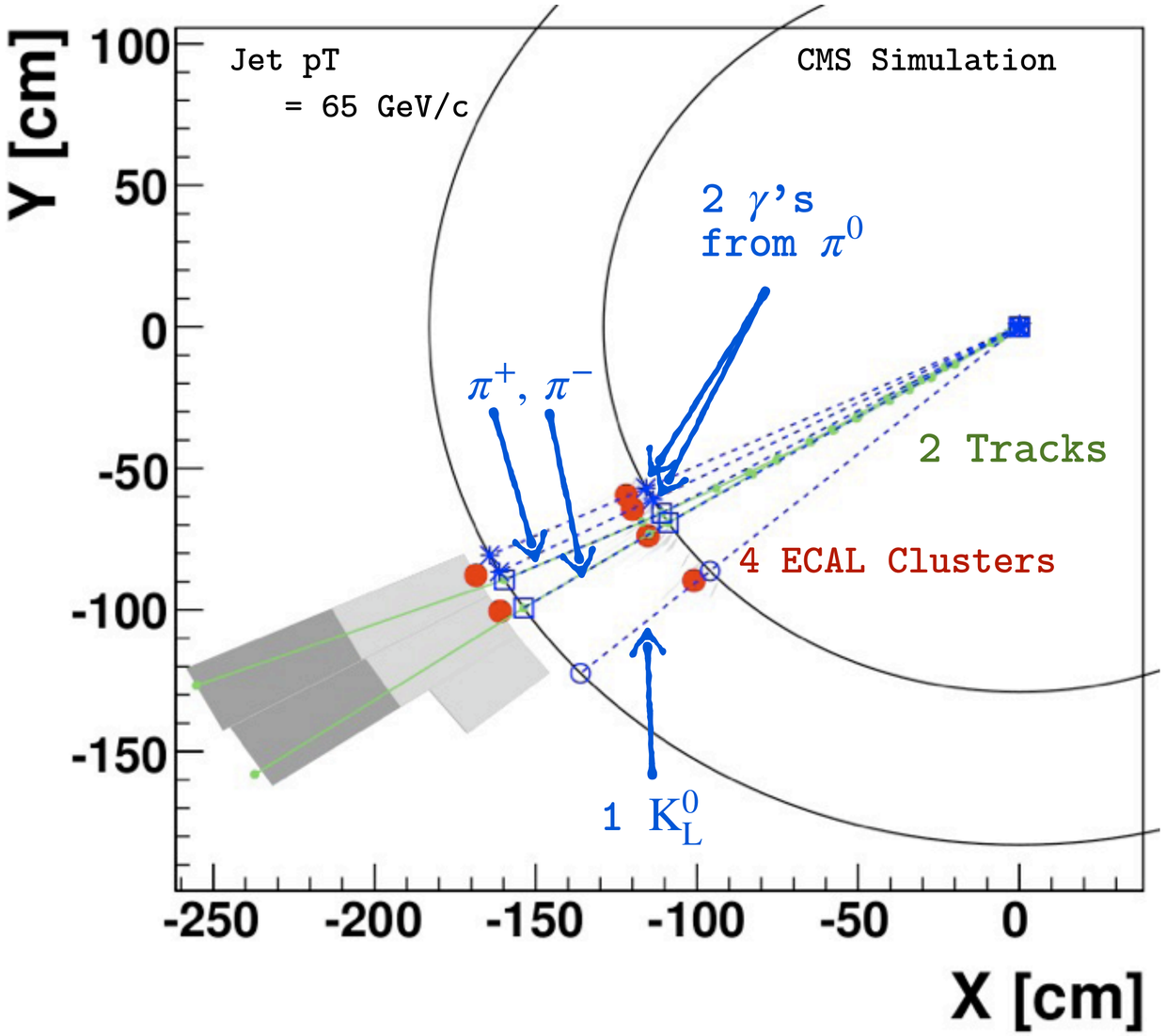
$\pi^+, \pi^-, \pi^0, K_L^0$

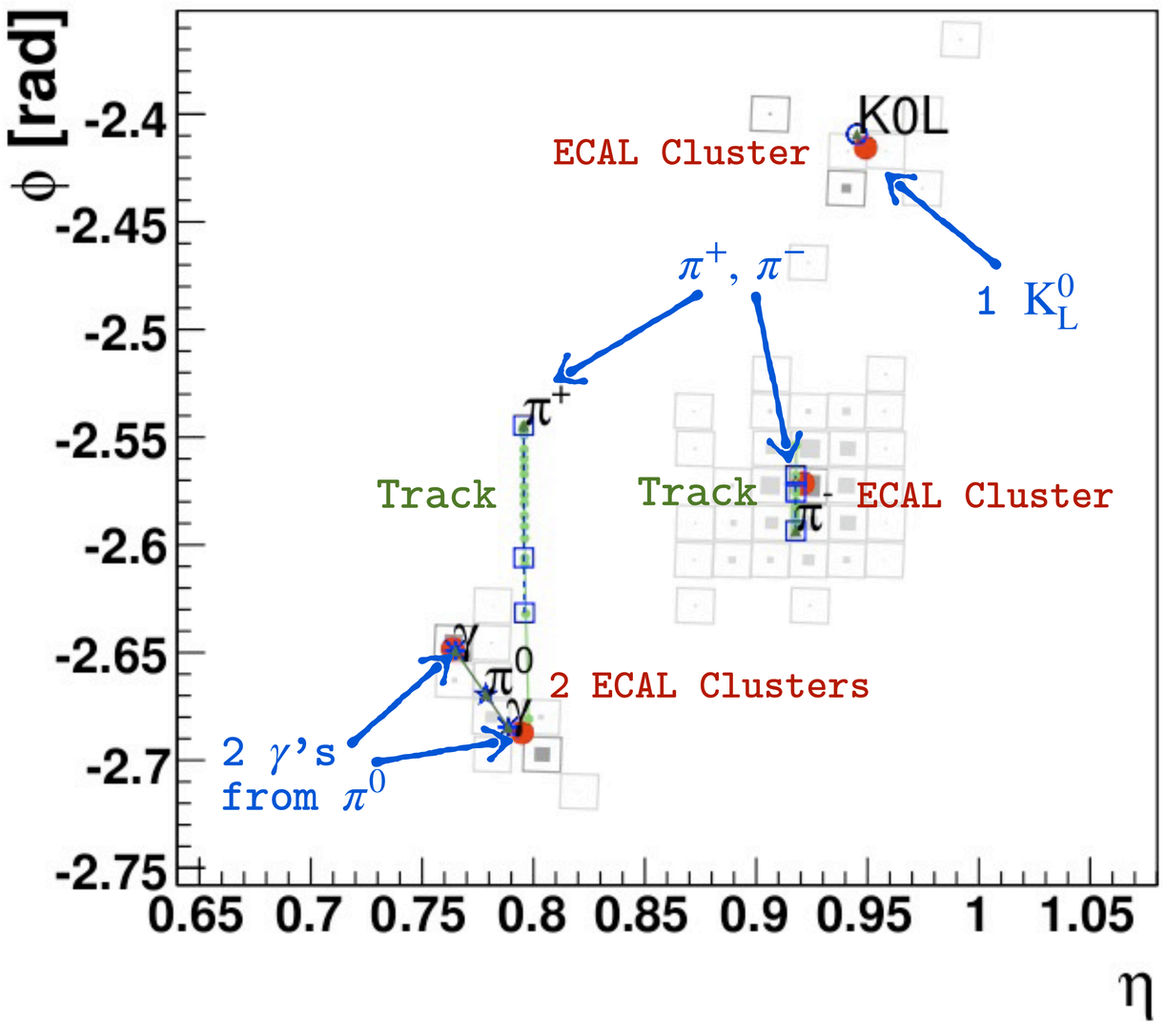




Four true particles:

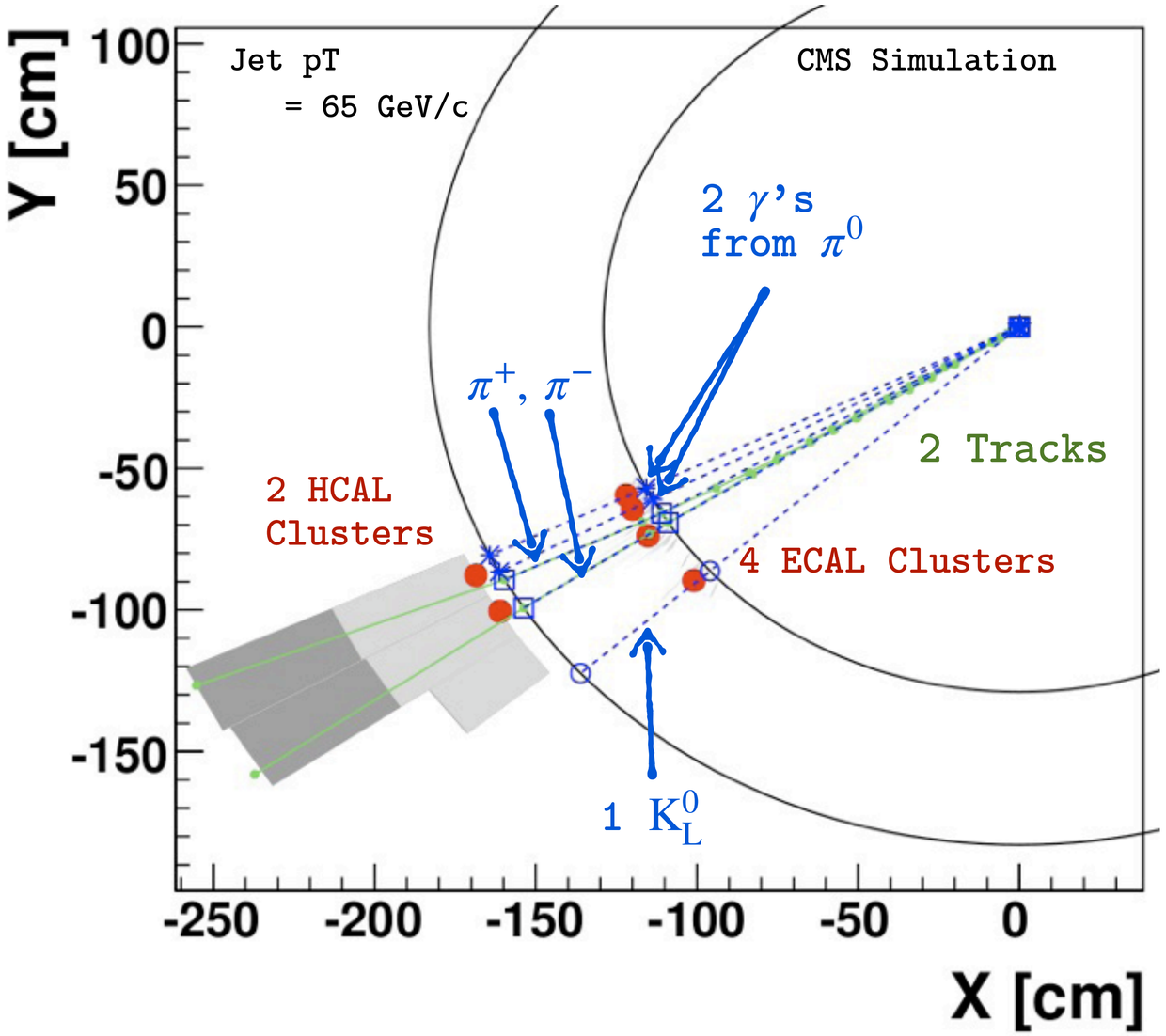
$\pi^+, \pi^-, \pi^0, K_L^0$

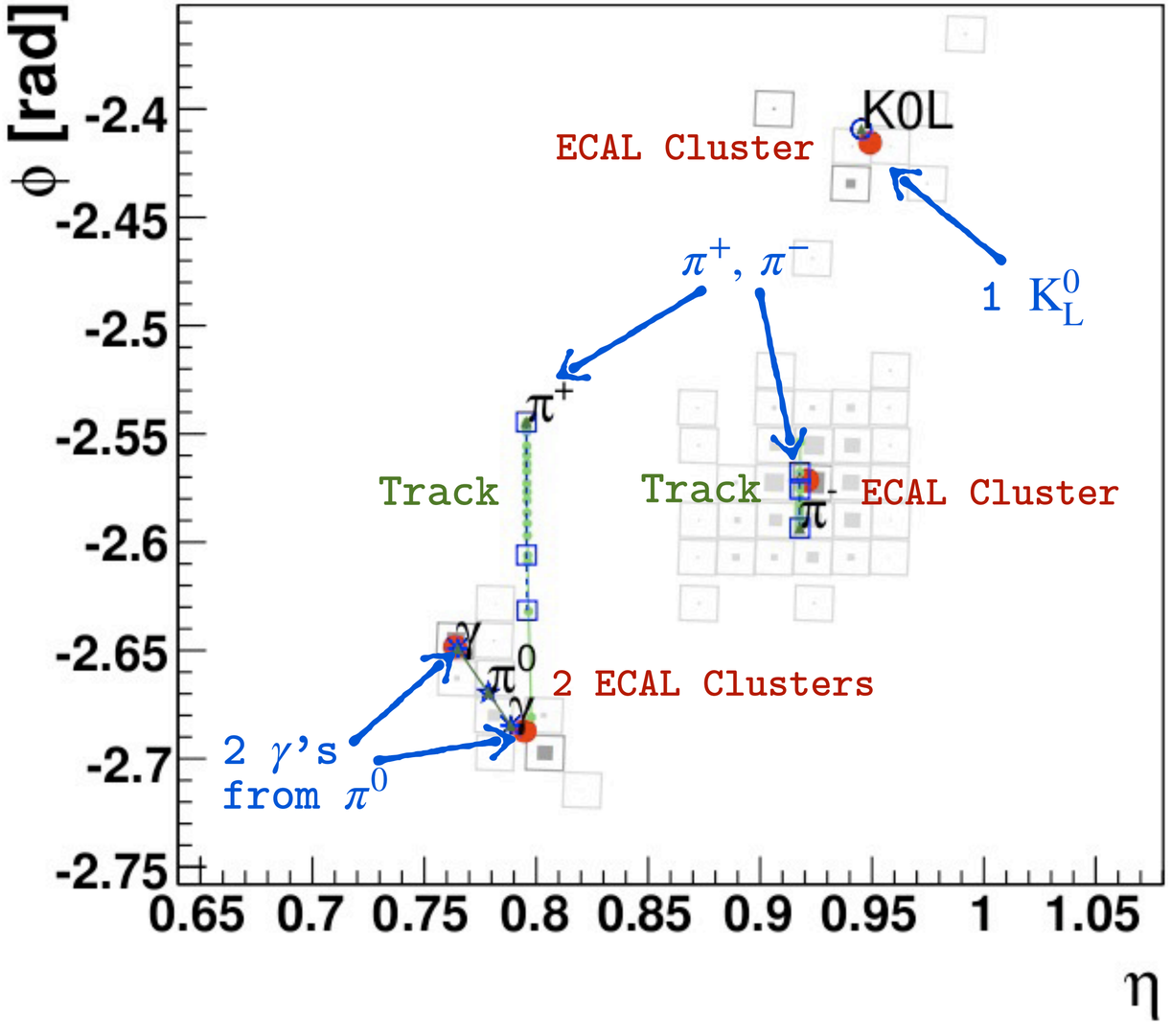




Four true particles:

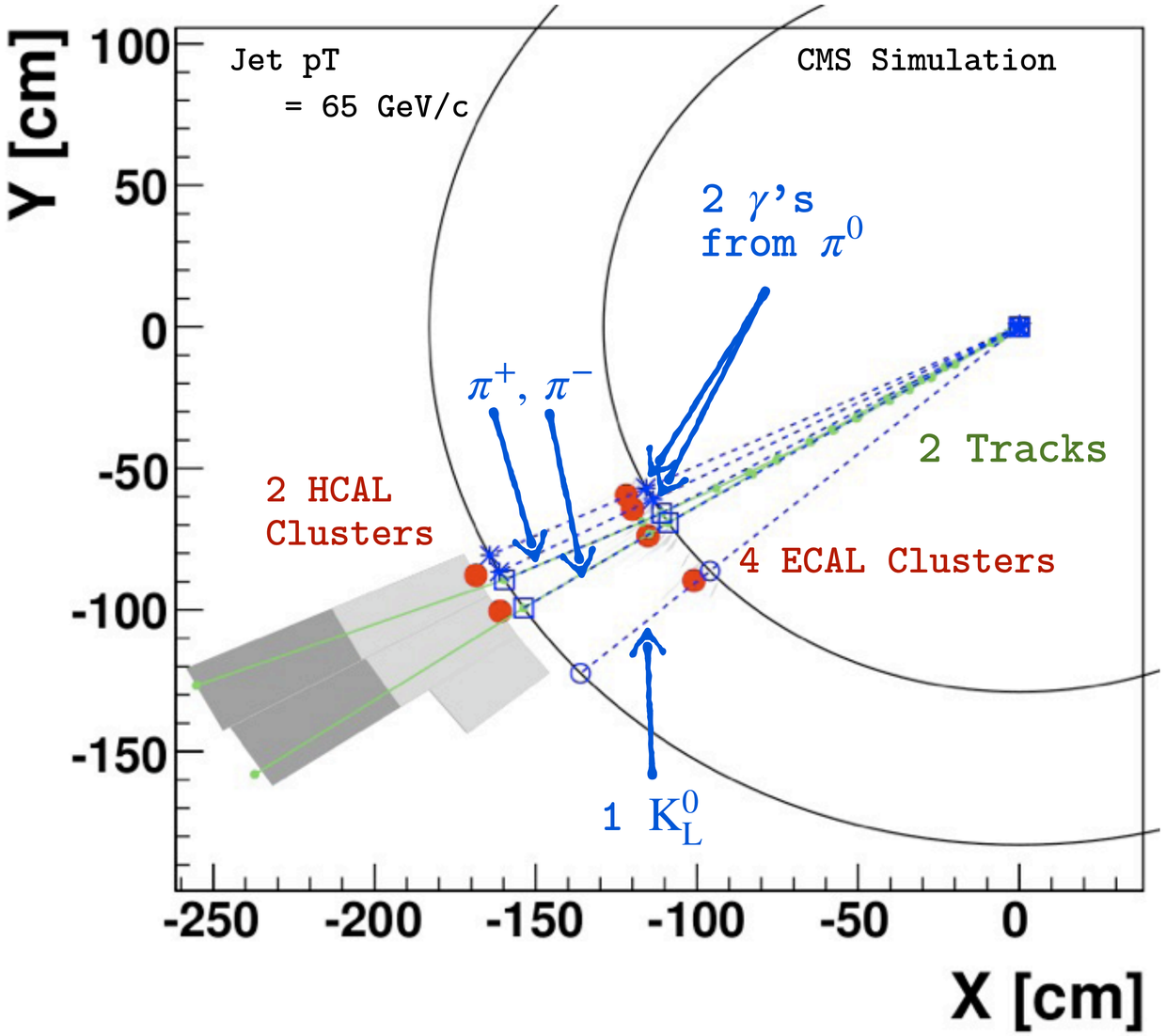
$\pi^+, \pi^-, \pi^0, K_L^0$

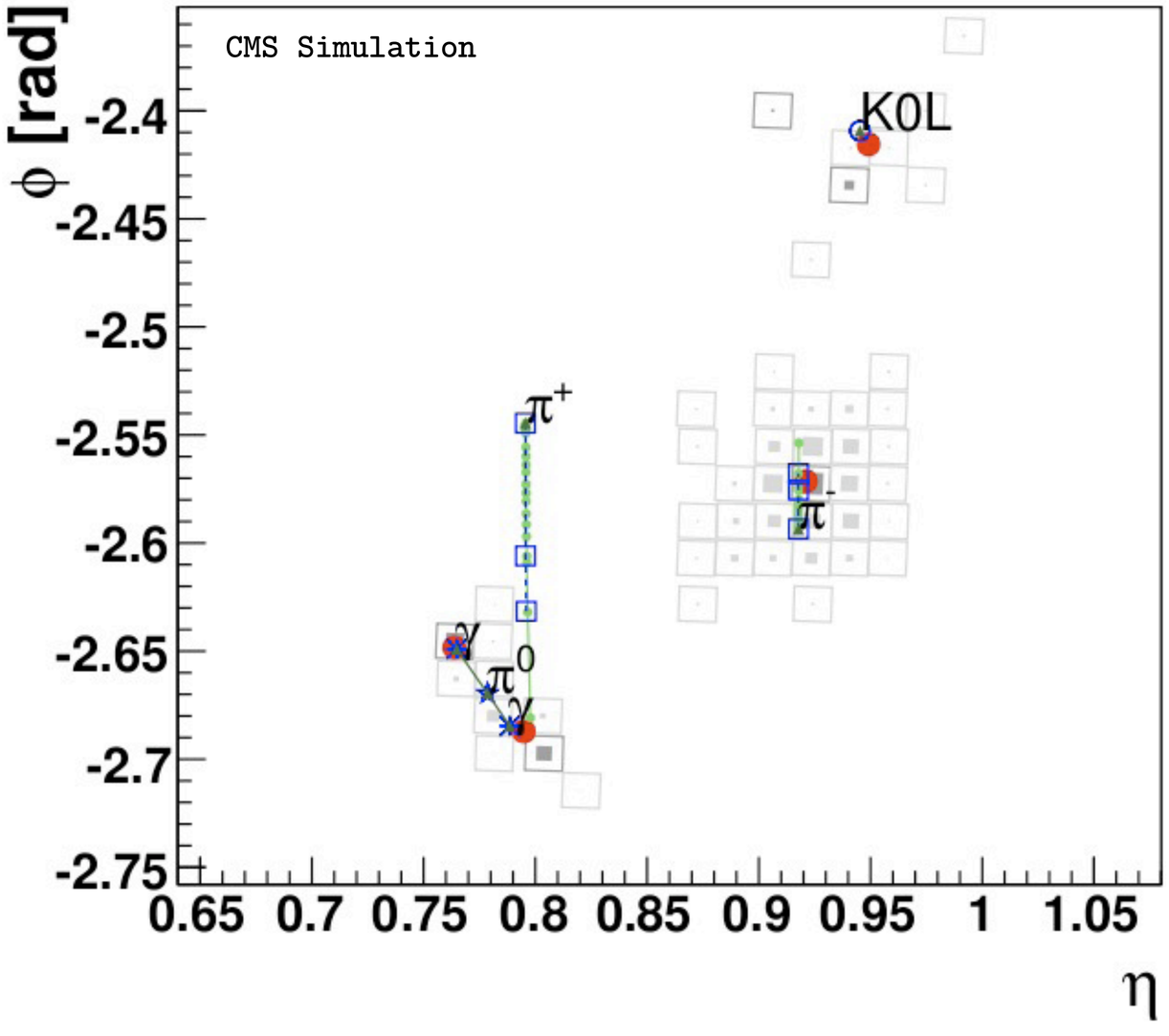




Four true particles:

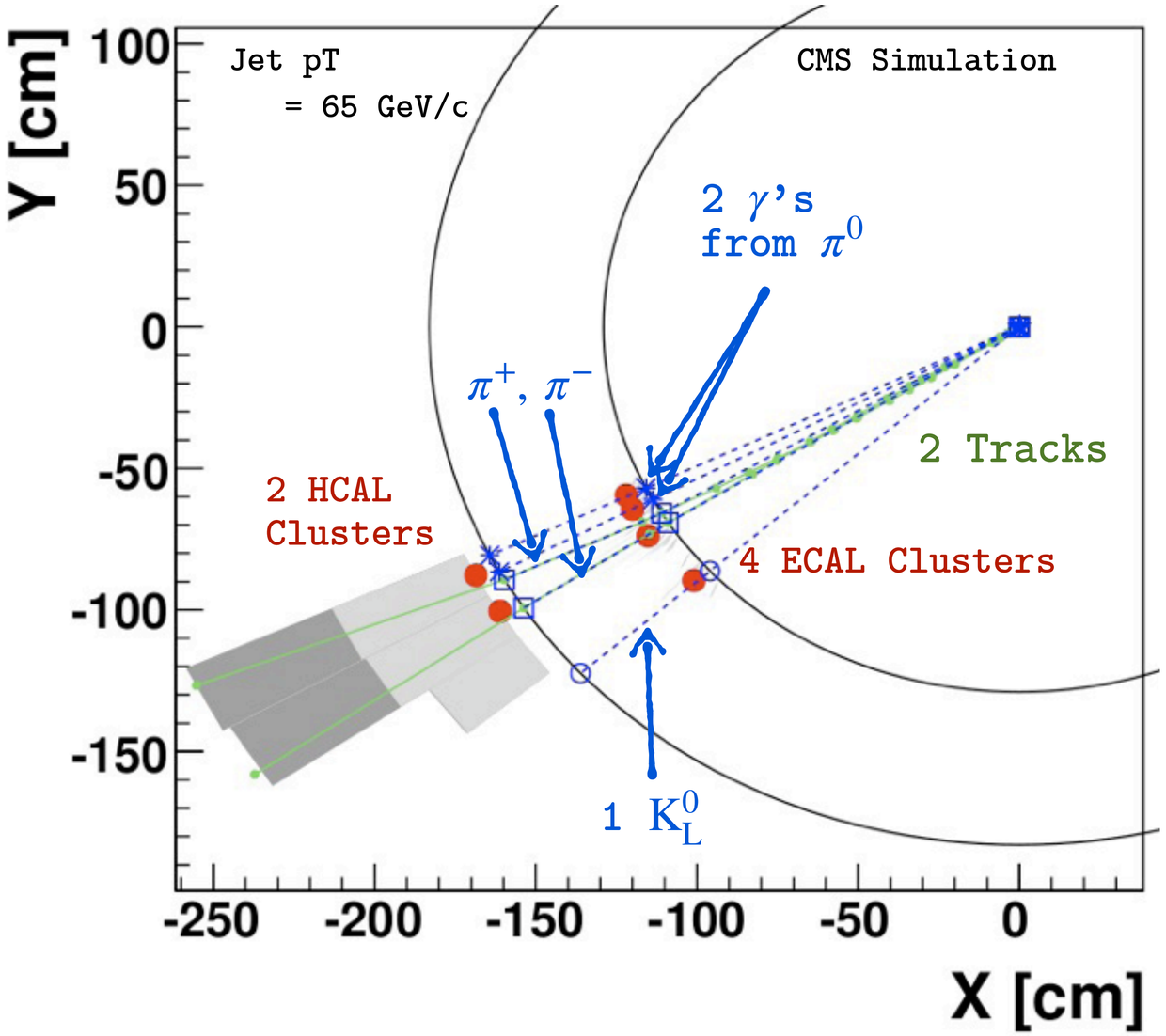
$\pi^+, \pi^-, \pi^0, K_L^0$

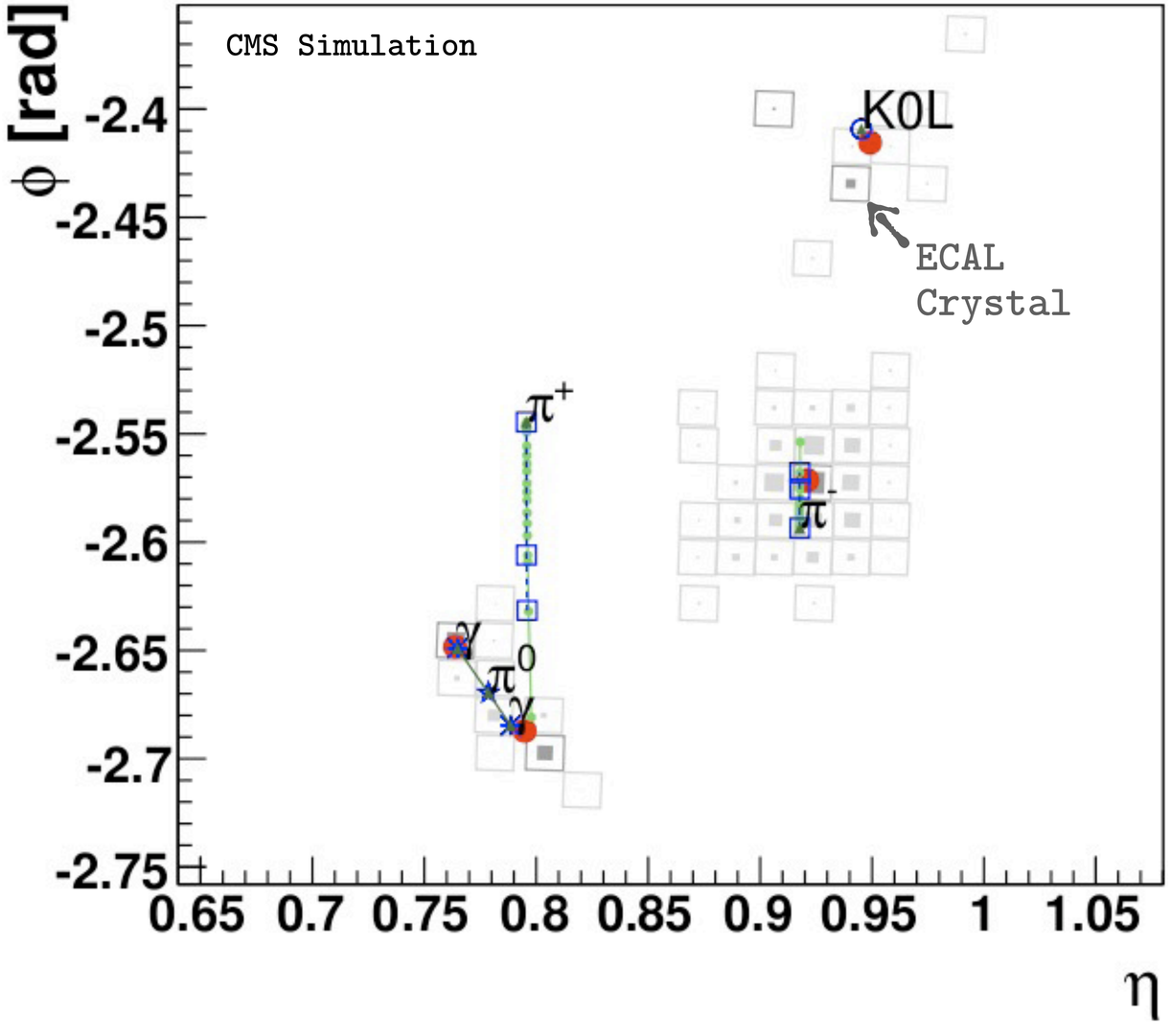




Four true particles:

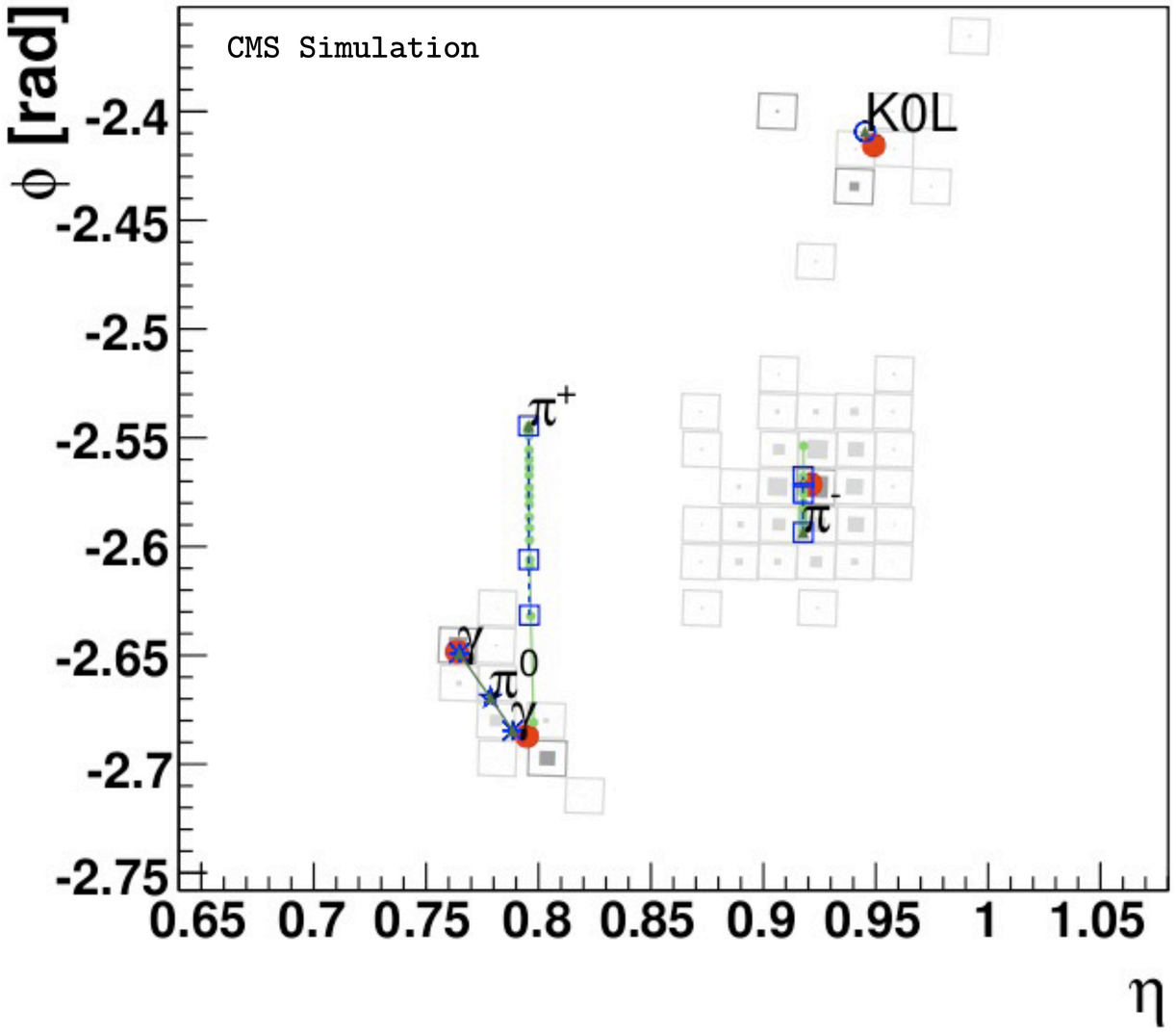
$\pi^+, \pi^-, \pi^0, K_L^0$





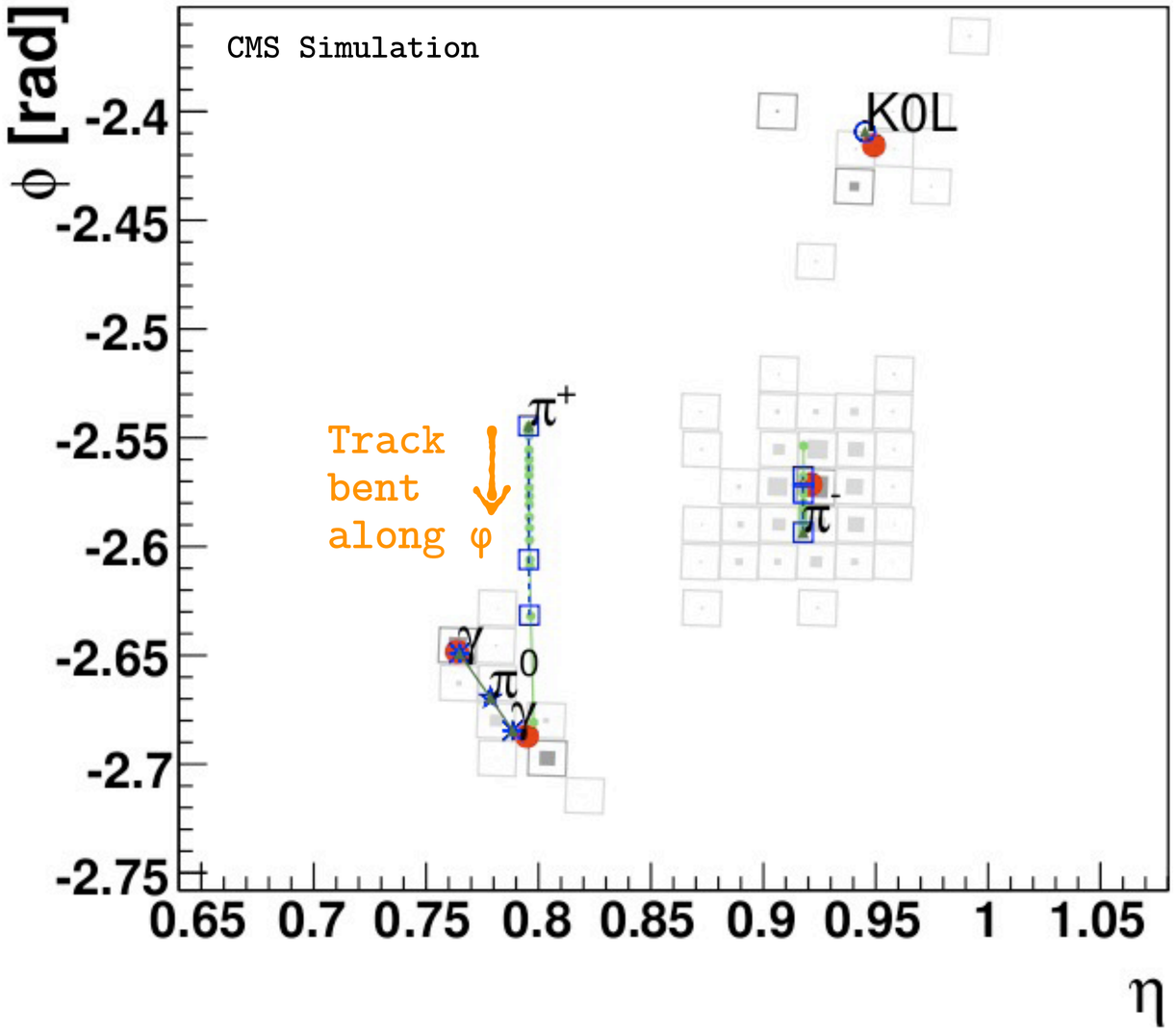
Four true particles:

$$\pi^+, \pi^-, \pi^0, K_L^0$$



Four true particles:

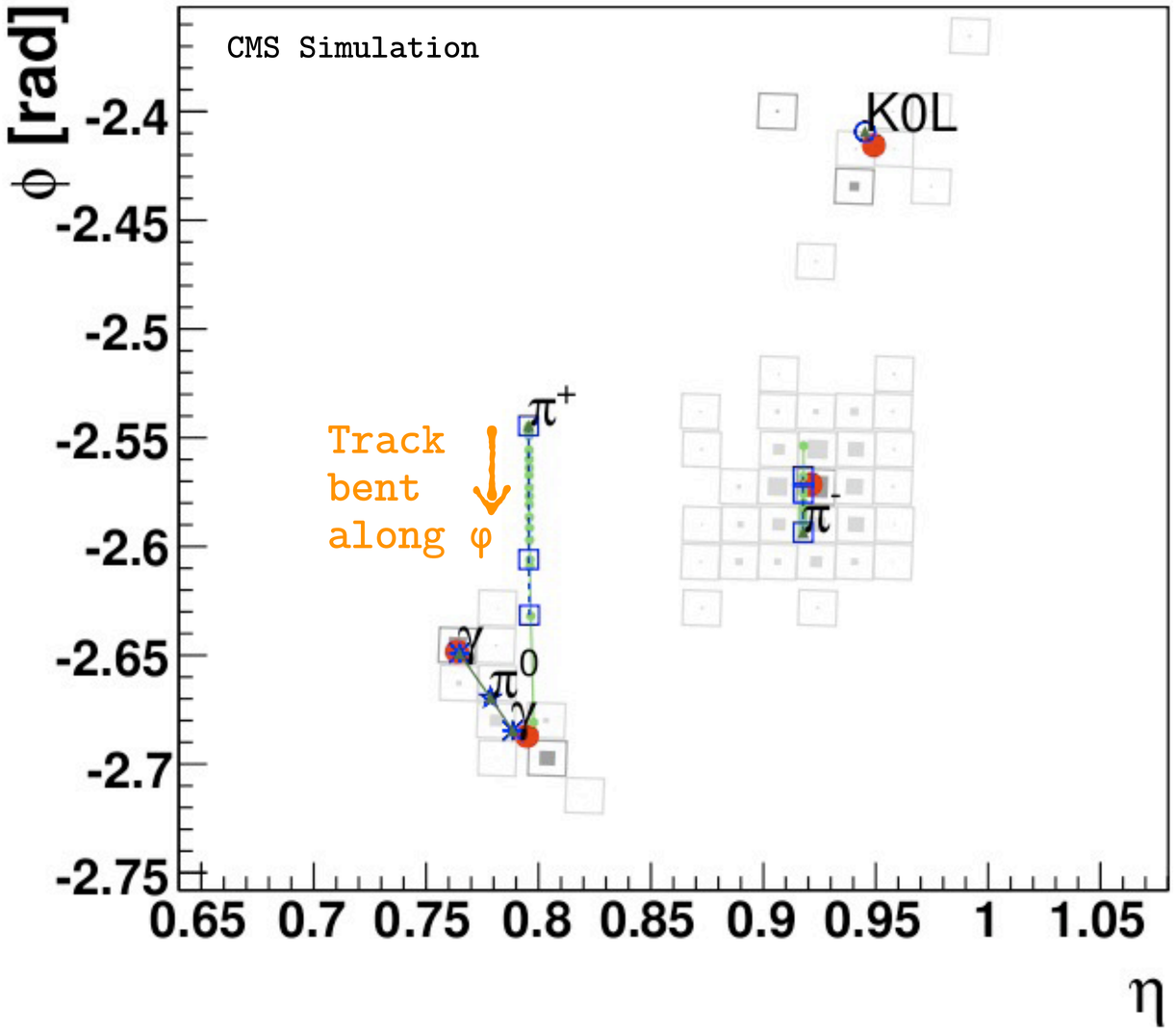
$$\pi^+, \pi^-, \pi^0, K_L^0$$



Four true particles:

$$\pi^+, \pi^-, \pi^0, K_L^0$$

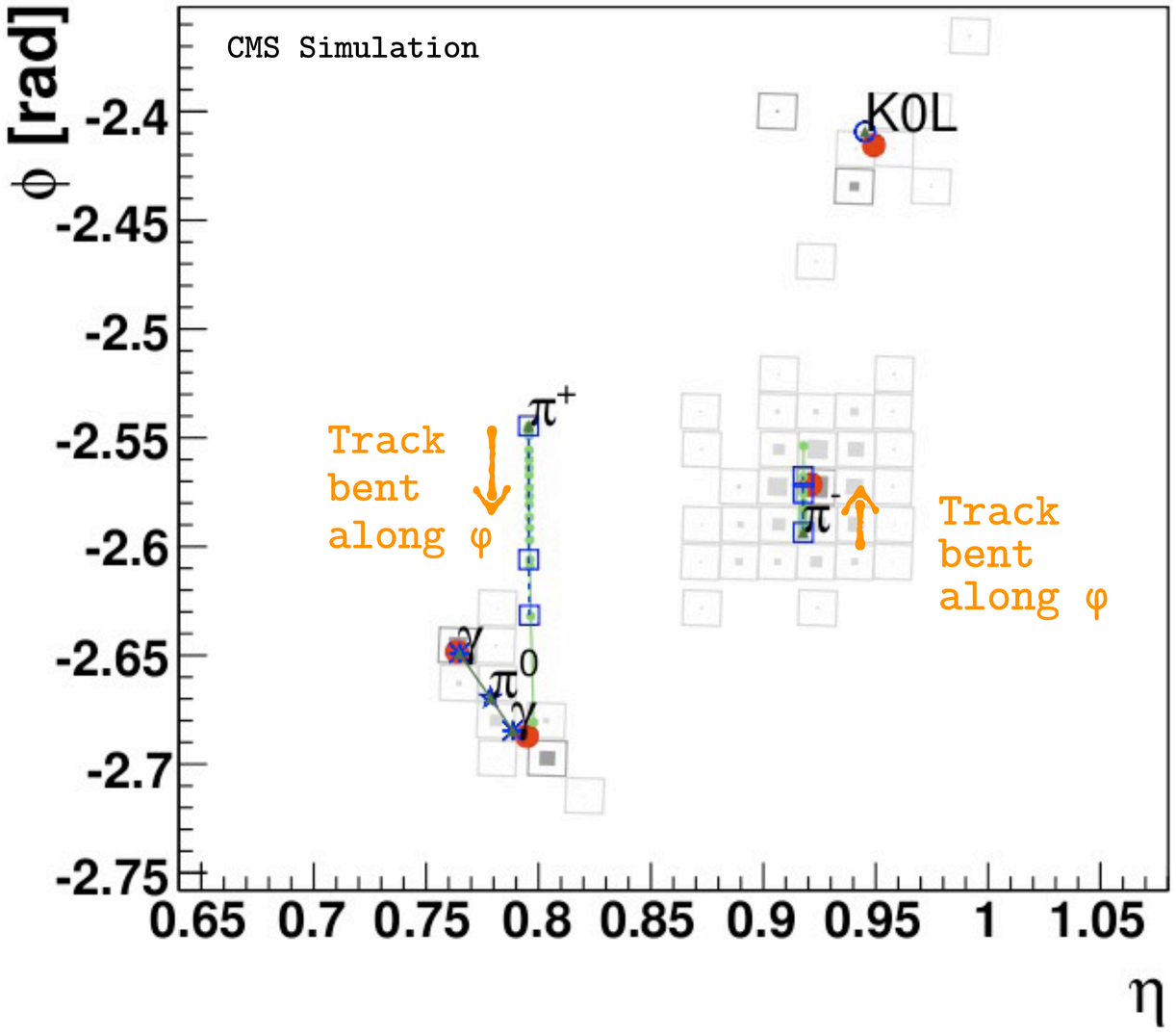
Track



Four true particles:

$$\pi^+, \pi^-, \pi^0, K_L^0$$

Track

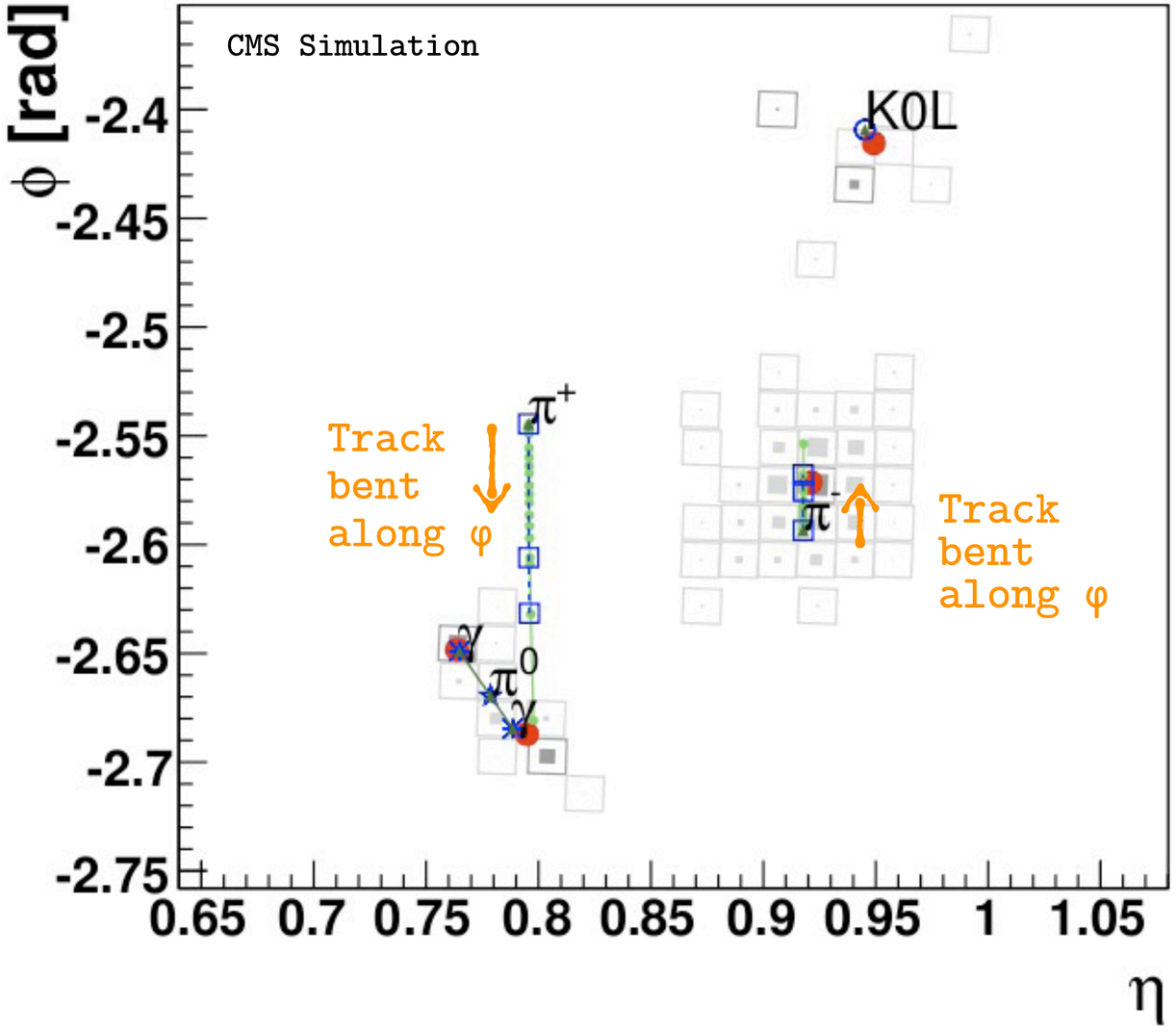


Four true particles:

$$\pi^+, \pi^-, \pi^0, K_L^0$$

Track

Track

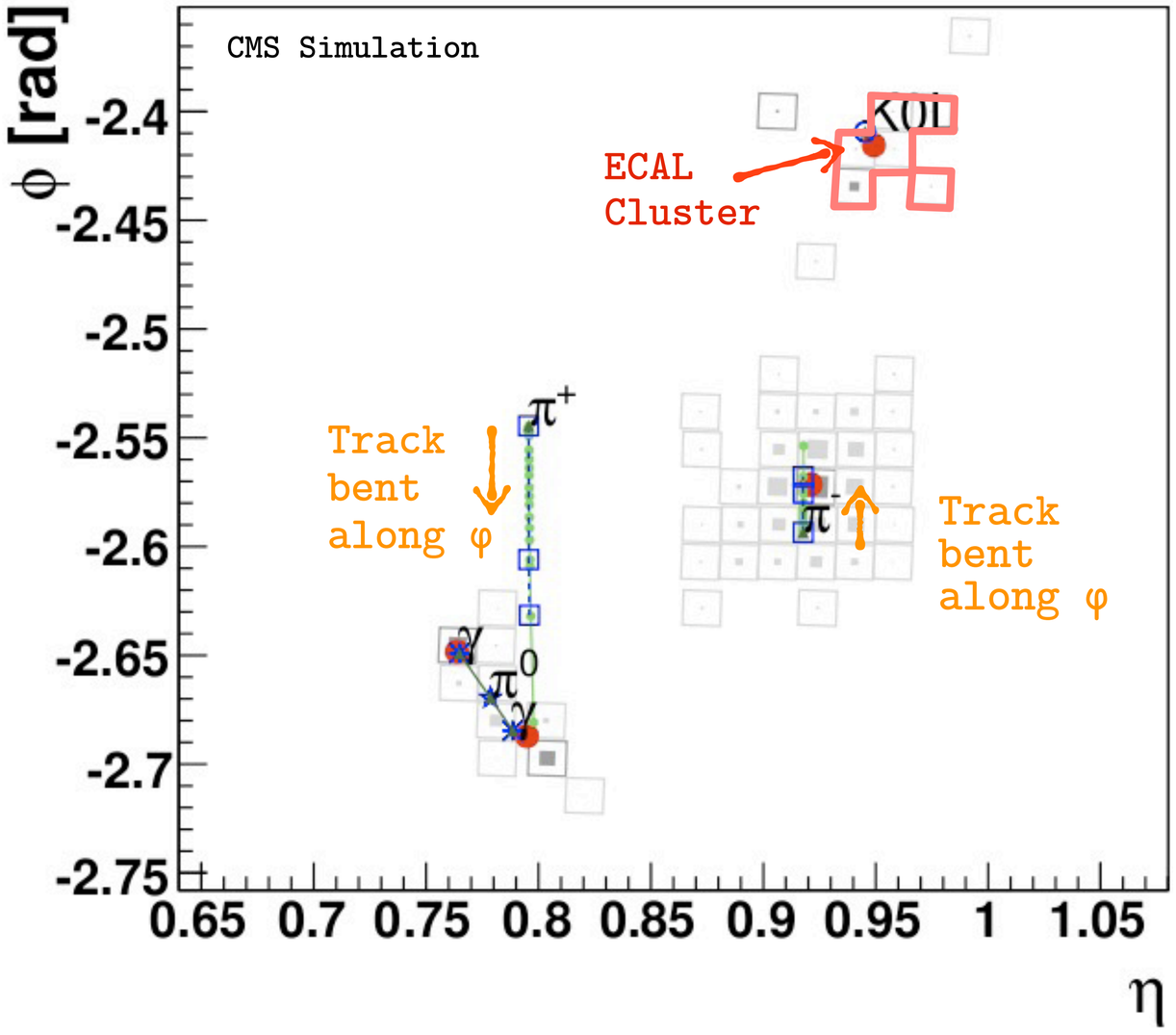


Four true particles:

$$\pi^+, \pi^-, \pi^0, K_L^0$$

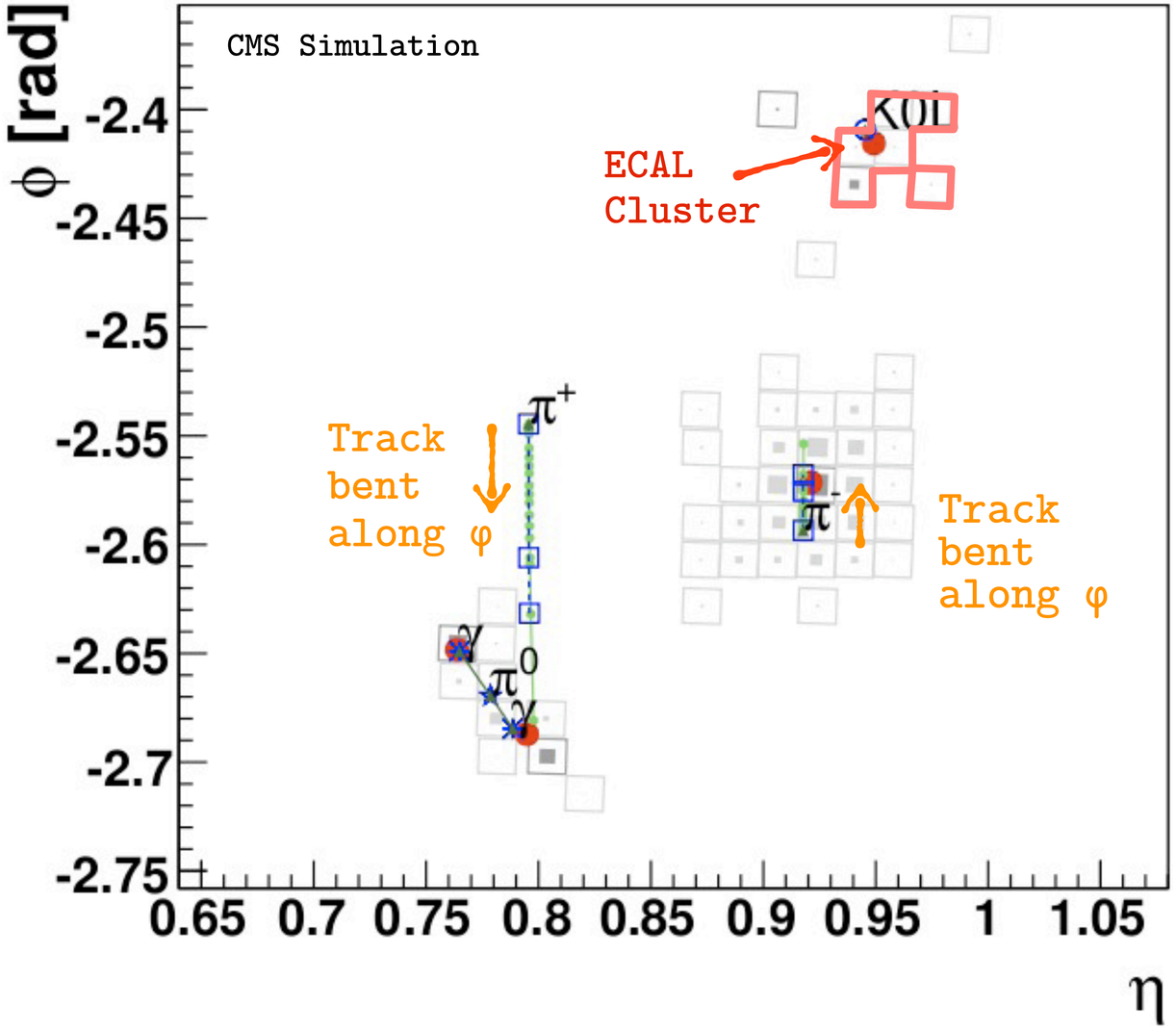
Track

Track



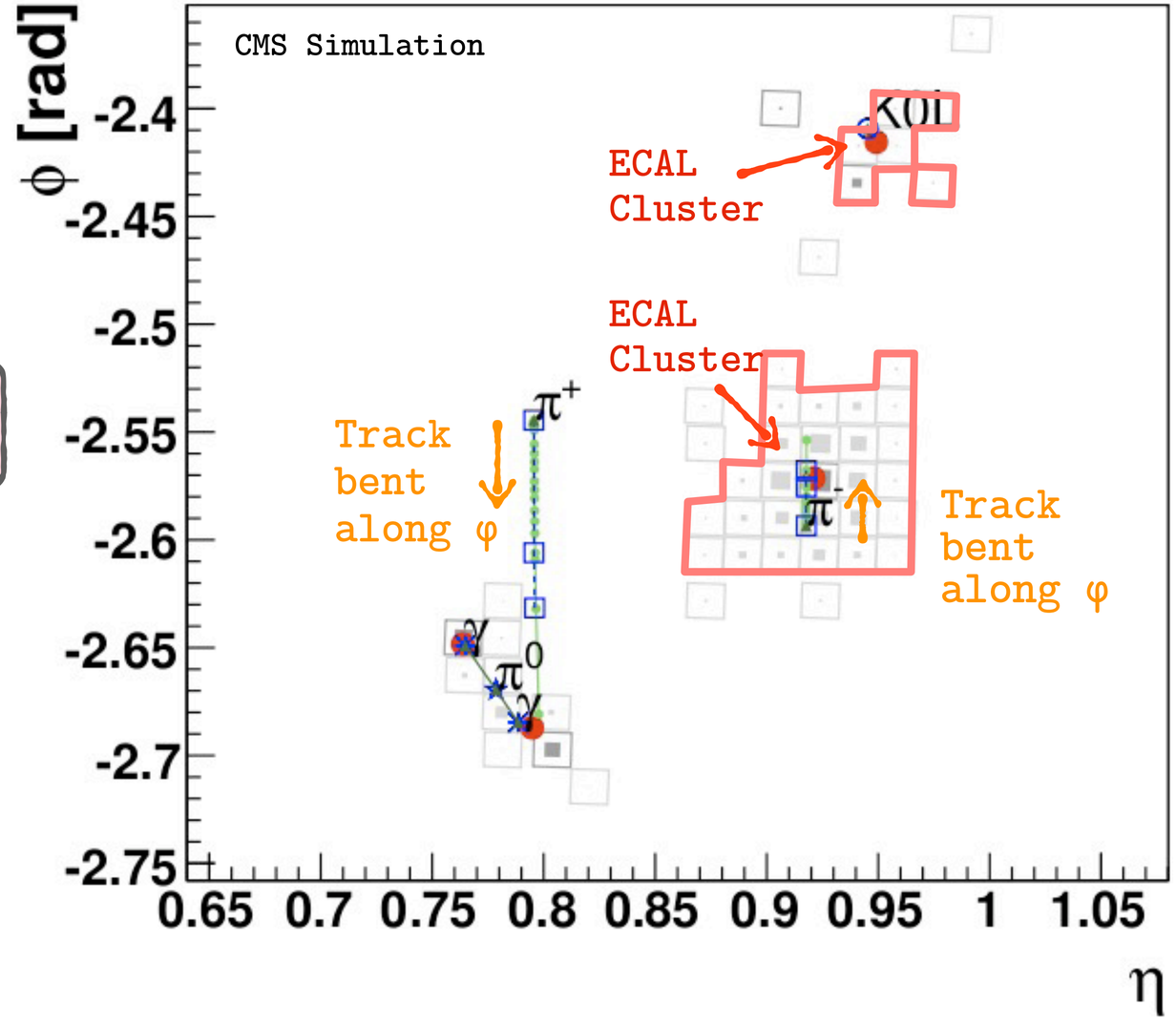
Four true particles:

$$\pi^+, \pi^-, \pi^0, K_L^0$$



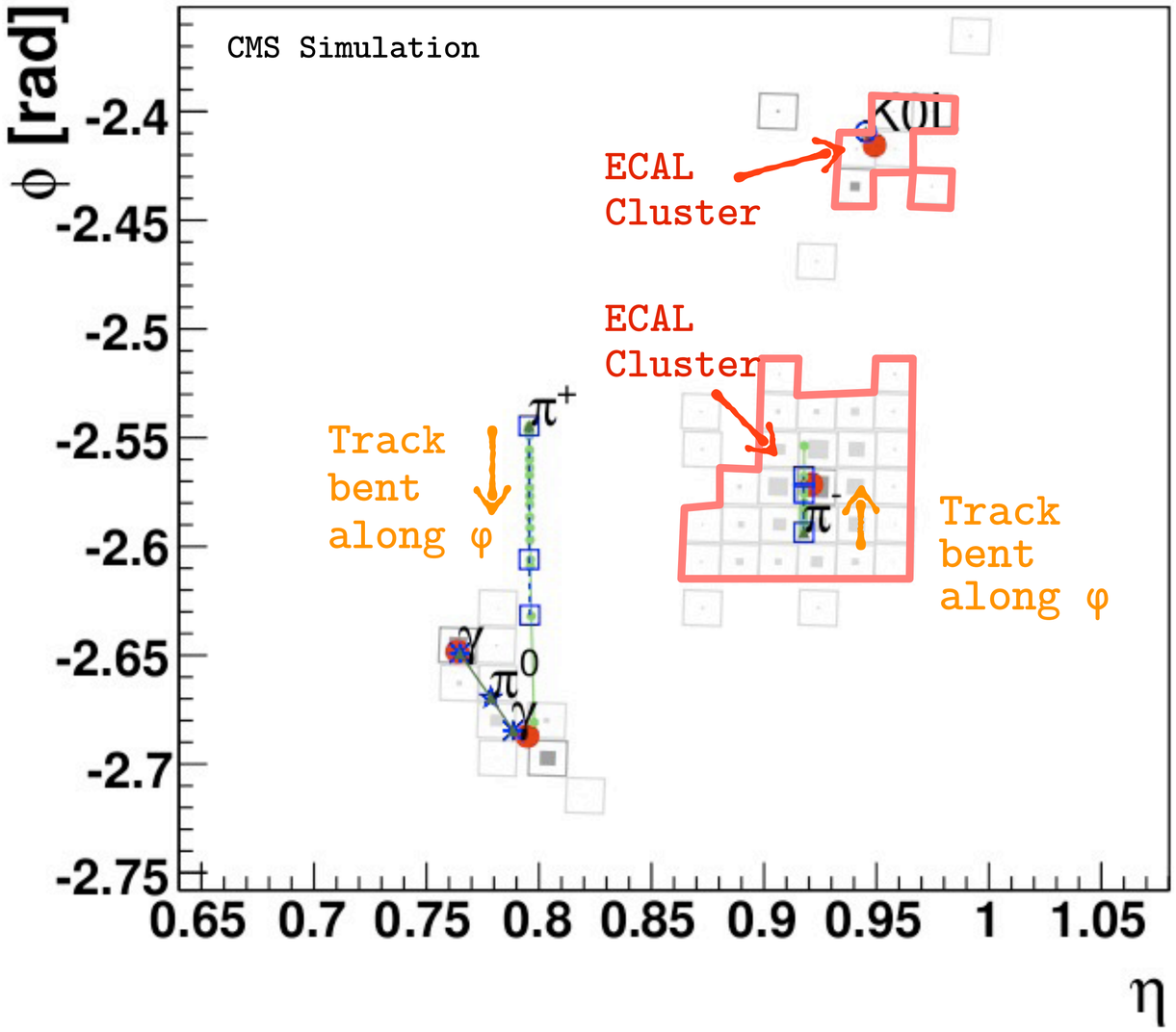
Four true particles:

$$\pi^+, \pi^-, \pi^0, K_L^0$$



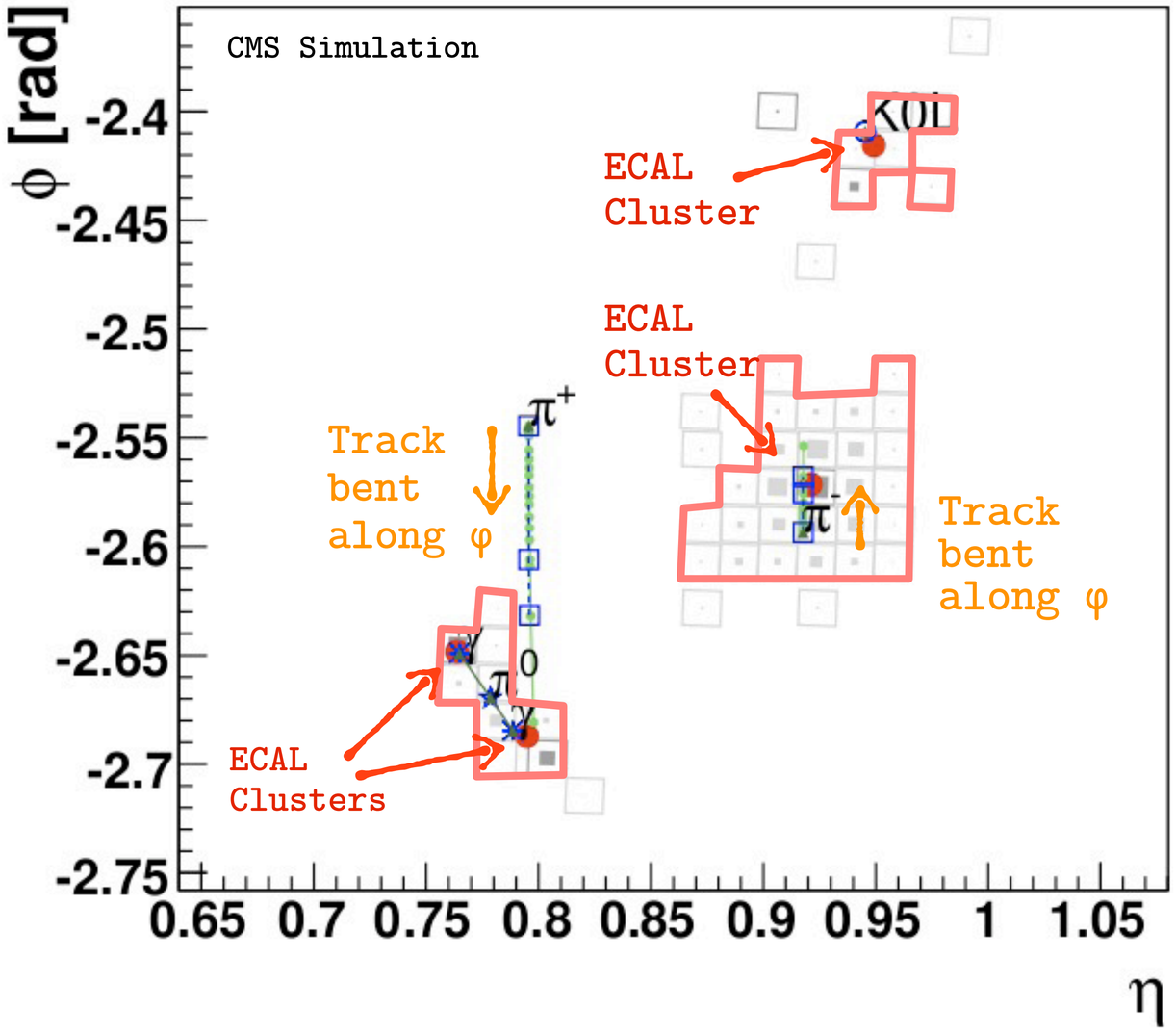
Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$



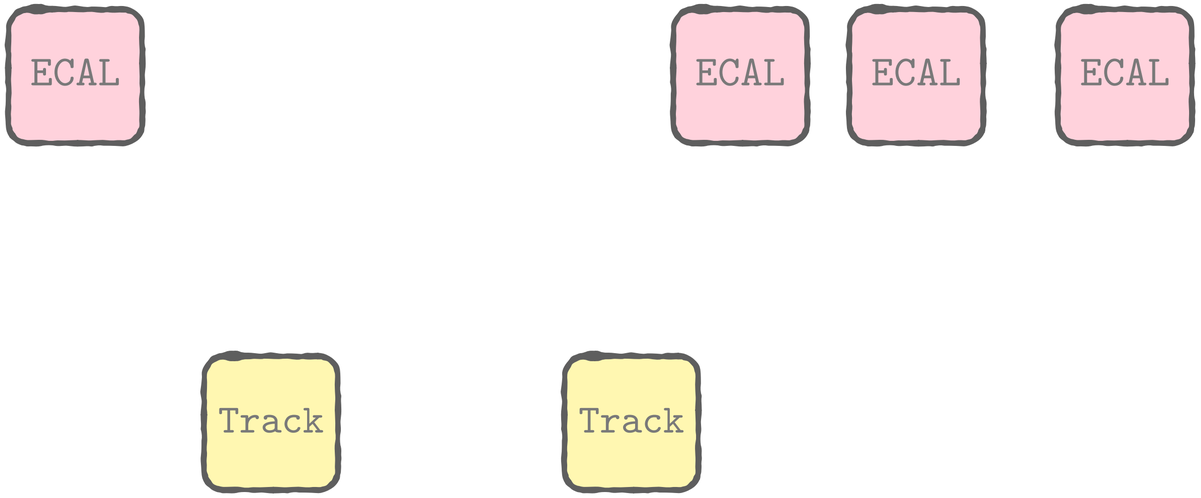
Four true particles:

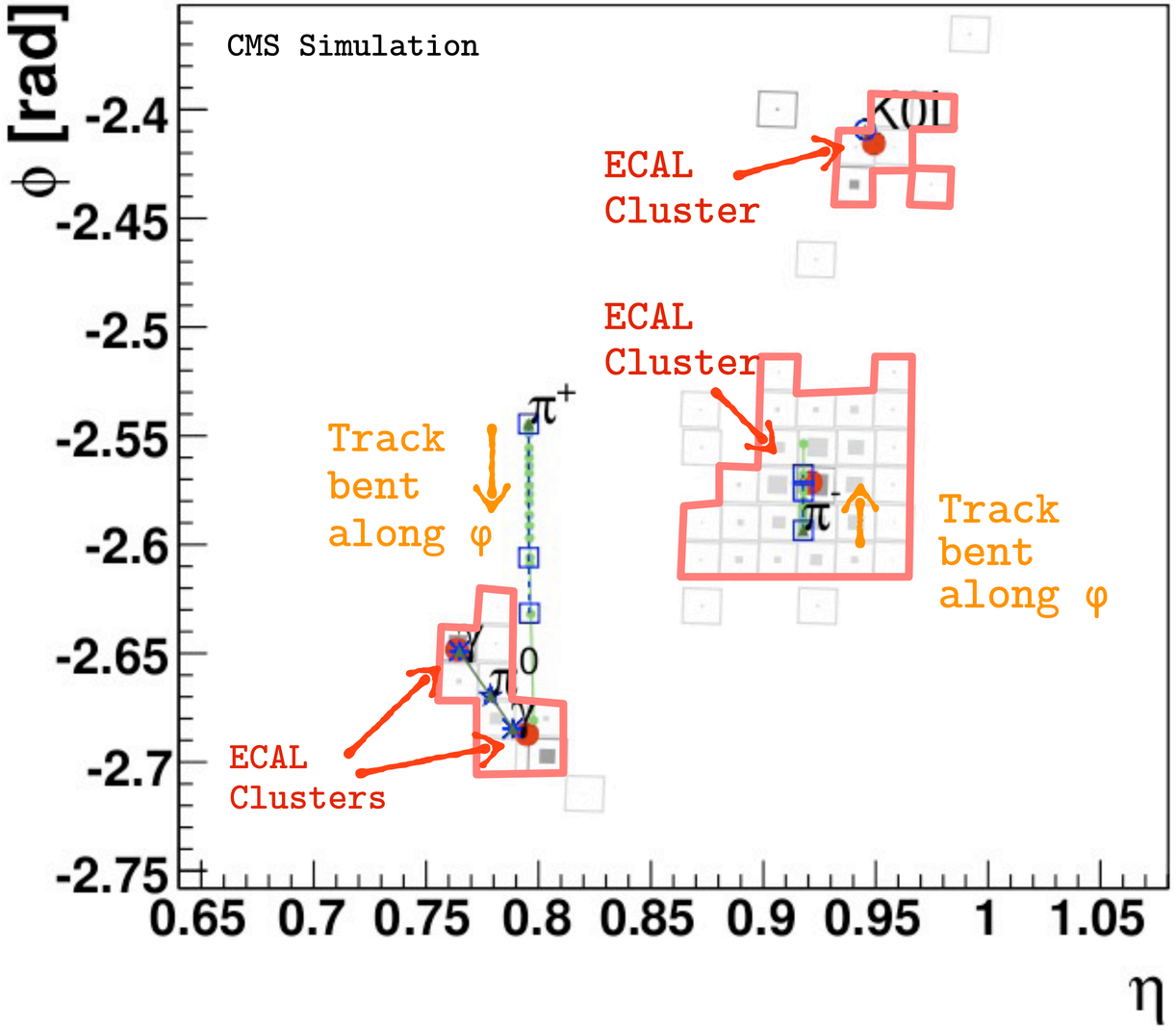
$\pi^+, \pi^-, \pi^0, K_L^0$



Four true particles:

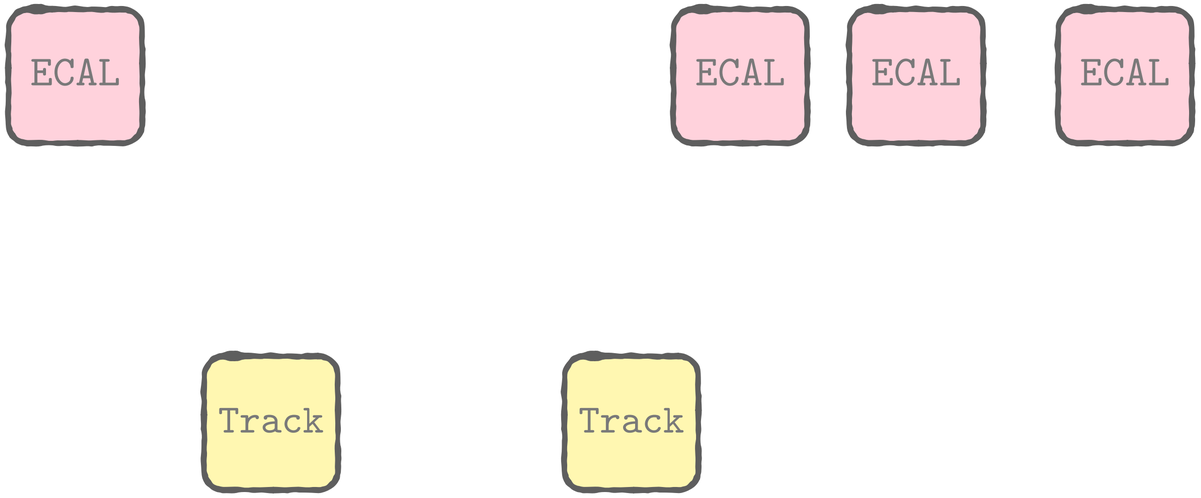
$\pi^+, \pi^-, \pi^0, K_L^0$

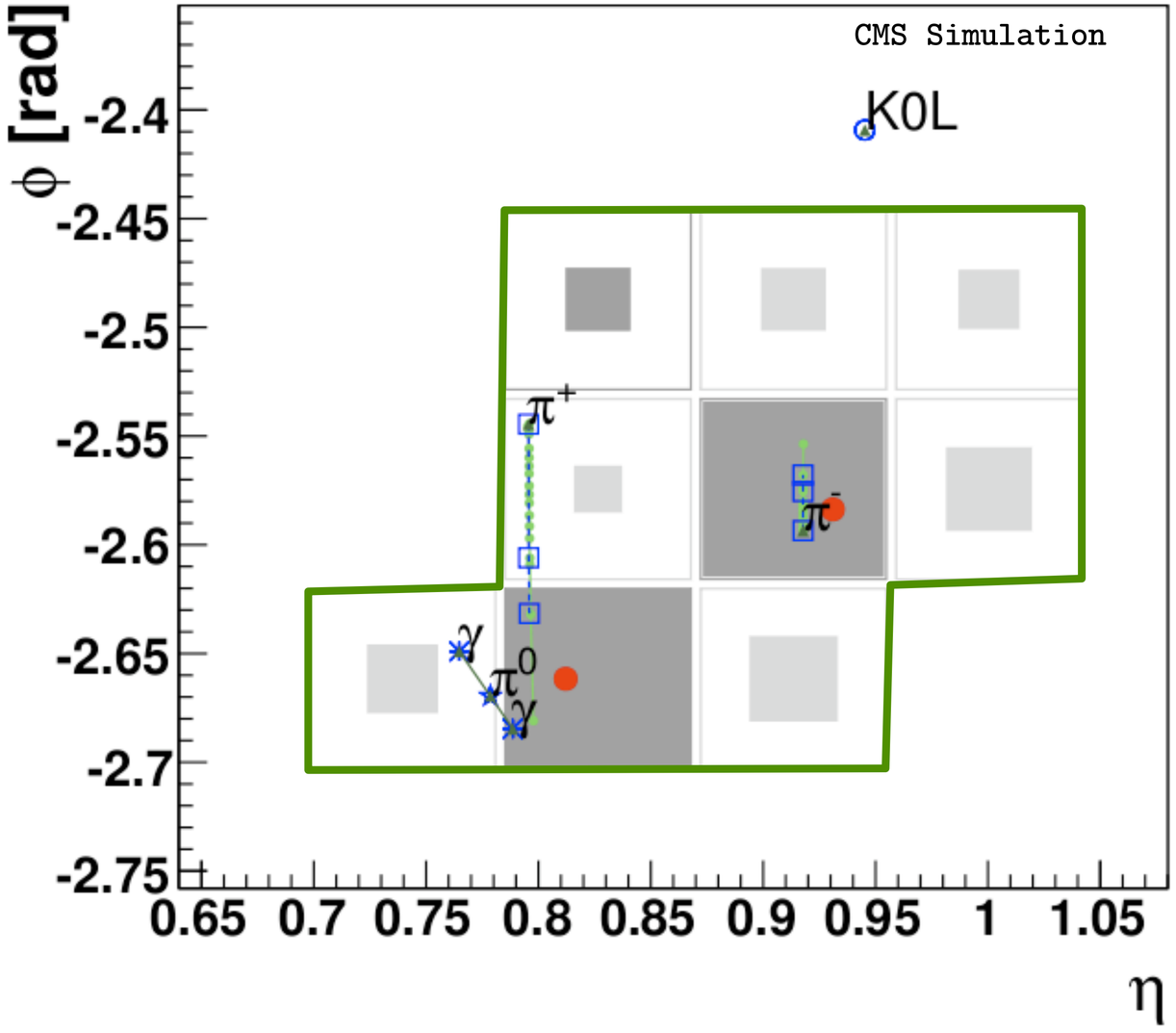




Four true particles:

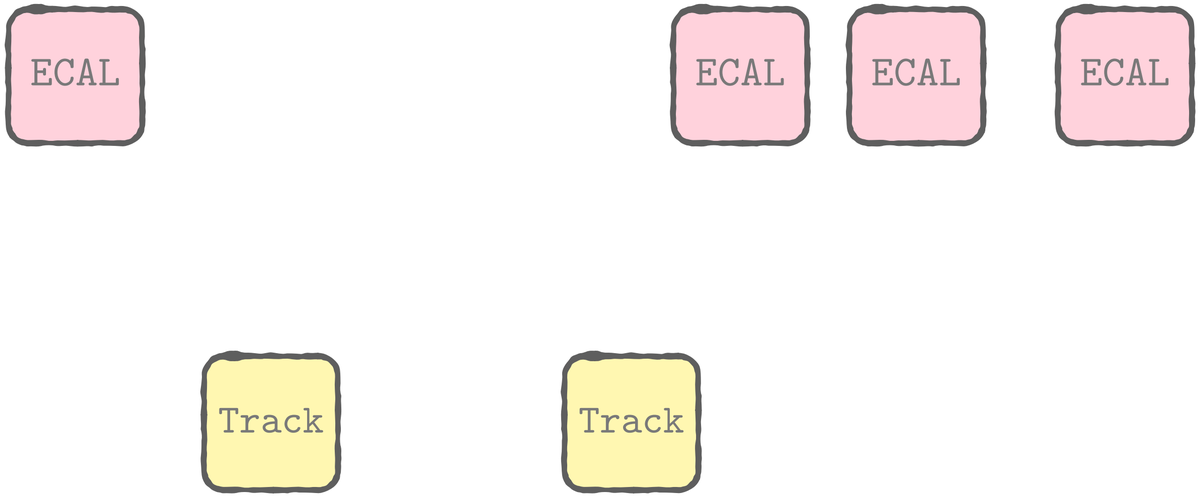
$\pi^+, \pi^-, \pi^0, K_L^0$

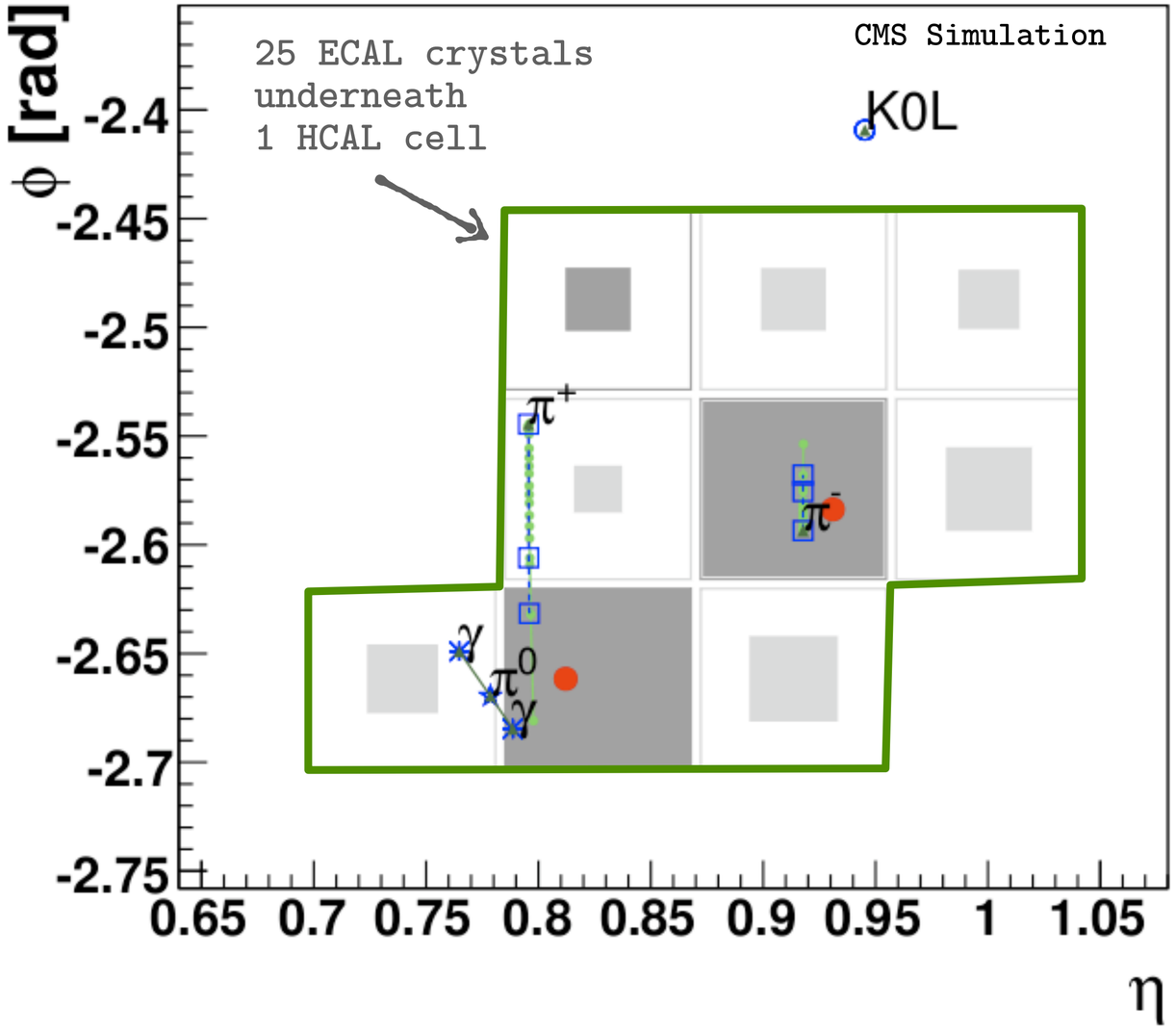




Four true particles:

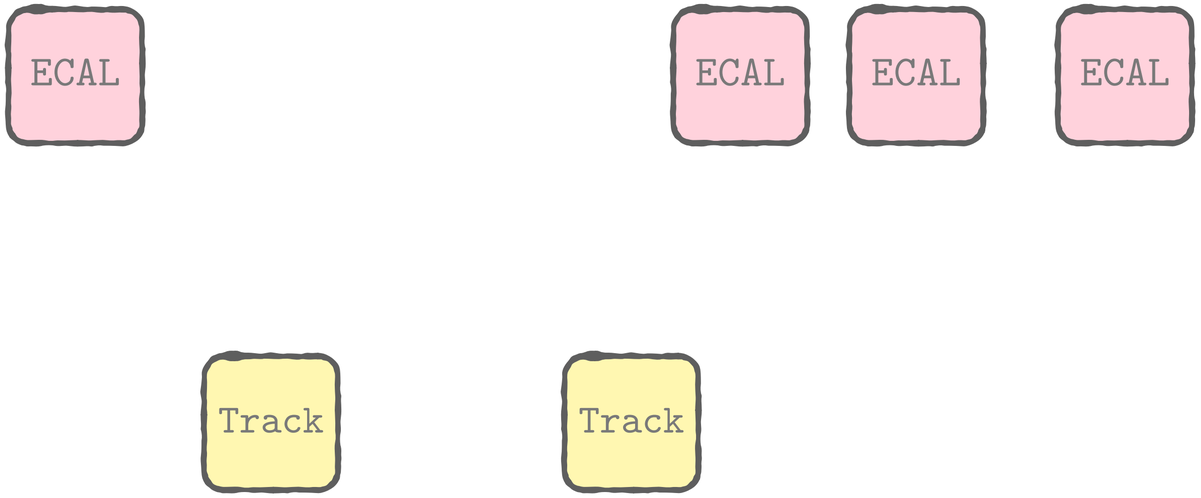
$\pi^+, \pi^-, \pi^0, K_L^0$

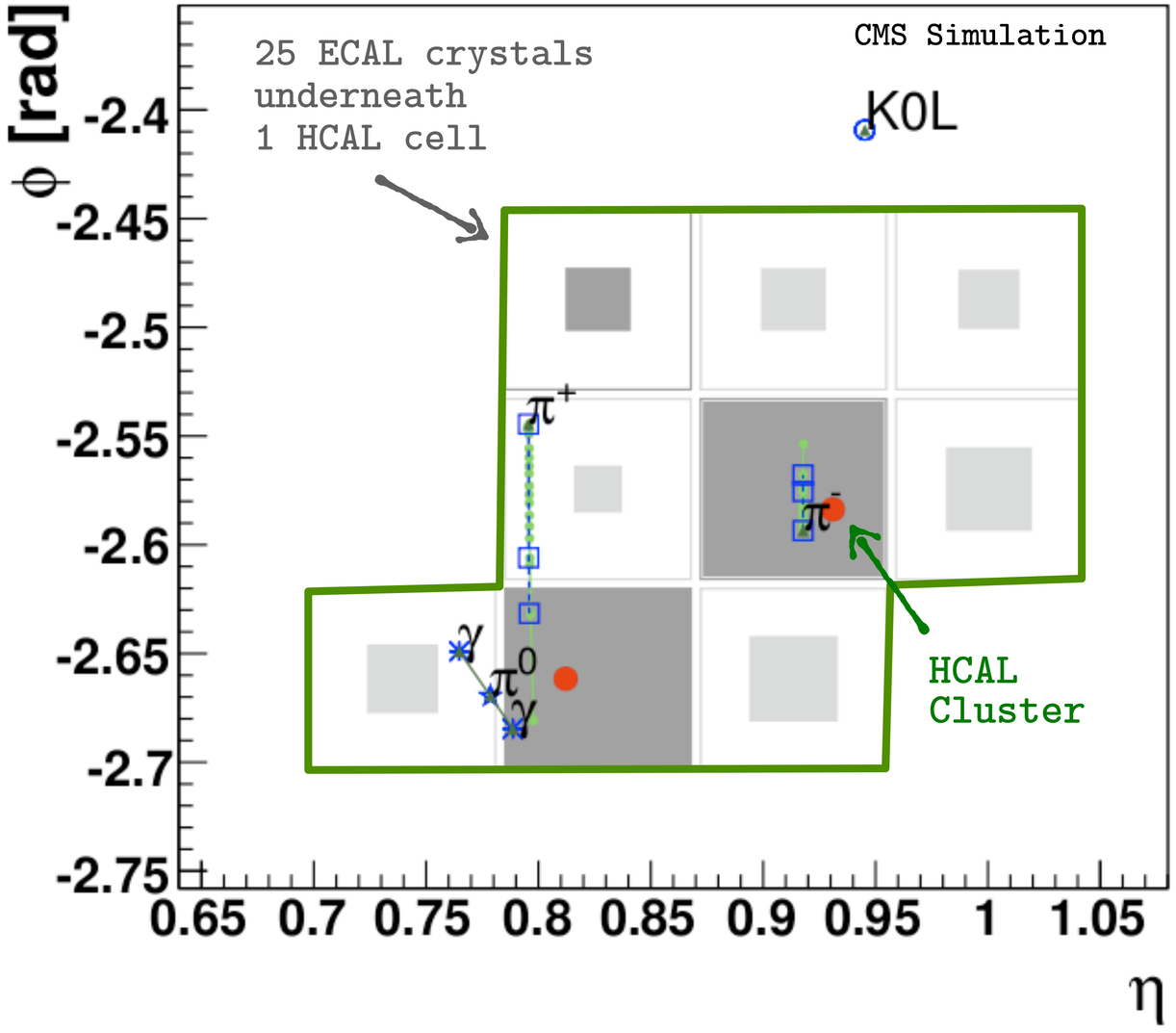




Four true particles:

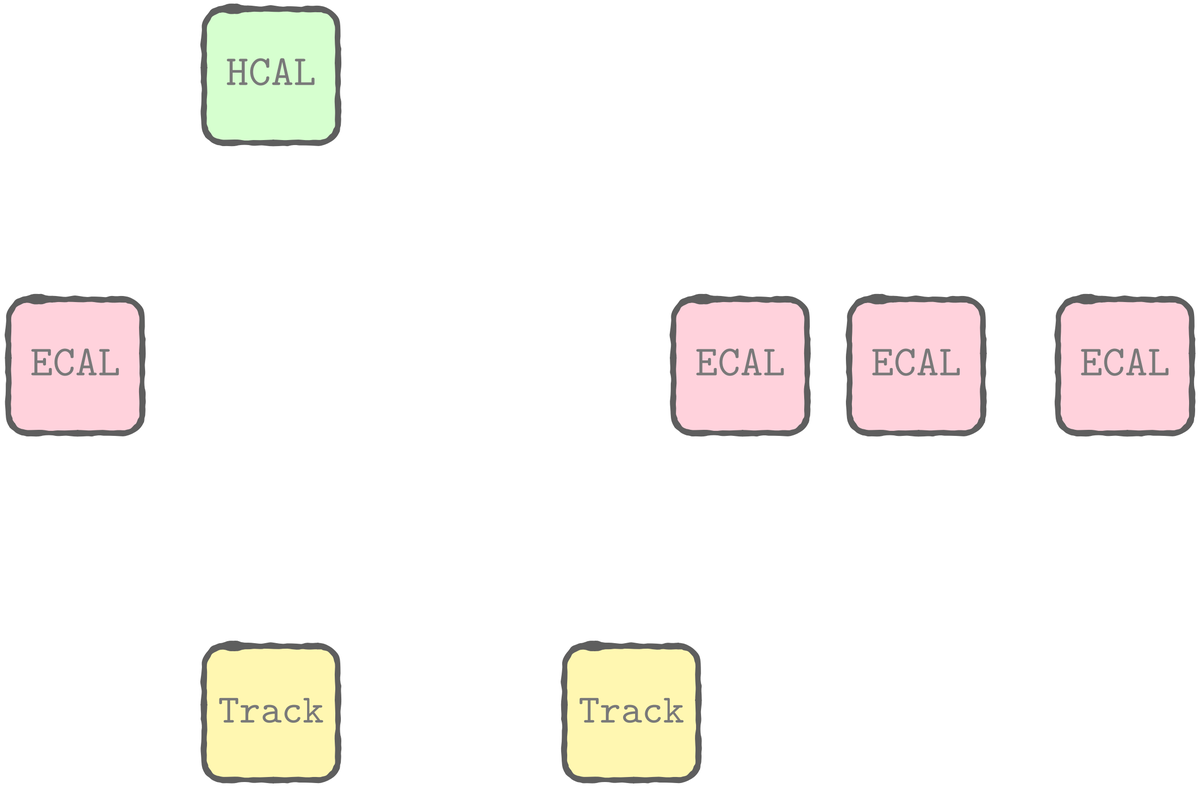
$\pi^+, \pi^-, \pi^0, K_L^0$

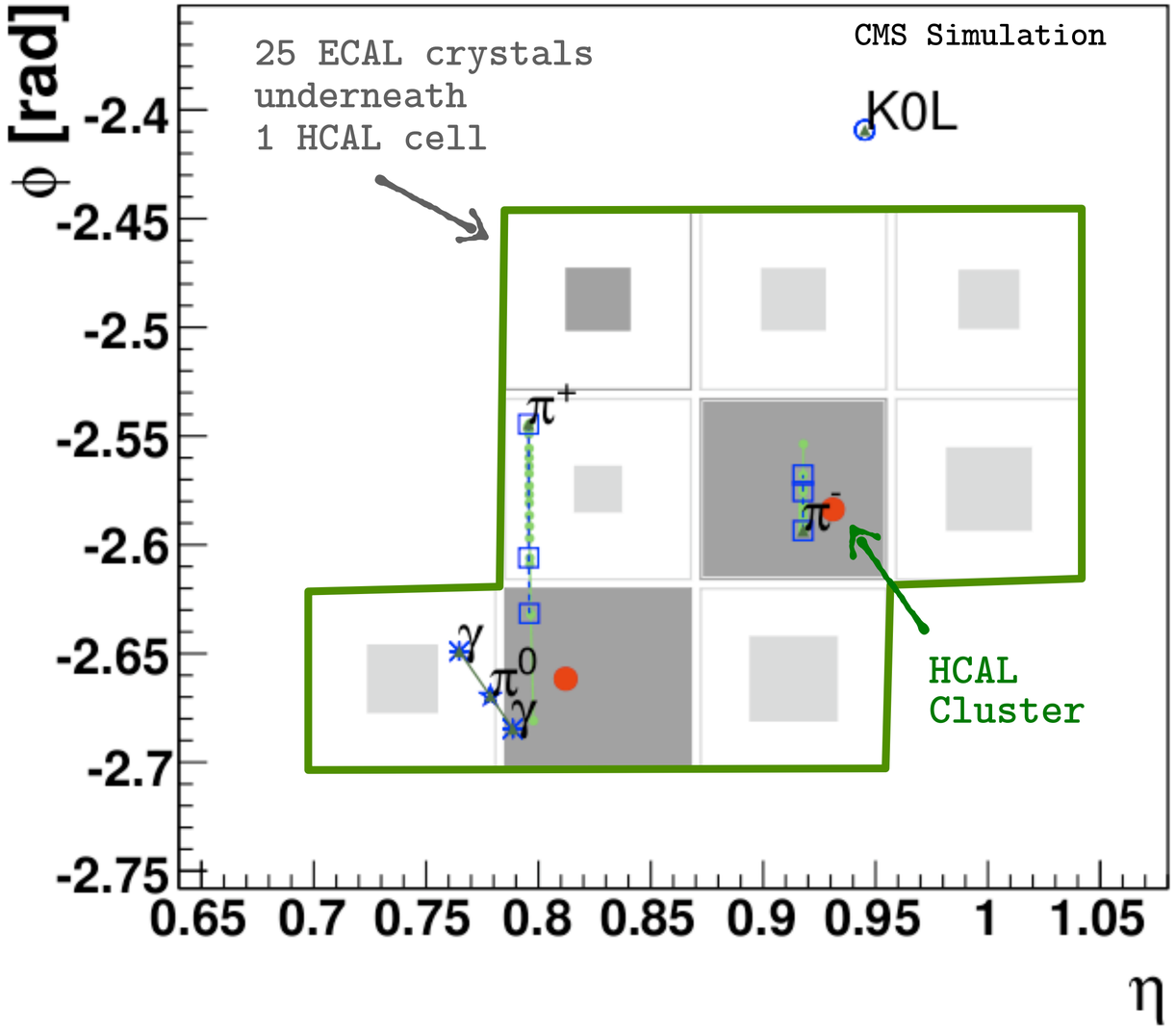




Four true particles:

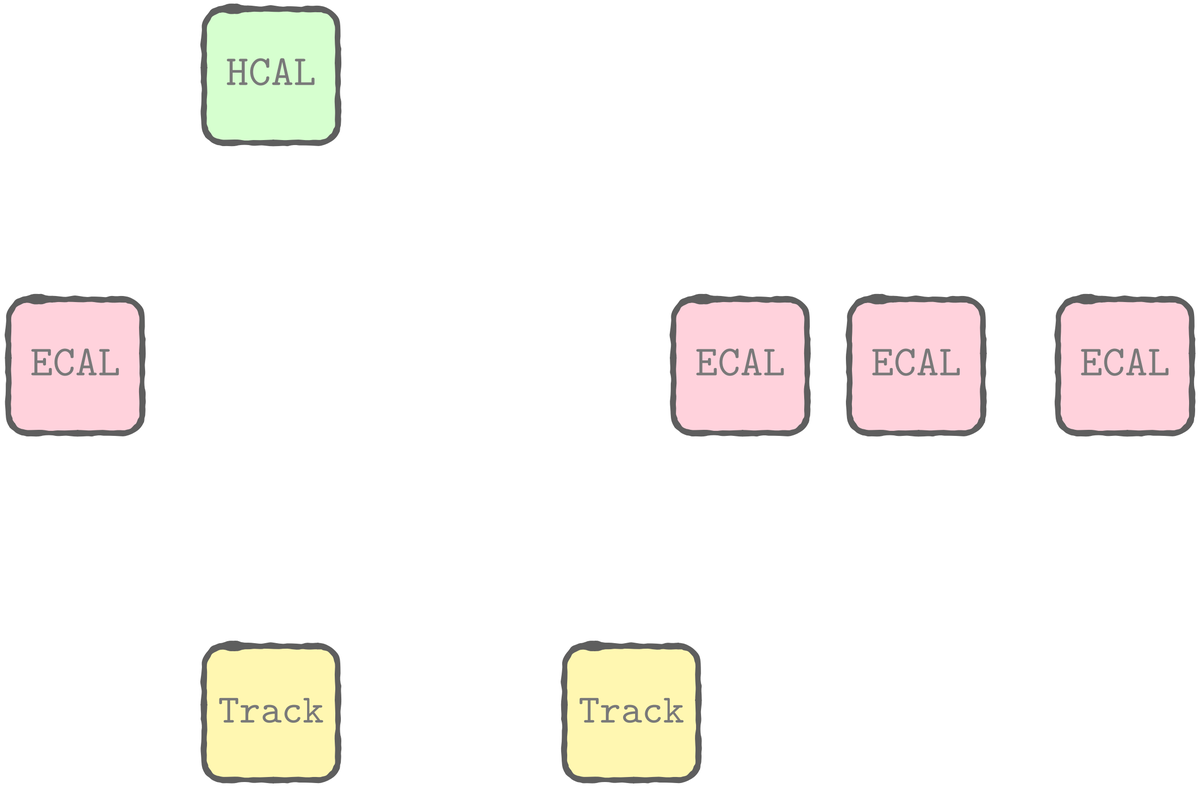
$\pi^+, \pi^-, \pi^0, K_L^0$

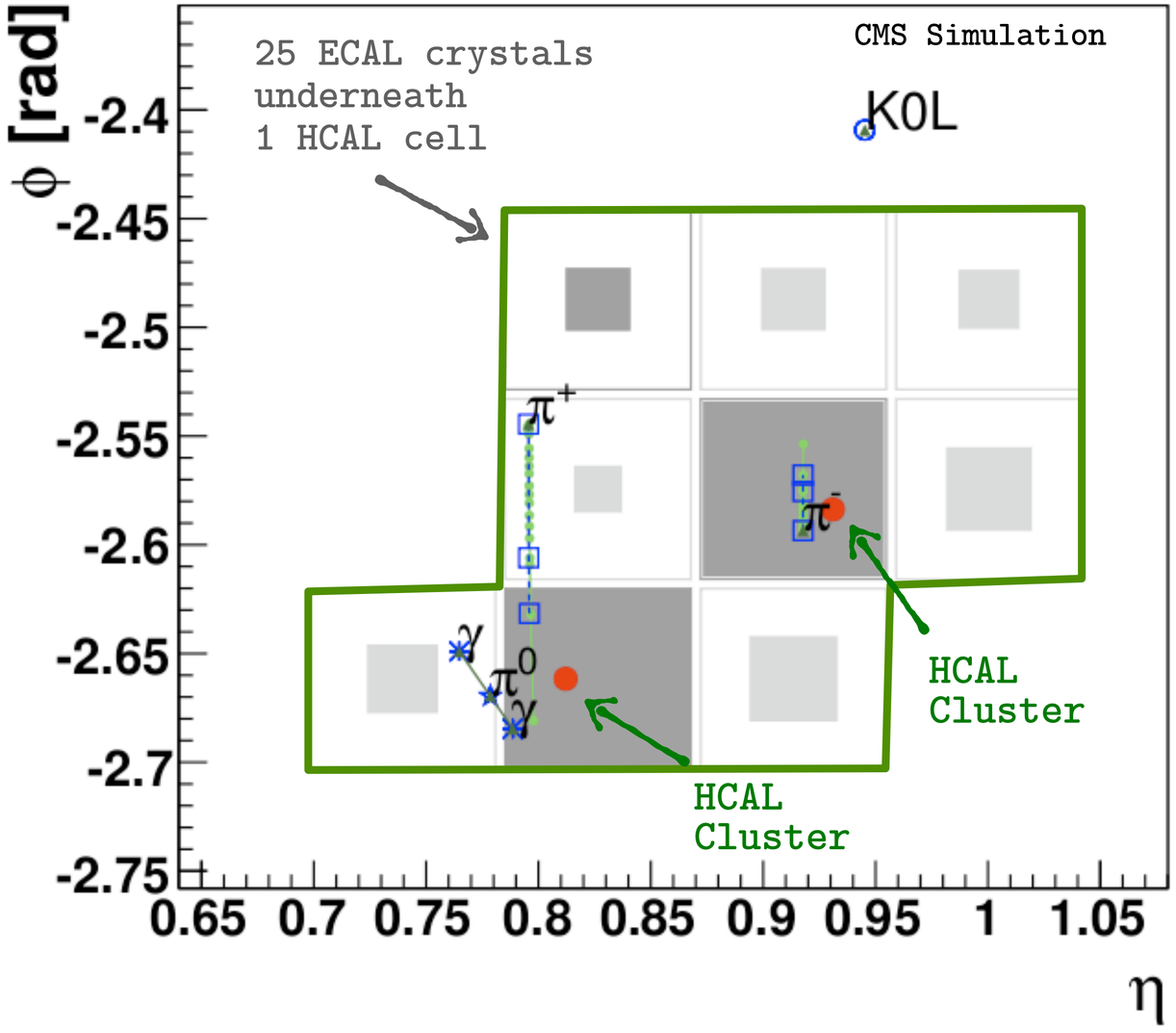




Four true particles:

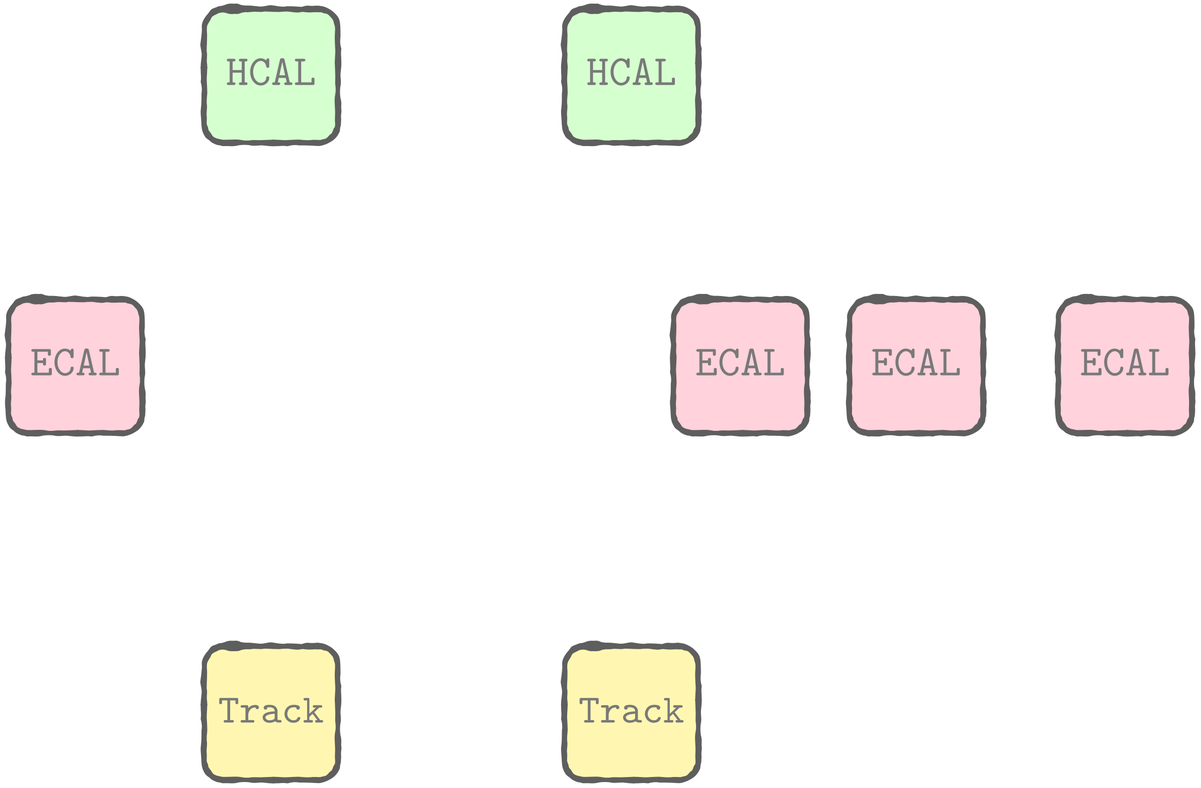
$\pi^+, \pi^-, \pi^0, K_L^0$

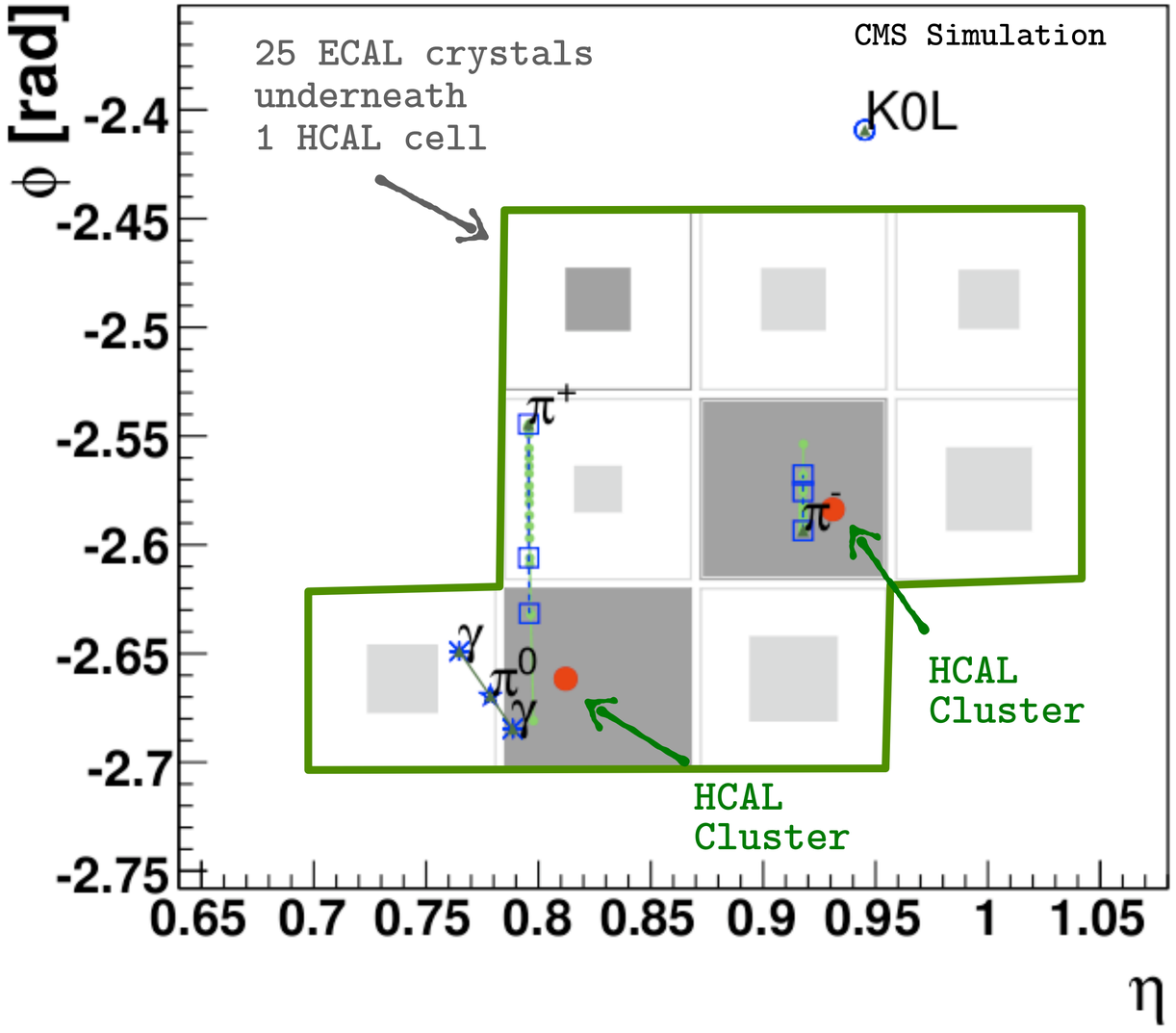




Four true particles:

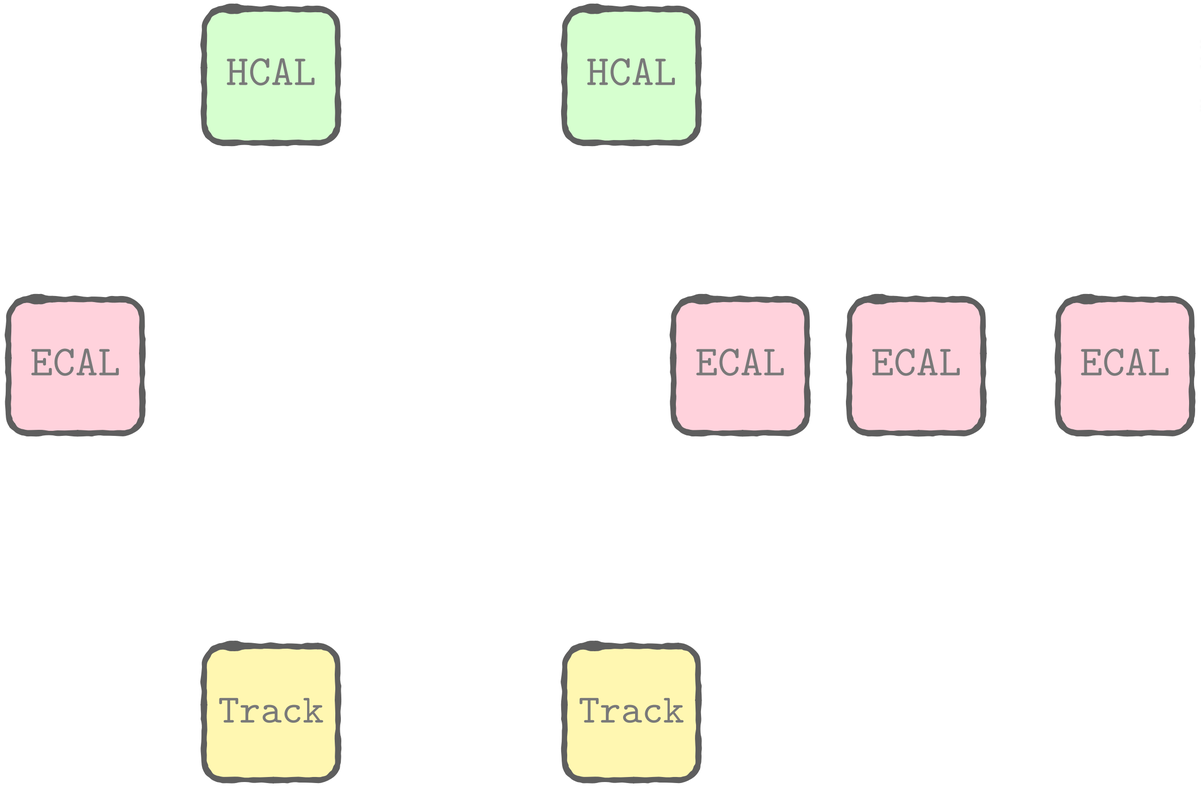
$\pi^+, \pi^-, \pi^0, K_L^0$

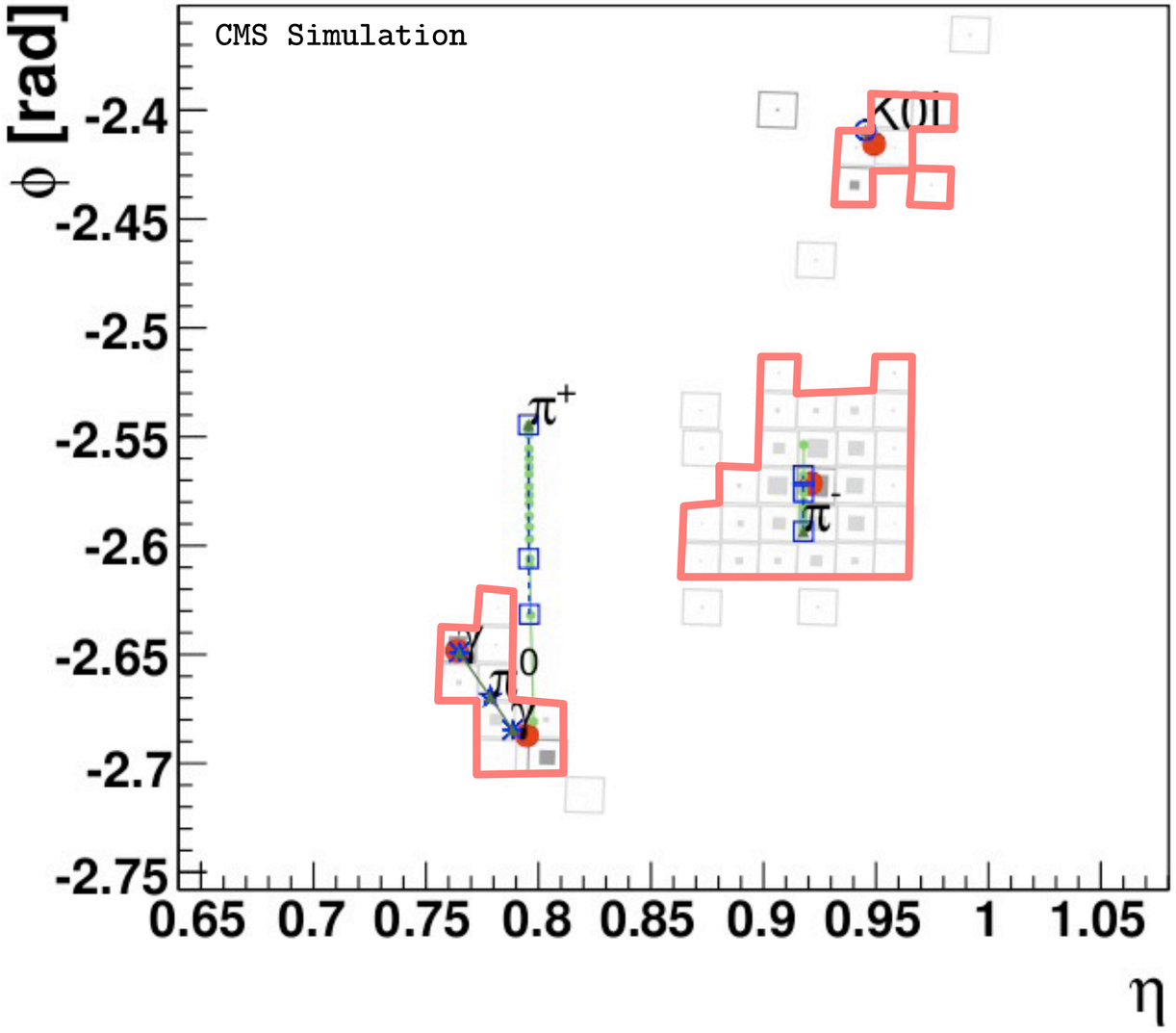




Four true particles:

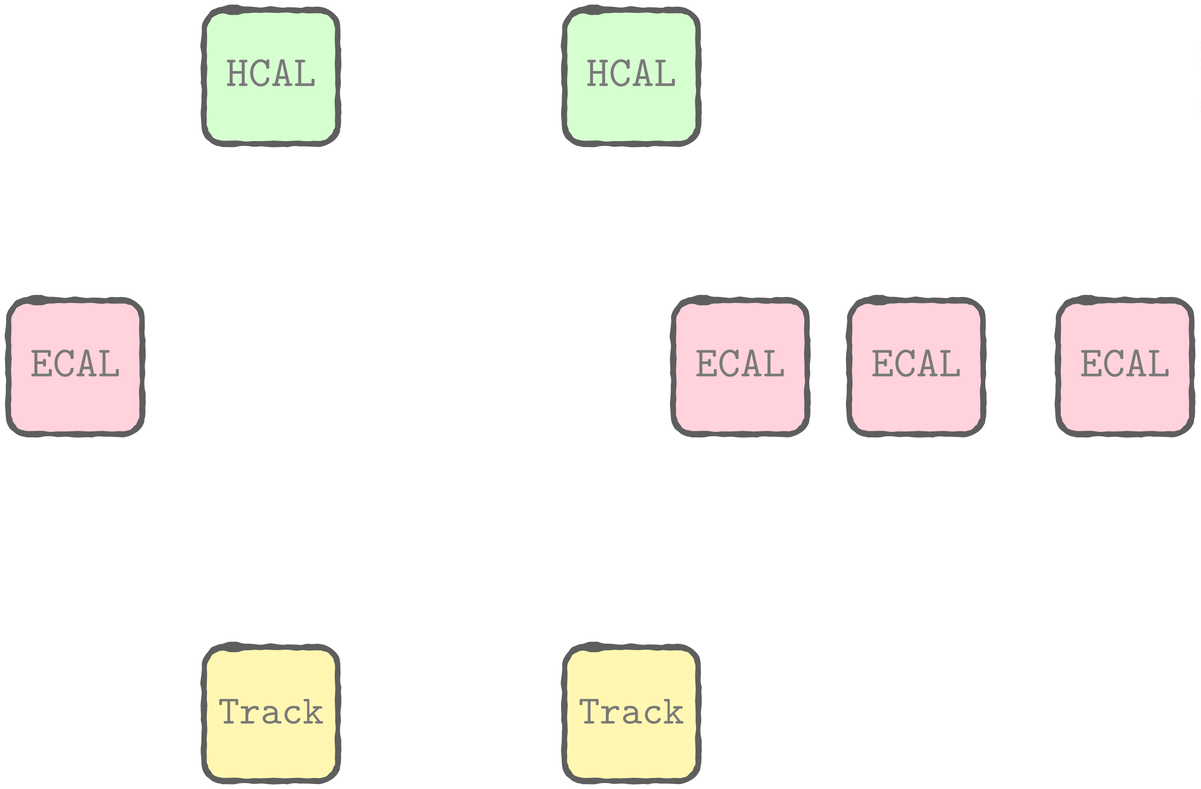
$\pi^+, \pi^-, \pi^0, K_L^0$

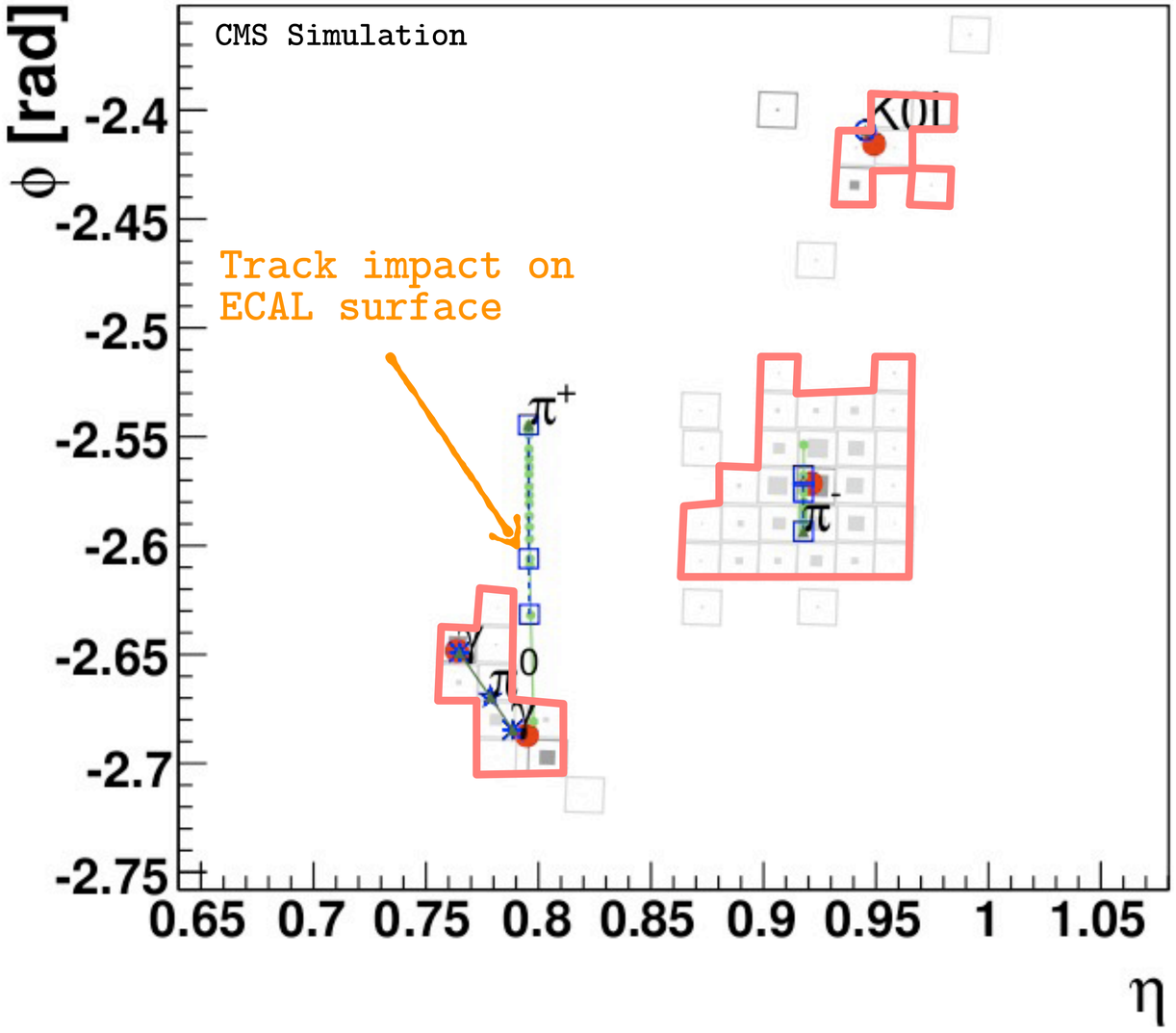




Four true particles:

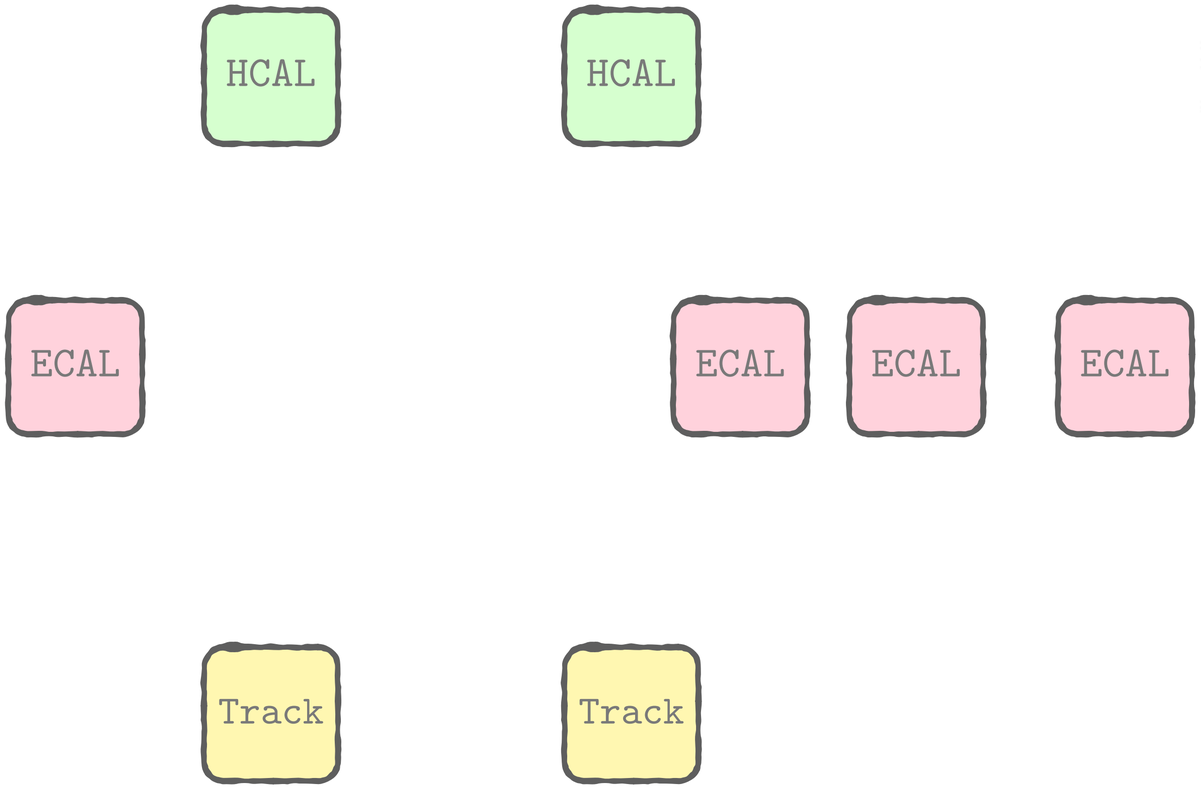
$\pi^+, \pi^-, \pi^0, K_L^0$

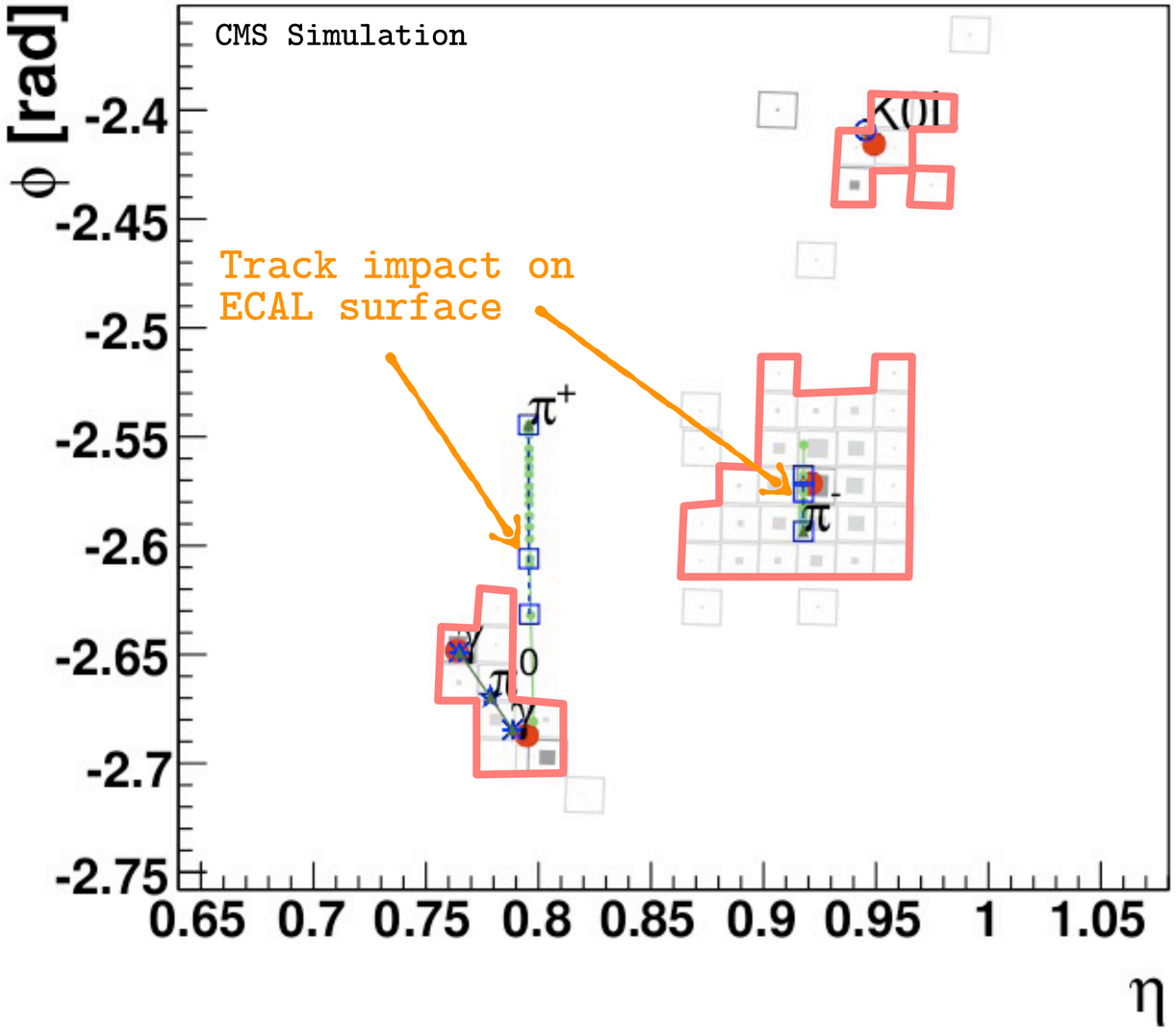




Four true particles:

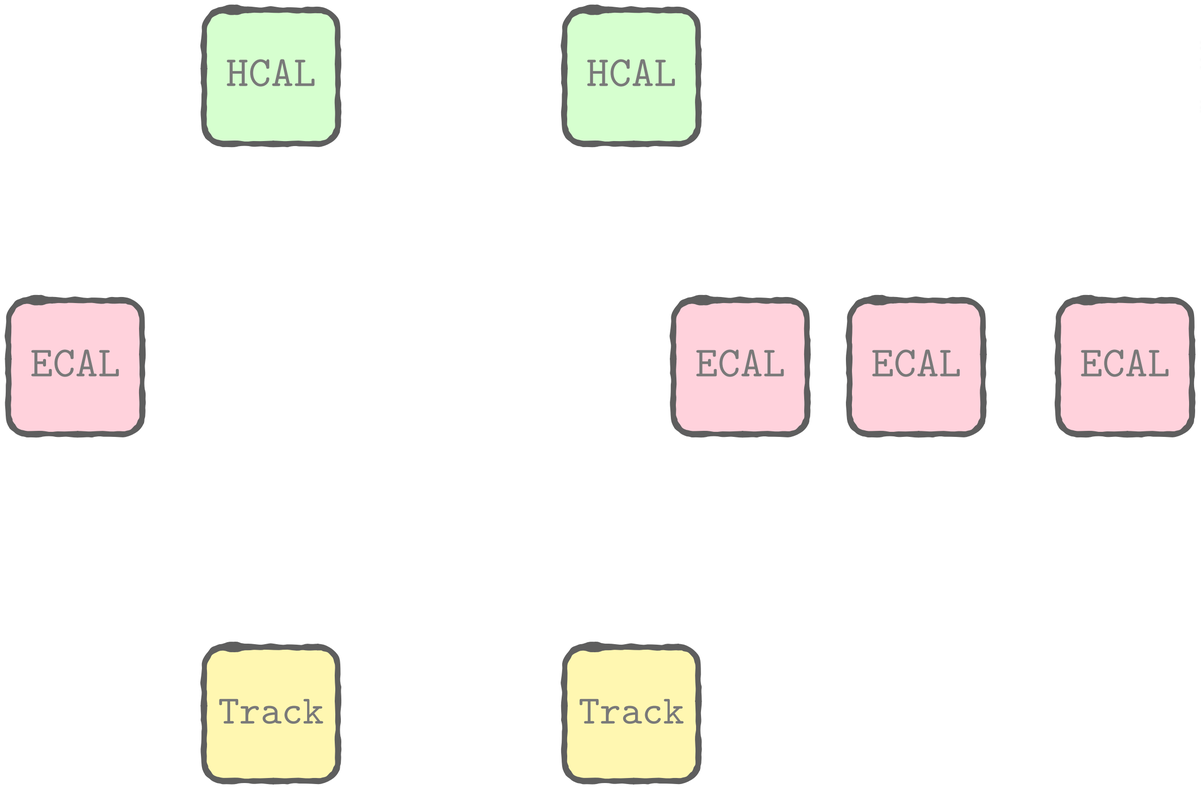
$\pi^+, \pi^-, \pi^0, K_L^0$

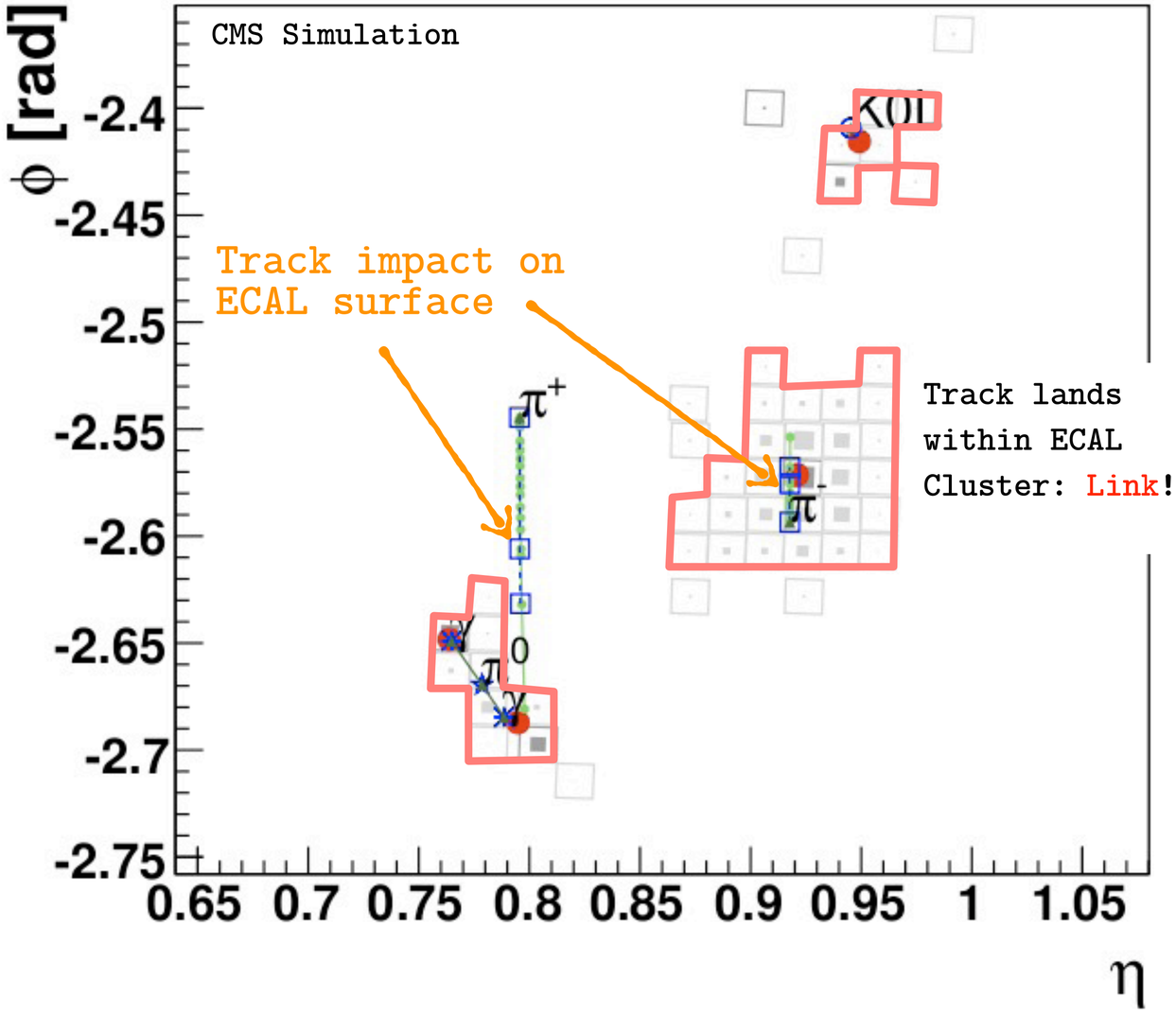




Four true particles:

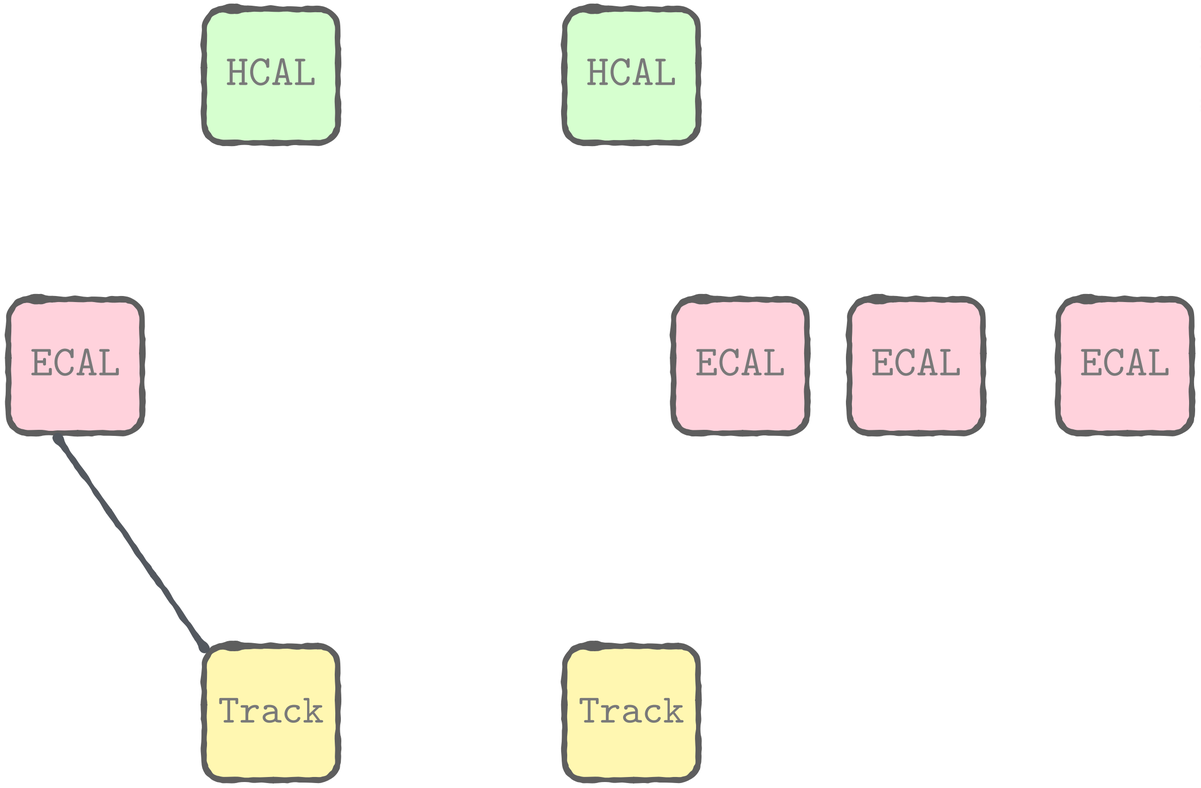
$\pi^+, \pi^-, \pi^0, K_L^0$

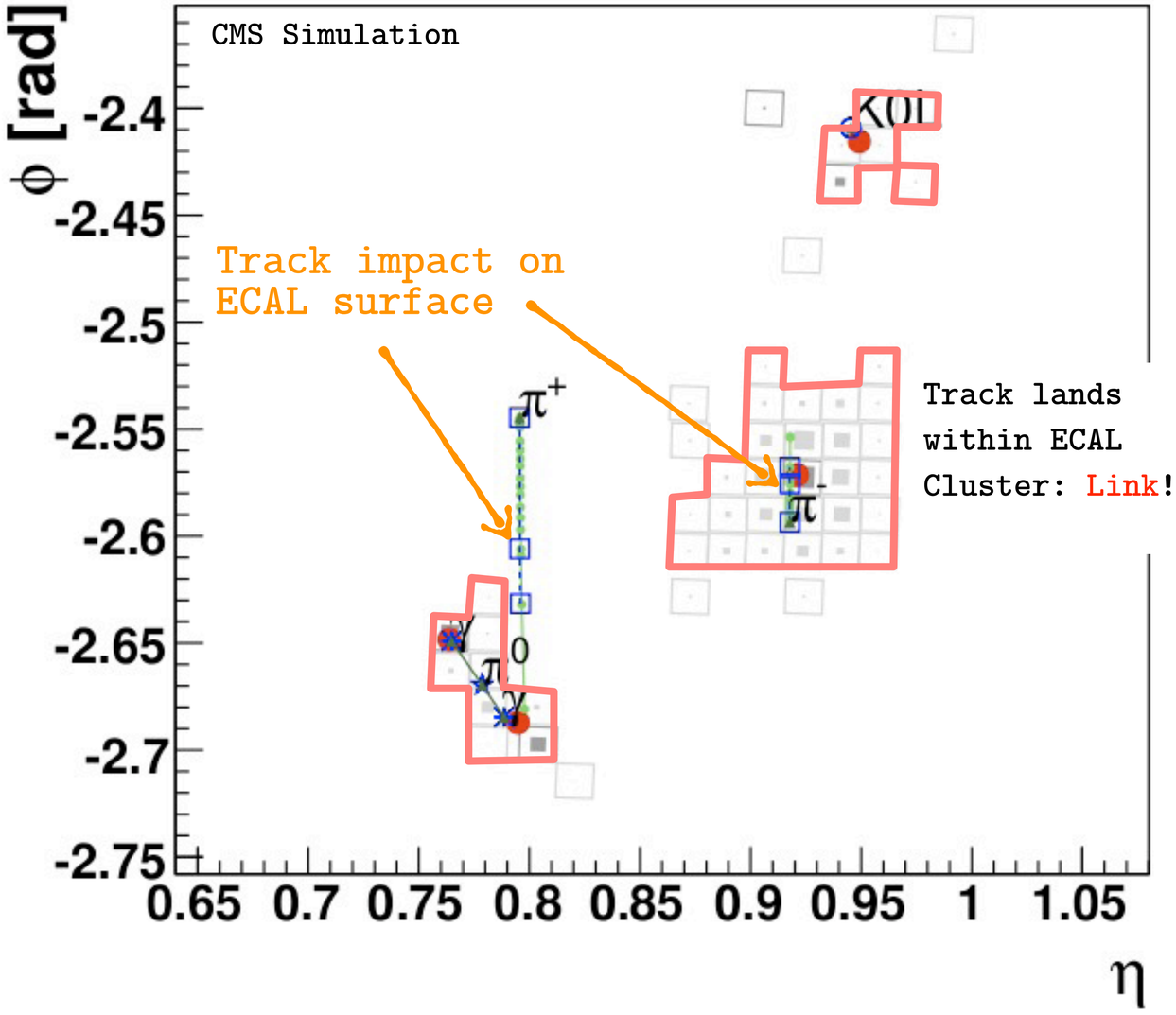




Four true particles:

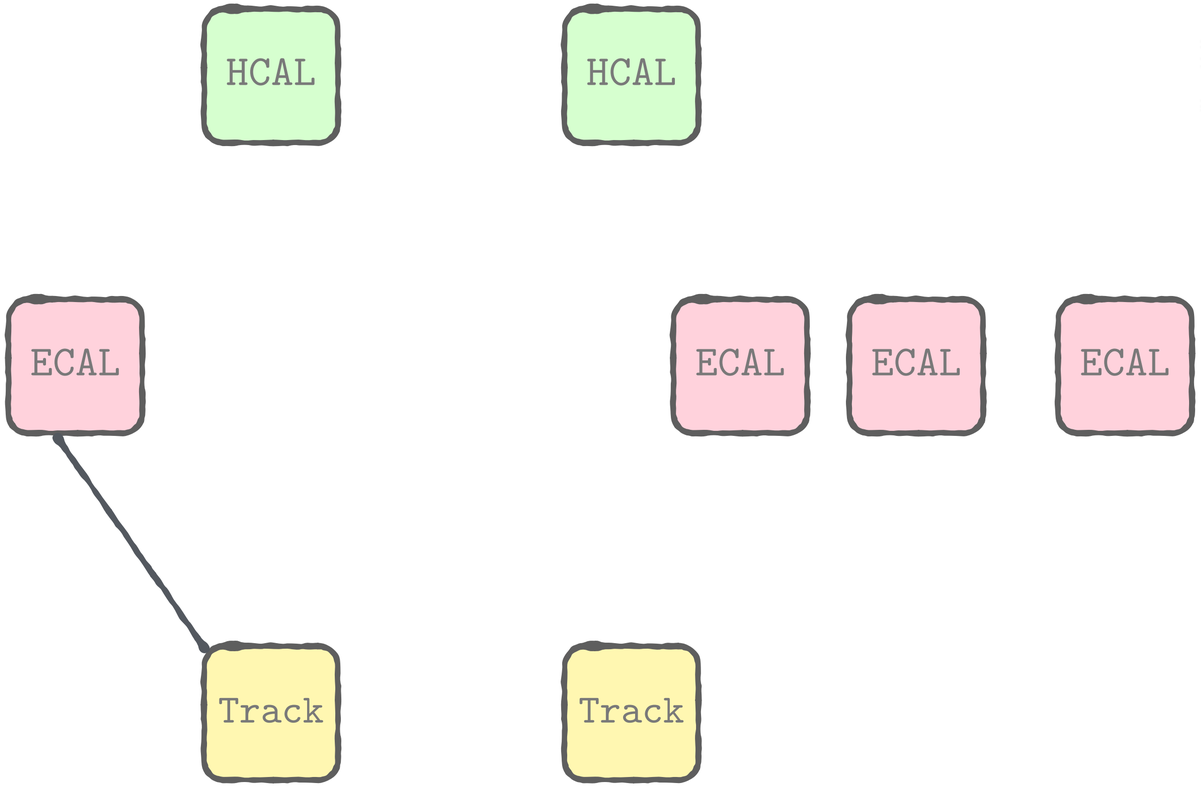
$\pi^+, \pi^-, \pi^0, K_L^0$

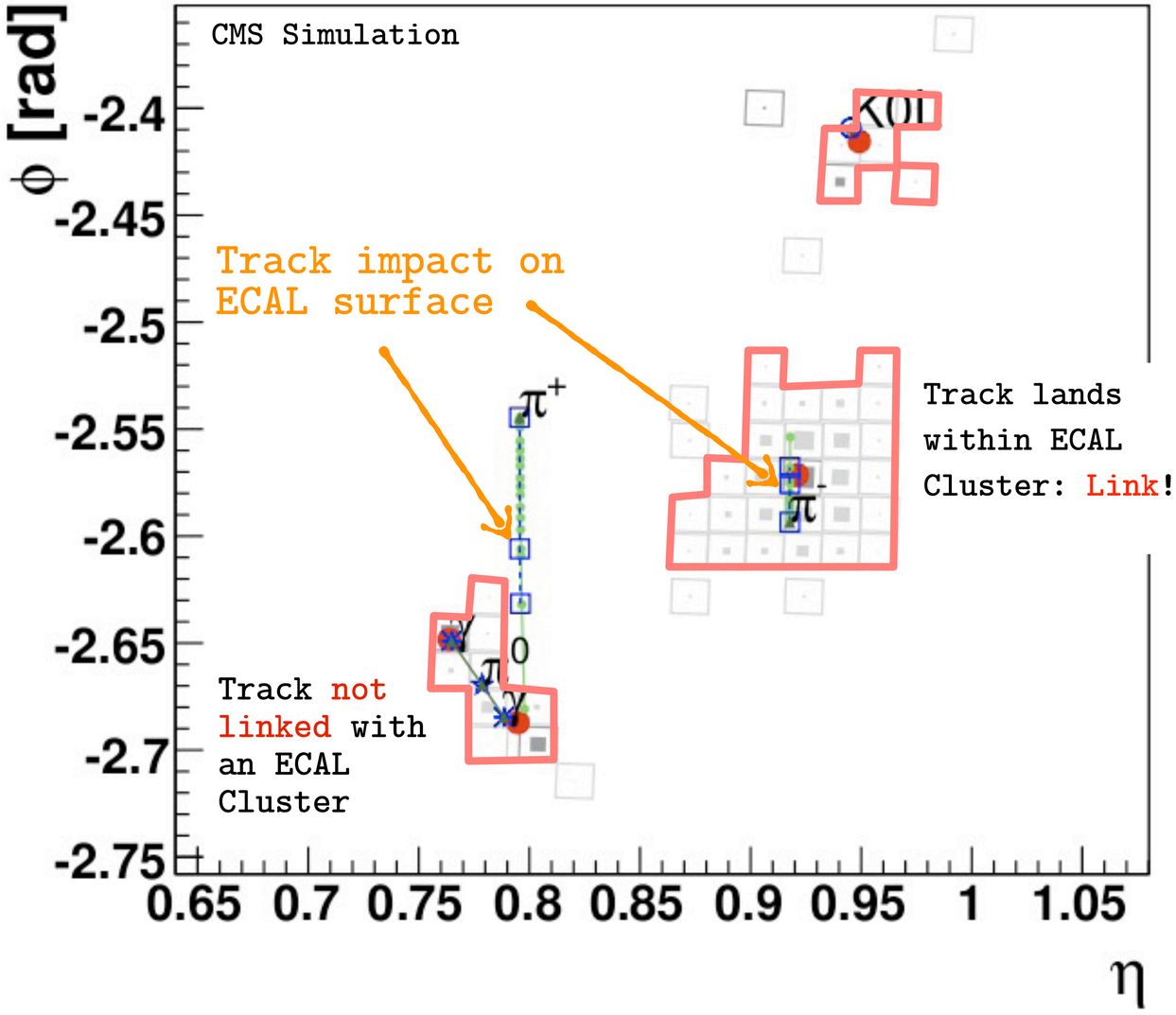




Four true particles:

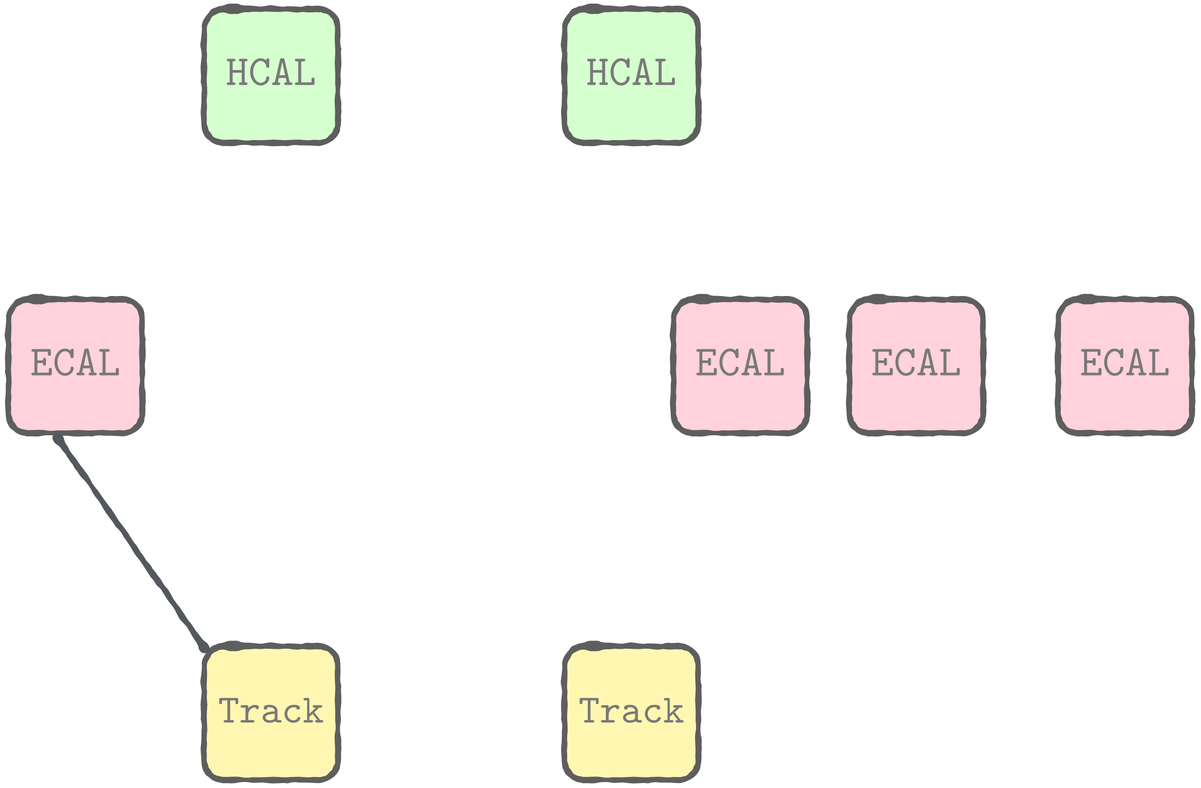
$\pi^+, \pi^-, \pi^0, K_L^0$

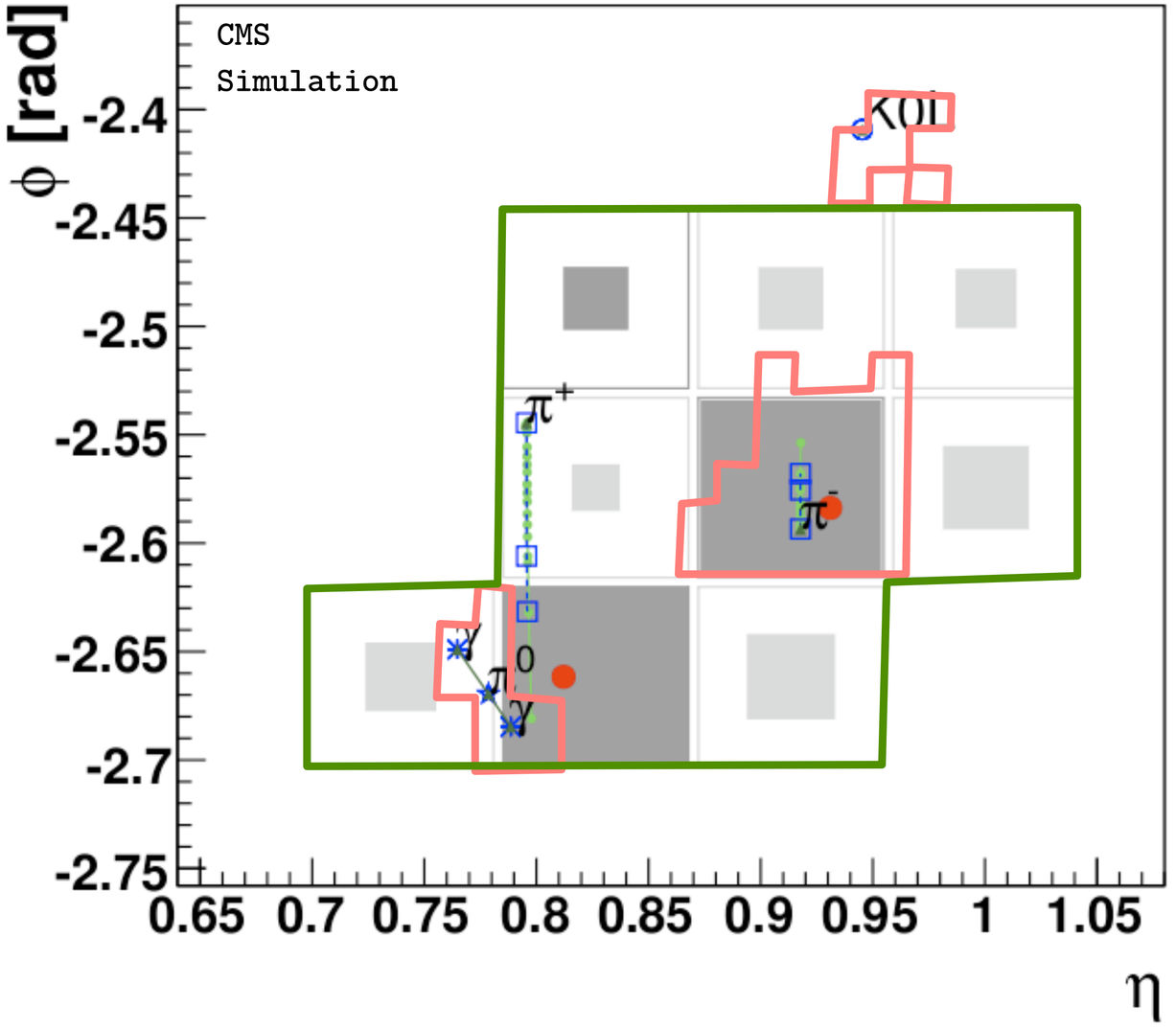




Four true particles:

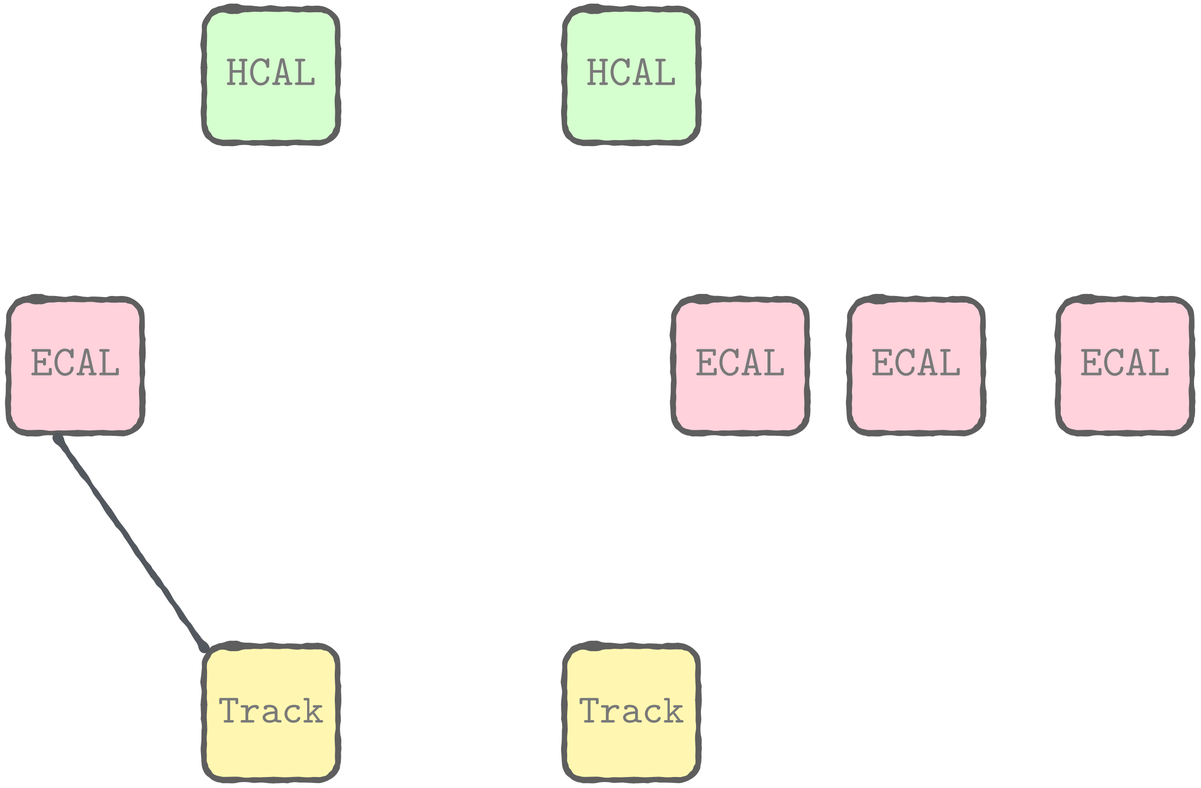
$\pi^+, \pi^-, \pi^0, K_L^0$

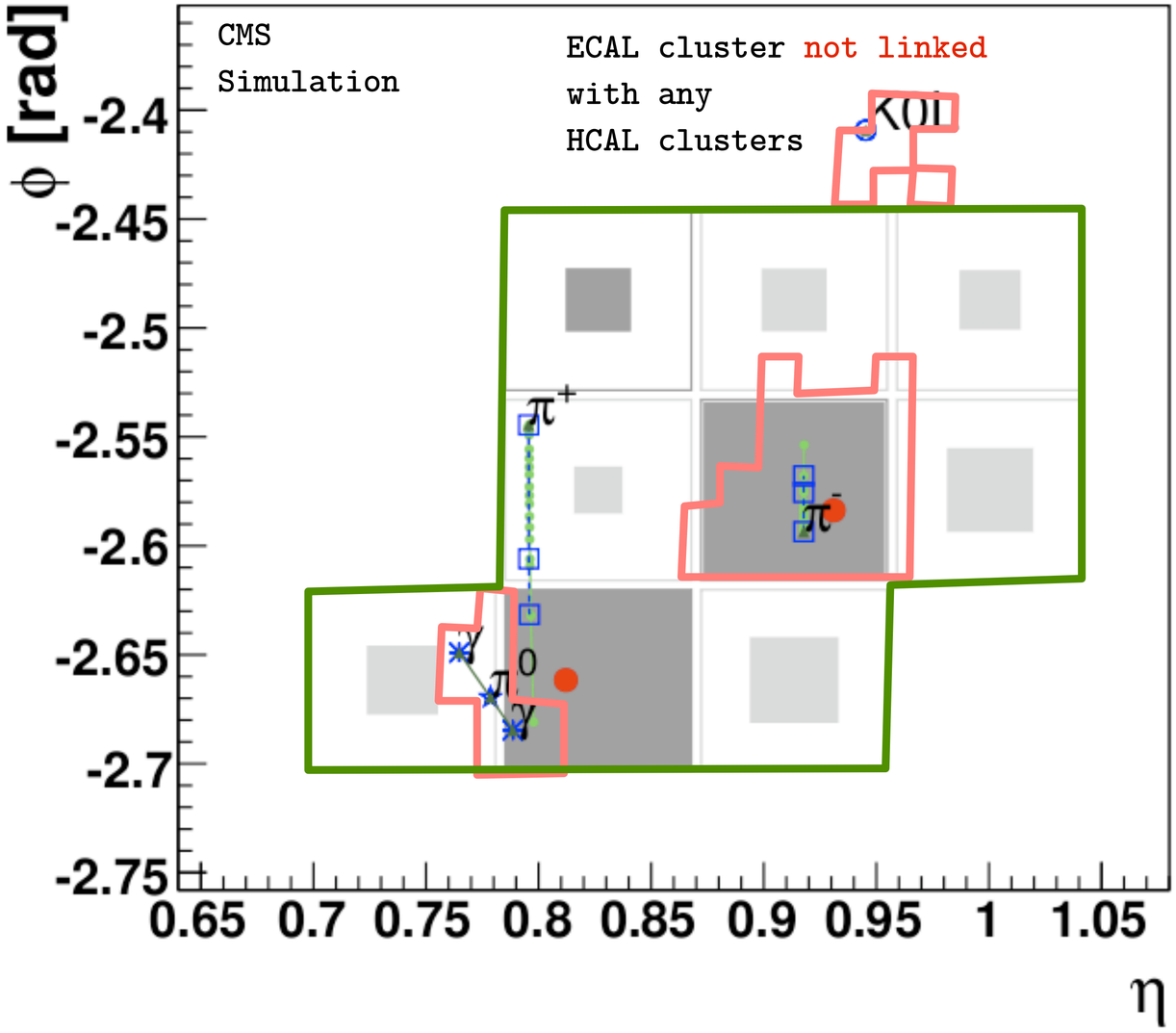




Four true particles:

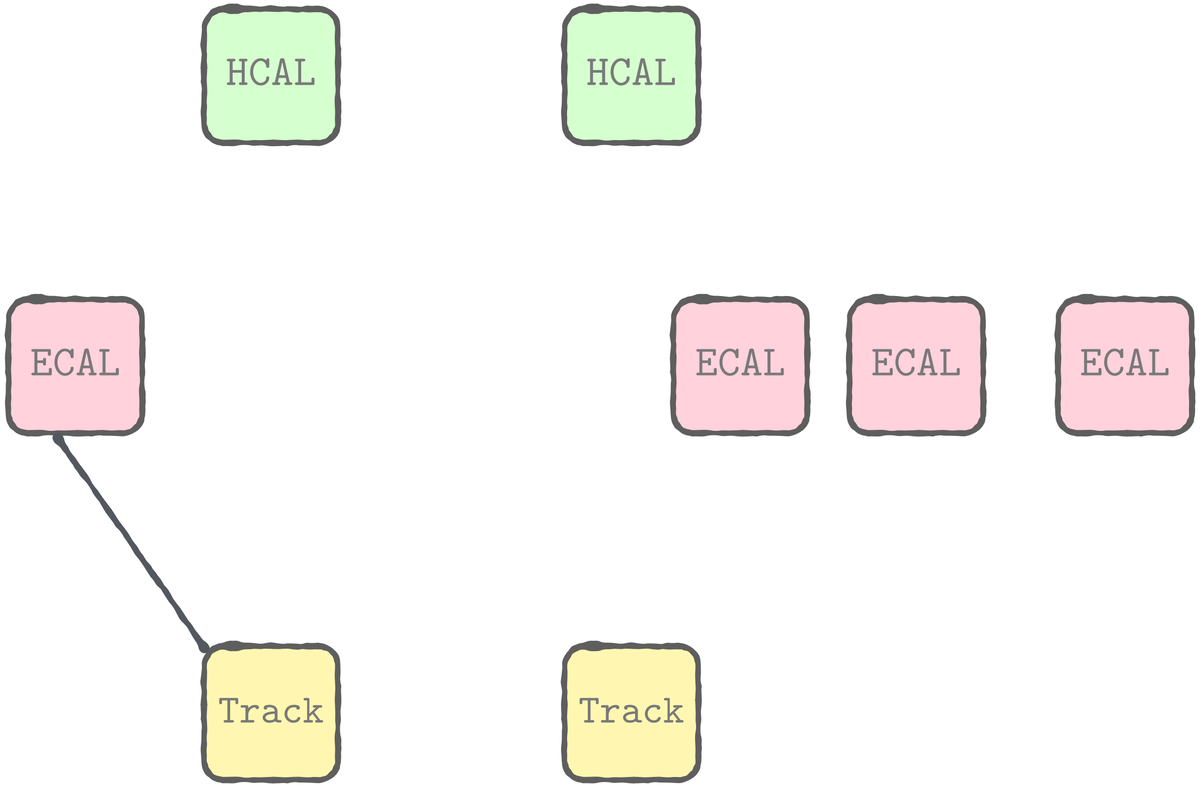
$$\pi^+, \pi^-, \pi^0, K_L^0$$

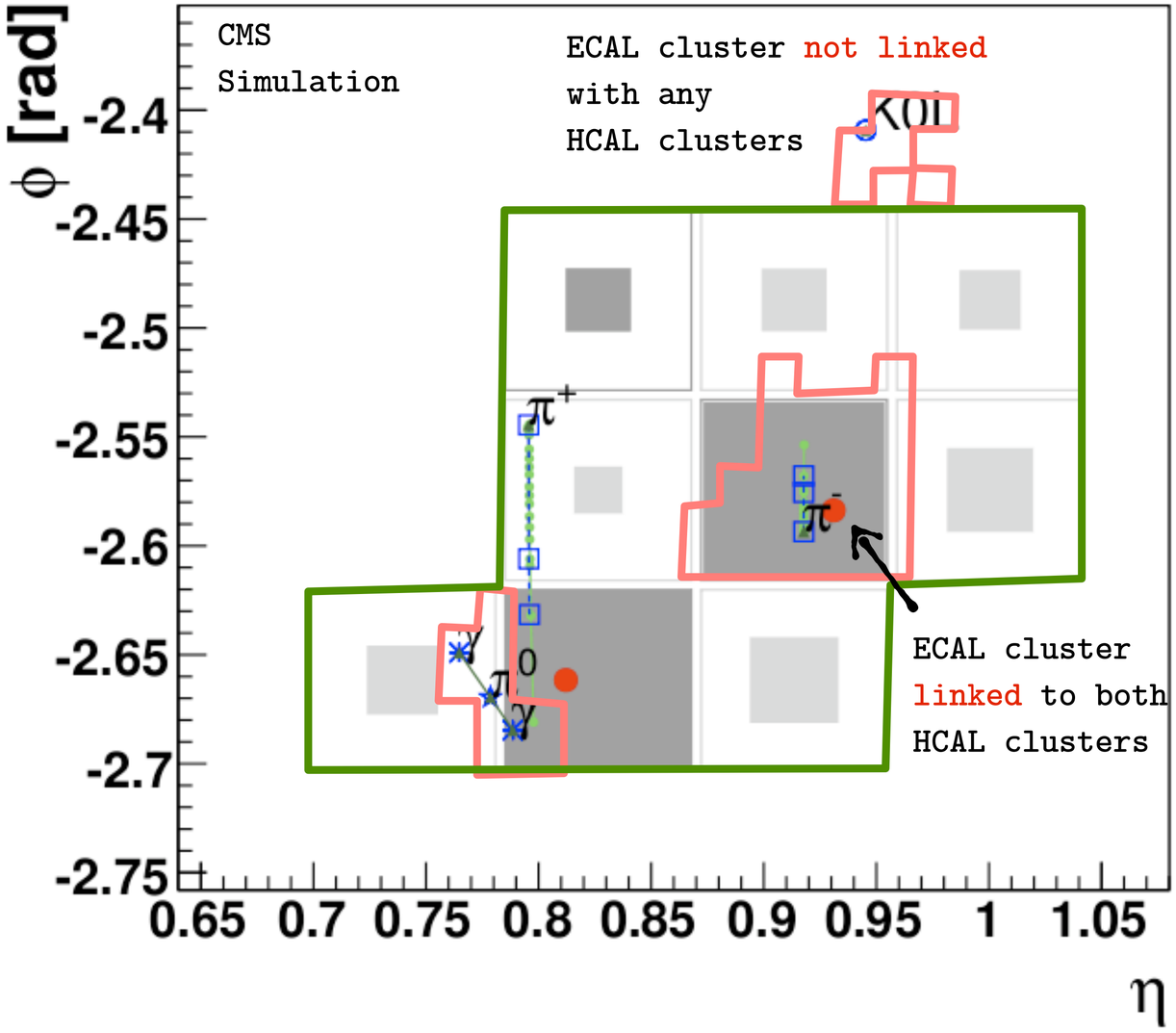




Four true particles:

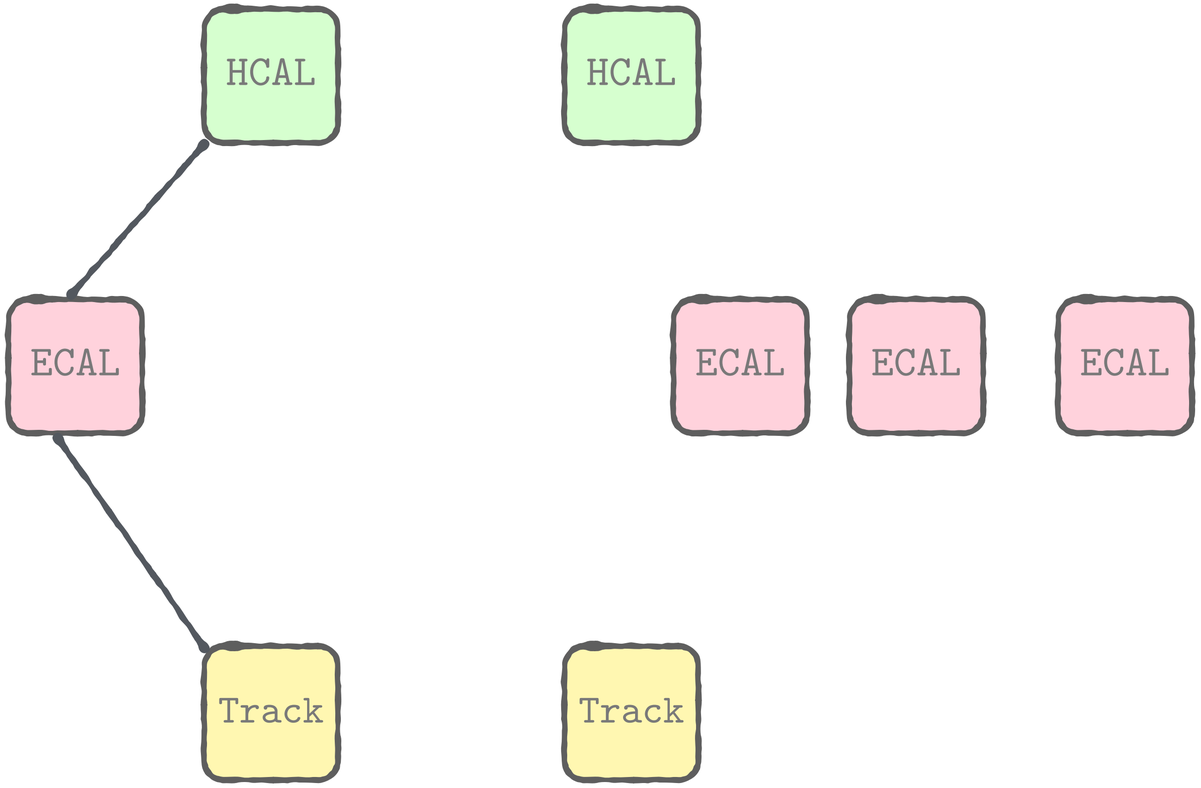
$$\pi^+, \pi^-, \pi^0, K_L^0$$

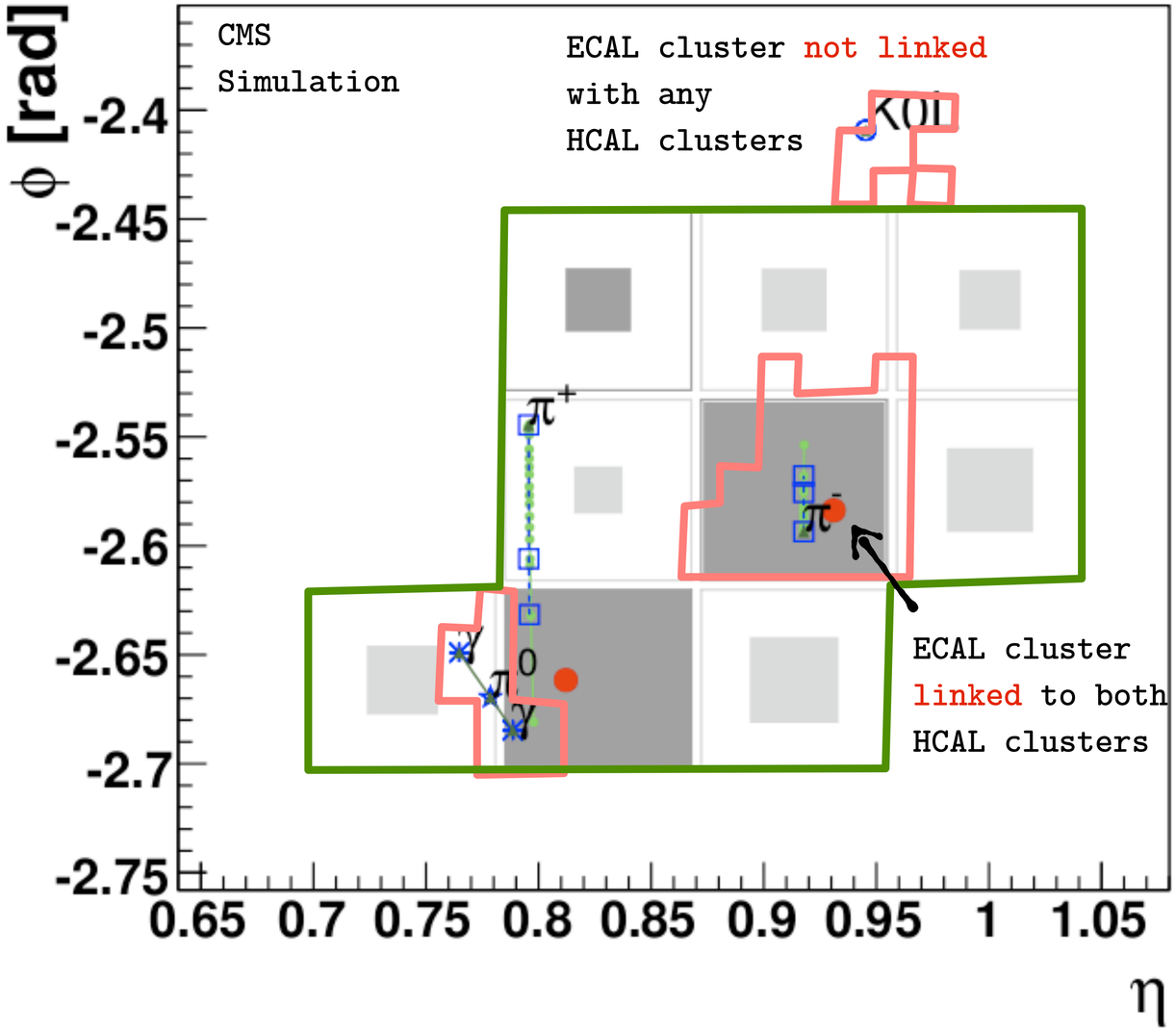




Four true particles:

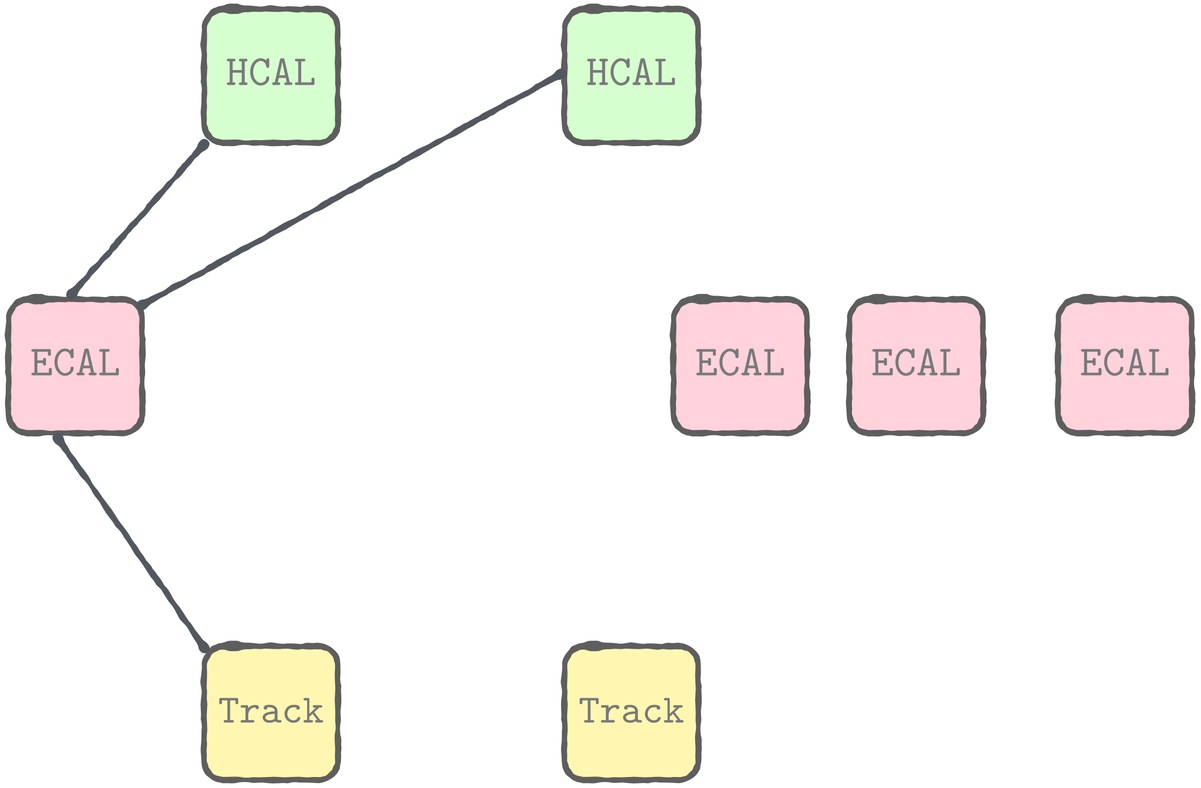
$$\pi^+, \pi^-, \pi^0, K_L^0$$

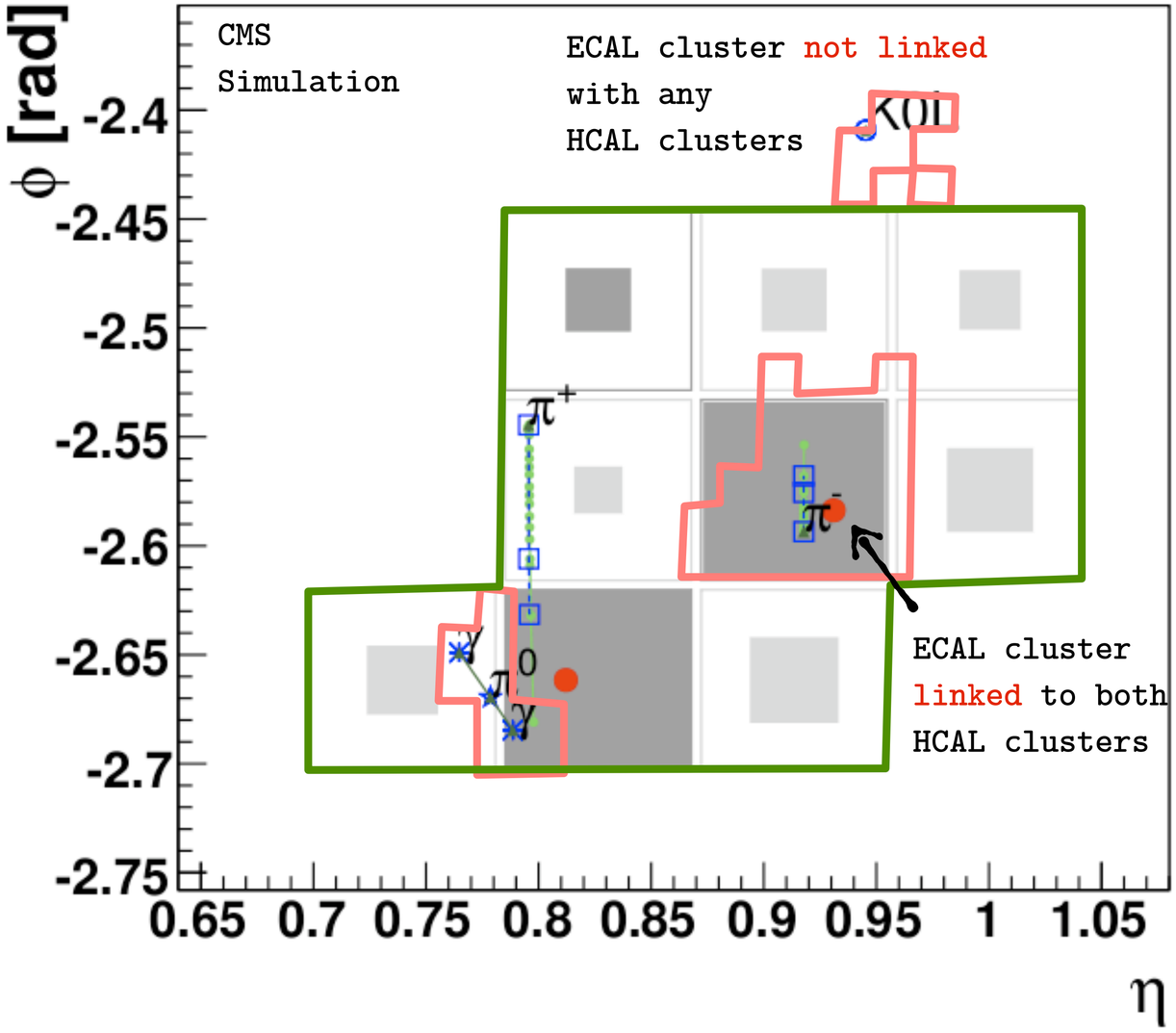




Four true particles:

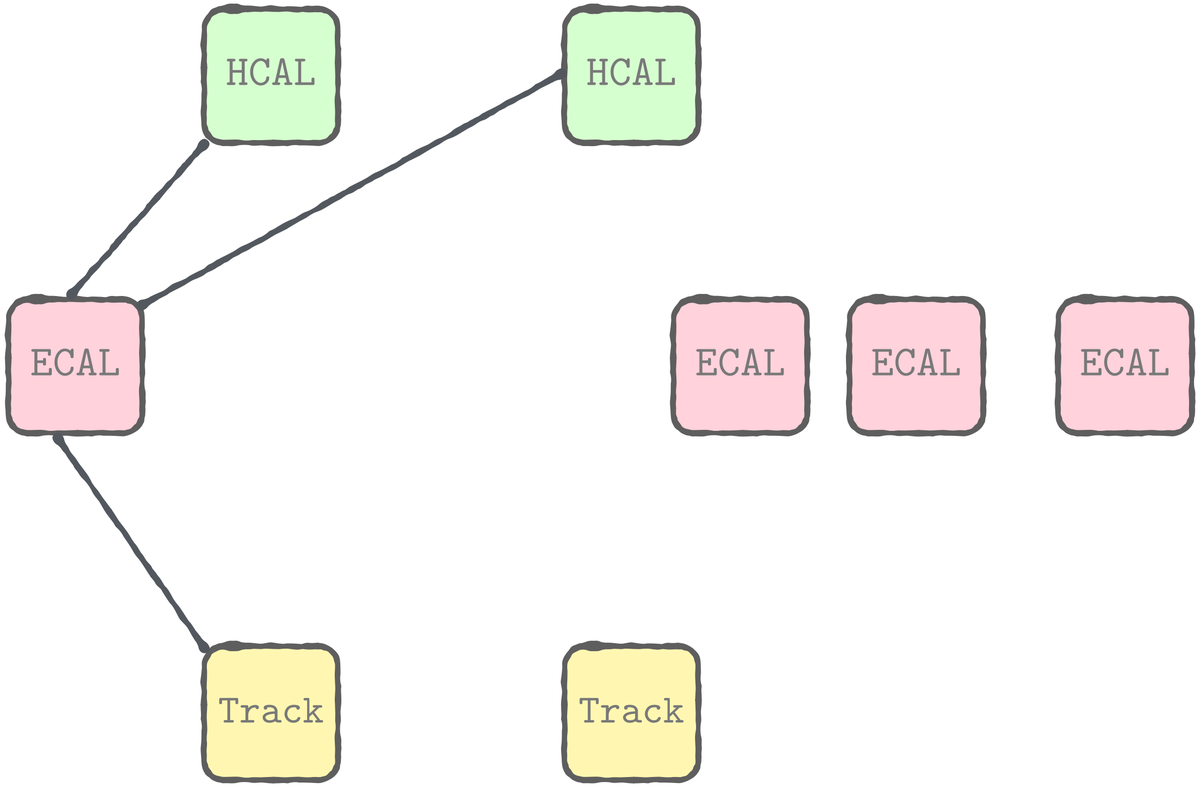
$$\pi^+, \pi^-, \pi^0, K_L^0$$

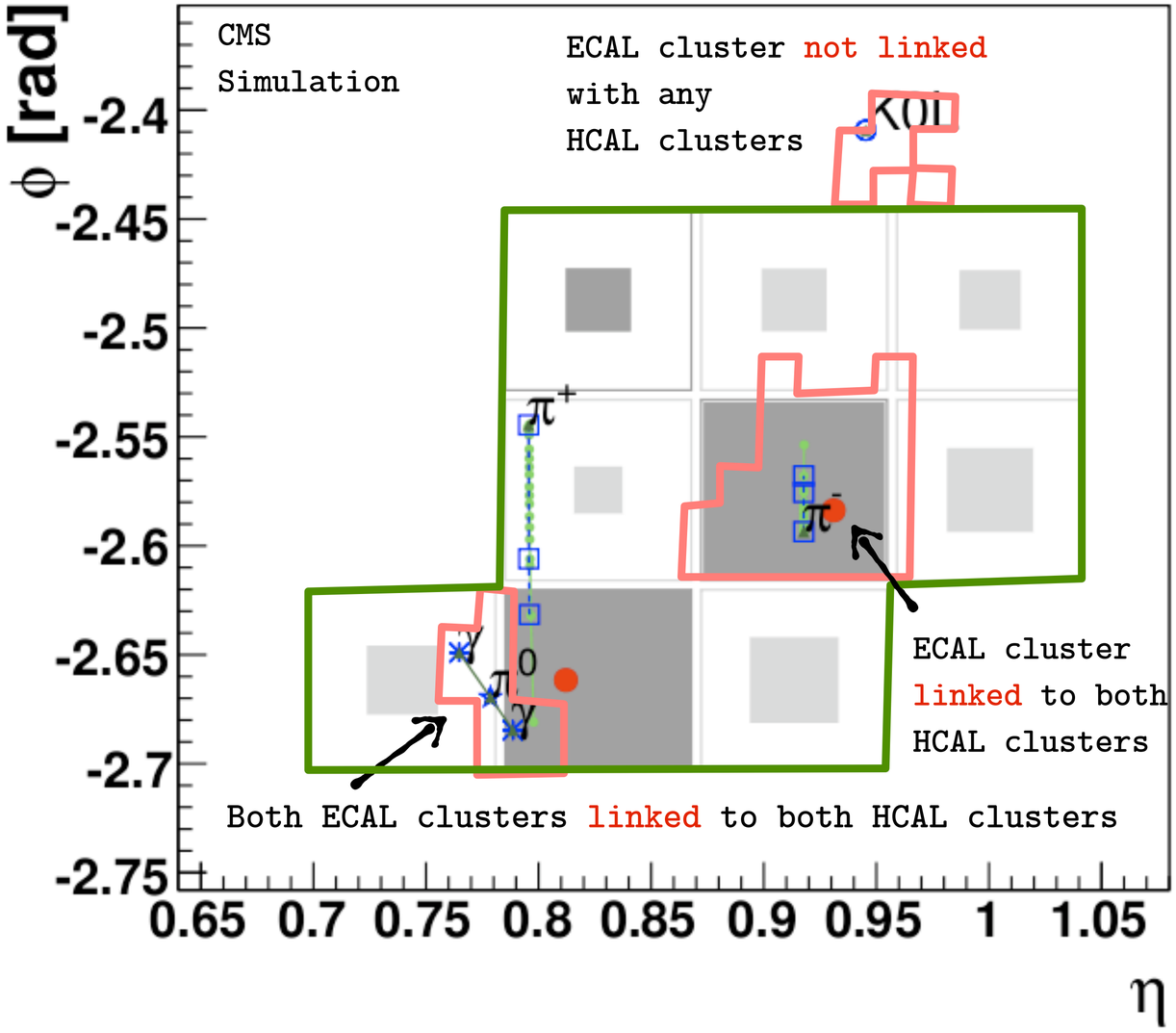




Four true particles:

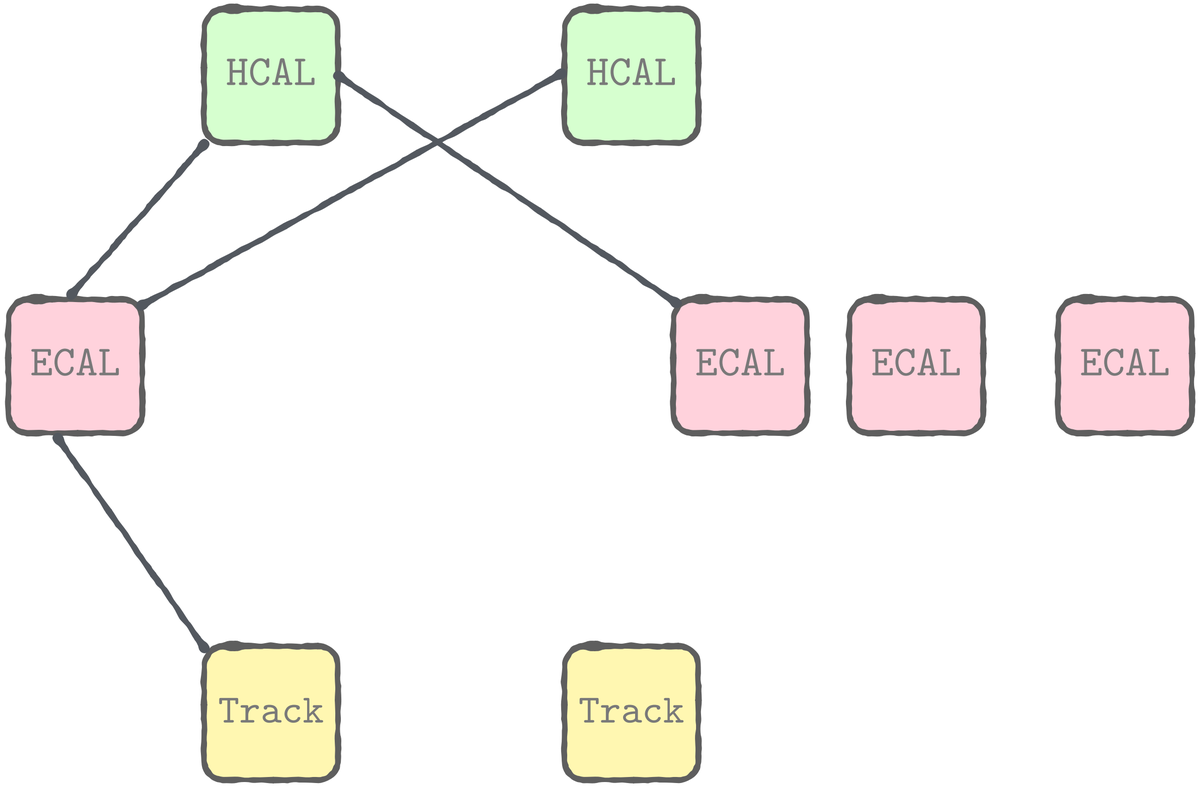
$\pi^+, \pi^-, \pi^0, K_L^0$

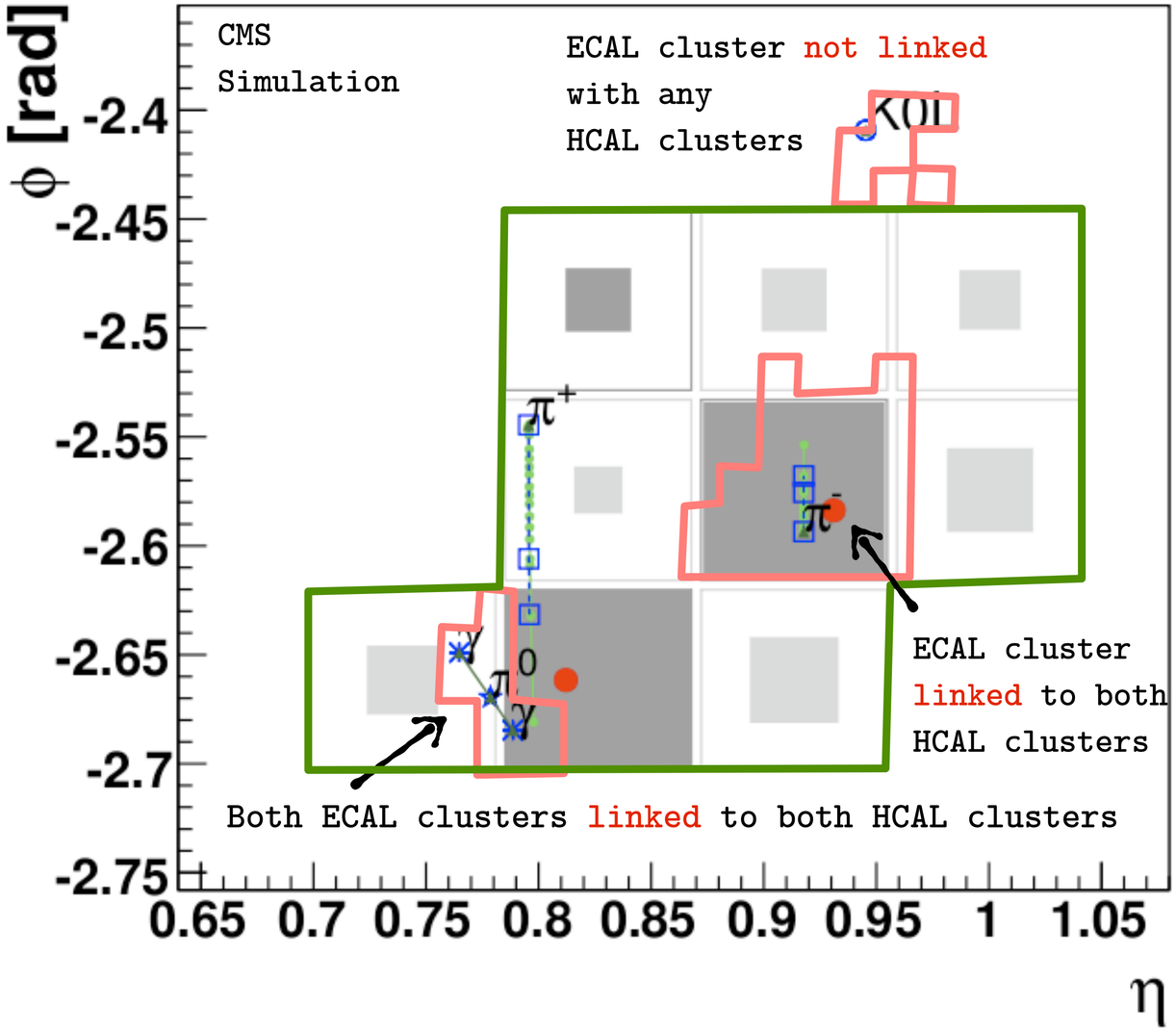




Four true particles:

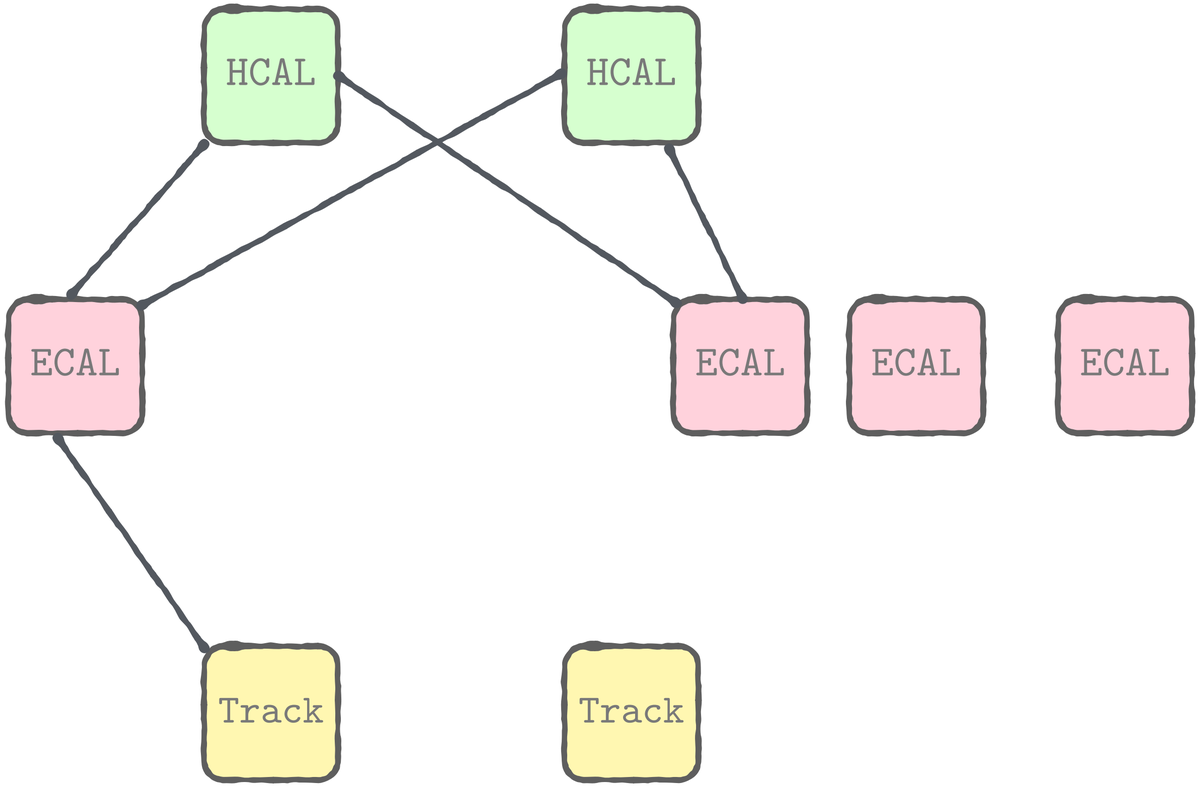
$$\pi^+, \pi^-, \pi^0, K_L^0$$

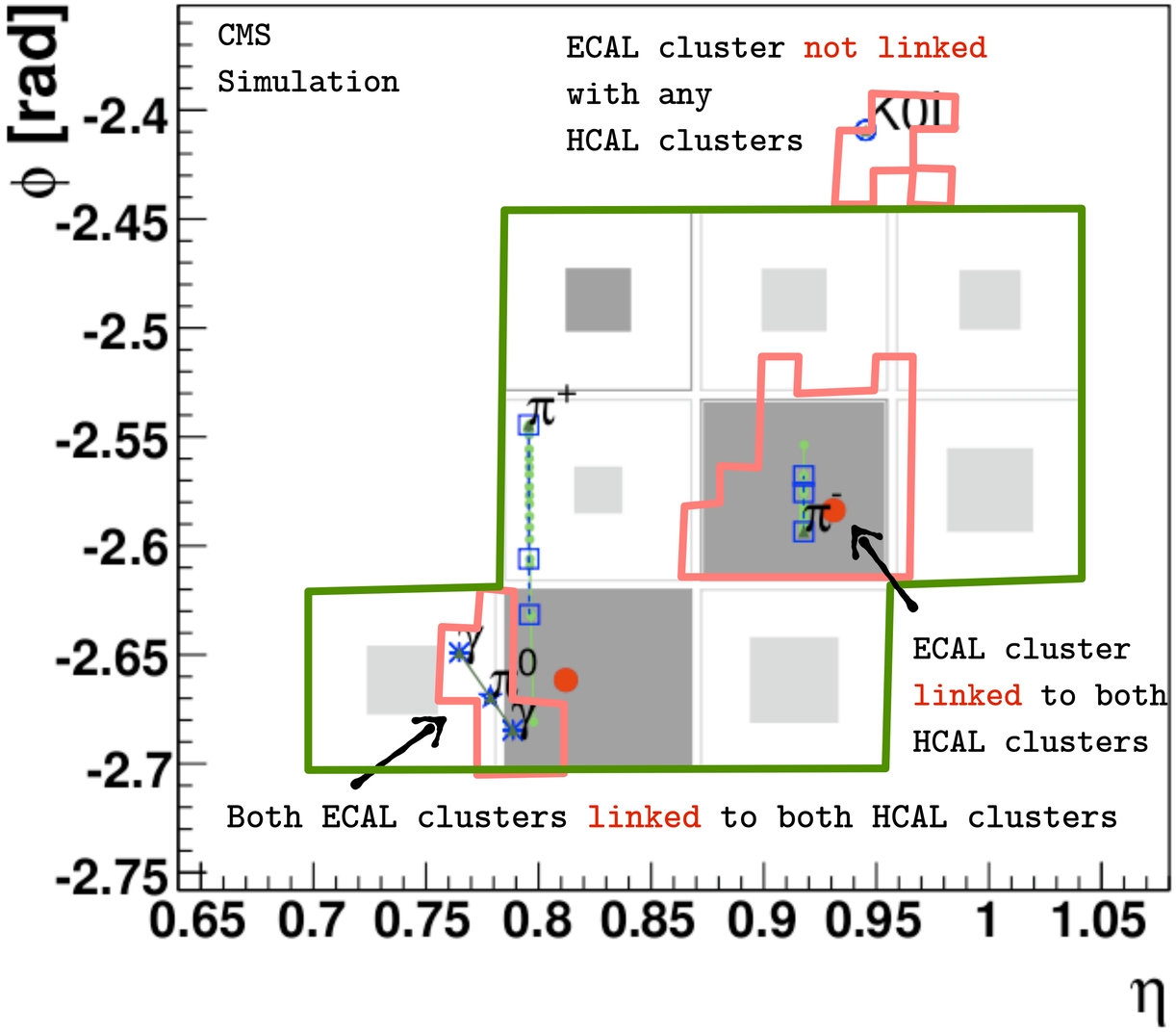




Four true particles:

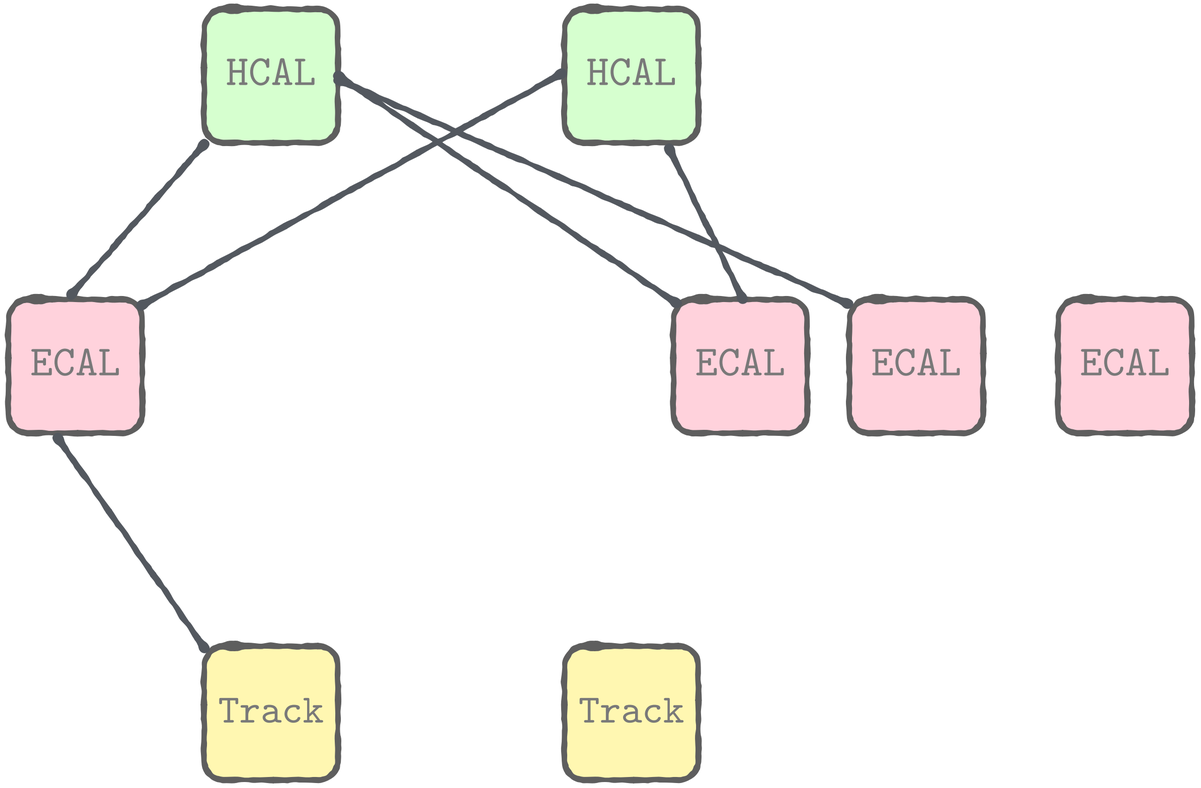
$$\pi^+, \pi^-, \pi^0, K_L^0$$

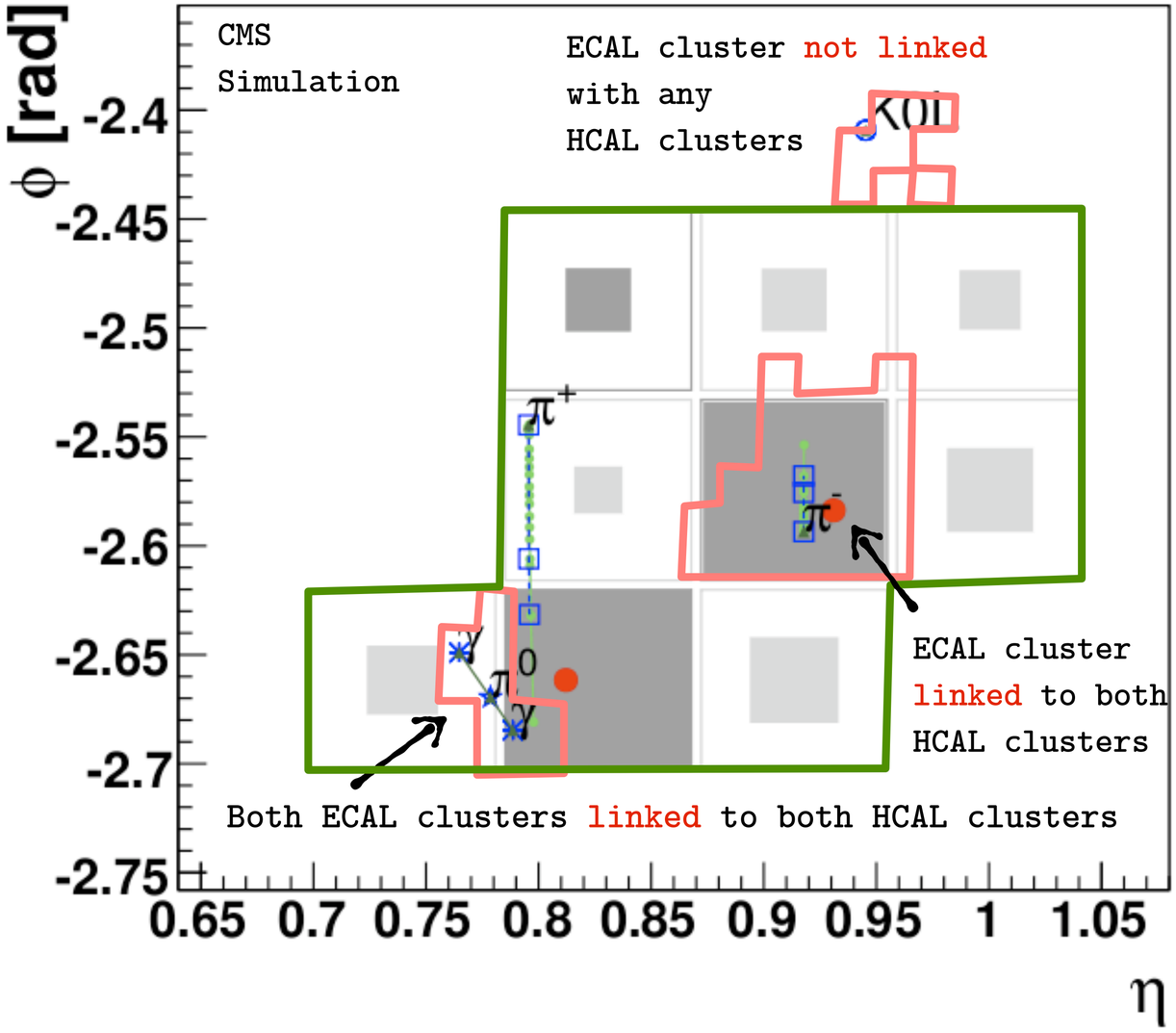




Four true particles:

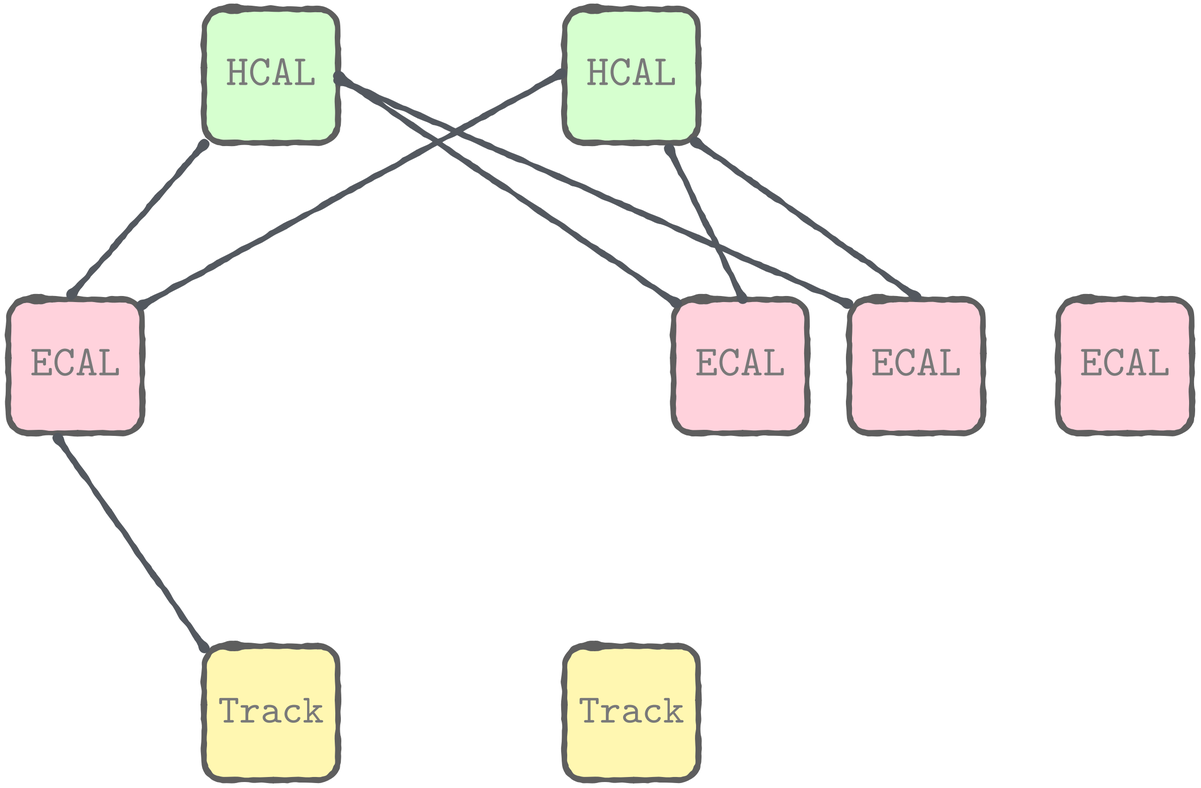
$$\pi^+, \pi^-, \pi^0, K_L^0$$

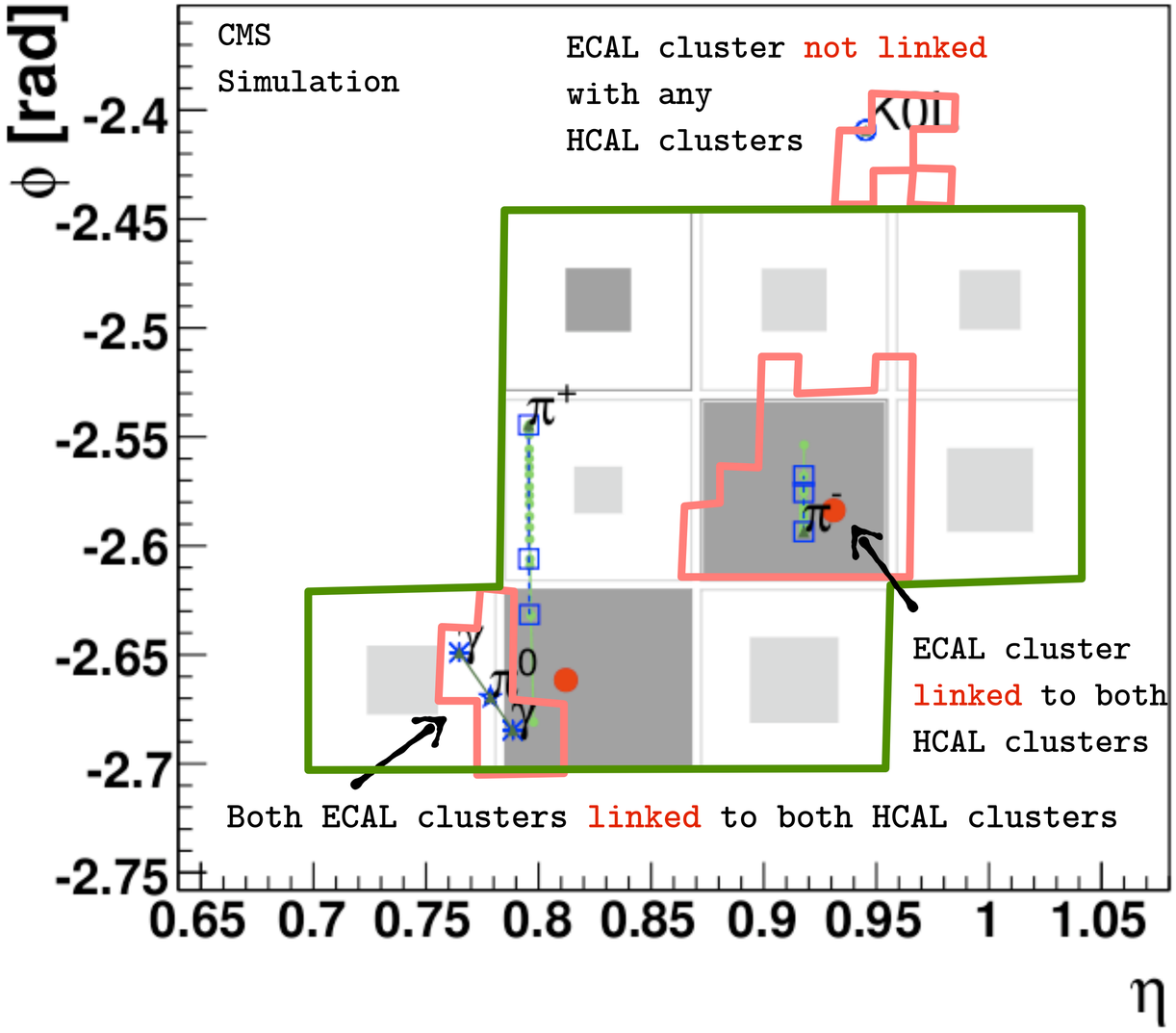




Four true particles:

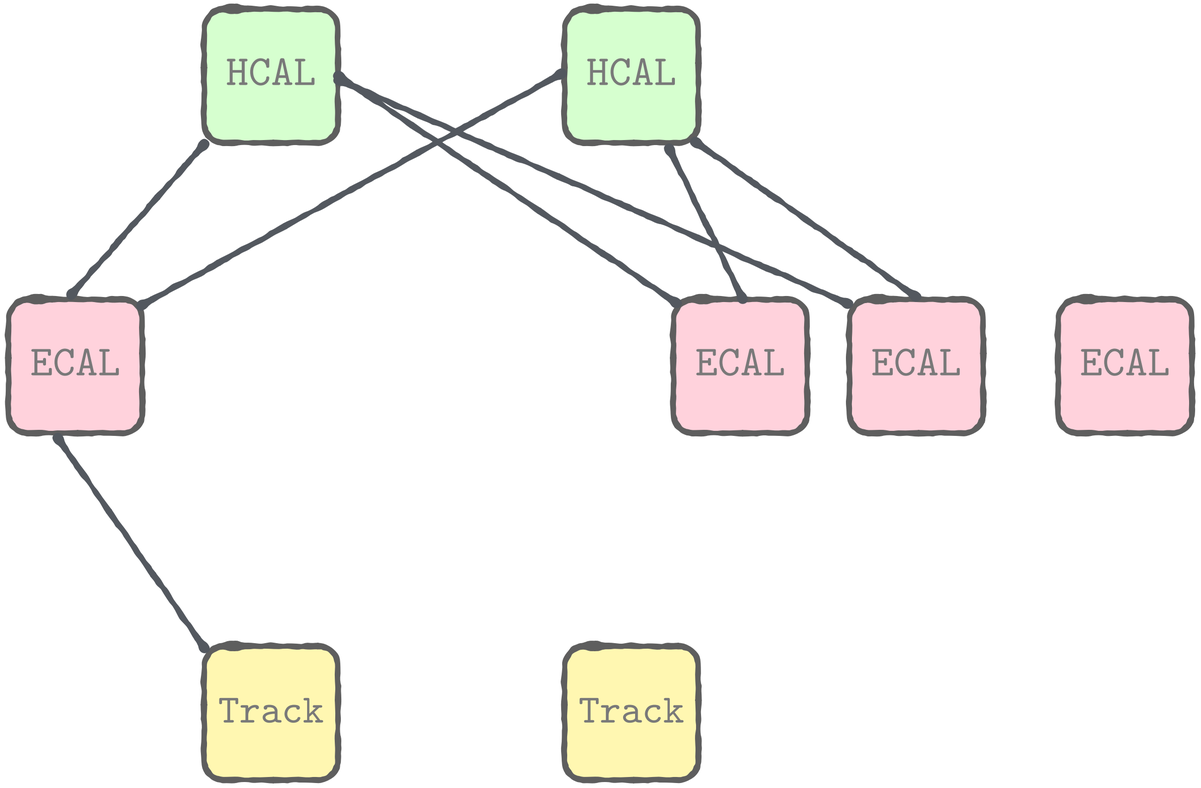
$$\pi^+, \pi^-, \pi^0, K_L^0$$

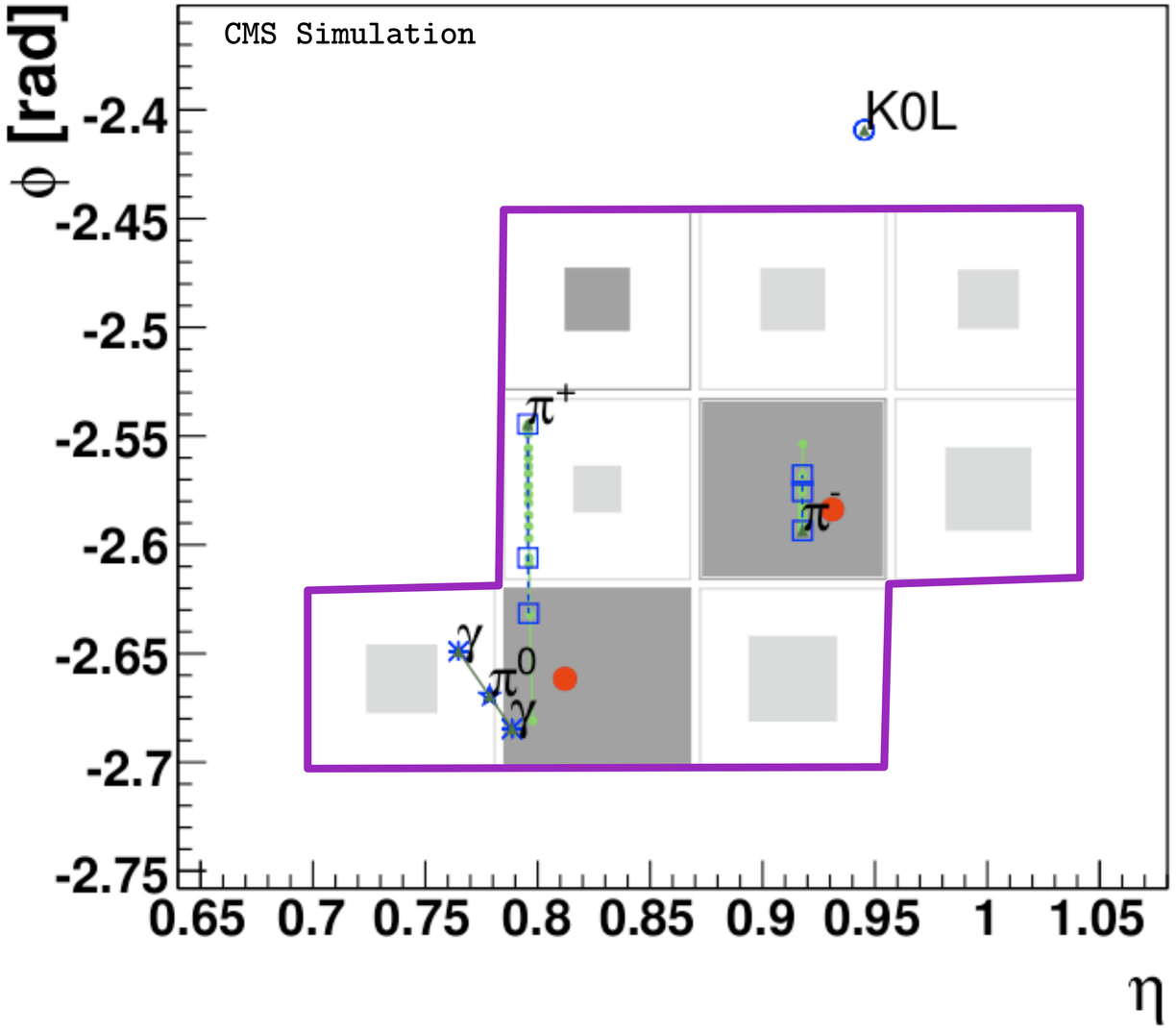




Four true particles:

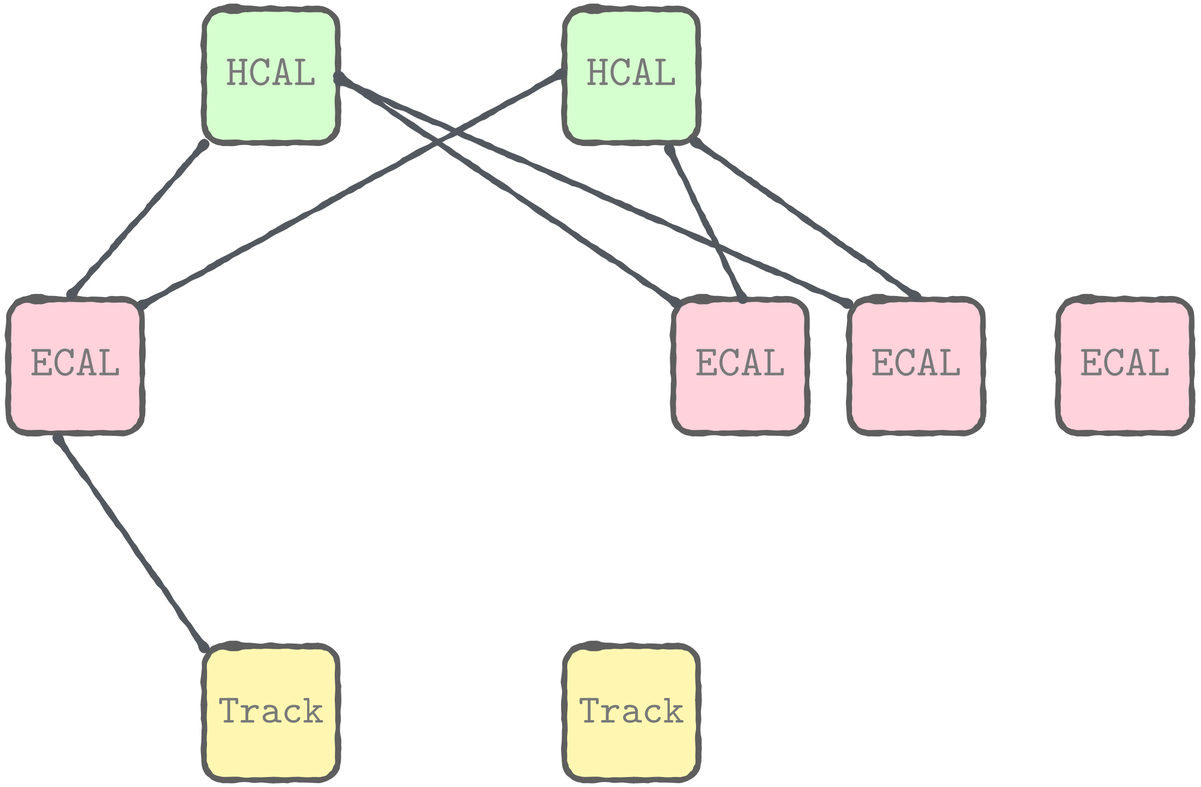
$\pi^+, \pi^-, \pi^0, K_L^0$

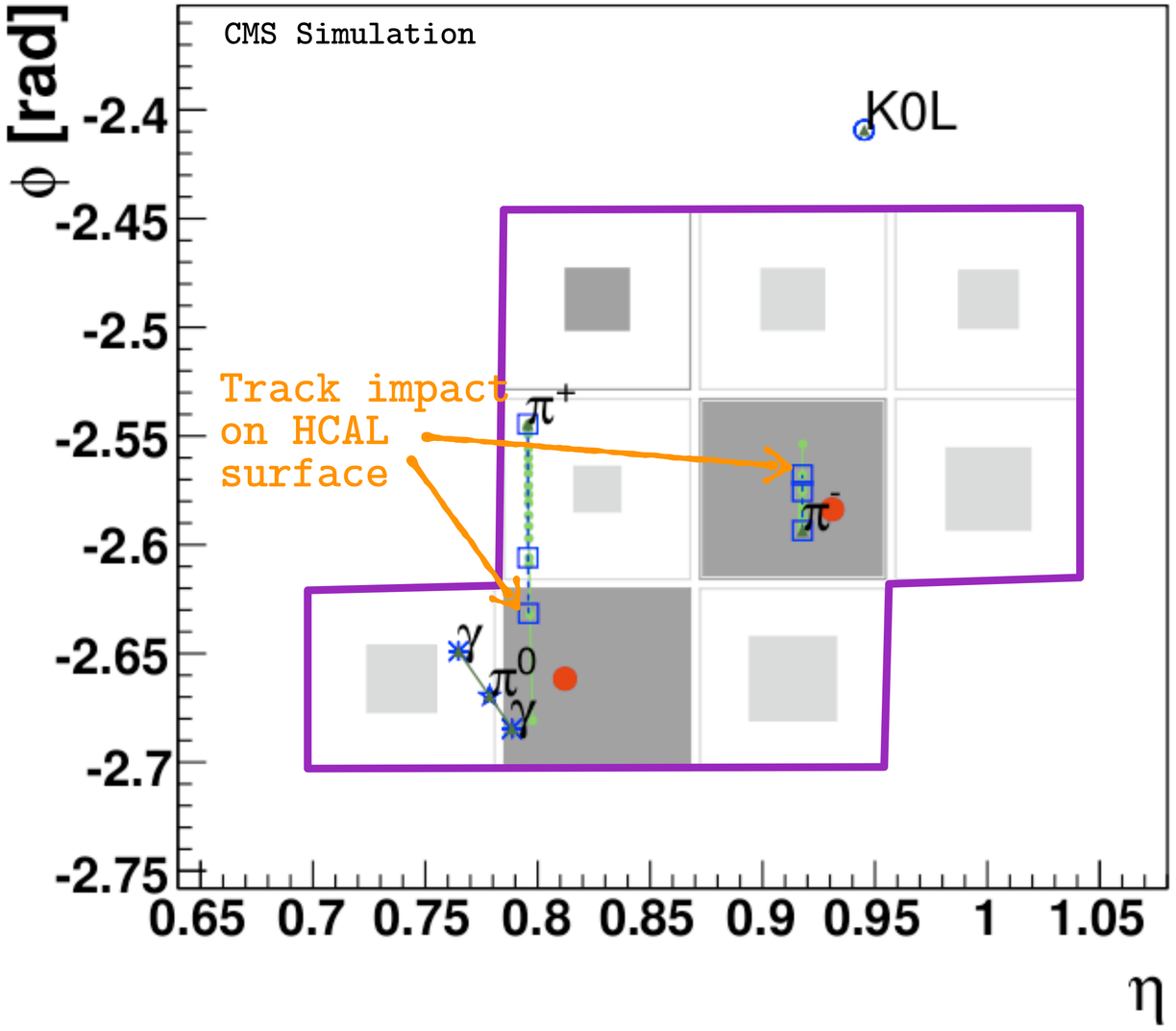




Four true particles:

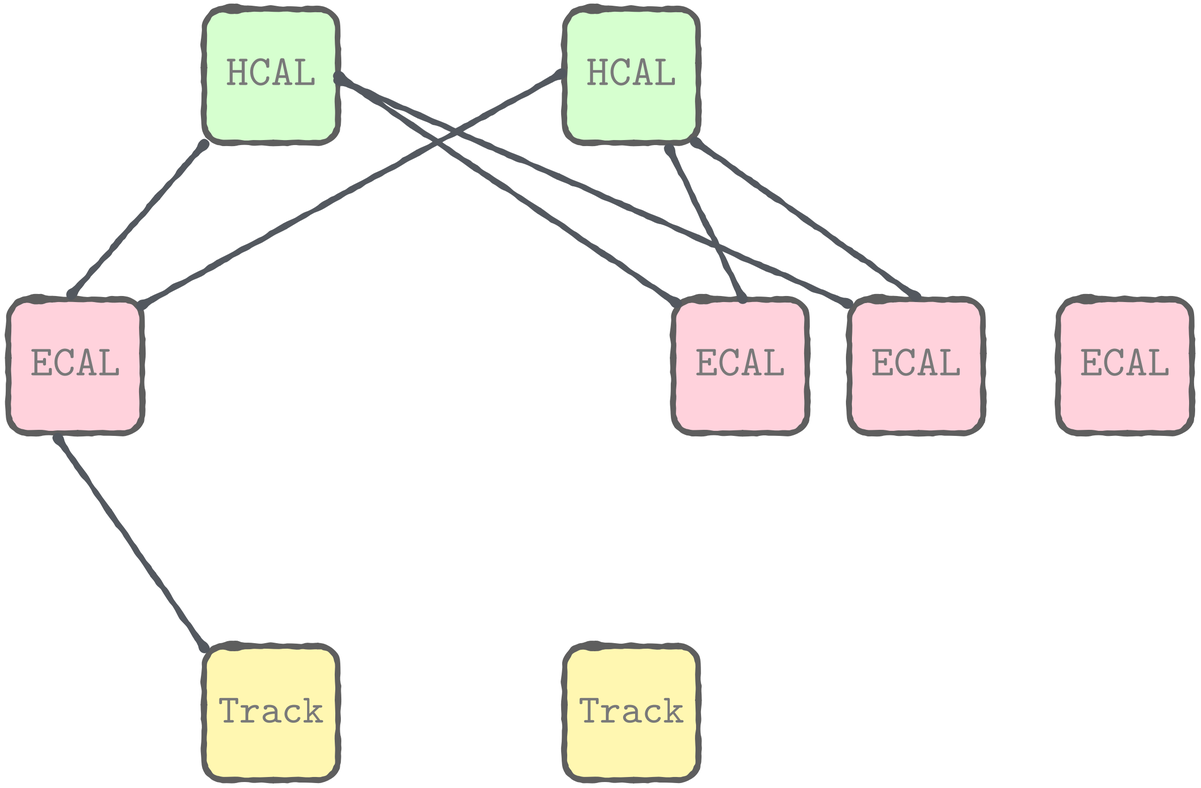
$\pi^+, \pi^-, \pi^0, K_L^0$

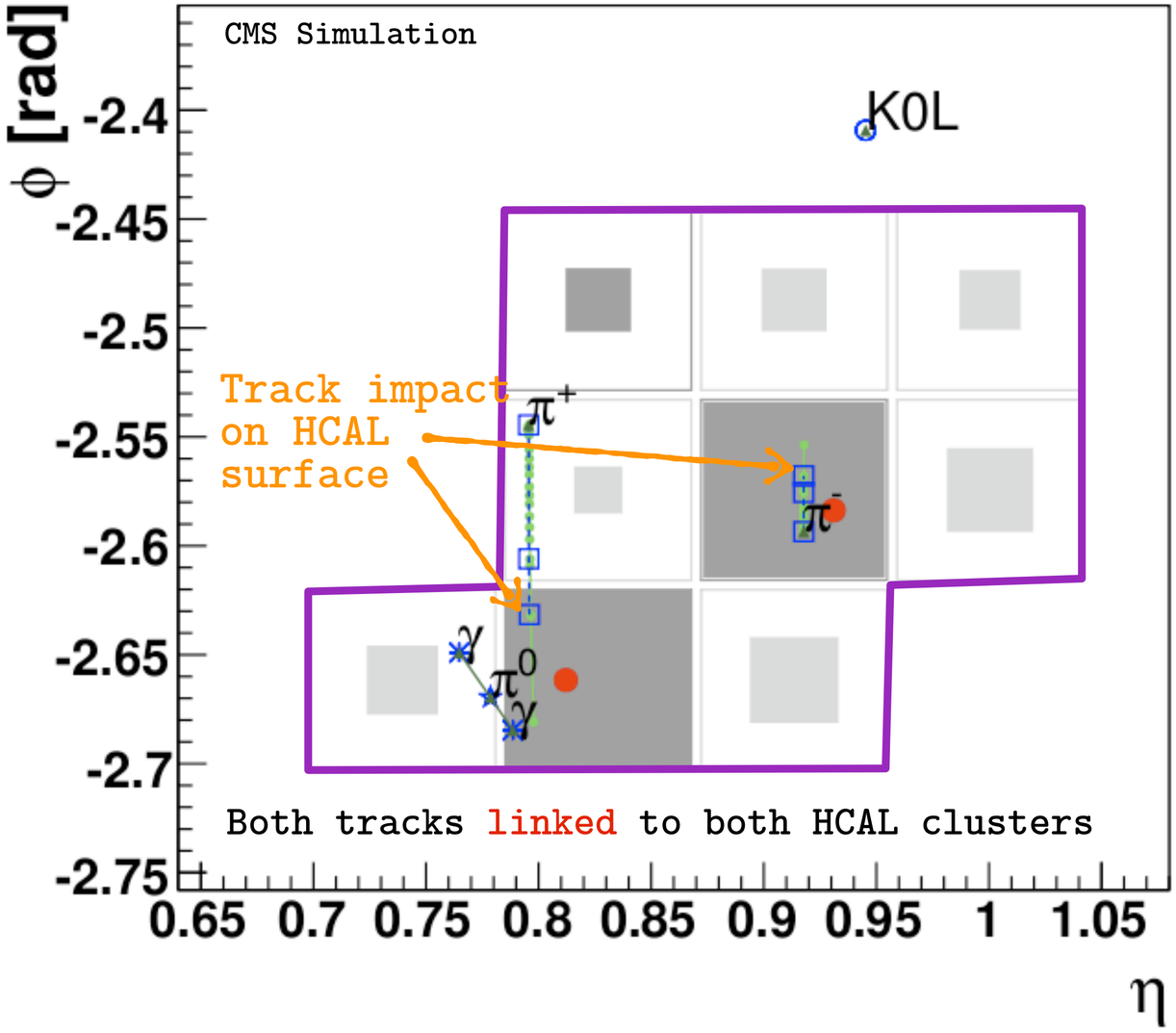




Four true particles:

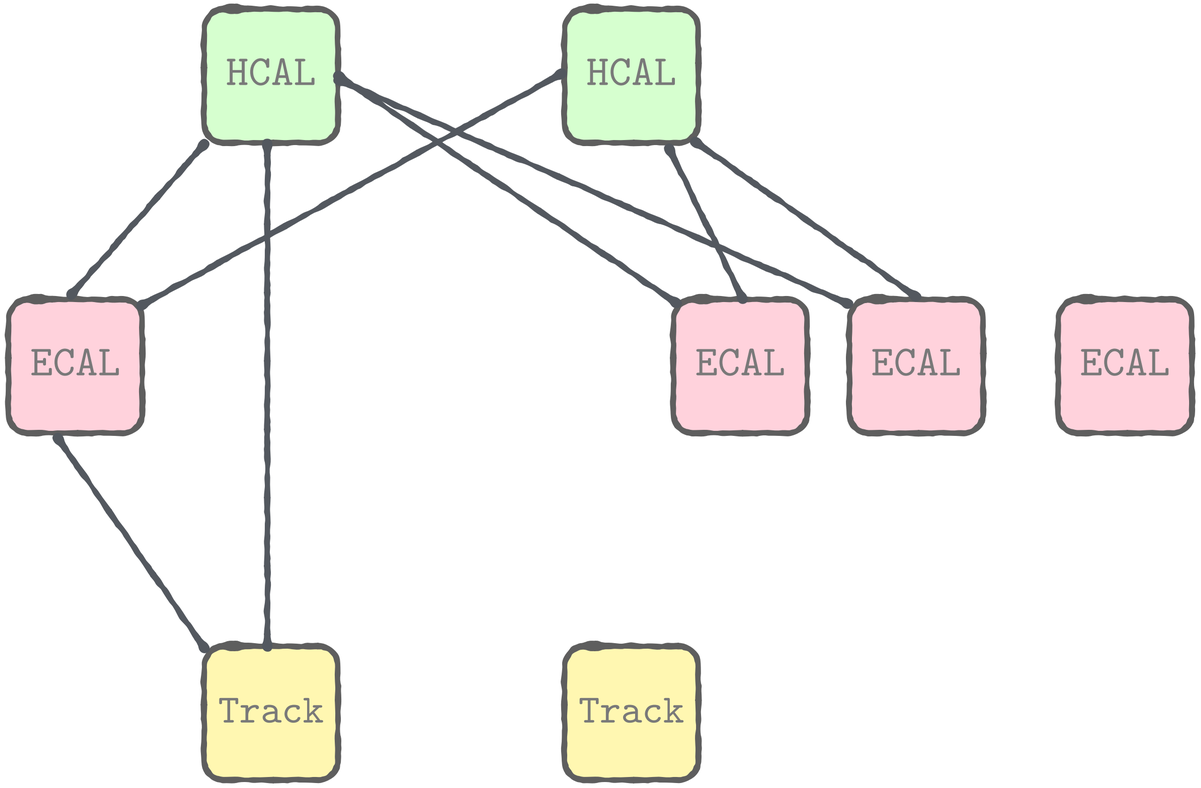
$\pi^+, \pi^-, \pi^0, K_L^0$

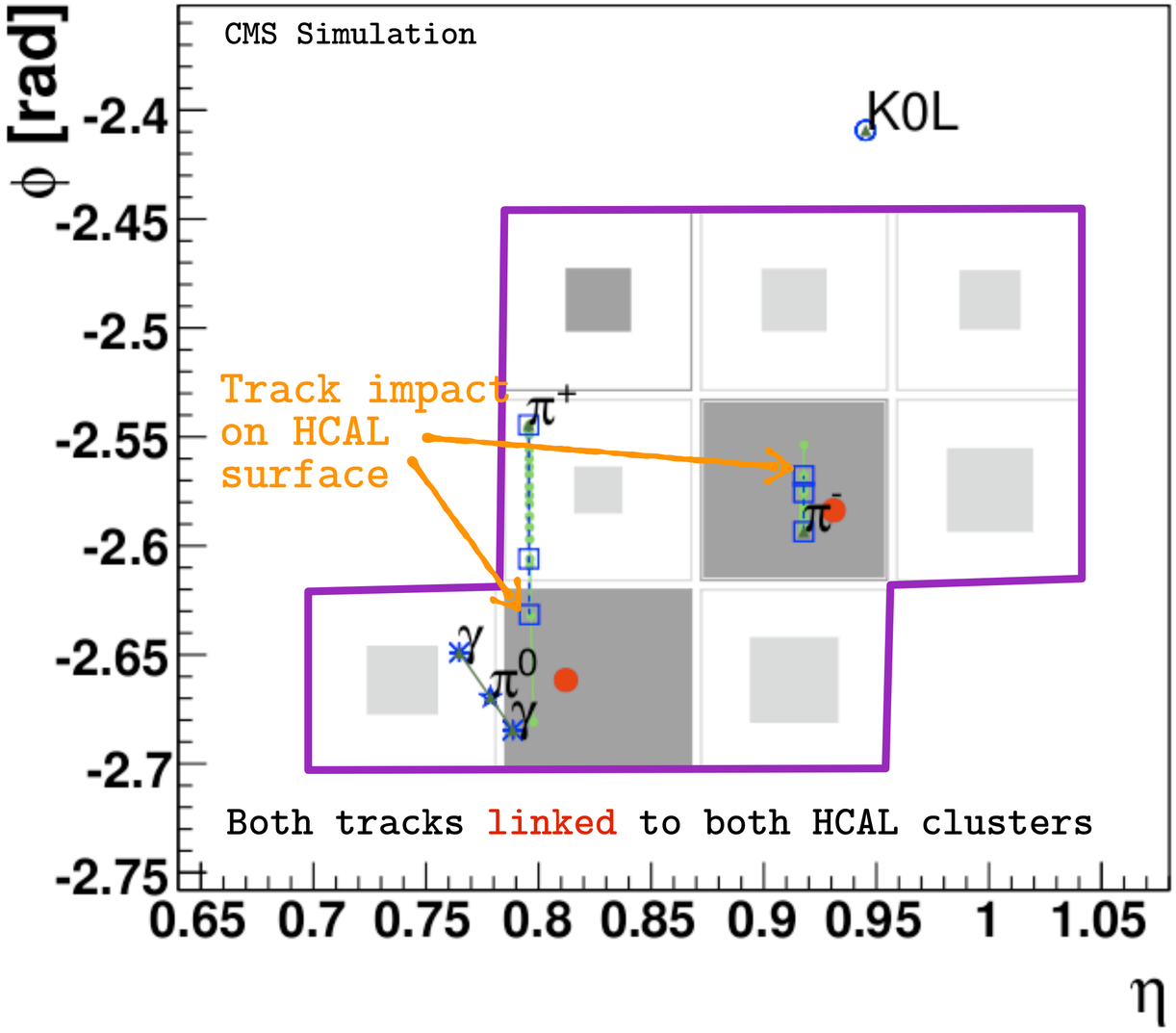




Four true particles:

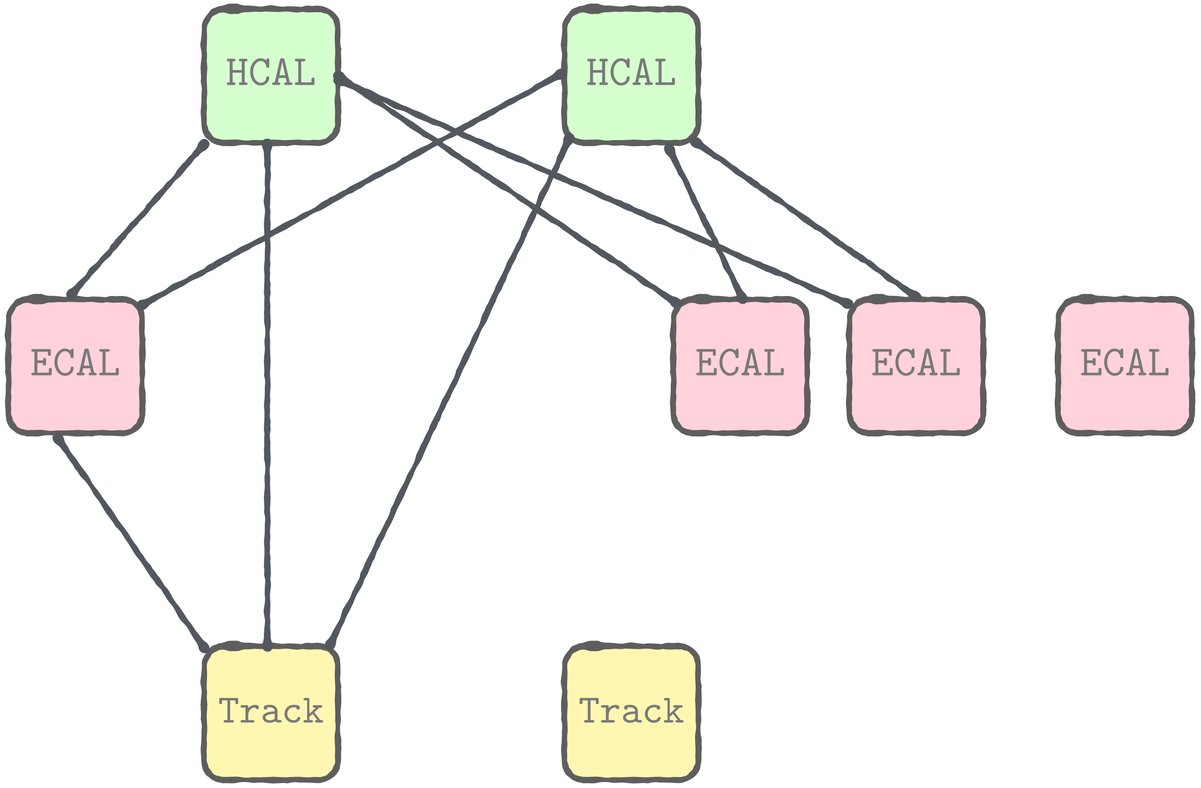
$$\pi^+, \pi^-, \pi^0, K_L^0$$

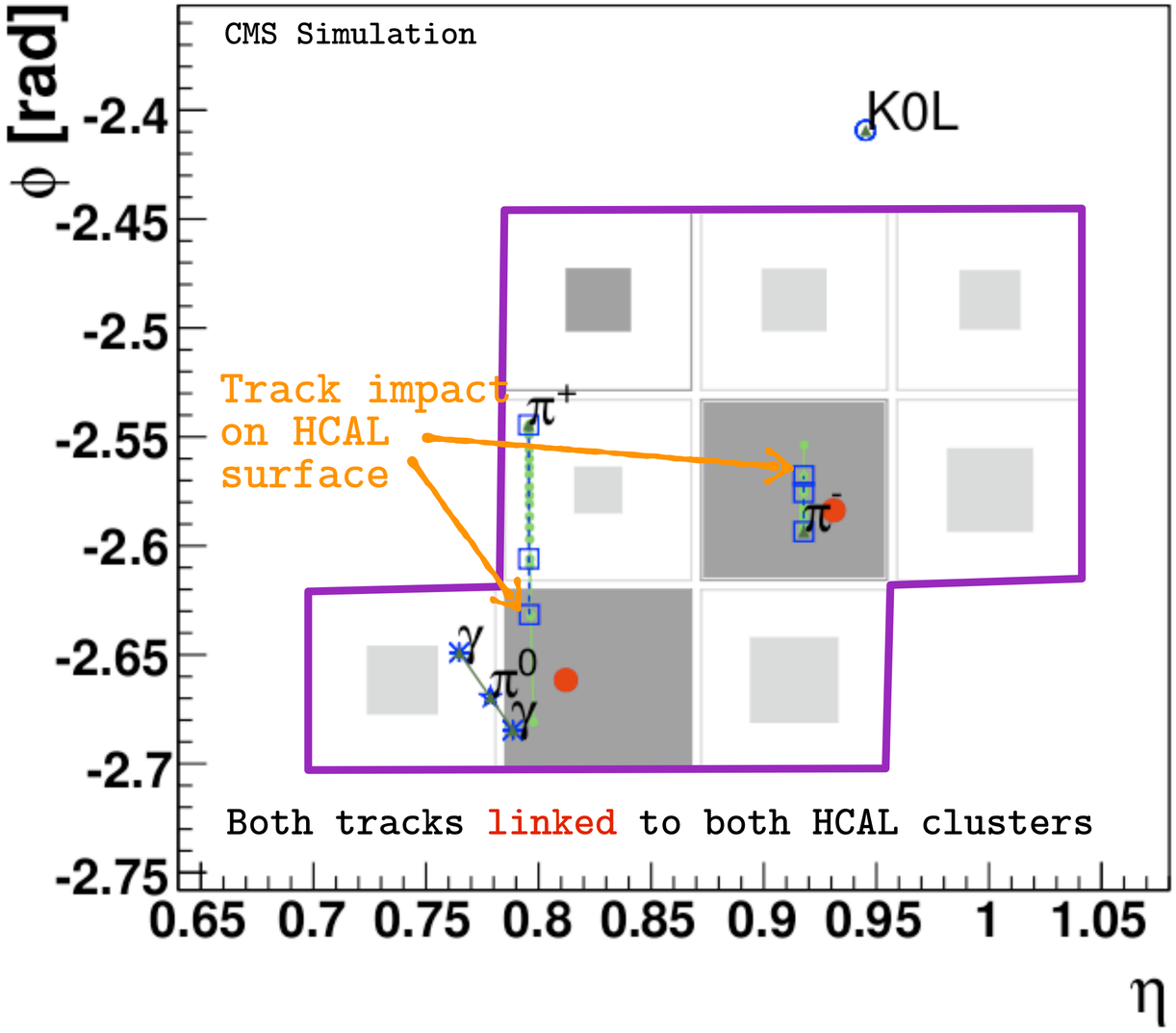




Four true particles:

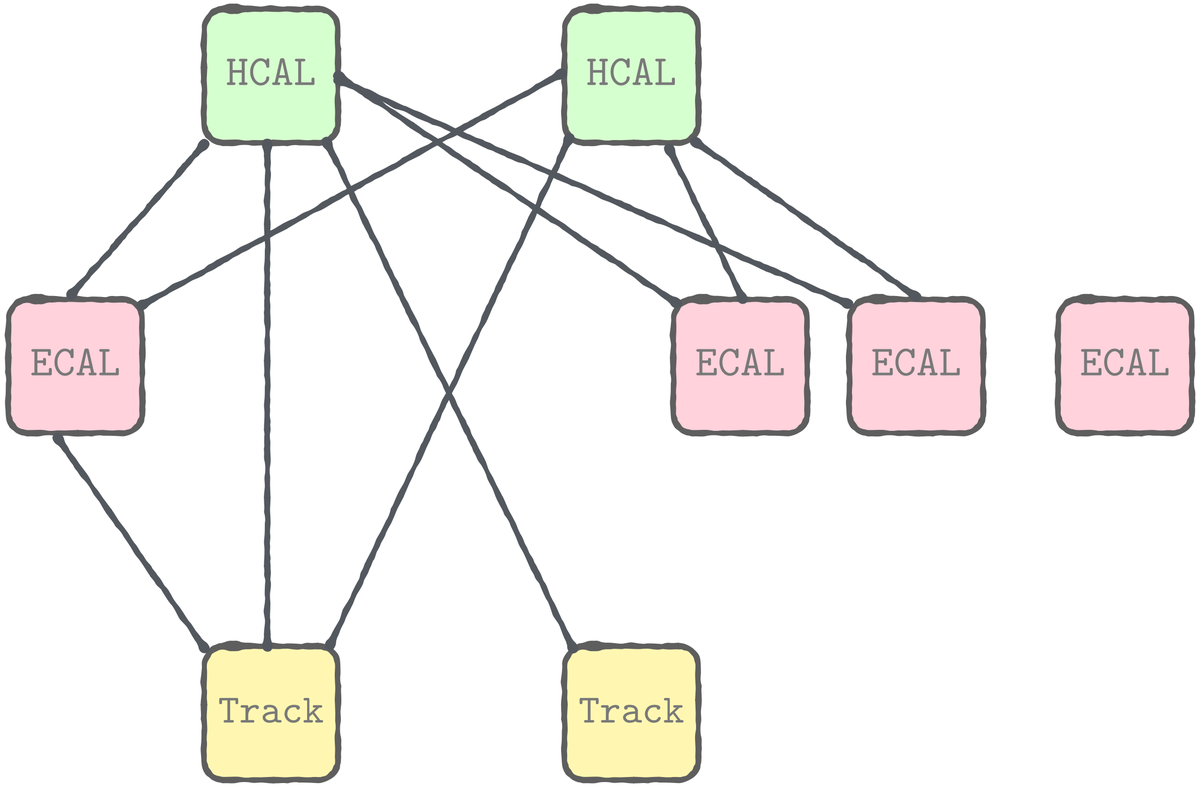
$$\pi^+, \pi^-, \pi^0, K_L^0$$

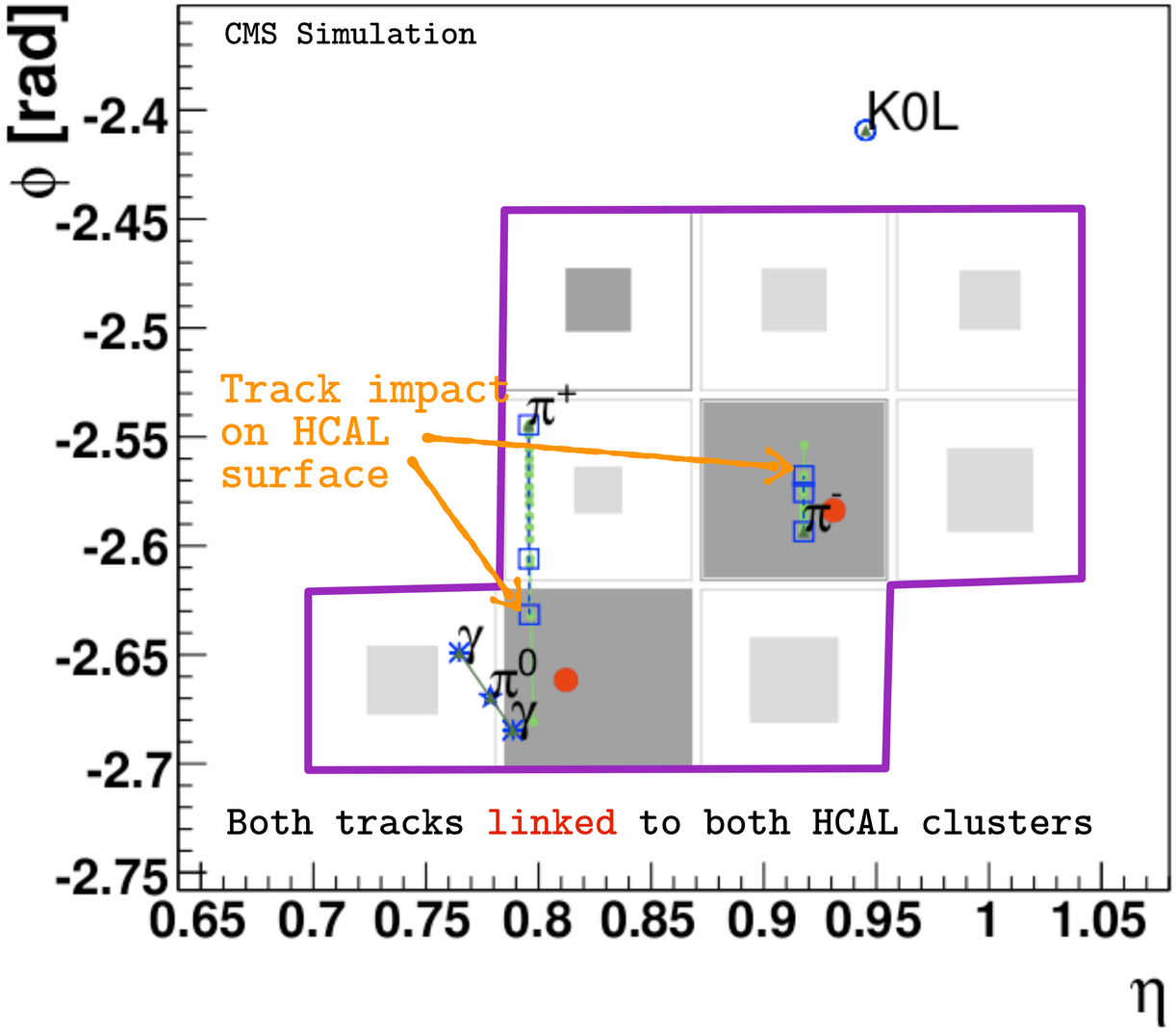




Four true particles:

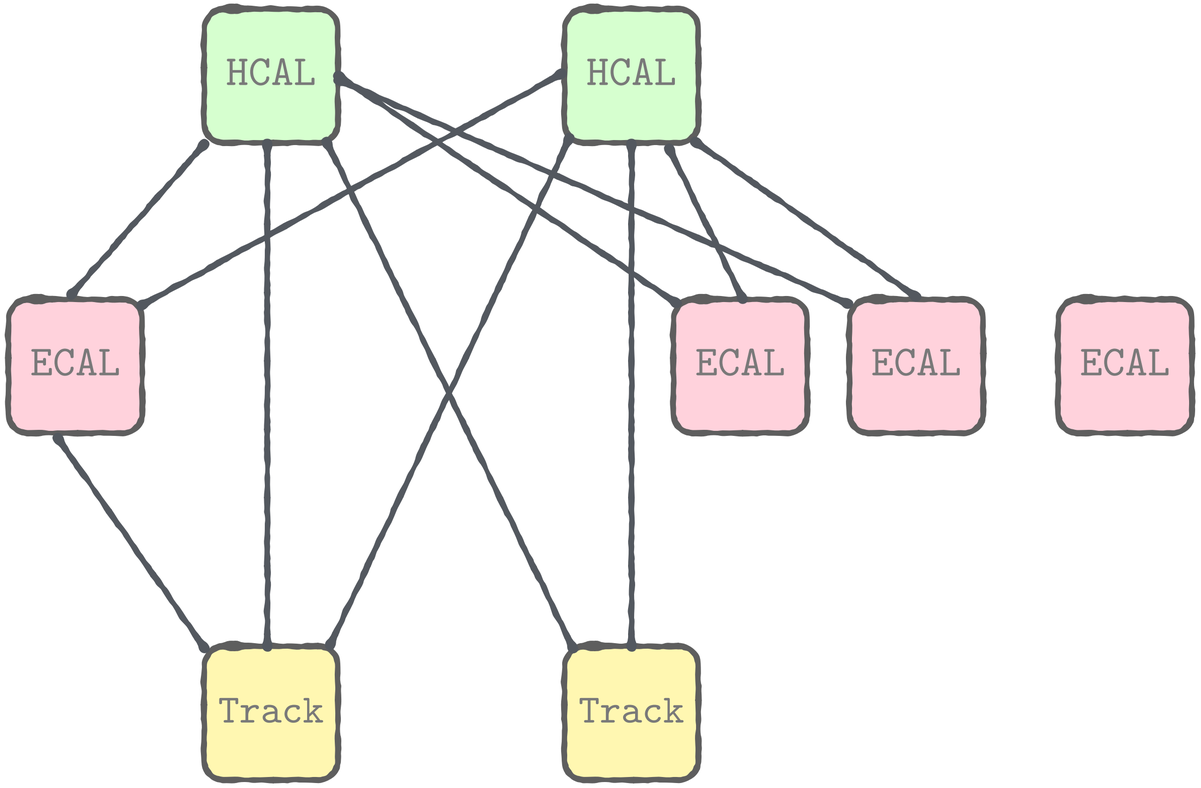
$$\pi^+, \pi^-, \pi^0, K_L^0$$

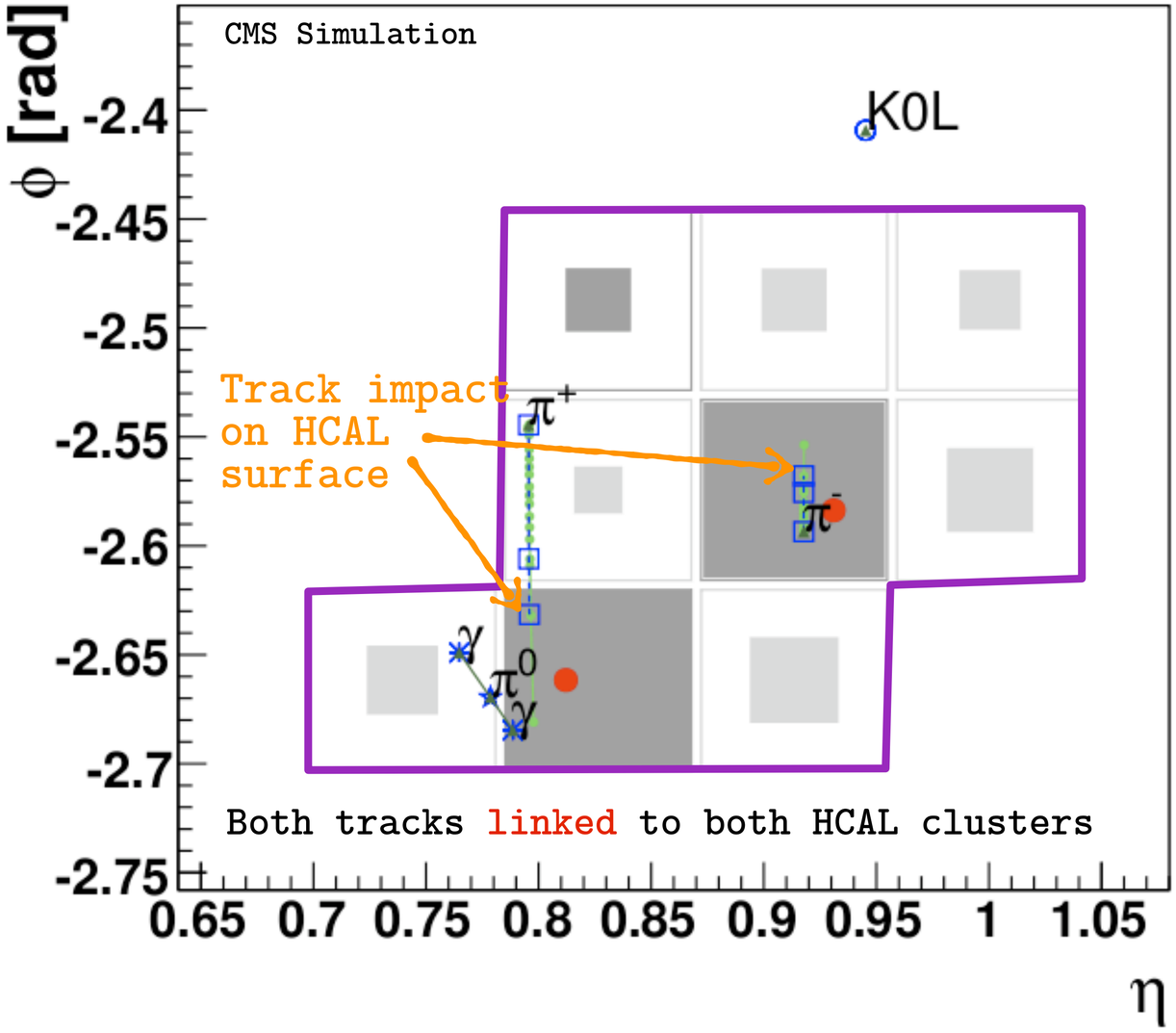




Four true particles:

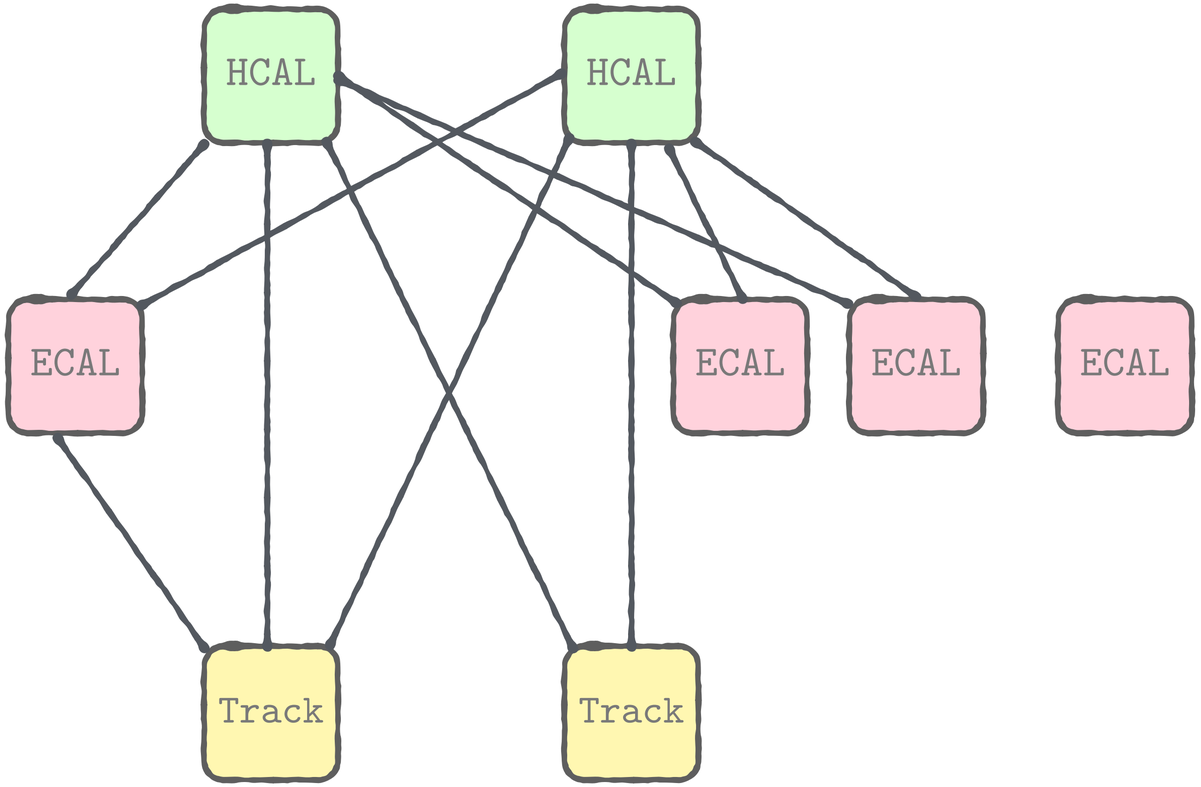
$$\pi^+, \pi^-, \pi^0, K_L^0$$





Four true particles:

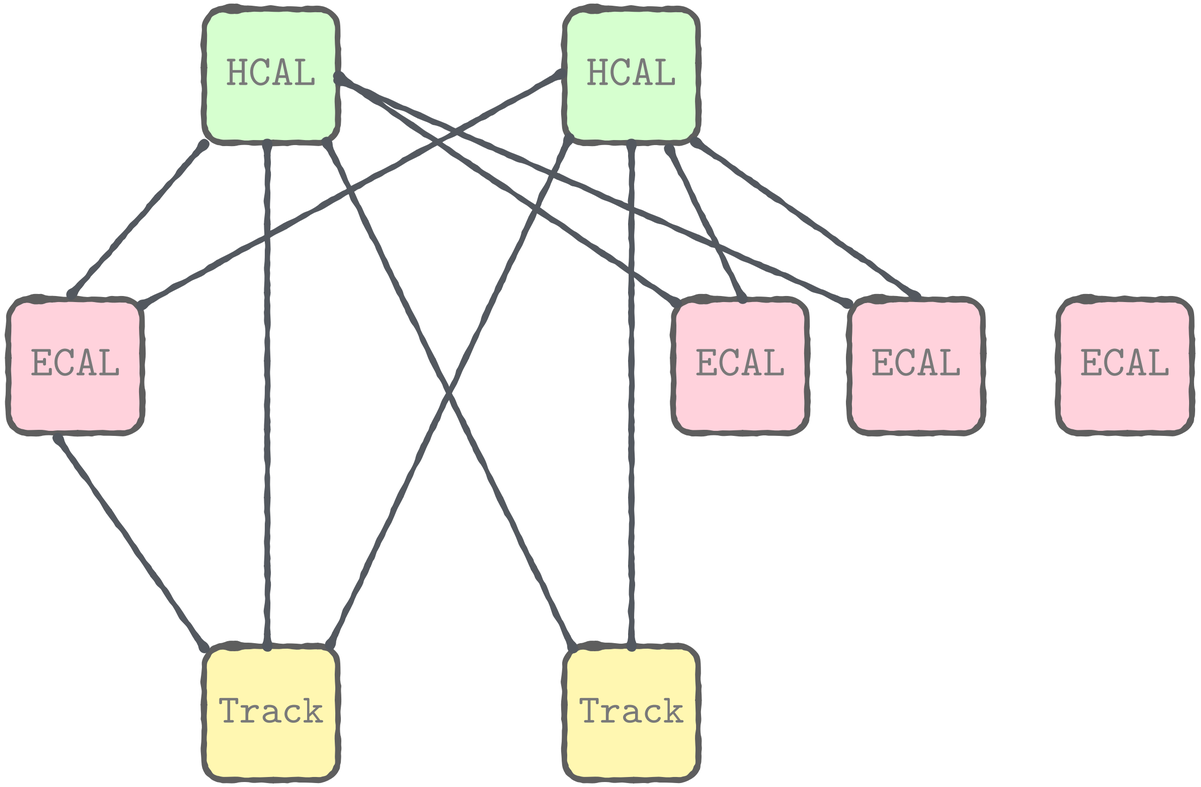
$$\pi^+, \pi^-, \pi^0, K_L^0$$



Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

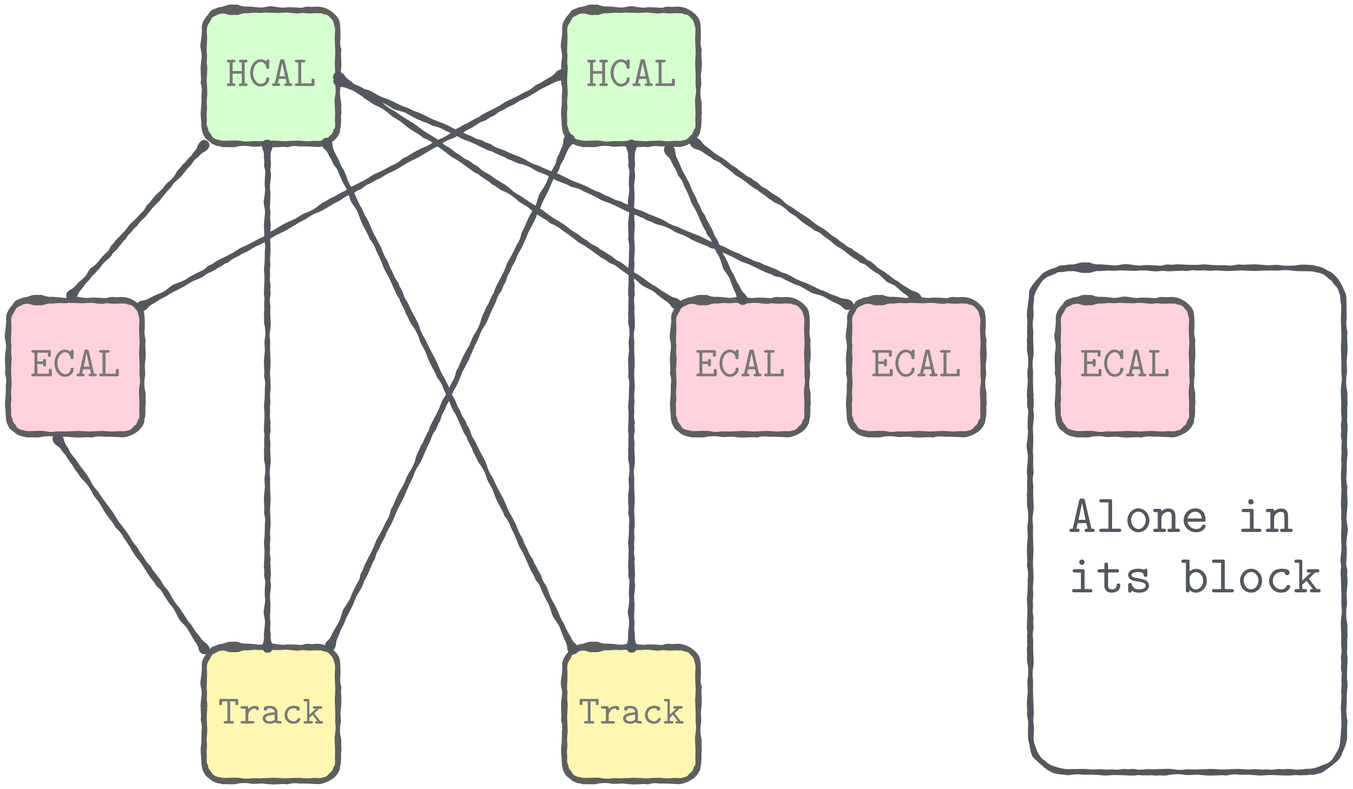
Build "blocks" of linked elements



Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

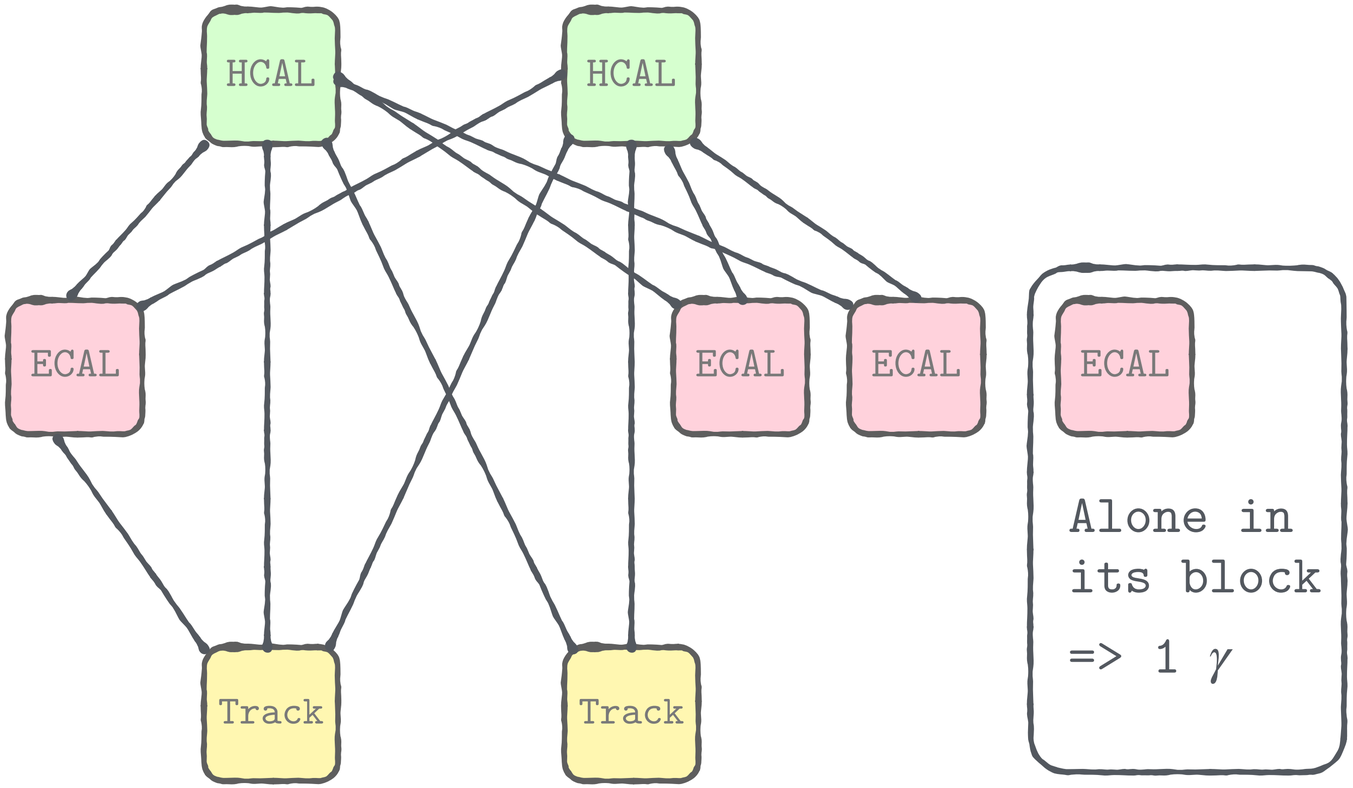
Build "blocks" of linked elements



Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

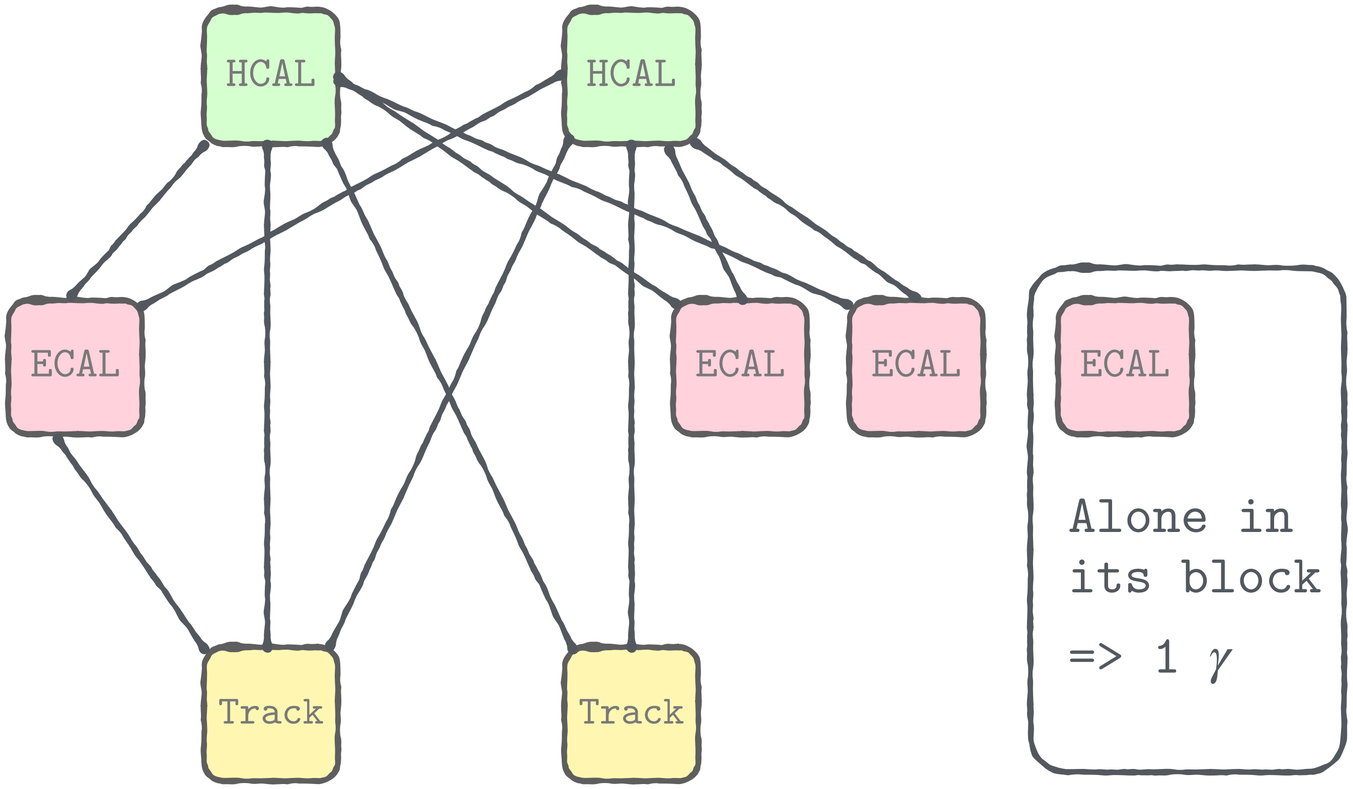
Build "blocks" of linked elements



Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Build "blocks" of linked elements

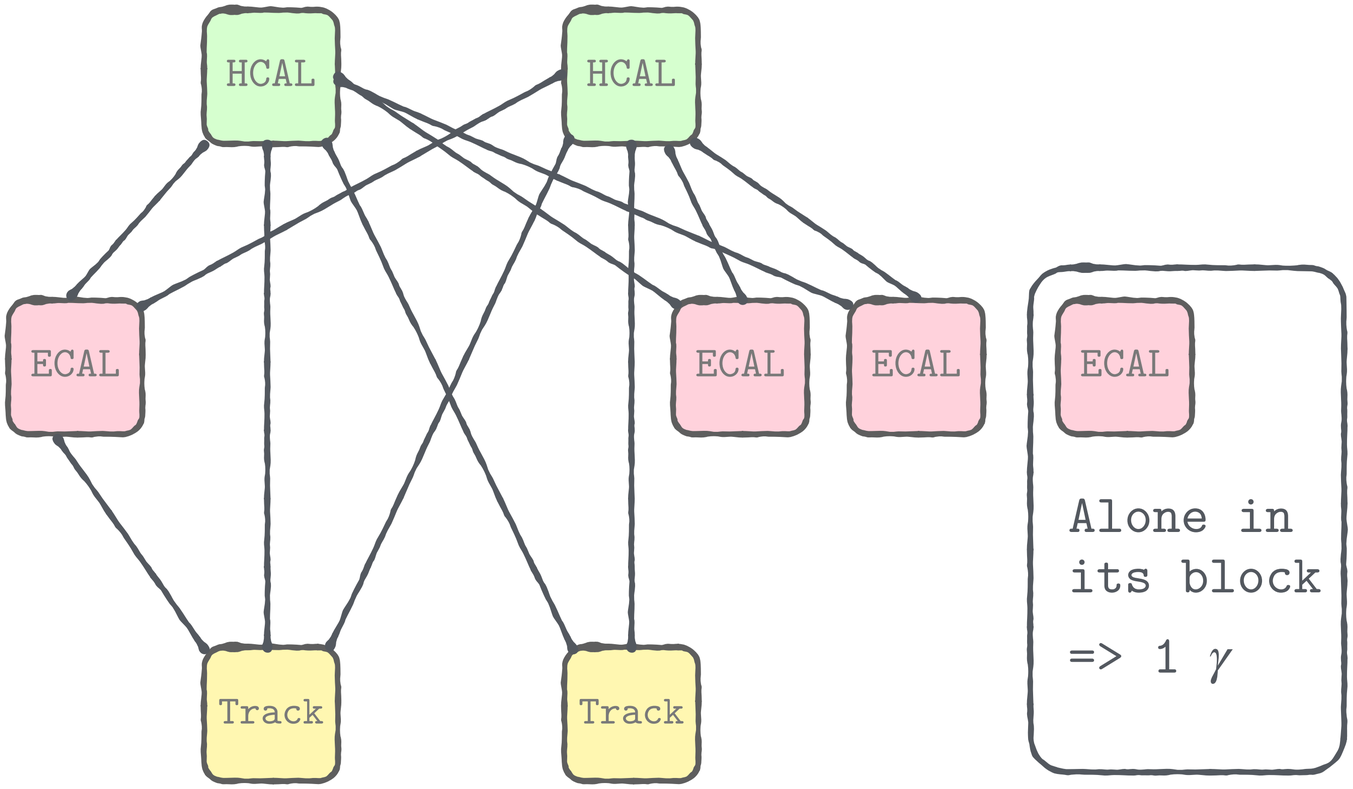


List of reconstructed (candidate) particles

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Build "blocks" of linked elements



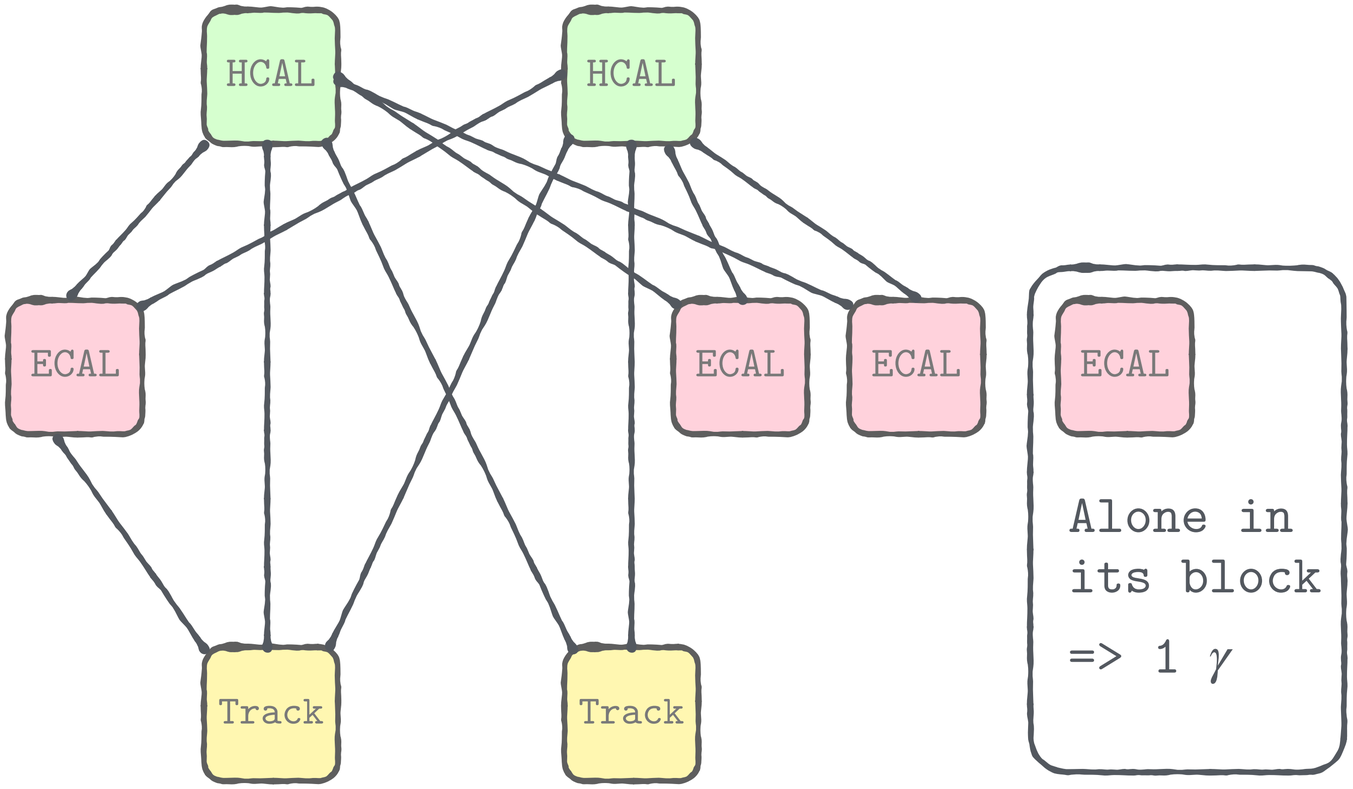
List of reconstructed (candidate) particles

{

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Build "blocks" of linked elements

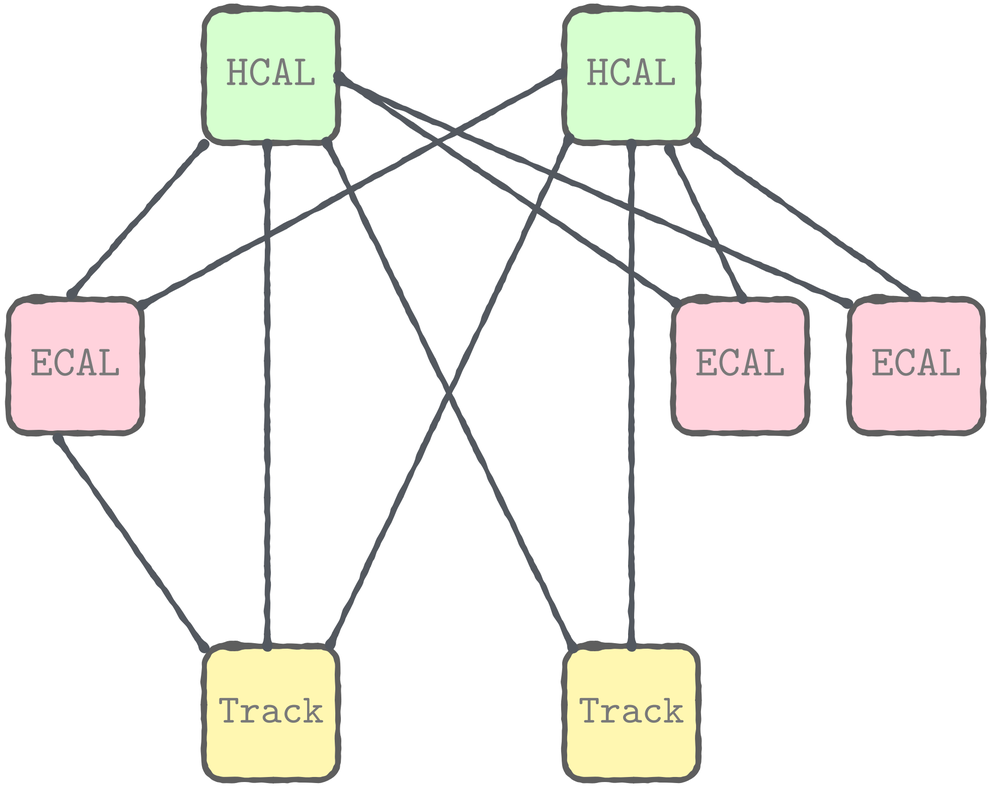


List of reconstructed (candidate) particles

{ γ

Four true particles:

$$\pi^+, \pi^-, \pi^0, K_L^0$$



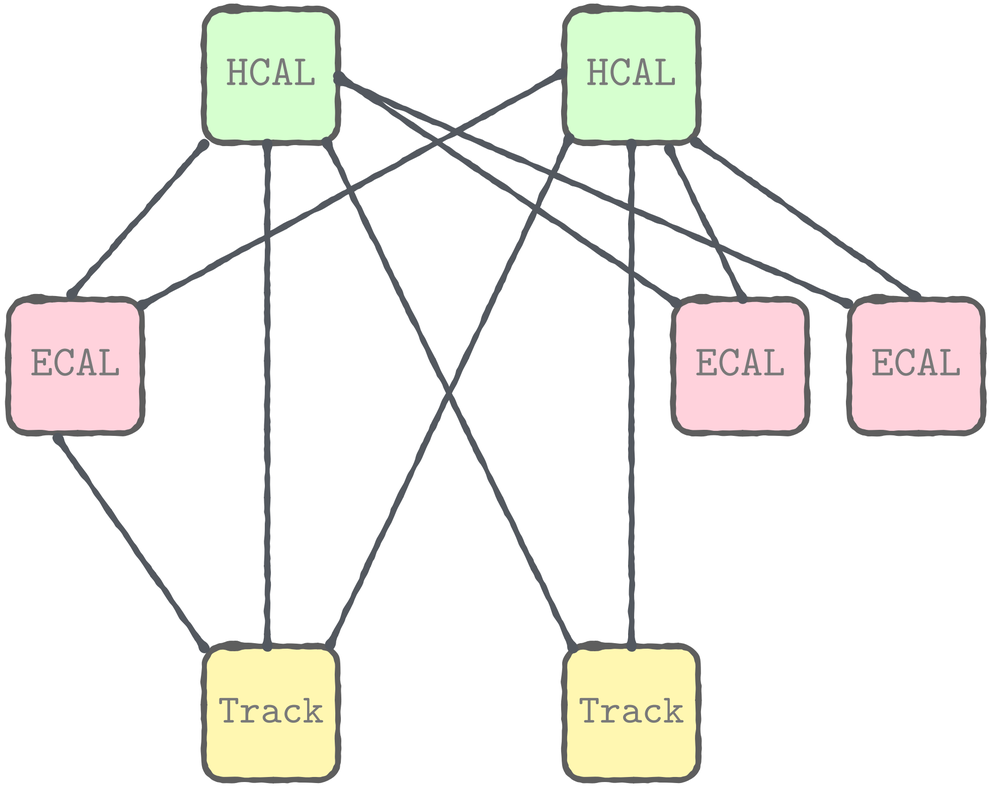
List of reconstructed (candidate) particles

{ γ

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Find photons in the blocks



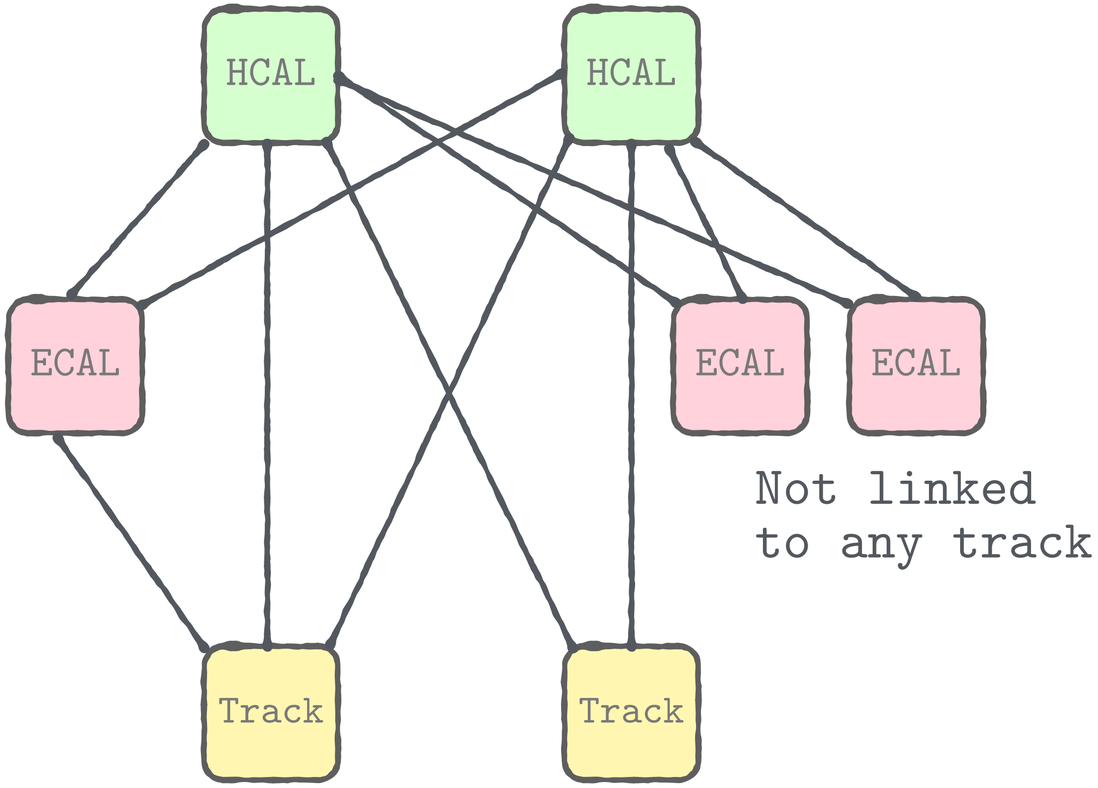
List of reconstructed (candidate) particles

{ γ

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Find photons in the blocks



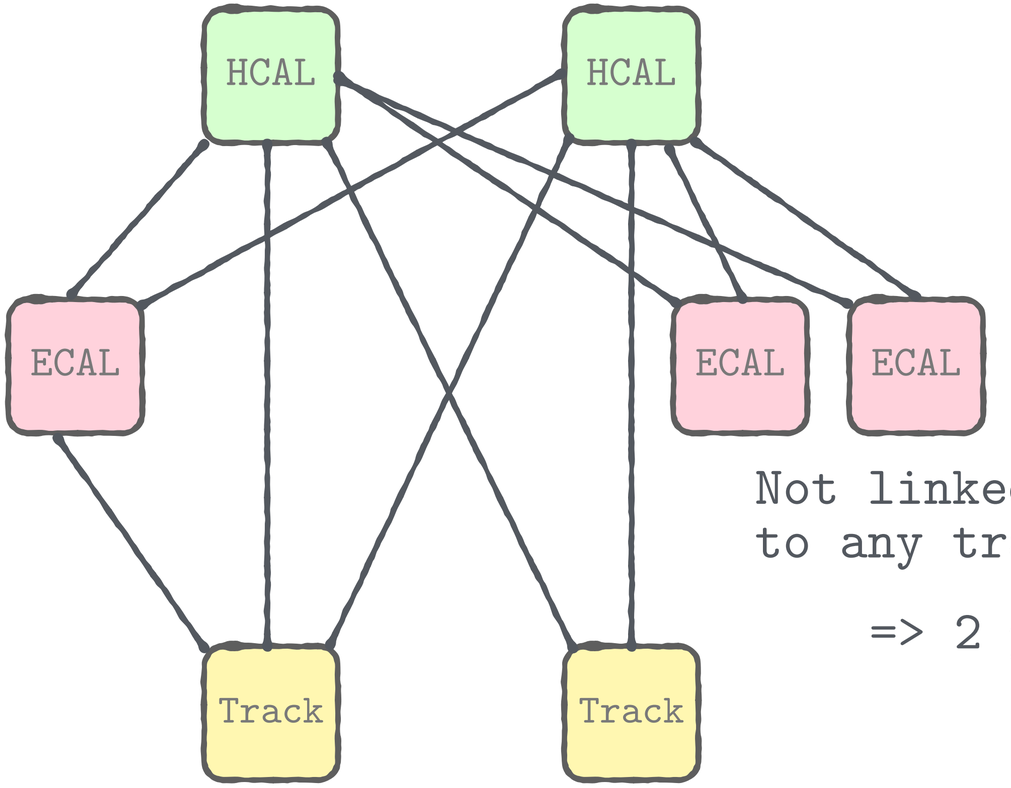
List of reconstructed (candidate) particles

{ γ

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Find photons in the blocks



Not linked
to any track

=> 2 γ's

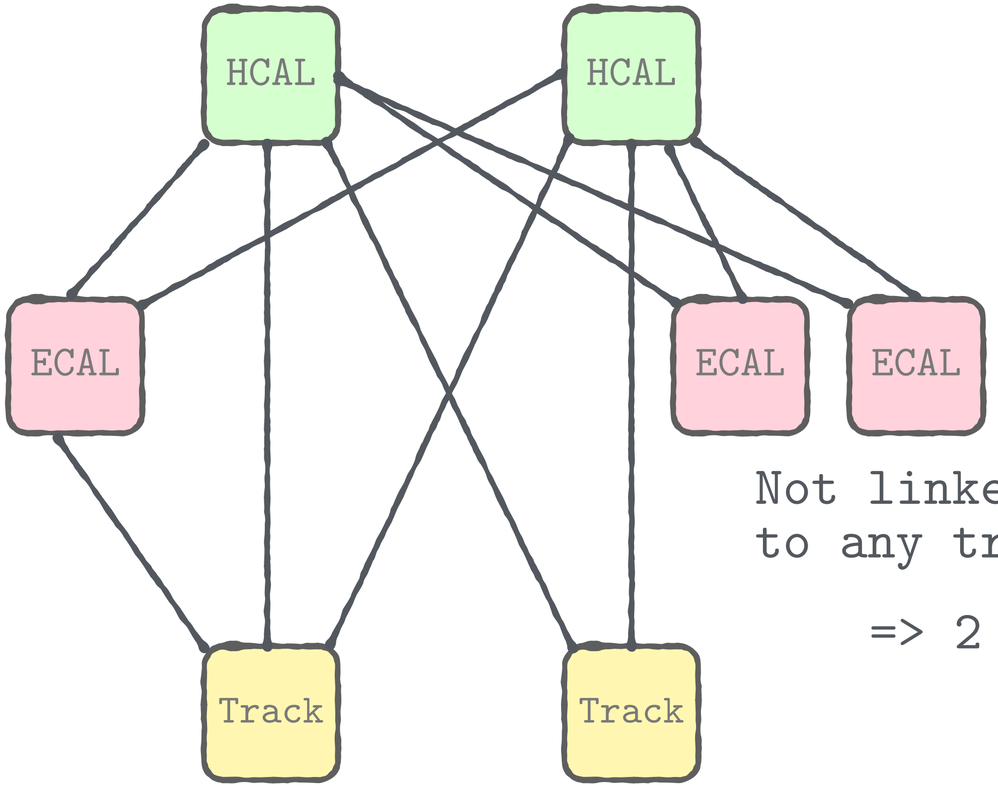
List of reconstructed (candidate) particles

{ γ

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Find photons in the blocks



Please ask: Why not neutral hadrons???

Not linked to any track

=> 2 γ's

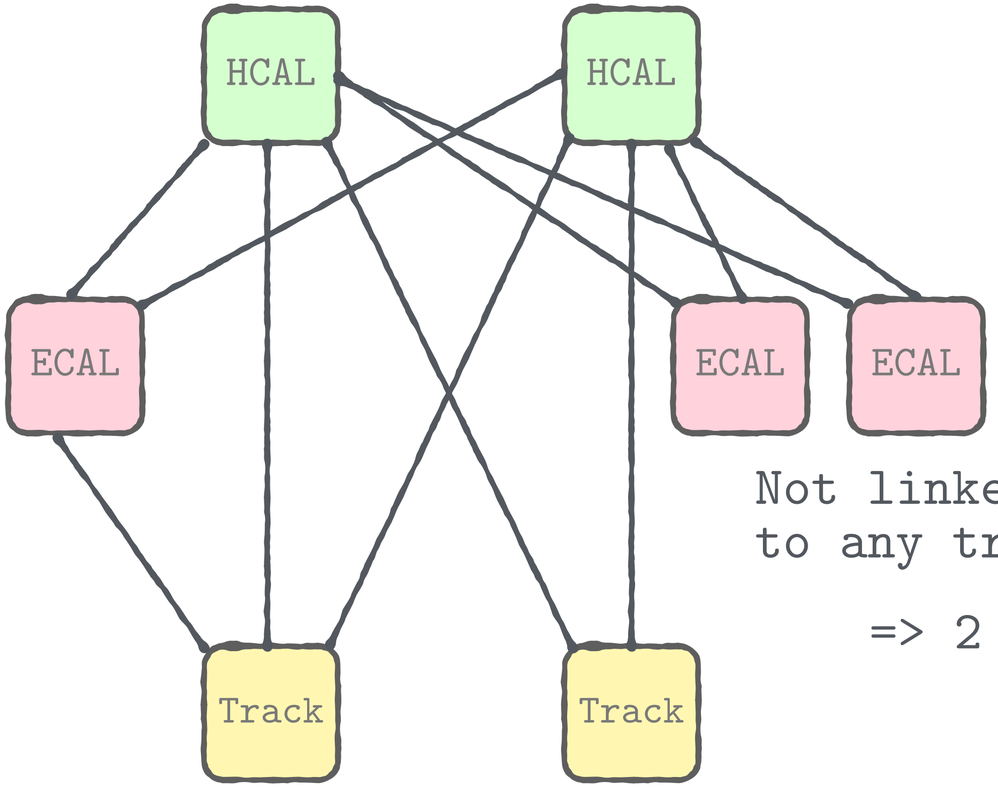
List of reconstructed (candidate) particles

{ γ

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Find photons in the blocks



Not linked
to any track

=> 2 γ's

Please ask: Why not neutral hadrons???

Quick A: Give precedence to photons!
Jet: O(25%) photons, O(10%) neutral hadrons

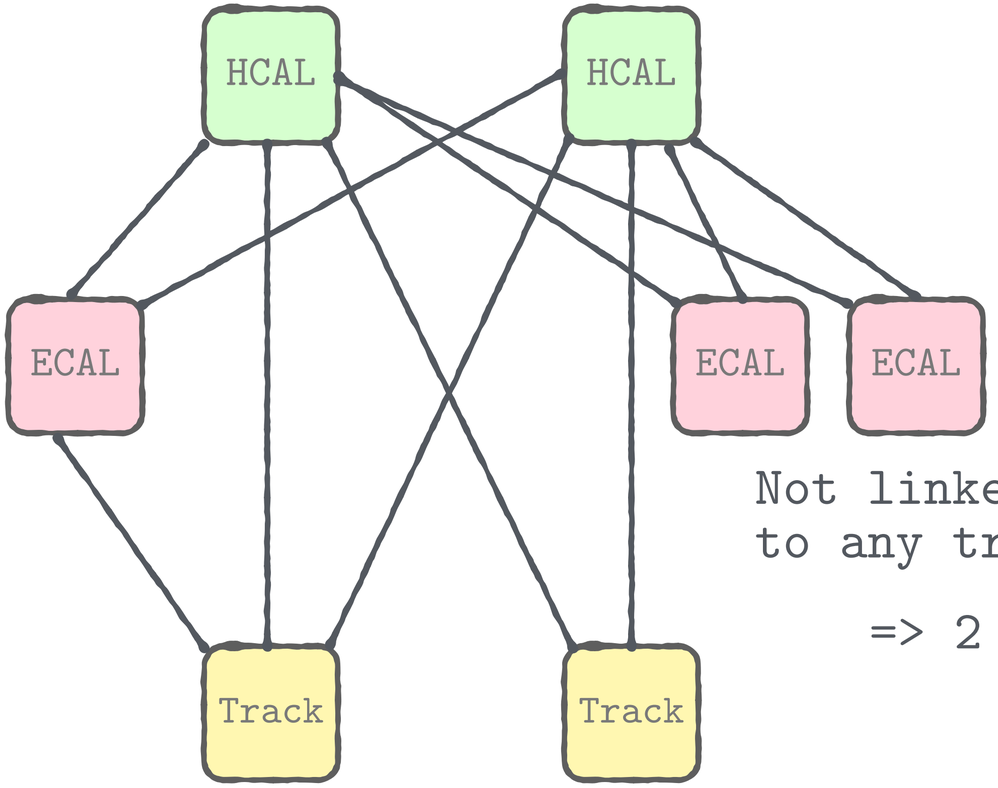
List of reconstructed (candidate) particles

{ γ

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Find photons in the blocks



Please ask: Why not neutral hadrons???

Not linked to any track

=> 2 γ's

Quick A: Give precedence to photons!
Jet: O(25%) photons, O(10%) neutral hadrons

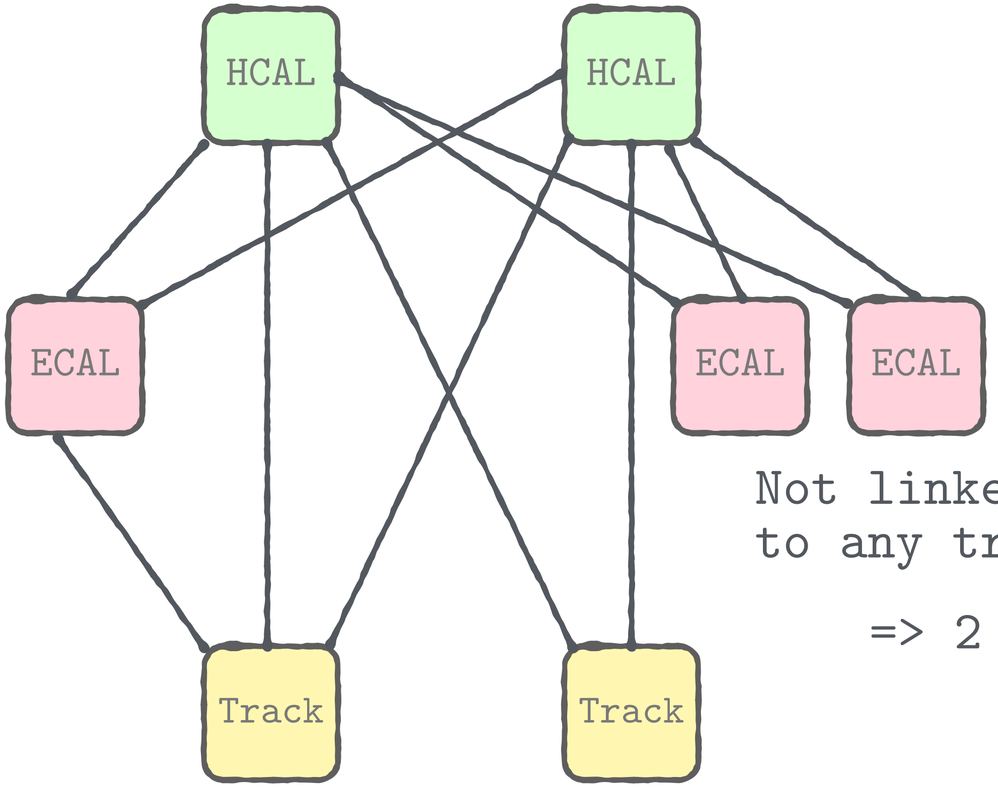
List of reconstructed (candidate) particles

{ γ,

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Find photons in the blocks



Not linked
to any track

=> 2 γ's

Please ask: Why not neutral hadrons???

Quick A: Give precedence to photons!
Jet: O(25%) photons, O(10%) neutral hadrons

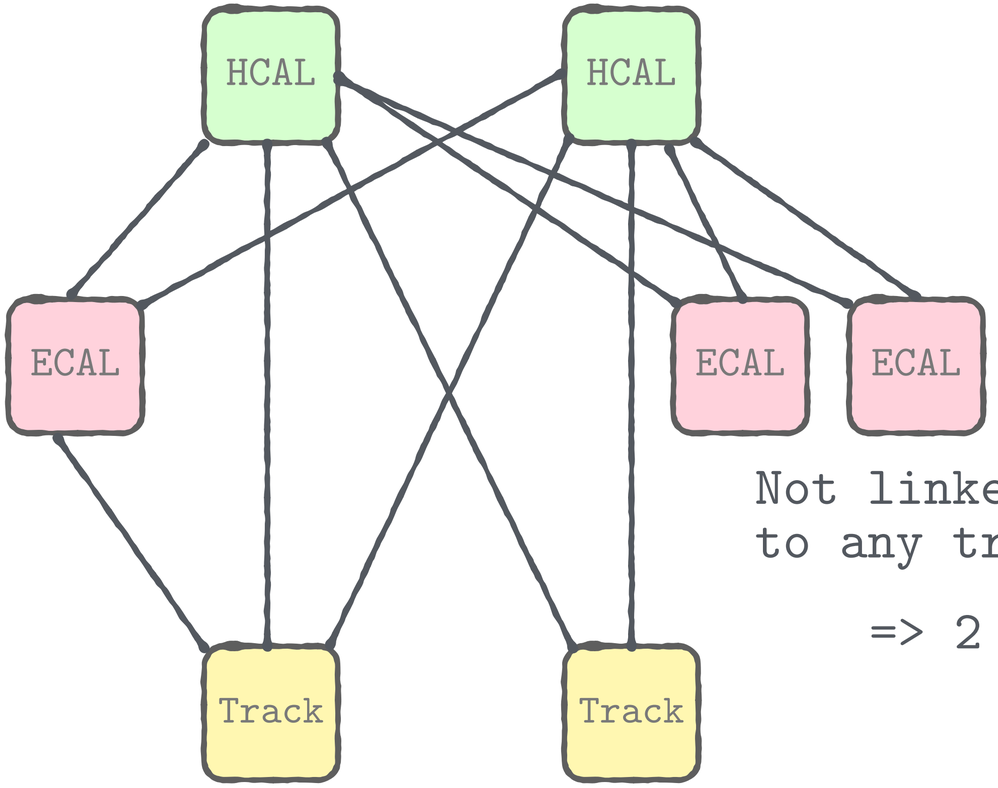
List of reconstructed (candidate) particles

{ γ , γ

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Find photons in the blocks



Please ask: Why not neutral hadrons???

Not linked to any track

=> 2 γ's

Quick A: Give precedence to photons!
Jet: O(25%) photons, O(10%) neutral hadrons

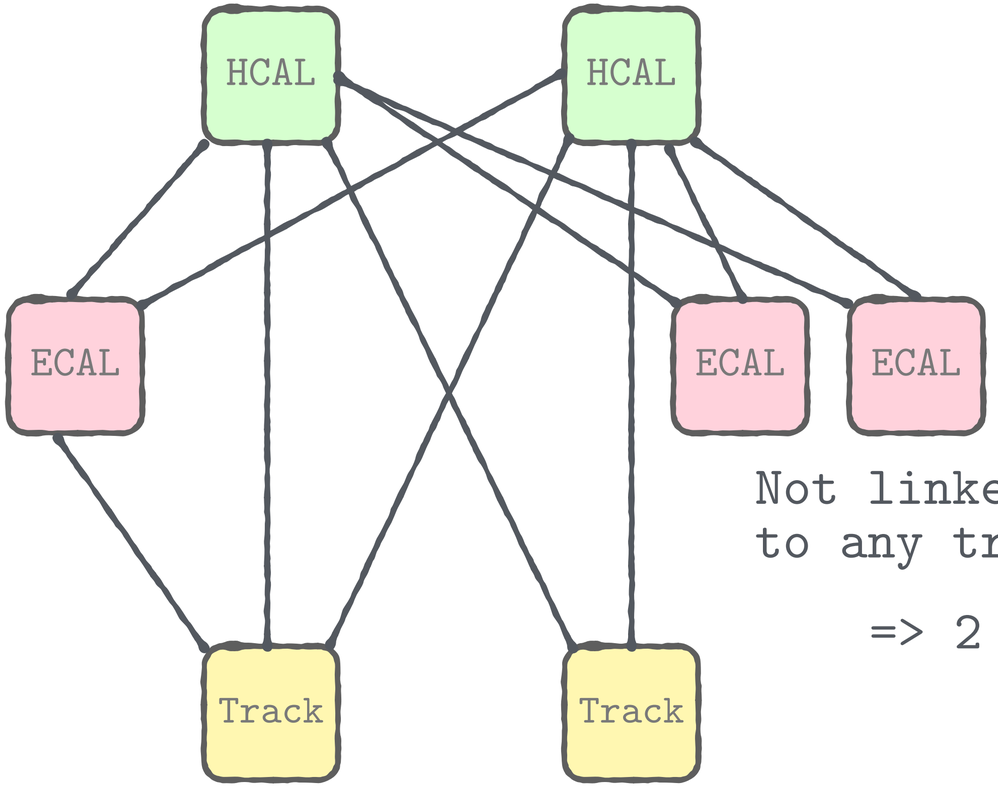
List of reconstructed (candidate) particles

{ γ, γ,

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Find photons in the blocks



Please ask: Why not neutral hadrons???

Not linked to any track

=> 2 γ's

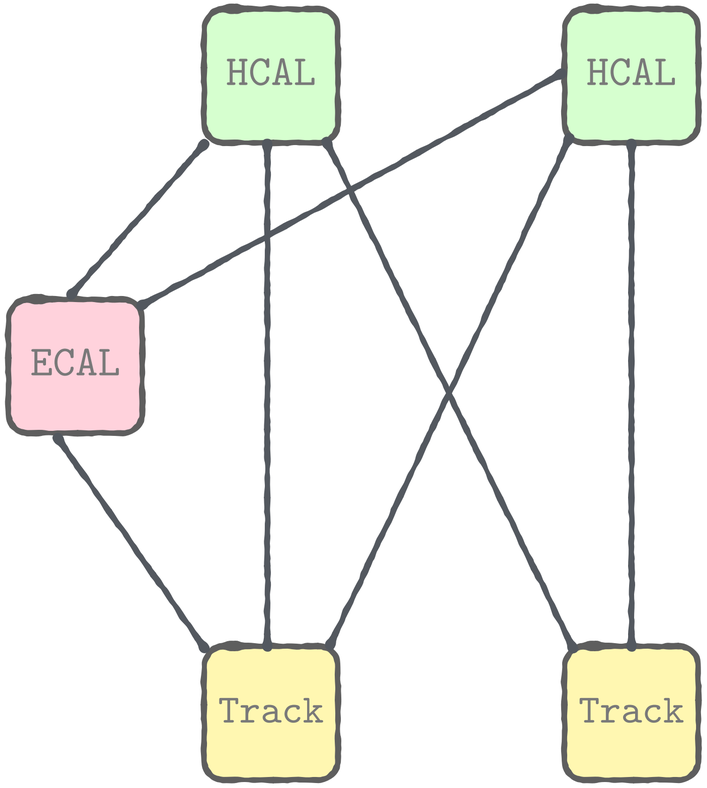
Quick A: Give precedence to photons!
Jet: O(25%) photons, O(10%) neutral hadrons

List of reconstructed (candidate) particles

{ γ, γ, γ

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$



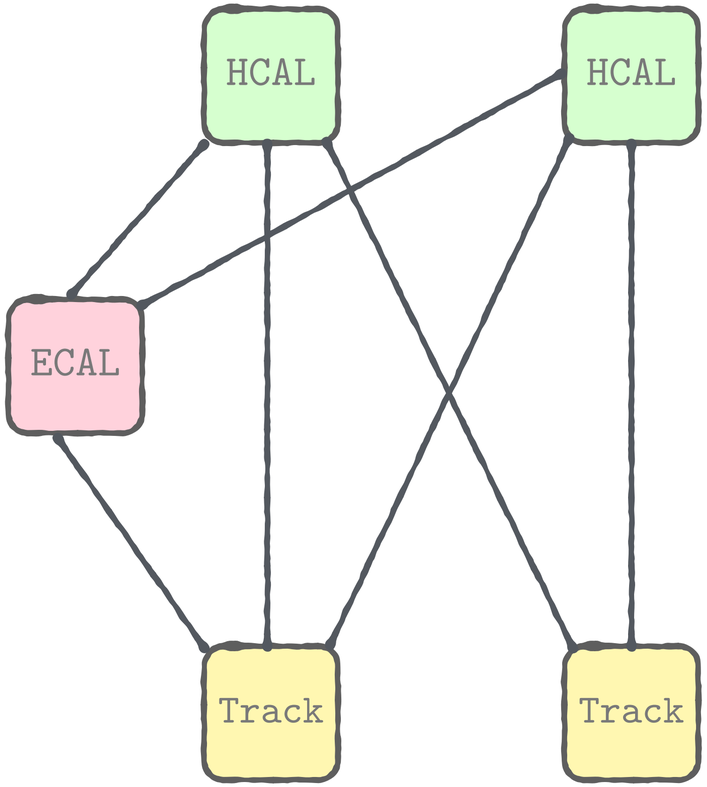
List of reconstructed (candidate) particles

{ γ, γ, γ

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (1st step)



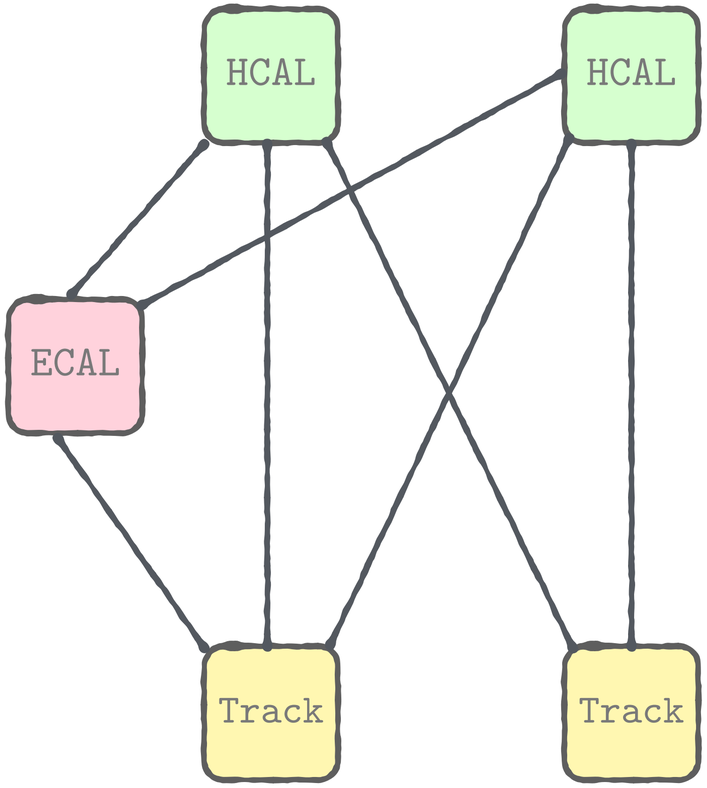
List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (1st step)



Optimise the use of HCAL granularity

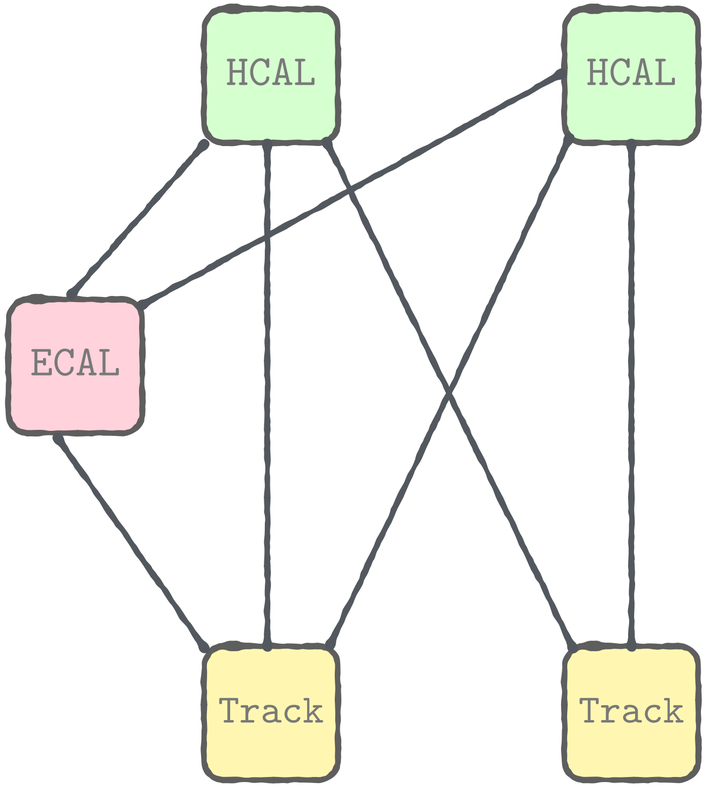
List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (1st step)



Optimise the use of HCAL granularity

Keep only the link to the closest cluster

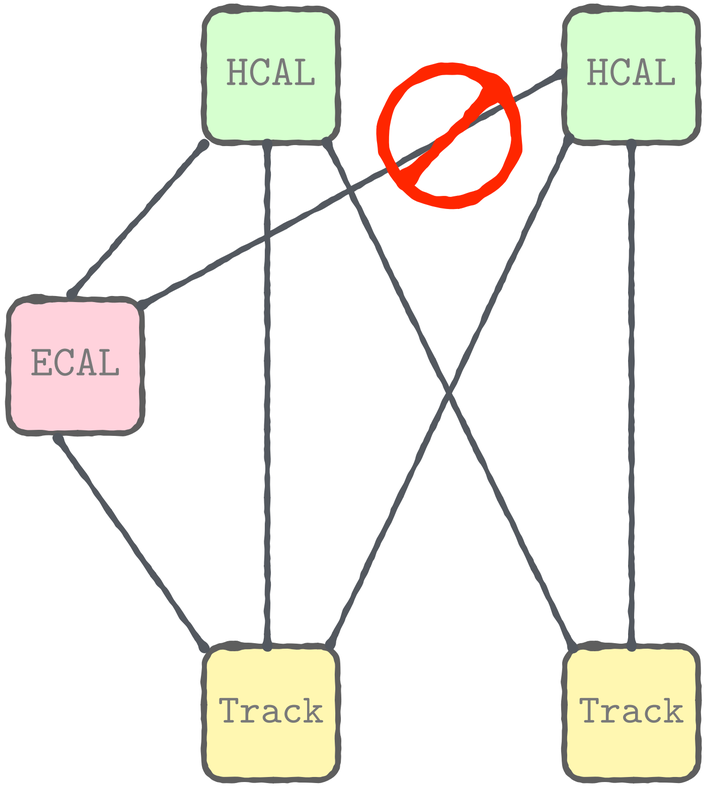
List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (1st step)



Optimise the use of HCAL granularity

Keep only the link to the closest cluster

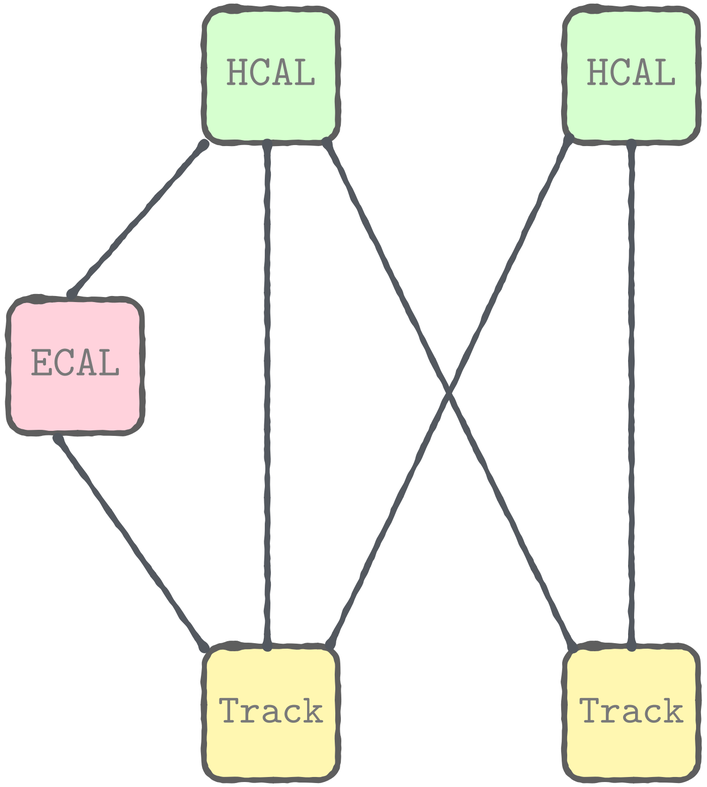
List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (1st step)



Optimise the use of HCAL granularity

Keep only the link to the closest cluster

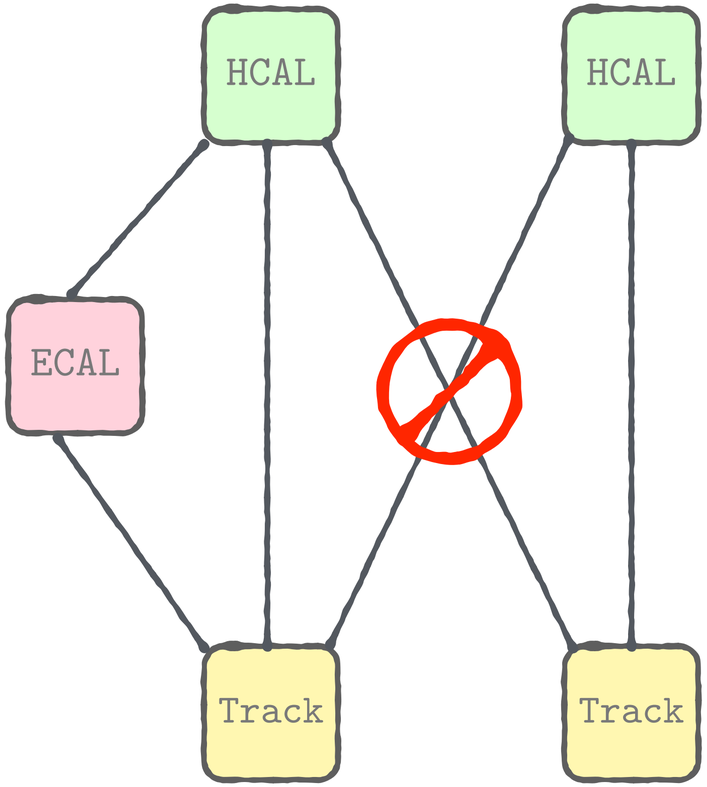
List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (1st step)



Optimise the use of HCAL granularity

Keep only the link to the closest cluster

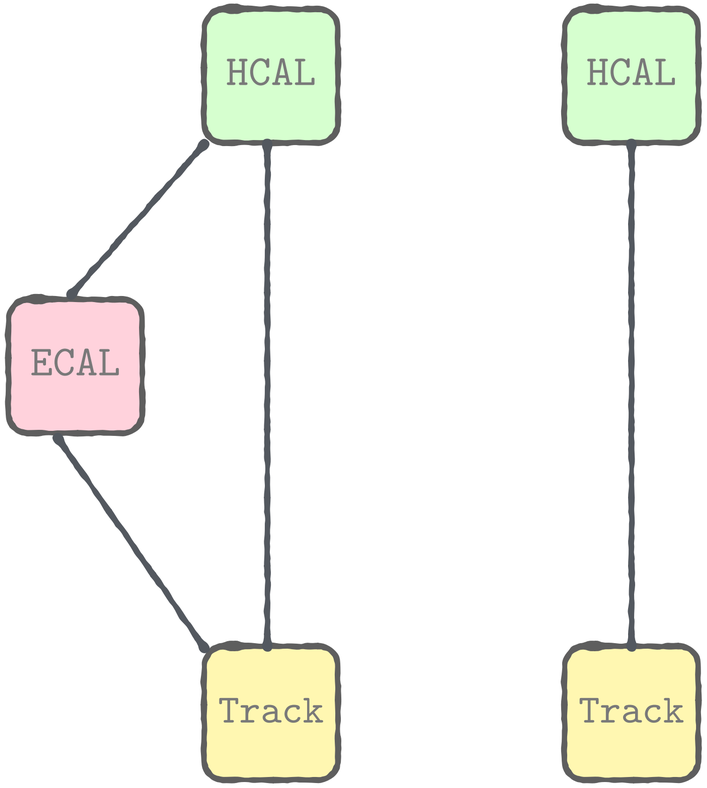
List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (1st step)



Optimise the use of HCAL granularity

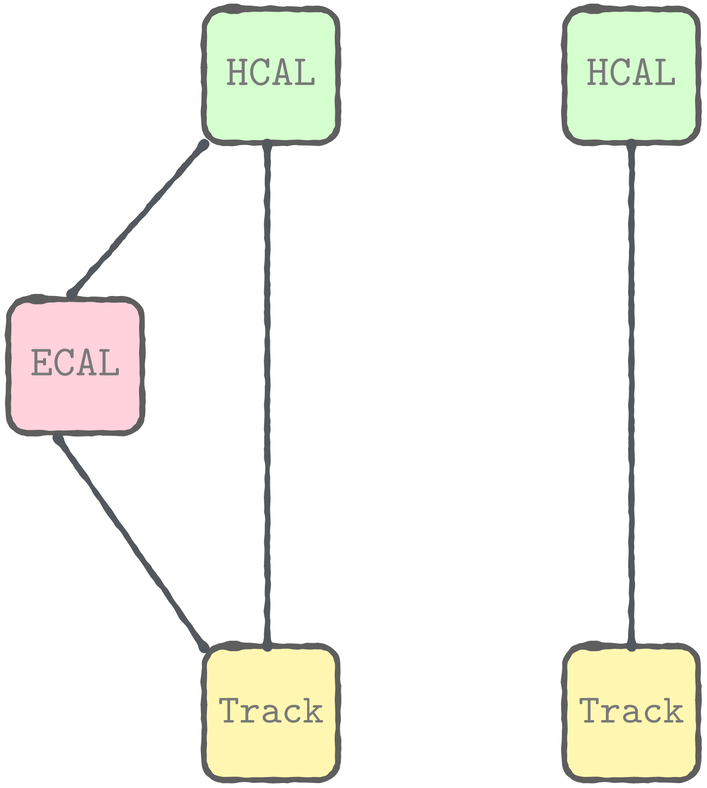
Keep only the link to the closest cluster

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma$

Four true particles:

$$\pi^+, \pi^-, \pi^0, K_L^0$$



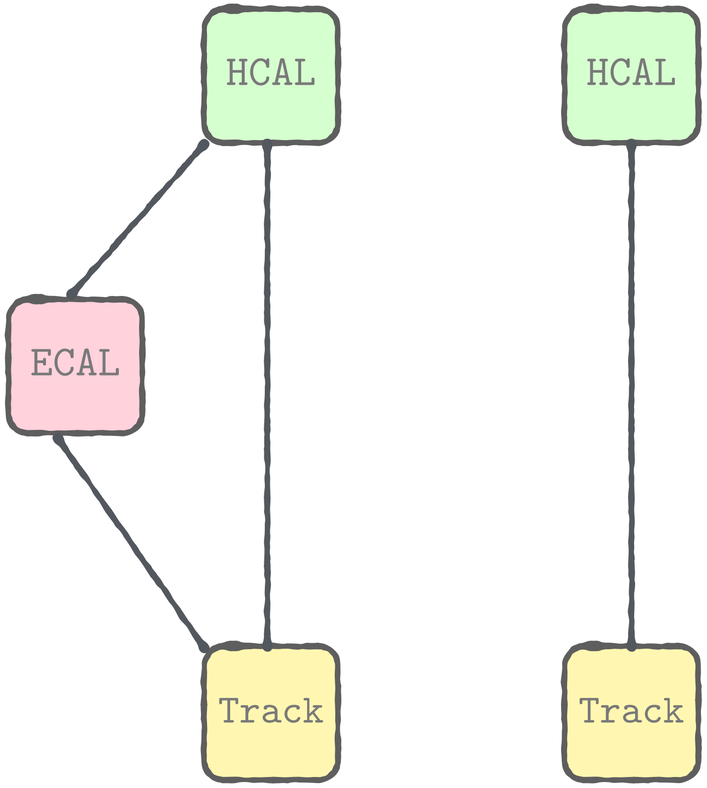
List of reconstructed (candidate) particles

{ γ, γ, γ

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)



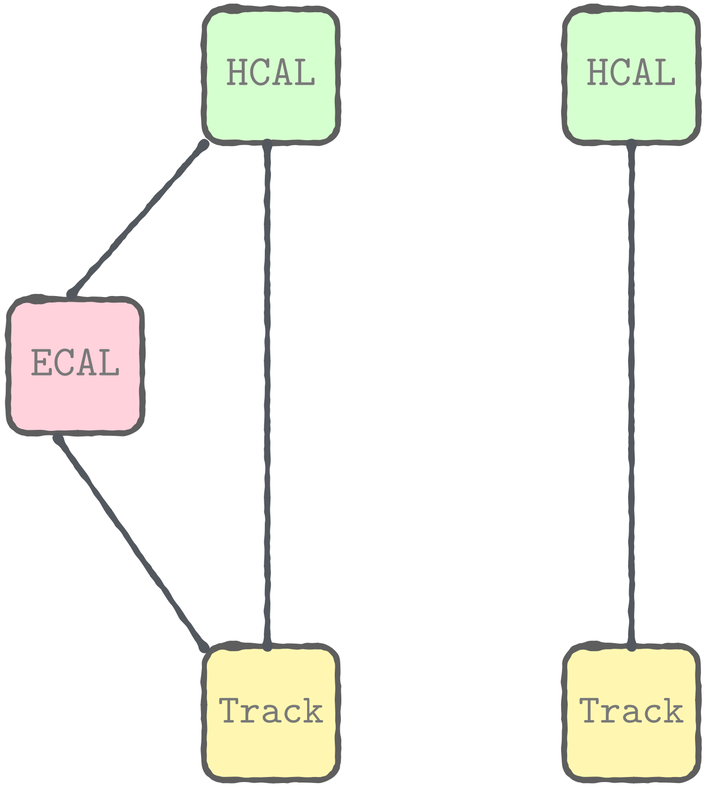
List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)



List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma$

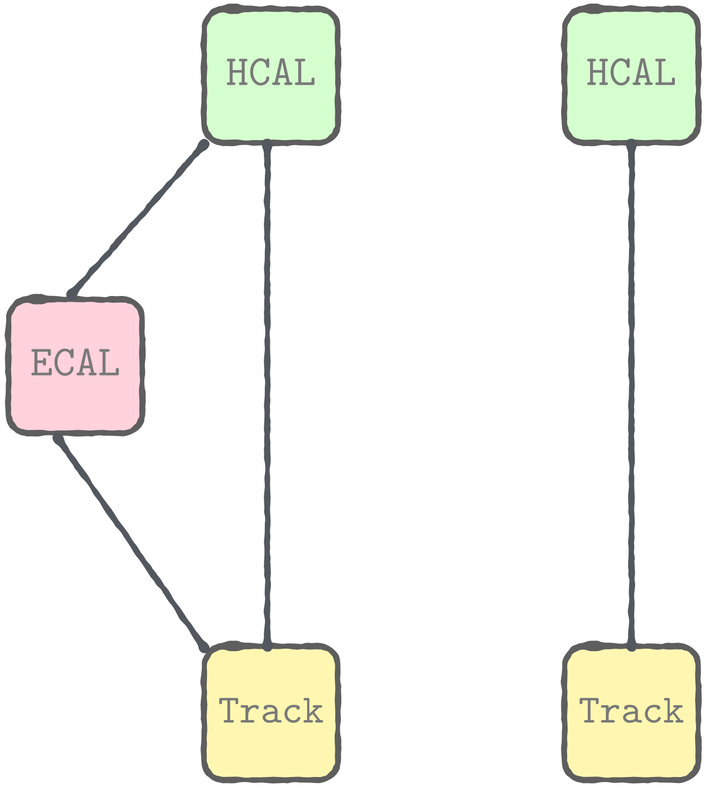
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



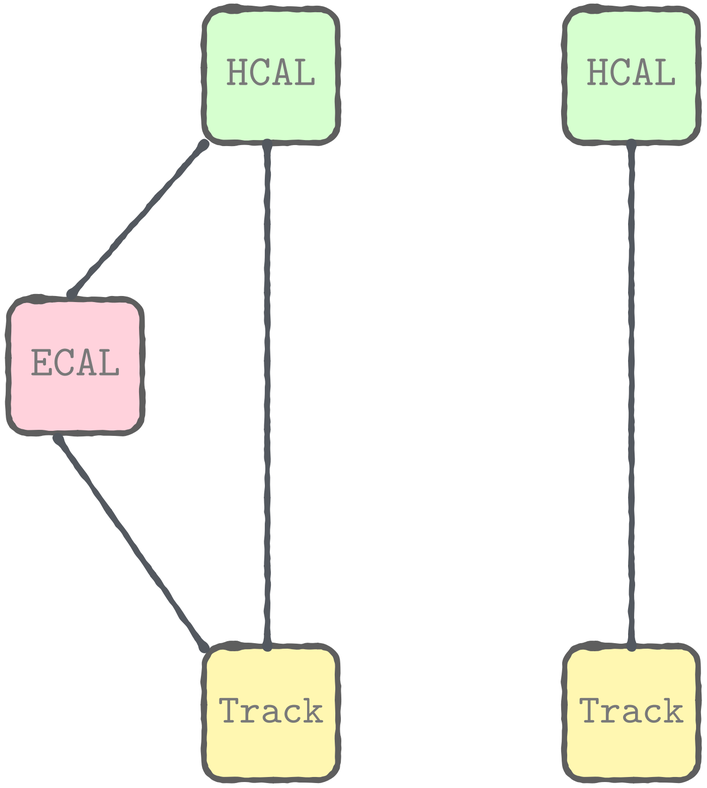
List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma$

(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$



Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons

For each HCAL Cluster compare:

- Sum of track momenta p
- Sum of cluster energies E
- linked to the tracks in ECAL & in HCAL "hadron calibrated" (see later)

List of reconstructed (candidate) particles

{ γ, γ, γ

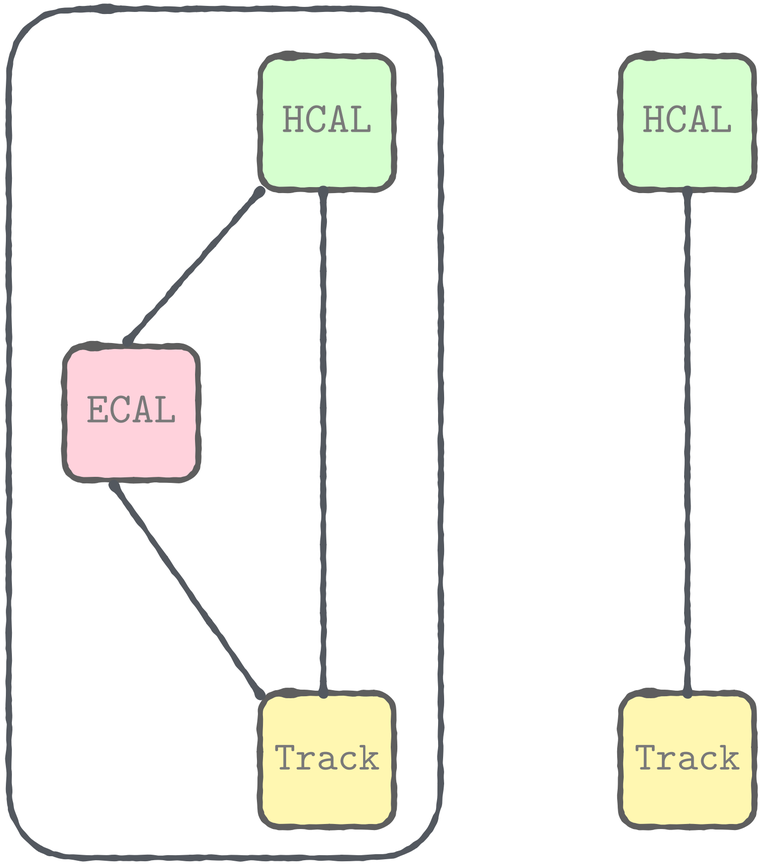
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum of track momenta p
- Sum of cluster energies E
- linked to the tracks in ECAL & in HCAL "hadron calibrated" (see later)

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma$

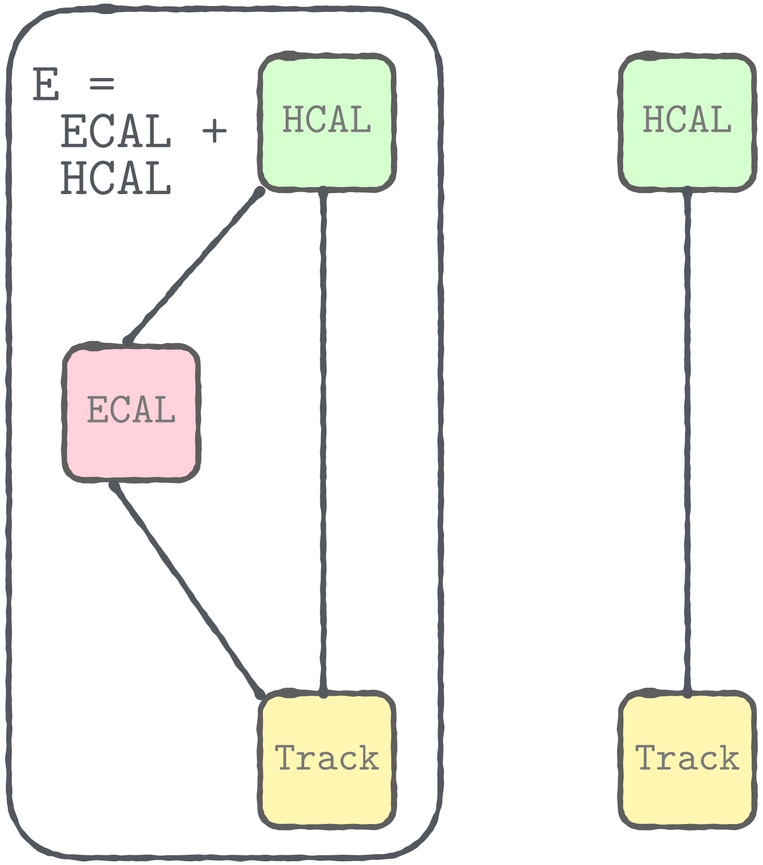
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum of track momenta p
- Sum of cluster energies E
- linked to the tracks in ECAL & in HCAL "hadron calibrated" (see later)

List of reconstructed (candidate) particles

{ γ, γ, γ

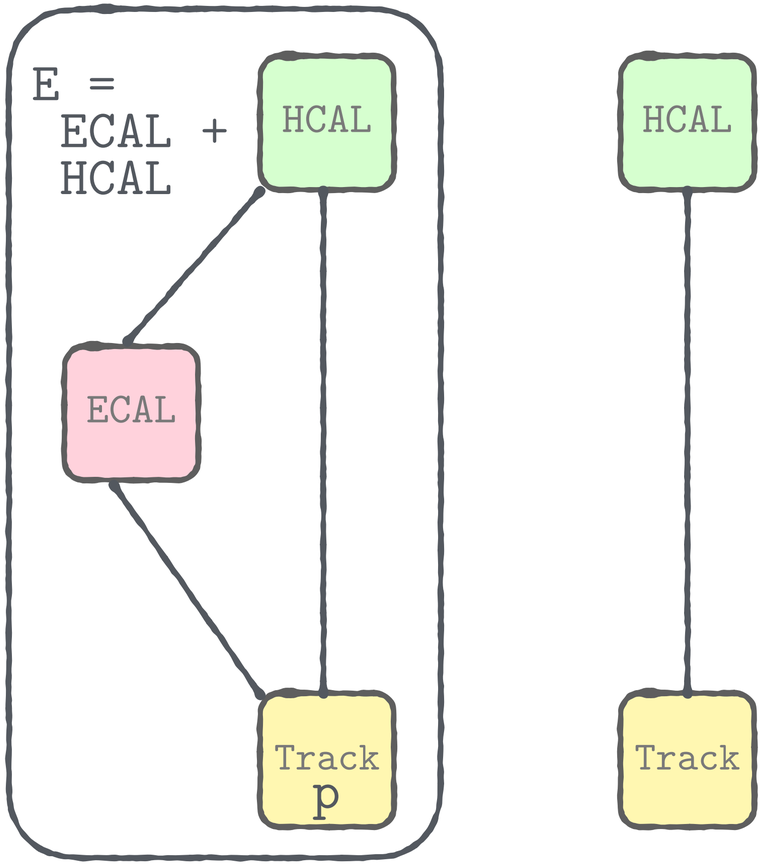
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum of track momenta p
- Sum of cluster energies E
- linked to the tracks in ECAL & in HCAL "hadron calibrated" (see later)

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma$

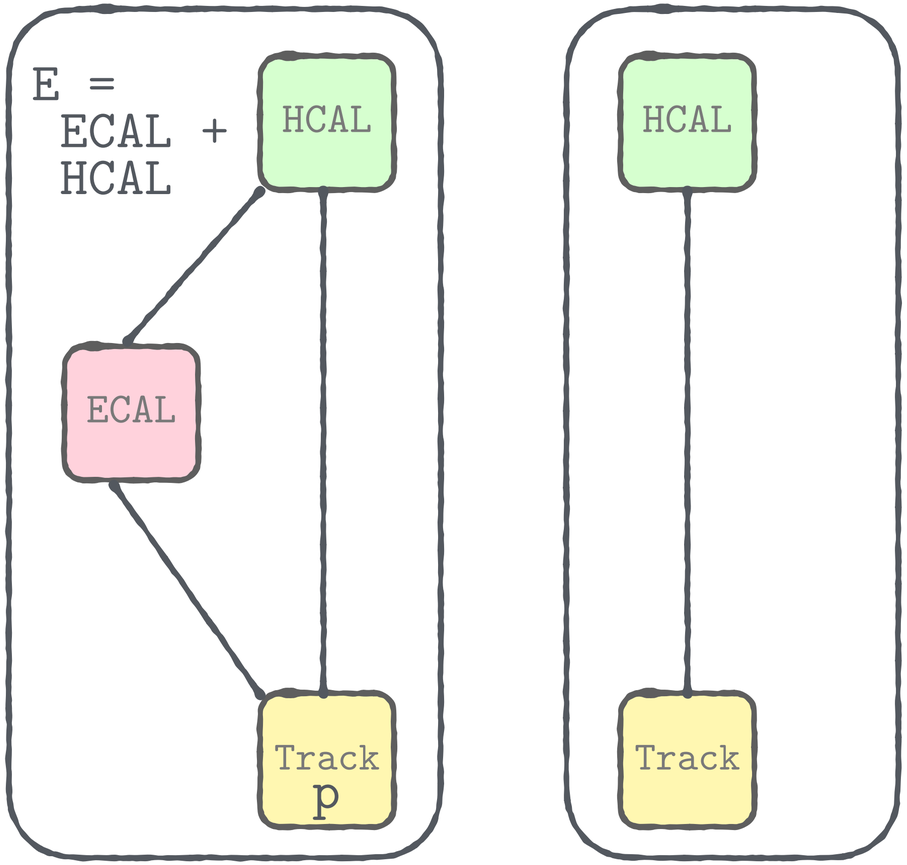
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum of track momenta p
- Sum of cluster energies E
- linked to the tracks in ECAL & in HCAL "hadron calibrated" (see later)

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma$

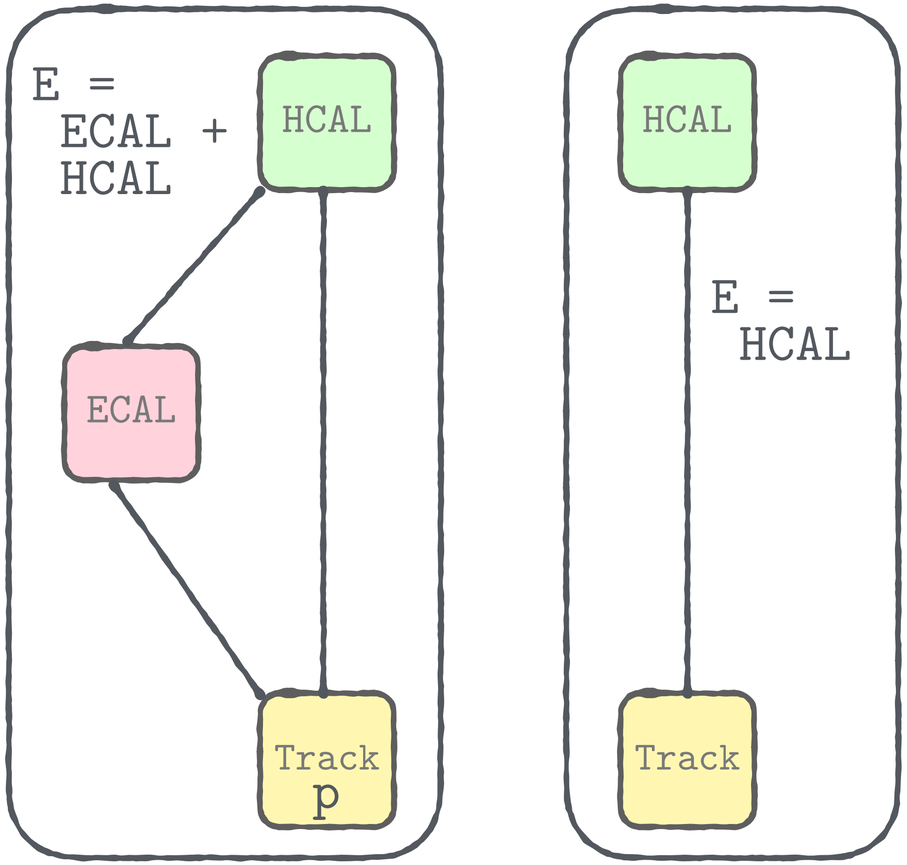
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum of track momenta p
- Sum of cluster energies E
- linked to the tracks in ECAL & in HCAL "hadron calibrated" (see later)

List of reconstructed (candidate) particles

{ γ, γ, γ

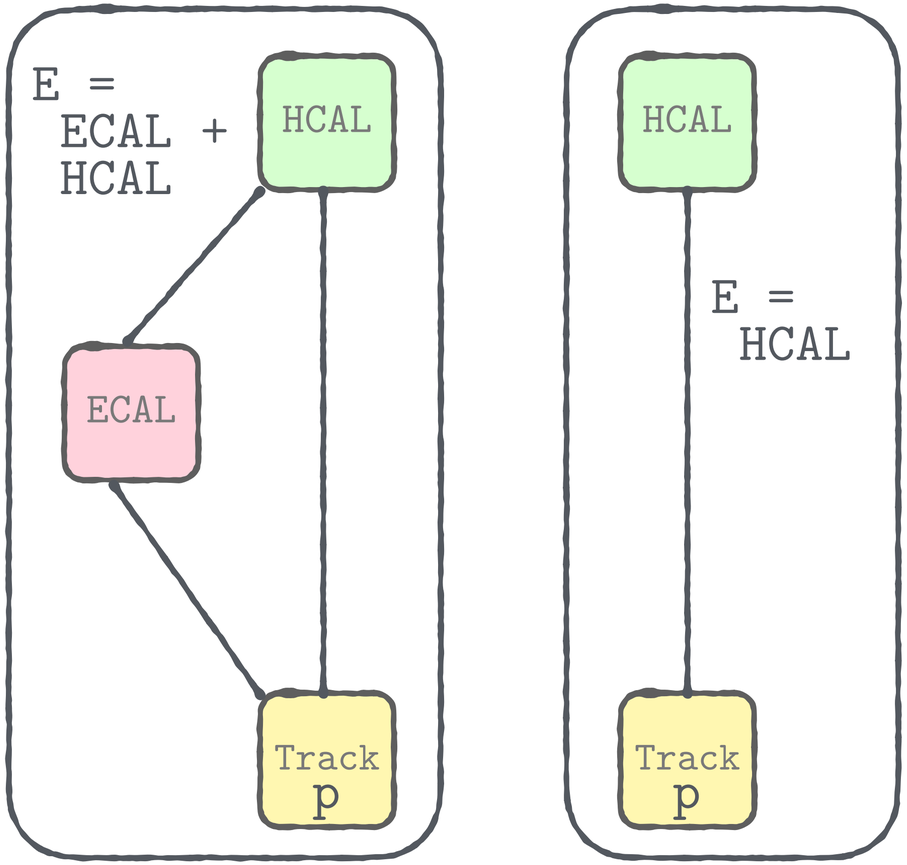
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum of track momenta p
- Sum of cluster energies E
- linked to the tracks in ECAL & in HCAL "hadron calibrated" (see later)

List of reconstructed (candidate) particles

{ γ, γ, γ

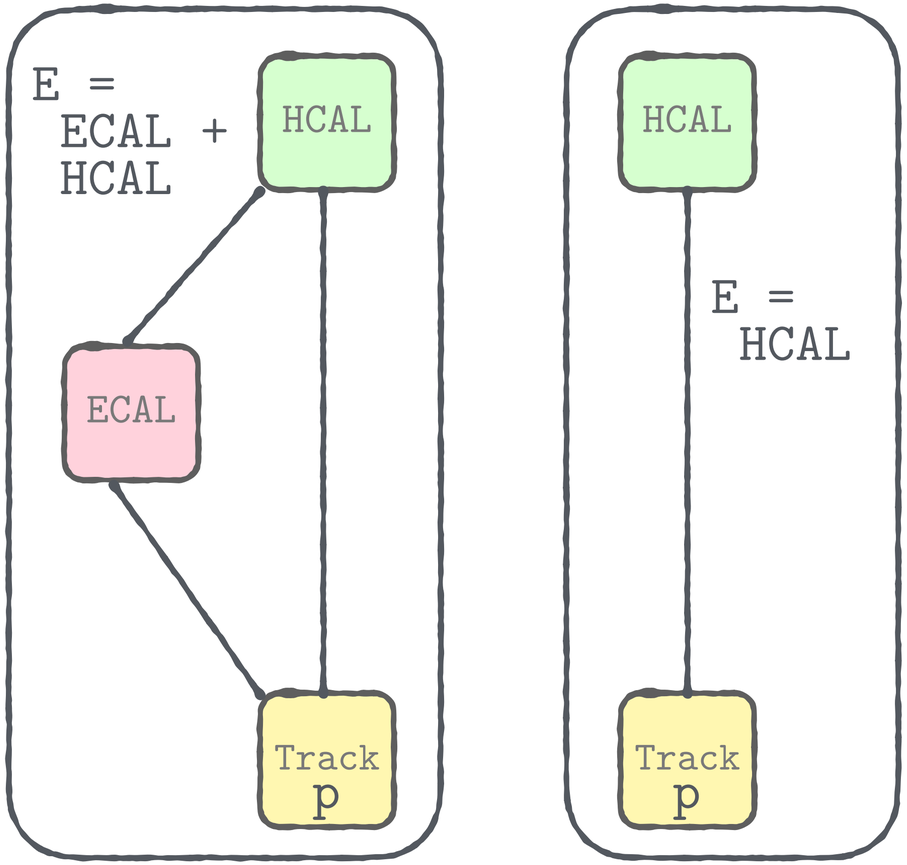
(blocks are usually very small)

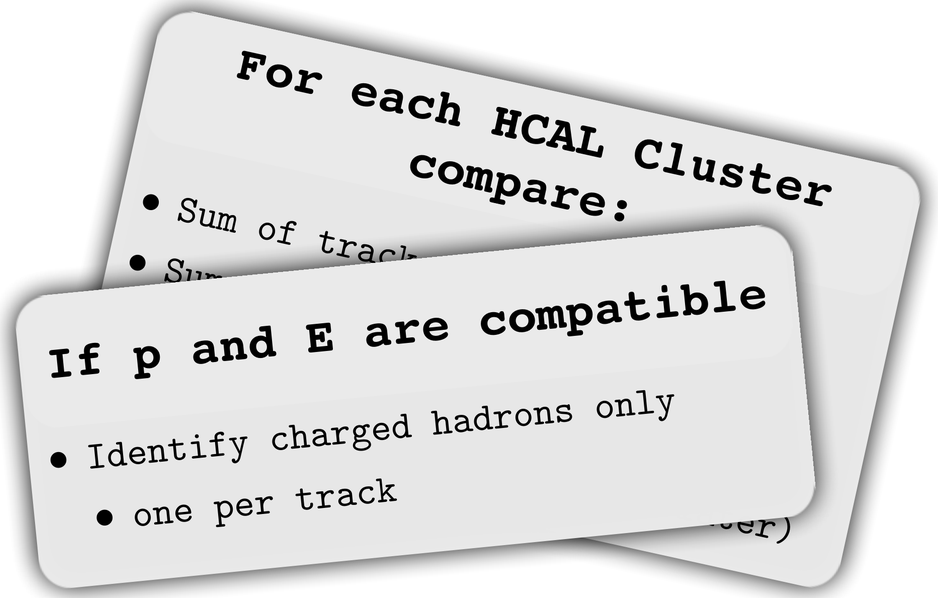
Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons





List of reconstructed (candidate) particles

{ γ, γ, γ

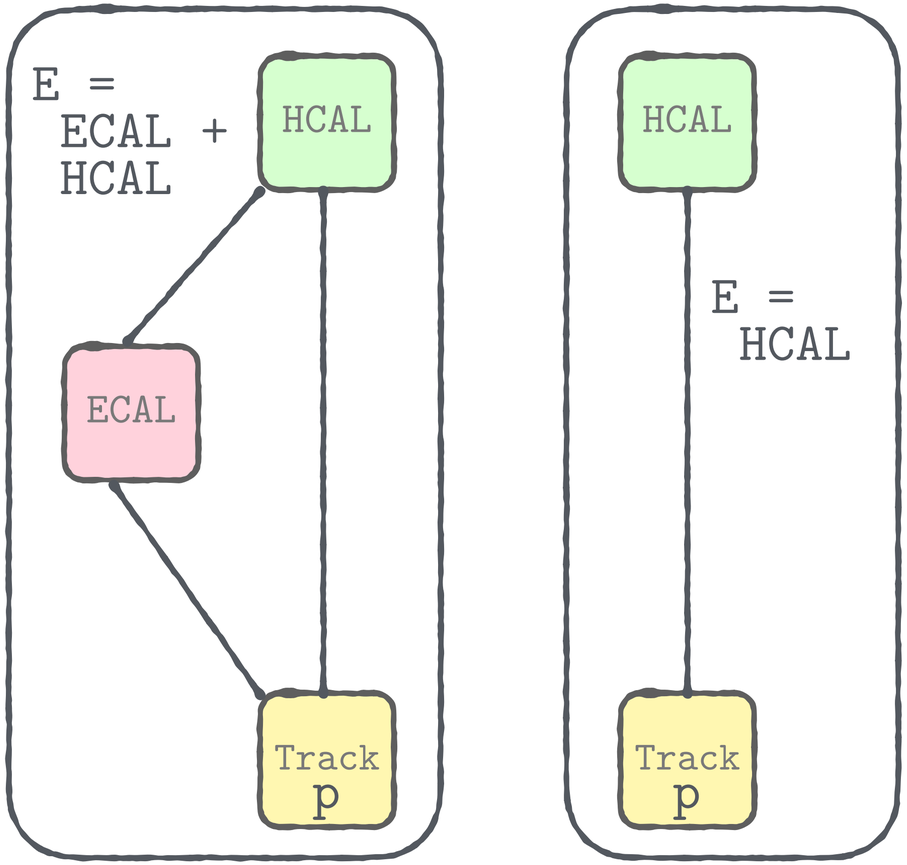
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum of track
- Su

If p and E are compatible

- Identify charged hadrons only
  - one per track

If E > p + σ(E)

- Identify charged hadrons + photons / neutral hadrons

List of reconstructed (candidate) particles

{ γ, γ, γ

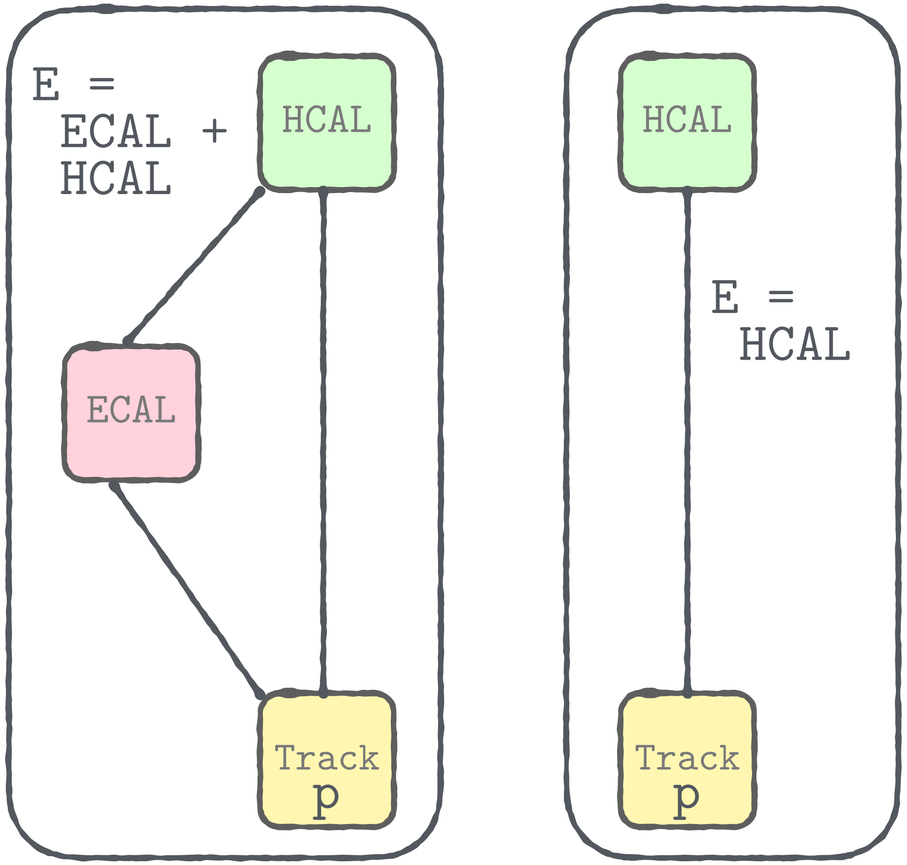
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum of track
- Su

If p and E are compatible

- Identify charged hadrons only
  - one per track

If E > p + σ(E)

- Identify charged hadrons + photons / neutral hadrons

If E << p

- Something odd going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ

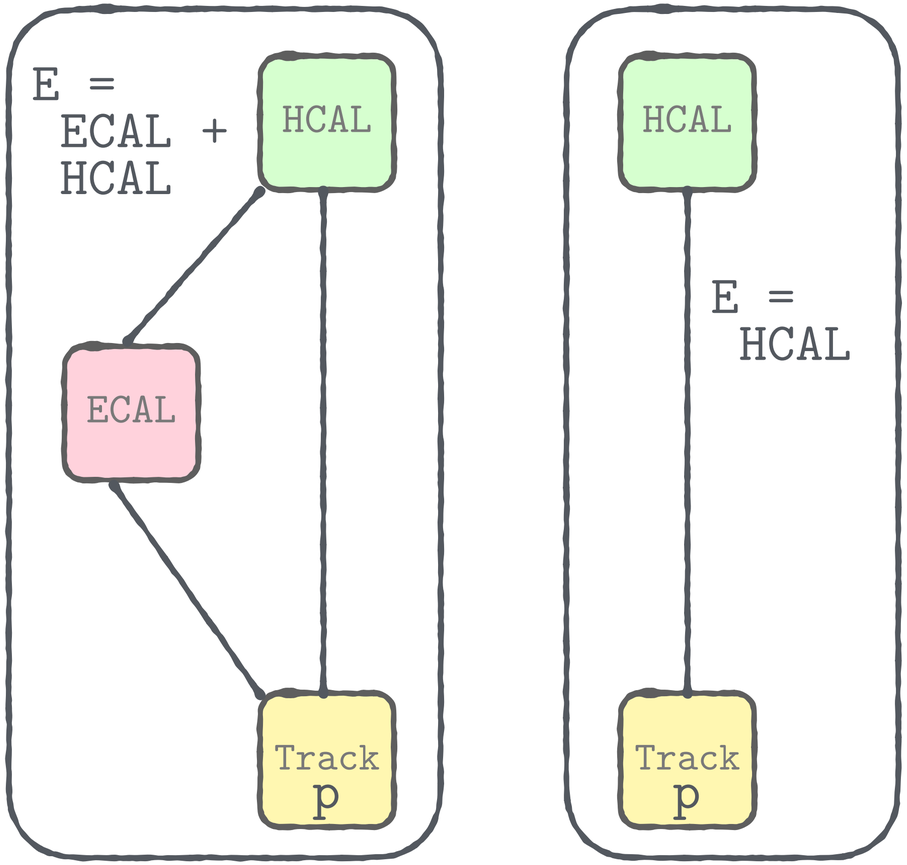
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum of track
- Su

If p and E are compatible

- Identify charged hadrons only
  - one per track

If E << p

- Something odd going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

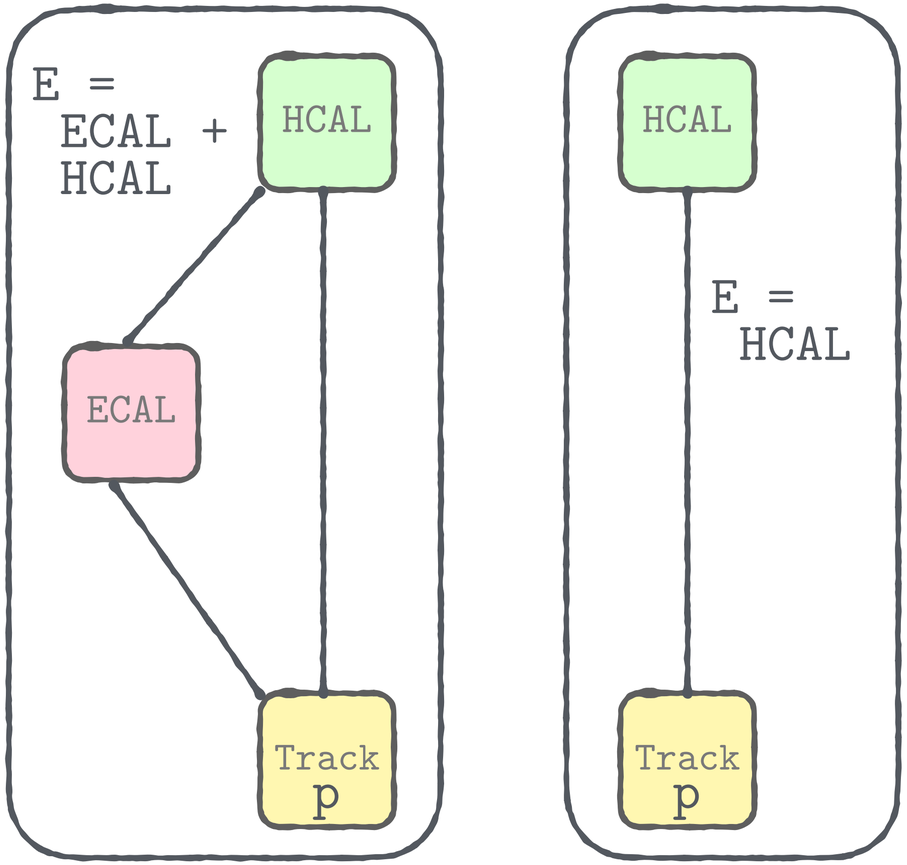
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs
attention (doesn't happen often)

List of reconstructed (candidate) particles

{ $\gamma, \gamma, \gamma,$

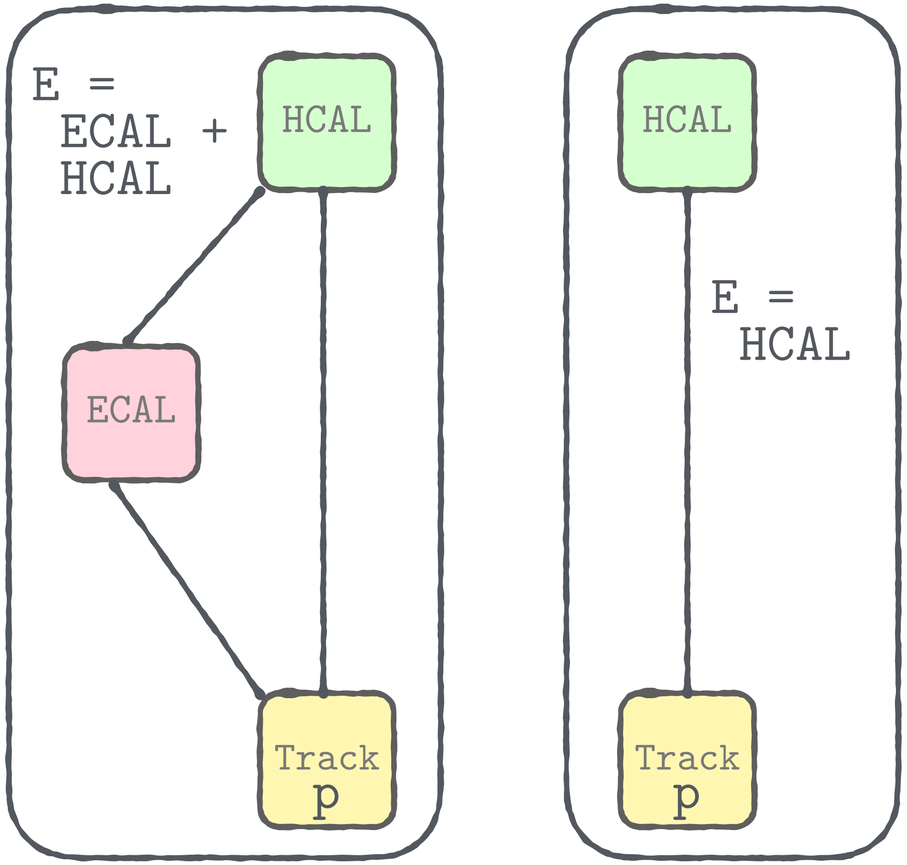
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs
attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

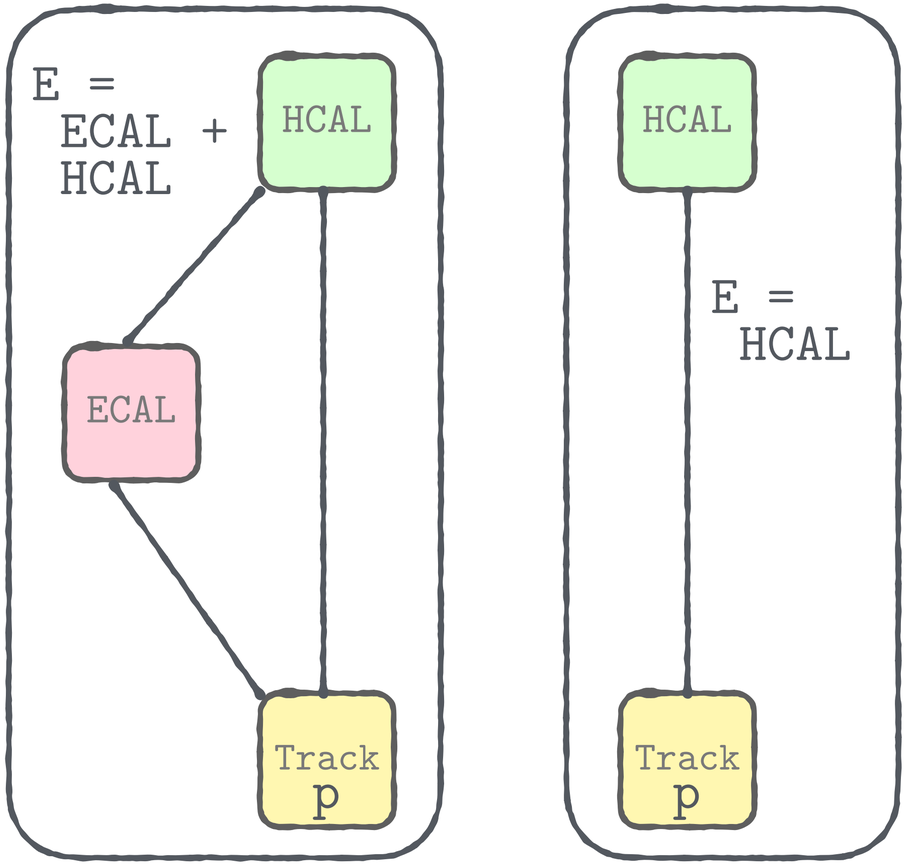
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

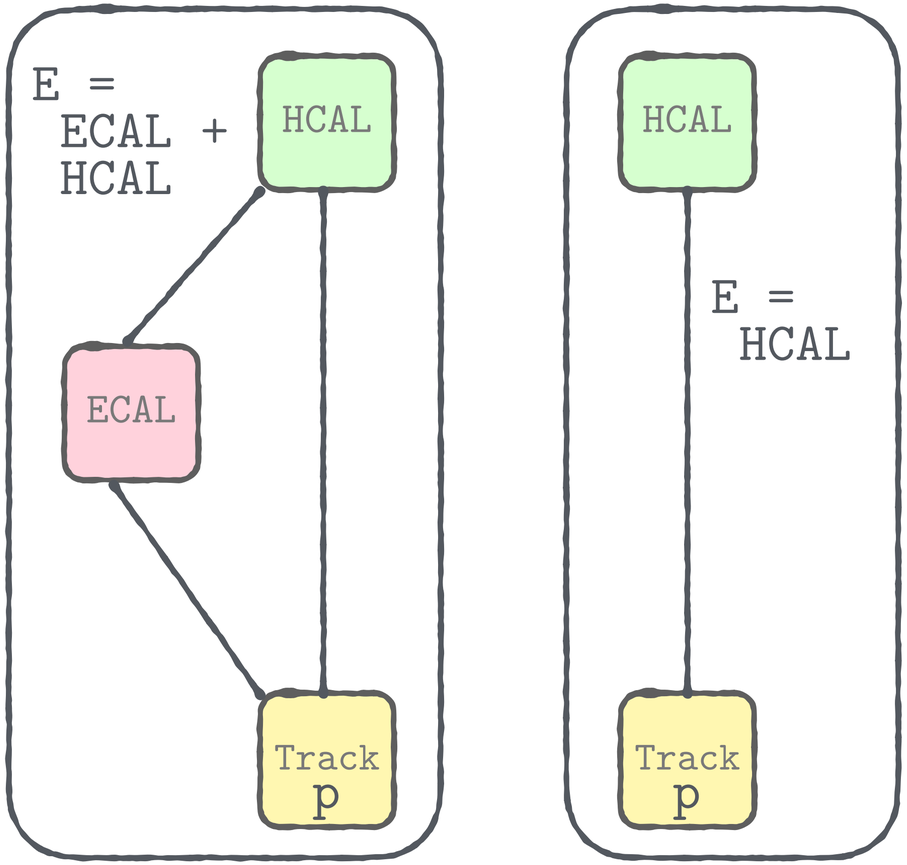
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

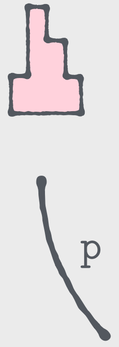
- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b



attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

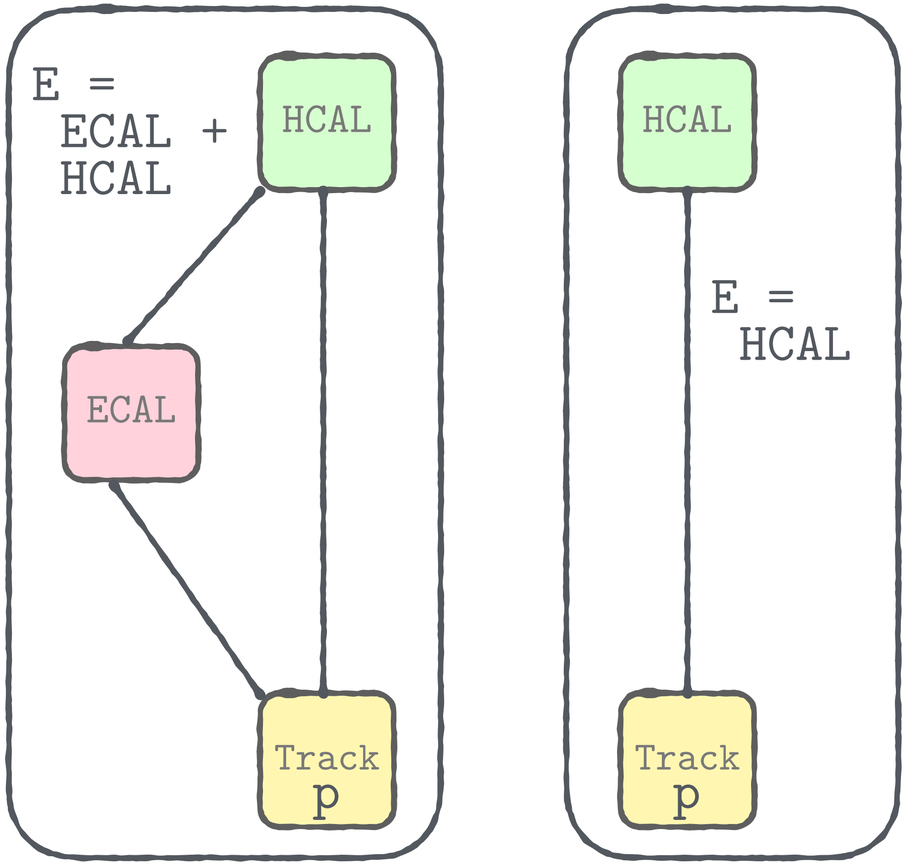
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

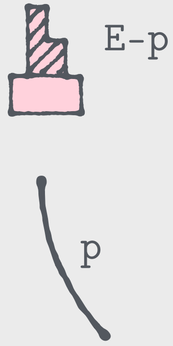
- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b



attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

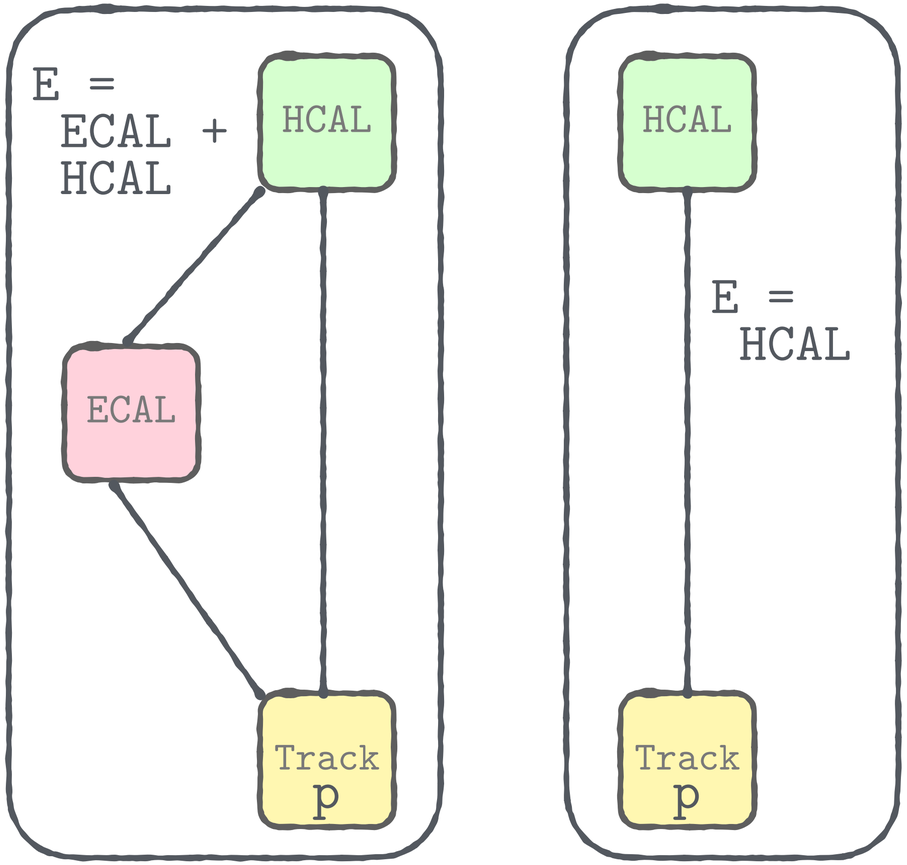
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

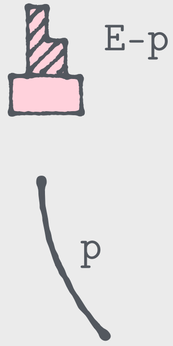
- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b



attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ $\gamma, \gamma, \gamma, \pi^-$

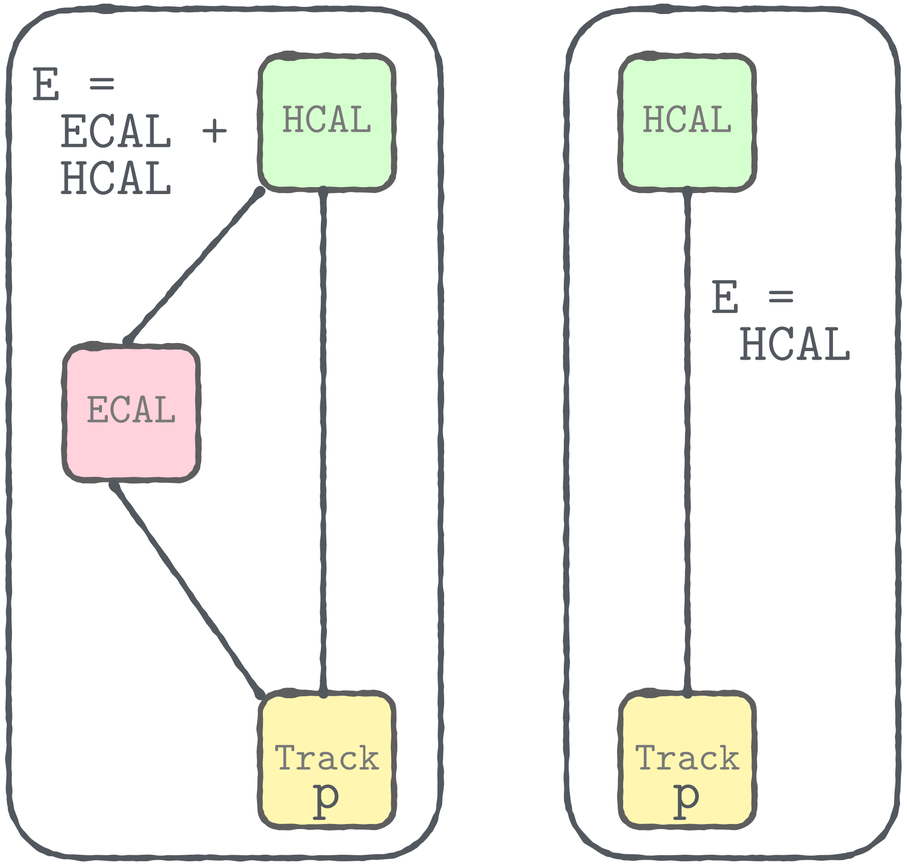
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

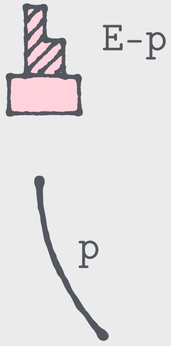
- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b



attention (doesn't happen often)

List of reconstructed (candidate) particles

{ $\gamma, \gamma, \gamma, \pi^-$,

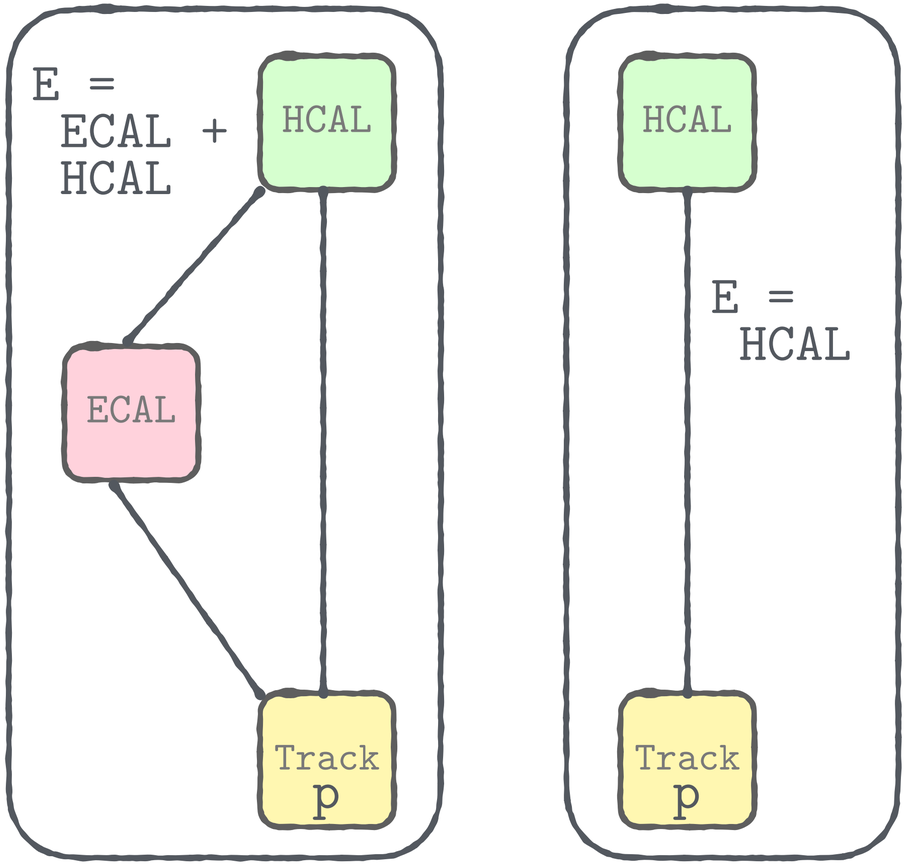
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

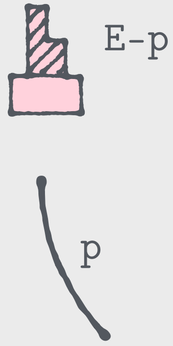
- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b



attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

$\{\gamma, \gamma, \gamma, \pi^-, \gamma$

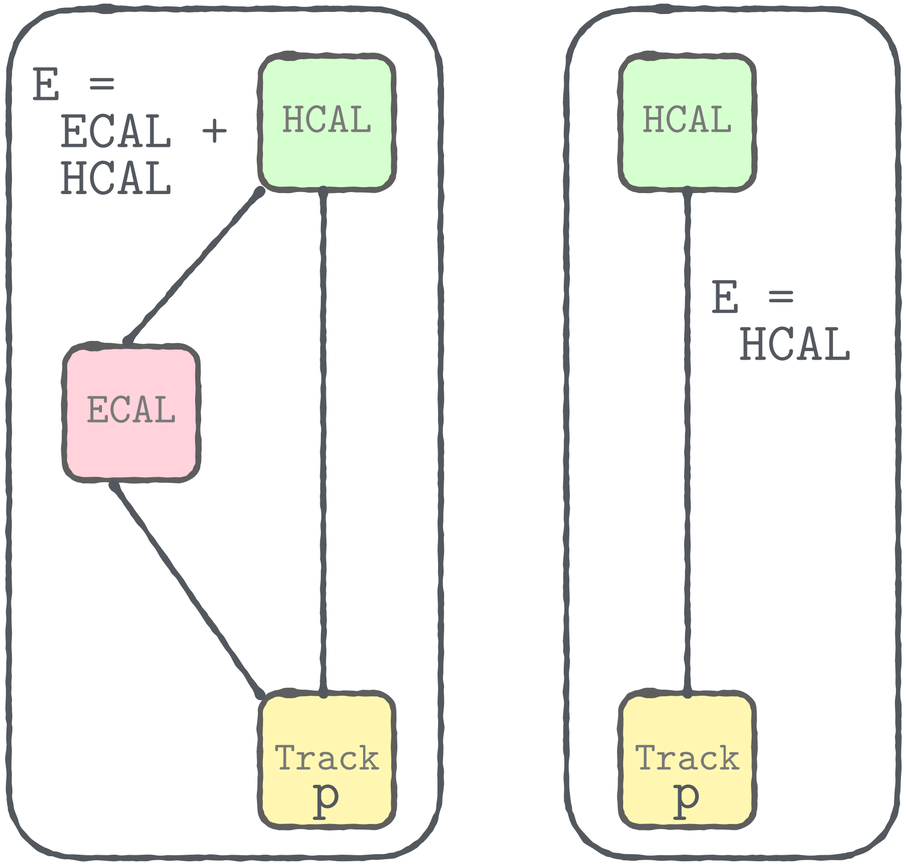
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

p

...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

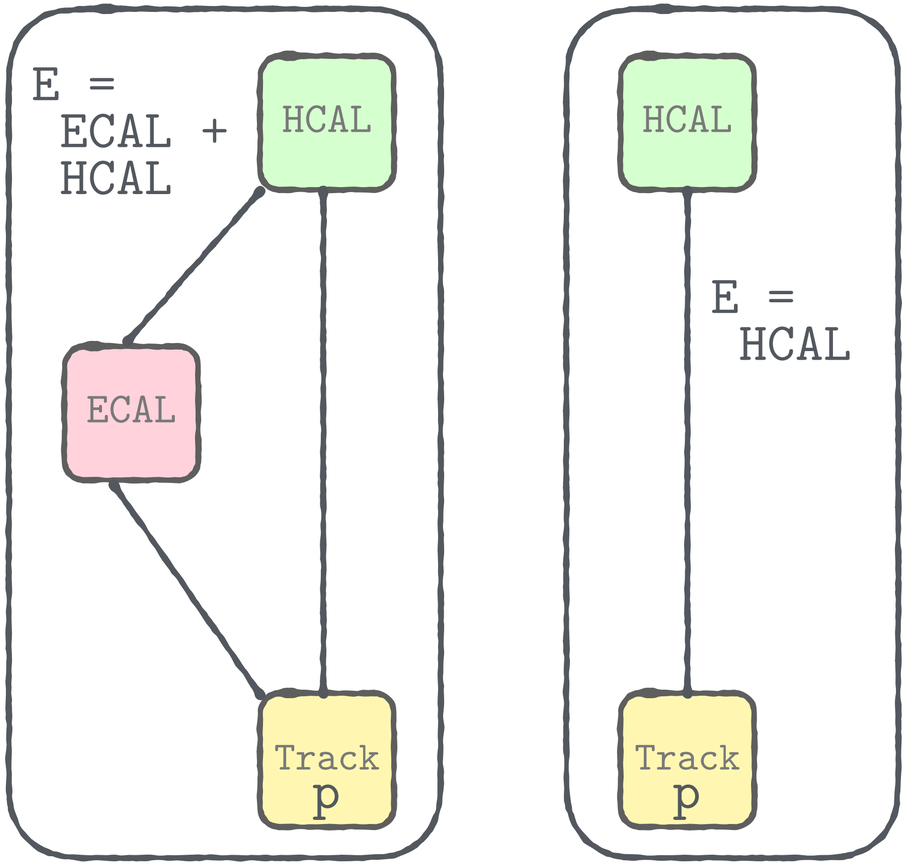
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

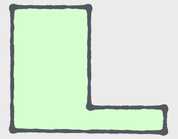
- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b



attention (...going on...needs
attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

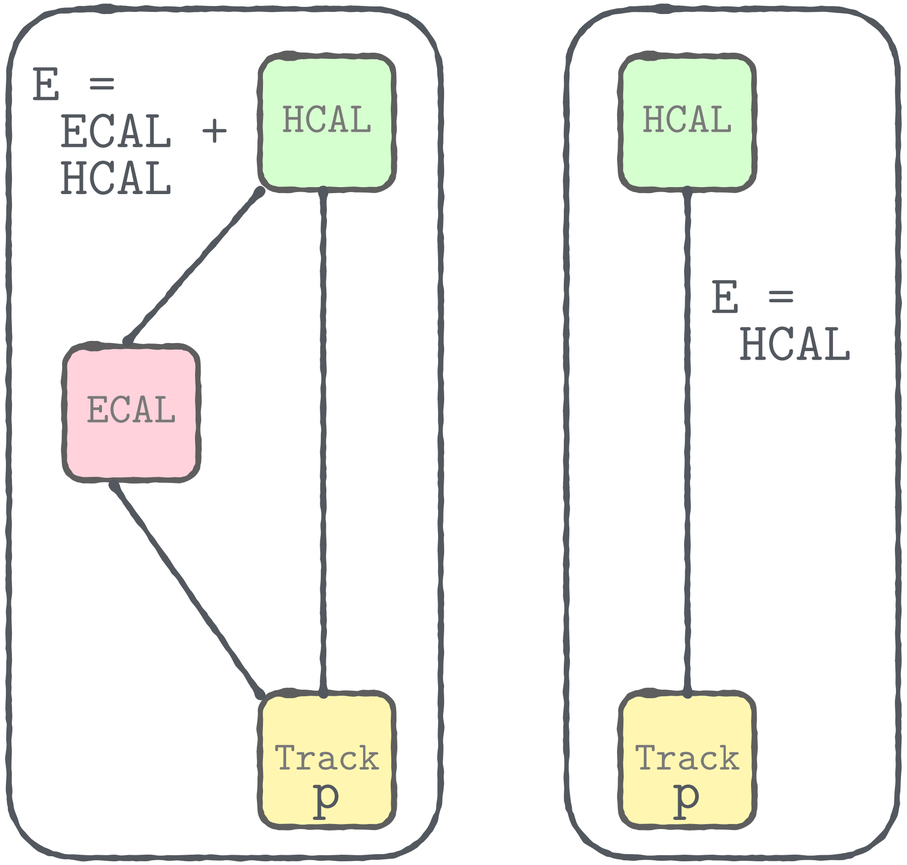
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs
attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

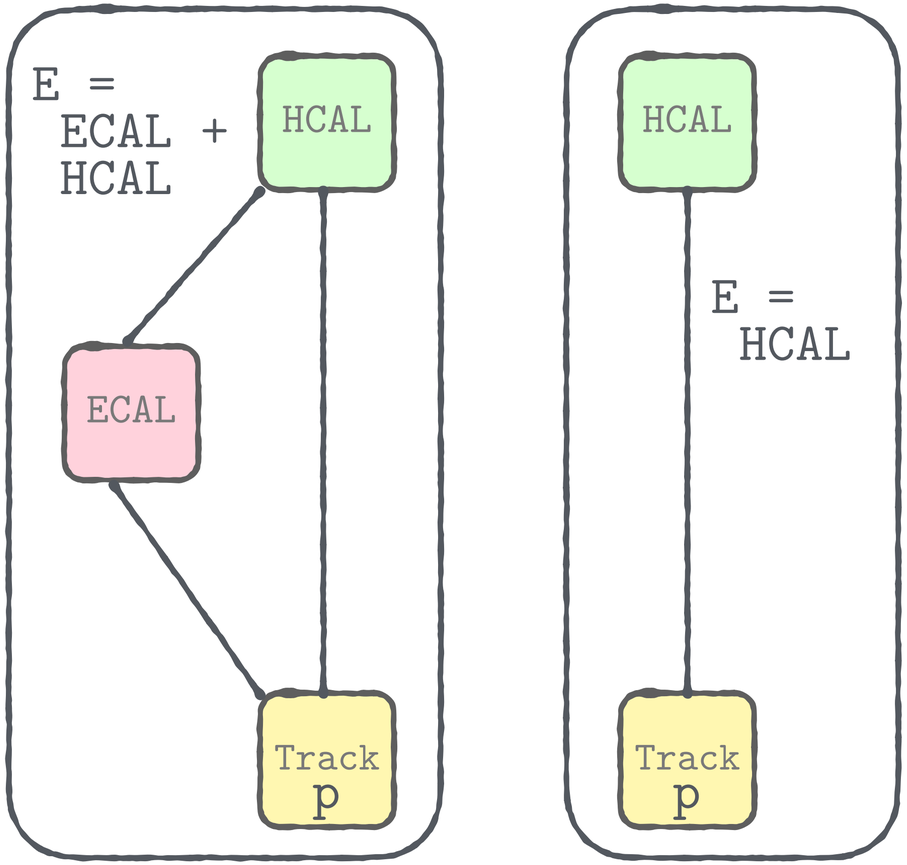
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs
attention (doesn't happen often)

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^- \}$

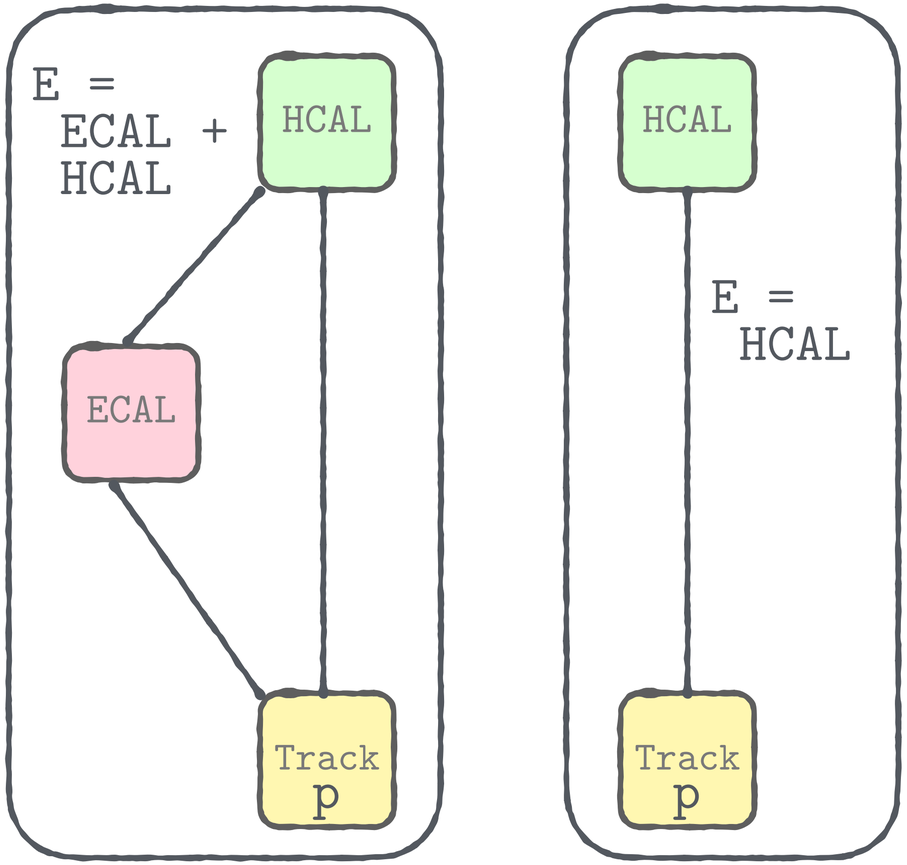
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs
attention (doesn't happen often)

List of reconstructed (candidate) particles

{ $\gamma, \gamma, \gamma, \pi^-$,

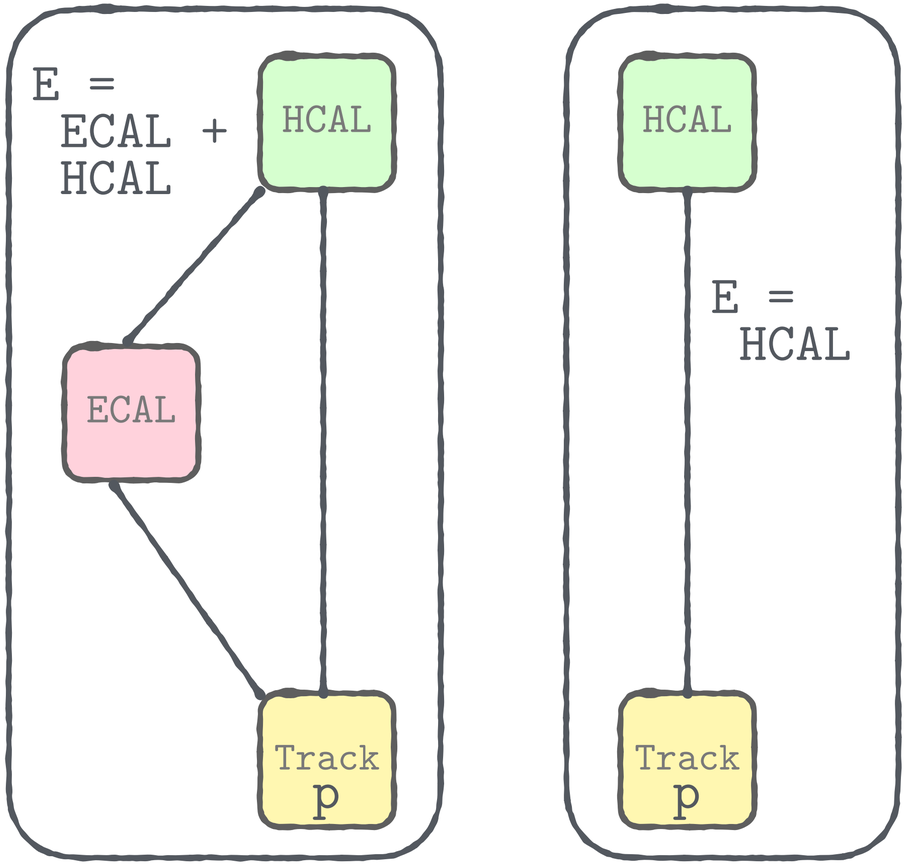
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs
attention (doesn't happen often)

List of reconstructed (candidate) particles

{ $\gamma, \gamma, \gamma, \pi^-, \mathrm{K}_\mathrm{L}^0$

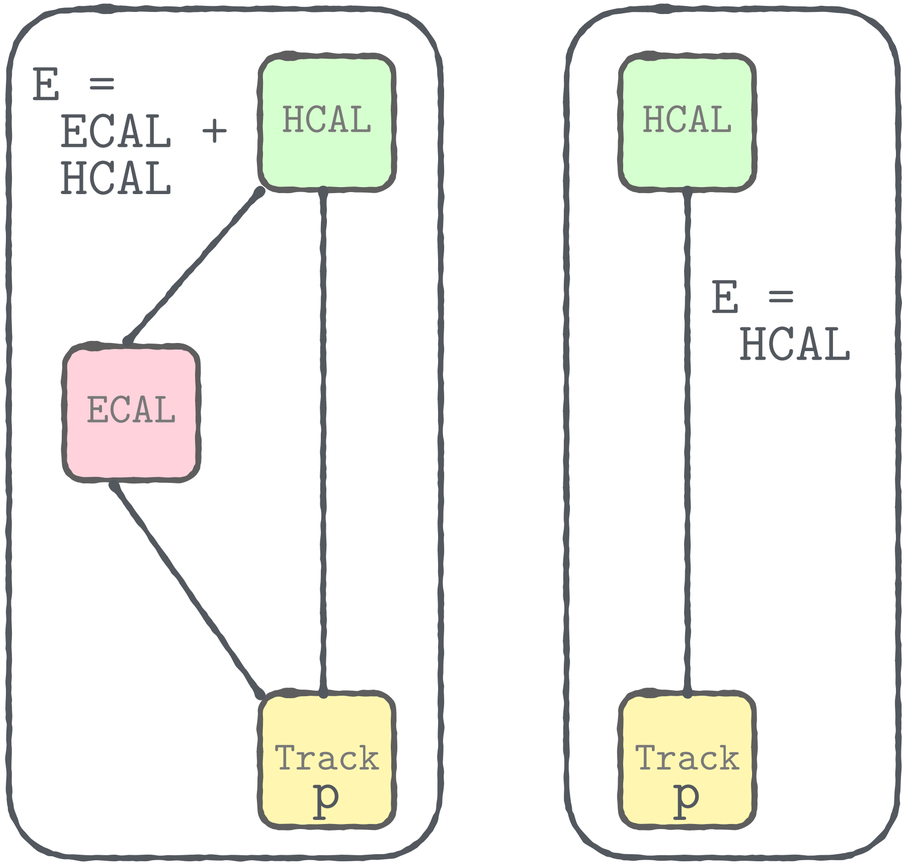
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

p

...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

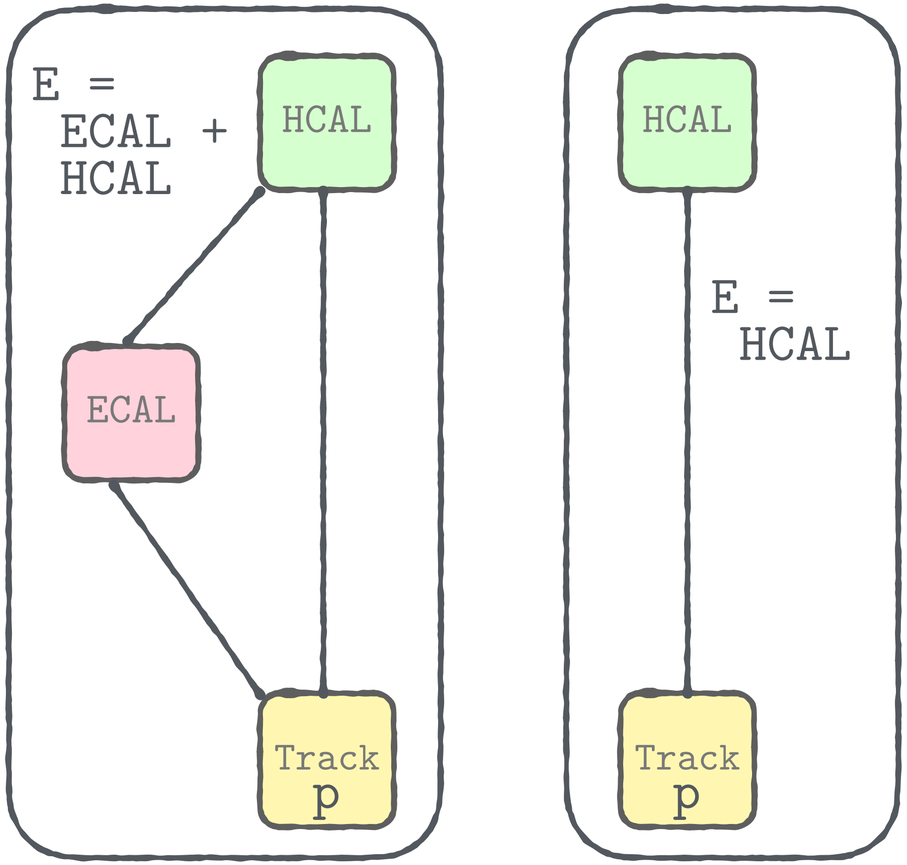
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

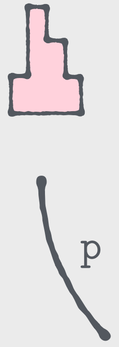
- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b



attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

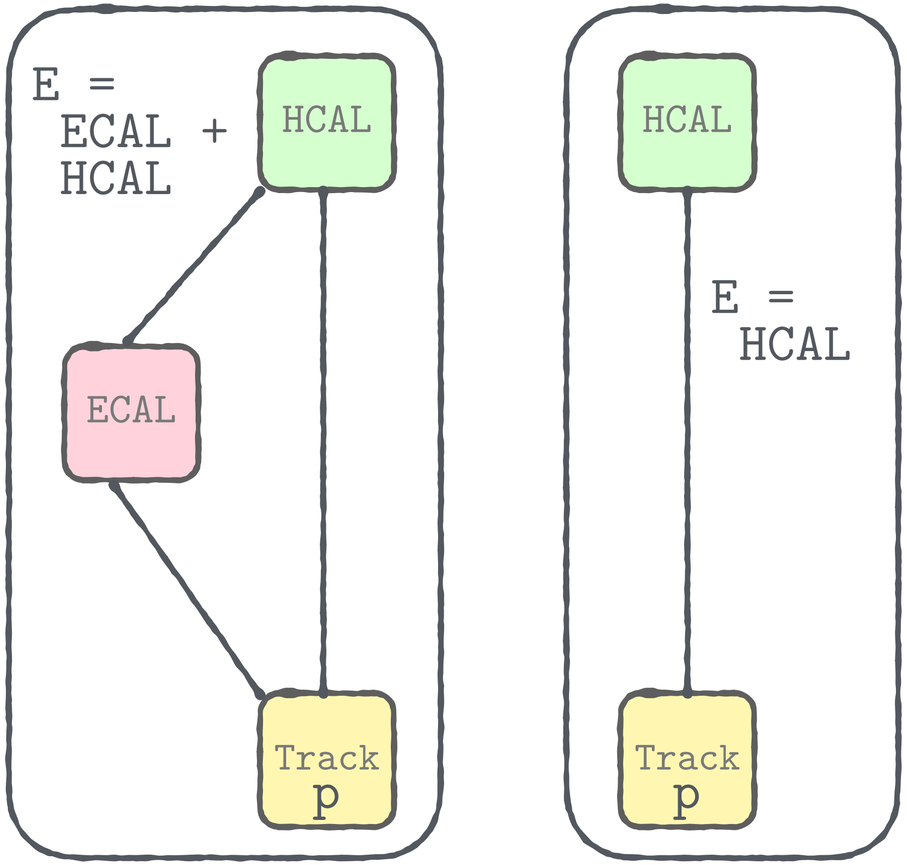
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

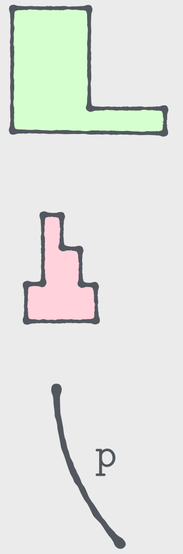
- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b



attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

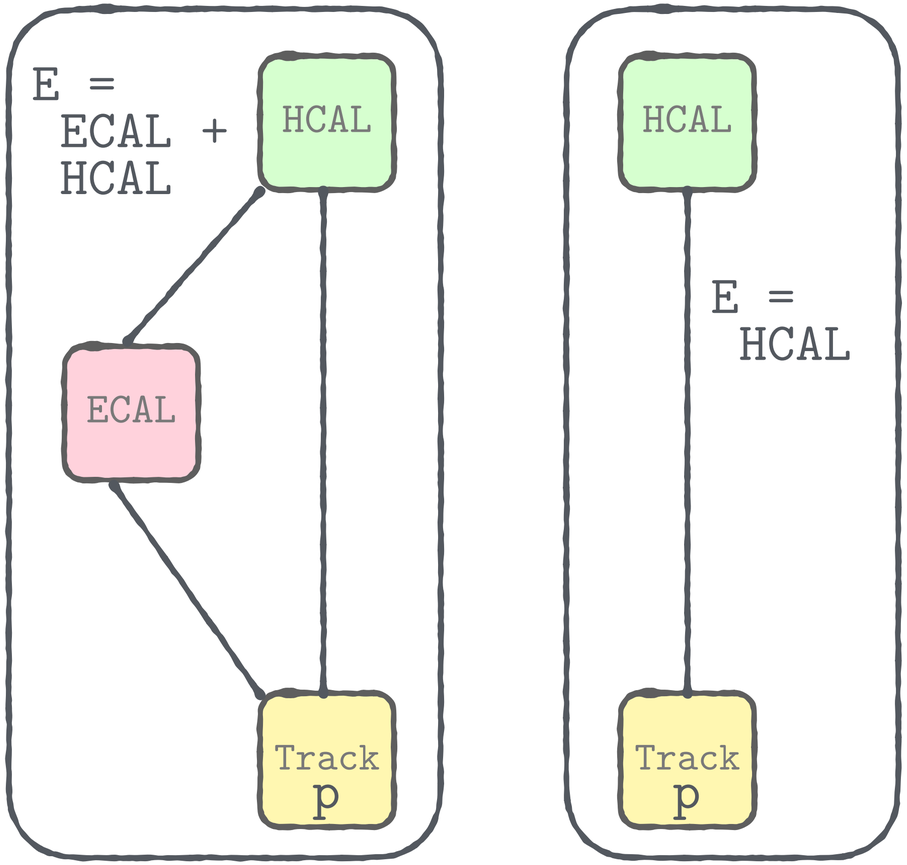
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

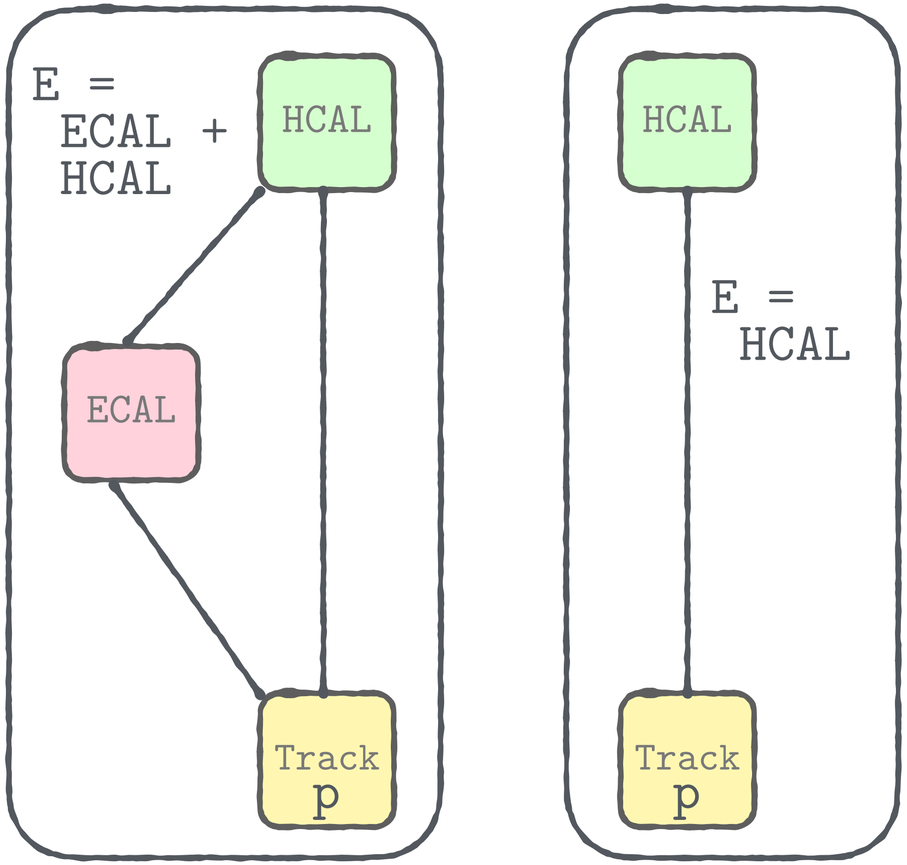
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

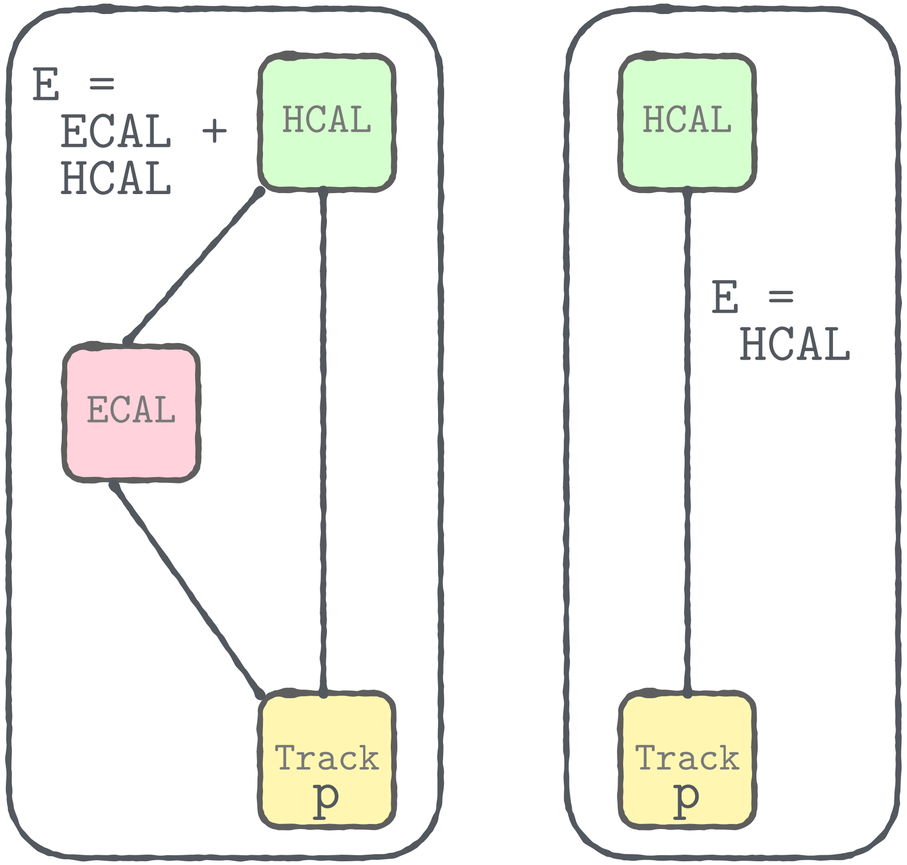
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^- \}$

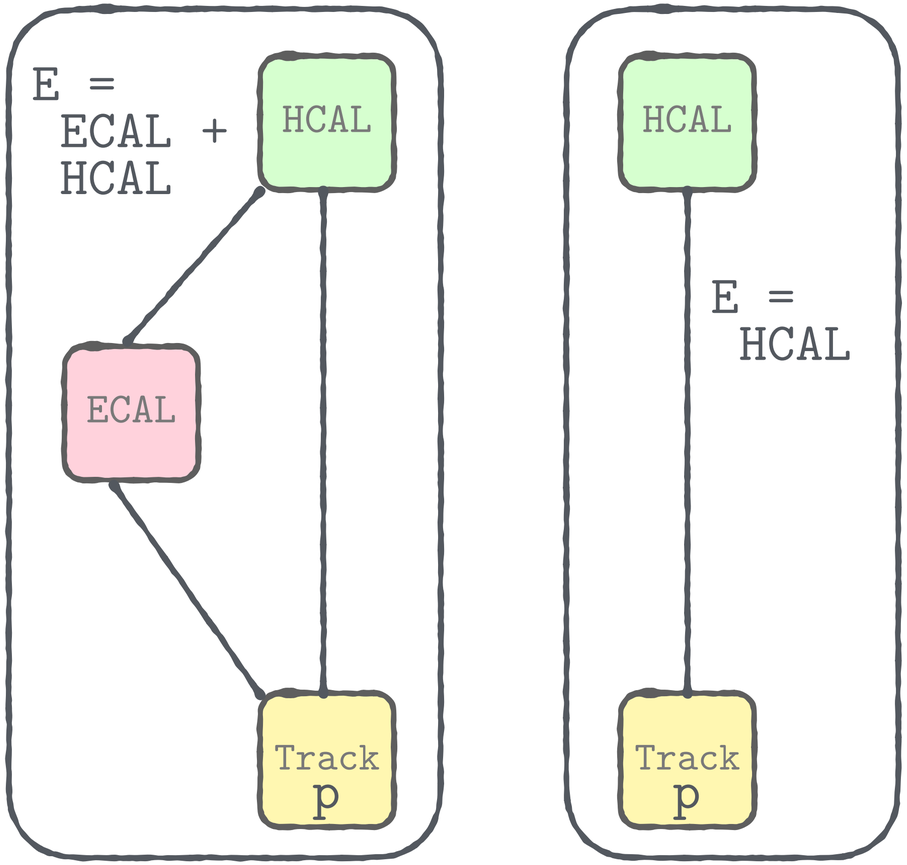
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ, π^-,

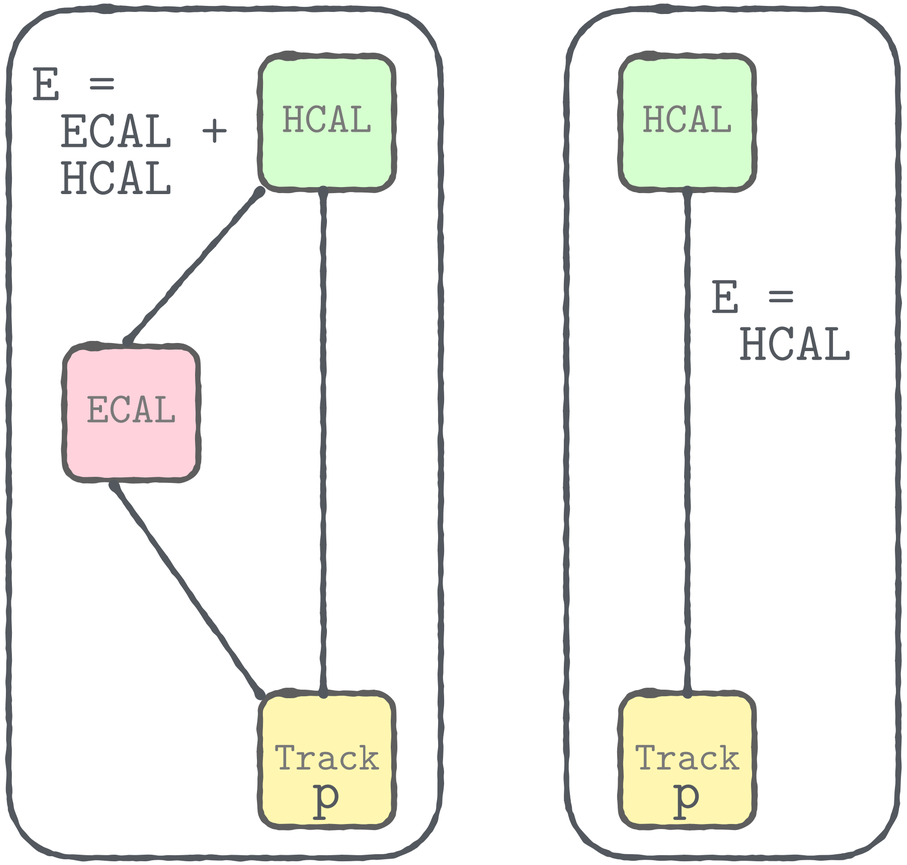
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

$\{\gamma, \gamma, \gamma, \pi^-, \gamma$

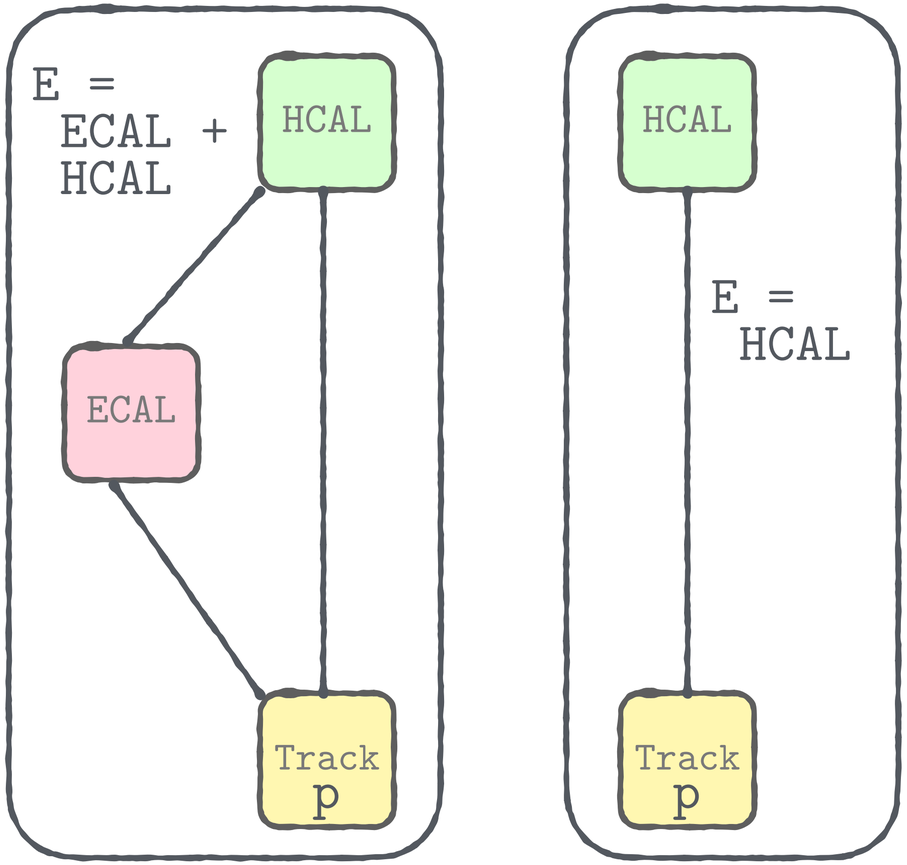
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^-, \gamma, \ldots$

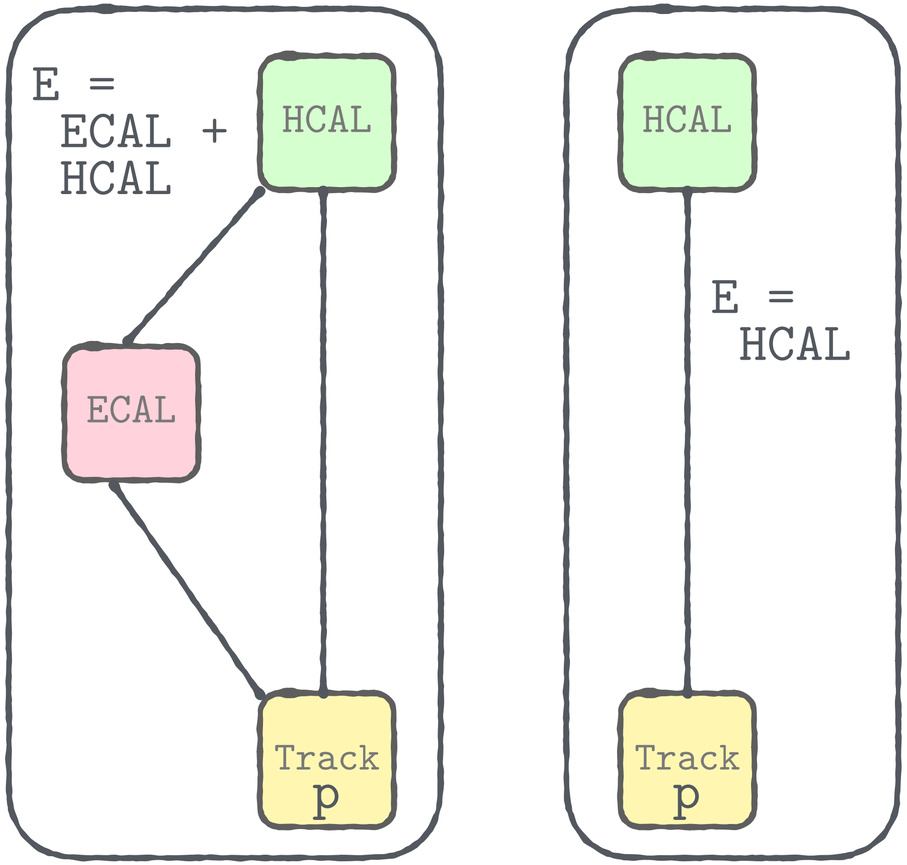
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^-, \gamma, K_L^0$

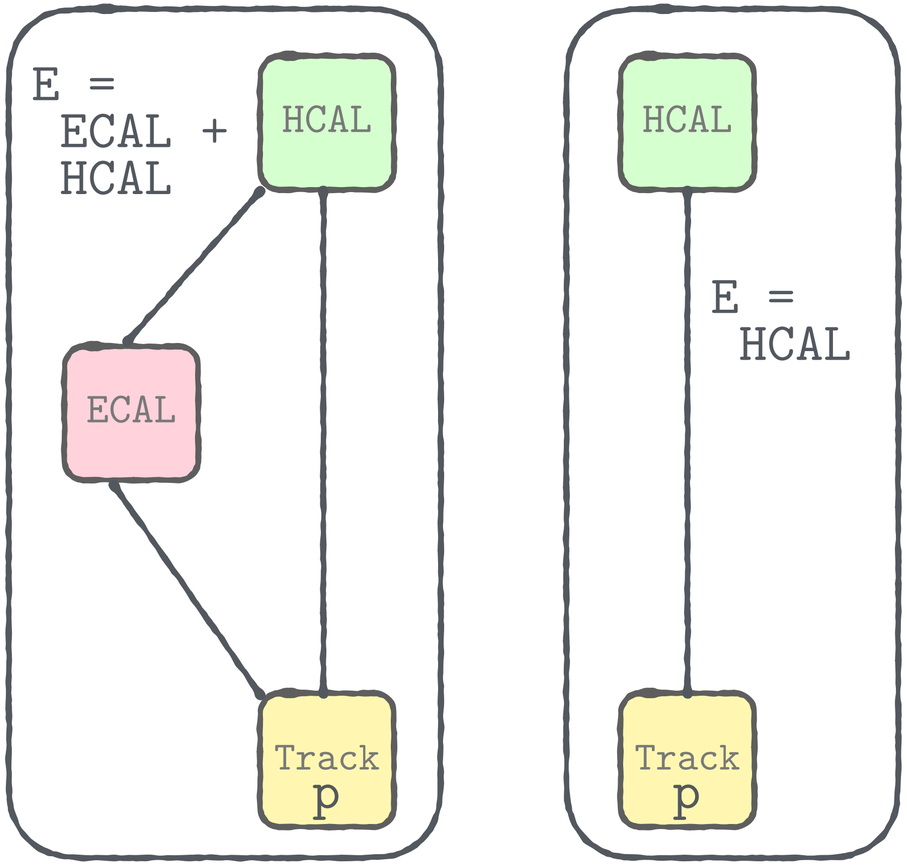
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

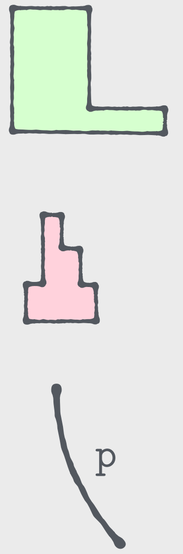
- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b



attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

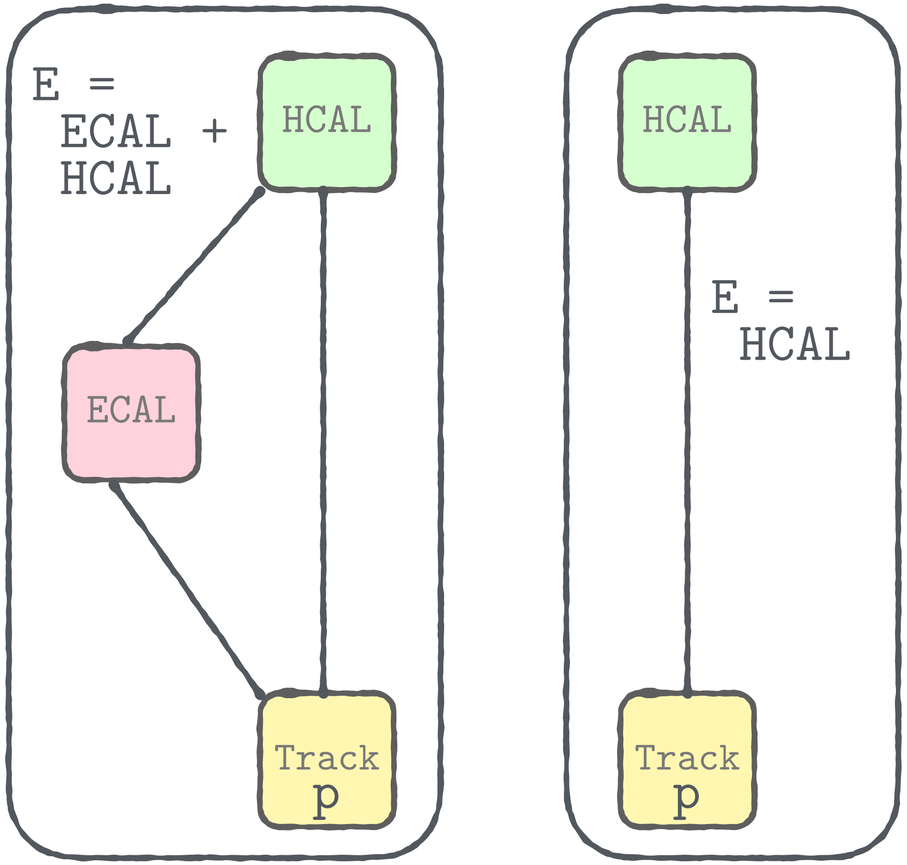
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ,

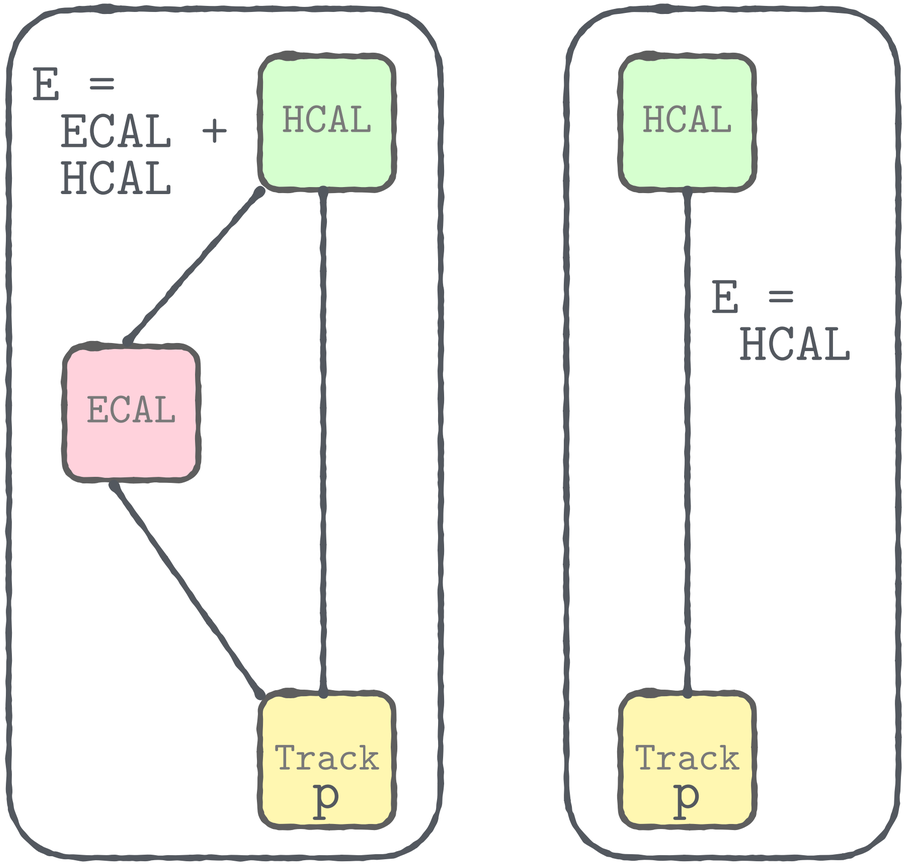
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^- \}$

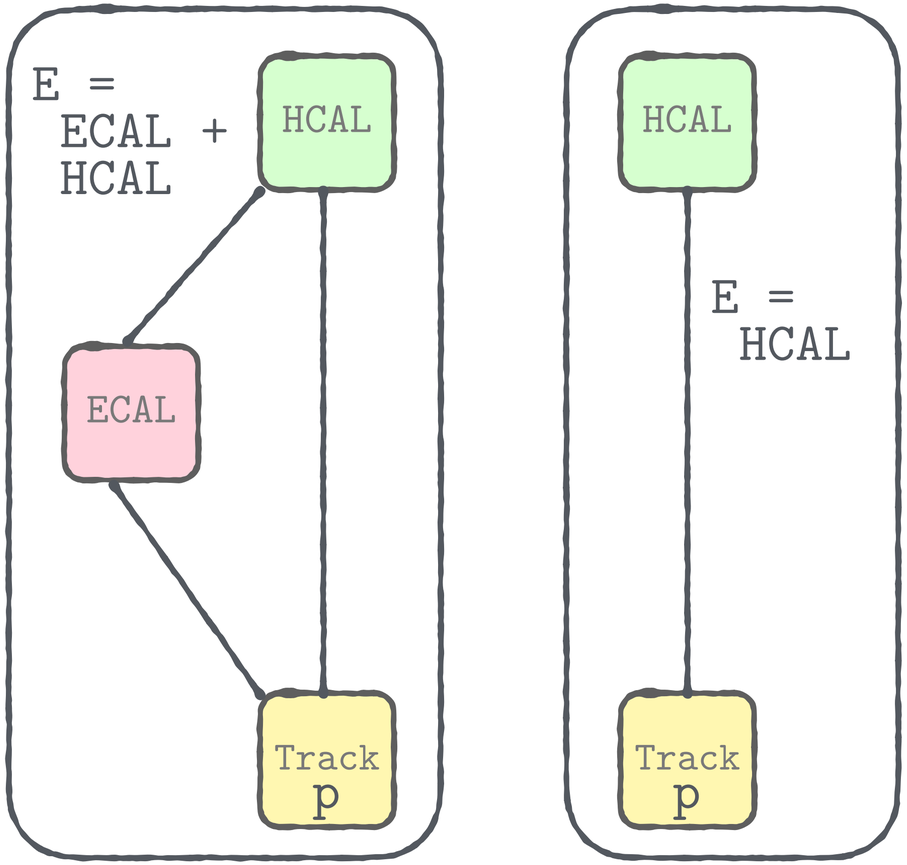
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ, π^-,

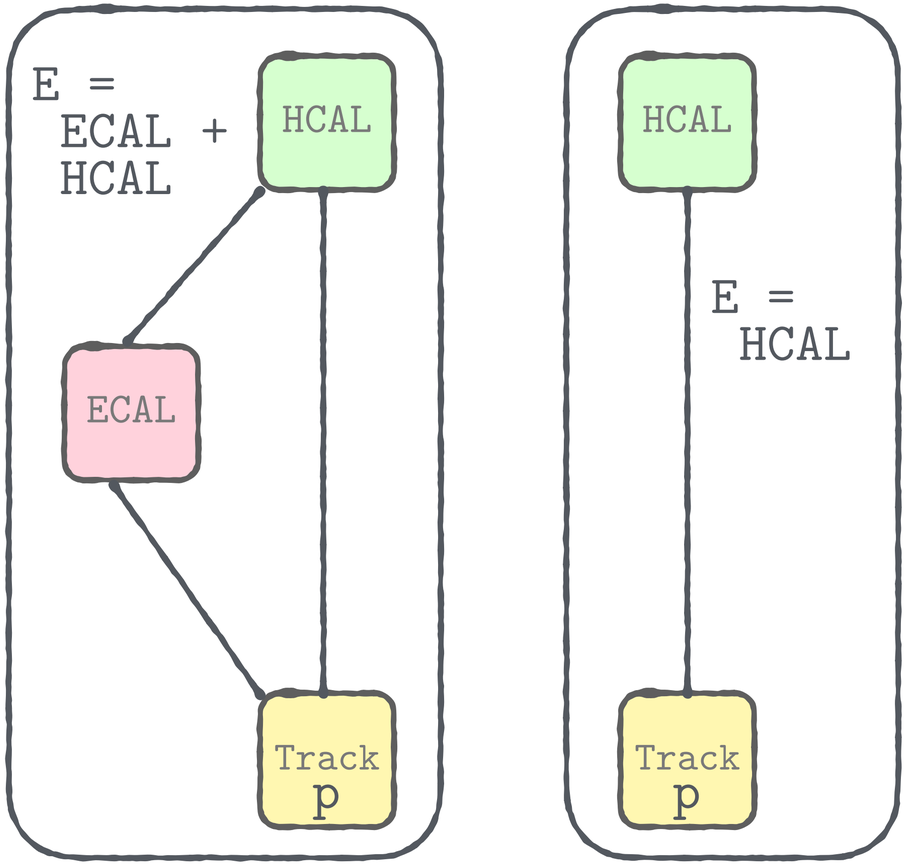
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Su

If p an

- Identify
  - one pe

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ $\gamma, \gamma, \gamma, \pi^-, \gamma$

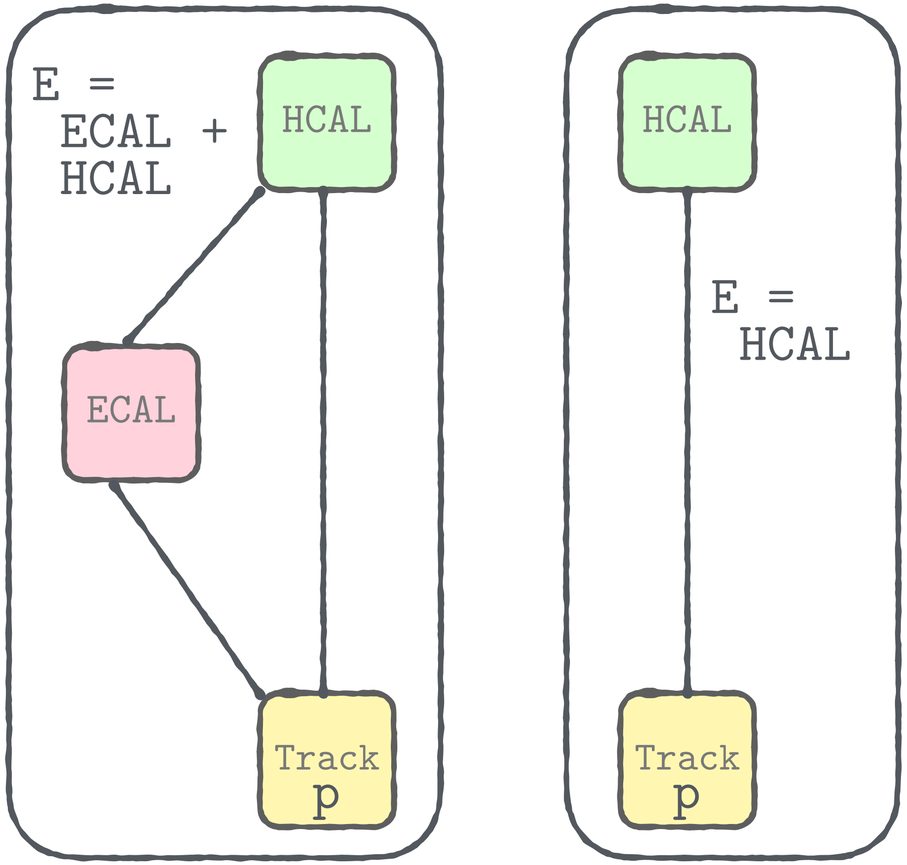
(blocks are usually very small)

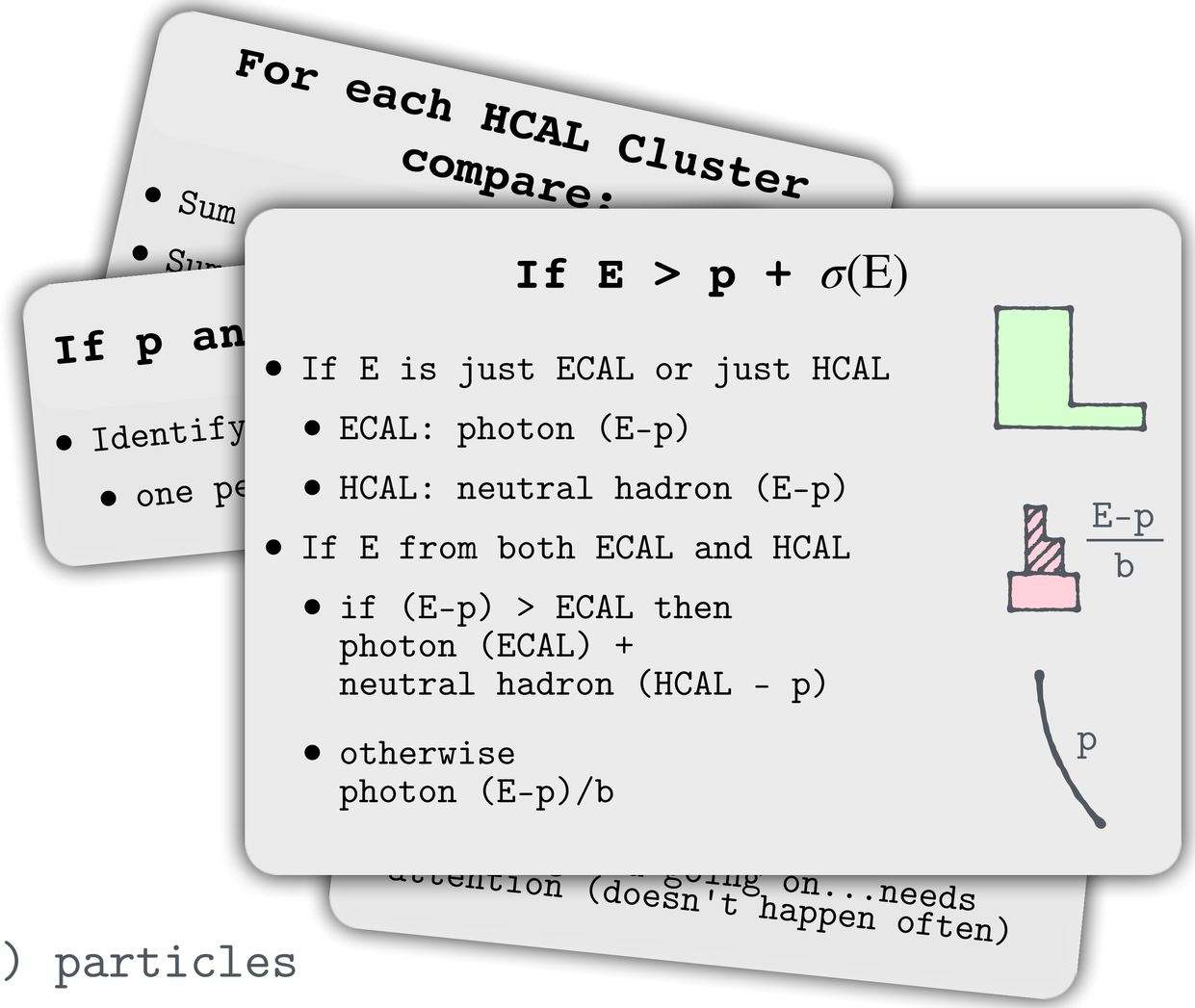
Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons





List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^-, \gamma$

Always give precedence to photons in ECAL

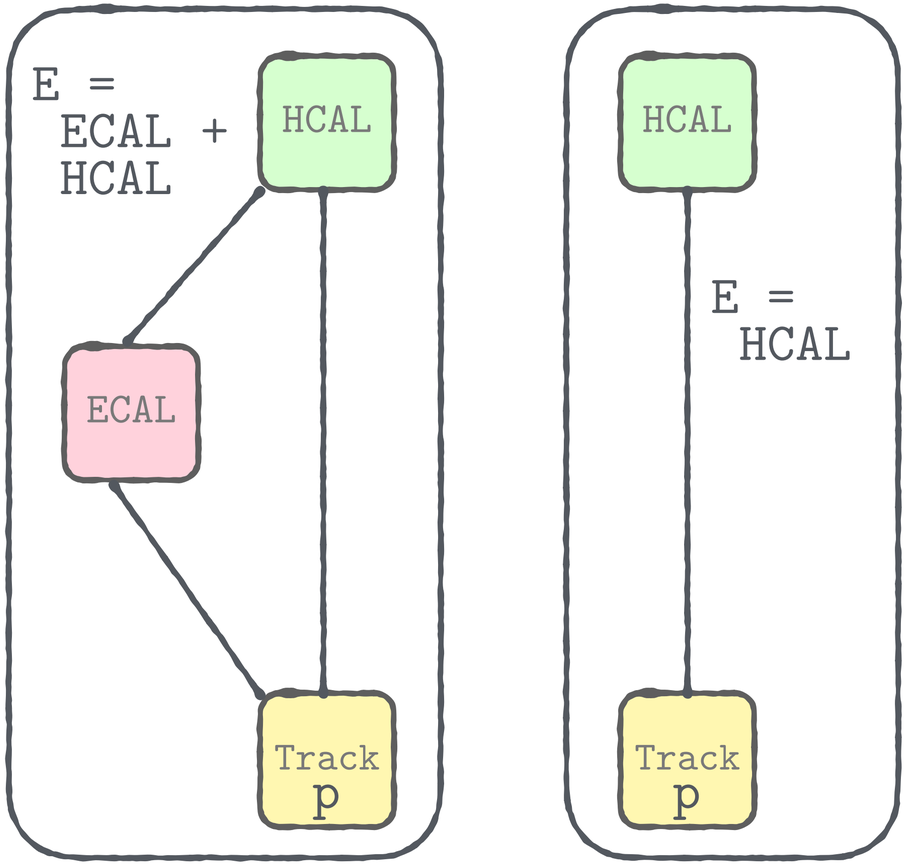
(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons



For each HCAL Cluster compare:

- Sum
- Sum
- lin... in ... "had...

If E > p + σ(E)

- If E is just ECAL or just HCAL
  - ECAL: photon (E-p)
  - HCAL: neutral hadron (E-p)
- If E from both ECAL and HCAL
  - if (E-p) > ECAL then
    photon (ECAL) +
    neutral hadron (HCAL - p)
  - otherwise
    photon (E-p)/b

...attention (...going on...needs attention (doesn't happen often)

List of reconstructed (candidate) particles

{ γ, γ, γ, ...

Always give precedence to photons in ECAL

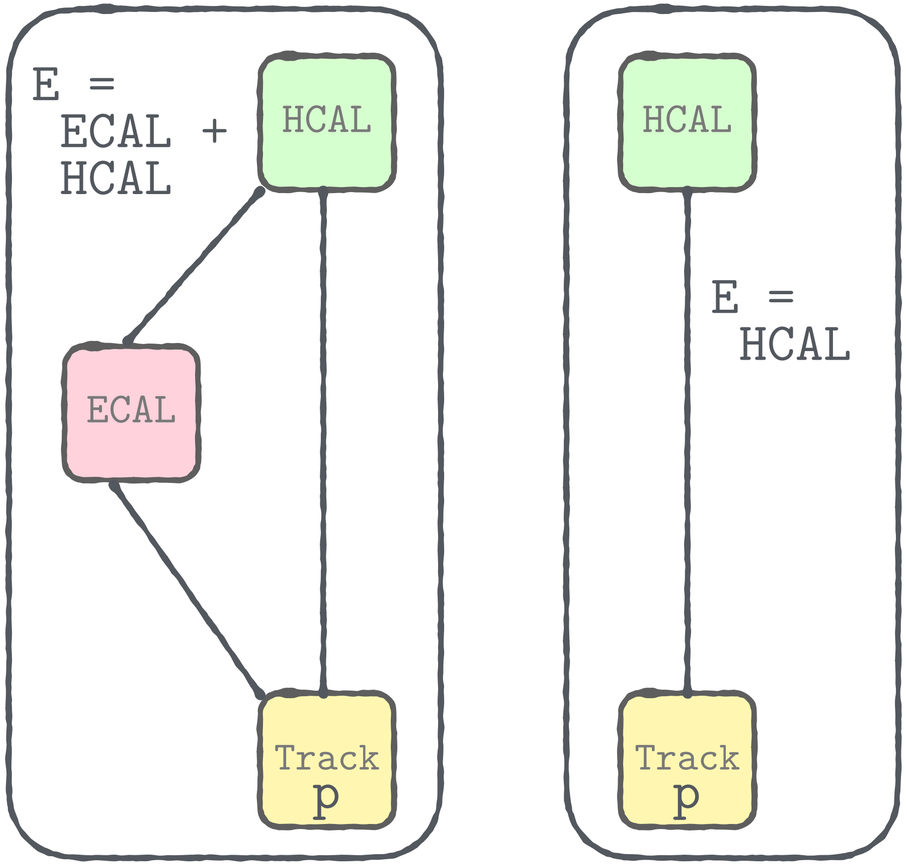
(blocks are usually very small)

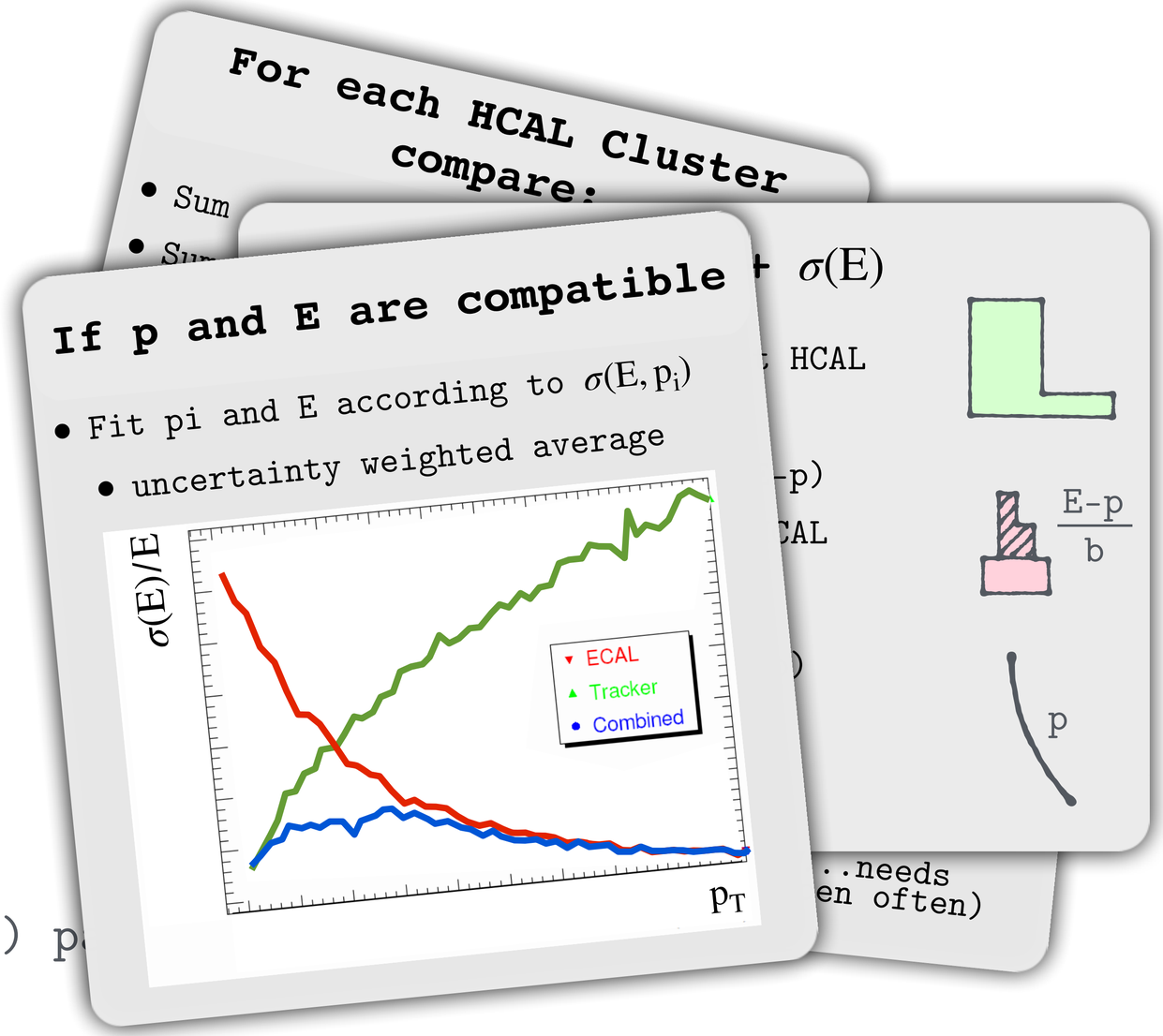
Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons





List of reconstructed (candidate) p

{ $\gamma, \gamma, \gamma,$

Always give precedence to photons in ECAL

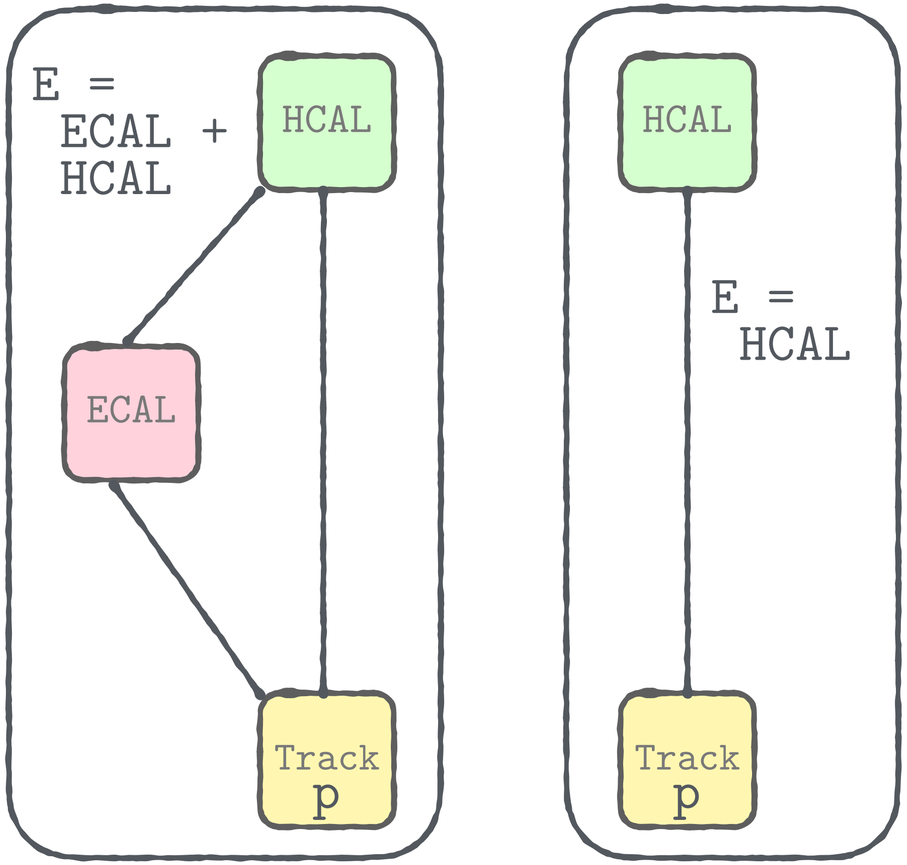
(blocks are usually very small)

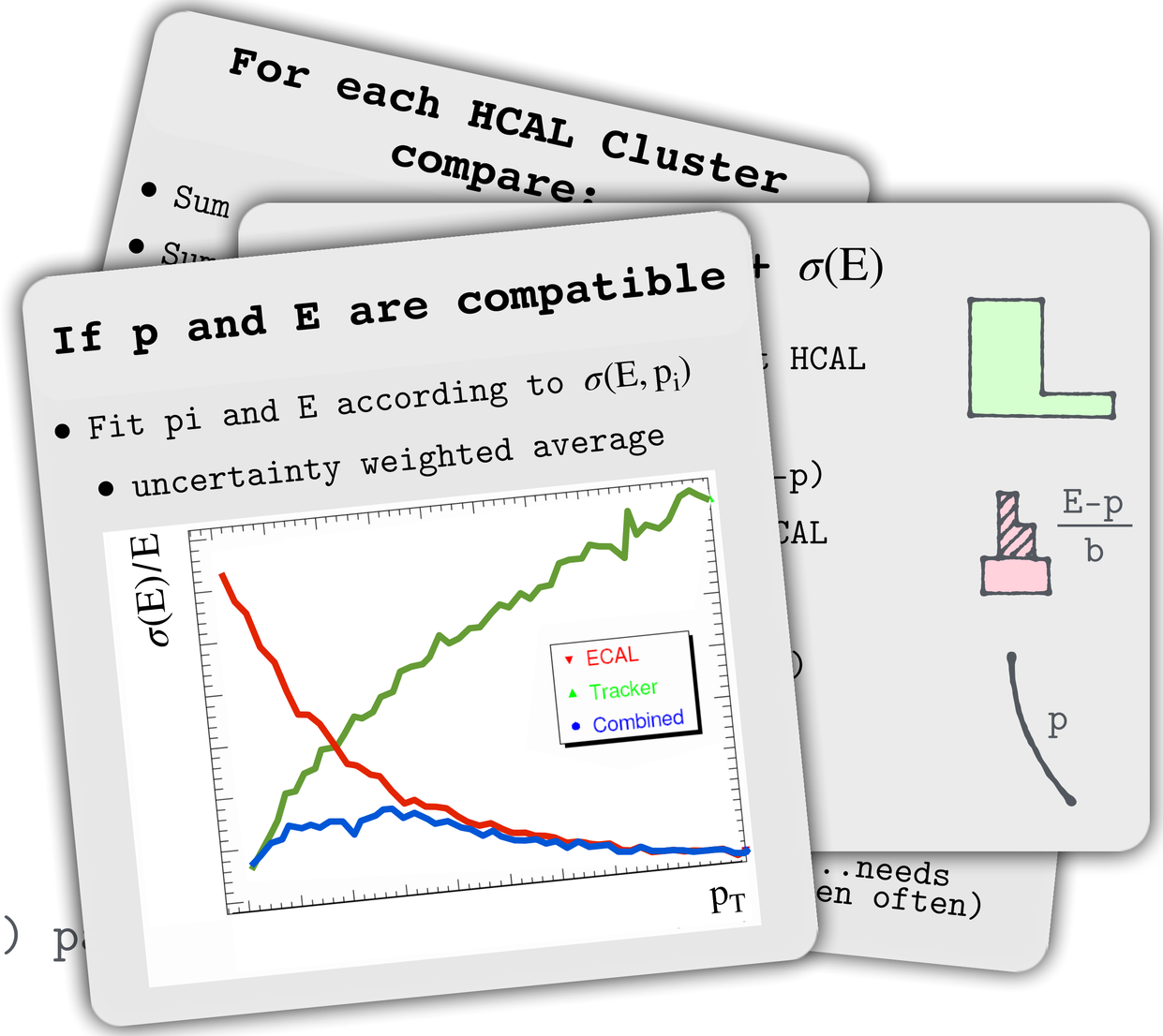
Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons





List of reconstructed (candidate) p

{ $\gamma, \gamma, \gamma, \pi^+$

Always give precedence to photons in ECAL

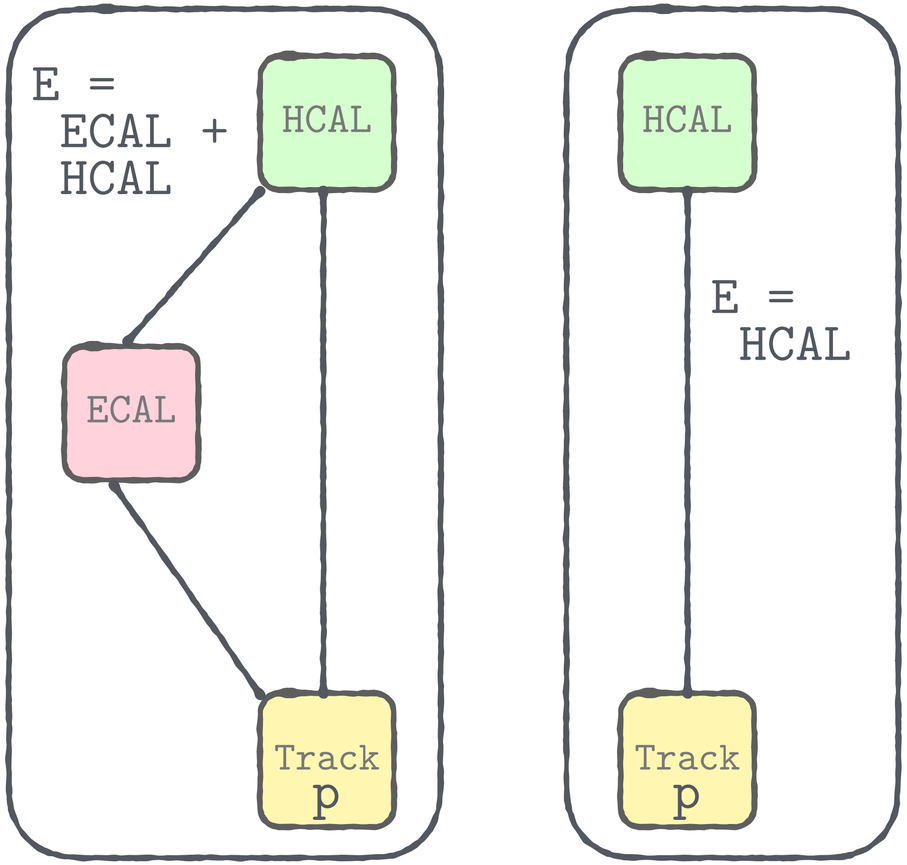
(blocks are usually very small)

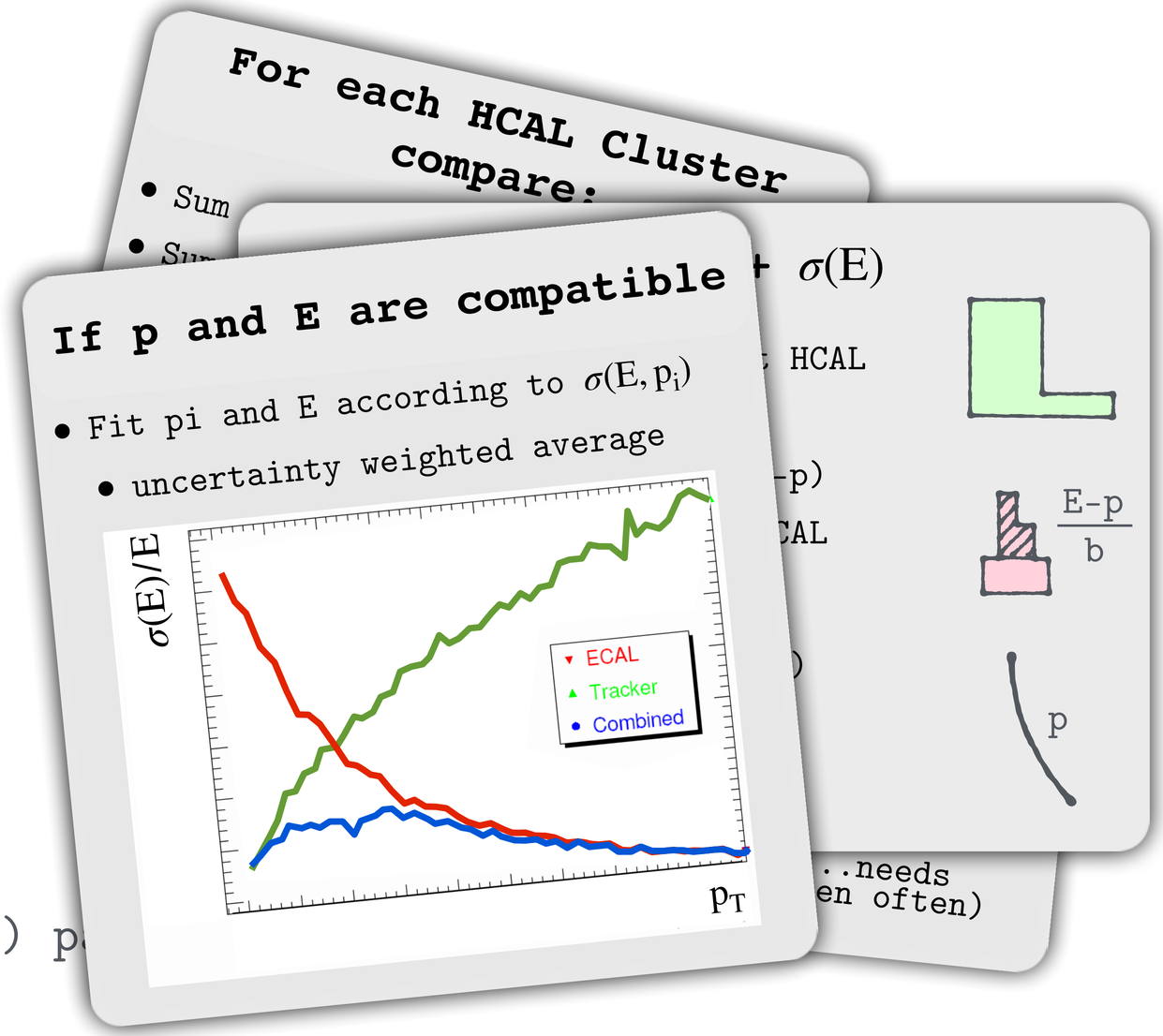
Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons





List of reconstructed (candidate) p

{ $\gamma, \gamma, \gamma, \pi^+,$

Always give precedence to photons in ECAL

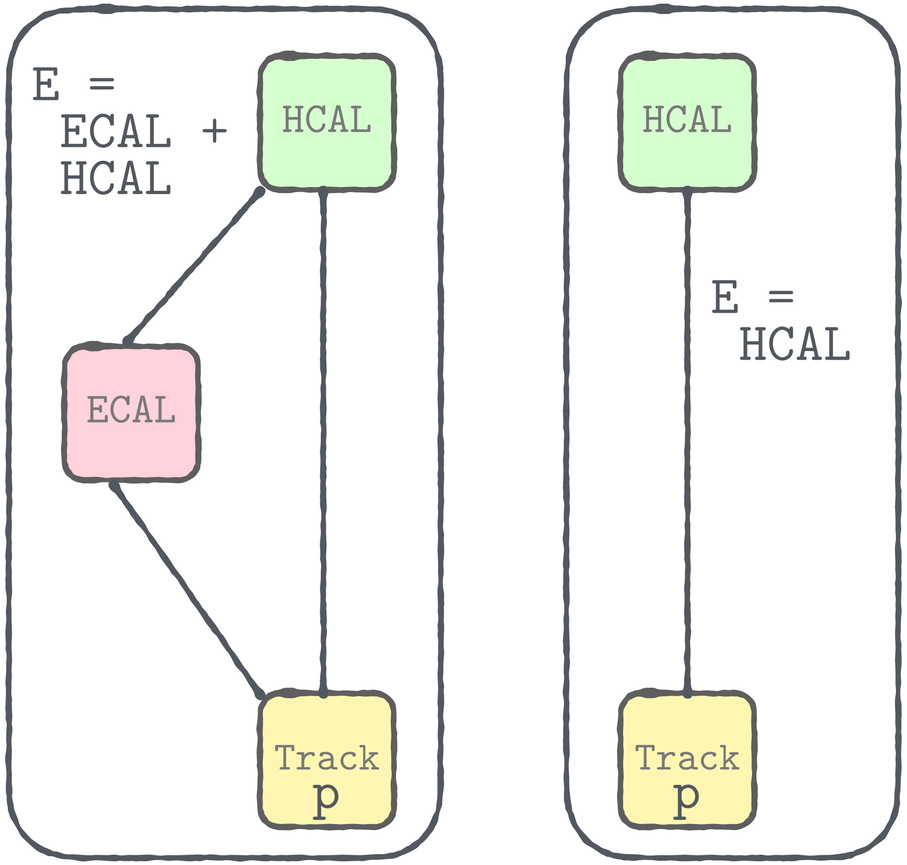
(blocks are usually very small)

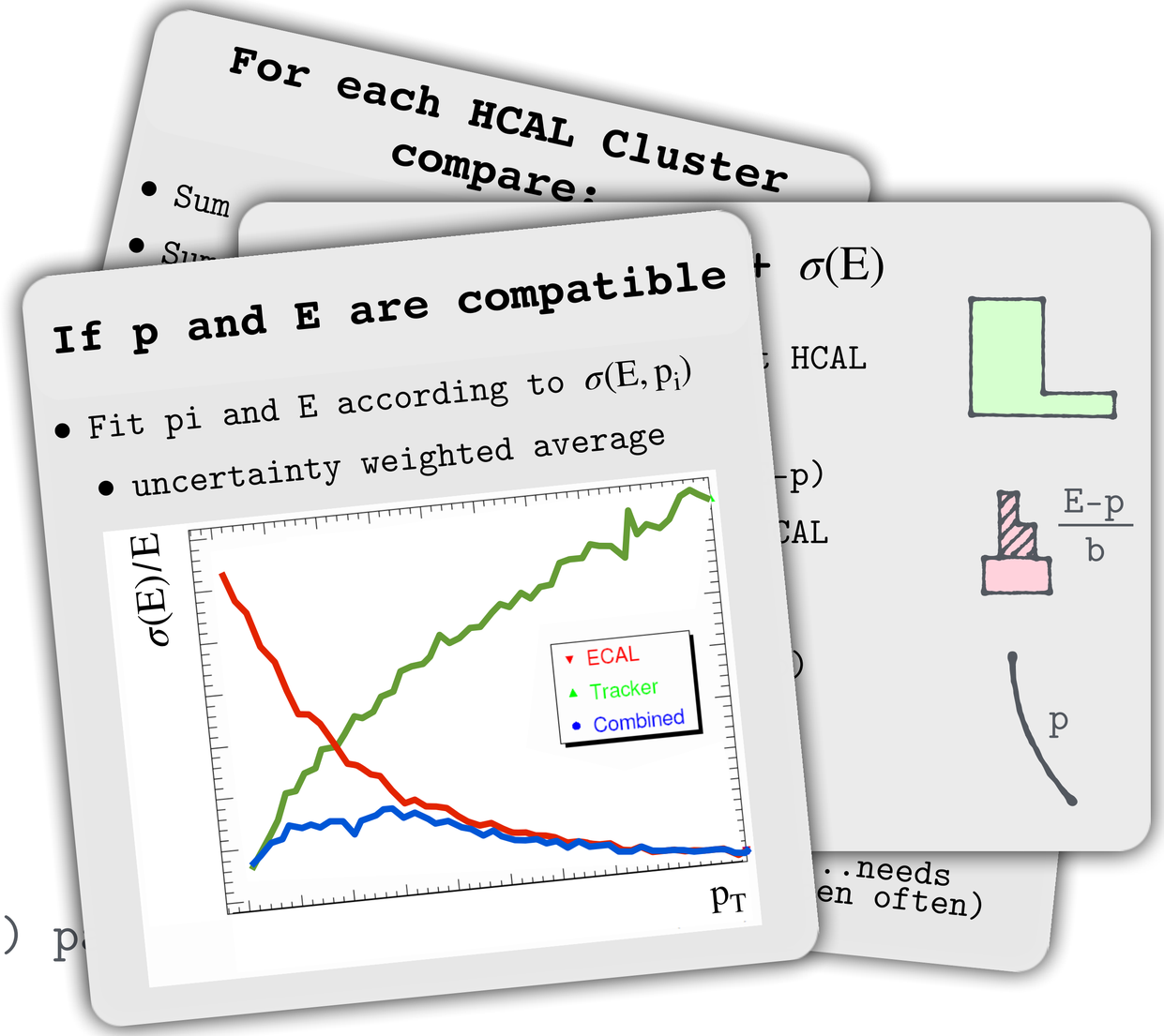
Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons





List of reconstructed (candidate) p

{ $\gamma, \gamma, \gamma, \pi^+, \pi^-$

Always give precedence to photons in ECAL

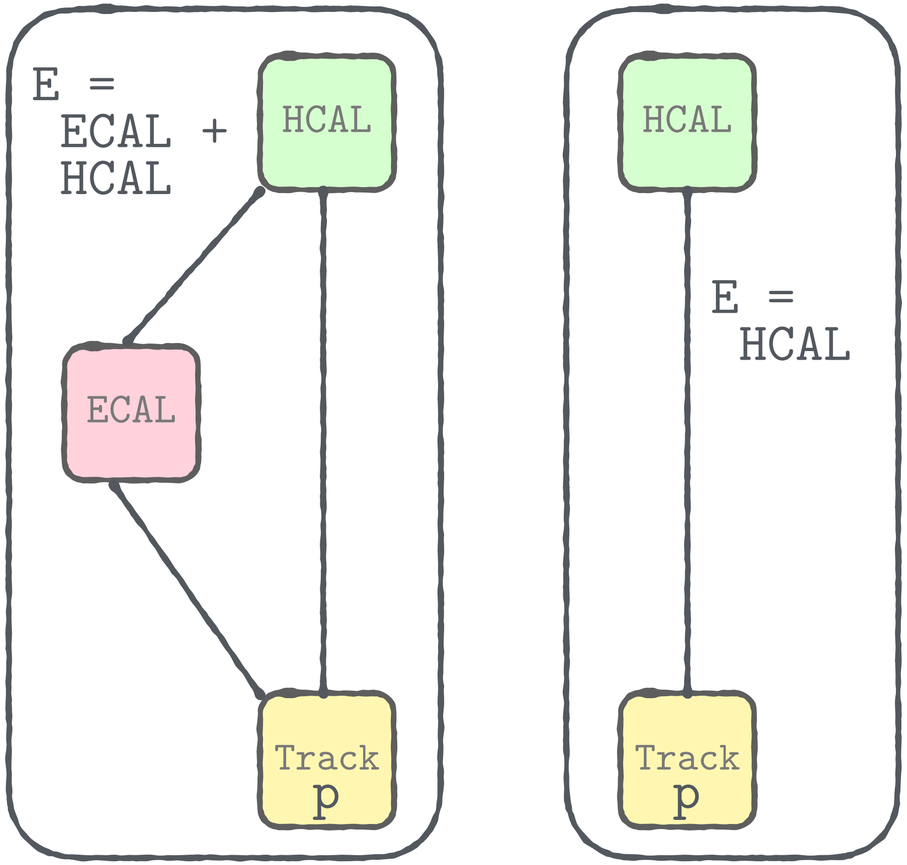
(blocks are usually very small)

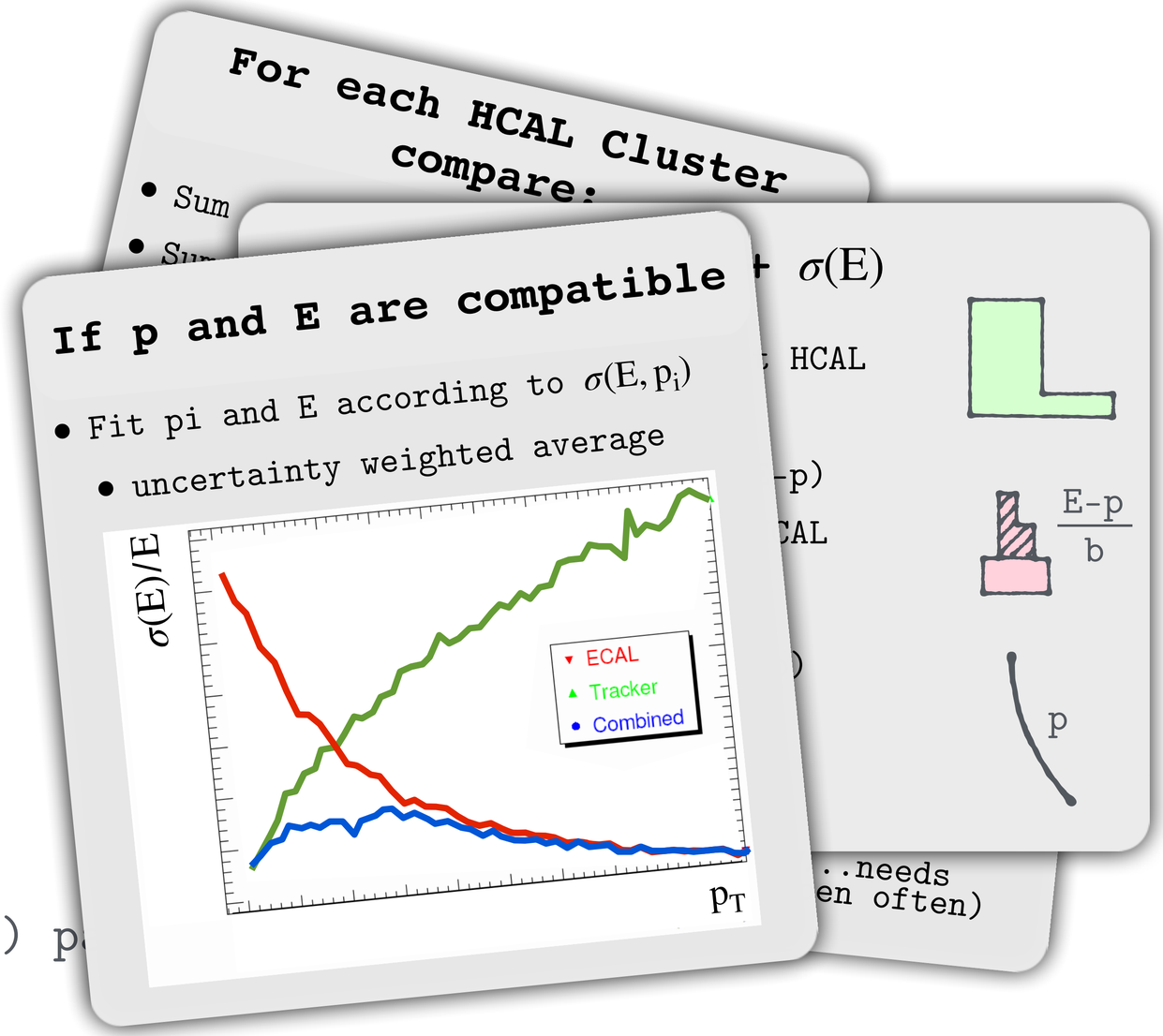
Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Simplified block (2nd step)

Find charged hadrons & merged photons / neutral hadrons





List of reconstructed (candidate) p

$\{ \gamma, \gamma, \gamma, \pi^+, \pi^- \}$

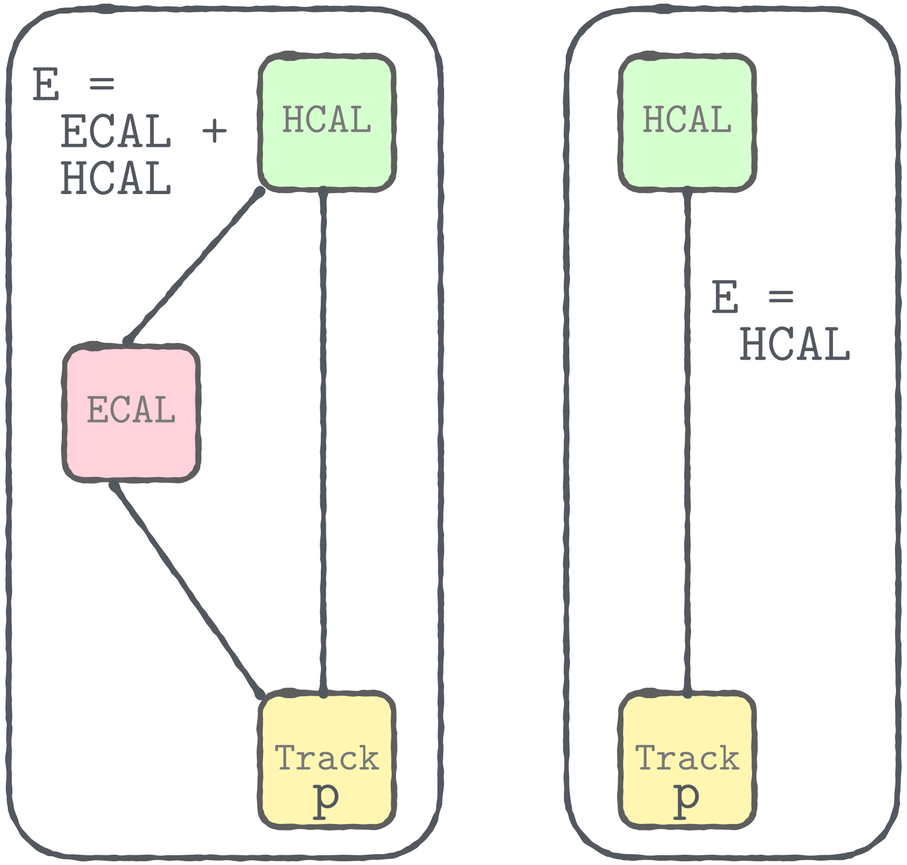
Always give precedence to photons in ECAL

(blocks are usually very small)

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Conclusion from this simple Example



List of reconstructed (candidate) particles

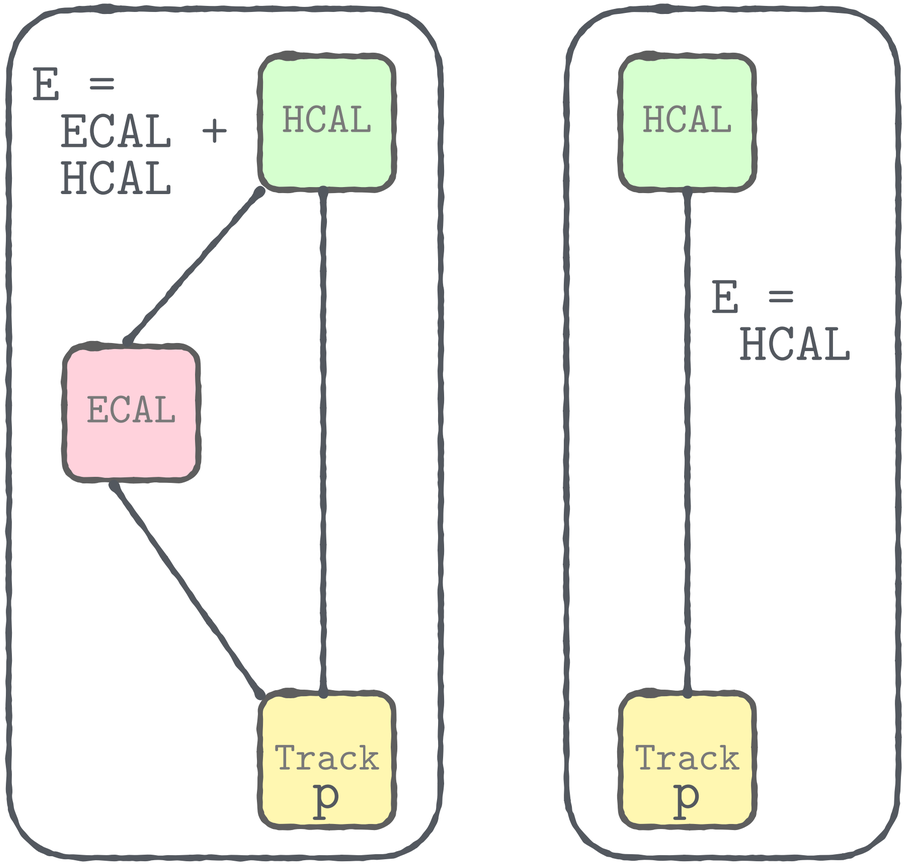
$\{ \gamma, \gamma, \gamma, \pi^+, \pi^- \}$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$

Conclusion from this simple Example

- Five particles reconstructed

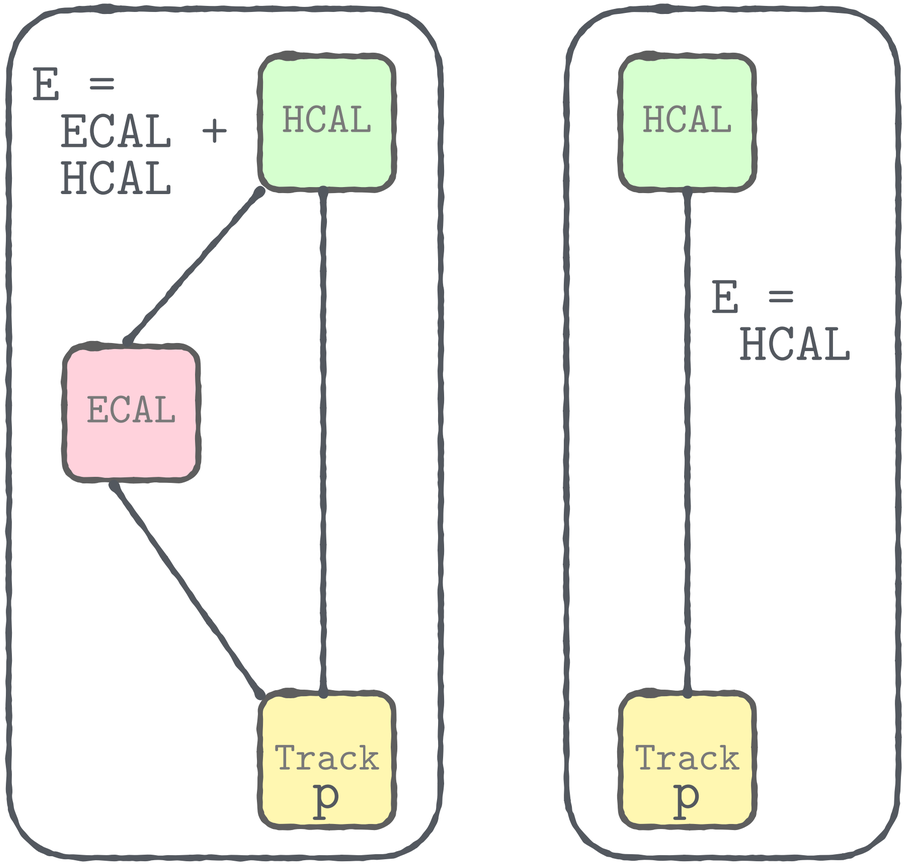


List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^+, \pi^- \}$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$



Conclusion from this simple Example

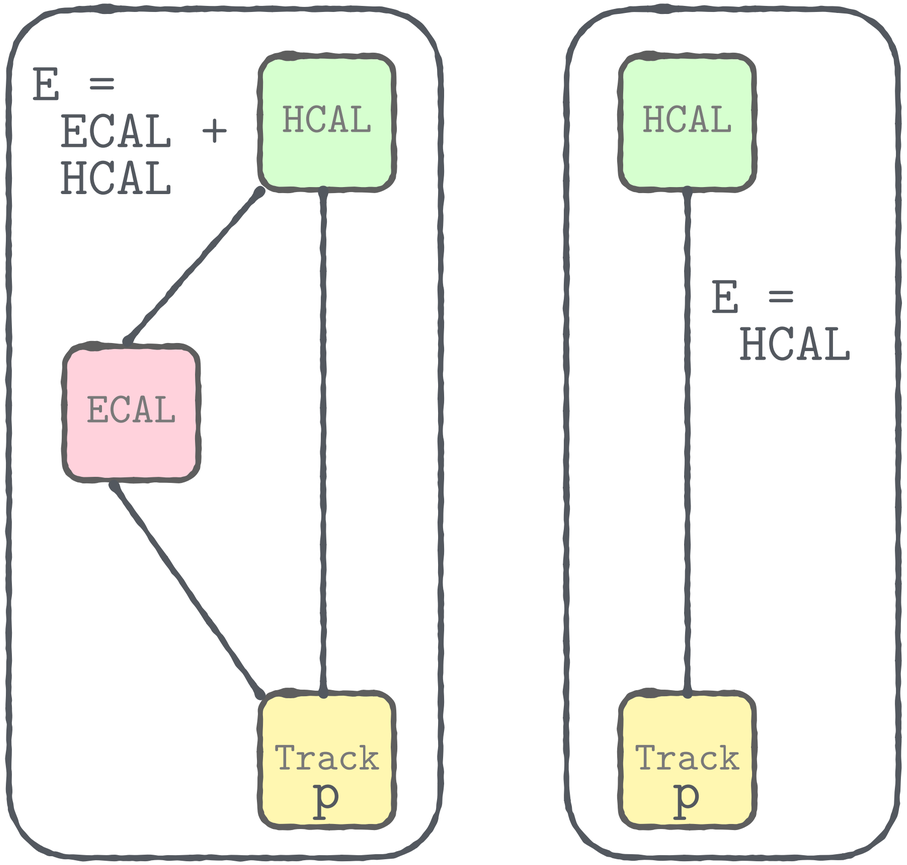
- Five particles reconstructed
  - two oppositely charged hadrons (π^+ and π^-)

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^+, \pi^- \}$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$



Conclusion from this simple Example

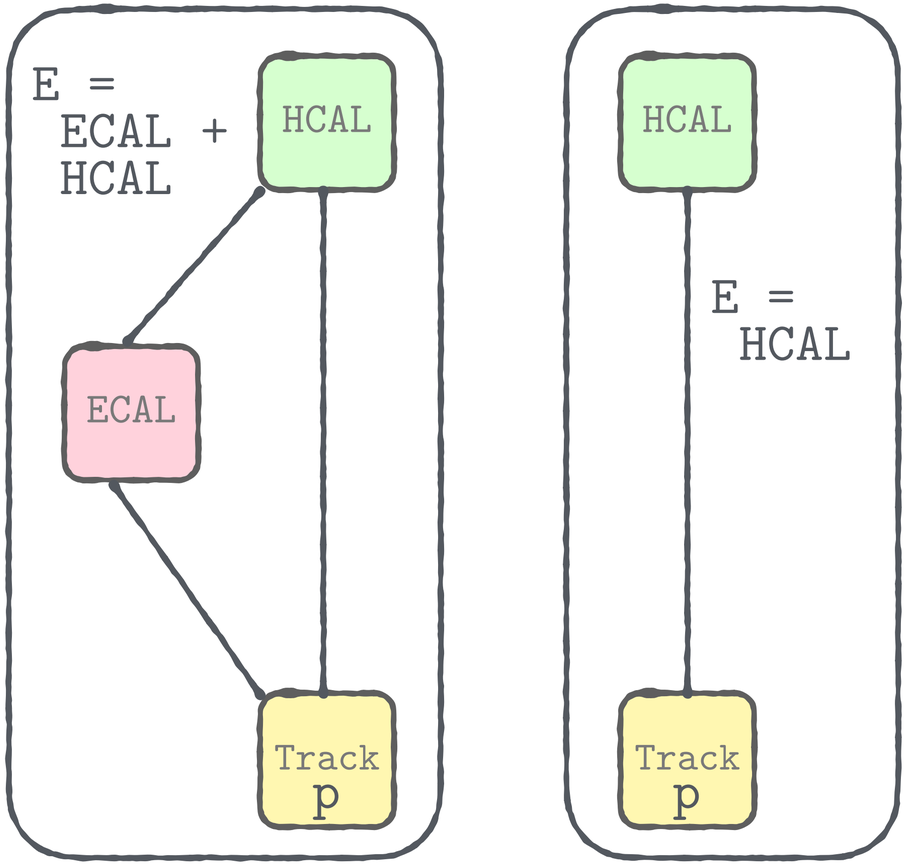
- Five particles reconstructed
  - two oppositely charged hadrons (π^+ and π^-)
  - three photons

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^+, \pi^- \}$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$



Conclusion from this simple Example

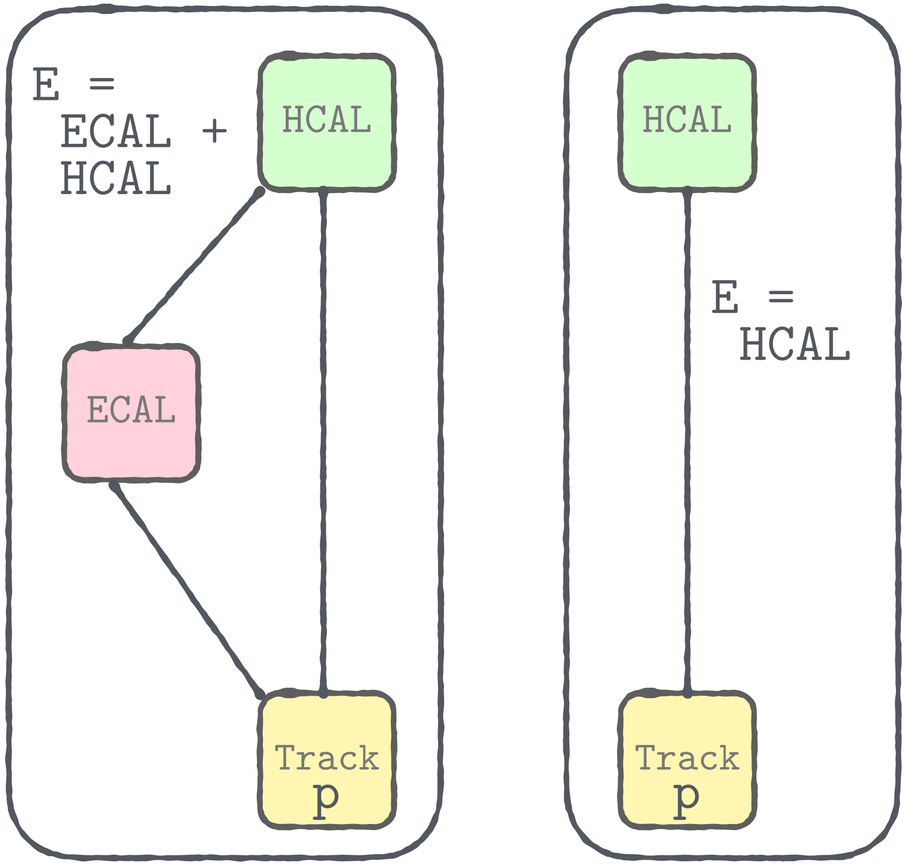
- Five particles reconstructed
  - two oppositely charged hadrons (π^+ and π^-)
  - three photons
    - two from π^0 decay and one from K_L^0 energy deposit

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^+, \pi^- \}$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$



Conclusion from this simple Example

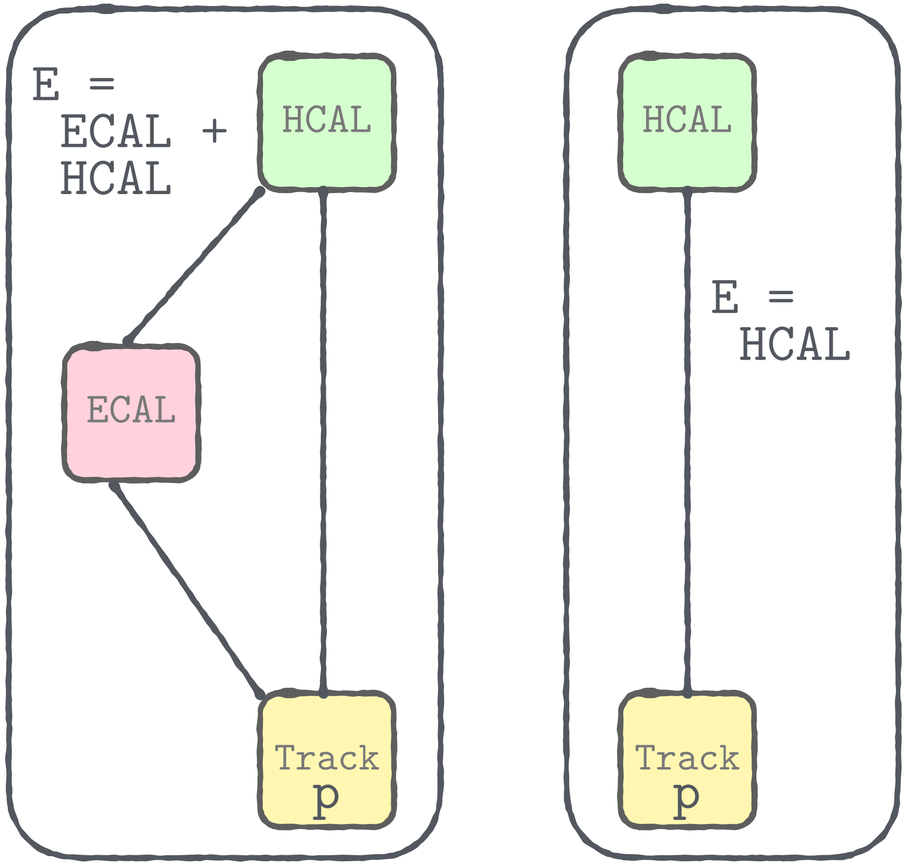
- Five particles reconstructed
  - two oppositely charged hadrons (π^+ and π^-)
  - three photons
    - two from π^0 decay and one from K_L^0 energy deposit
  - no neutral hadron

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^+, \pi^- \}$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$



Conclusion from this simple Example

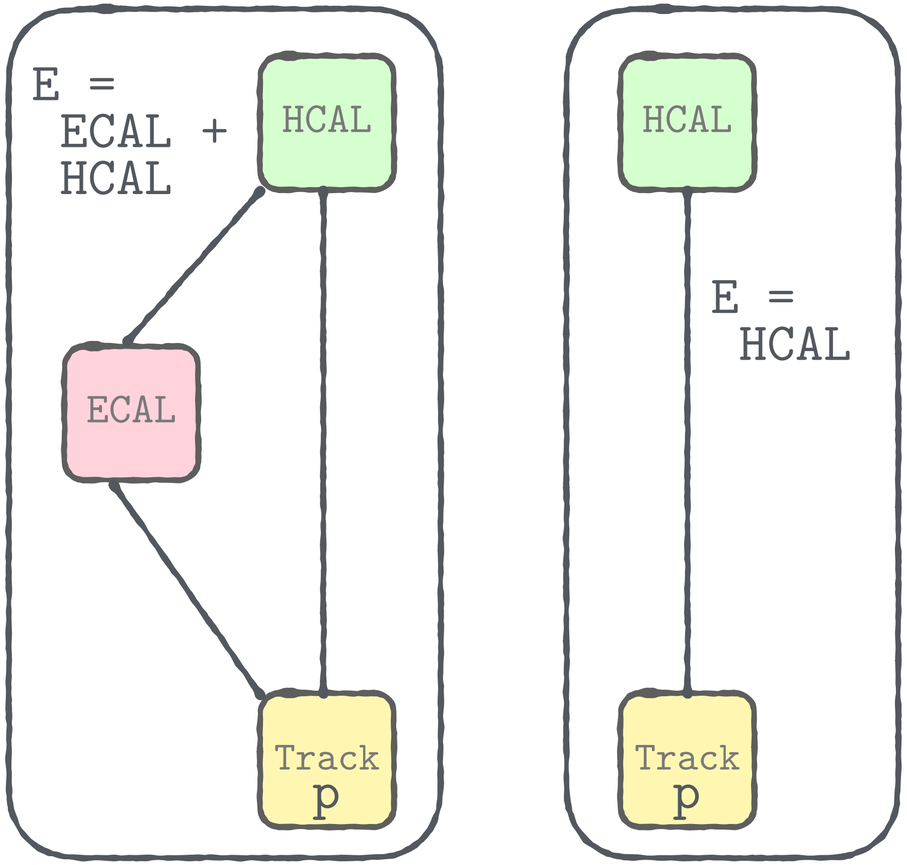
- Five particles reconstructed
  - two oppositely charged hadrons (π^+ and π^-)
  - three photons
    - two from π^0 decay and one from K_L^0 energy deposit
  - no neutral hadron
    - because for both tracks, E compatible with p

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^+, \pi^- \}$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$



Conclusion from this simple Example

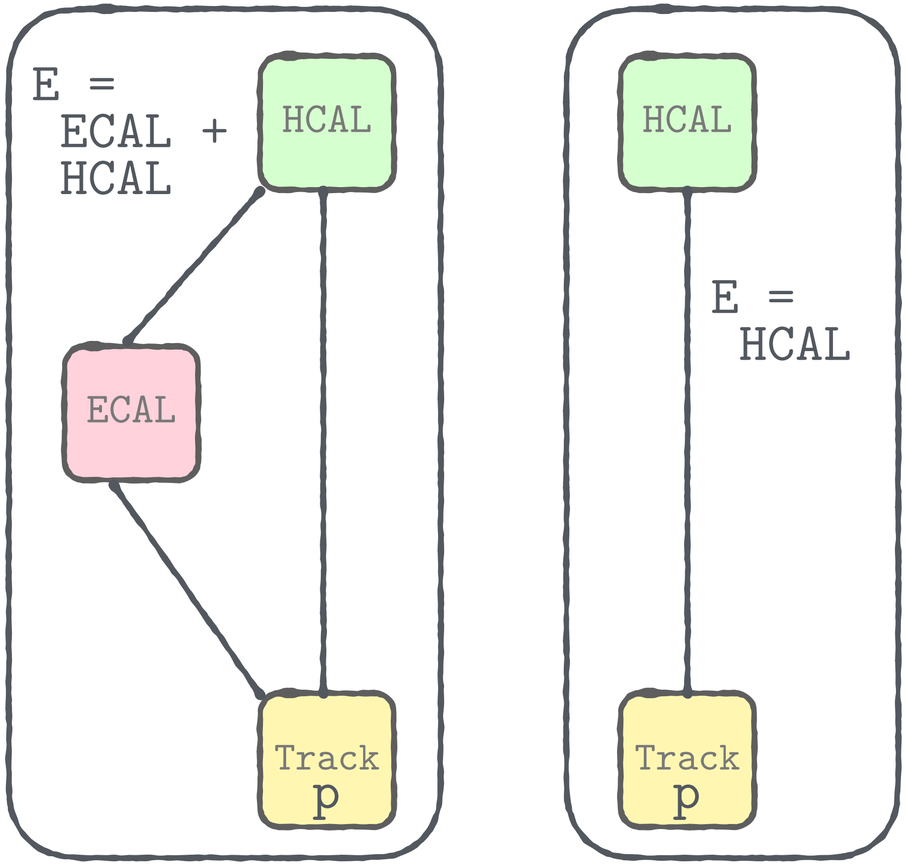
- Five particles reconstructed
  - two oppositely charged hadrons (π^+ and π^-)
  - three photons
    - two from π^0 decay and one from K_L^0 energy deposit
  - no neutral hadron
    - because for both tracks, E compatible with p
- Note: the precedence given to photon ID in ECAL

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^+, \pi^- \}$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$



Conclusion from this simple Example

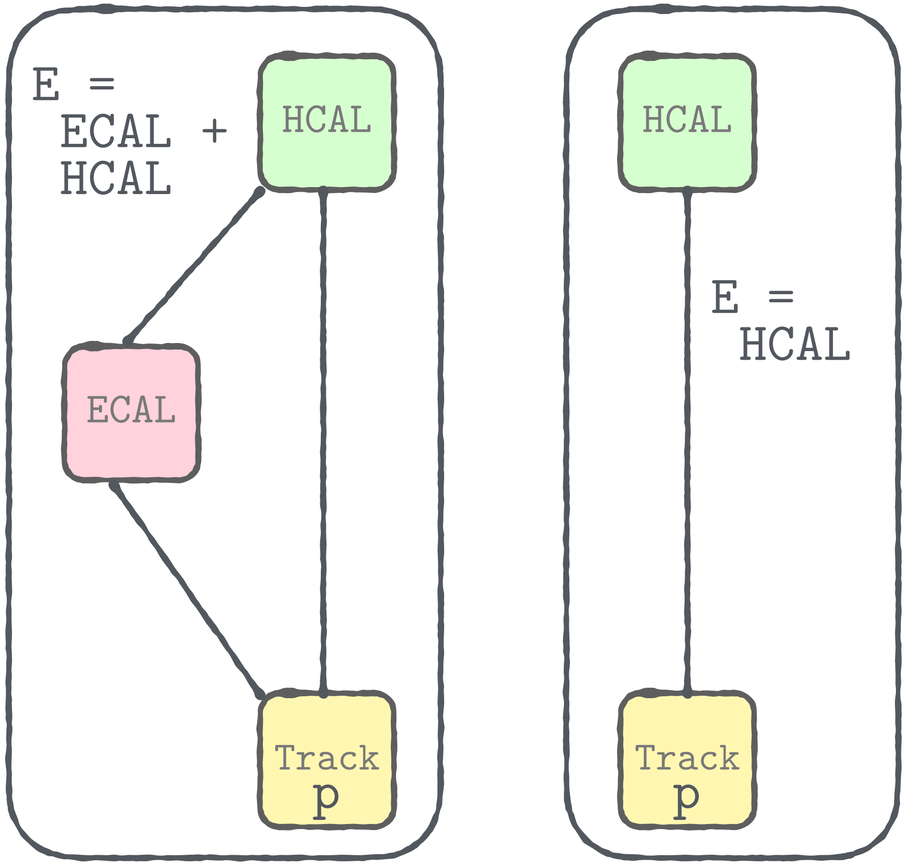
- Five particles reconstructed
  - two oppositely charged hadrons (π^+ and π^-)
  - three photons
    - two from π^0 decay and one from K_L^0 energy deposit
  - no neutral hadron
    - because for both tracks, E compatible with p
- Note: the precedence given to photon ID in ECAL
  - Could underestimate ECAL deposits of neutral hadrons

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^+, \pi^- \}$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$



Conclusion from this simple Example

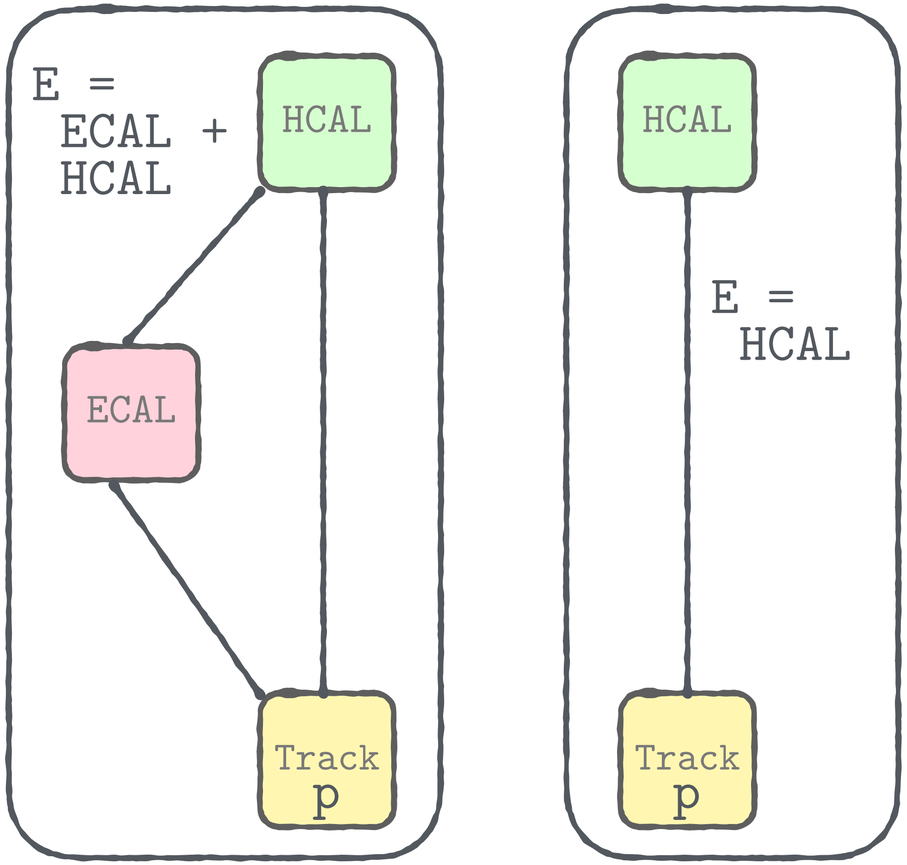
- Five particles reconstructed
  - two oppositely charged hadrons (π^+ and π^-)
  - three photons
    - two from π^0 decay and one from K_L^0 energy deposit
  - no neutral hadron
    - because for both tracks, E compatible with p
- Note: the precedence given to photon ID in ECAL
  - Could underestimate ECAL deposits of neutral hadrons
  - But neutral hadron energy deposited in ECAL is

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^+, \pi^- \}$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$



Conclusion from this simple Example

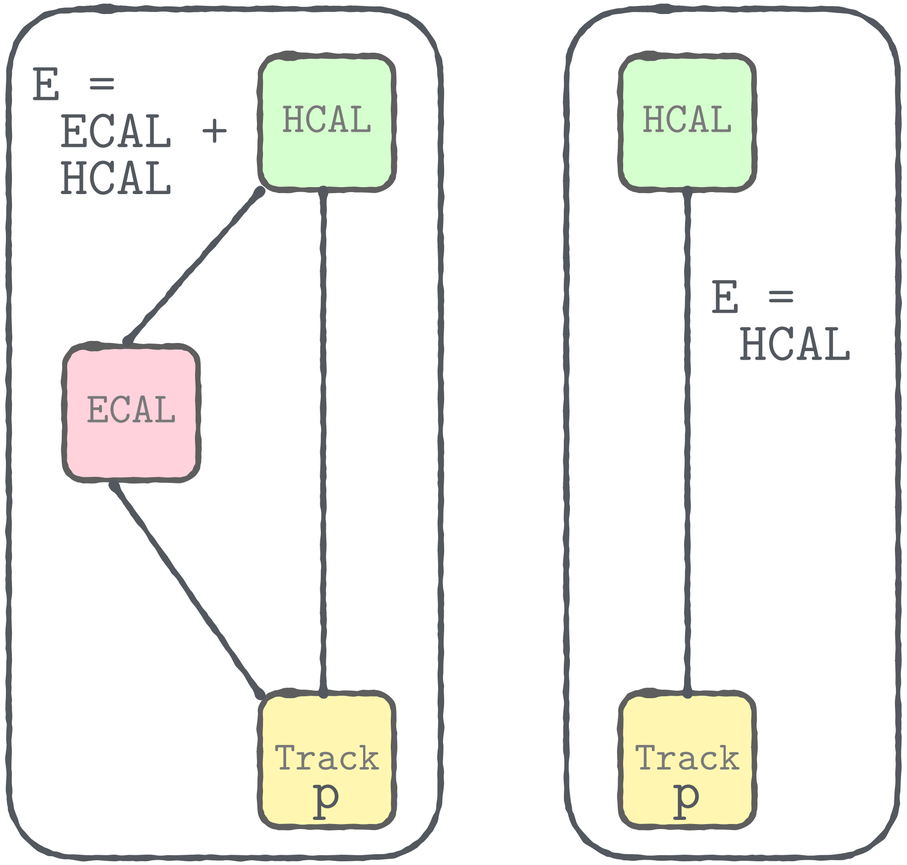
- Five particles reconstructed
  - two oppositely charged hadrons (π^+ and π^-)
  - three photons
    - two from π^0 decay and one from K_L^0 energy deposit
  - no neutral hadron
    - because for both tracks, E compatible with p
- Note: the precedence given to photon ID in ECAL
  - Could underestimate ECAL deposits of neutral hadrons
  - But neutral hadron energy deposited in ECAL is
    - (10% neutral hadron energy) x (30% ECAL fraction) = 3% of event energy

List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^+, \pi^- \}$

Four true particles:

$\pi^+, \pi^-, \pi^0, K_L^0$



Conclusion from this simple Example

- Five particles reconstructed
  - two oppositely charged hadrons (π^+ and π^-)
  - three photons
    - two from π^0 decay and one from K_L^0 energy deposit
  - no neutral hadron
    - because for both tracks, E compatible with p
- Note: the precedence given to photon ID in ECAL
  - Could underestimate ECAL deposits of neutral hadrons
  - But neutral hadron energy deposited in ECAL is
    - (10% neutral hadron energy) x (30% ECAL fraction) = 3% of event energy
    - might loose < 0.5% of event energy due to this choice

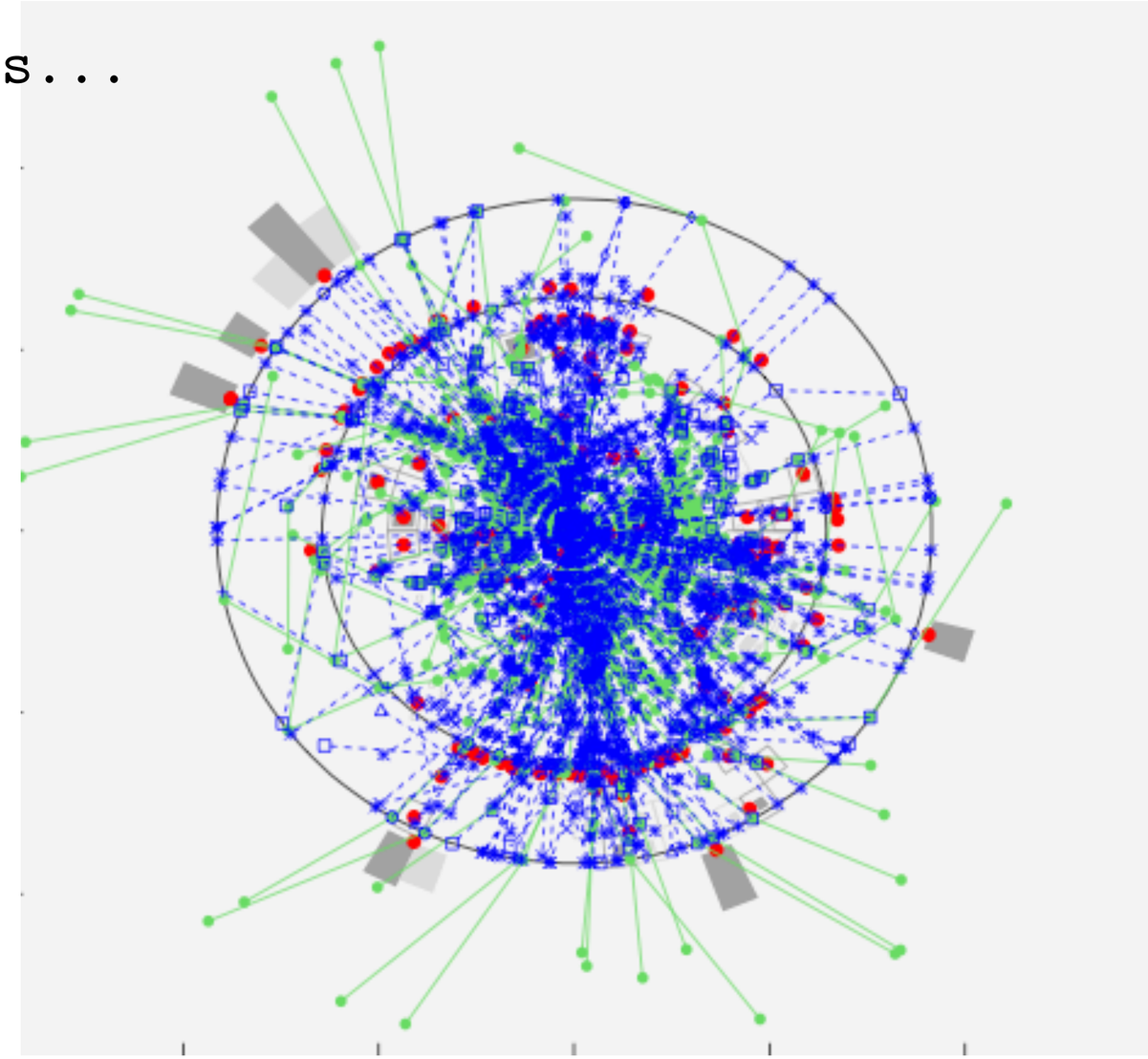
List of reconstructed (candidate) particles

$\{ \gamma, \gamma, \gamma, \pi^+, \pi^- \}$

That was a simple example, nevertheless...

...The Particle Flow algorithm scales to large particle multiplicities!

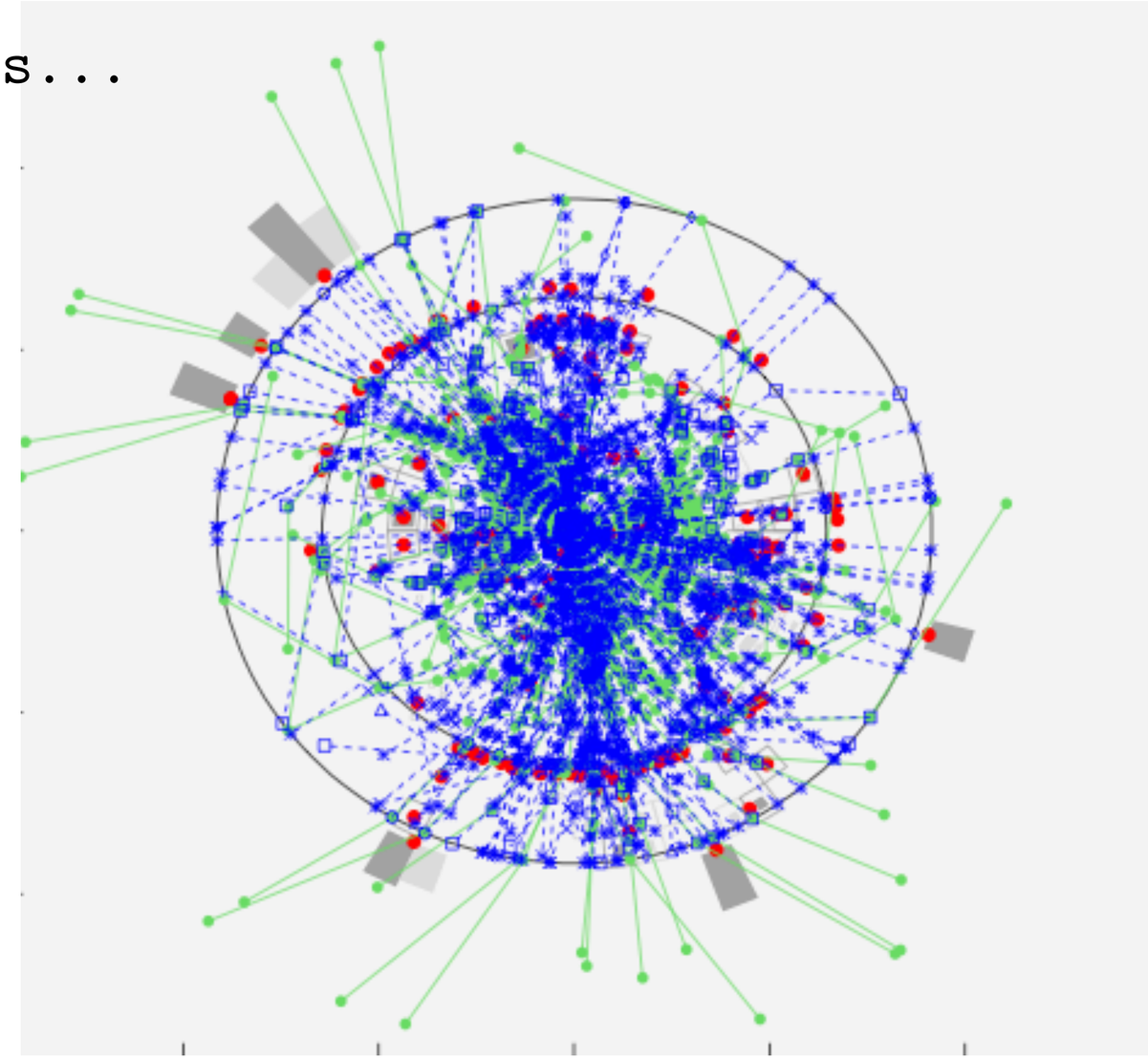
Analysis of the leading jet from simulated event of all-hadronic top-quark pairs:



That was a simple example, nevertheless...

...The Particle Flow algorithm scales to large particle multiplicities!

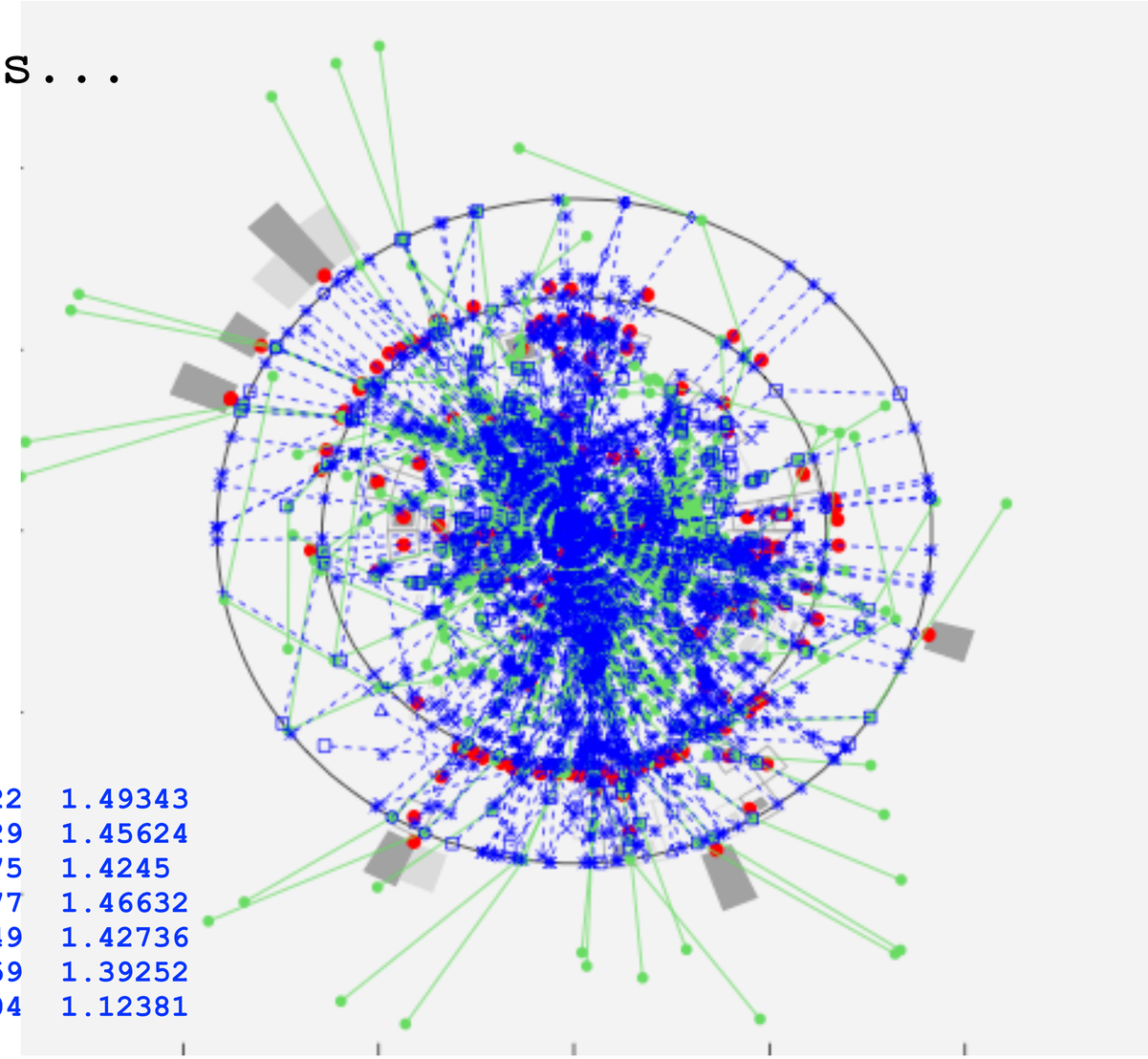
Analysis of the leading jet from simulated event of all-hadronic top-quark pairs:



That was a simple example, nevertheless...

...The Particle Flow algorithm scales to large particle multiplicities!

Analysis of the leading jet from simulated event of all-hadronic top-quark pairs:

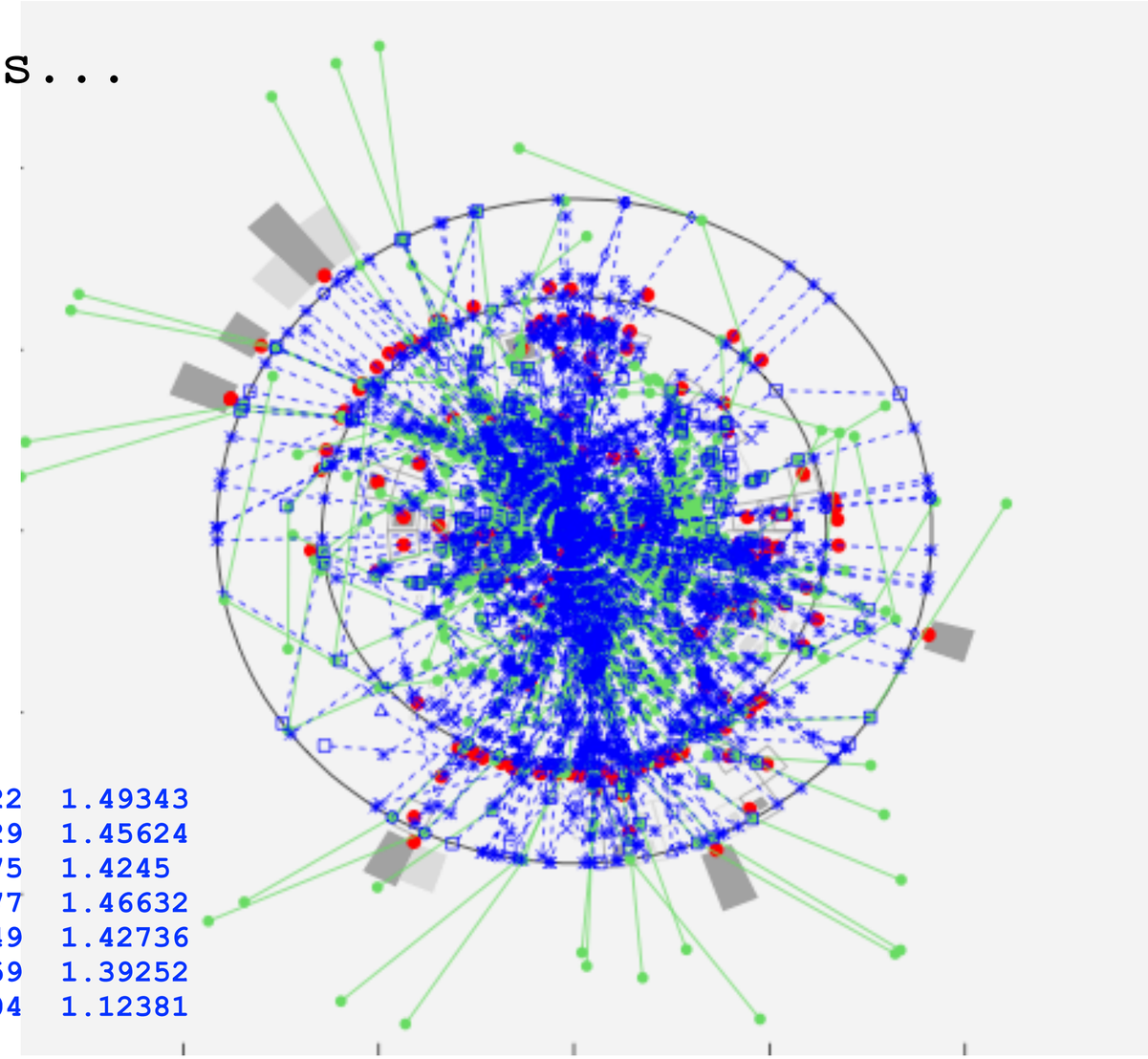


```
True Particles
    #0  PDG code:130,      p/pt/eta/phi: 20.3845  16.7688  -0.645422  1.49343
    #1  PDG code:211,      p/pt/eta/phi: 17.2954  15.0452  -0.540329  1.45624
    #2  PDG code:211,      p/pt/eta/phi: 11.453    9.82512 -0.567975  1.4245
    #3  PDG code:22,       p/pt/eta/phi: 7.75683   6.52999 -0.603777  1.46632
    #4  PDG code:22,       p/pt/eta/phi: 7.26097   6.17551 -0.584549  1.42736
    #5  PDG code:22,       p/pt/eta/phi: 6.56173   5.52903 -0.602059  1.39252
    #6  PDG code:2212,     p/pt/eta/phi: 5.69095   5.14257 -0.457804  1.12381
    ...
```

That was a simple example, nevertheless...

...The Particle Flow algorithm scales to large particle multiplicities!

Analysis of the leading jet from simulated event of all-hadronic top-quark pairs:



True Particles

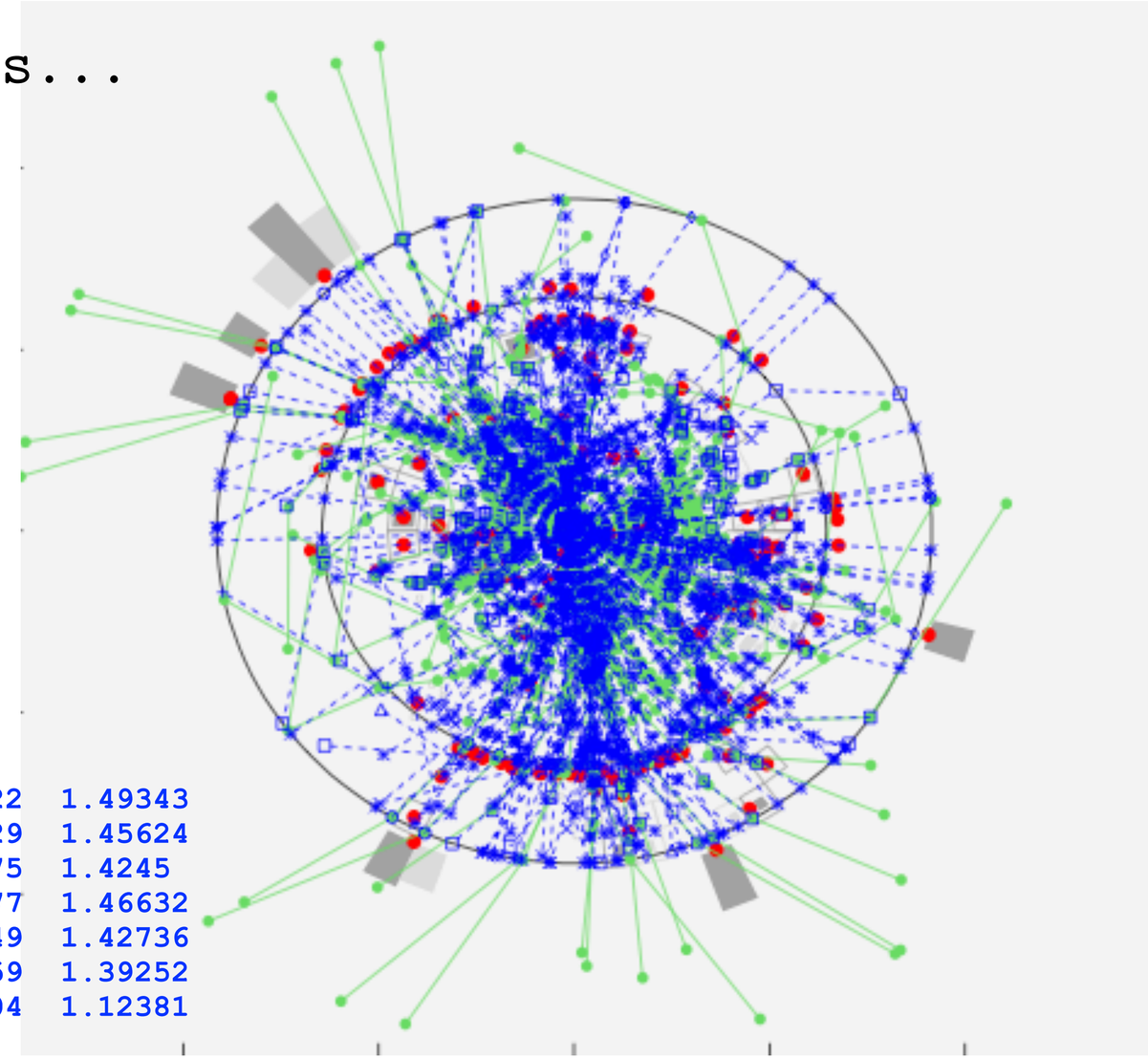
```
#0   PDG code:130,     p/pt/eta/phi: 20.3845  16.7688  -0.645422  1.49343
#1   PDG code:211,     p/pt/eta/phi: 17.2954  15.0452  -0.540329  1.45624
#2   PDG code:211,     p/pt/eta/phi: 11.453    9.82512 -0.567975  1.4245
#3   PDG code:22,      p/pt/eta/phi: 7.75683   6.52999 -0.603777  1.46632
#4   PDG code:22,      p/pt/eta/phi: 7.26097   6.17551 -0.584549  1.42736
#5   PDG code:22,      p/pt/eta/phi: 6.56173   5.52903 -0.602059  1.39252
#6   PDG code:2212,    p/pt/eta/phi: 5.69095   5.14257 -0.457804  1.12381
...
```

Reco Particles

That was a simple example, nevertheless...

...The Particle Flow algorithm scales to large particle multiplicities!

Analysis of the leading jet from simulated event of all-hadronic top-quark pairs:



True Particles

```
#0  PDG code:130,    p/pt/eta/phi: 20.3845  16.7688  -0.645422  1.49343
#1  PDG code:211,    p/pt/eta/phi: 17.2954  15.0452  -0.540329  1.45624
#2  PDG code:211,    p/pt/eta/phi: 11.453    9.82512 -0.567975  1.4245
#3  PDG code:22,     p/pt/eta/phi: 7.75683   6.52999 -0.603777  1.46632
#4  PDG code:22,     p/pt/eta/phi: 7.26097   6.17551 -0.584549  1.42736
#5  PDG code:22,     p/pt/eta/phi: 6.56173   5.52903 -0.602059  1.39252
#6  PDG code:2212,   p/pt/eta/phi: 5.69095   5.14257 -0.457804  1.12381
...
```

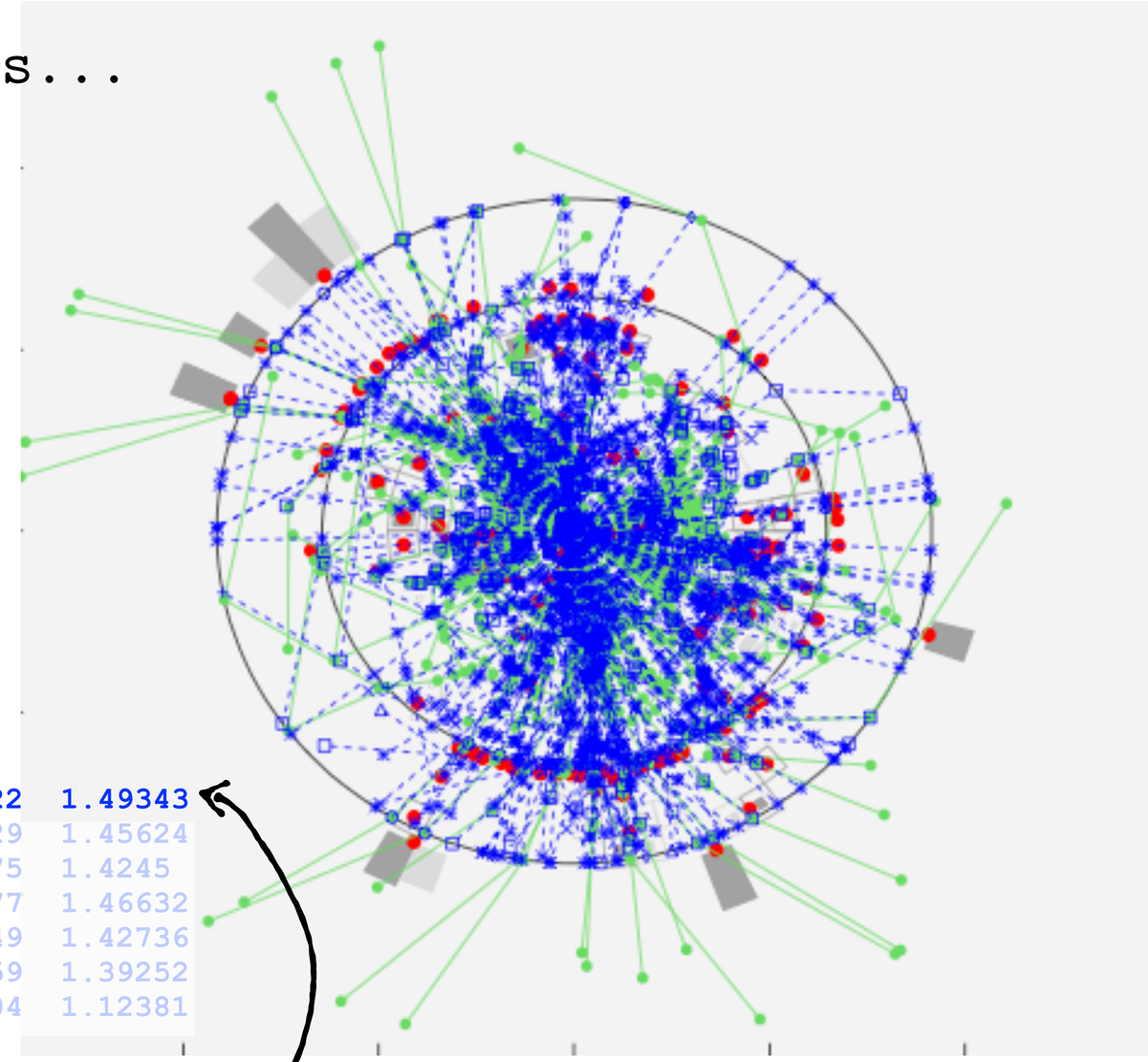
Reco Particles

```
#0  PFCandidate type: 5 E/pT/eta/phi 31.929   26.176   -0.651    1.493,
#1  PFCandidate type: 1 E/pT/eta/phi 17.237   14.994   -0.540    1.456,
#2  PFCandidate type: 1 E/pT/eta/phi 11.540    9.900   -0.568    1.425,
#3  PFCandidate type: 4 E/pT/eta/phi  9.684    8.195   -0.594    1.420,
#4  PFCandidate type: 4 E/pT/eta/phi  6.663    5.602   -0.606    1.388,
#5  PFCandidate type: 1 E/pT/eta/phi  5.720    5.170   -0.457    1.124,
...
```

That was a simple example, nevertheless...

...The Particle Flow algorithm scales to large particle multiplicities!

Analysis of the leading jet from simulated event of all-hadronic top-quark pairs:



True Particles

```
#0  PDG code:130,      p/pt/eta/phi: 20.3845  16.7688  -0.645422  1.49343
#1  PDG code:211,      p/pt/eta/phi: 17.2954  15.0452  -0.540329  1.45624
#2  PDG code:211,      p/pt/eta/phi: 11.453    9.82512 -0.567975  1.4245
#3  PDG code:22,       p/pt/eta/phi: 7.75683   6.52999 -0.603777  1.46632
#4  PDG code:22,       p/pt/eta/phi: 7.26097   6.17551 -0.584549  1.42736
#5  PDG code:22,       p/pt/eta/phi: 6.56173   5.52903 -0.602059  1.39252
#6  PDG code:2212,     p/pt/eta/phi: 5.69095   5.14257 -0.457804  1.12381
...
```

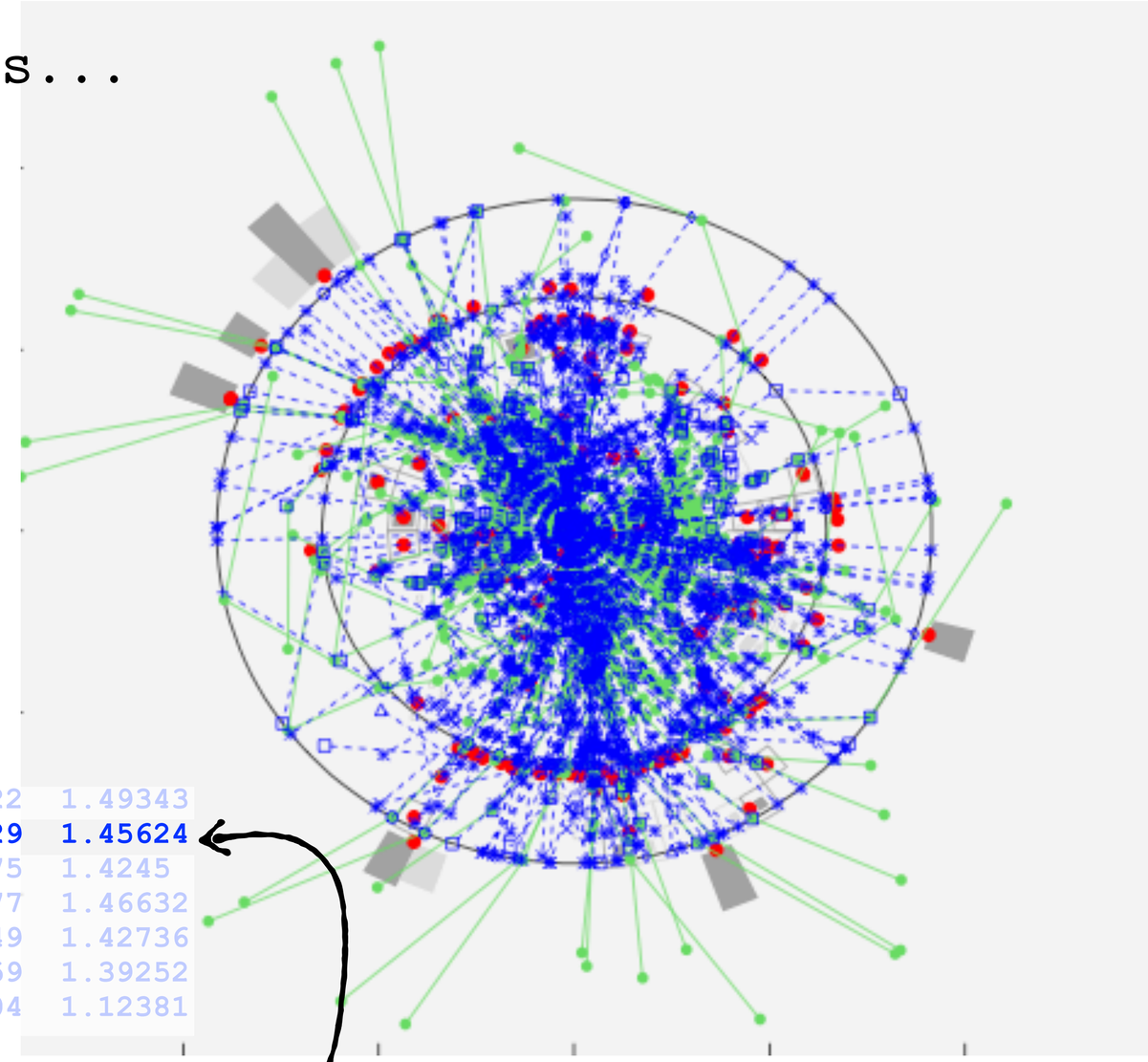
Reco Particles

```
#0  PFCandidate type: 5 E/pT/eta/phi 31.929    26.176   -0.651     1.493,
#1  PFCandidate type: 1 E/pT/eta/phi 17.237    14.994   -0.540     1.456,
#2  PFCandidate type: 1 E/pT/eta/phi 11.540     9.900   -0.568     1.425,
#3  PFCandidate type: 4 E/pT/eta/phi  9.684     8.195   -0.594     1.420,
#4  PFCandidate type: 4 E/pT/eta/phi  6.663     5.602   -0.606     1.388,
#5  PFCandidate type: 1 E/pT/eta/phi  5.720     5.170   -0.457     1.124,
...
```

That was a simple example, nevertheless...

...The Particle Flow algorithm scales to large particle multiplicities!

Analysis of the leading jet from simulated event of all-hadronic top-quark pairs:



True Particles

```
#0  PDG code:130,    p/pt/eta/phi: 20.3845  16.7688  -0.645422  1.49343
#1  PDG code:211,    p/pt/eta/phi: 17.2954  15.0452  -0.540329  1.45624
#2  PDG code:211,    p/pt/eta/phi: 11.453    9.82512 -0.567975  1.4245
#3  PDG code:22,     p/pt/eta/phi: 7.75683   6.52999 -0.603777  1.46632
#4  PDG code:22,     p/pt/eta/phi: 7.26097   6.17551 -0.584549  1.42736
#5  PDG code:22,     p/pt/eta/phi: 6.56173   5.52903 -0.602059  1.39252
#6  PDG code:2212,   p/pt/eta/phi: 5.69095   5.14257 -0.457804  1.12381
...
```

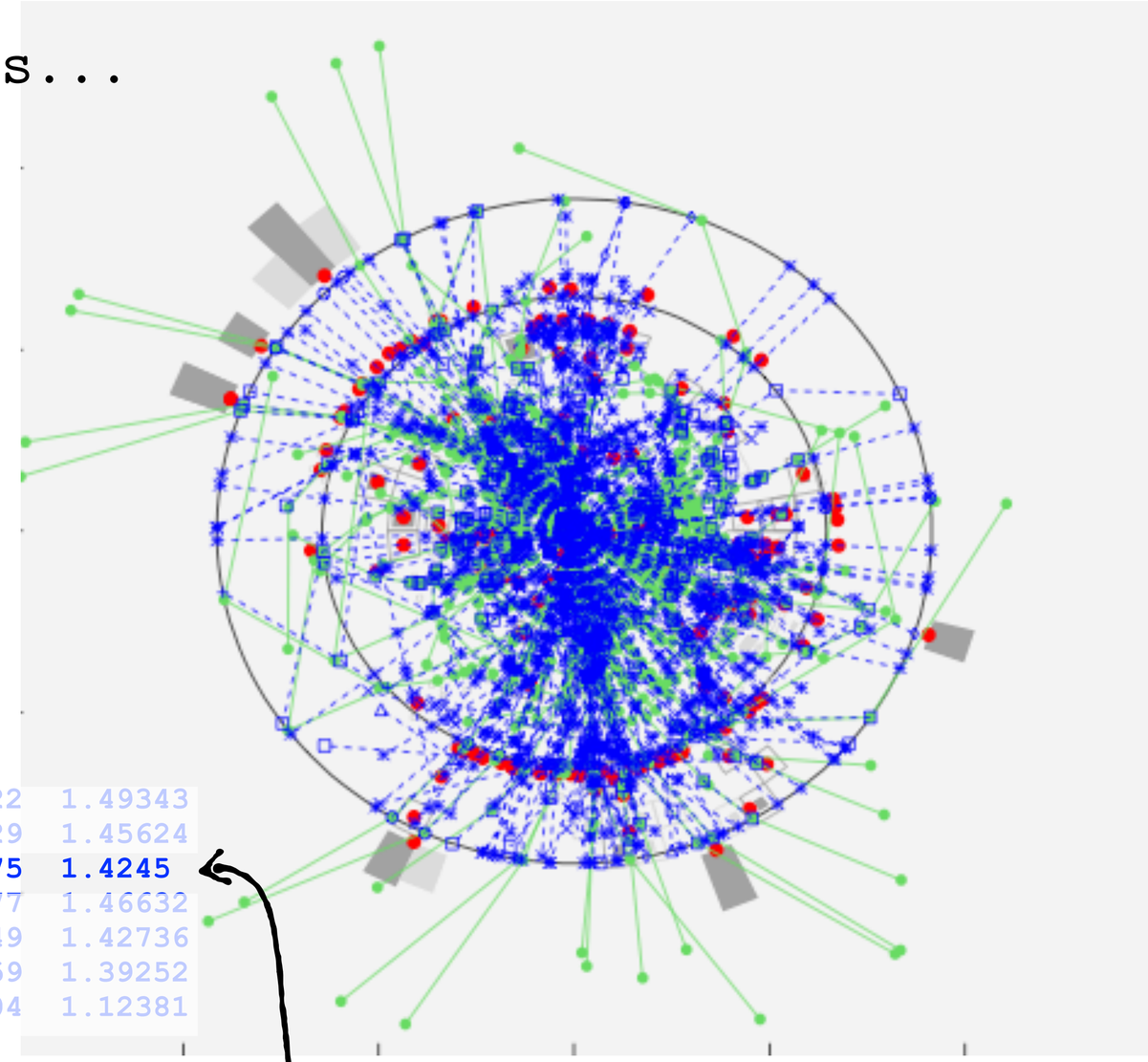
Reco Particles

```
#0  PFCandidate type: 5 E/pT/eta/phi 31.929   26.176   -0.651    1.493,
#1  PFCandidate type: 1 E/pT/eta/phi 17.237   14.994   -0.540    1.456,
#2  PFCandidate type: 1 E/pT/eta/phi 11.540    9.900   -0.568    1.425,
#3  PFCandidate type: 4 E/pT/eta/phi  9.684    8.195   -0.594    1.420,
#4  PFCandidate type: 4 E/pT/eta/phi  6.663    5.602   -0.606    1.388,
#5  PFCandidate type: 1 E/pT/eta/phi  5.720    5.170   -0.457    1.124,
...
```

That was a simple example, nevertheless...

...The Particle Flow algorithm scales to large particle multiplicities!

Analysis of the leading jet from simulated event of all-hadronic top-quark pairs:



True Particles

```
#0  PDG code:130,      p/pt/eta/phi: 20.3845  16.7688  -0.645422  1.49343
#1  PDG code:211,      p/pt/eta/phi: 17.2954  15.0452  -0.540329  1.45624
#2  PDG code:211,      p/pt/eta/phi: 11.453    9.82512 -0.567975  1.4245
#3  PDG code:22,       p/pt/eta/phi: 7.75683   6.52999 -0.603777  1.46632
#4  PDG code:22,       p/pt/eta/phi: 7.26097   6.17551 -0.584549  1.42736
#5  PDG code:22,       p/pt/eta/phi: 6.56173   5.52903 -0.602059  1.39252
#6  PDG code:2212,     p/pt/eta/phi: 5.69095   5.14257 -0.457804  1.12381
...
```

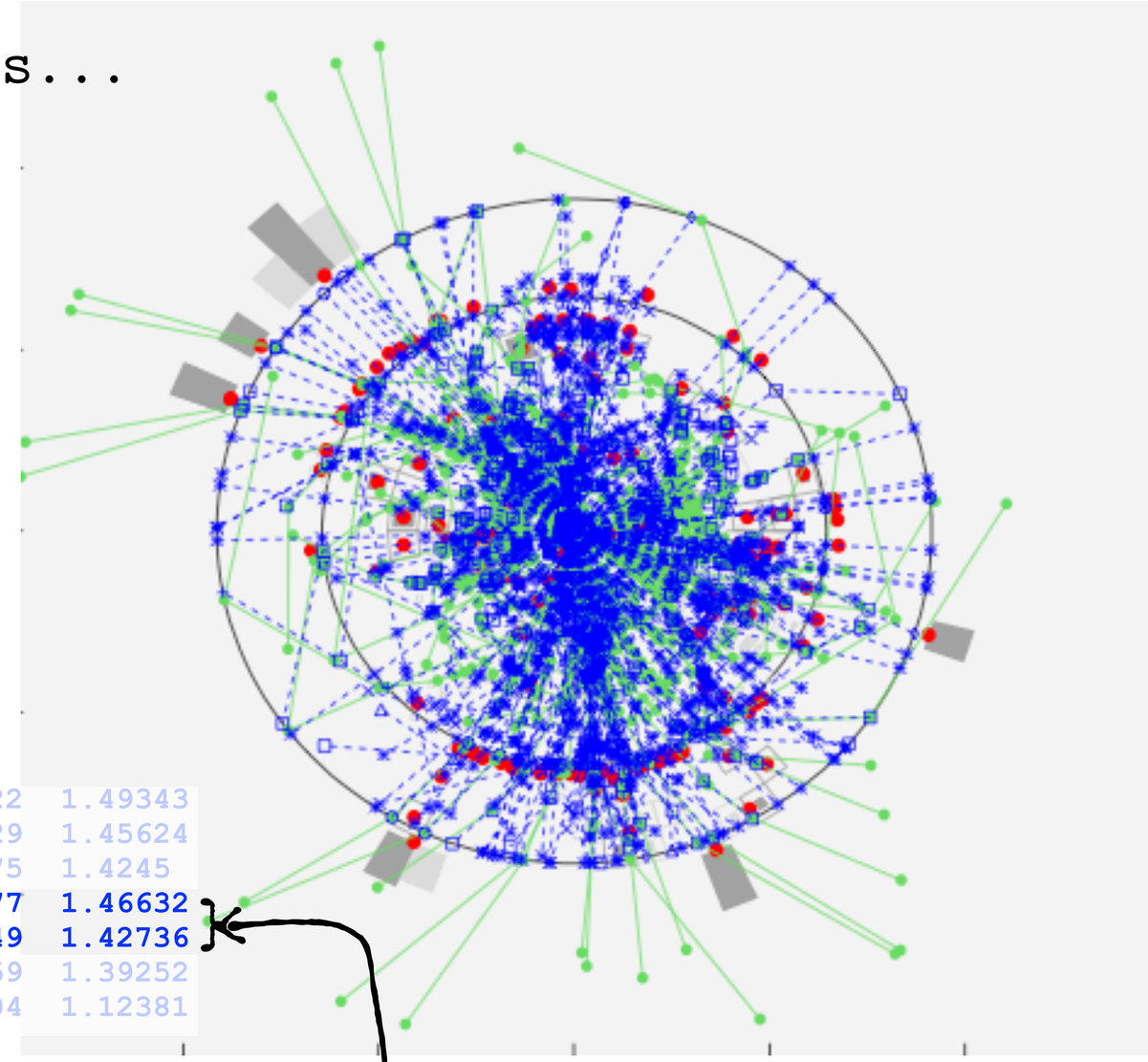
Reco Particles

```
#0  PFCandidate type: 5 E/pT/eta/phi 31.929    26.176   -0.651     1.493,
#1  PFCandidate type: 1 E/pT/eta/phi 17.237    14.994   -0.540     1.456,
#2  PFCandidate type: 1 E/pT/eta/phi 11.540     9.900   -0.568     1.425,
#3  PFCandidate type: 4 E/pT/eta/phi  9.684     8.195   -0.594     1.420,
#4  PFCandidate type: 4 E/pT/eta/phi  6.663     5.602   -0.606     1.388,
#5  PFCandidate type: 1 E/pT/eta/phi  5.720     5.170   -0.457     1.124,
...
```

That was a simple example, nevertheless...

...The Particle Flow algorithm scales to large particle multiplicities!

Analysis of the leading jet from simulated event of all-hadronic top-quark pairs:



True Particles

```
#0  PDG code:130,    p/pt/eta/phi: 20.3845  16.7688  -0.645422  1.49343
#1  PDG code:211,    p/pt/eta/phi: 17.2954  15.0452  -0.540329  1.45624
#2  PDG code:211,    p/pt/eta/phi: 11.453    9.82512 -0.567975  1.4245
#3  PDG code:22,     p/pt/eta/phi: 7.75683   6.52999 -0.603777  1.46632
#4  PDG code:22,     p/pt/eta/phi: 7.26097   6.17551 -0.584549  1.42736
#5  PDG code:22,     p/pt/eta/phi: 6.56173   5.52903 -0.602059  1.39252
#6  PDG code:2212,   p/pt/eta/phi: 5.69095   5.14257 -0.457804  1.12381
...
```

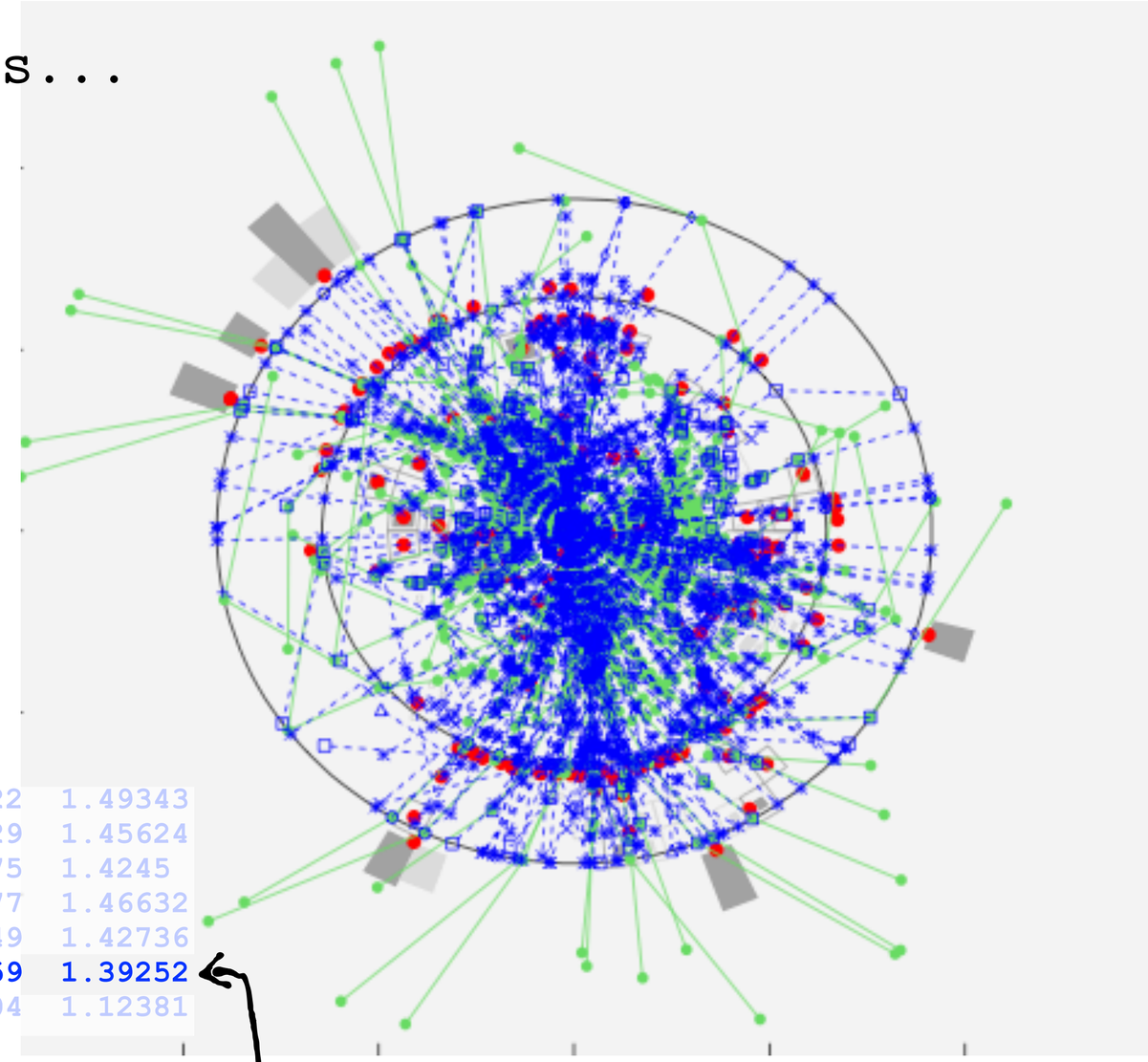
Reco Particles

```
#0  PFCandidate type: 5 E/pT/eta/phi 31.929   26.176   -0.651    1.493,
#1  PFCandidate type: 1 E/pT/eta/phi 17.237   14.994   -0.540    1.456,
#2  PFCandidate type: 1 E/pT/eta/phi 11.540    9.900   -0.568    1.425,
#3  PFCandidate type: 4 E/pT/eta/phi  9.684    8.195   -0.594    1.420,
#4  PFCandidate type: 4 E/pT/eta/phi  6.663    5.602   -0.606    1.388,
#5  PFCandidate type: 1 E/pT/eta/phi  5.720    5.170   -0.457    1.124,
...
```

That was a simple example, nevertheless...

...The Particle Flow algorithm scales to large particle multiplicities!

Analysis of the leading jet from simulated event of all-hadronic top-quark pairs:



True Particles

```
#0  PDG code:130,    p/pt/eta/phi: 20.3845  16.7688  -0.645422  1.49343
#1  PDG code:211,    p/pt/eta/phi: 17.2954  15.0452  -0.540329  1.45624
#2  PDG code:211,    p/pt/eta/phi: 11.453    9.82512 -0.567975  1.4245
#3  PDG code:22,     p/pt/eta/phi: 7.75683   6.52999 -0.603777  1.46632
#4  PDG code:22,     p/pt/eta/phi: 7.26097   6.17551 -0.584549  1.42736
#5  PDG code:22,     p/pt/eta/phi: 6.56173   5.52903 -0.602059  1.39252
#6  PDG code:2212,   p/pt/eta/phi: 5.69095   5.14257 -0.457804  1.12381
...
```

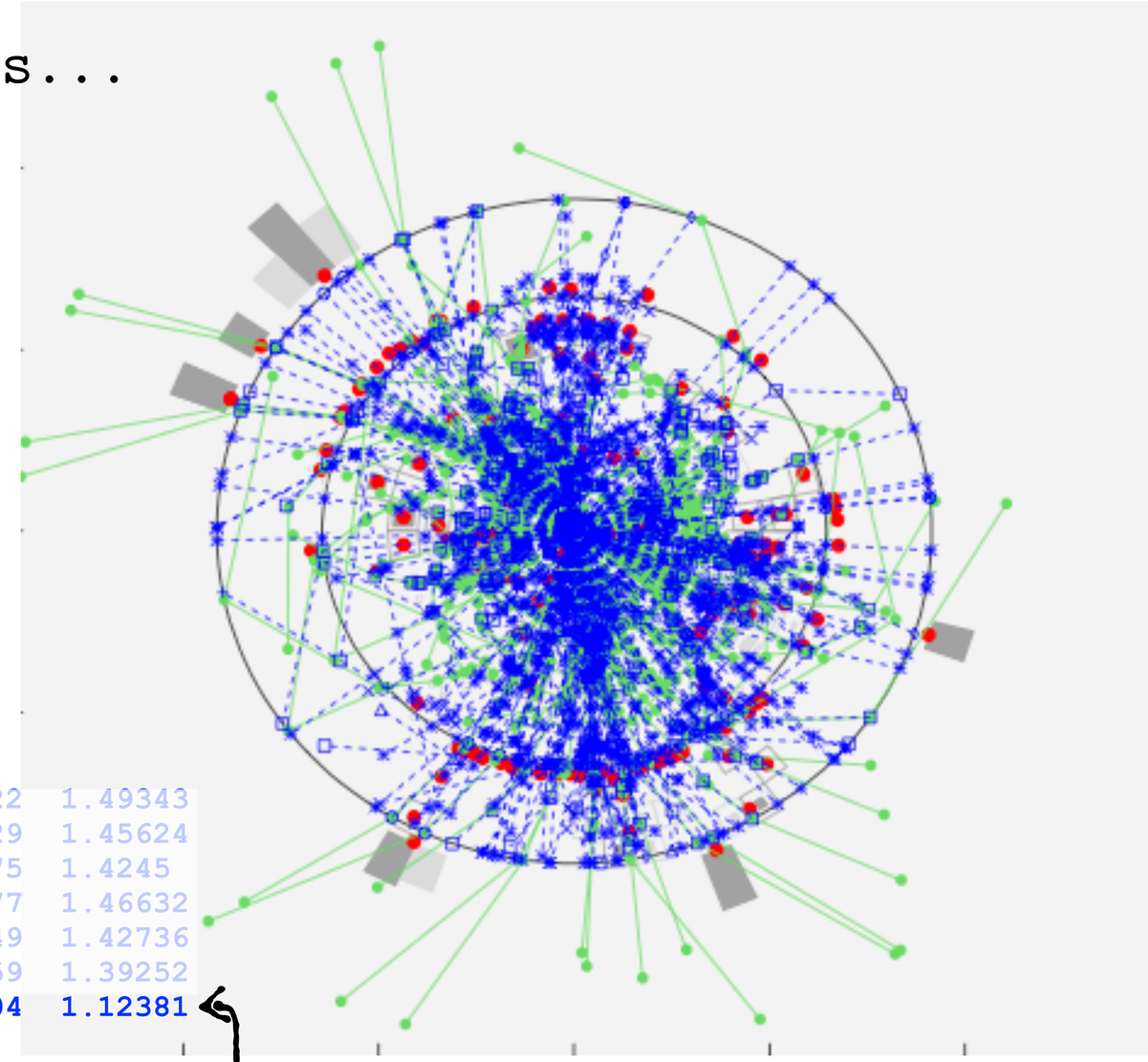
Reco Particles

```
#0  PFCandidate type: 5 E/pT/eta/phi 31.929   26.176   -0.651    1.493,
#1  PFCandidate type: 1 E/pT/eta/phi 17.237   14.994   -0.540    1.456,
#2  PFCandidate type: 1 E/pT/eta/phi 11.540    9.900   -0.568    1.425,
#3  PFCandidate type: 4 E/pT/eta/phi  9.684    8.195   -0.594    1.420,
#4  PFCandidate type: 4 E/pT/eta/phi  6.663    5.602   -0.606    1.388,
#5  PFCandidate type: 1 E/pT/eta/phi  5.720    5.170   -0.457    1.124,
...
```

That was a simple example, nevertheless...

...The Particle Flow algorithm scales to large particle multiplicities!

Analysis of the leading jet from simulated event of all-hadronic top-quark pairs:



True Particles

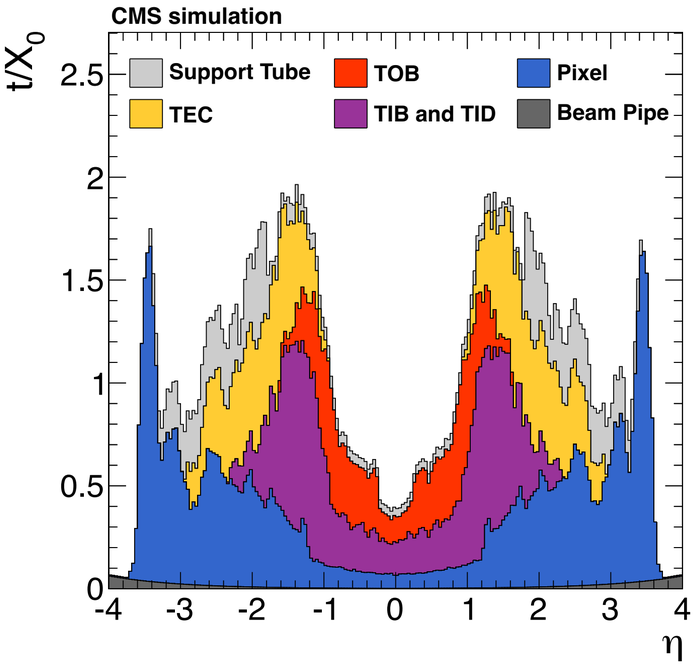
```
#0  PDG code:130,    p/pt/eta/phi: 20.3845  16.7688  -0.645422  1.49343
#1  PDG code:211,    p/pt/eta/phi: 17.2954  15.0452  -0.540329  1.45624
#2  PDG code:211,    p/pt/eta/phi: 11.453    9.82512 -0.567975  1.4245
#3  PDG code:22,     p/pt/eta/phi: 7.75683   6.52999 -0.603777  1.46632
#4  PDG code:22,     p/pt/eta/phi: 7.26097   6.17551 -0.584549  1.42736
#5  PDG code:22,     p/pt/eta/phi: 6.56173   5.52903 -0.602059  1.39252
#6  PDG code:2212,   p/pt/eta/phi: 5.69095   5.14257 -0.457804  1.12381
...
```

Reco Particles

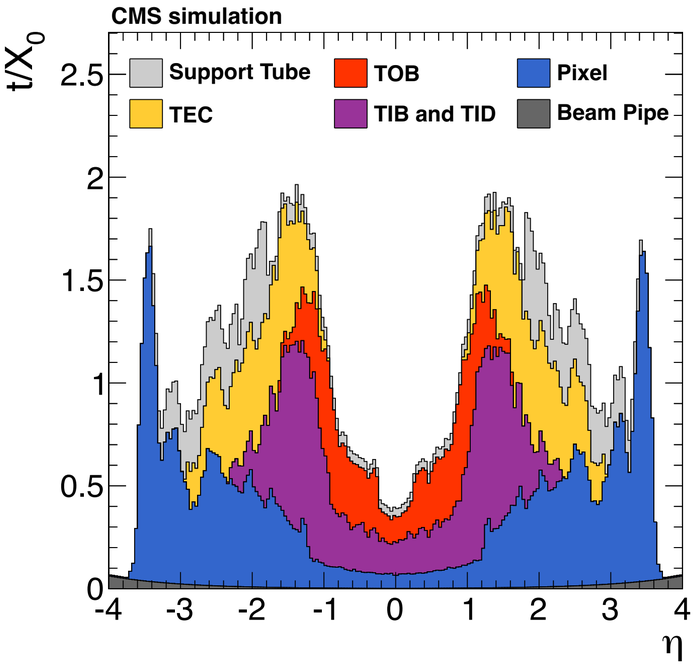
```
#0  PFCandidate type: 5 E/pT/eta/phi 31.929   26.176   -0.651    1.493,
#1  PFCandidate type: 1 E/pT/eta/phi 17.237   14.994   -0.540    1.456,
#2  PFCandidate type: 1 E/pT/eta/phi 11.540    9.900   -0.568    1.425,
#3  PFCandidate type: 4 E/pT/eta/phi  9.684    8.195   -0.594    1.420,
#4  PFCandidate type: 4 E/pT/eta/phi  6.663    5.602   -0.606    1.388,
#5  PFCandidate type: 1 E/pT/eta/phi  5.720    5.170   -0.457    1.124,
...
```

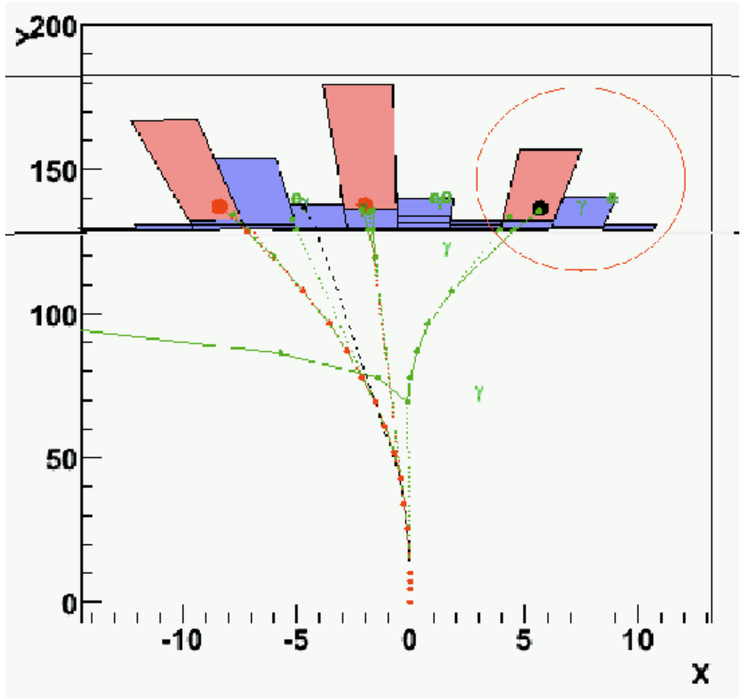
Important details (again!) Tracker Material Budget

Important details (again!) Tracker Material Budget

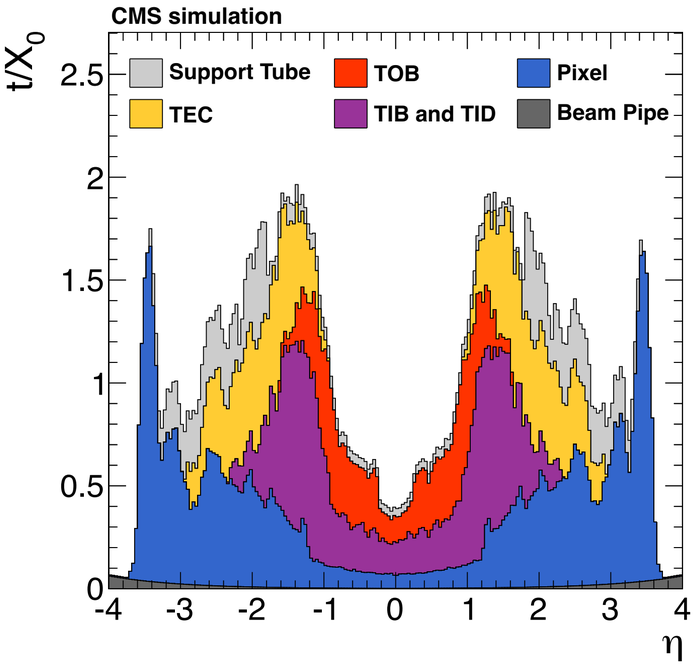


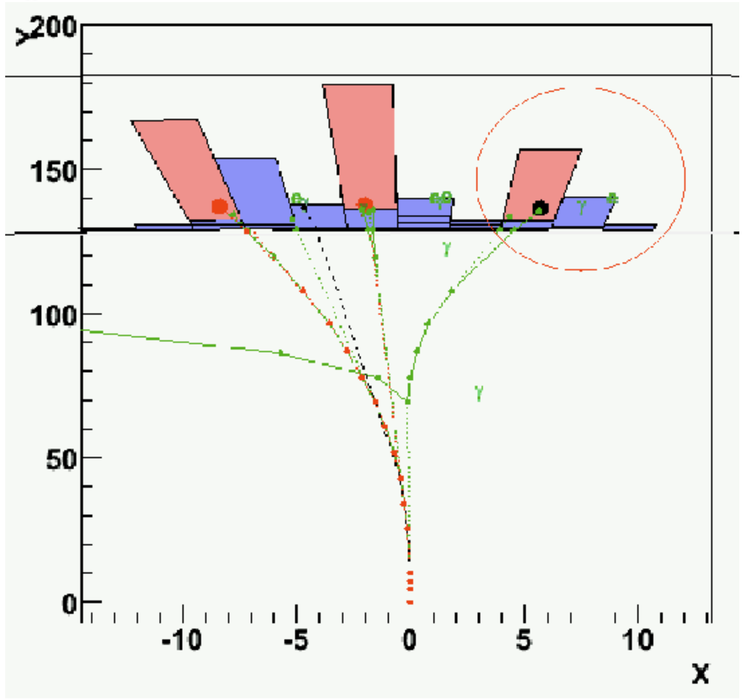
Important details (again!) Tracker Material Budget

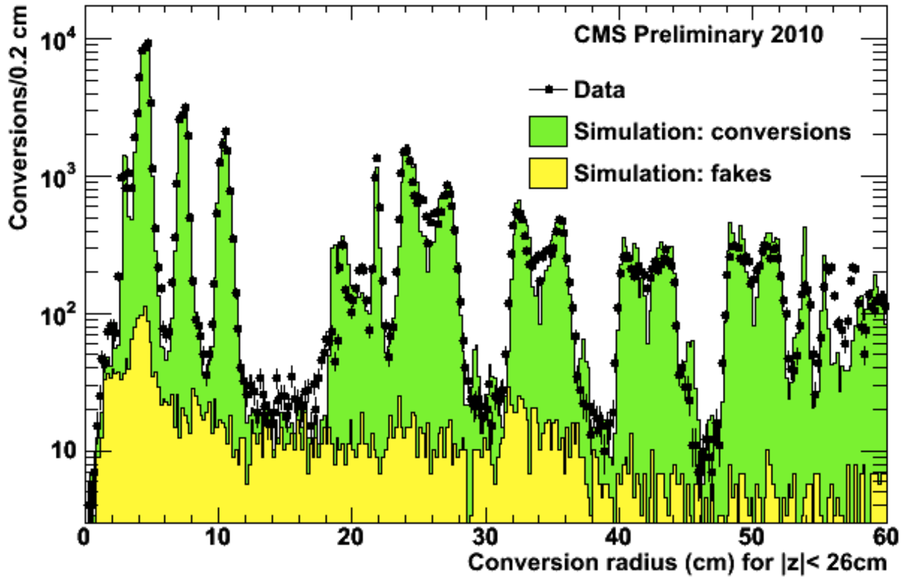




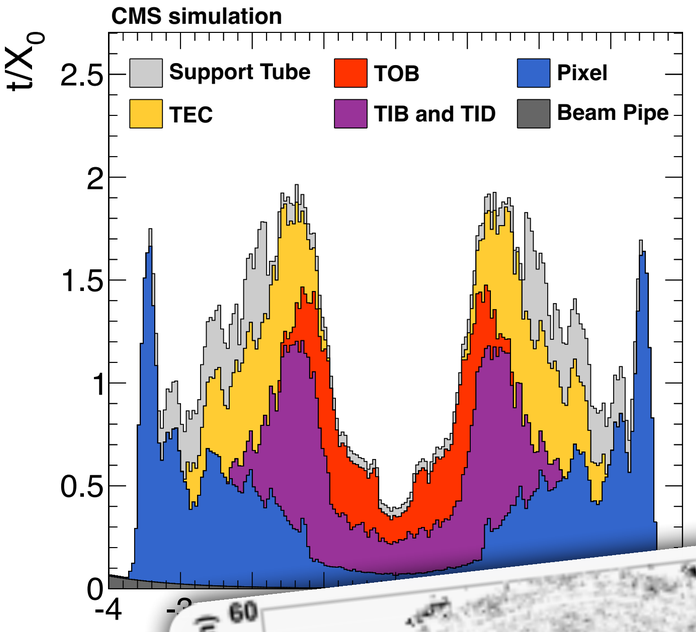
Important details (again!) Tracker Material Budget

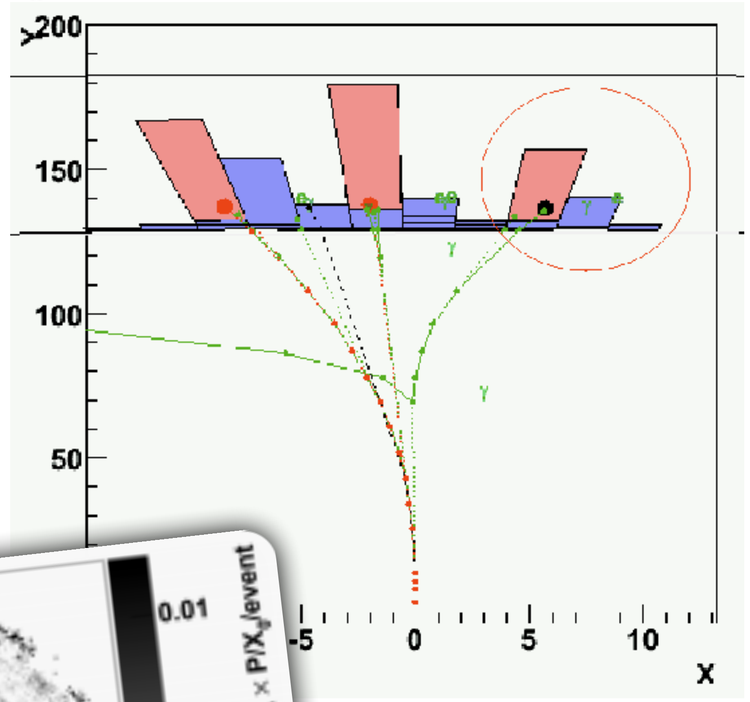


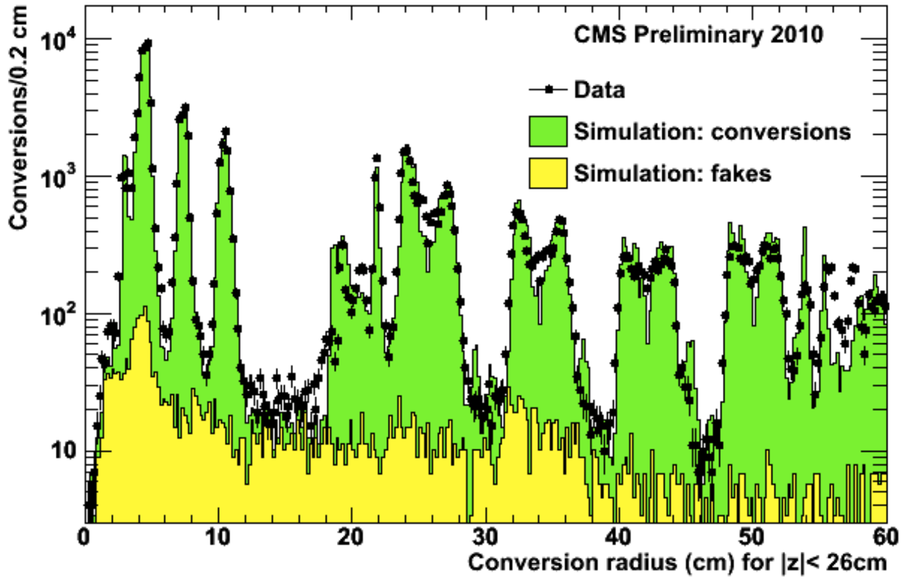


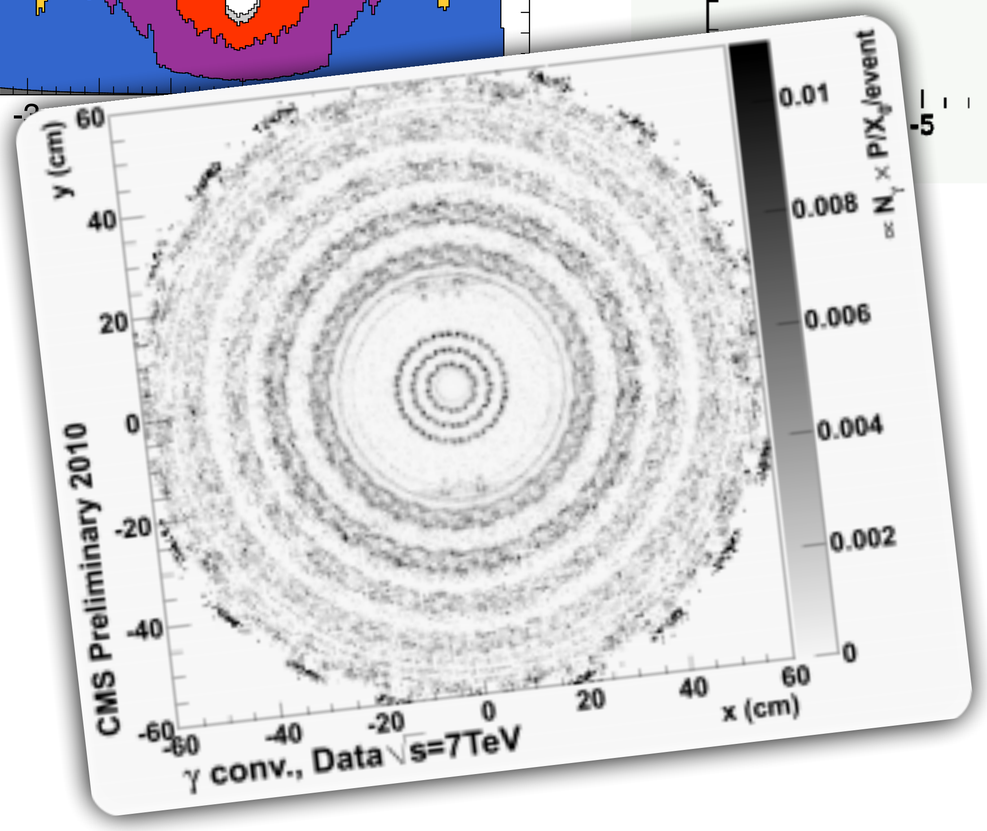


Important details (again!) Tracker Material Budget

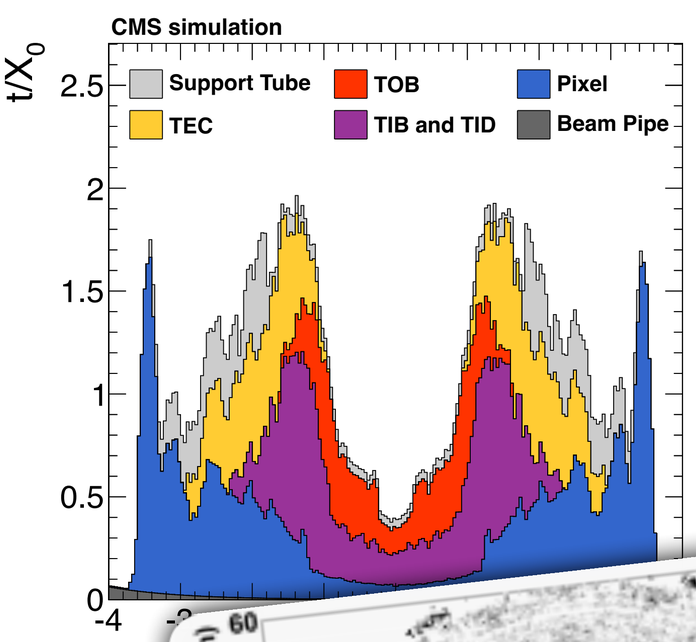


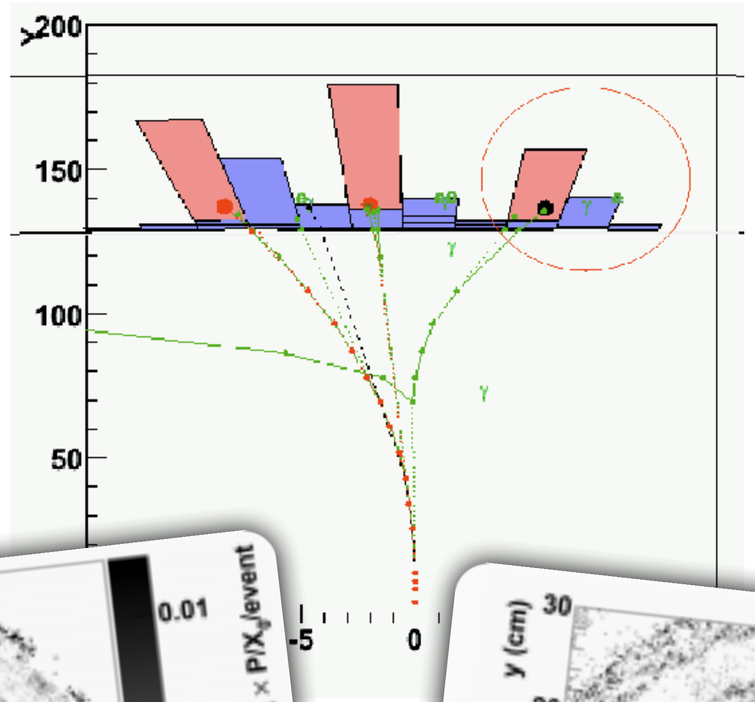


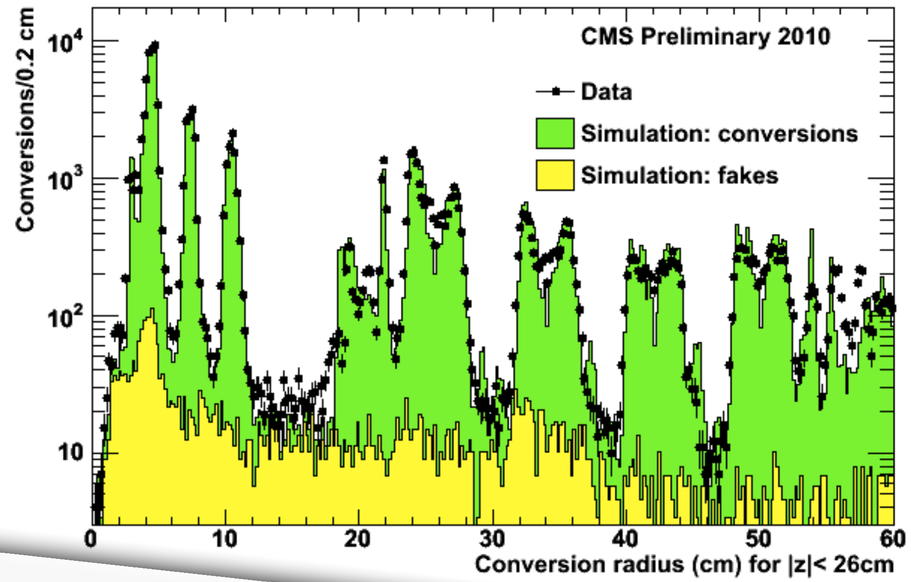


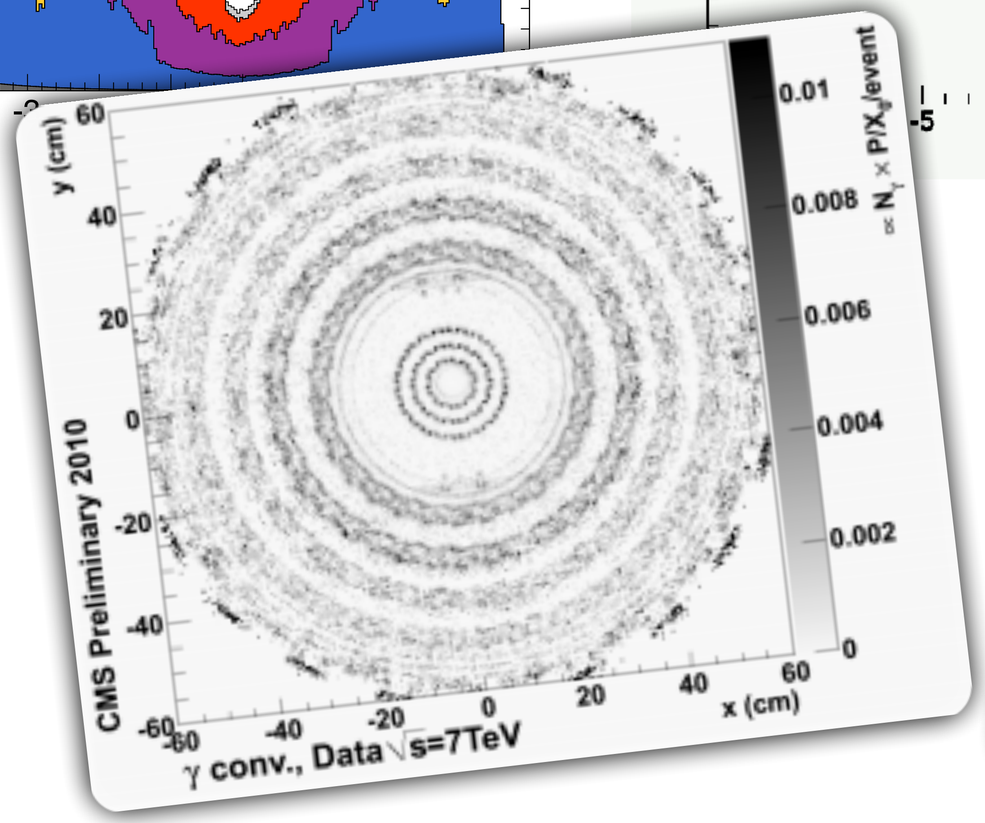


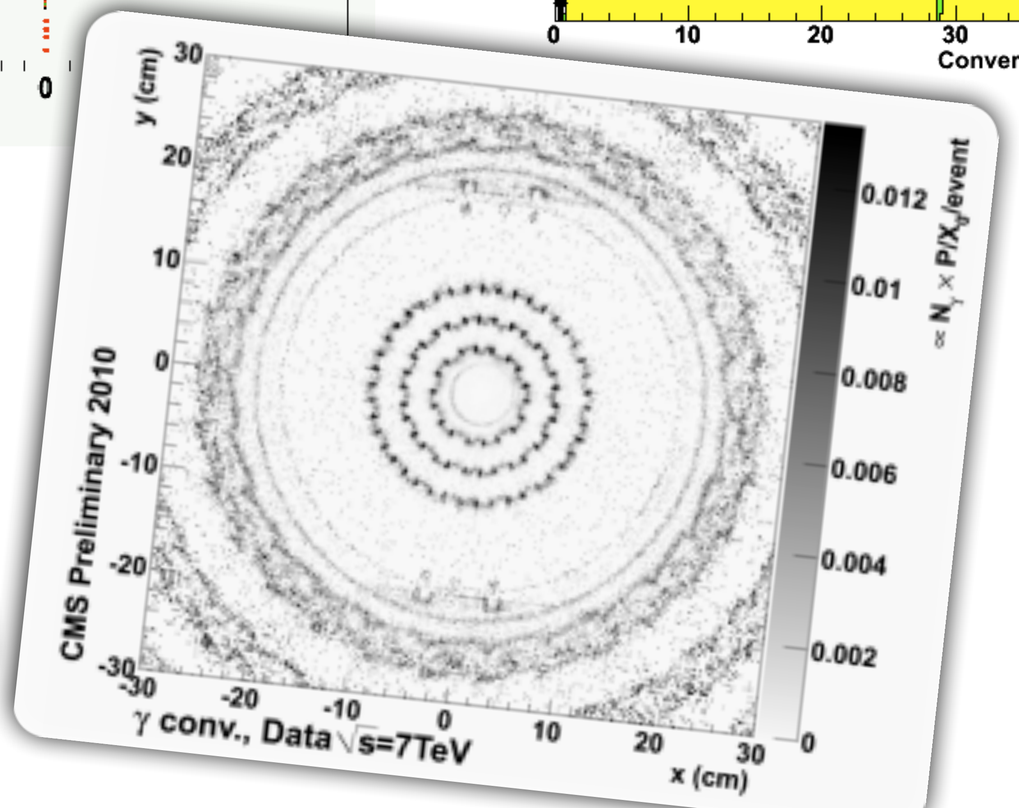
Important details (again!) Tracker Material Budget



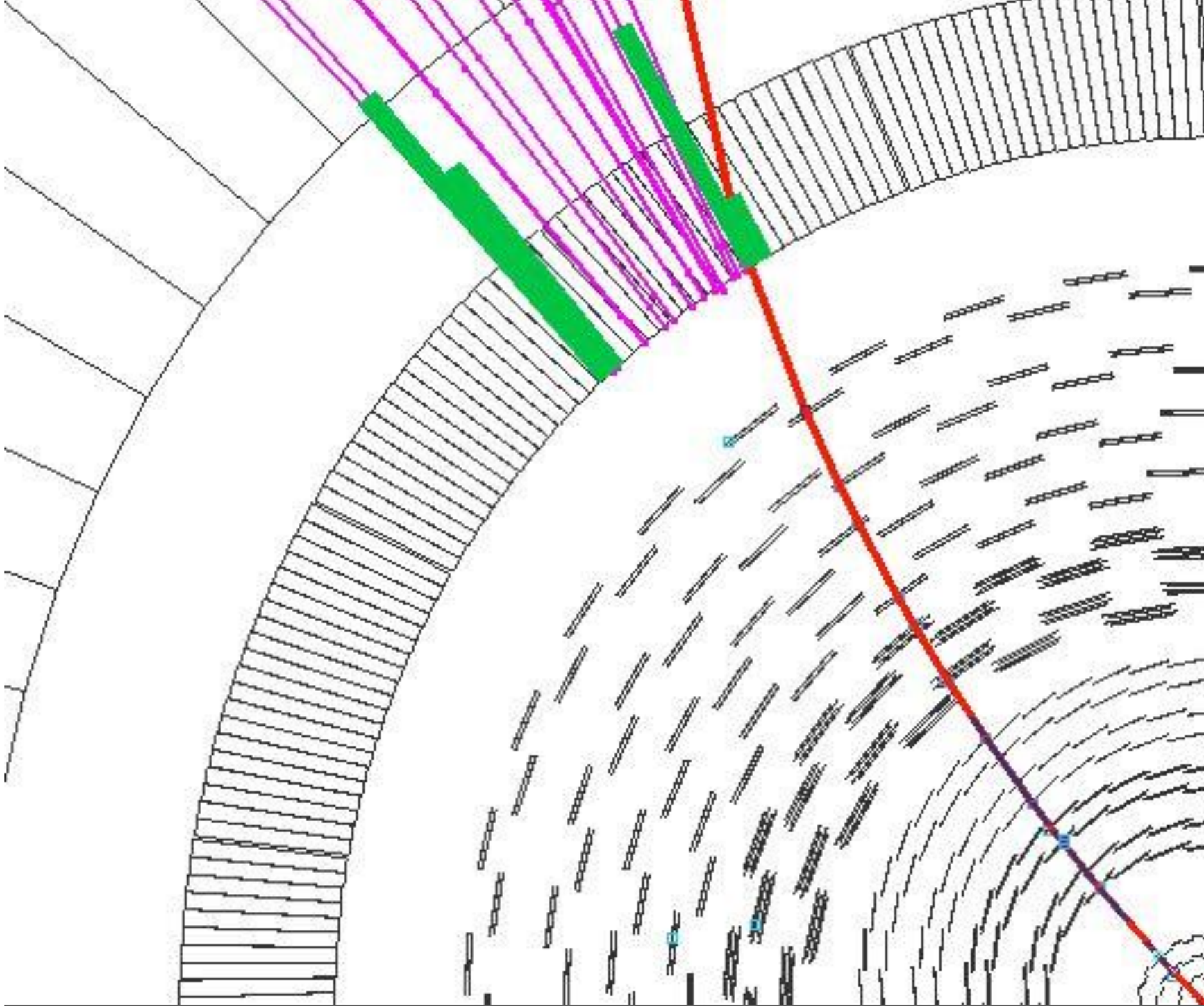








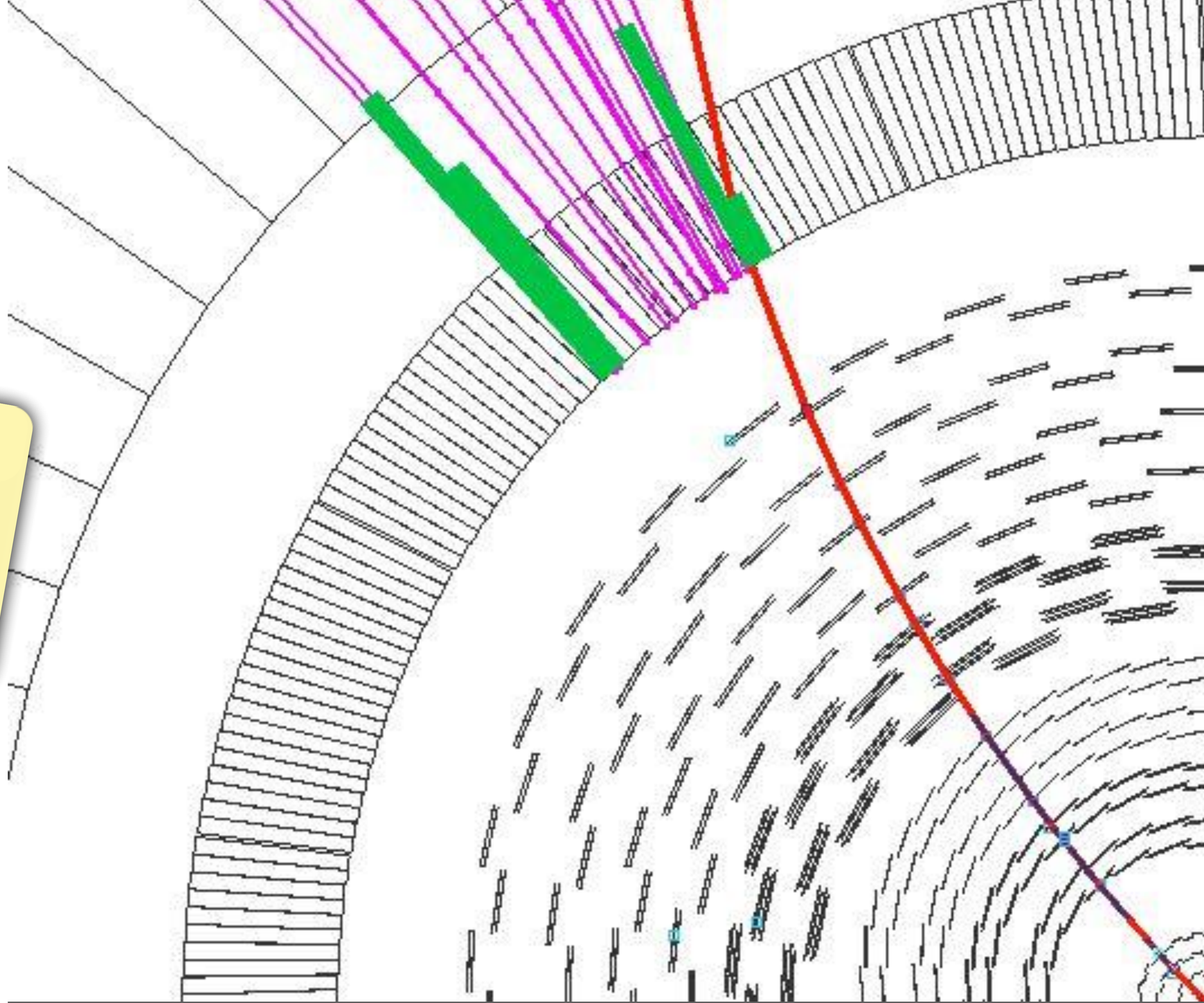
Electron ID



Electron ID

Huge Effect!

- Electrons radiate on average $\approx$ 70% of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field



Electron ID

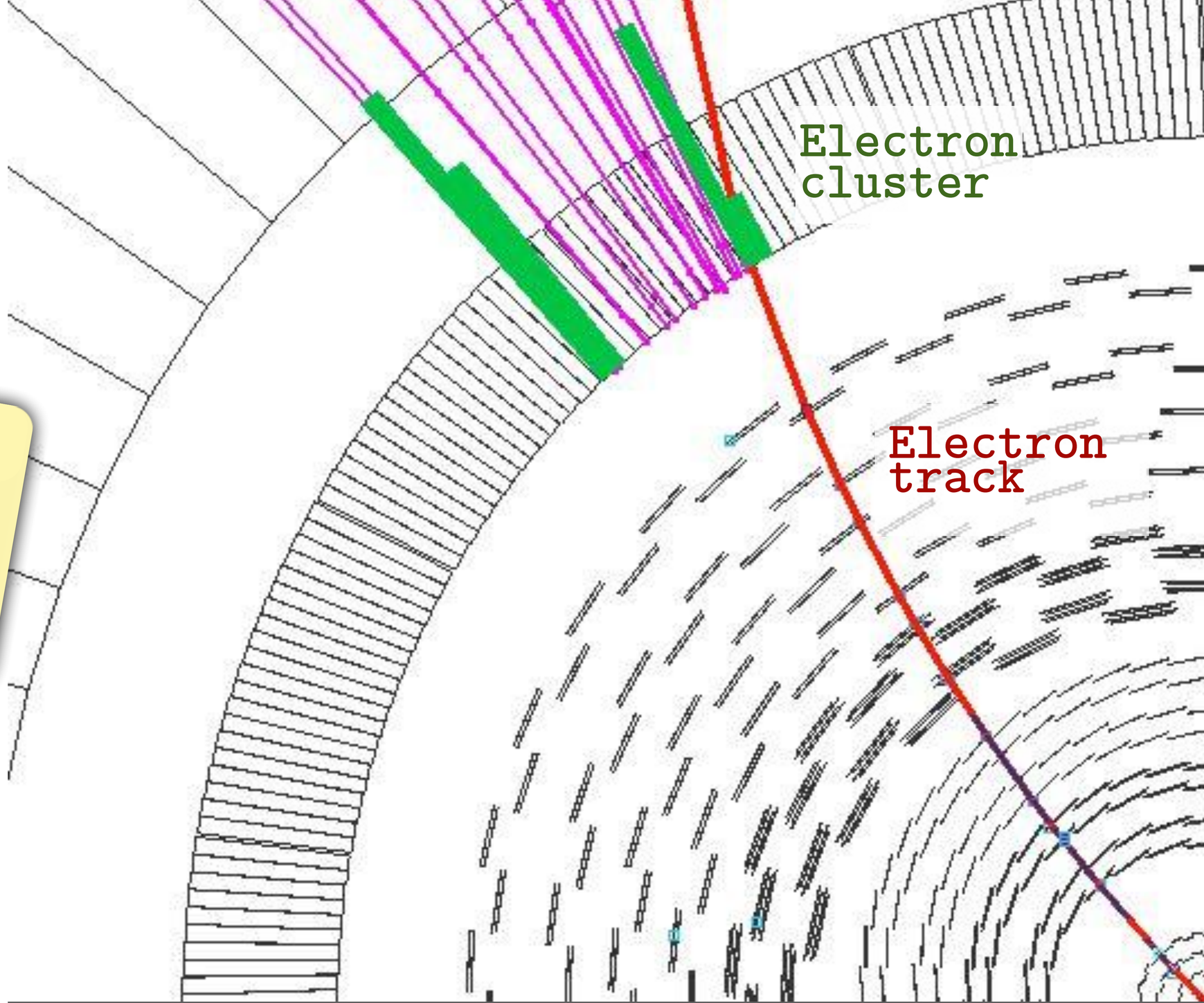
Huge Effect!

- Electrons radiate on average $\approx$ 70% of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field

Electron ID

Huge Effect!

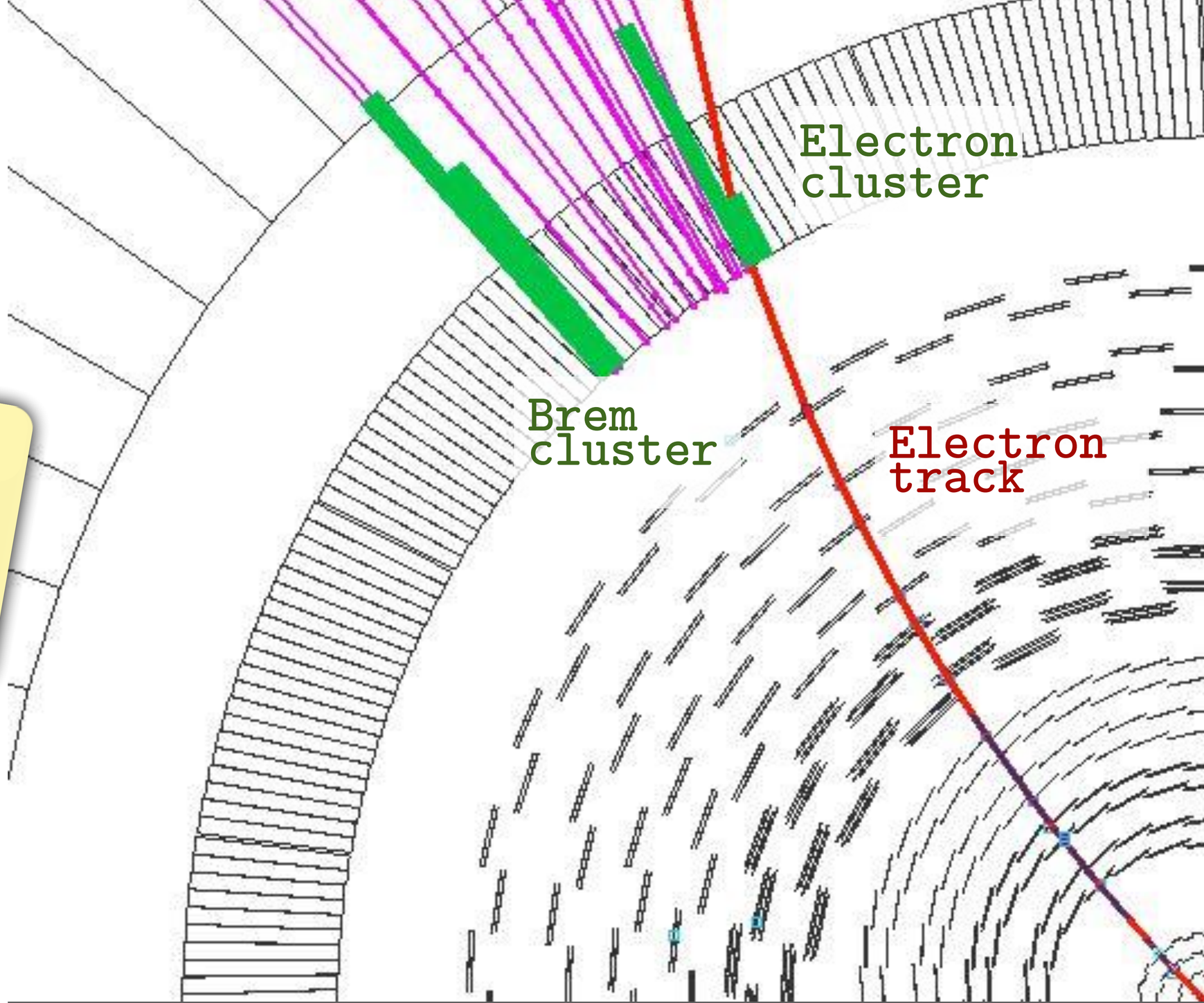
- Electrons radiate on average ≈ 70% of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field



Electron ID

Huge Effect!

- Electrons radiate on average ≈ 70% of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field



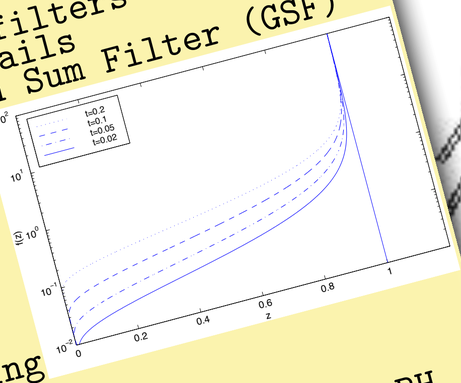
Electron ID

Huge Effect!

- Electrons radiate on average $\approx \frac{2}{3}$ of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field

Standard KF tracking fails for electrons!

- Because they radiate! Kalman filter pattern recognition quickly gives up
- Use sum of Kalman filters to account for non-Gaussian tails
  => Gaussian Sum Filter (GSF)
  - attempts to approximates Bethe-Heitler energy loss
  - resembles a set of Kalman filters running in parallel
    - each corresponding to one gaussian component describing BH energy loss
    - Computationally expensive!

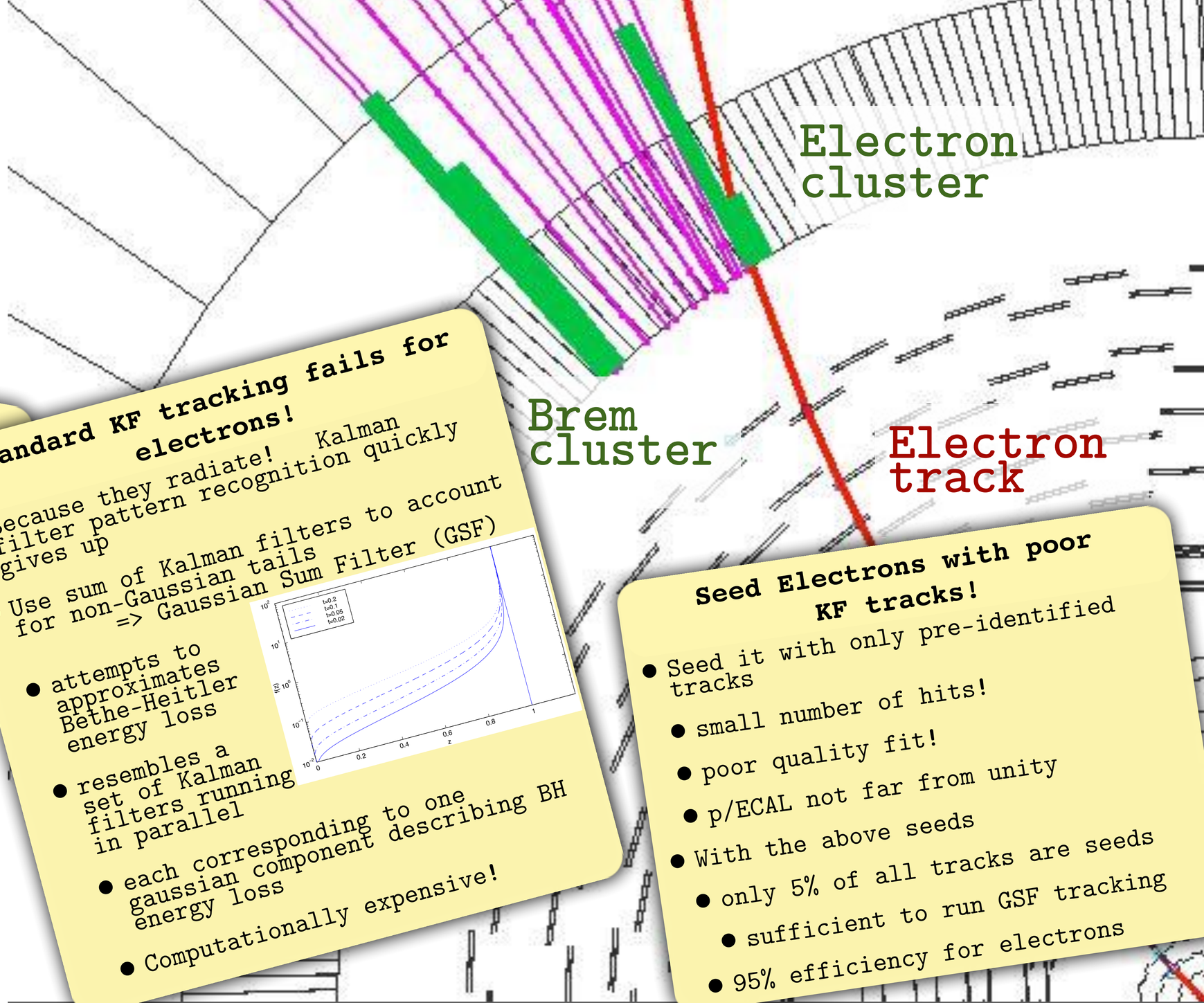


Electron cluster

Brem cluster

Electron track

Electron ID



Huge Effect!

- Electrons radiate on average $\approx \frac{2}{3}$ of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field

Standard KF tracking fails for electrons!

- Because they radiate! Kalman filter pattern recognition quickly gives up
- Use sum of Kalman filters to account for non-Gaussian tails
  => Gaussian Sum Filter (GSF)
  - attempts to approximates Bethe-Heitler energy loss
  - resembles a set of Kalman filters running in parallel
    - each corresponding to one gaussian component describing BH energy loss
    - Computationally expensive!

Seed Electrons with poor KF tracks!

- Seed it with only pre-identified tracks
  - small number of hits!
  - poor quality fit!
  - p/ECAL not far from unity
- With the above seeds
  - only 5% of all tracks are seeds
    - sufficient to run GSF tracking
  - 95% efficiency for electrons

Electron ID

New type of link: "by tangent"

- Violet lines are tangents to the GSF track
  - starting from each tracker layer

If tangent points to a ECAL cluster

- link cluster to track

Another possible handle

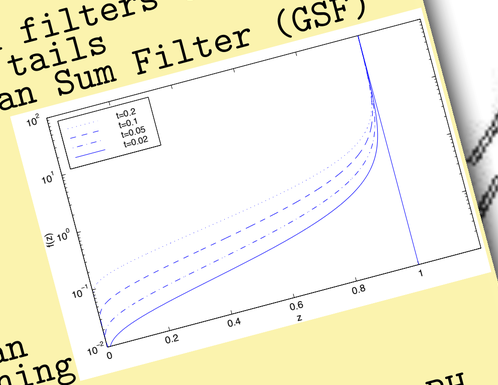
- test compatibility between ECAL cluster and Δp along GSF track

Huge Effect!

- Electrons radiate on average ≈ ⅓ of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field

Standard KF tra… fails for electrons!

- Because they radiate! Kalman filter pattern recognition quickly gives up
- Use sum of Kalman filters to account for non-Gaussian tails => Gaussian Sum Filter (GSF)
  - attempts to approximates Bethe-Heitler energy loss
  - resembles a set of Kalman filters running in parallel
    - each corresponding to one gaussian component describing BH energy loss
    - Computationally expensive!



Seed Electrons with poor KF tracks!

- Seed it with only pre-identified tracks
  - small number of hits!
  - poor quality fit!
  - p/ECAL not far from unity
- With the above seeds
  - only 5% of all tracks are seeds
    - sufficient to run GSF tracking
  - 95% efficiency for electrons

Electron cluster

Brem cluster

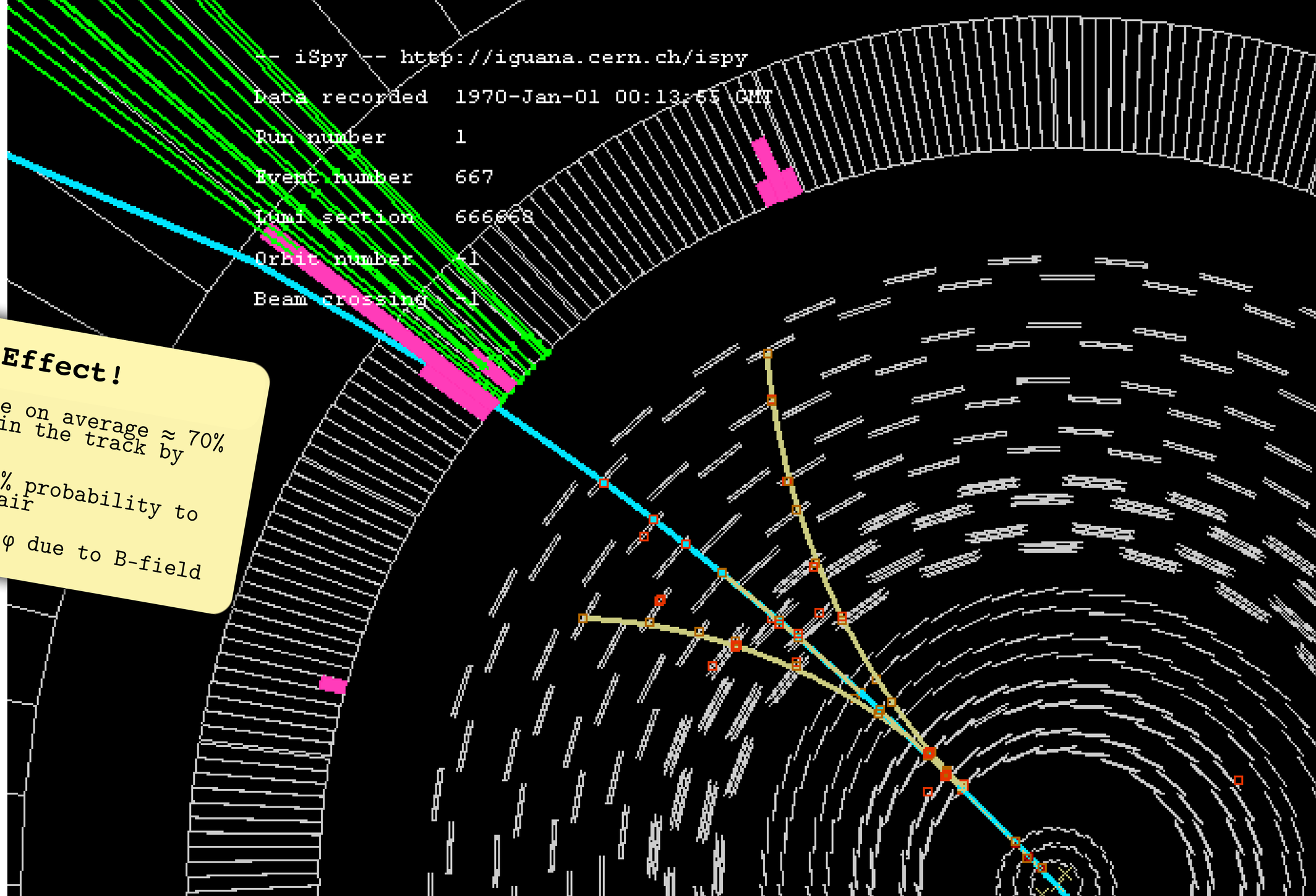
Electron track

Electron ID

Huge Effect!

- Electrons radiate on average ≈ 70% of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field

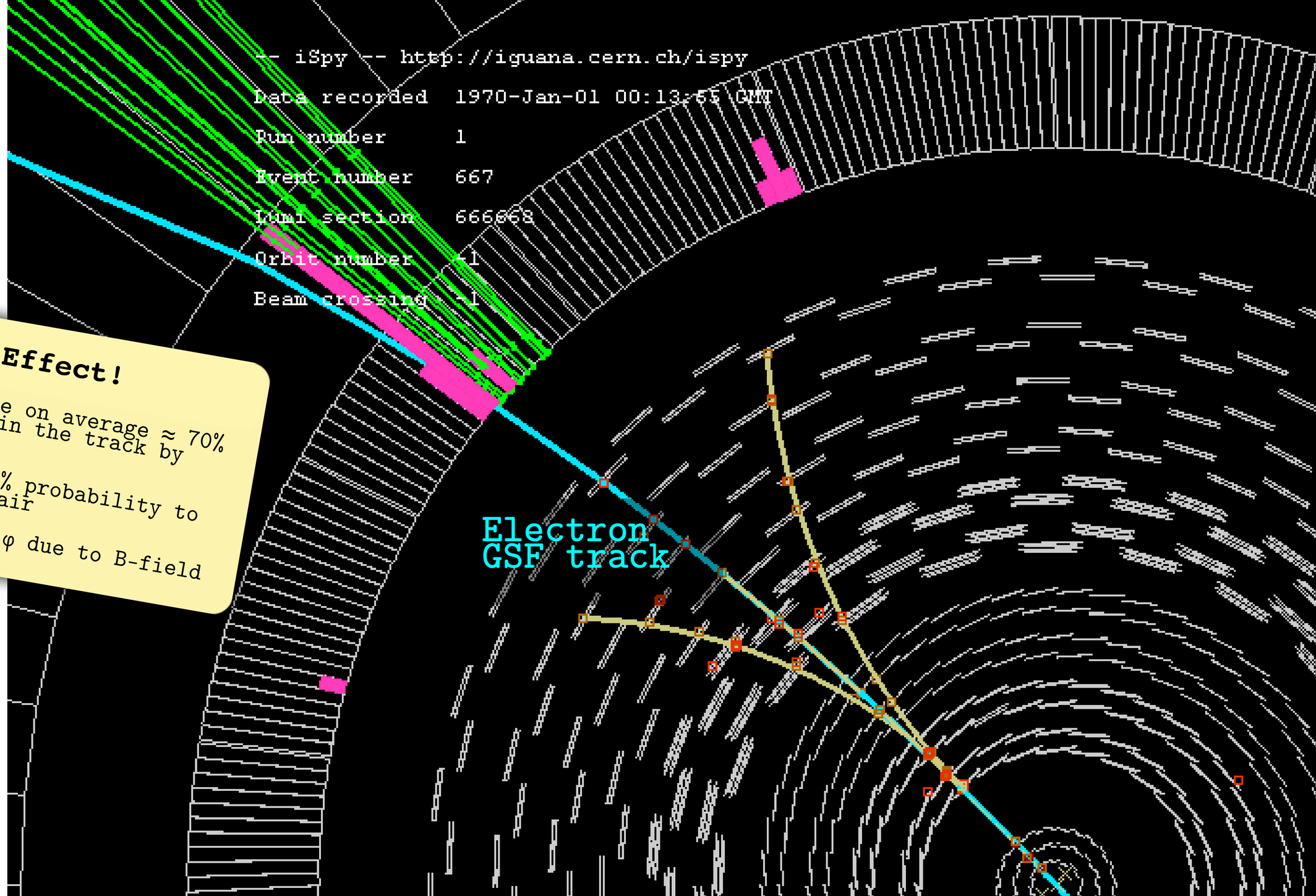
Electron ID



Huge Effect!

- Electrons radiate on average $\approx 70\%$ of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field

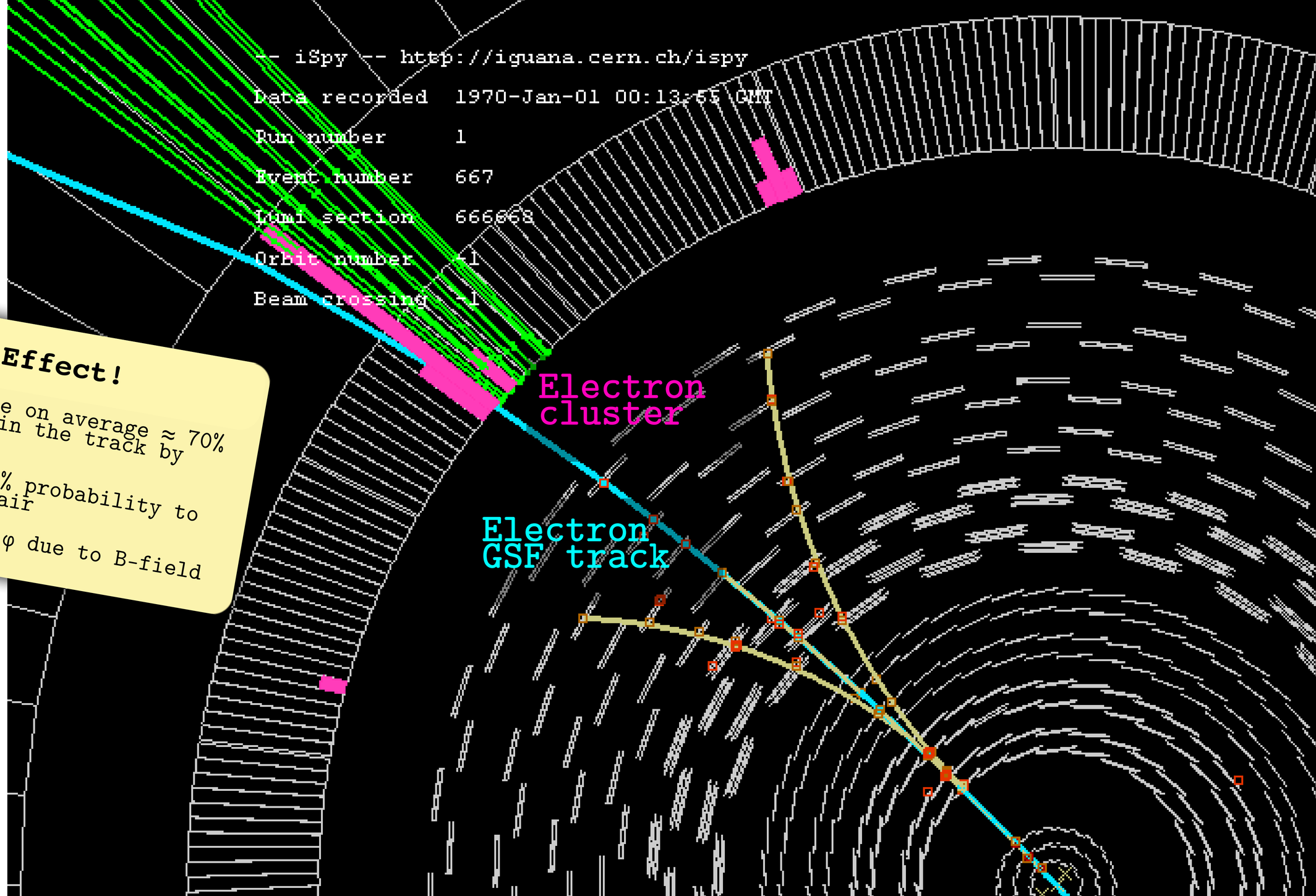
Electron ID



Huge Effect!

- Electrons radiate on average $\approx 70\%$ of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field

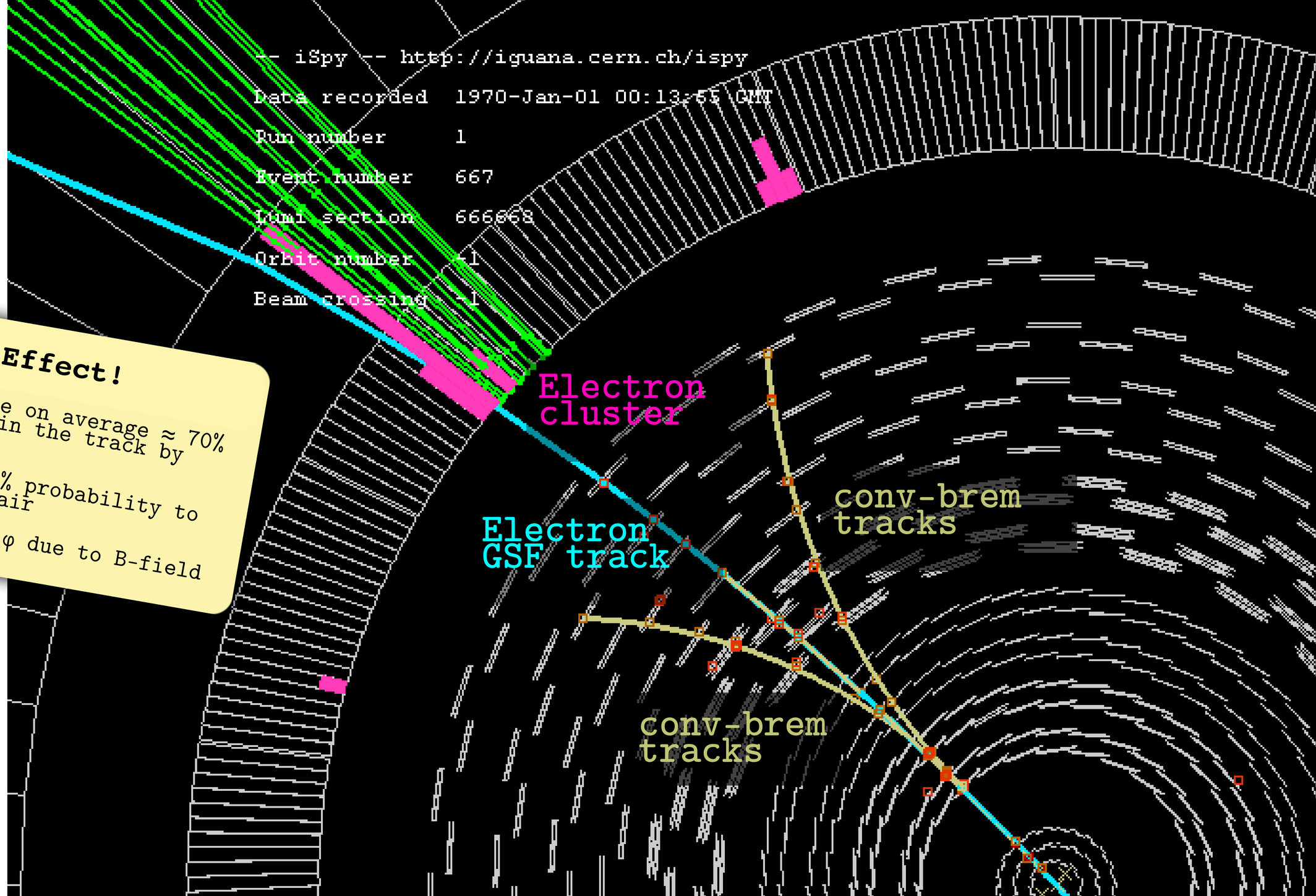
Electron ID



Huge Effect!

- Electrons radiate on average $\approx 70\%$ of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field

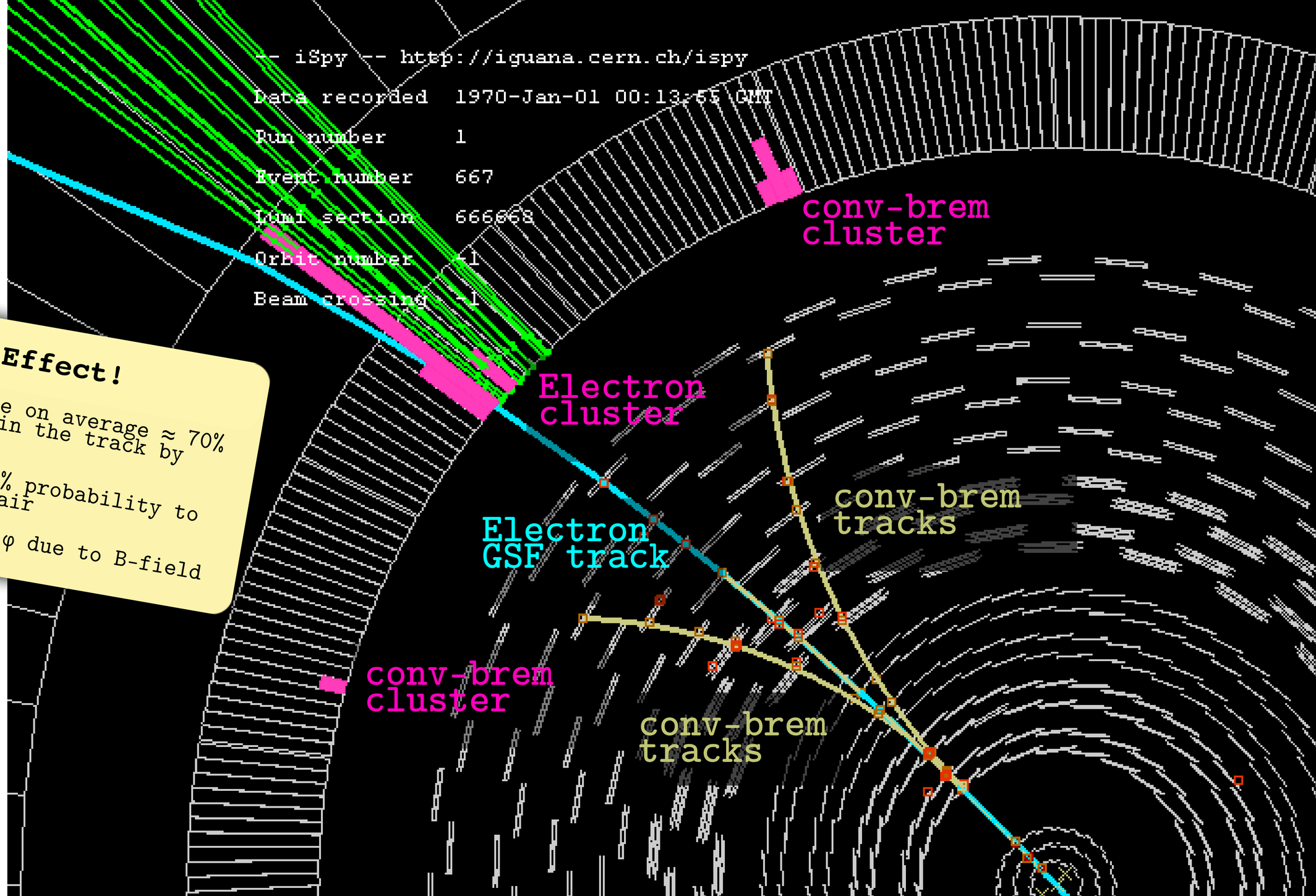
Electron ID



Huge Effect!

- Electrons radiate on average ≈ 70% of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field

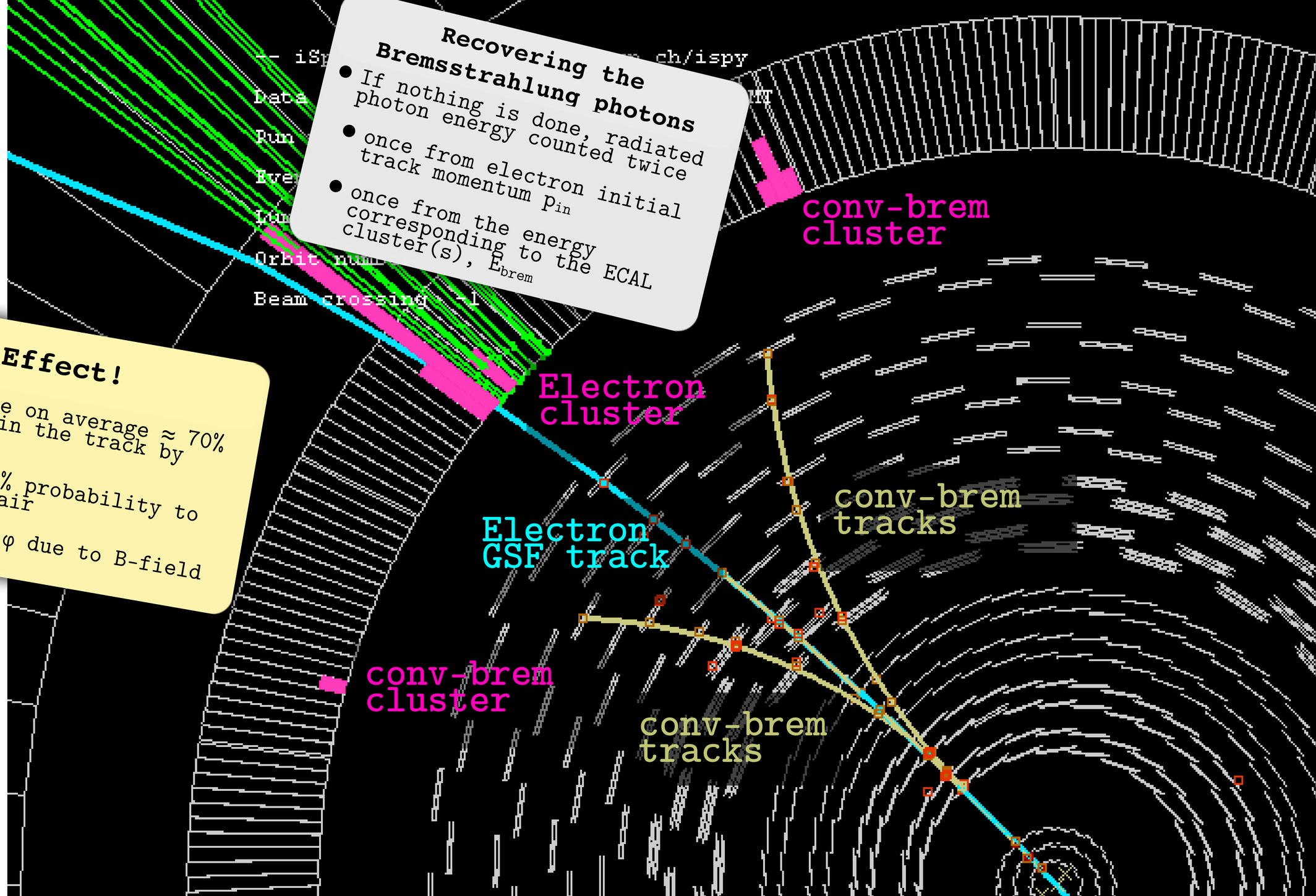
Electron ID



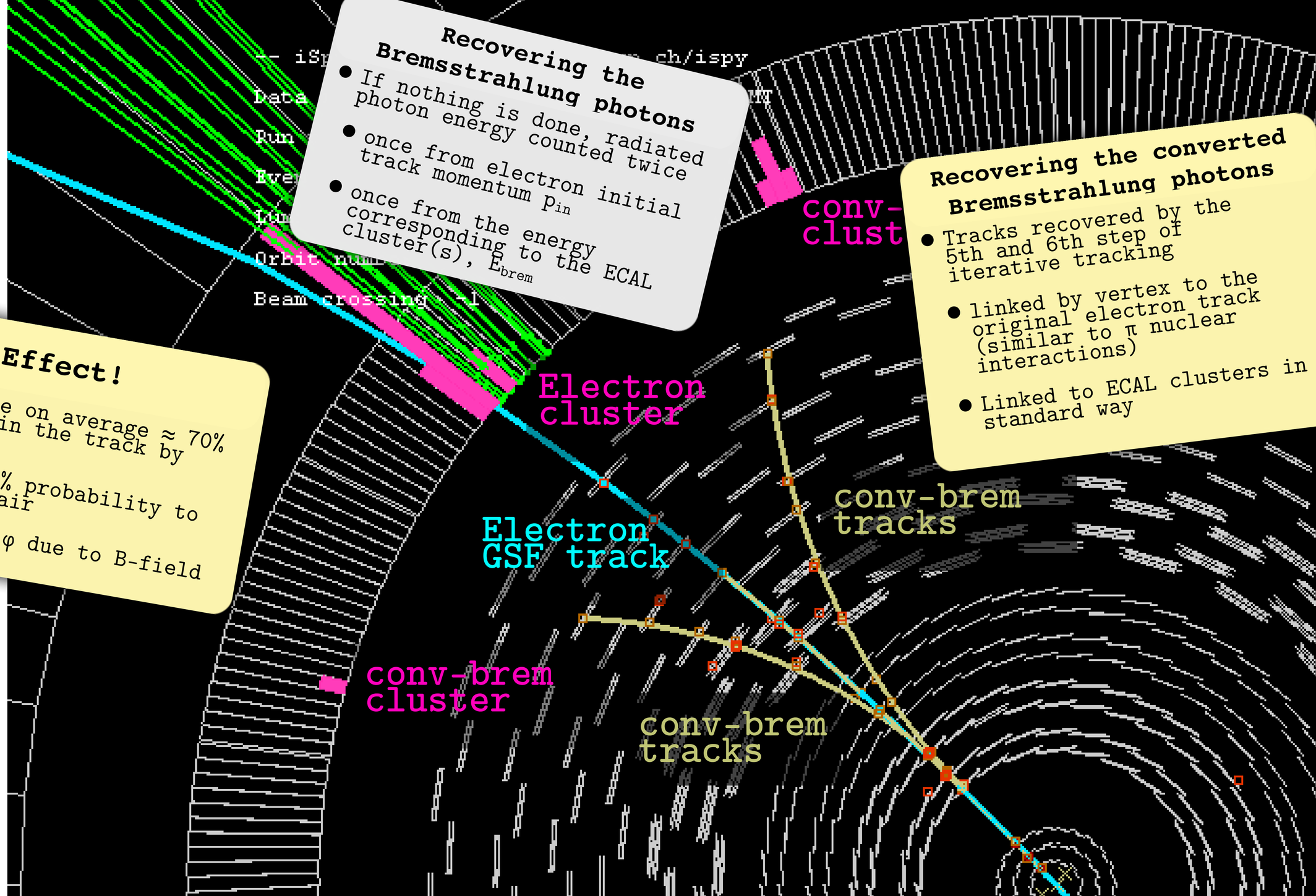
Huge Effect!

- Electrons radiate on average ≈ 70% of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field

Electron ID


Recovering the Bremsstrahlung photons
If nothing is done, radiated photon energy counted twice
once from electron initial track momentum p_{in}
once from the energy corresponding to the ECAL cluster(s), E_{brem}
conv-brem cluster
Electron cluster
Electron GSF track
conv-brem tracks
conv-brem cluster
conv-brem tracks
Huge Effect!
Electrons radiate on average ≈ 70% of their energy in the track by bremsstrahlung
photons have > 50% probability to convert to e+e- pair
energy spreads in φ due to B-field

Electron ID



Recovering the Bremsstrahlung photons

- If nothing is done, radiated photon energy counted twice
- once from electron initial track momentum p_{in}
- once from the energy corresponding to the ECAL cluster(s), E_{brem}

Recovering the converted Bremsstrahlung photons

- Tracks recovered by the 5th and 6th step of iterative tracking
- linked by vertex to the original electron track (similar to π nuclear interactions)
- Linked to ECAL clusters in standard way

Huge Effect!

- Electrons radiate on average ≈ 70% of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field

Question from Yesterday!

How to distinguish/calibrate between e's & π's ?

Huge Effect!

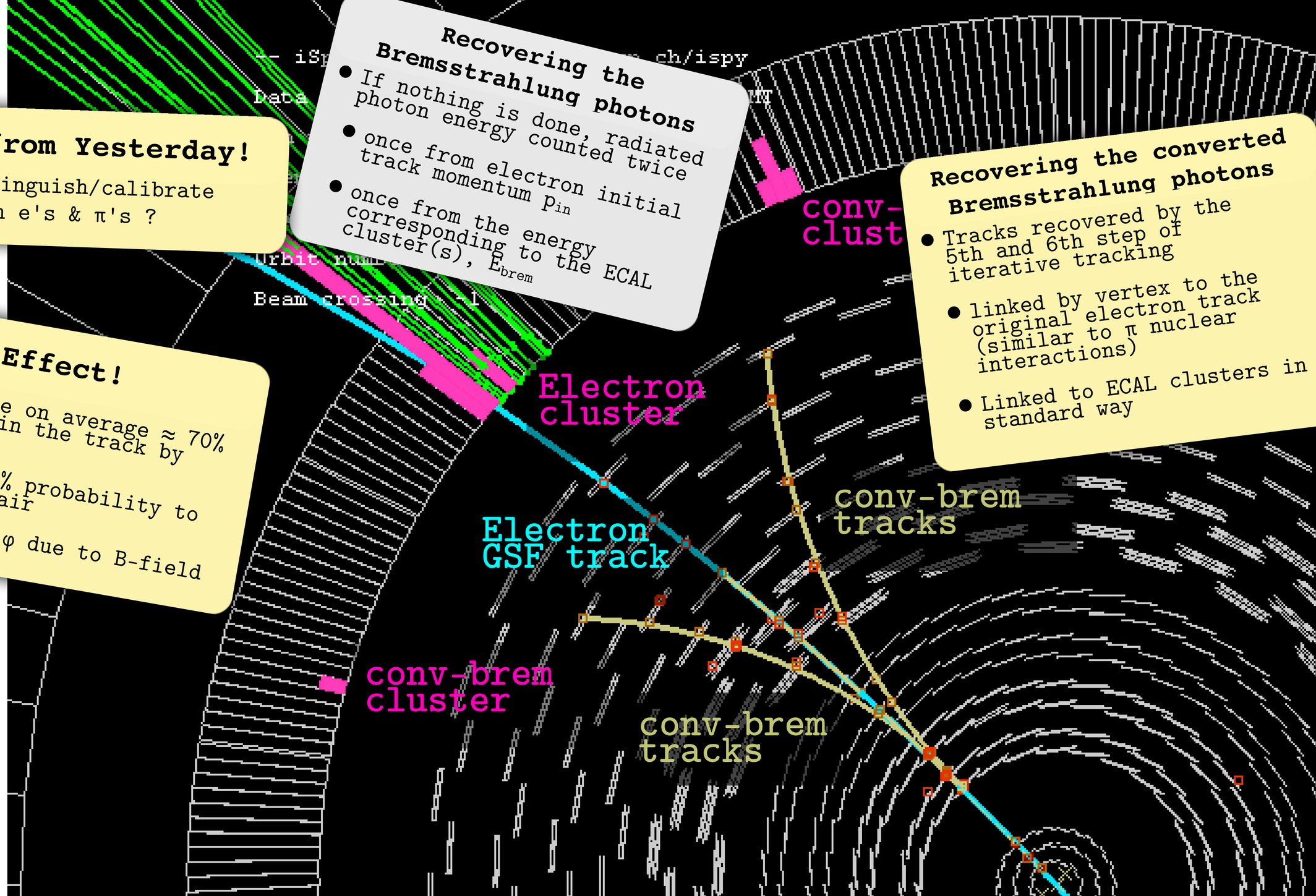
- Electrons radiate on average ≈ 70% of their energy in the track by bremsstrahlung
- photons have > 50% probability to convert to e+e- pair
- energy spreads in φ due to B-field

Recovering the Bremsstrahlung photons

- If nothing is done, radiated photon energy counted twice
- once from electron initial track momentum p_{in}
- once from the energy corresponding to the ECAL cluster(s), E_{brem}

Recovering the converted Bremsstrahlung photons

- Tracks recovered by the 5th and 6th step of iterative tracking
- linked by vertex to the original electron track (similar to π nuclear interactions)
- Linked to ECAL clusters in standard way



Electron ID

Question from Yesterday!

How to distinguish/calibrate between e's & π's ?

e's radiate; π's don't!

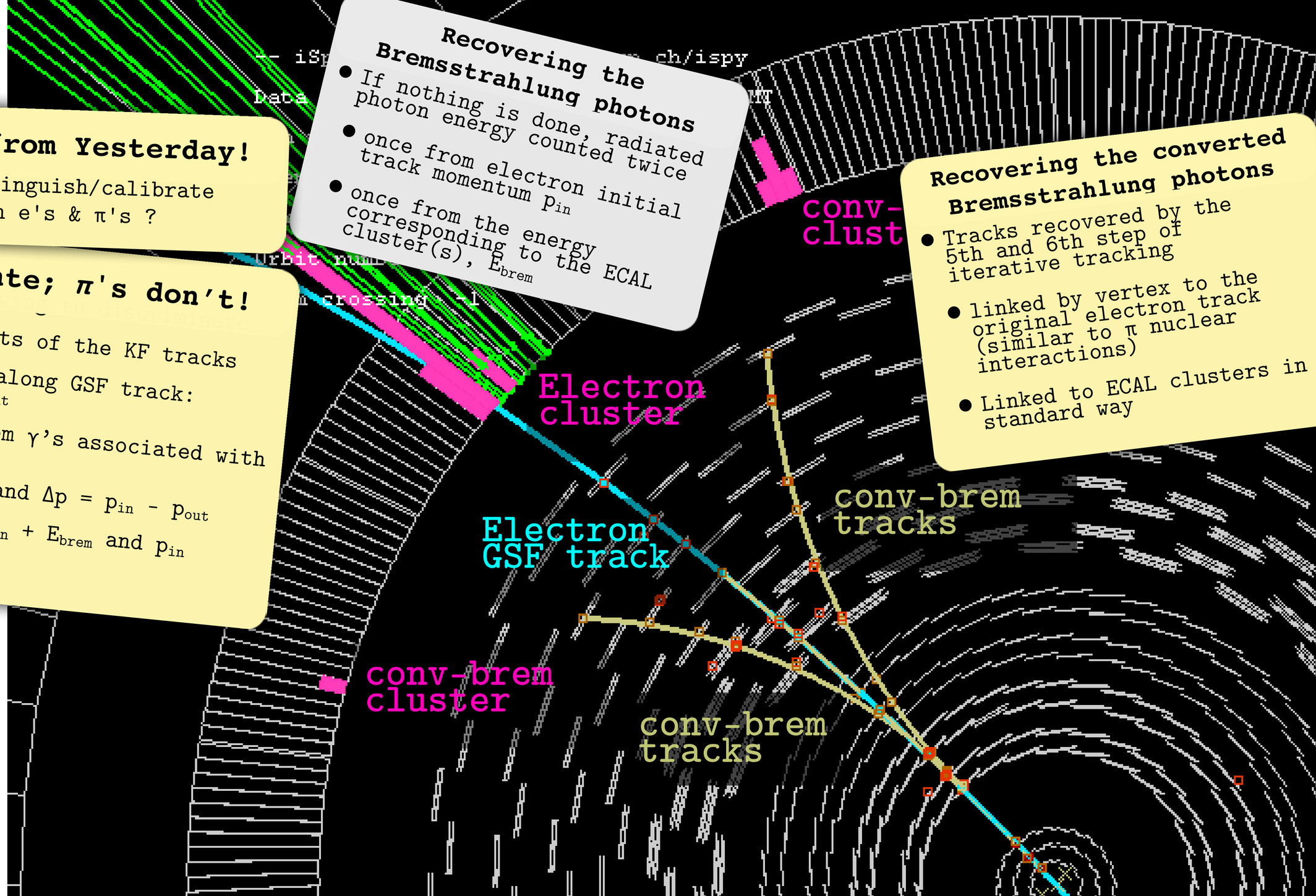
- Number of hits of the KF tracks
- Energy loss along GSF track: $\Delta p = p_{in} - p_{out}$
- Number of Brem γ's associated with track
- Compare E_{brem} and $\Delta p = p_{in} - p_{out}$
- Compare $E_{electron} + E_{brem}$ and p_{in}
- etc

Recovering the Bremsstrahlung photons

- If nothing is done, radiated photon energy counted twice
- once from electron initial track momentum p_{in}
- once from the energy corresponding to the ECAL cluster(s), E_{brem}

Recovering the converted Bremsstrahlung photons

- Tracks recovered by the 5th and 6th step of iterative tracking
- linked by vertex to the original electron track (similar to π nuclear interactions)
- Linked to ECAL clusters in standard way



Electron ID

Question from Yesterday!

How to distinguish/calibrate between e's & π's ?

e's radiate; π's don't!

- Number of hits of the KF tracks
- Energy loss along GSF track: $\Delta p = p_{in} - p_{out}$
- Number of Brem γ's associated with track
- Compare E_{brem} and $\Delta p = p_{in} - p_{out}$
- Compare $E_{electron} + E_{brem}$ and p_{in}
- etc

Use cluster information!

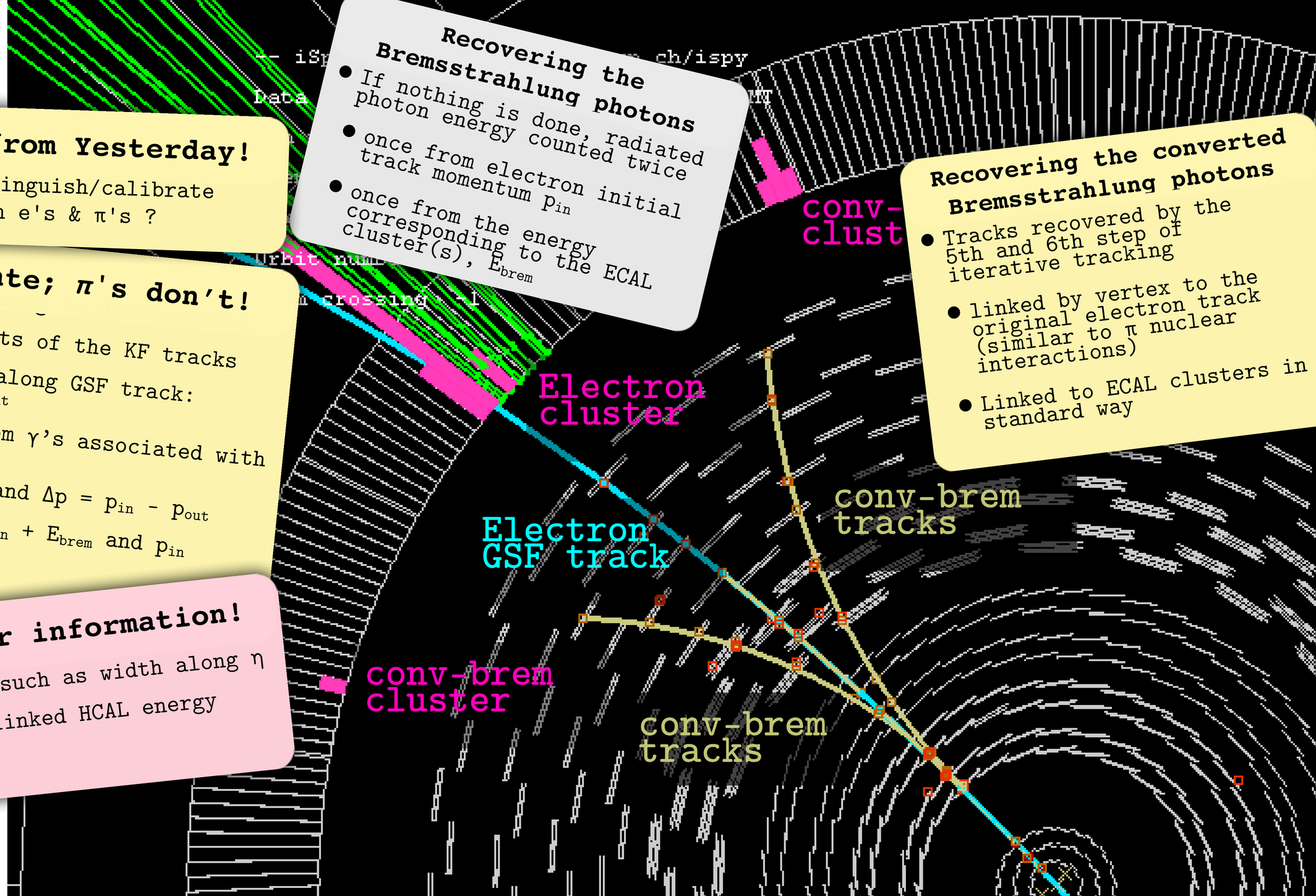
- Shower shape, such as width along η
- any possible linked HCAL energy
- etc

Recovering the Bremsstrahlung photons

- If nothing is done, radiated photon energy counted twice
- once from electron initial track momentum p_{in}
- once from the energy corresponding to the ECAL cluster(s), E_{brem}

Recovering the converted Bremsstrahlung photons

- Tracks recovered by the 5th and 6th step of iterative tracking
- linked by vertex to the original electron track (similar to π nuclear interactions)
- Linked to ECAL clusters in standard way



Electron ID

Question from Yesterday!

How to distinguish/calibrate between e's & π's ?

e's radiate; π's don't!

- Number of hits of the KF tracks
- Energy loss along GSF track: $\Delta p = p_{in} - p_{out}$
- Number of Brem γ's associated with track
- Compare E_{brem} and $\Delta p = p_{in} - p_{out}$
- Compare $E_{electron} + E_{brem}$ and p_{in}
- etc

Use cluster information!

- Shower shape, such as width along η
- any possible linked HCAL energy
- etc

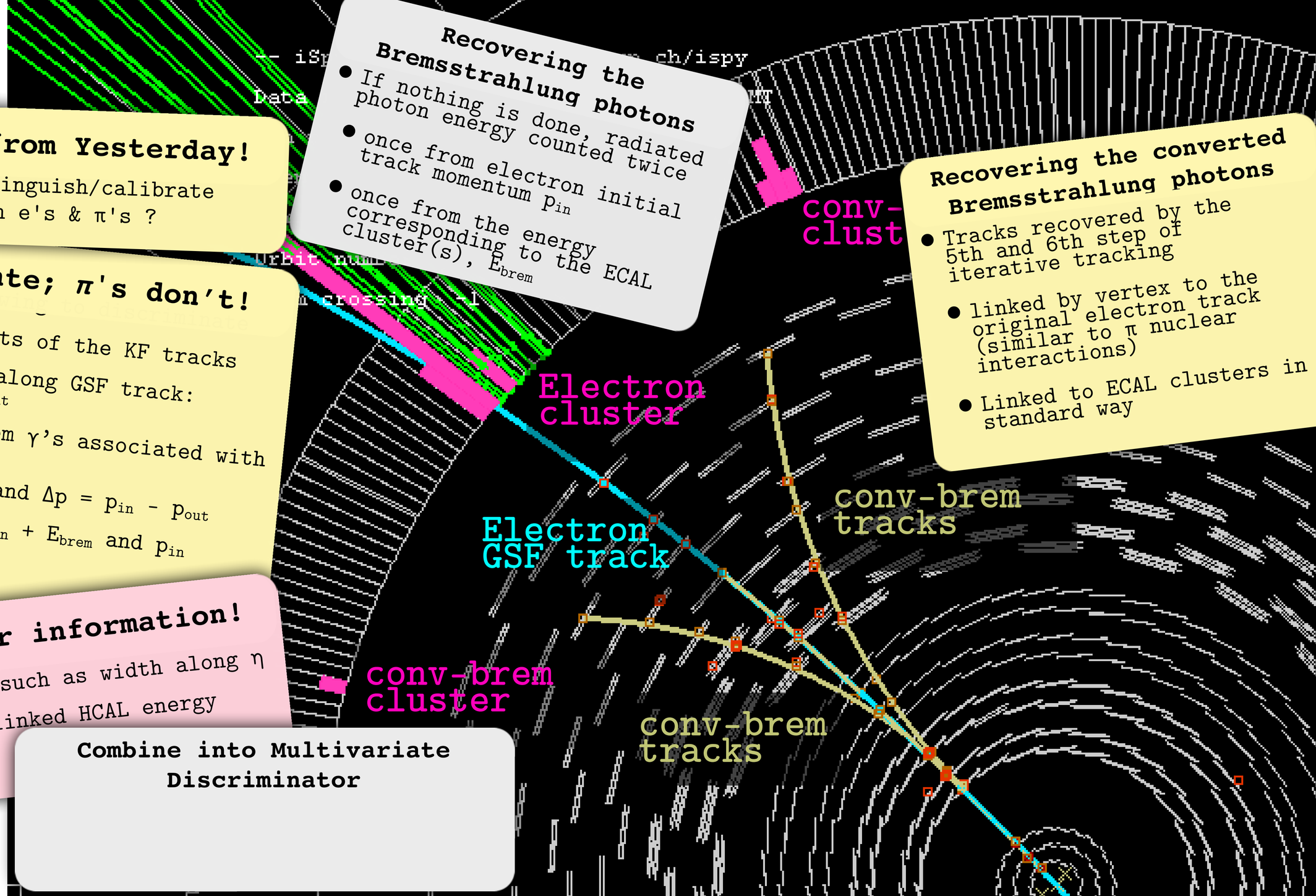
Combine into Multivariate Discriminator

Recovering the Bremsstrahlung photons

- If nothing is done, radiated photon energy counted twice
- once from electron initial track momentum p_{in}
- once from the energy corresponding to the ECAL cluster(s), E_{brem}

Recovering the converted Bremsstrahlung photons

- Tracks recovered by the 5th and 6th step of iterative tracking
- linked by vertex to the original electron track (similar to π nuclear interactions)
- Linked to ECAL clusters in standard way



Electron ID

Question from Yesterday!

How to distinguish/calibrate between e's & π's ?

e's radiate; π's don't!

- Number of hits of the KF tracks
- Energy loss along GSF track: $\Delta p = p_{in} - p_{out}$
- Number of Brem γ's associated with track
- Compare E_{brem} and $\Delta p = p_{in} - p_{out}$
- Compare $E_{electron} + E_{brem}$ and p_{in}
- etc

Use cluster information!

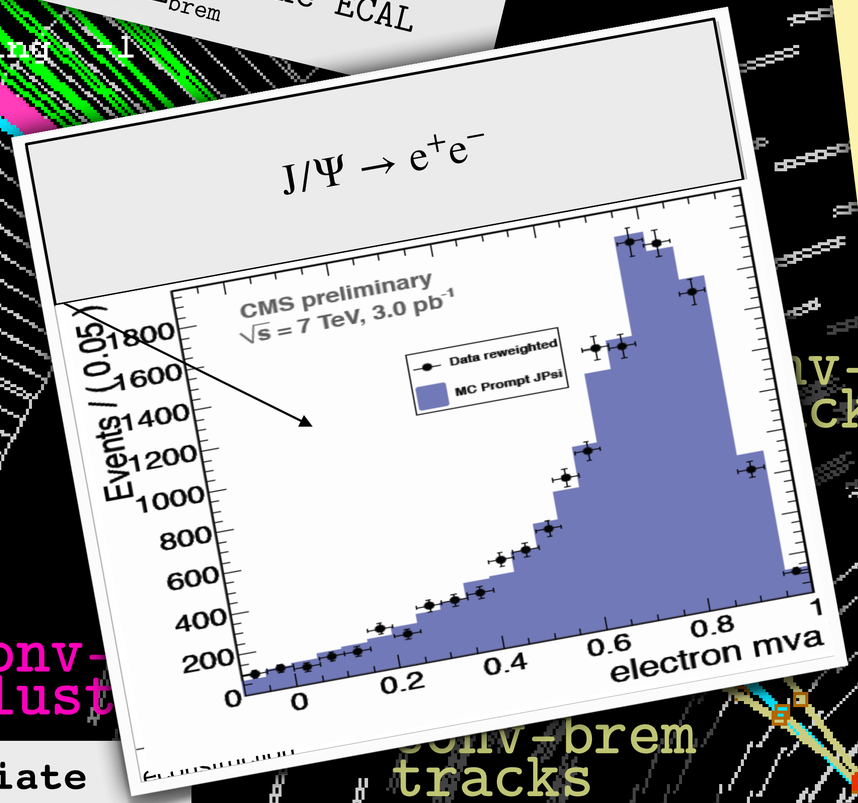
- Shower shape, such as width along η
- any possible linked HCAL energy
- etc

Recovering the Bremsstrahlung photons

- If nothing is done, radiated photon energy counted twice
  - once from electron initial track momentum p_{in}
  - once from the energy corresponding to the ECAL cluster(s), E_{brem}

Recovering the converted Bremsstrahlung photons

- Tracks recovered by the 5th and 6th step of iterative tracking
  - linked by vertex to the original electron track (similar to π nuclear interactions)
  - Linked to ECAL clusters in standard way



Combine into Multivariate Discriminator

Question from Yesterday!

How to distinguish/calibrate between e's & π's ?

Recovering the Bremsstrahlung photons

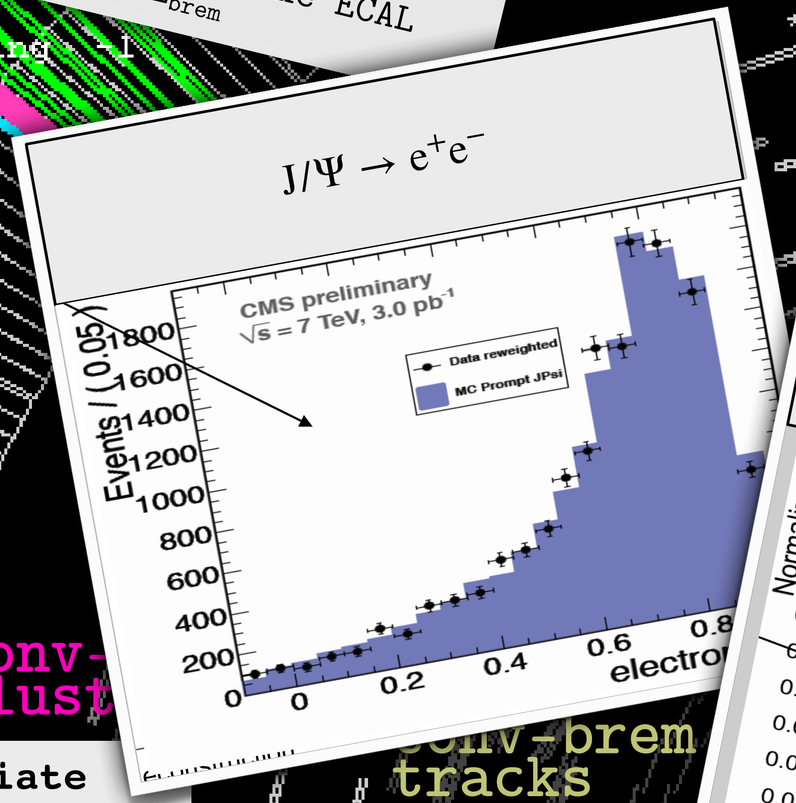
- If nothing is done, radiated photon energy counted twice
  - once from electron initial track momentum p_{in}
  - once from the energy corresponding to the ECAL cluster(s), E_{brem}

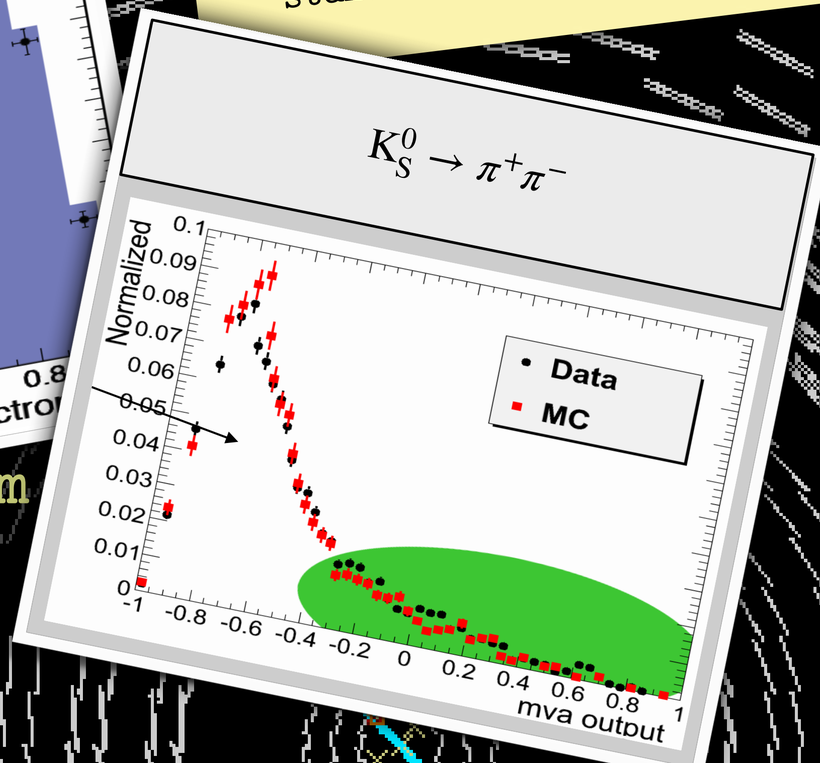
Recovering the converted Bremsstrahlung photons

- Tracks recovered by the 5th and 6th step of iterative tracking
  - linked by vertex to the original electron track (similar to π nuclear interactions)
  - Linked to ECAL clusters in standard way

e's radiate; π's don't!

- Number of hits of the KF tracks
- Energy loss along GSF track: $\Delta p = p_{in} - p_{out}$
- Number of Brem γ's associated with track
- Compare E_{brem} and $\Delta p = p_{in} - p_{out}$
- Compare $E_{electron} + E_{brem}$ and p_{in}
- etc





Use cluster information!

- Shower shape, such as width along η
- any possible linked HCAL energy
- etc

Combine into Multivariate Discriminator

Electron ID

Question from Yesterday!

How to distinguish/calibrate between e's & π's ?

Recovering the Bremsstrahlung photons

- If nothing is done, radiated photon energy counted twice
- once from electron initial track momentum p_{in}
- once from the energy corresponding to the ECAL cluster(s), E_{brem}

Recovering the converted Bremsstrahlung photons

- Tracks recovered by the 5th and 6th step of iterative tracking
- linked by vertex to the original electron track (similar to π nuclear interactions)
- Linked to ECAL clusters in standard way

e's radiate; π's don't!

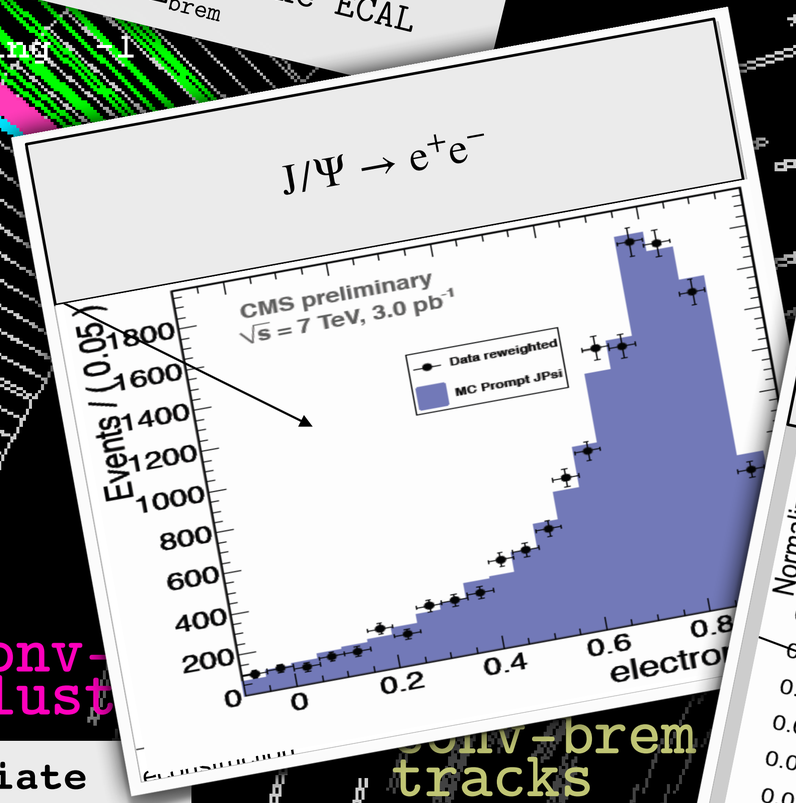
- Number of hits of the KF tracks
- Energy loss along GSF track: $\Delta p = p_{in} - p_{out}$
- Number of Brem γ's associated with track
- Compare E_{brem} and $\Delta p = p_{in} - p_{out}$
- Compare $E_{electron} + E_{brem}$ and p_{in}
- etc

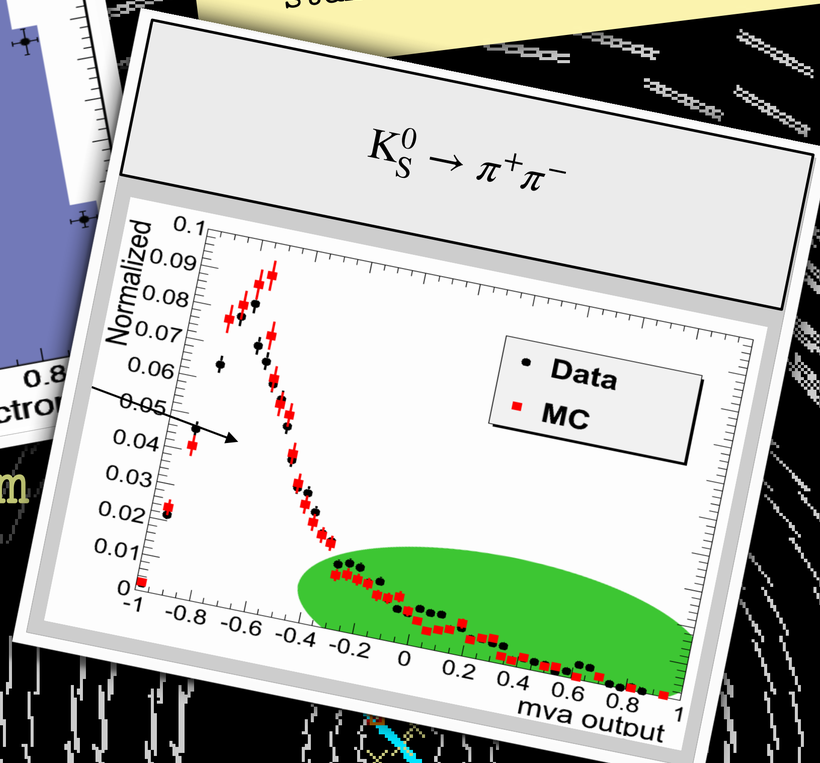
Use cluster information!

- Shower shape, such as width along η
- any possible linked HCAL energy
- etc

Combine into Multivariate Discriminator

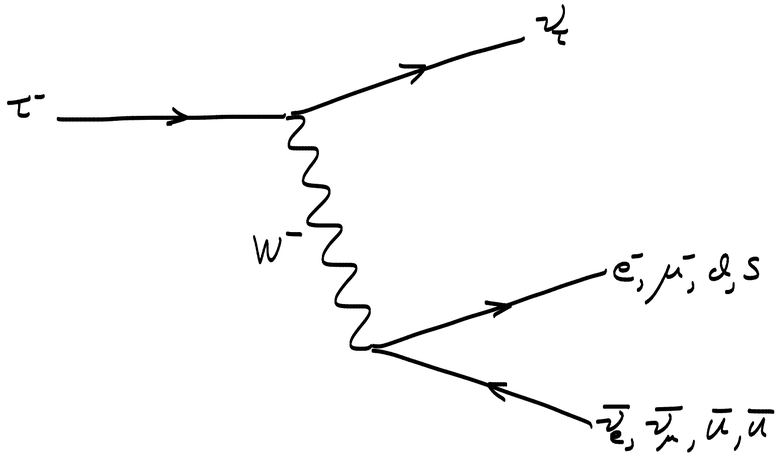
- 95% efficiency for isolated electrons
- 70-80% efficiency in jets





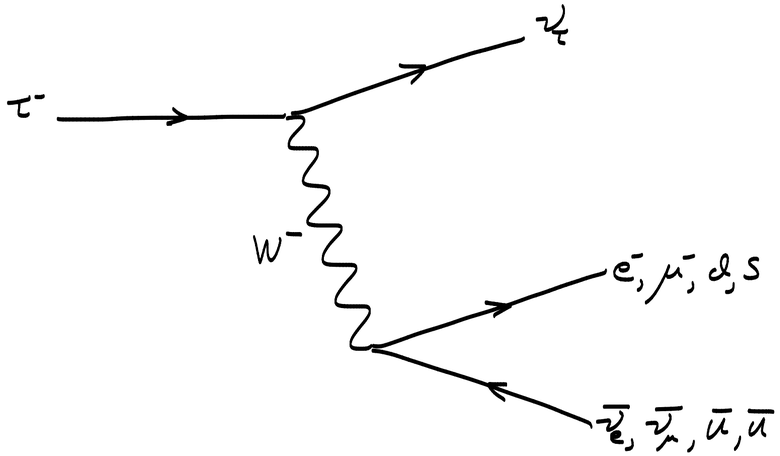
Tau ID & Reconstruction

Tau ID & Reconstruction



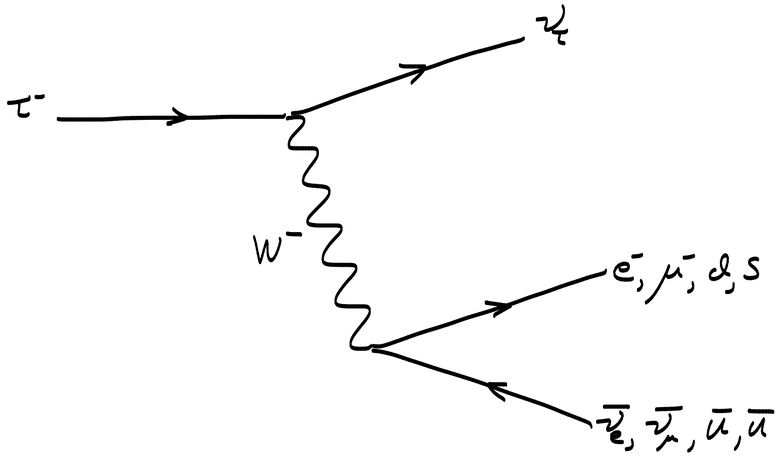
Tau ID & Reconstruction

- Massive, (relatively) long lived



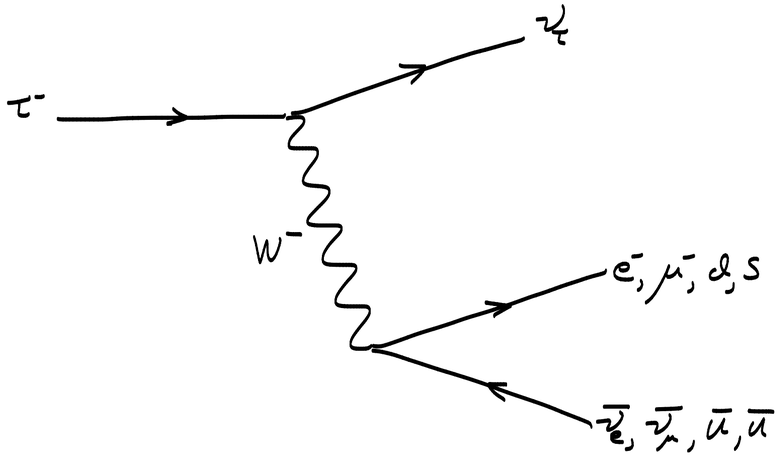
Tau ID & Reconstruction

- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV



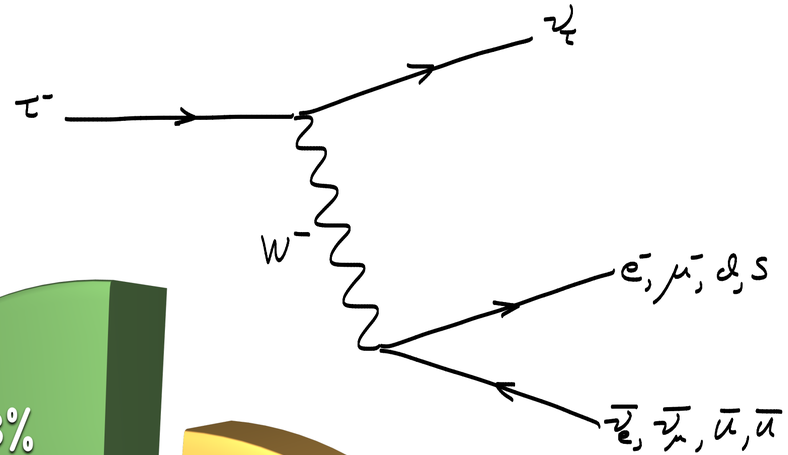
Tau ID & Reconstruction

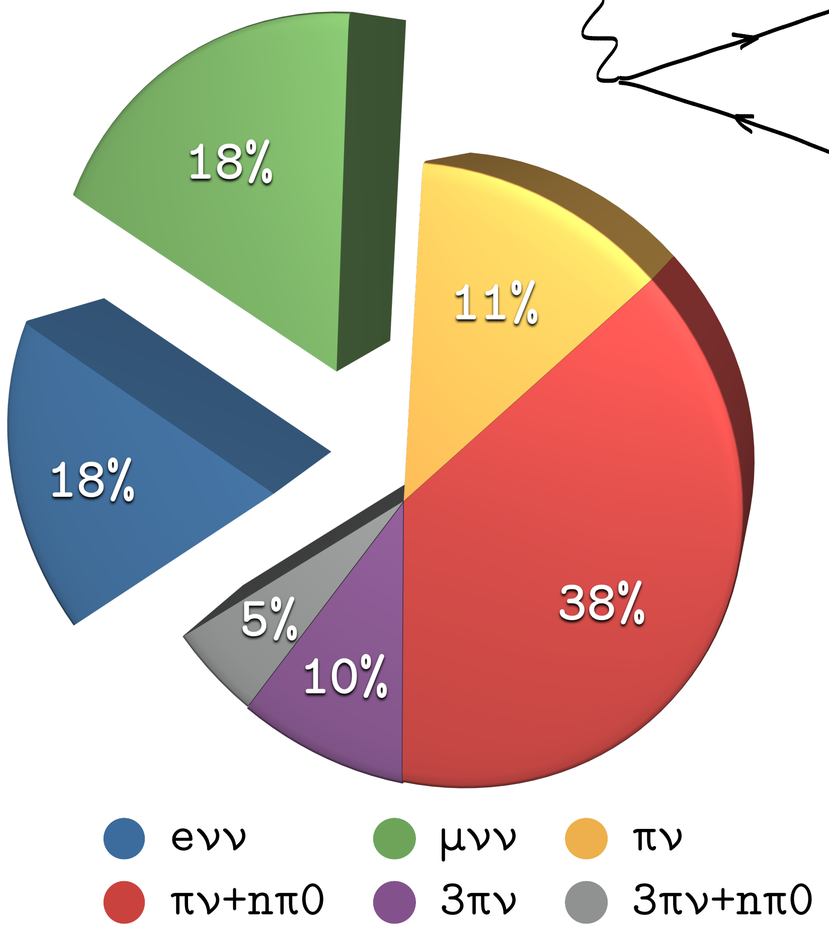
- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87$ μm



Tau ID & Reconstruction

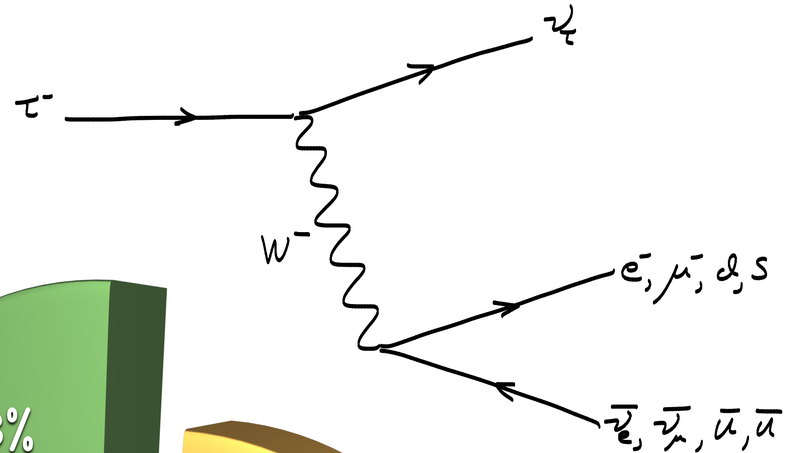
- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87$ μm
- leptonic decays: 35%

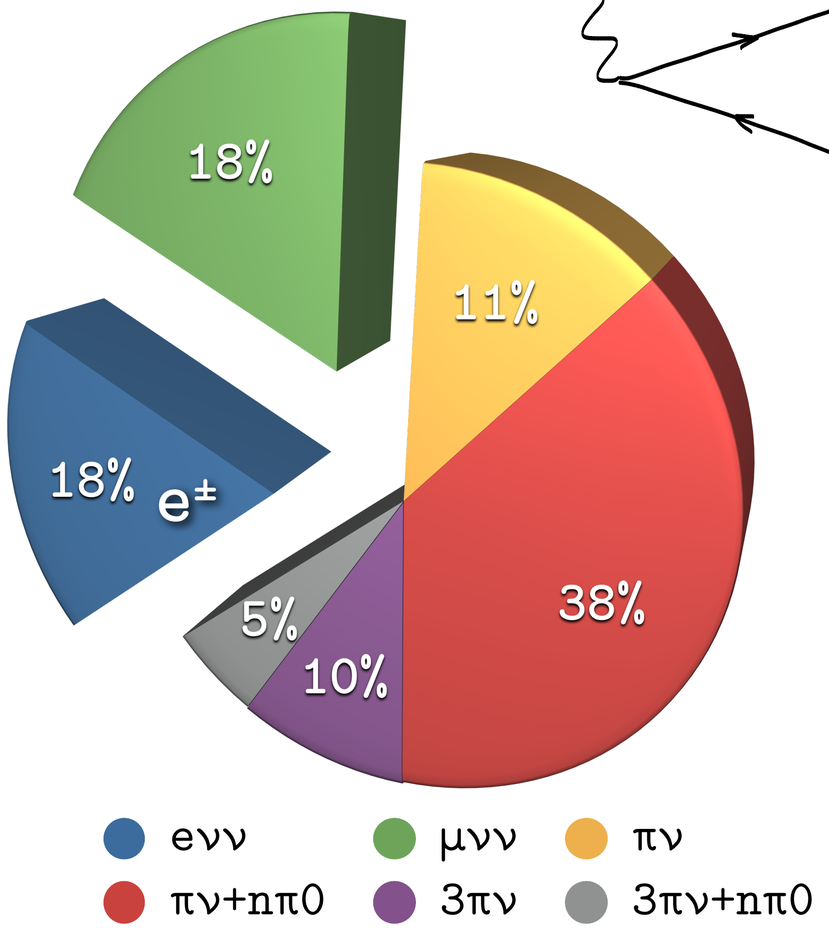




Tau ID & Reconstruction

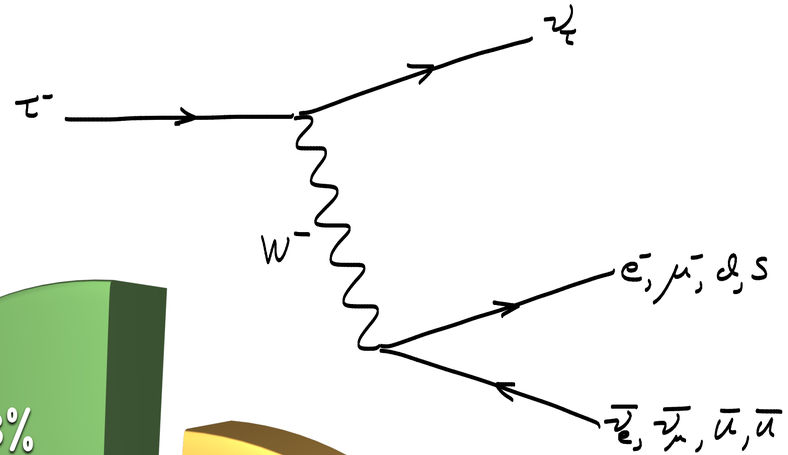
- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87$ μm
- leptonic decays: 35%
  - electron: 17.5%

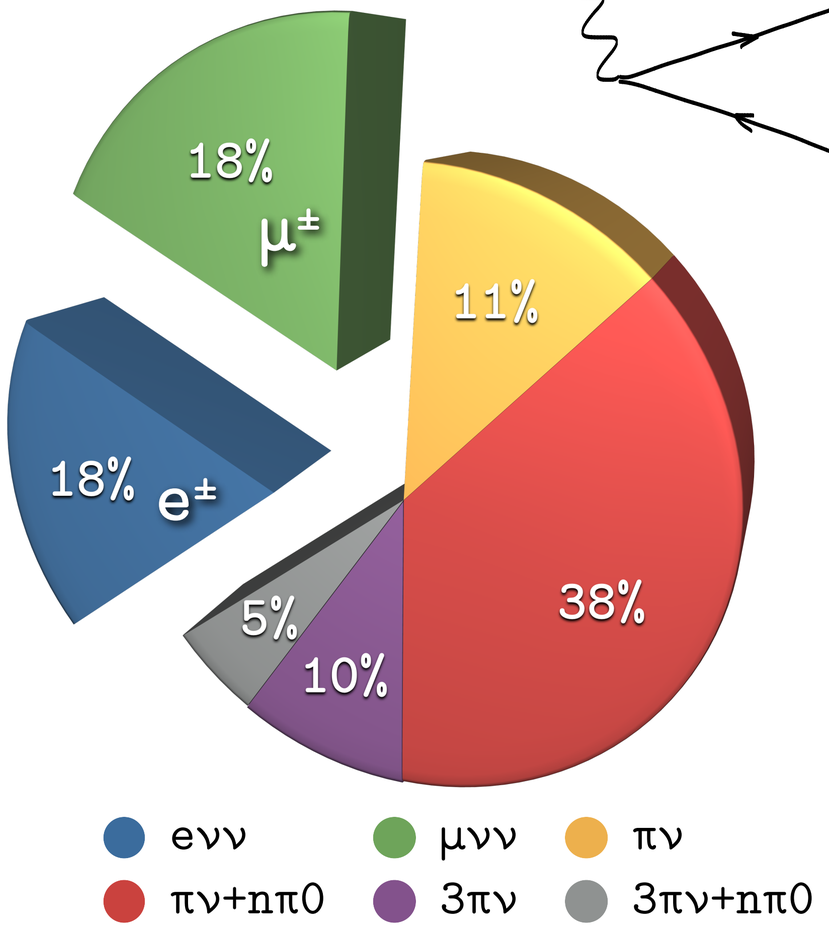




Tau ID & Reconstruction

- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87\ \mu m$
- leptonic decays: 35%
  - electron: 17.5%
  - muon: 17.5%

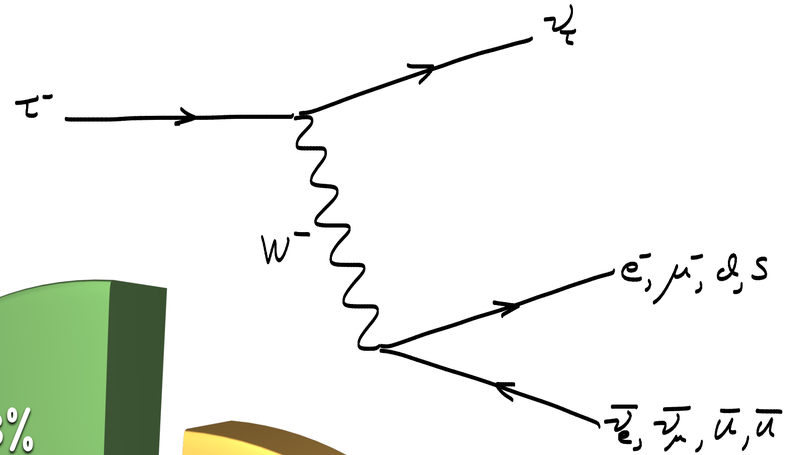


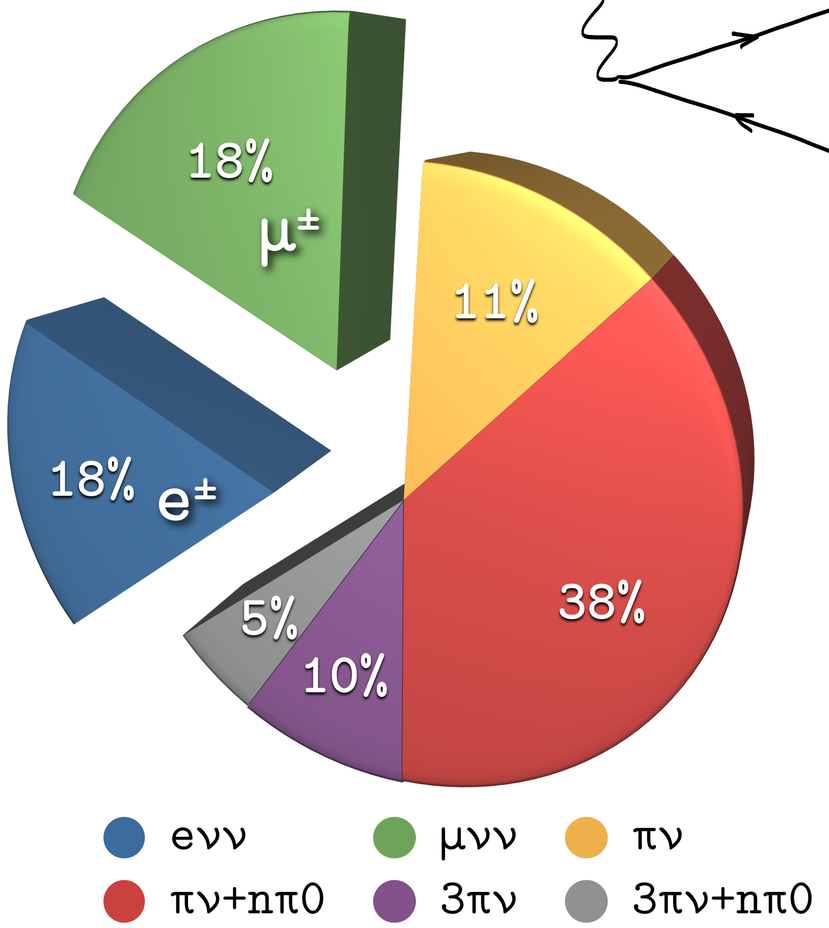


Tau ID & Reconstruction

- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87\ \mu m$
- leptonic decays: 35%
  - electron: 17.5%
  - muon: 17.5%

I won't talk about leptonic decays

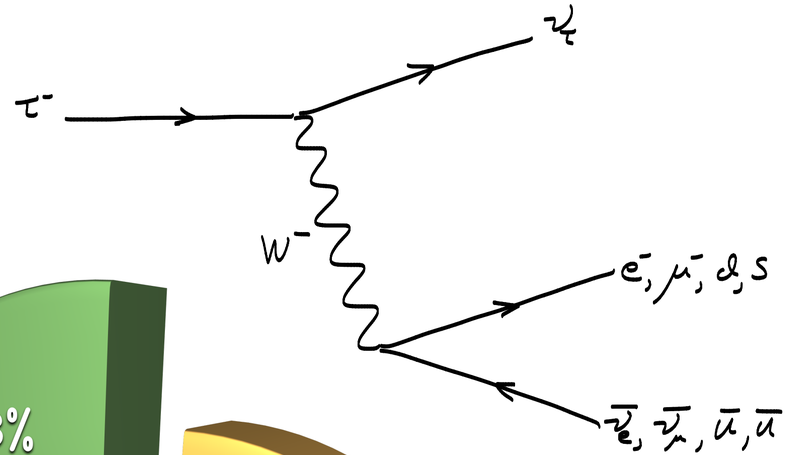


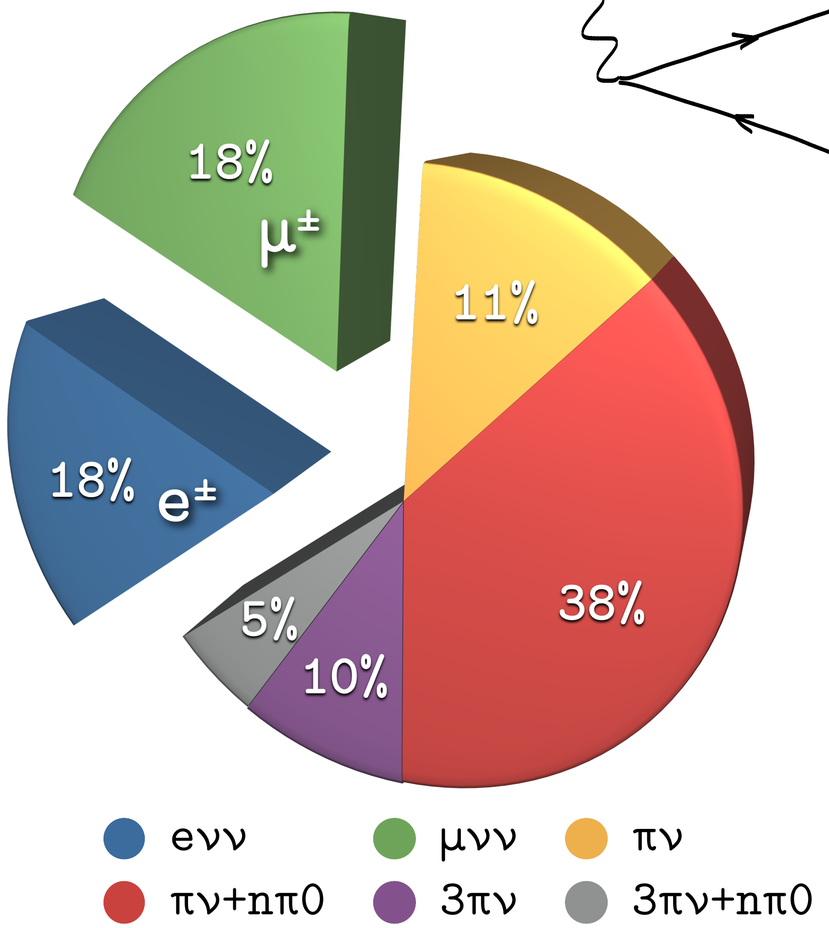


Tau ID & Reconstruction

- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87\ \mu m$
- leptonic decays: 35%
  - electron: 17.5%
  - muon: 17.5%
- hadronic decays: 65%

I won't talk about leptonic decays

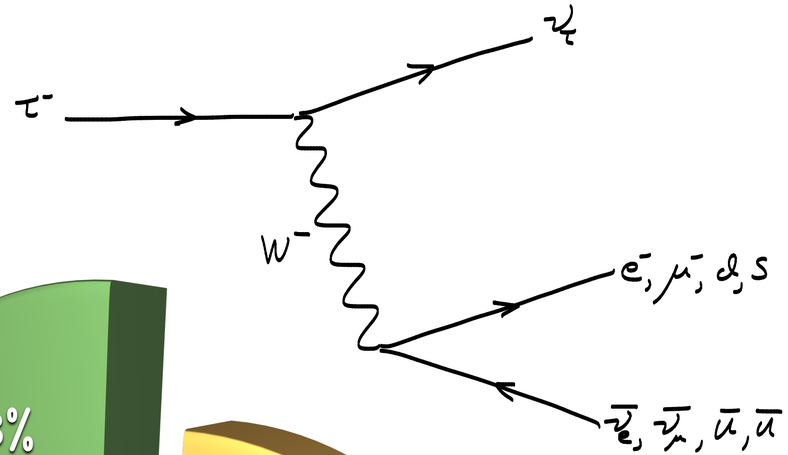


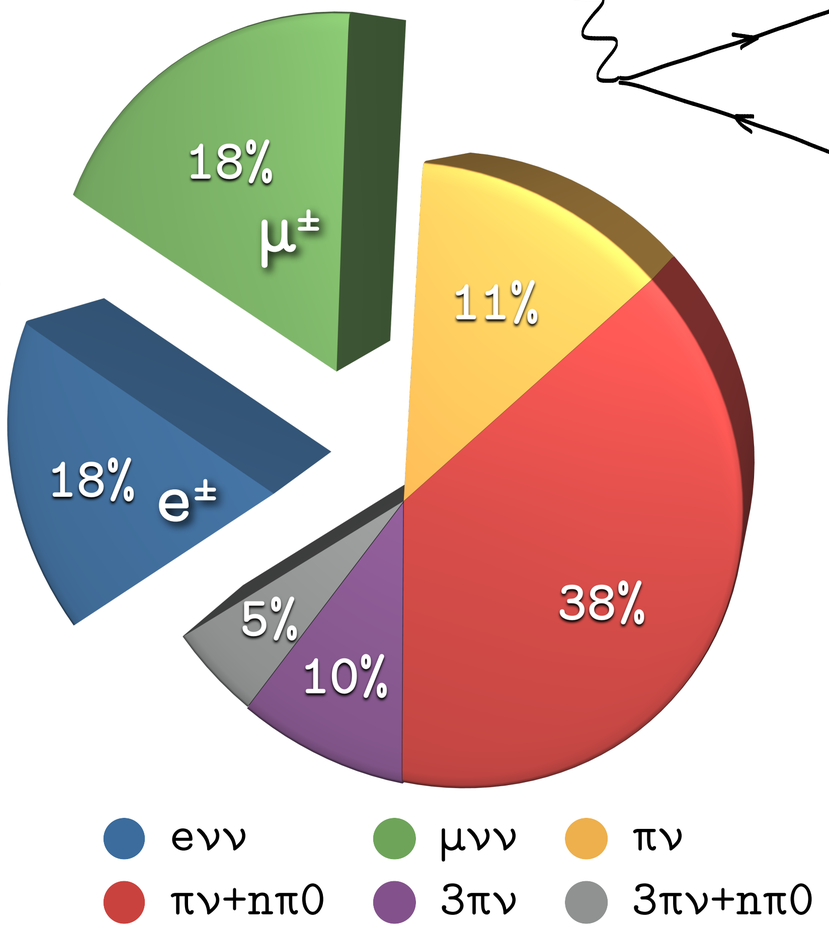


Tau ID & Reconstruction

- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87\ \mu m$
- leptonic decays: 35%
  - electron: 17.5%
  - muon: 17.5%
- hadronic decays: 65%
  - single prong: 49%

I won't talk about leptonic decays

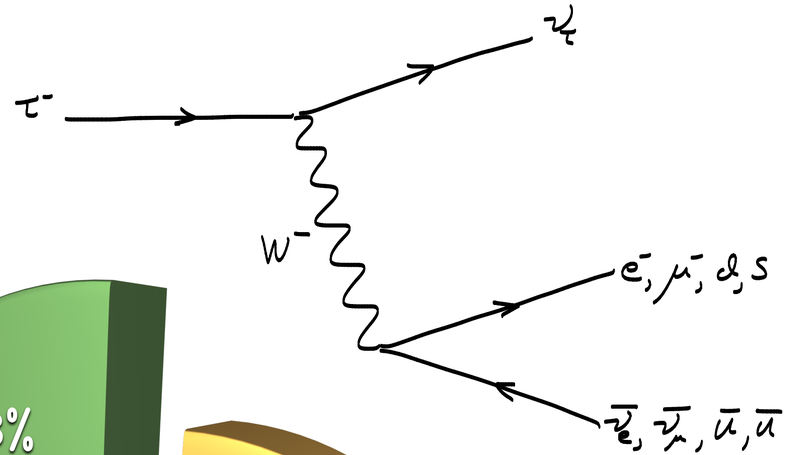


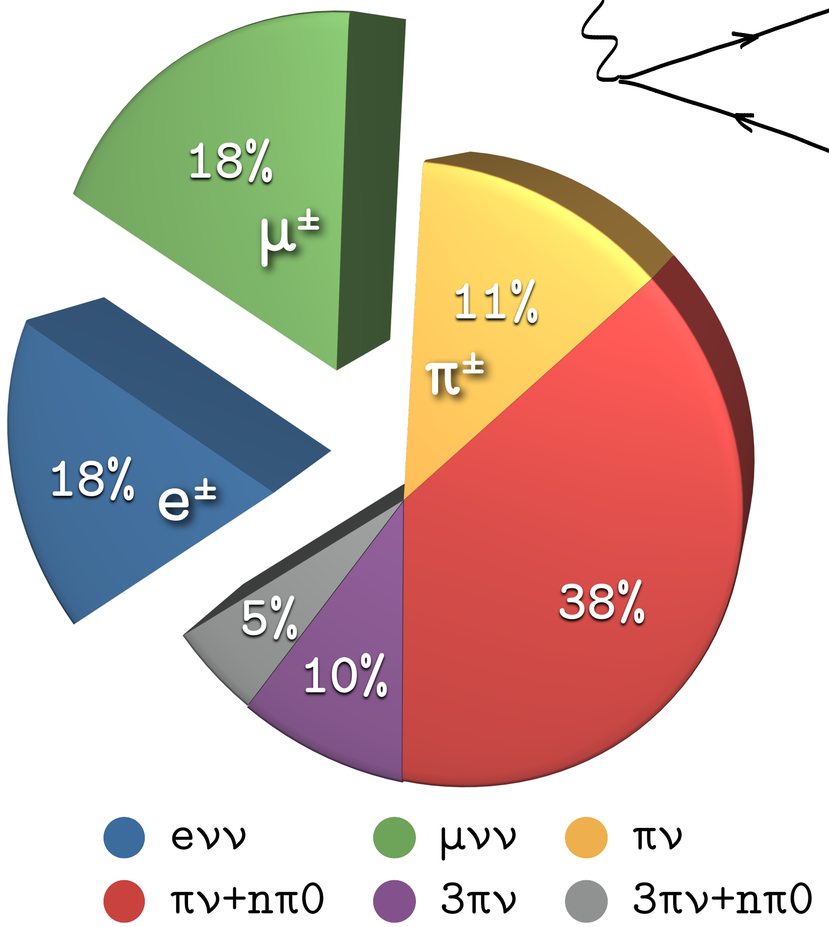


Tau ID & Reconstruction

- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87$ μm
- leptonic decays: 35%
  - electron: 17.5%
  - muon: 17.5%
- hadronic decays: 65%
  - single prong: 49%
    - 11% $\pi^{\pm}$

I won't talk about leptonic decays

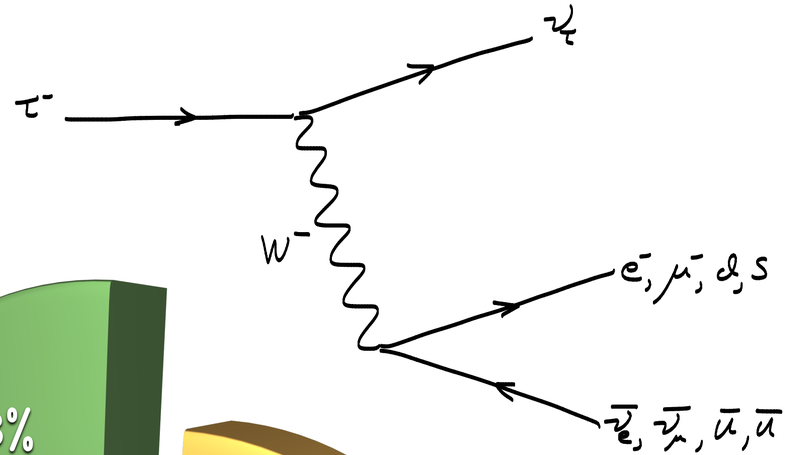


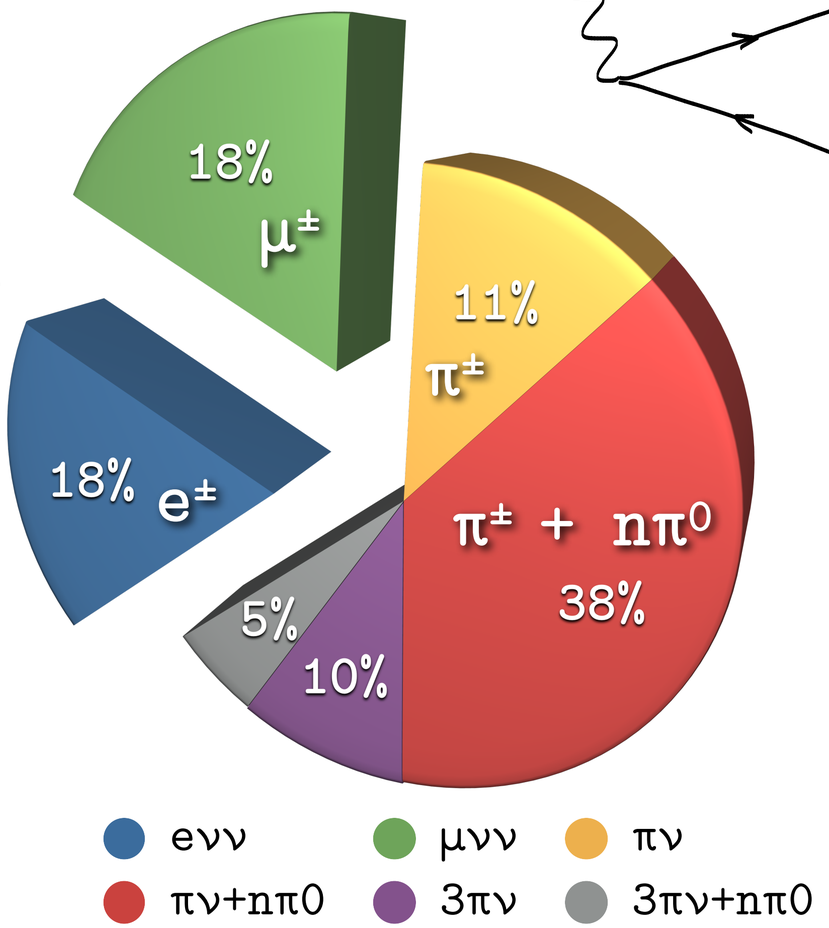


Tau ID & Reconstruction

- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87\ \mu m$
- leptonic decays: 35%
  - electron: 17.5%
  - muon: 17.5%
- hadronic decays: 65%
  - single prong: 49%
    - 11% $\pi^{\pm}$
    - 38% $\pi^{\pm} + n\pi^0$

I won't talk about leptonic decays

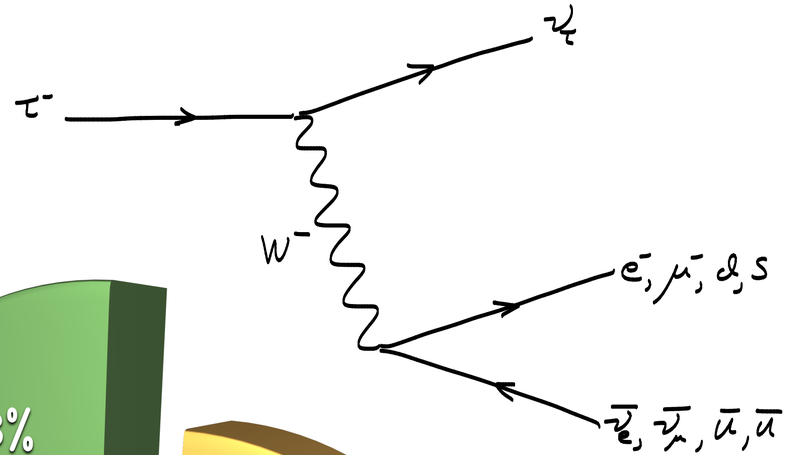


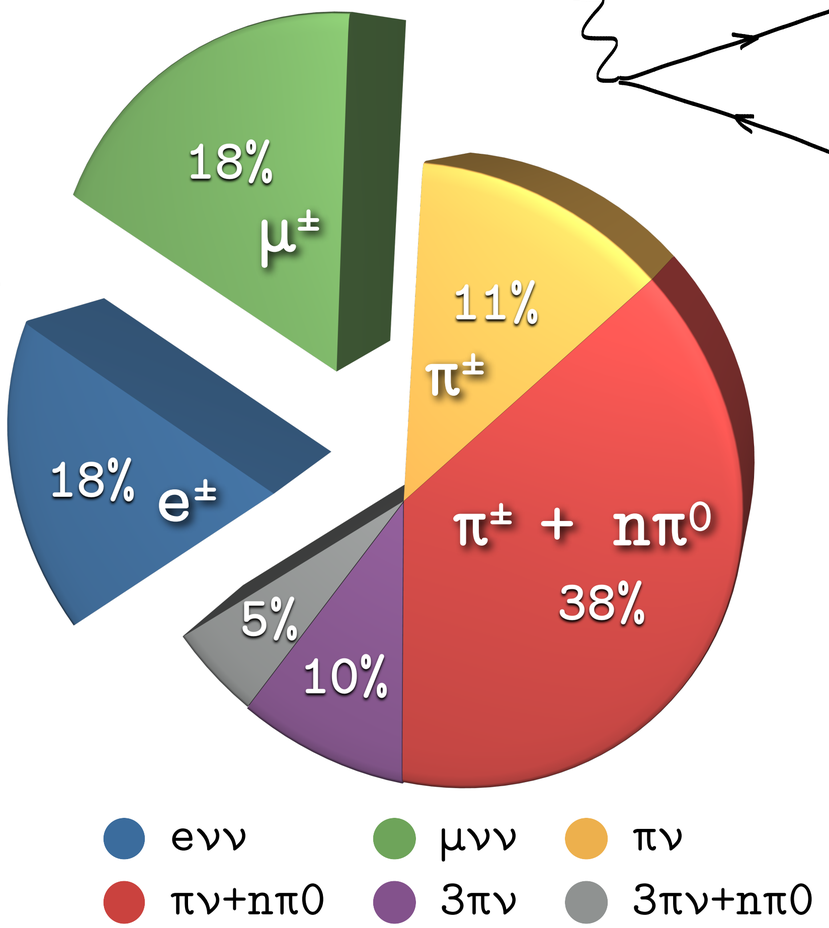


Tau ID & Reconstruction

- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87\ \mu m$
- leptonic decays: 35%
  - electron: 17.5%
  - muon: 17.5%
- hadronic decays: 65%
  - single prong: 49%
    - 11% $\pi^{\pm}$
    - 38% $\pi^{\pm} + n\pi^0$
  - three prong: 15%

I won't talk about leptonic decays

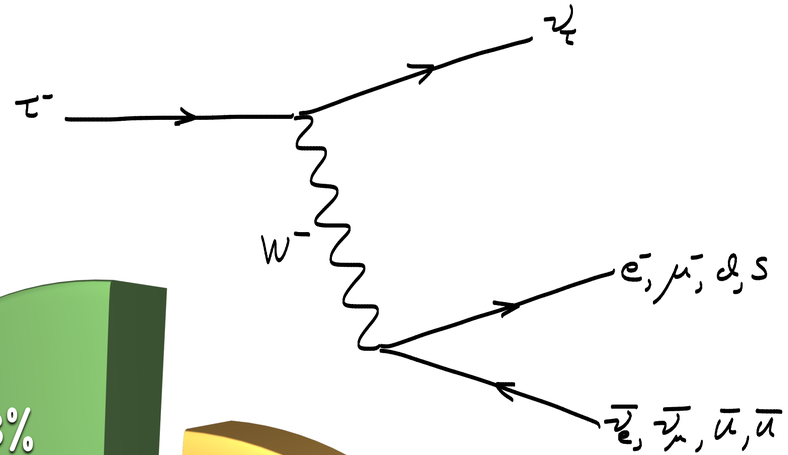


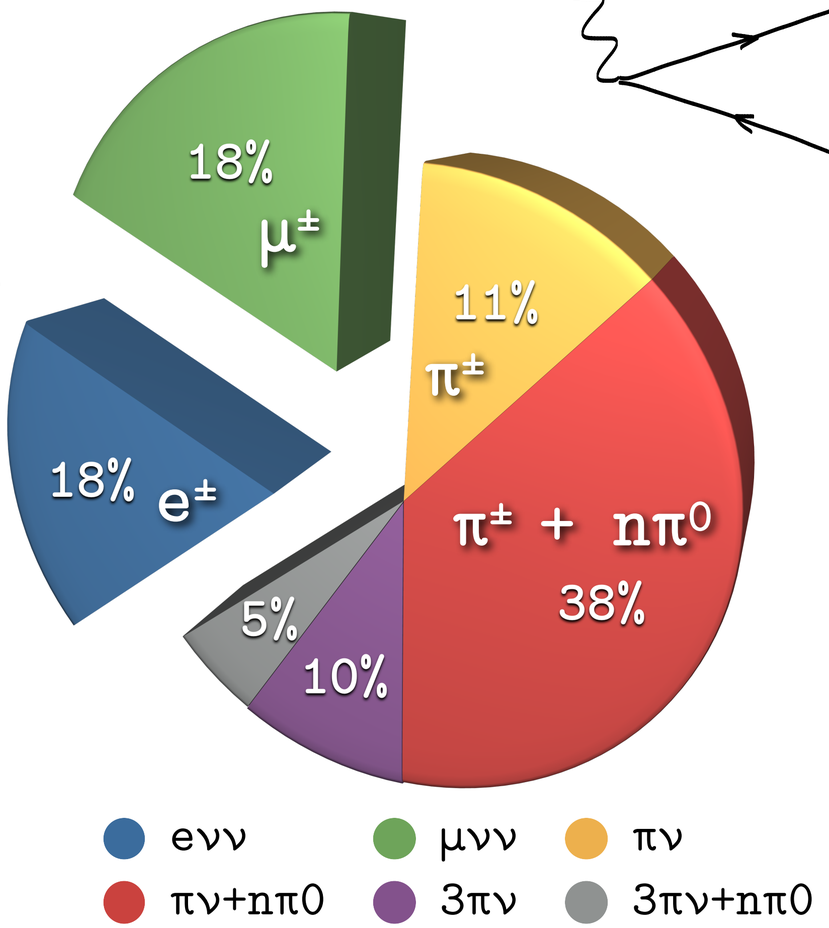


Tau ID & Reconstruction

- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87\ \mu m$
- leptonic decays: 35%
  - electron: 17.5%
  - muon: 17.5%
- hadronic decays: 65%
  - single prong: 49%
    - 11% $\pi^{\pm}$
    - 38% $\pi^{\pm} + n\pi^0$
  - three prong: 15%
    - 10% $\pi^{\pm}\ \pi^{\pm}\ \pi^{\mp}$

I won't talk about leptonic decays

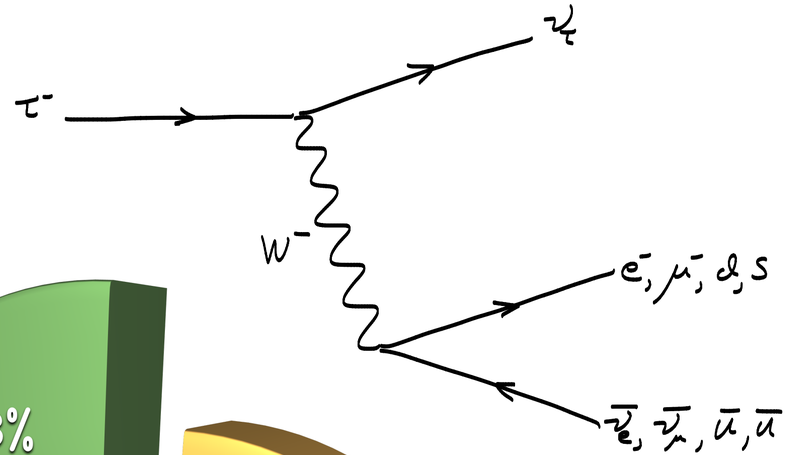


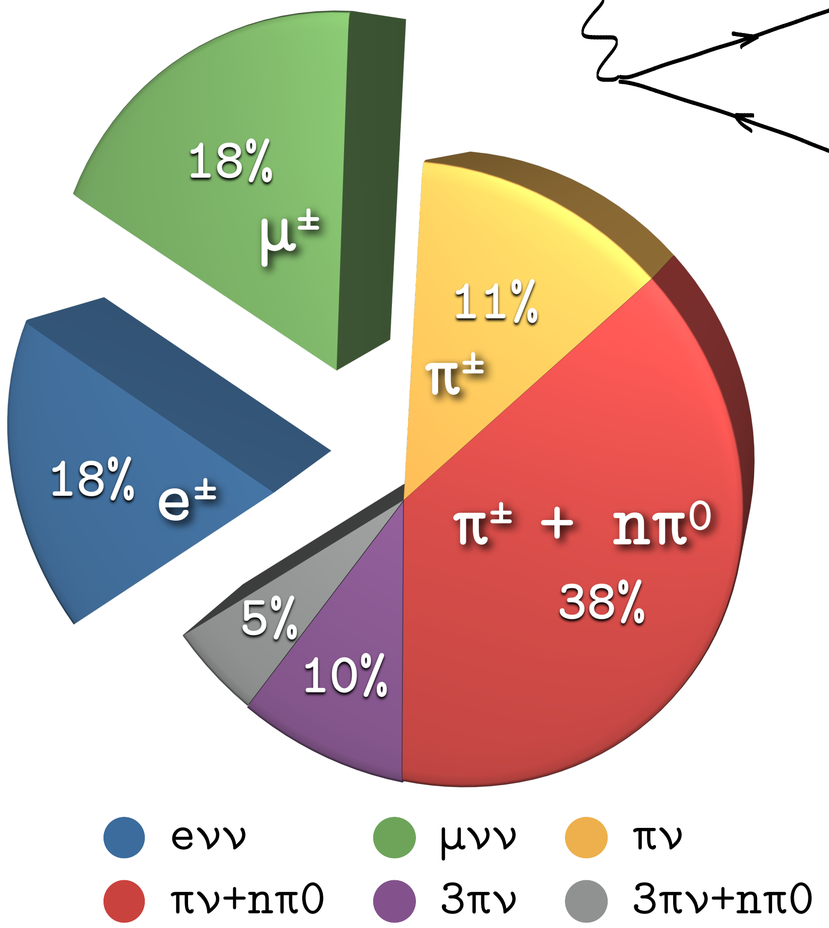


Tau ID & Reconstruction

- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87$ μm
- leptonic decays: 35%
  - electron: 17.5%
  - muon: 17.5%
- hadronic decays: 65%
  - single prong: 49%
    - 11% $\pi^{\pm}$
    - 38% $\pi^{\pm} + n\pi^0$
  - three prong: 15%
    - 10% $\pi^{\pm}\ \pi^{\pm}\ \pi^{\mp}$
    - 5% $\pi^{\pm}\ \pi^{\pm}\ \pi^{\mp} + n\pi^0$

I won't talk about leptonic decays

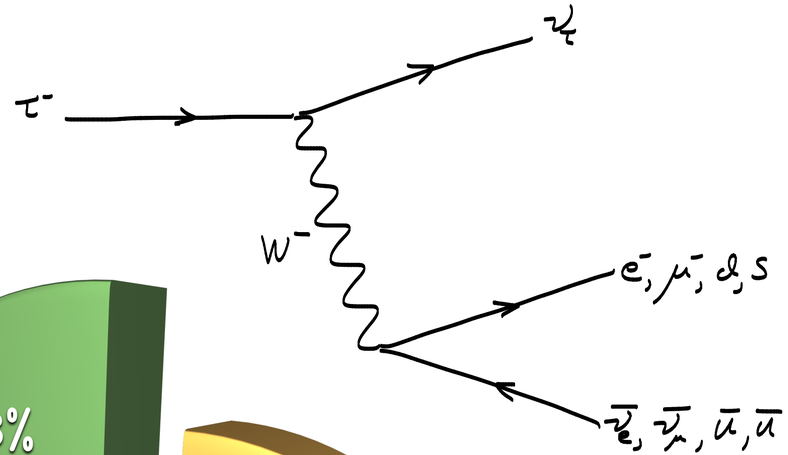


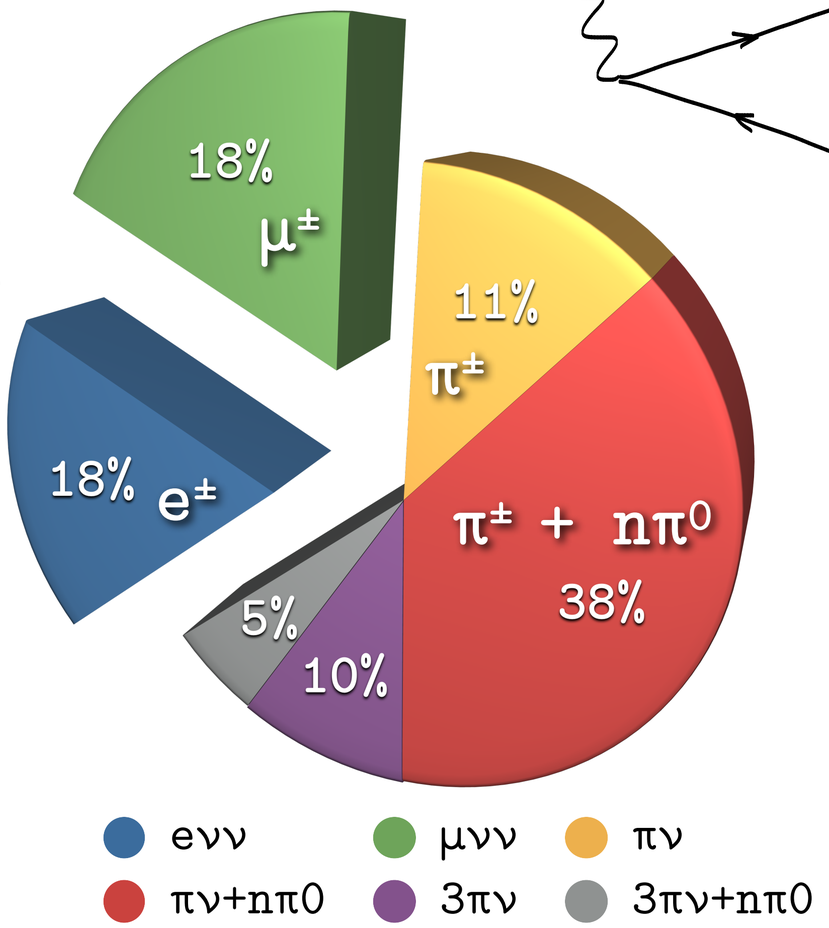


Tau ID & Reconstruction

- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87\ \mu m$
- leptonic decays: 35%
  - electron: 17.5%
  - muon: 17.5%
- hadronic decays: 65%
  - single prong: 49%
    - 11% $\pi^{\pm}$
    - 38% $\pi^{\pm} + n\pi^0$
  - three prong: 15%
    - 10% $\pi^{\pm}\ \pi^{\pm}\ \pi^{\mp}$
    - 5% $\pi^{\pm}\ \pi^{\pm}\ \pi^{\mp} + n\pi^0$
- Experimental inefficiencies and fakes

I won't talk about leptonic decays

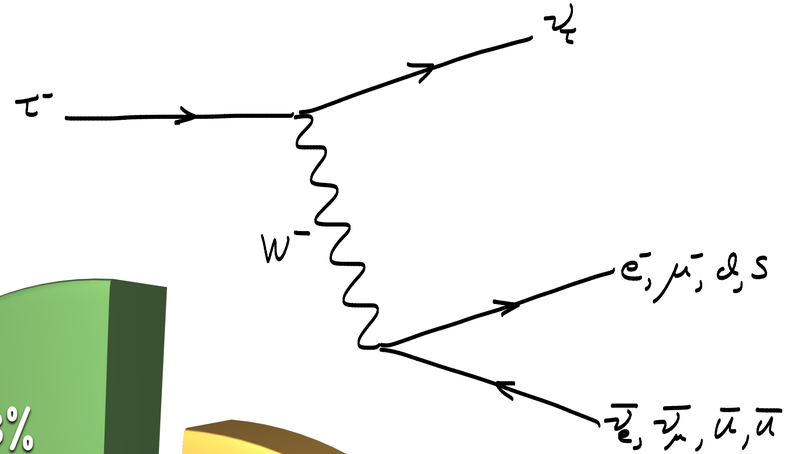


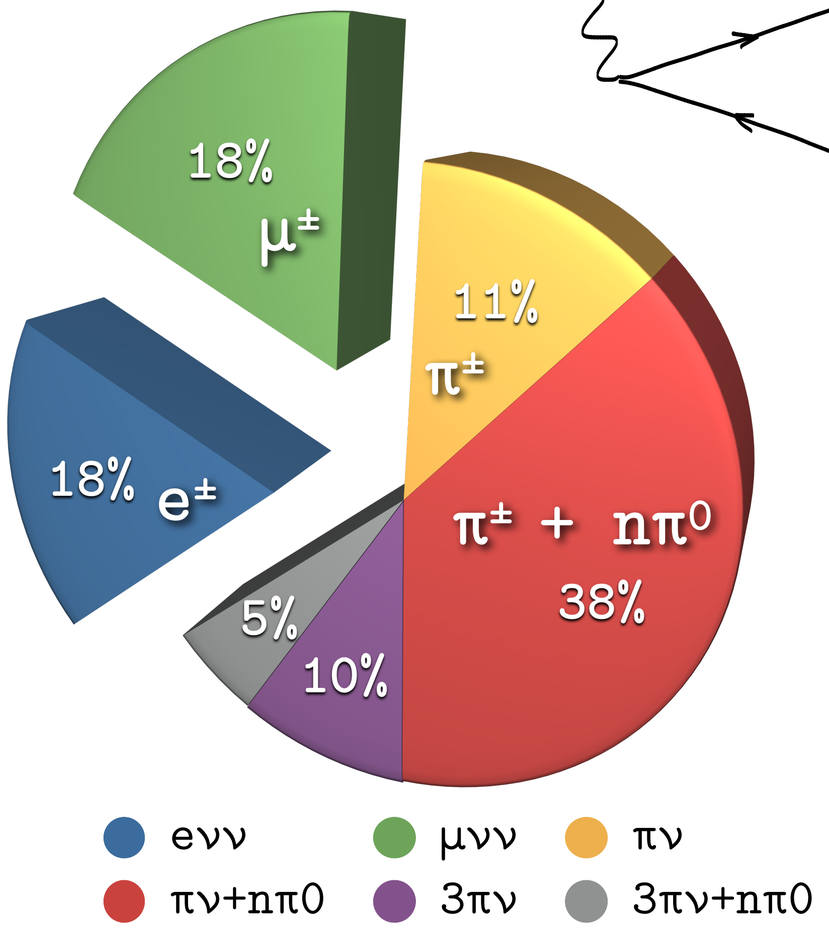


Tau ID & Reconstruction

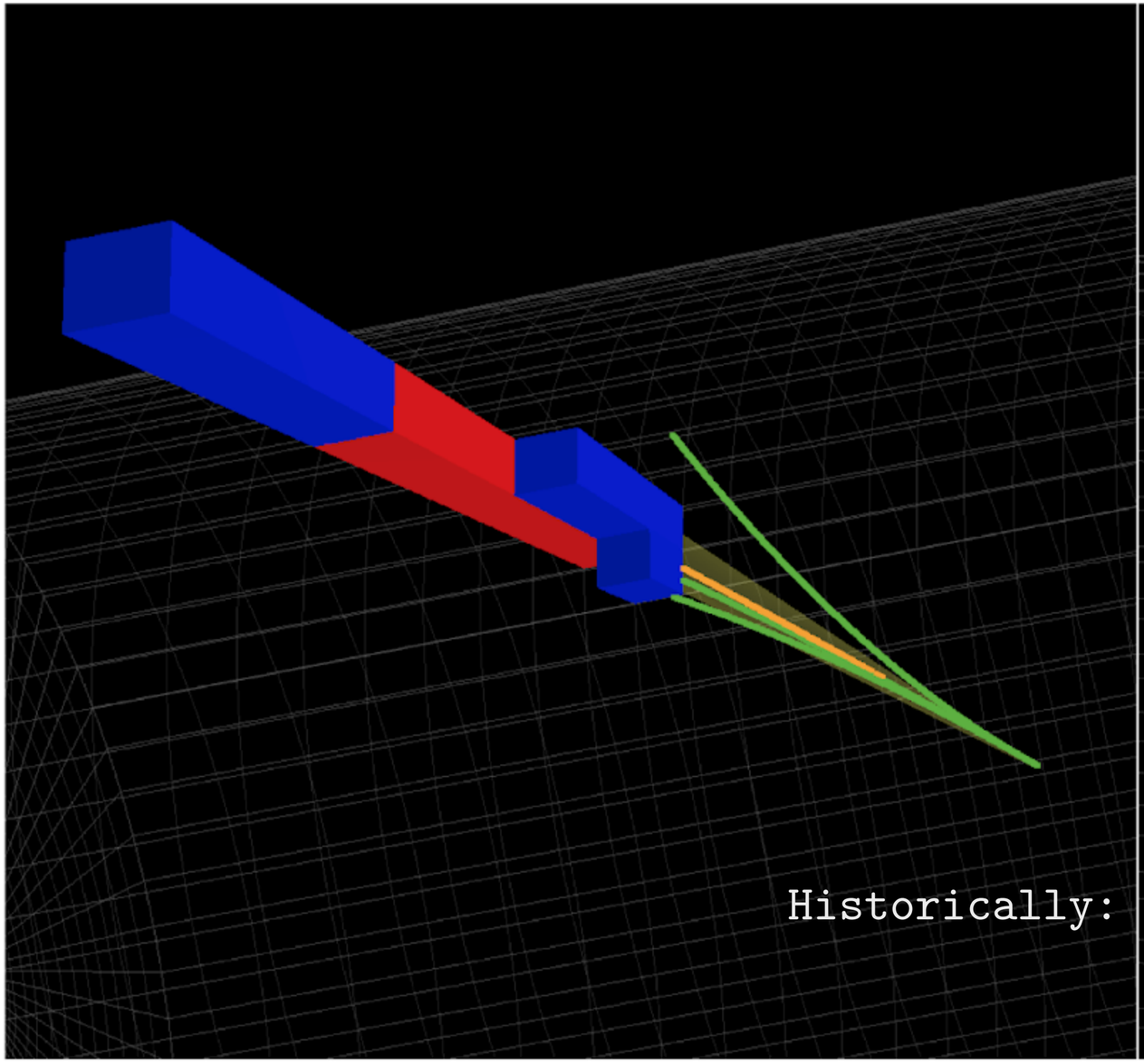
- Massive, (relatively) long lived
  - $m(\tau^{\pm}) = 1.7$ GeV
  - $c\tau = 87\ \mu m$
- leptonic decays: 35%
  - electron: 17.5%
  - muon: 17.5%
- hadronic decays: 65%
  - single prong: 49%
    - 11% $\pi^{\pm}$
    - 38% $\pi^{\pm} + n\pi^0$
  - three prong: 15%
    - 10% $\pi^{\pm}\ \pi^{\pm}\ \pi^{\mp}$
    - 5% $\pi^{\pm}\ \pi^{\pm}\ \pi^{\mp} + n\pi^0$
- Experimental inefficiencies and fakes
  - reality: 0,1,2,3,4 pions + 0,1,2,3,4+ photons

I won't talk about leptonic decays

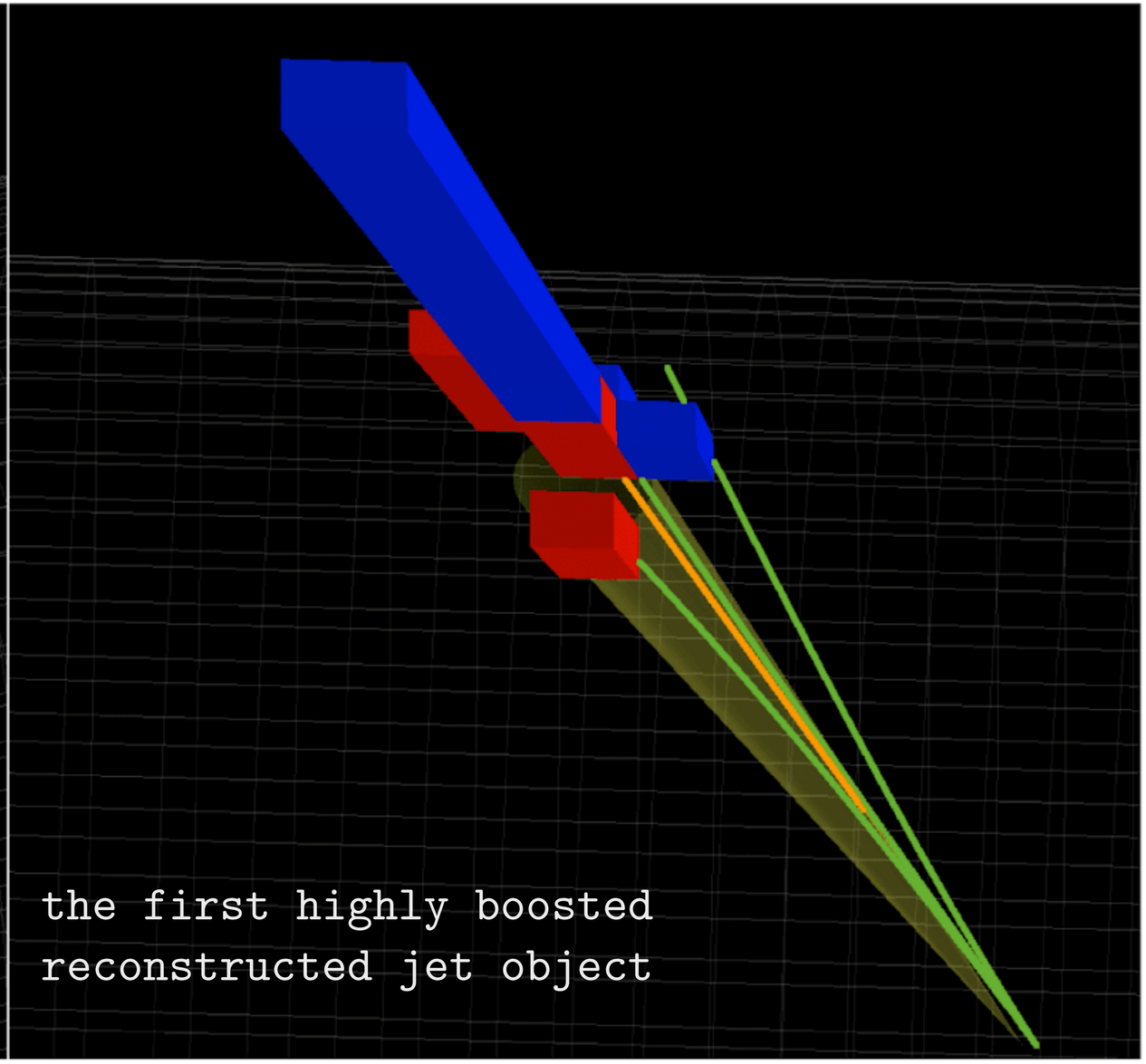




Tau ID & Reconstruction



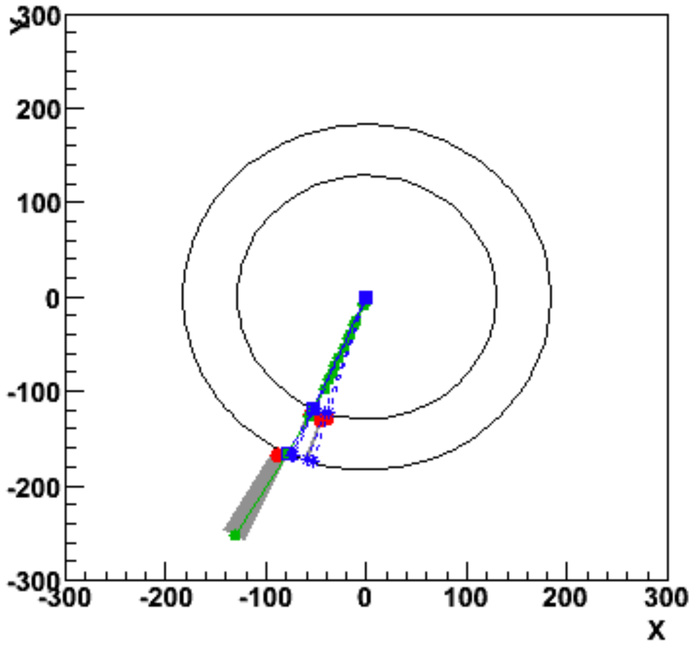
Historically:



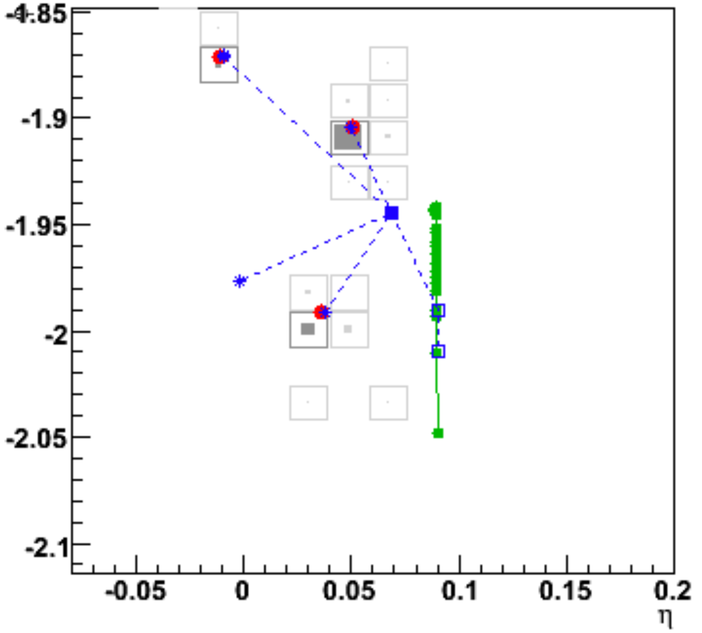
the first highly boosted reconstructed jet object

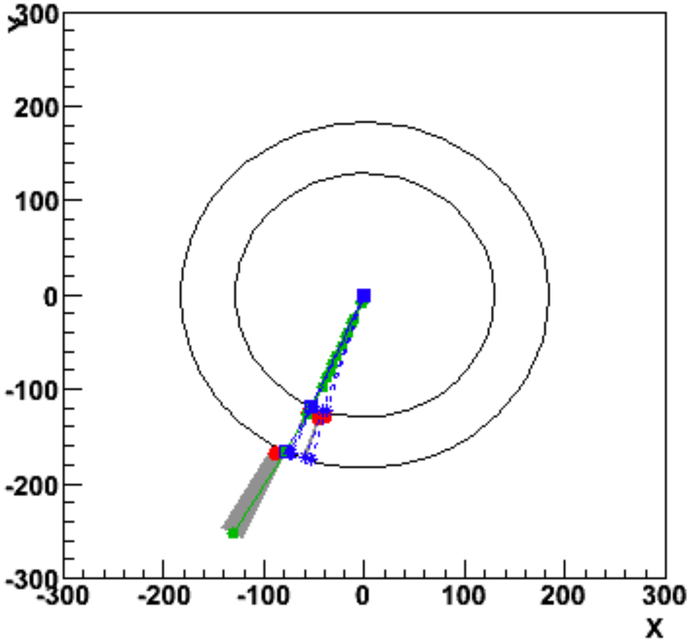
Tau ID & Reconstruction

Tau ID & Reconstruction

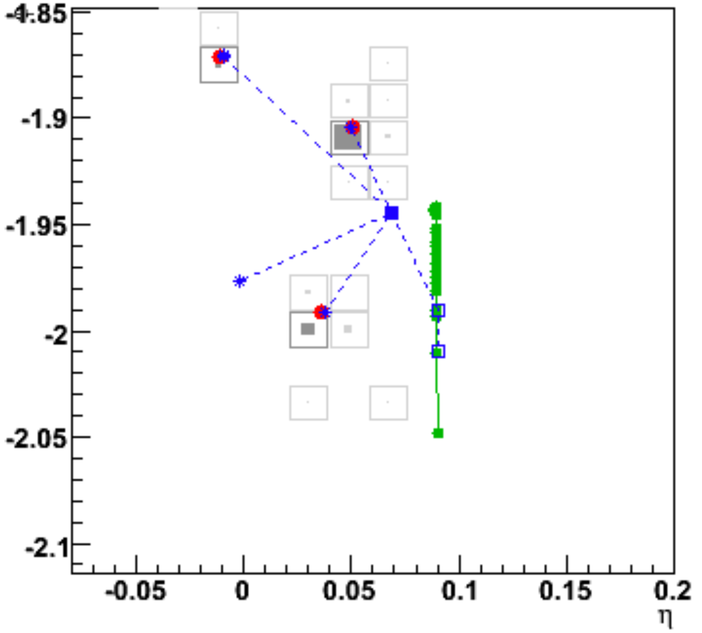


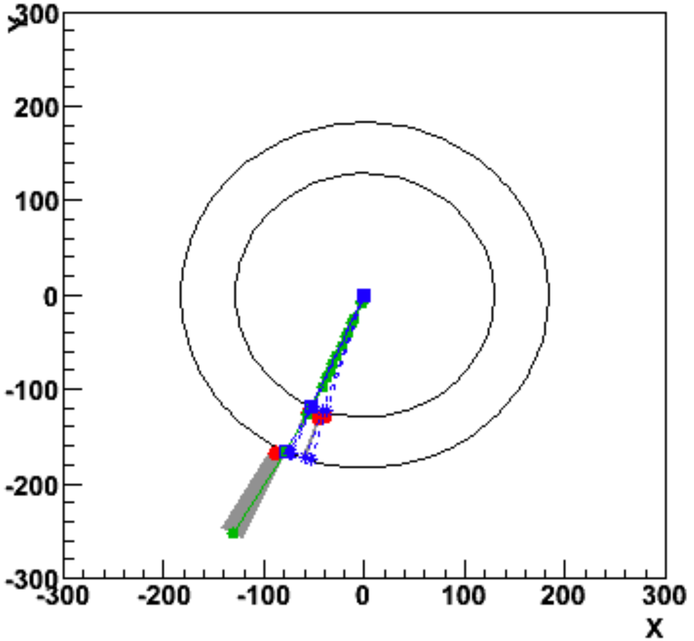
Tau ID & Reconstruction

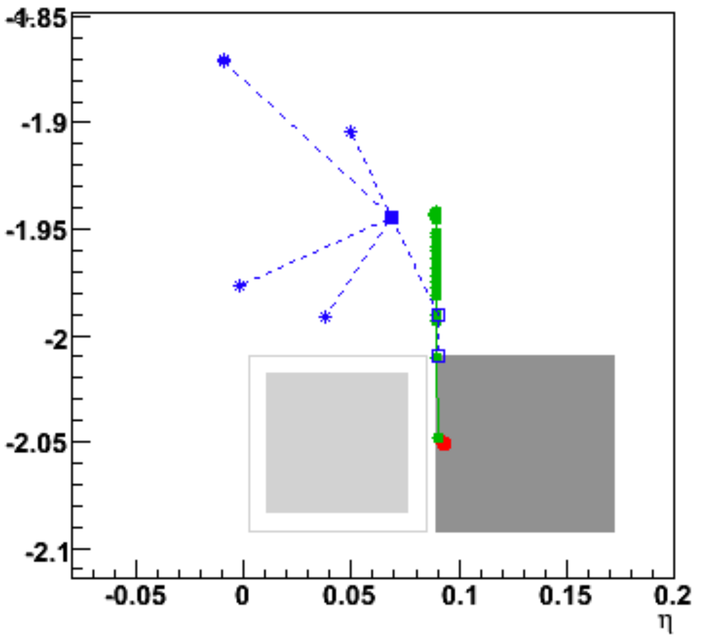




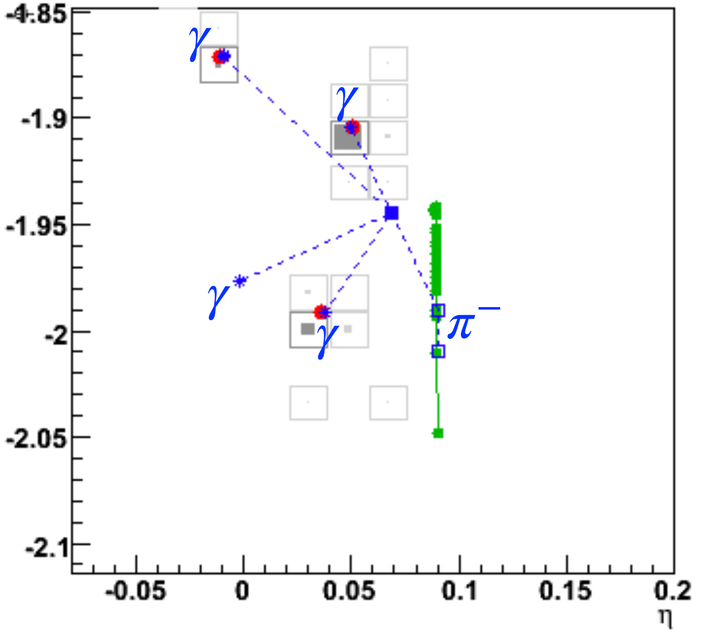
Tau ID & Reconstruction

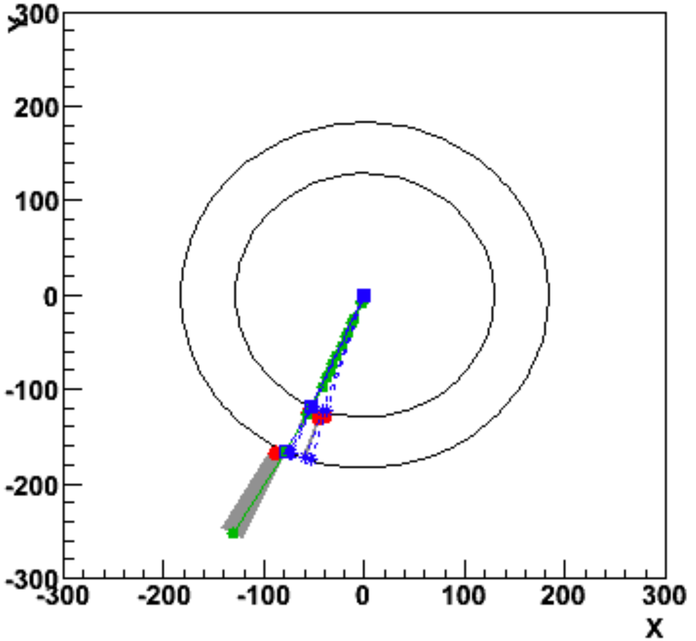


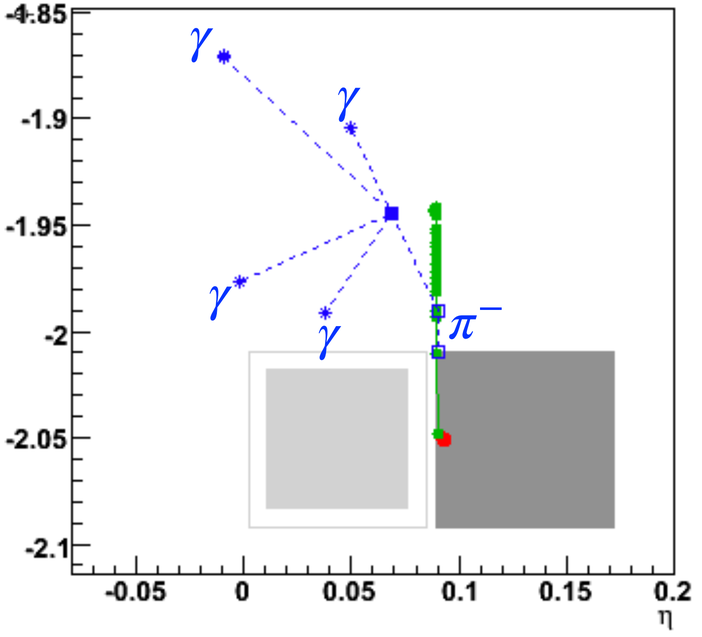




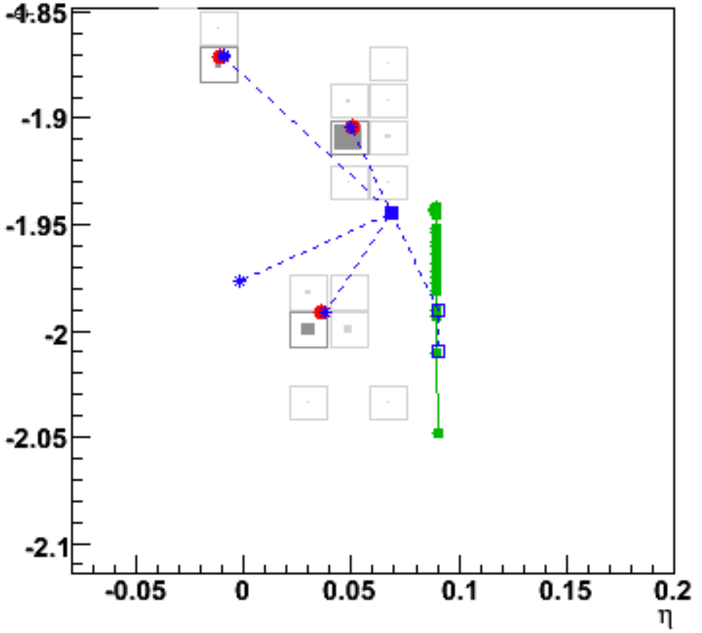
Tau ID & Reconstruction

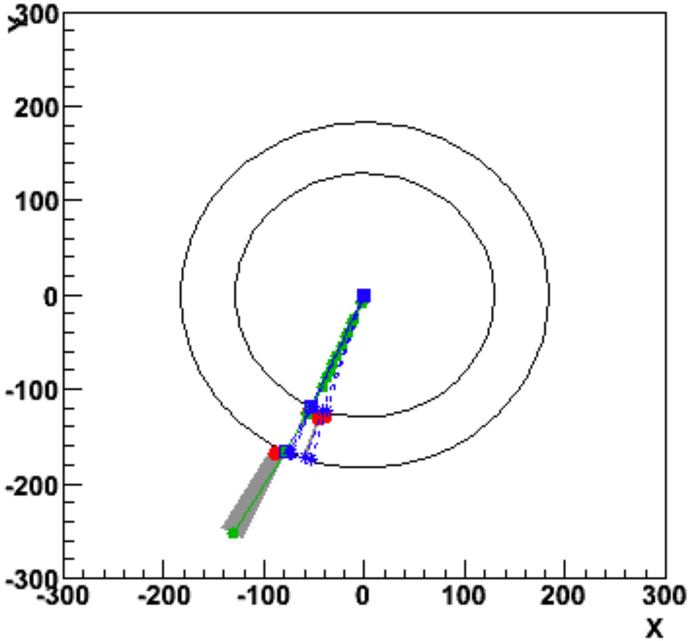


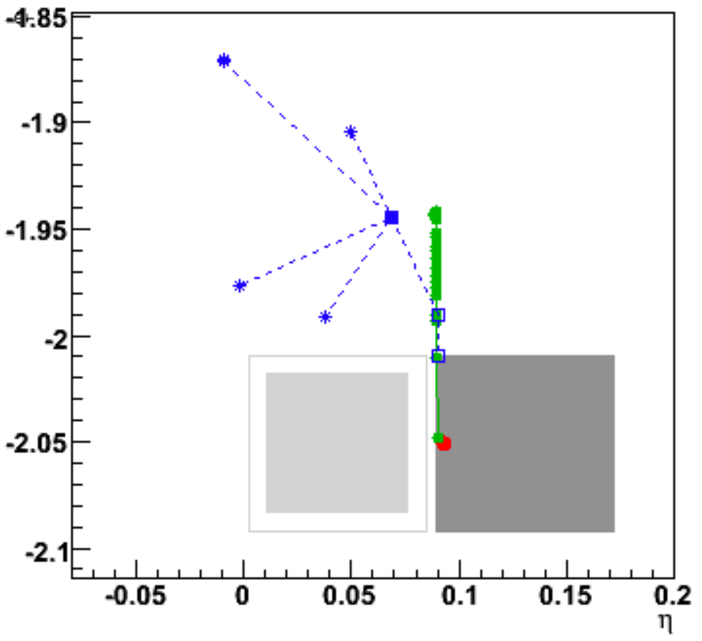




Tau ID & Reconstruction



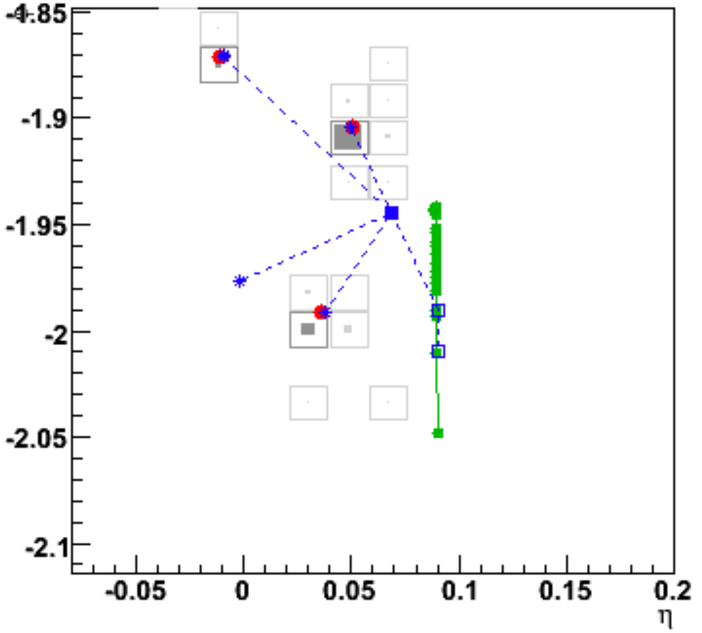


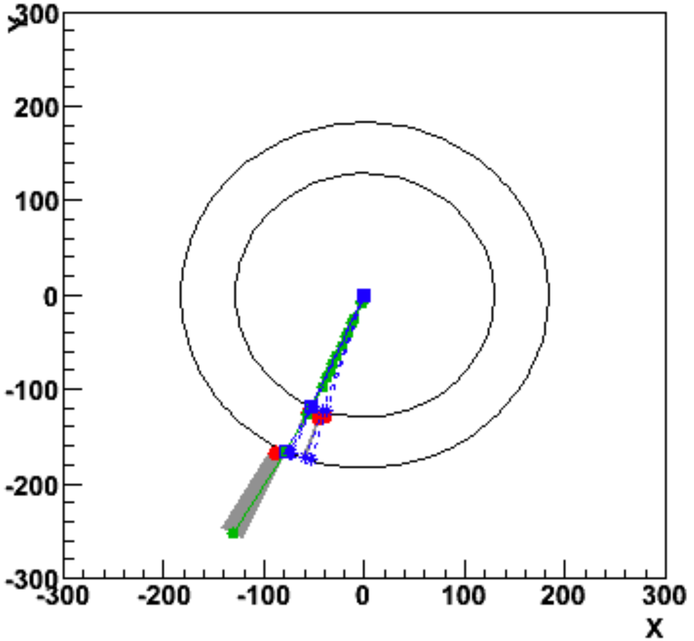


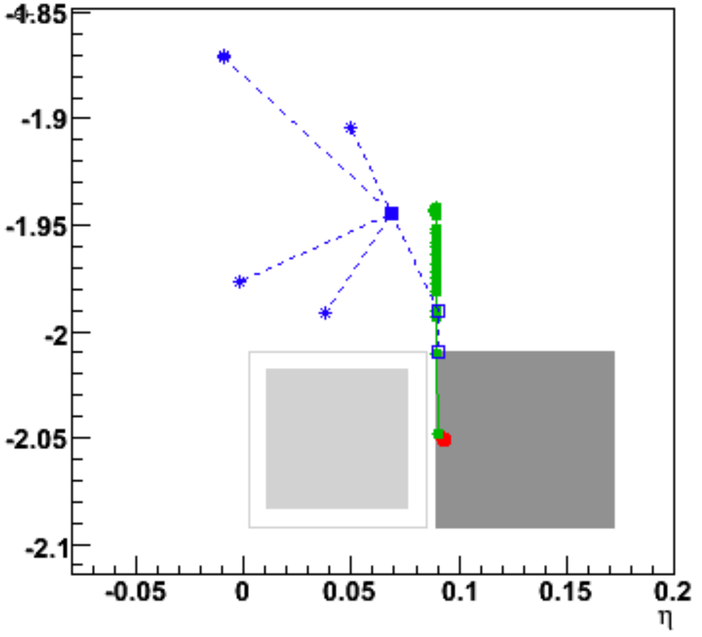
Tau ID & Reconstruction

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta \times \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates





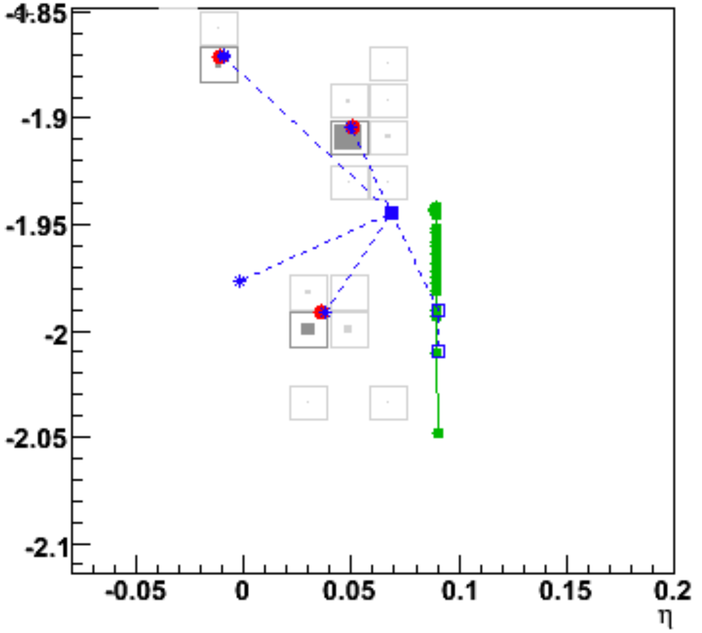


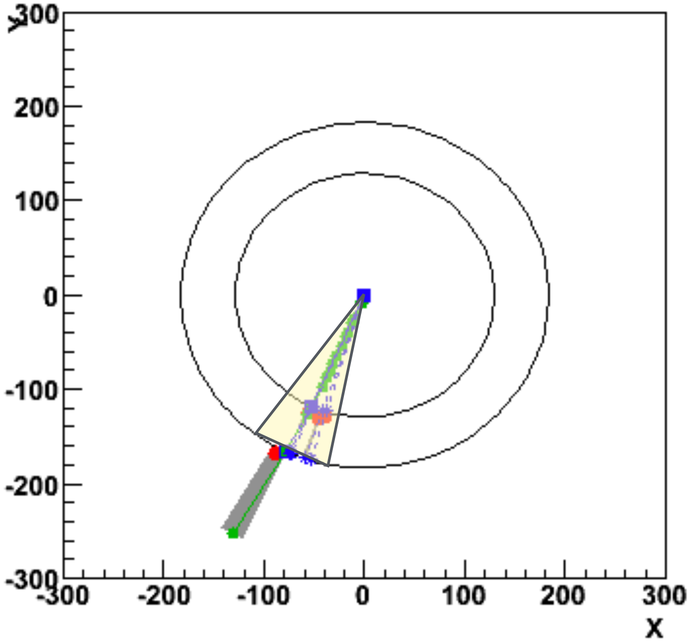
Tau ID & Reconstruction

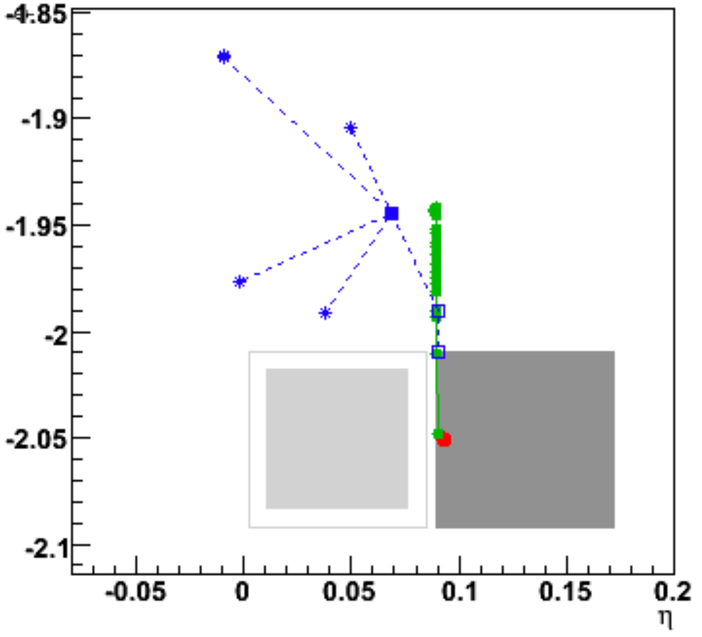
Richard Cavanaugh, Fermilab/UIC, HCP Summer School 31 July 2024

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates





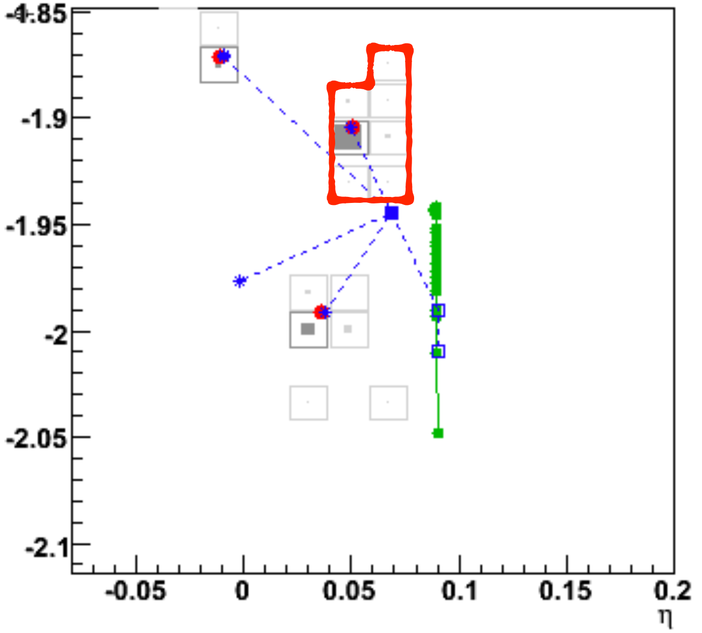


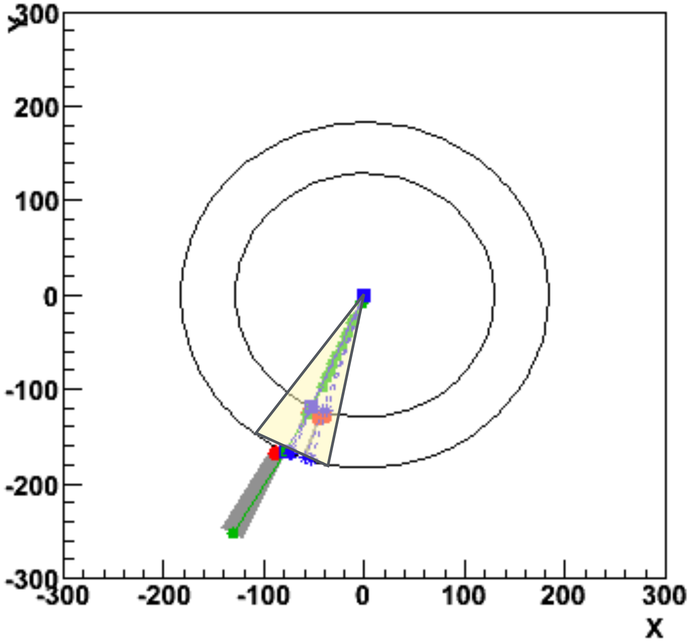
Tau ID & Reconstruction

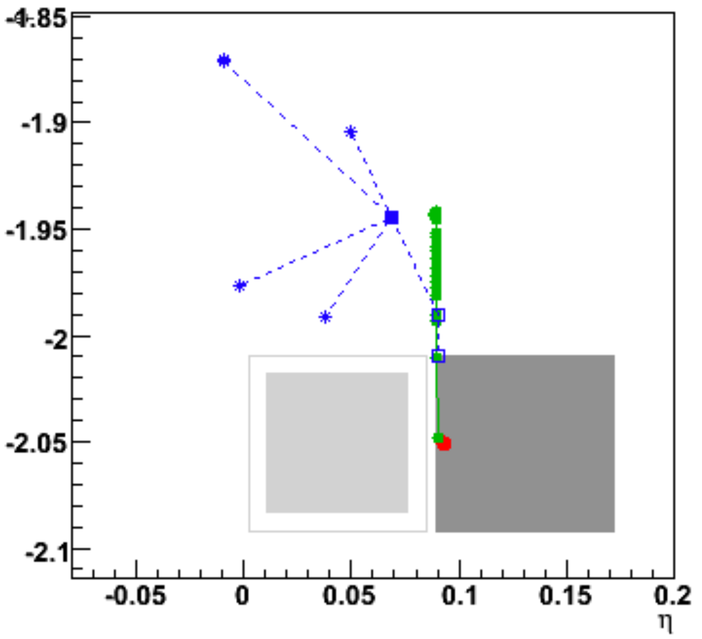
Richard Cavanaugh, Fermilab/UIC, HCP Summer School 31 July 2024

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates



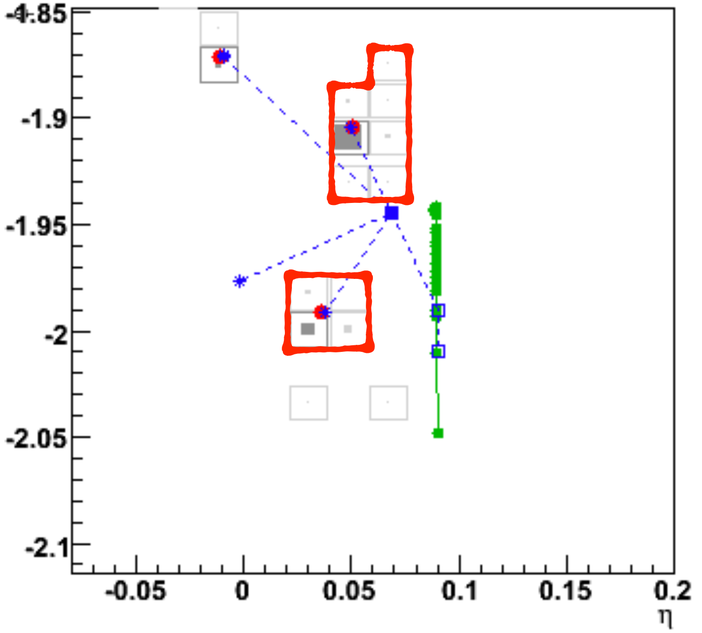


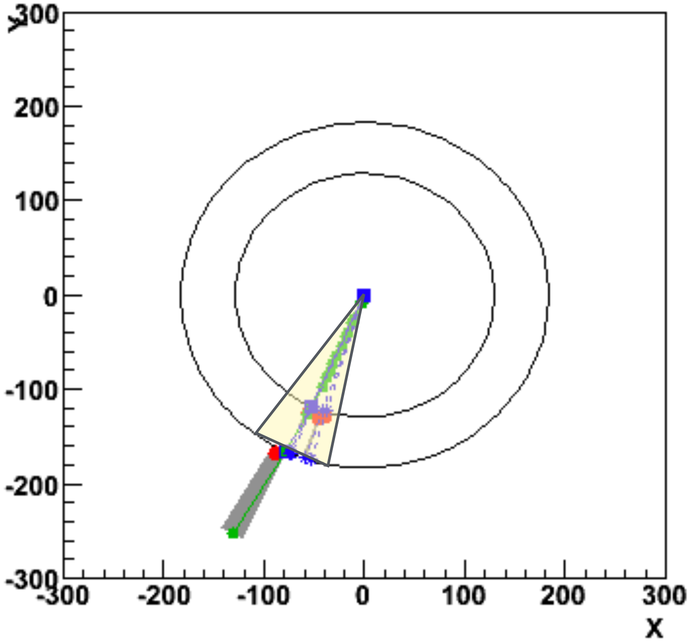


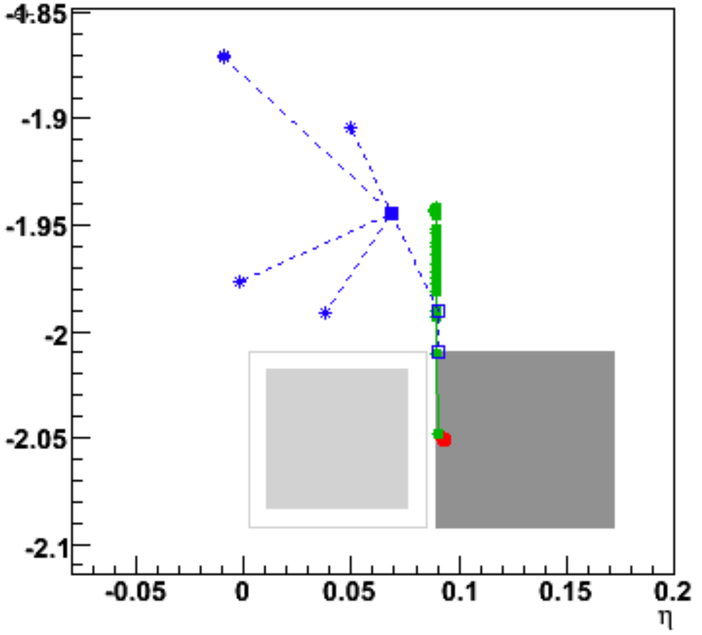
Tau ID & Reconstruction

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates



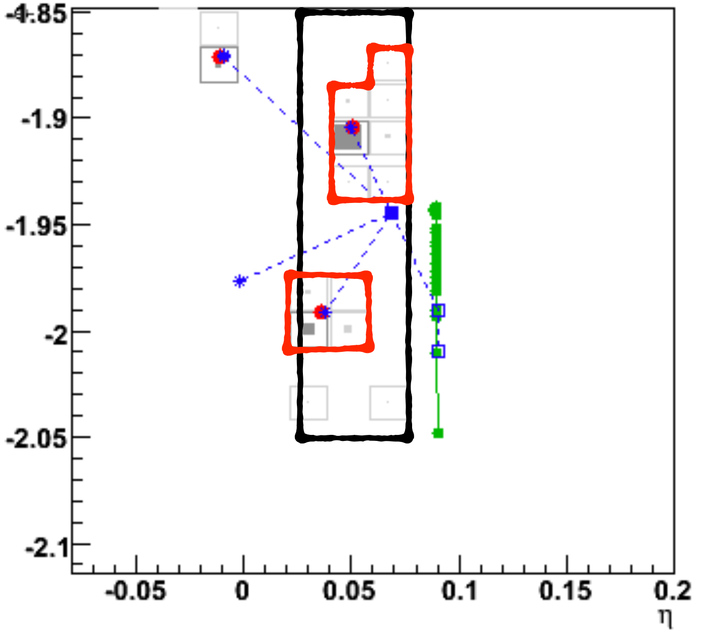


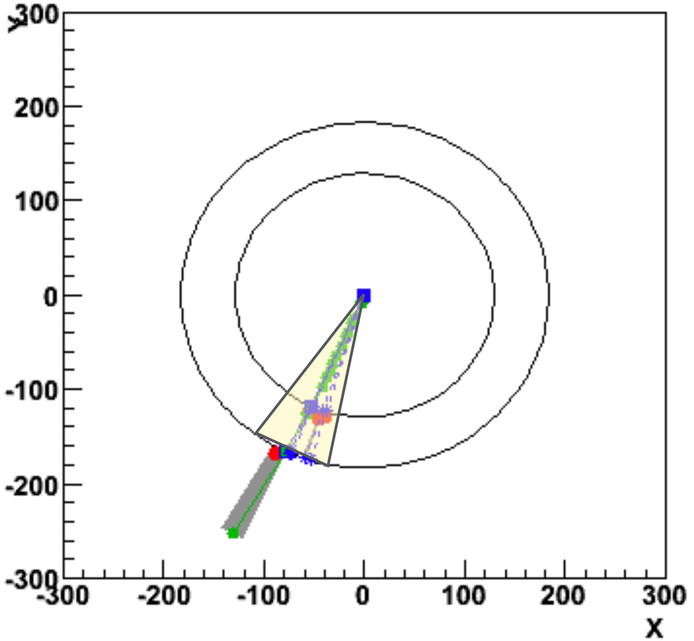


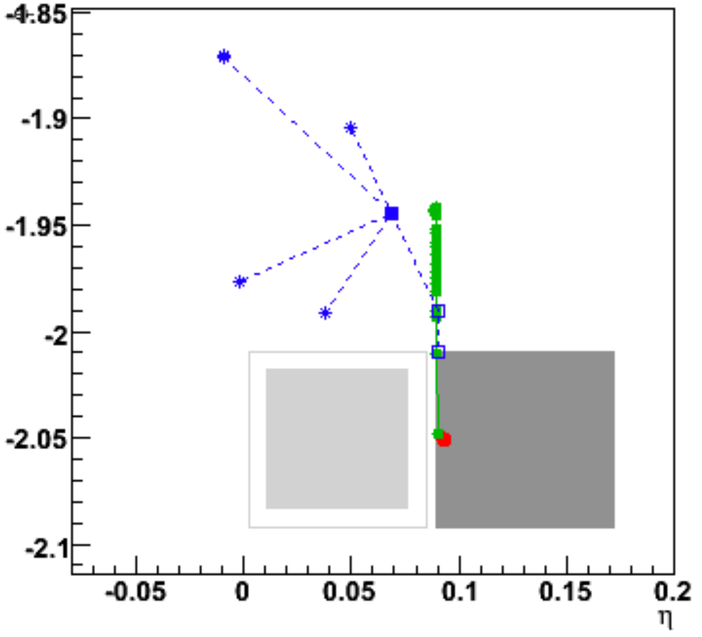
Tau ID & Reconstruction

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates



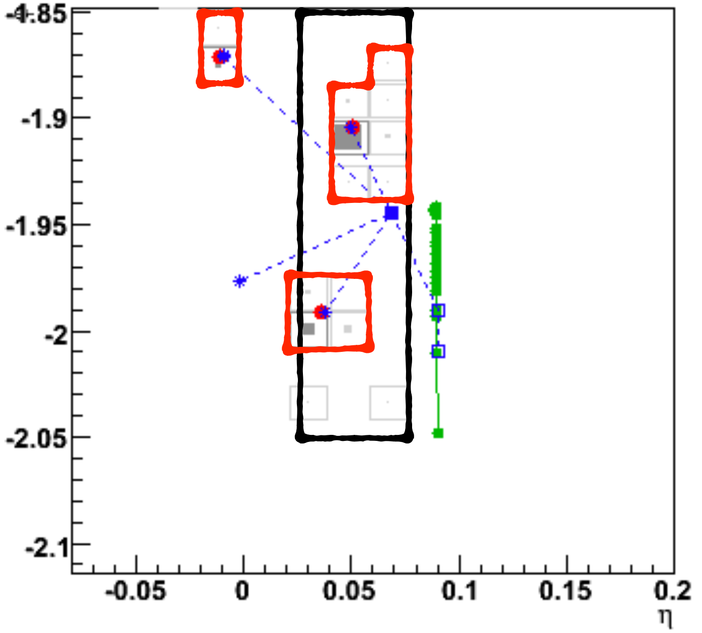


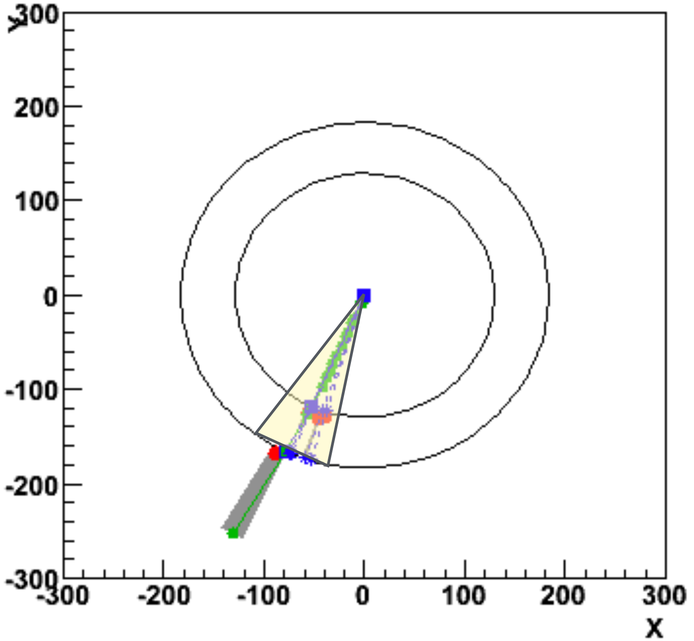


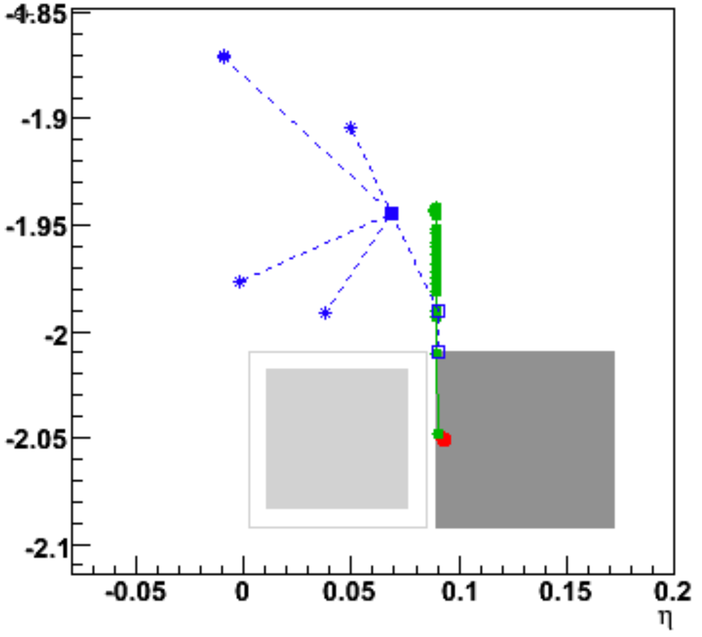
Tau ID & Reconstruction

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates



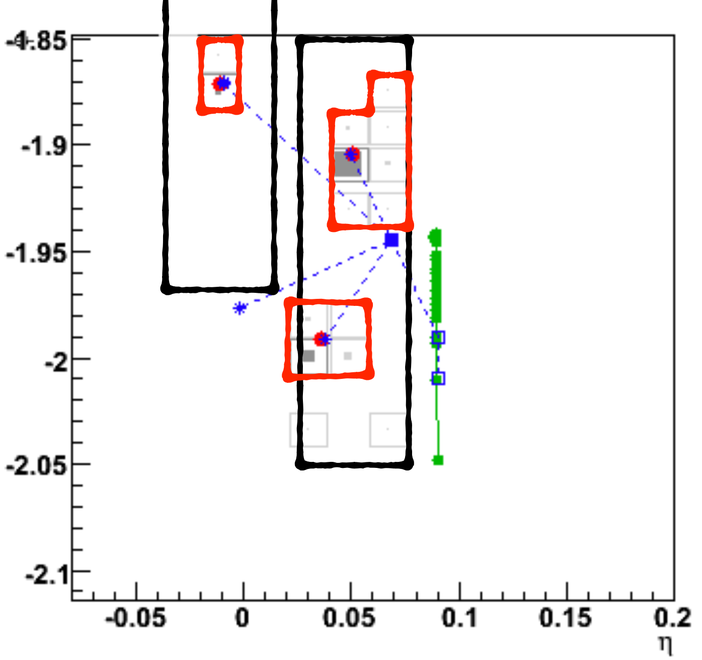


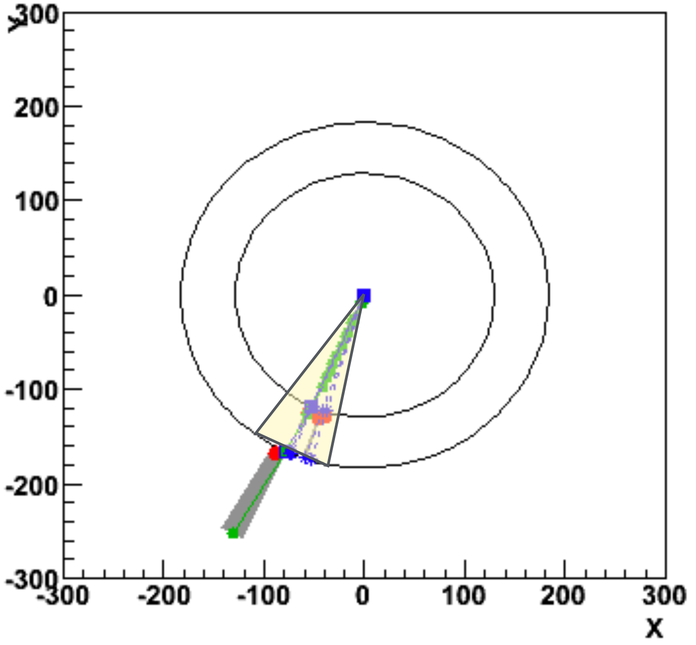


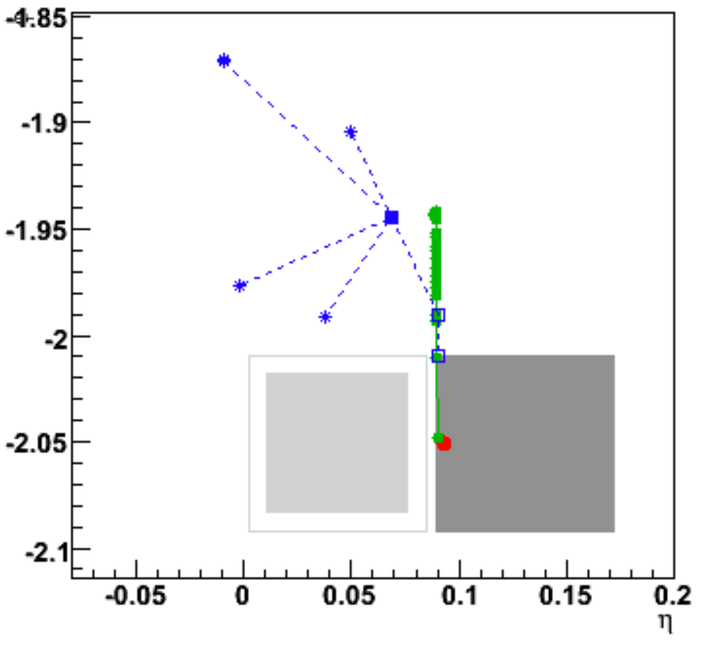
Tau ID & Reconstruction

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates



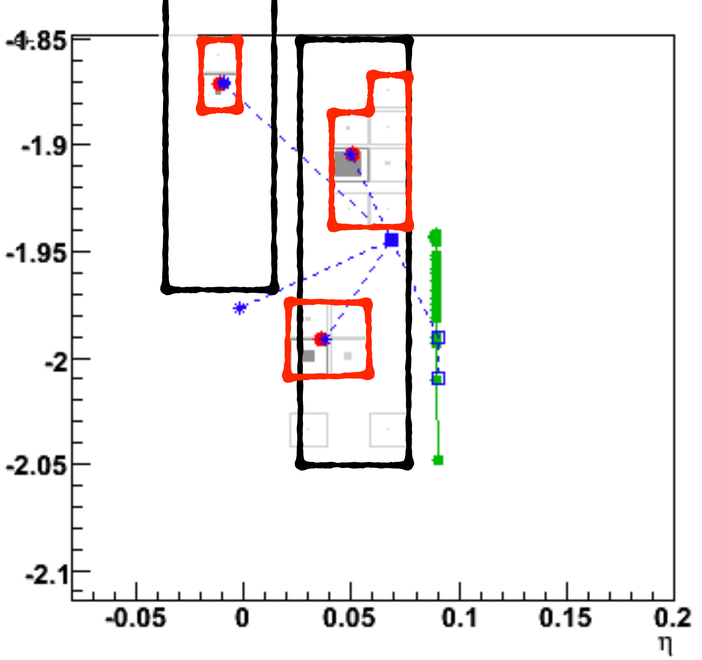


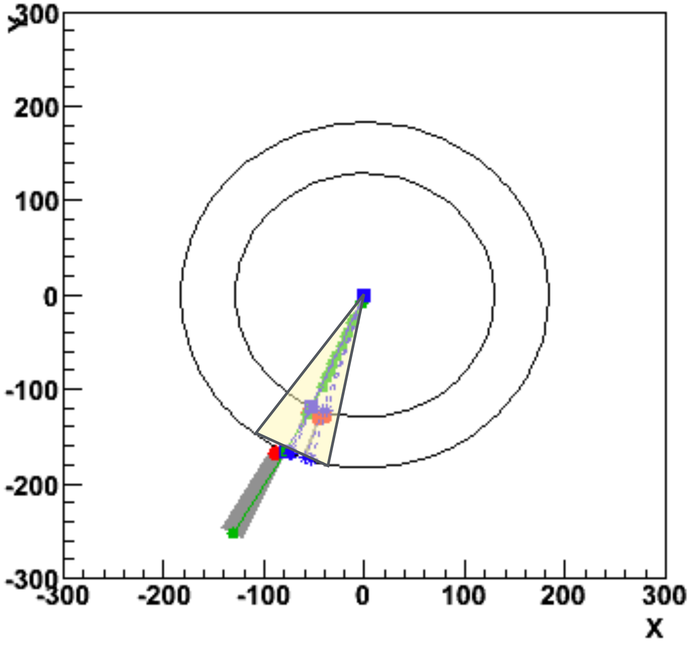


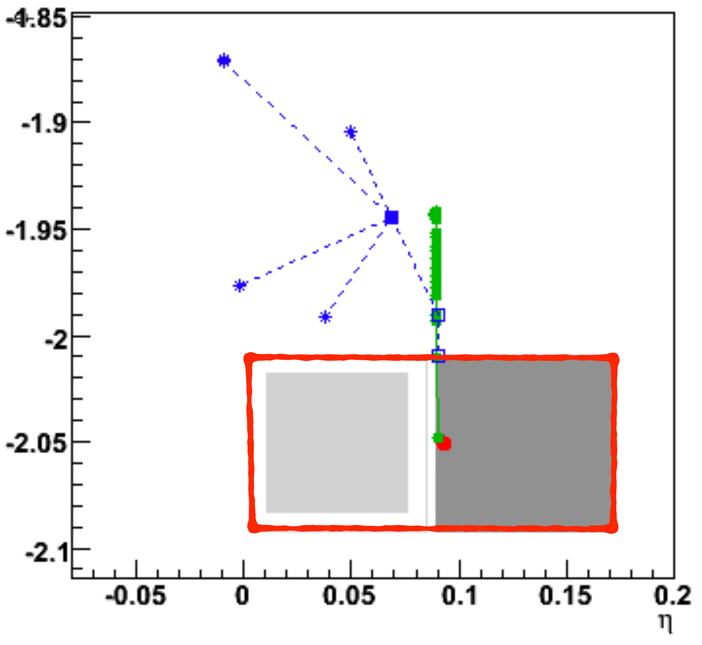
Tau ID & Reconstruction

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates







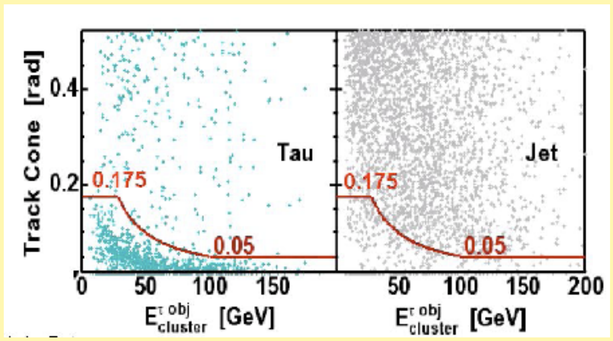
Tau ID & Reconstruction

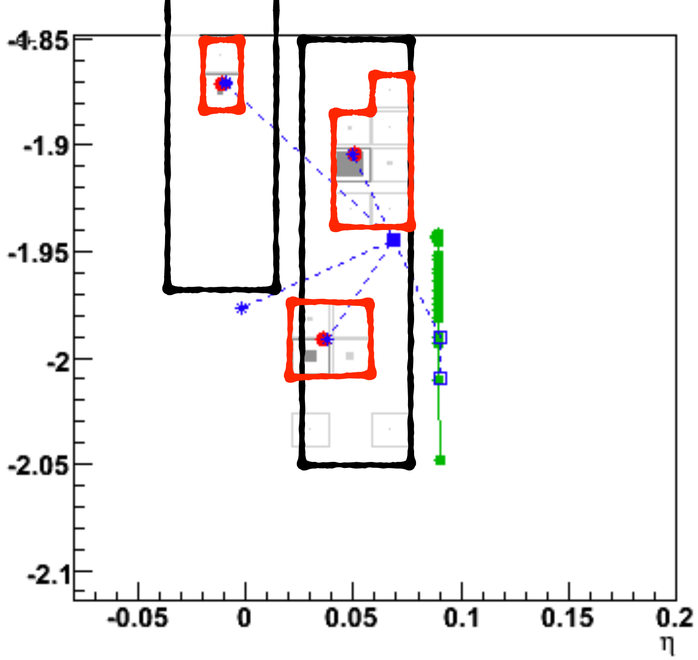
Basic Idea

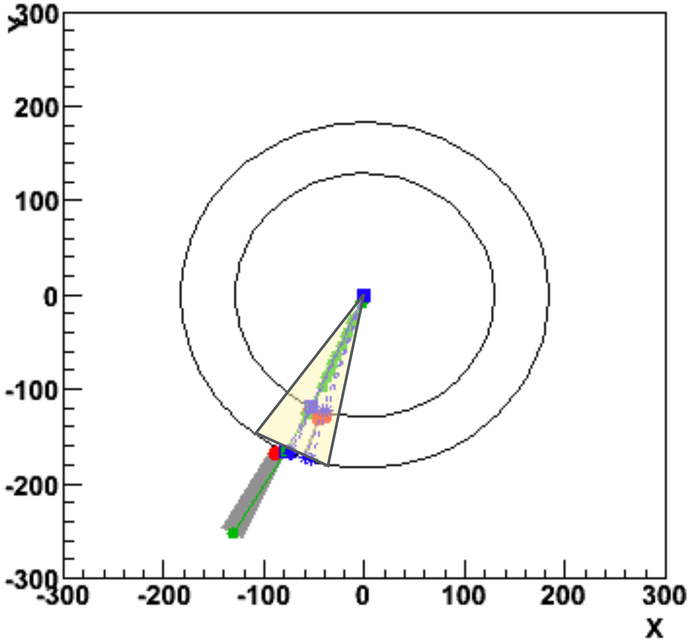
- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates

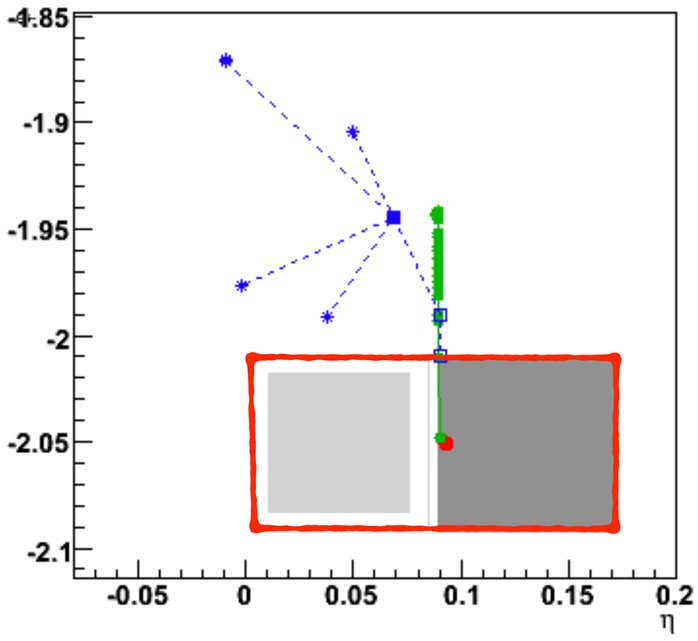
Signal Cone

- Tau constituents must lie within a shrinking $\Delta R \leq 3\ GeV / pT(\tau)$









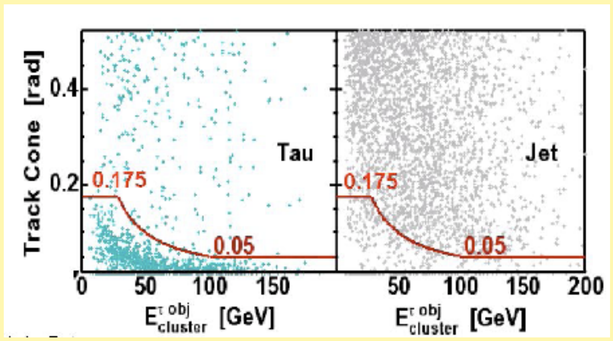
Tau ID & Reconstruction

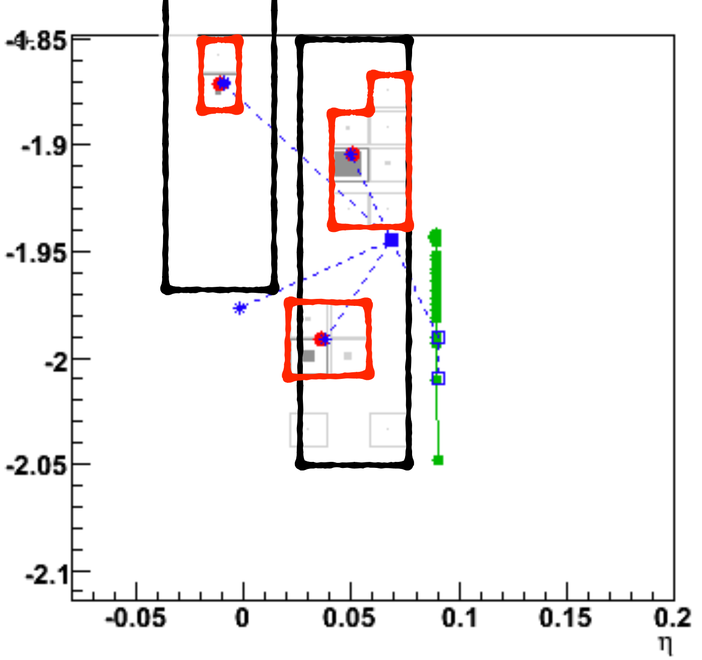
Basic Idea

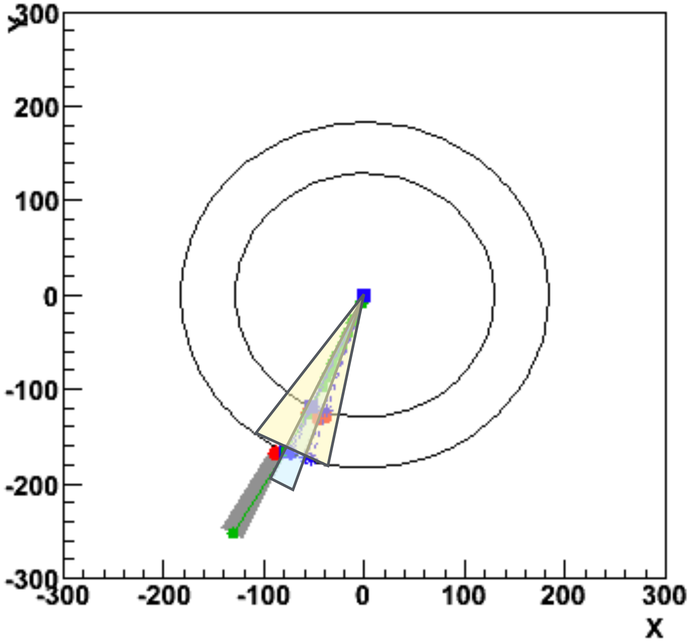
- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates

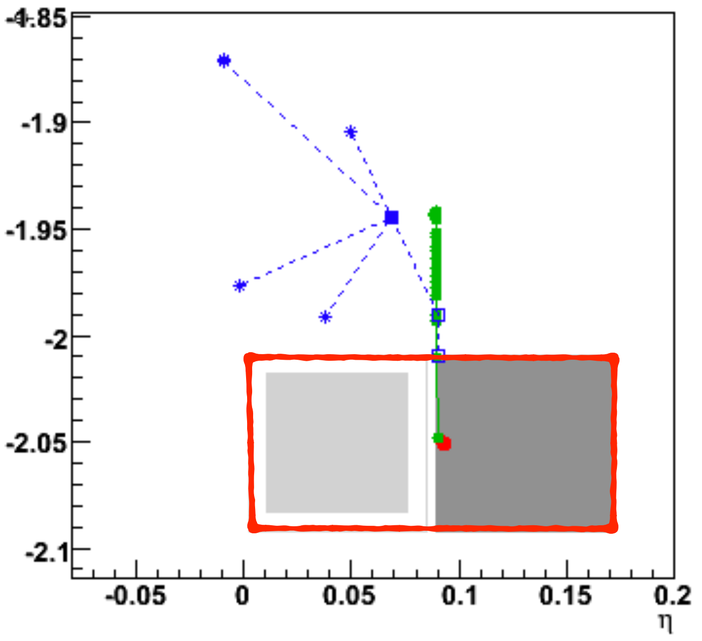
Signal Cone

- Tau constituents must lie within a shrinking $\Delta R \leq 3\ \text{GeV} / pT(\tau)$









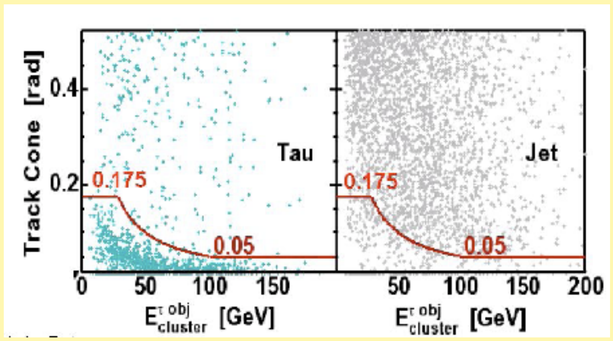
Tau ID & Reconstruction

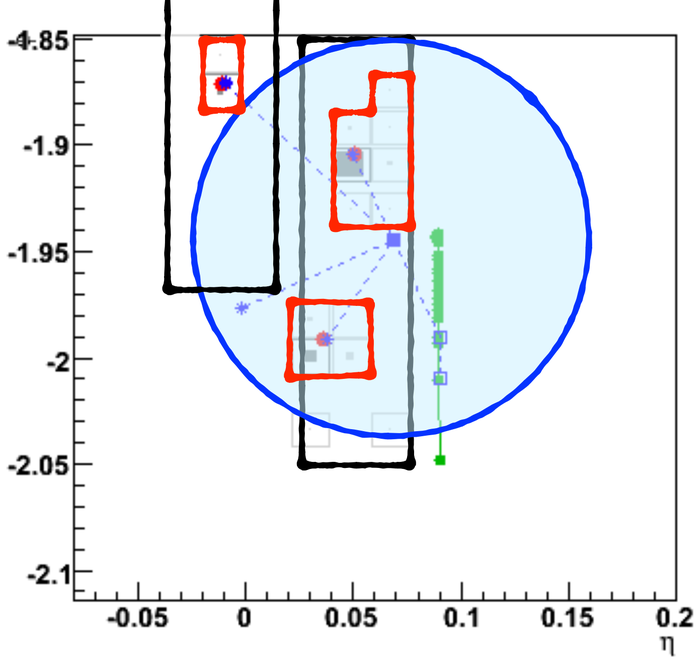
Basic Idea

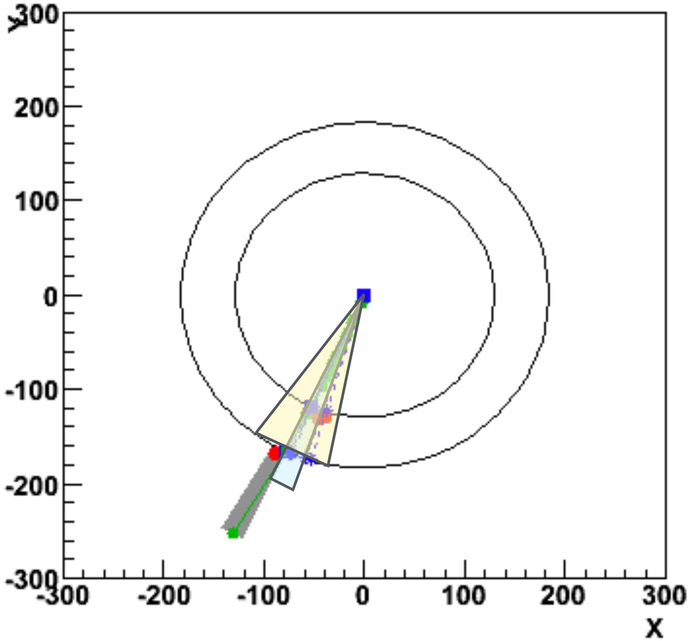
- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates

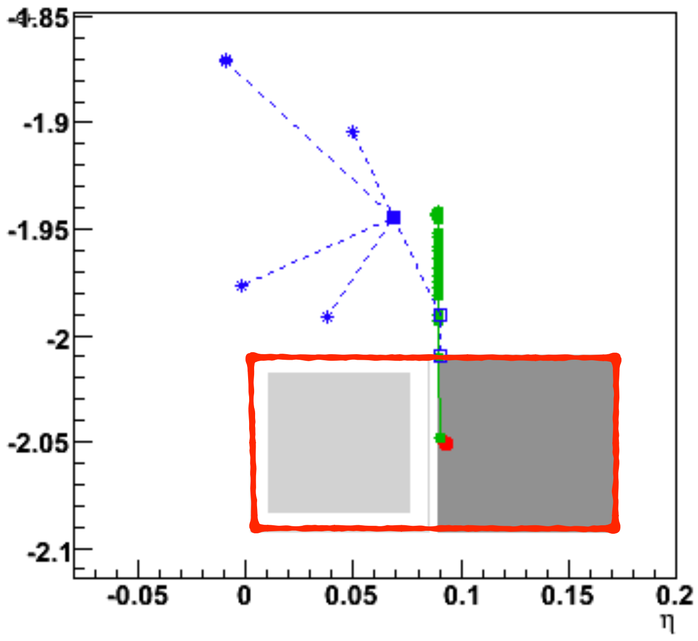
Signal Cone

- Tau constituents must lie within a shrinking $\Delta R \leq 3\ GeV / pT(\tau)$









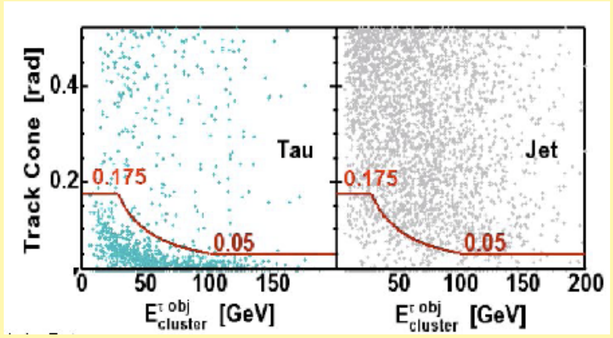
Tau ID & Reconstruc

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates

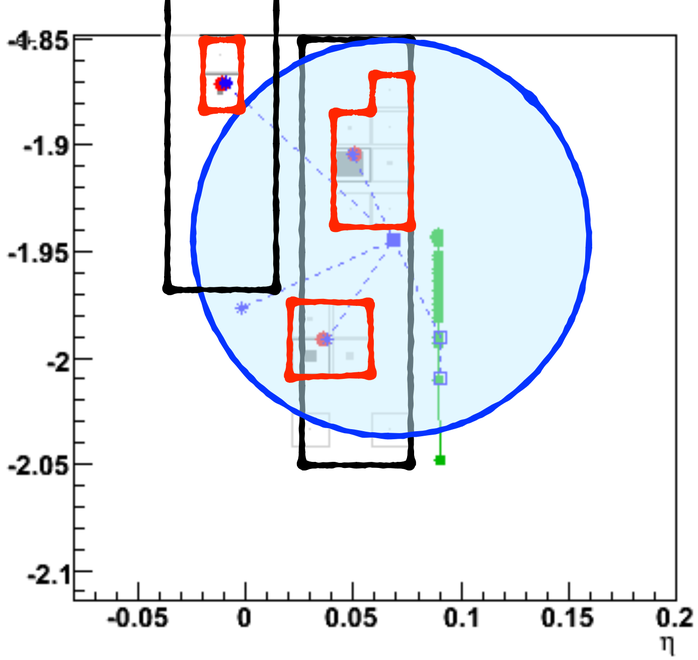
Signal Cone

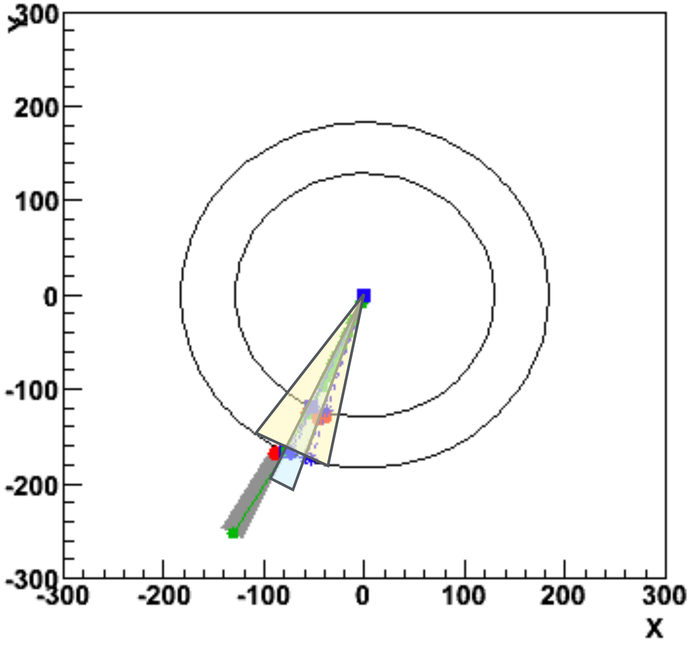
- Tau constituents must lie within a shrinking $\Delta R \leq 3\ GeV\ /\ pT(\tau)$

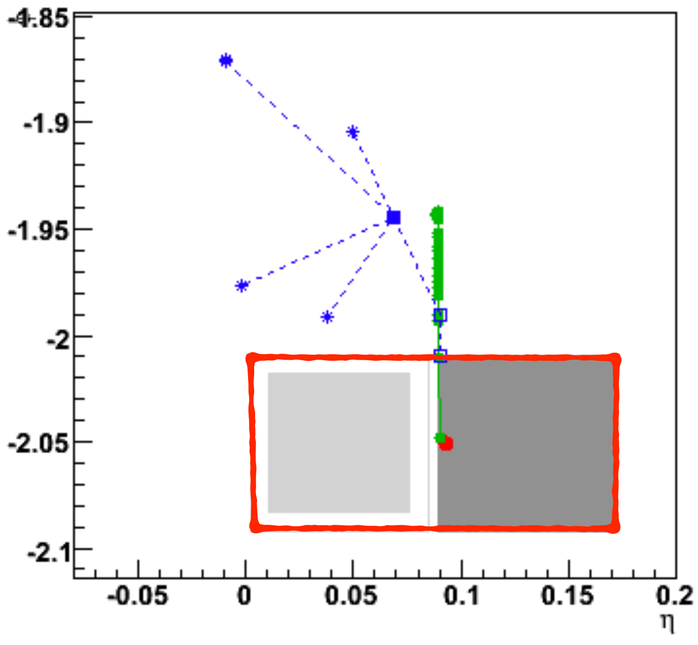


1 $\pi^{\pm}$, 0 π^0

- Branching Fraction: 11.6%
- Single isolated $\pi^{\pm}$







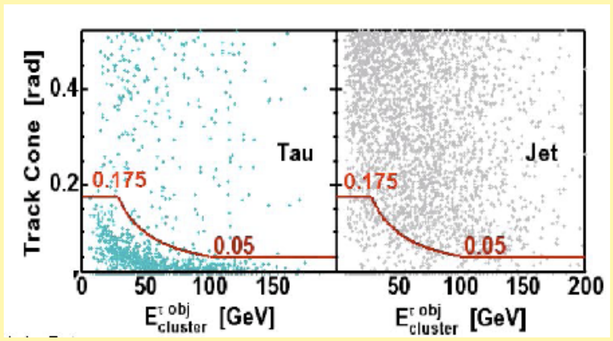
Tau ID & Reconstru

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates

Signal Cone

- Tau constituents must lie within a shrinking $\Delta R \leq 3\ GeV / pT(\tau)$

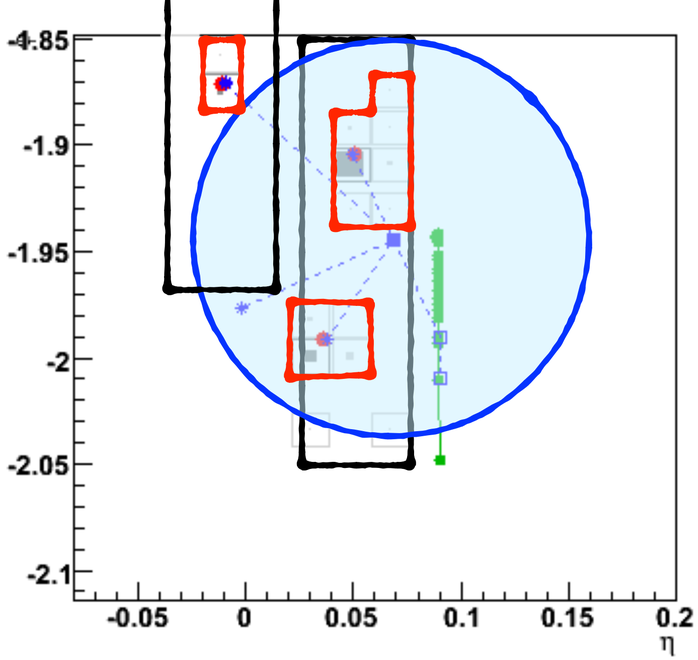


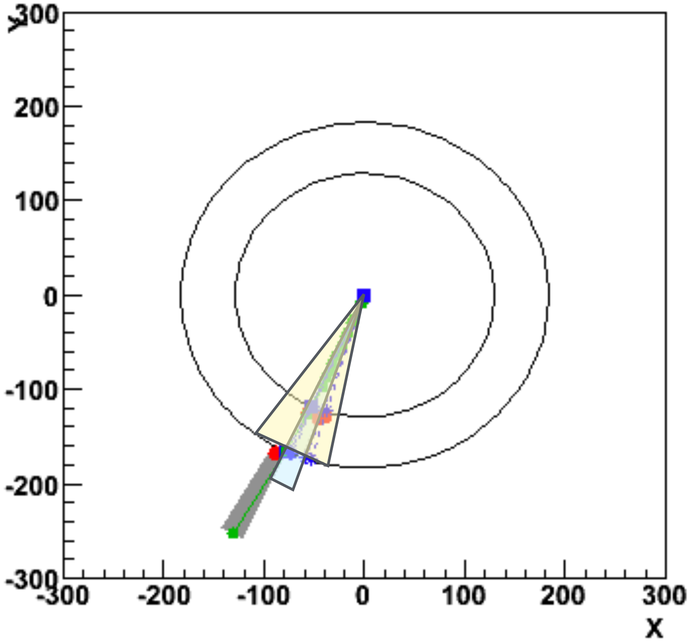
1 $\pi^{\pm}$, 0 π^0

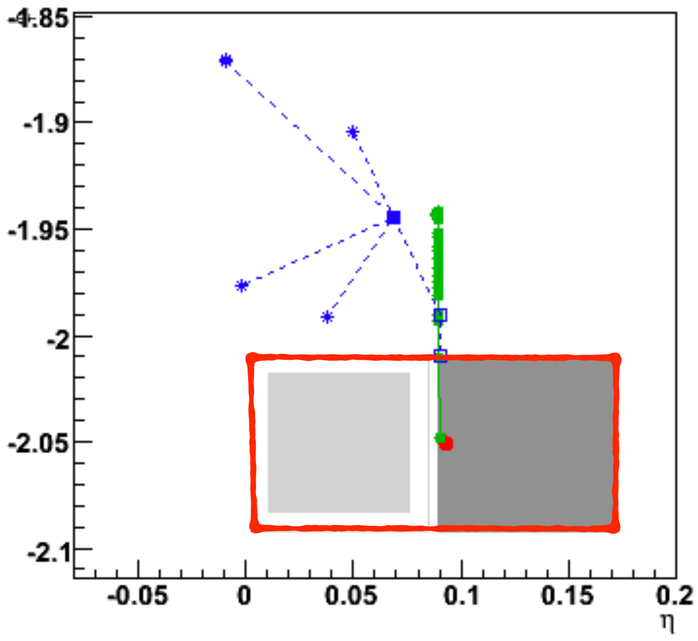
- Branching Fraction: 11.6%
- S

3 $\pi^{\pm}$, 0 π^0

- Branching Fraction: 9.8%
- three $\pi^{\pm} \approx a_1$ mass







Tau ID & Reconstru

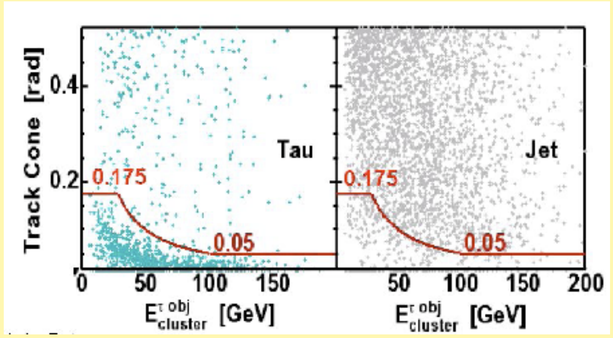
Richard Cavanaugh, Fermilab/UIC, HCP Summer School 31 July 2024

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates

Signal Cone

- Tau constituents must lie within a shrinking $\Delta R \leq 3\ GeV / pT(\tau)$



1 $\pi^{\pm}$, 0 π^0

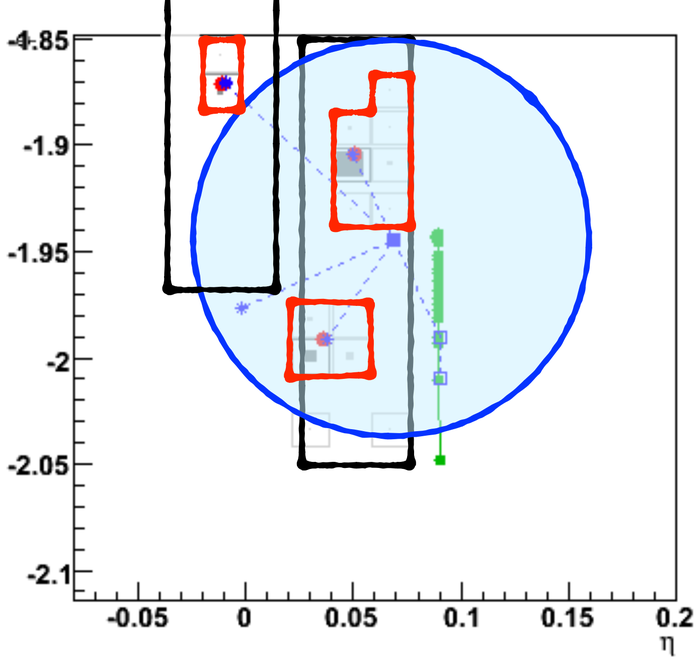
- Branching Fraction: 11.6%
- S

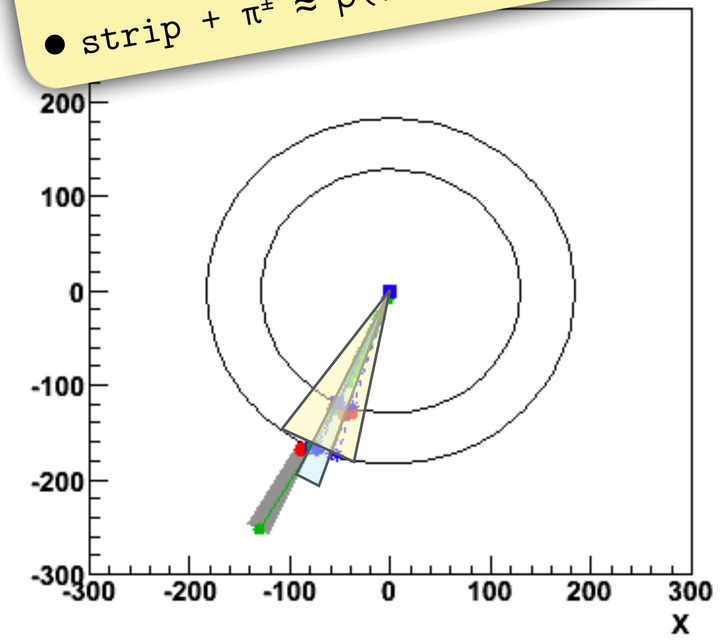
3 $\pi^{\pm}$, 0 π^0

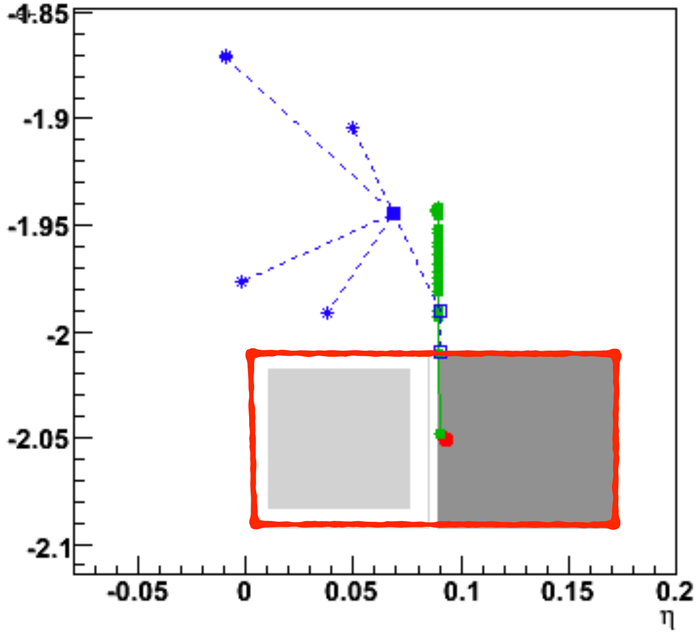
- Branching Fraction: 9.8%
- three $\pi^{\pm} \approx a_1$ mass

1 $\pi^{\pm}$, 1 π^0 (merged γγ)

- Branching Fraction: 26.0%
- single strip ≥ 1 GeV
- single $\pi^{\pm}$
- strip ≈ π^0 mass
- strip + $\pi^{\pm} \approx \rho(770)$ mass







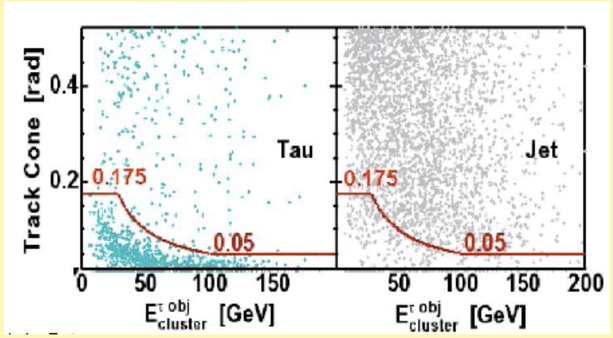
Tau ID & Reconstru

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates

Signal Cone

- Tau constituents must lie within a shrinking $\Delta R \leq 3\ GeV\ /\ pT(\tau)$



1 $\pi^{\pm}$, 0 π^0

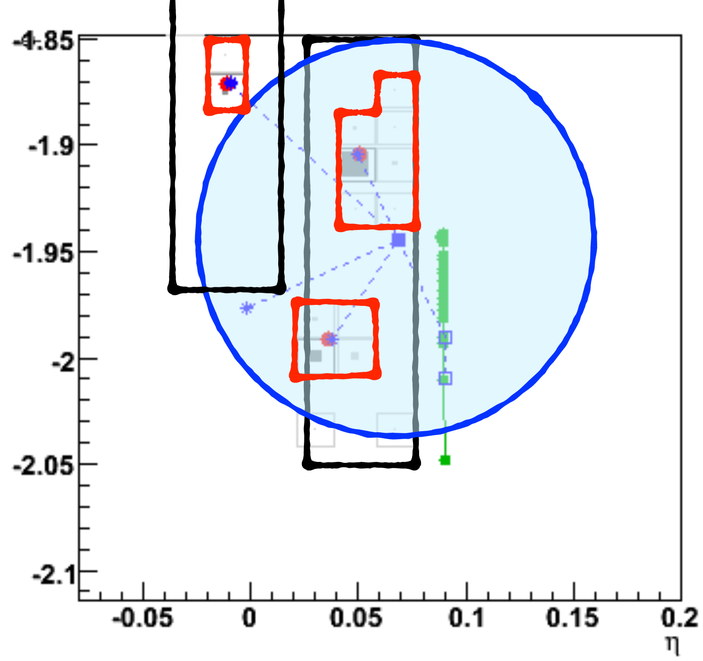
- Branching Fraction: 11.6%
- S

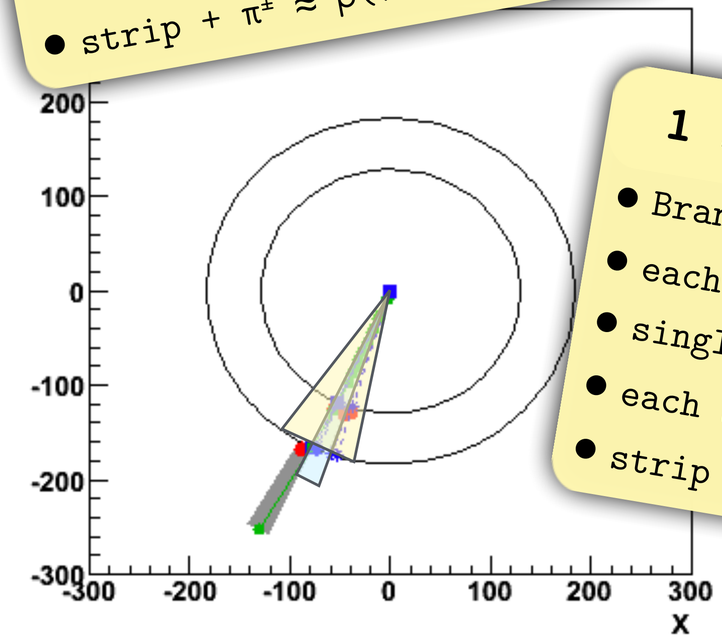
3 $\pi^{\pm}$, 0 π^0

- Branching Fraction: 9.8%
- three $\pi^{\pm} \approx a_1$ mass

1 $\pi^{\pm}$, 1 π^0 (merged γγ)

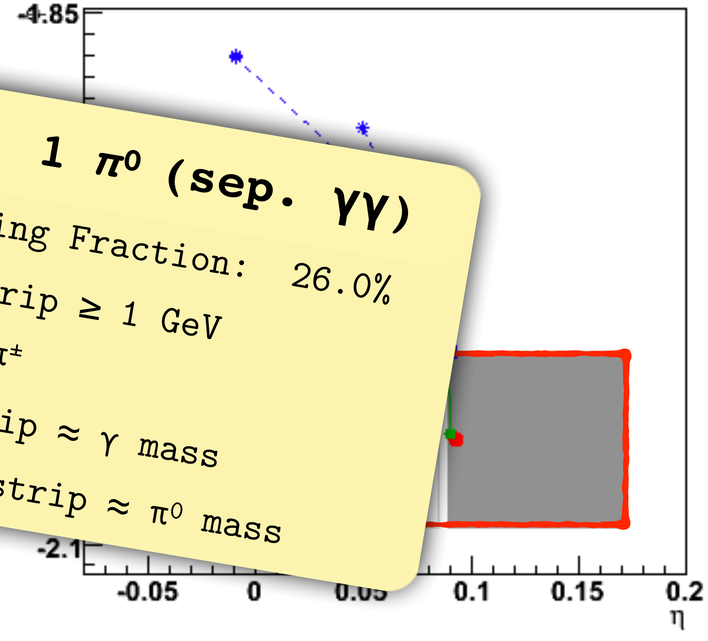
- Branching Fraction: 26.0%
- single strip ≥ 1 GeV
- single $\pi^{\pm}$
- strip ≈ π^0 mass
- strip + $\pi^{\pm}$ ≈ ρ(770) mass





1 $\pi^{\pm}$, 1 π^0 (sep. γγ)

- Branching Fraction: 26.0%
- each strip ≥ 1 GeV
- single $\pi^{\pm}$
- each strip ≈ γ mass
- strip + strip ≈ π^0 mass



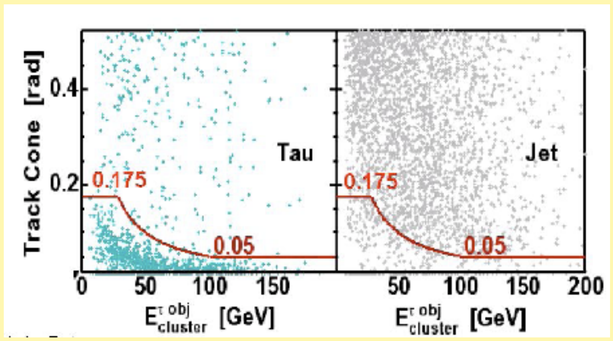
Tau ID & Reconstru

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta x \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates

Signal Cone

- Tau constituents must lie within a shrinking $\Delta R \leq 3\ GeV\ /\ pT(\tau)$



1 $\pi^{\pm}$, 0 π^0

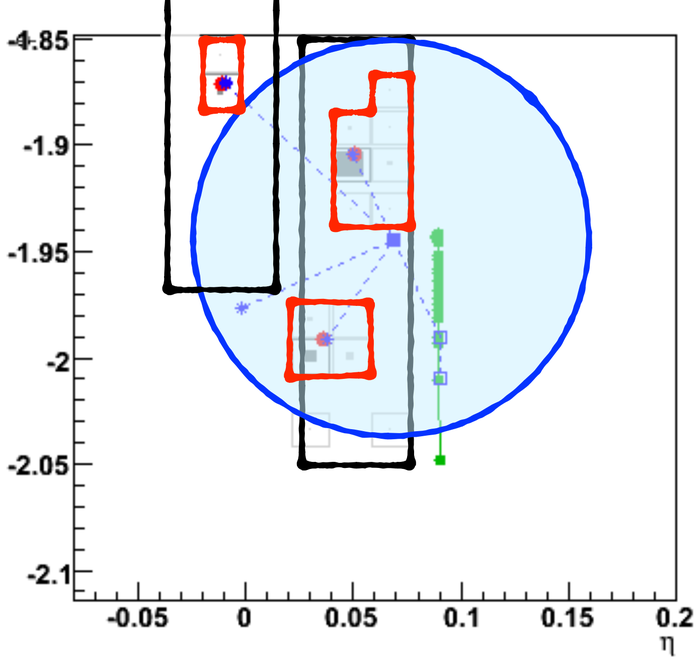
- Branching Fraction: 11.6%
- S

3 $\pi^{\pm}$, 0 π^0

- Branching Fraction: 9.8%
- three $\pi^{\pm} \approx a_1$ mass

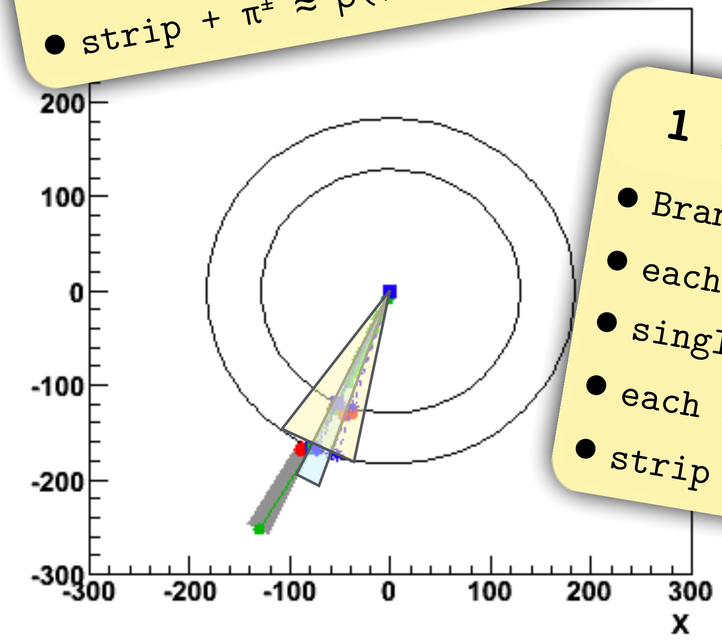
1 $\pi^{\pm}$, 1 π^0 (merged γγ)

- Branching Fraction: 26.0%
- single strip ≥ 1 GeV
- single $\pi^{\pm}$
- strip ≈ π^0 mass
- strip + $\pi^{\pm} \approx \rho(770)$ mass



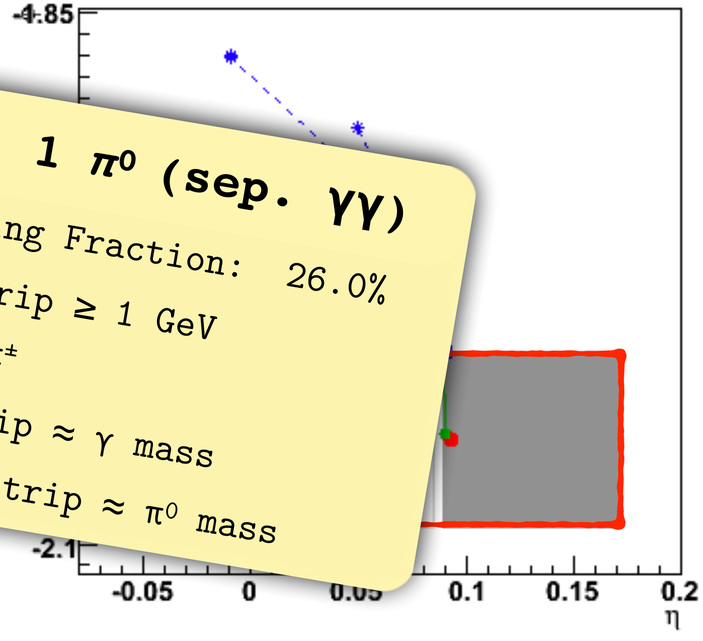
Could also do ID modes

1 $\pi^{\pm}$, 2 π^0's or 3 $\pi^{\pm}$, 1 π^0



1 $\pi^{\pm}$, 1 π^0 (sep. γγ)

- Branching Fraction: 26.0%
- each strip ≥ 1 GeV
- single $\pi^{\pm}$
- each strip ≈ γ mass
- strip + strip ≈ π^0 mass



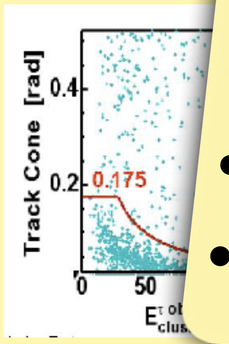
Tau ID & Reconstru

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta \times \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates

Sig

- Tau consti
  within a
  ΔR ≤ 3 Ge



Isolation

- Isolation cone of ΔR = 0.5
- summed pT of all particles, except constituents of tau-candidate
- loose: no $\pi^{\pm}$ with pT > 1.0 GeV; no γ with pT > 1.5 GeV
- medium: no $\pi^{\pm}$ with pT > 0.8 GeV; no γ with pT > 0.8 GeV
- tight: no $\pi^{\pm}$ with pT > 0.5 GeV; no γ with pT > 0.5 GeV

1 $\pi^{\pm}$, 0 π^0

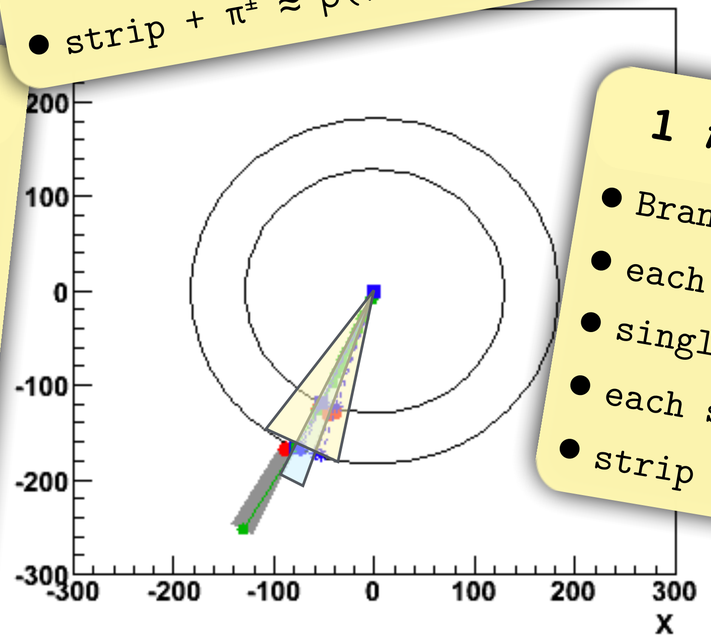
- Branching Fraction: 11.6%
- S

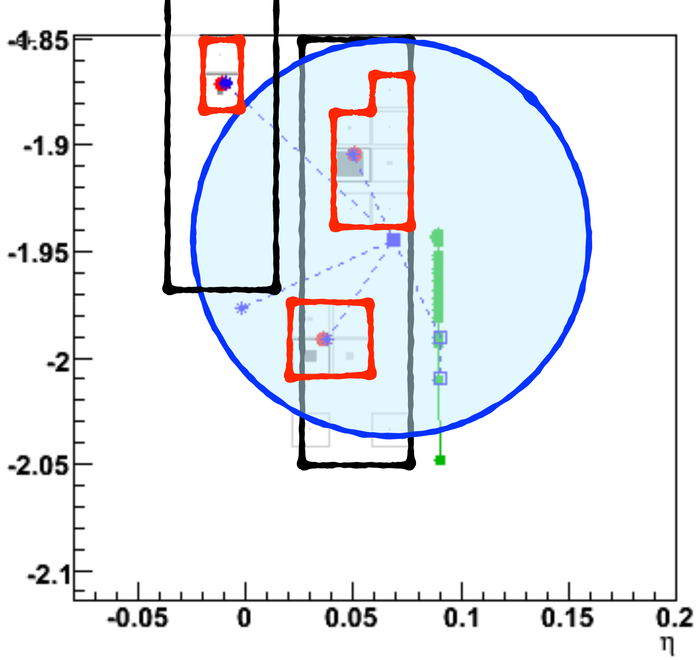
3 $\pi^{\pm}$, 0 π^0

- Branching Fraction: 9.8%
- three $\pi^{\pm} \approx a_1$ mass

1 $\pi^{\pm}$, 1 π^0 (merged γγ)

- Branching Fraction: 26.0%
- single strip ≥ 1 GeV
- single $\pi^{\pm}$
- strip ≈ π^0 mass
- strip + $\pi^{\pm} \approx \rho(770)$ mass



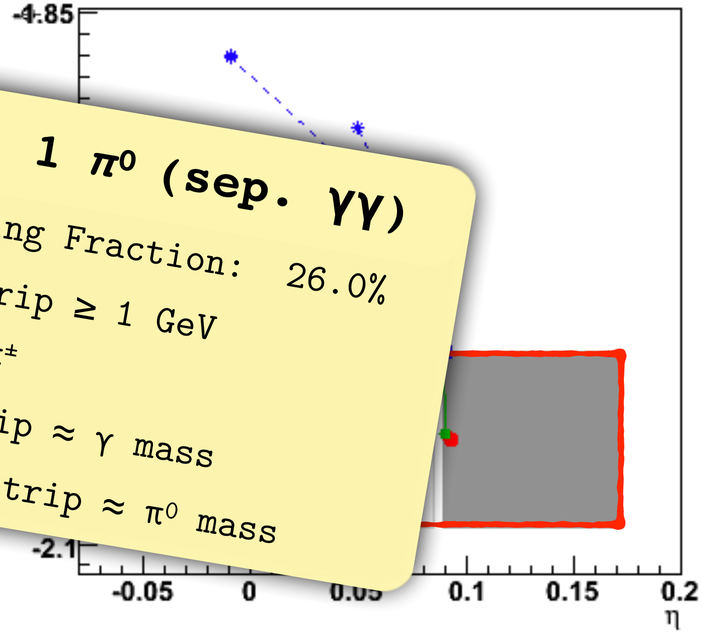


Could also do ID modes

1 $\pi^{\pm}$, 2 π^0's or 3 $\pi^{\pm}$, 1 π^0

1 $\pi^{\pm}$, 1 π^0 (sep. γγ)

- Branching Fraction: 26.0%
- each strip ≥ 1 GeV
- single $\pi^{\pm}$
- each strip ≈ γ mass
- strip + strip ≈ π^0 mass



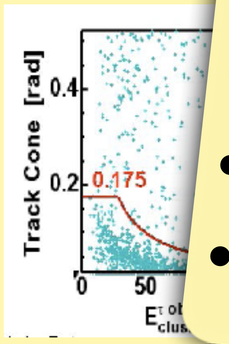
Tau ID & Reconstru

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with highest p_T γ or $e^{\pm}$ in jet
- Cluster all γ's or e's into strips
  - $\Delta\eta \times \Delta\varphi$ = 0.05x0.2
  - to capture all conversions
- Combine with $\pi^{\pm}$'s to form tau candidates

Sig

- Tau consti
  within a
  ΔR ≤ 3 Ge



Isolation

- Isolation cone of ΔR = 0.5
- summed pT of all particles, except constituents of tau-candidate
- loose: no $\pi^{\pm}$ with pT > 1.0 GeV; no γ with pT > 1.5 GeV
- medium: no $\pi^{\pm}$ with pT > 0.8 GeV; no γ with pT > 0.8 GeV
- tight: no $\pi^{\pm}$ with pT > 0.5 GeV; no γ with pT > 0.5 GeV

1 $\pi^{\pm}$, 0 π^0

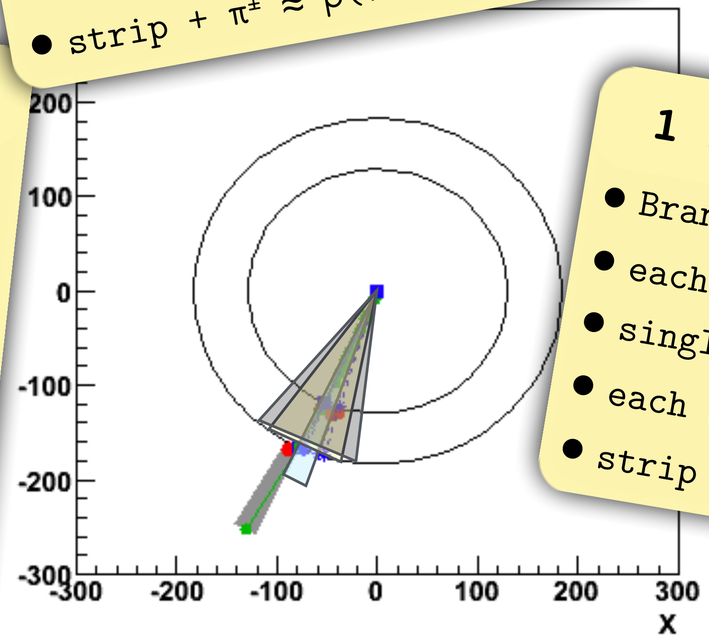
- Branching Fraction: 11.6%
- S

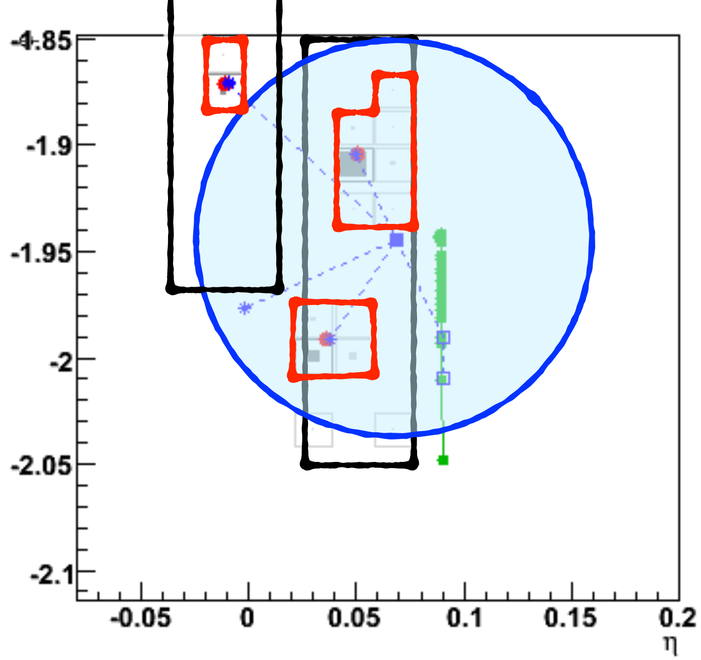
3 $\pi^{\pm}$, 0 π^0

- Branching Fraction: 9.8%
- three $\pi^{\pm} \approx a_1$ mass

1 $\pi^{\pm}$, 1 π^0 (merged γγ)

- Branching Fraction: 26.0%
- single strip ≥ 1 GeV
- single $\pi^{\pm}$
- strip ≈ π^0 mass
- strip + $\pi^{\pm} \approx \rho(770)$ mass



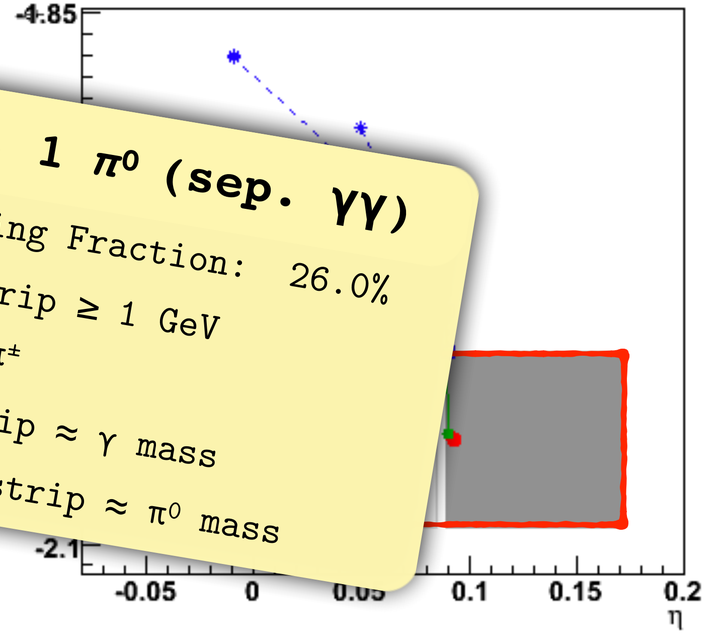


Could also do ID modes

1 $\pi^{\pm}$, 2 π^0's or 3 $\pi^{\pm}$, 1 π^0

1 $\pi^{\pm}$, 1 π^0 (sep. γγ)

- Branching Fraction: 26.0%
- each strip ≥ 1 GeV
- single $\pi^{\pm}$
- each strip ≈ γ mass
- strip + strip ≈ π^0 mass



Tau ID & Reconstruc

Basic Idea

- Seed with R=0.4 anti-kt jet
- Start with jet
- Cluster a strips
  - $\Delta\eta \times \Delta\varphi$ =
  - to capt
- Combine candidate

1 $\pi^{\pm}$, 0 π^0

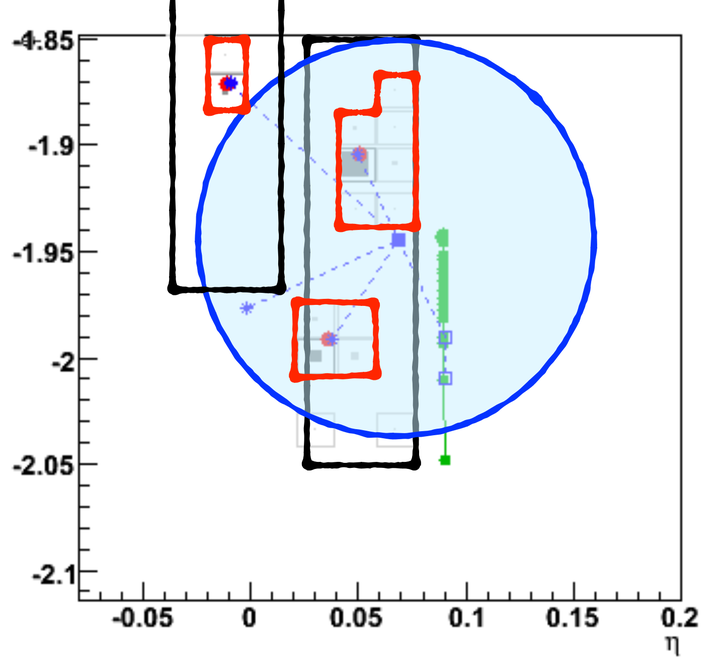
- Branching Fraction: 11.6%
- S

3 $\pi^{\pm}$, 0 π^0

- Branching Fraction: 9.8%
- strip ≈ π^0 mass
- strip + $\pi^{\pm}$ ≈ ρ(770) mass

Typical Reconstruction Efficiencies

Reconstructed	Generated		
	$\tau^- \to h^- \nu_\tau$	$\tau^- \to h^- \geq 1\pi^0 \nu_\tau$	$\tau^- \to h^- h^+ h^- \nu_\tau$
$\tau^- \to h^- \nu_\tau$	0.89	0.16	0.01
$\tau^- \to h^- \geq 1\pi^0 \nu_\tau$	0.11	0.83	0.02
$\tau^- \to h^- h^+ h^- \nu_\tau$	0.00	0.01	0.97

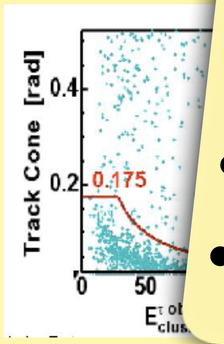


Could also do ID modes

1 $\pi^{\pm}$, 2 π^0's or 3 $\pi^{\pm}$, 1 π^0

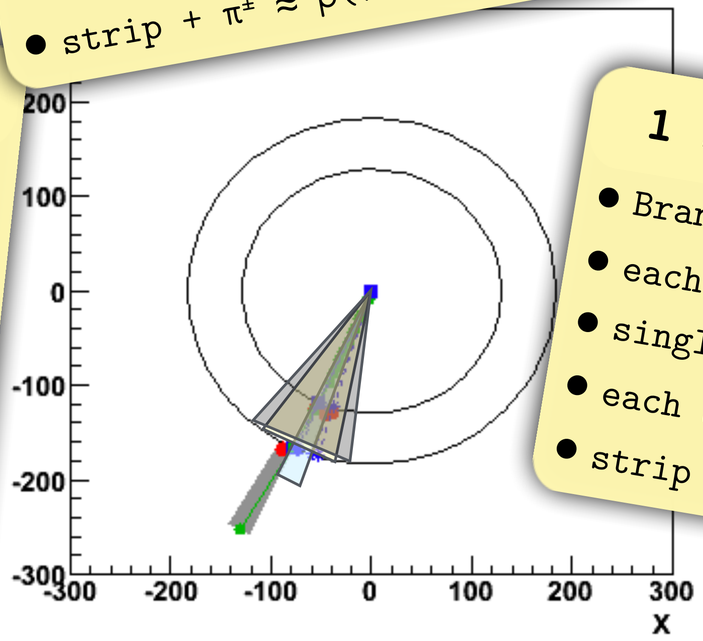
Sig

- Tau consti within a ΔR ≤ 3 Ge



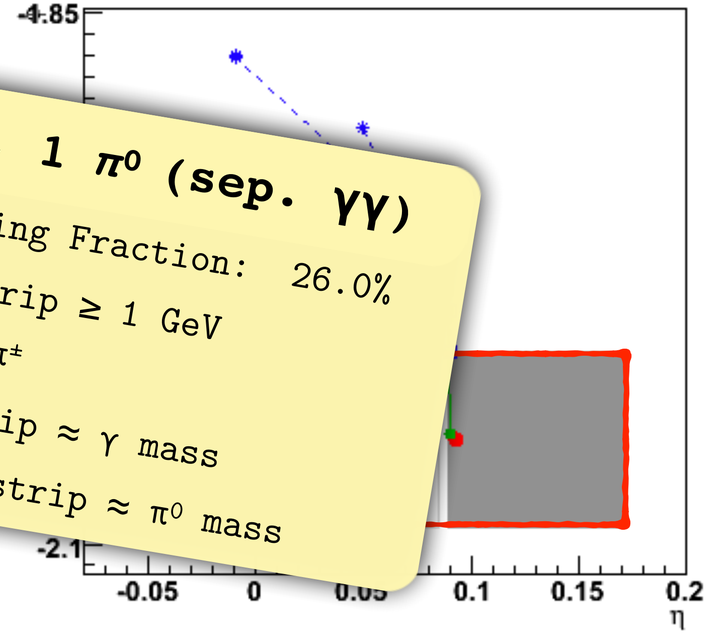
Isolation

- Isolation cone of ΔR = 0.5
- summed pT of all particles, except constituents of tau-candidate
- loose: no $\pi^{\pm}$ with pT > 1.0 GeV; no γ with pT > 1.5 GeV
- medium: no $\pi^{\pm}$ with pT > 0.8 GeV; no γ with pT > 0.8 GeV
- tight: no $\pi^{\pm}$ with pT > 0.5 GeV; no γ with pT > 0.5 GeV

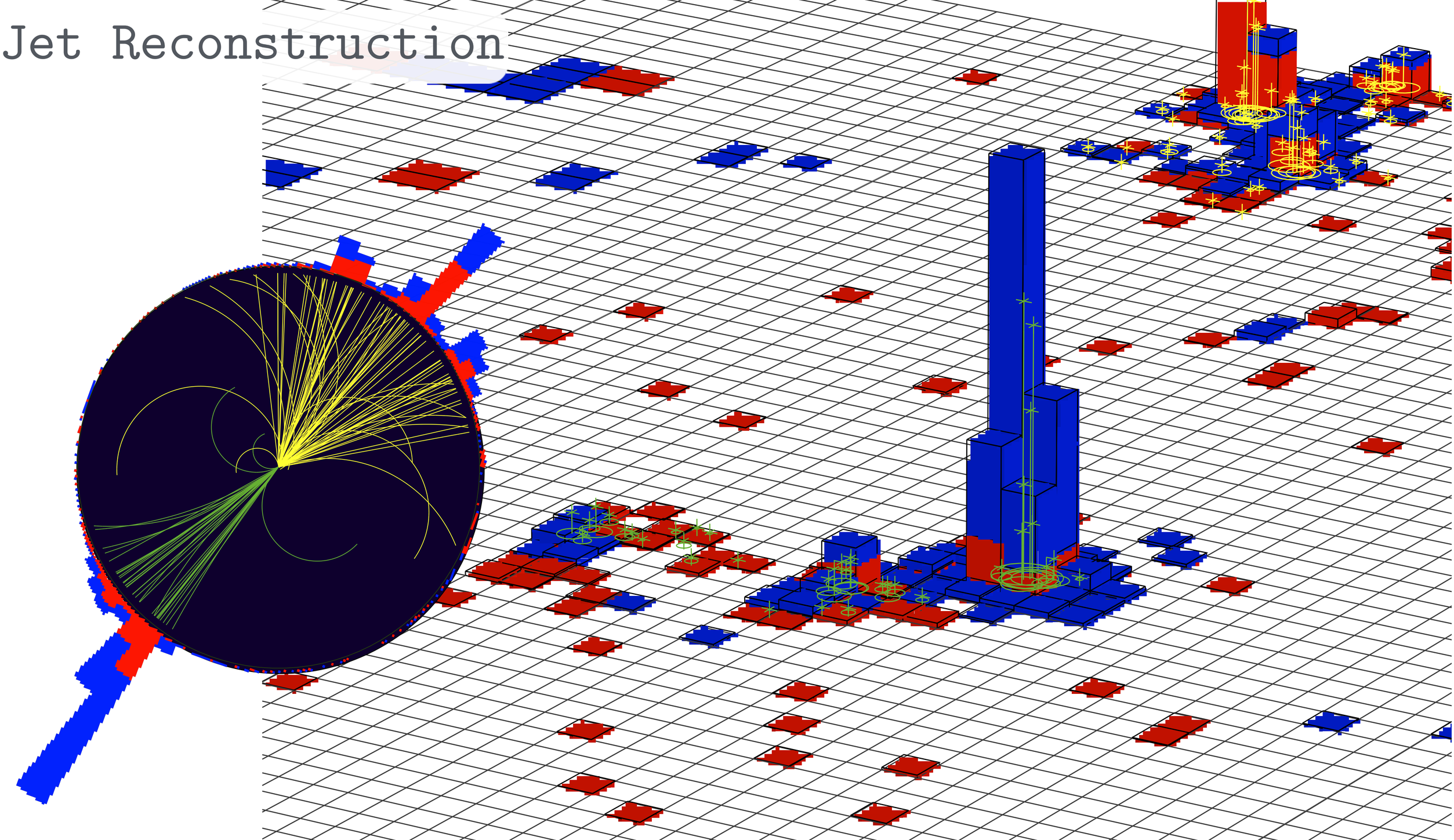


1 $\pi^{\pm}$, 1 π^0 (sep. γγ)

- Branching Fraction: 26.0%
- each strip ≥ 1 GeV
- single $\pi^{\pm}$
- each strip ≈ γ mass
- strip + strip ≈ π^0 mass



Jet Reconstruction



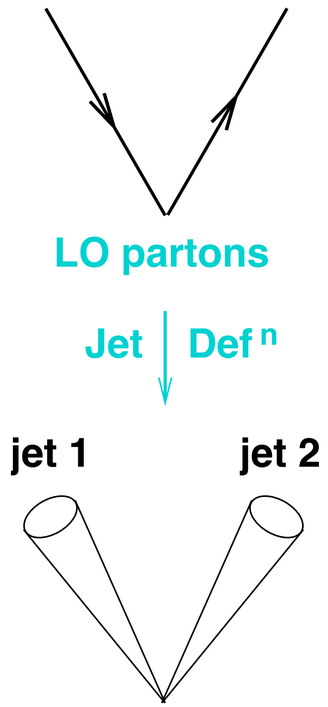
Richard Cavanaugh, Fermilab/UIC, HCP Summer School 31 July 2024

Jet Reconstruction

Richard Cavanaugh, Fermilab/UIC, HCP Summer School 31 July 2024

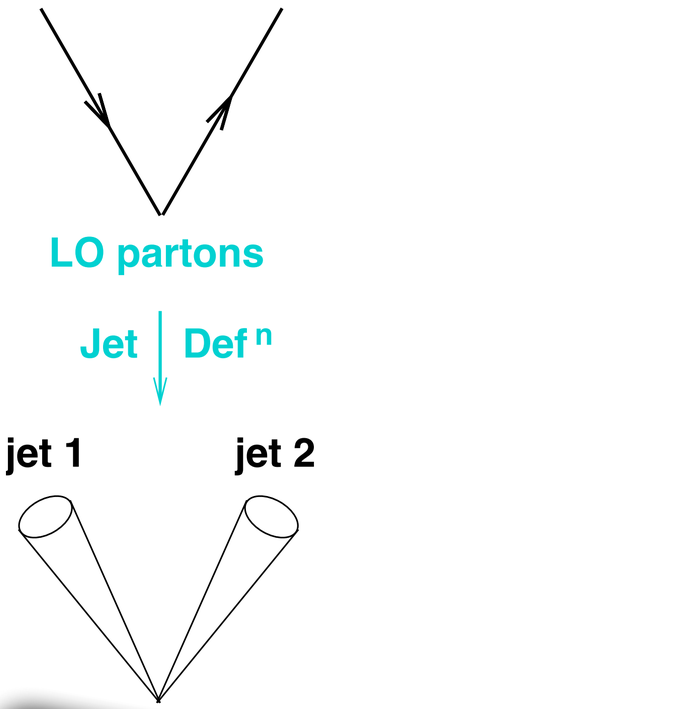
Jet Reconstruction

Diagram from Gavin Salam



Jet Reconstruction

Diagram from Gavin Salam

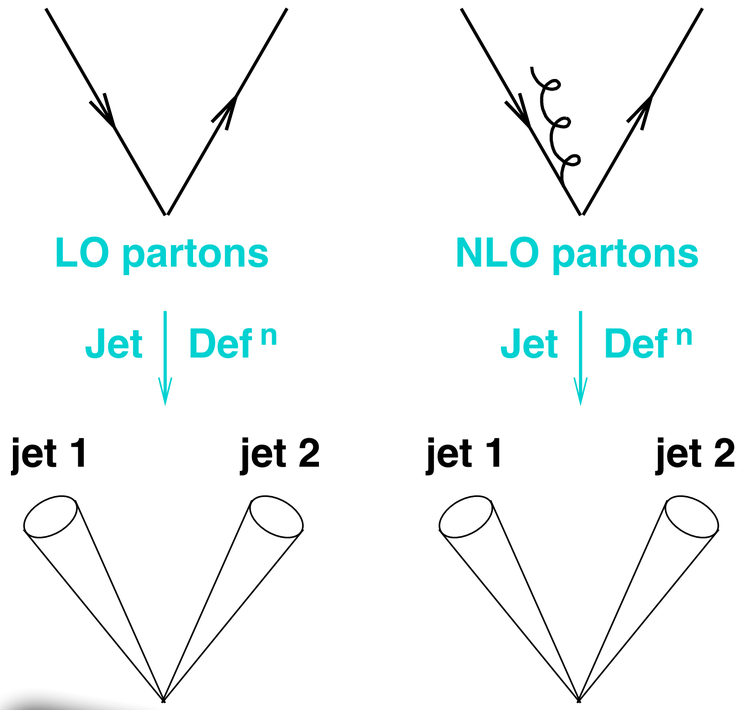


Coloured partons: quarks & gluons

- Central concepts of QCD
- Not directly observable (confinement)

Jet Reconstruction

Diagram from Gavin Salam

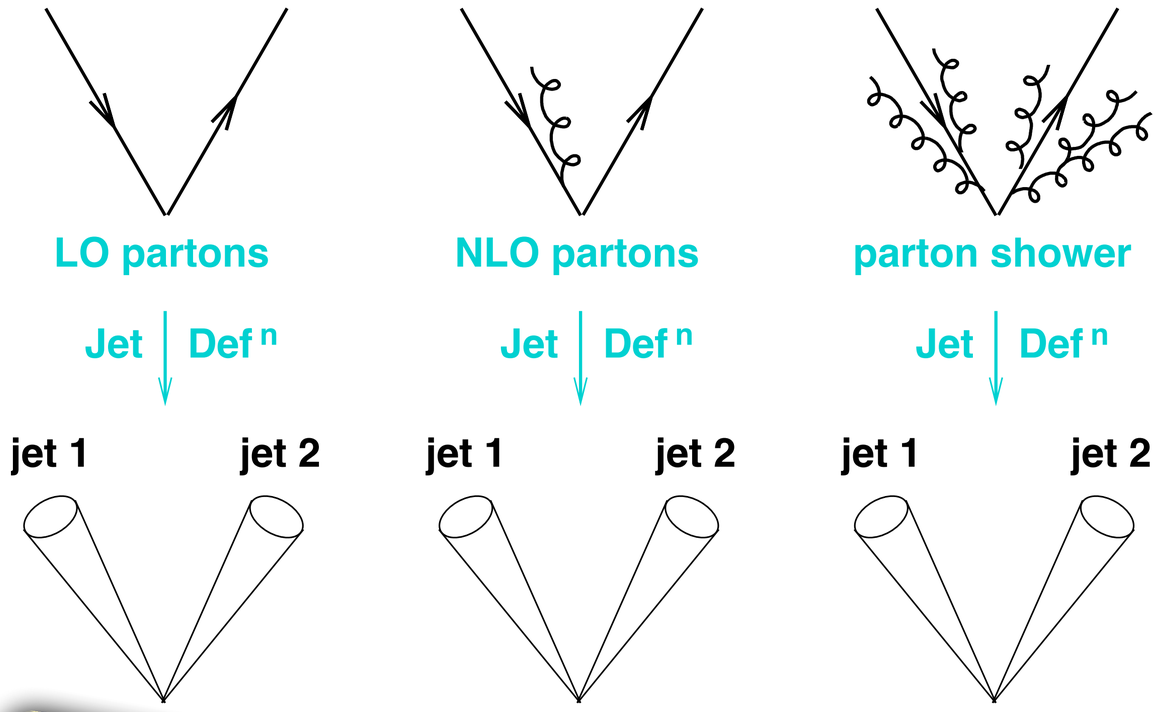


Coloured partons: quarks & gluons

- Central concepts of QCD
- Not directly observable (confinement)

Jet Reconstruction

Diagram from Gavin Salam

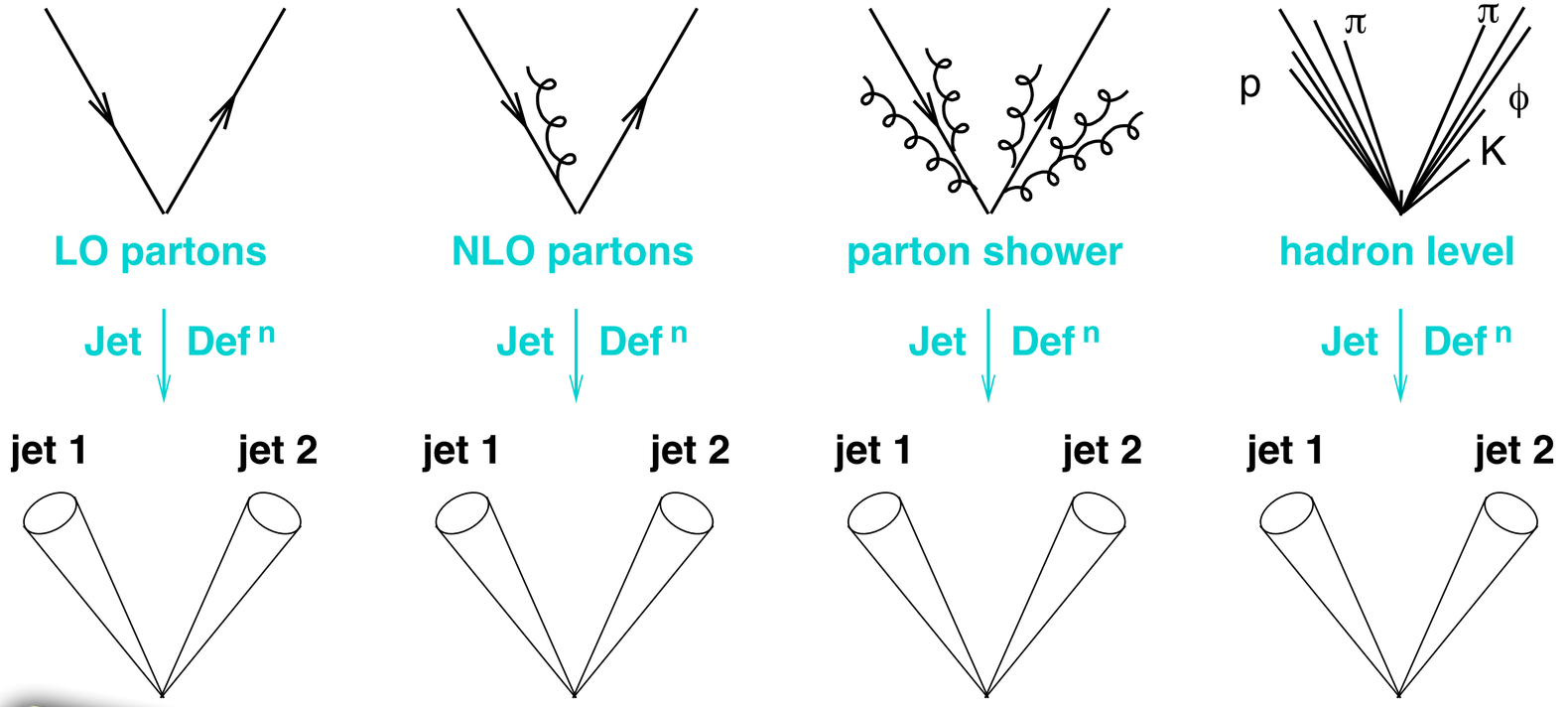


Coloured partons: quarks & gluons

- Central concepts of QCD
- Not directly observable (confinement)

Jet Reconstruction

Diagram from Gavin Salam

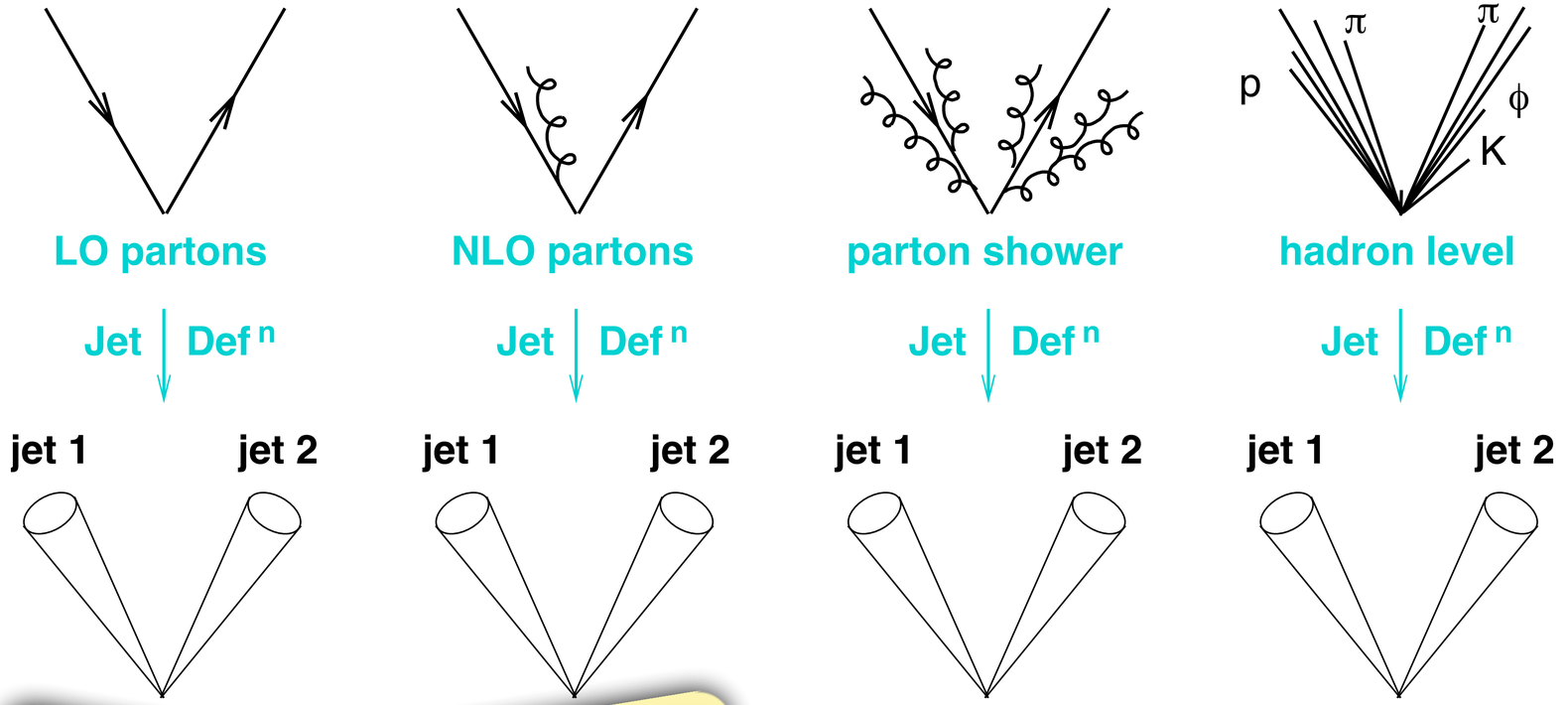


Coloured partons: quarks & gluons

- Central concepts of QCD
- Not directly observable (confinement)

Jet Reconstruction

Diagram from Gavin Salam

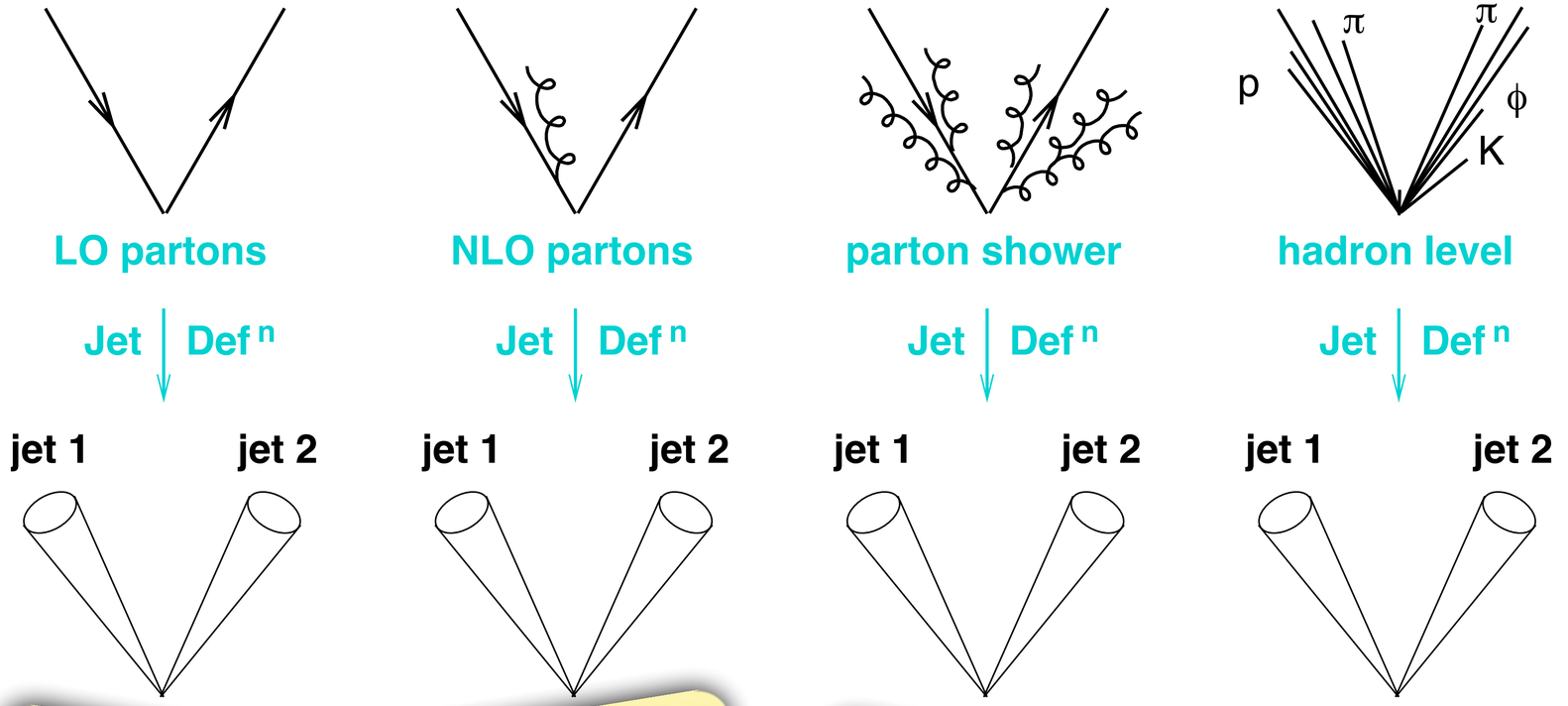


Jets

- signatures of quarks & gluons from hard scattering processes
- result from fragmentation (perturbative) & hadronisation (non-perturbative) of original partons

Jet Reconstruction

Diagram from Gavin Salam



Jets

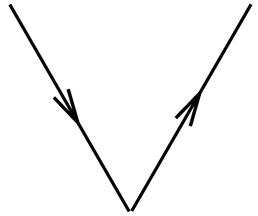
- signatures of quarks & gluons from hard scattering processes
- result from fragmentation (perturbative) & hadronisation (non-perturbative) of original partons

Jets

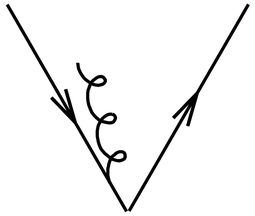
- are collimated bunches (sprays) of particles
- reflect kinematics & topology of partons
- really are what the algorithm defines them to be!

Jet Reconstruction

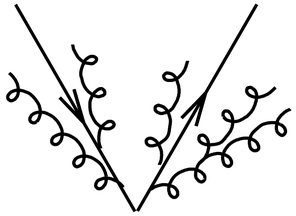
Diagram from Gavin Salam



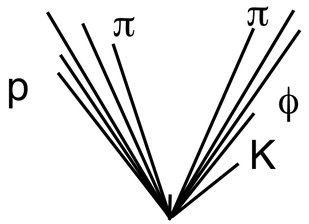
LO partons



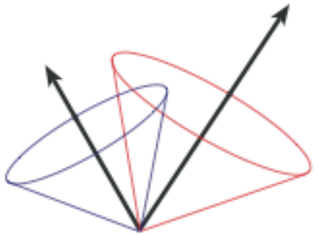
NLO partons

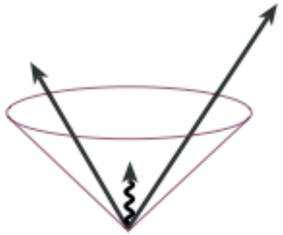


parton shower

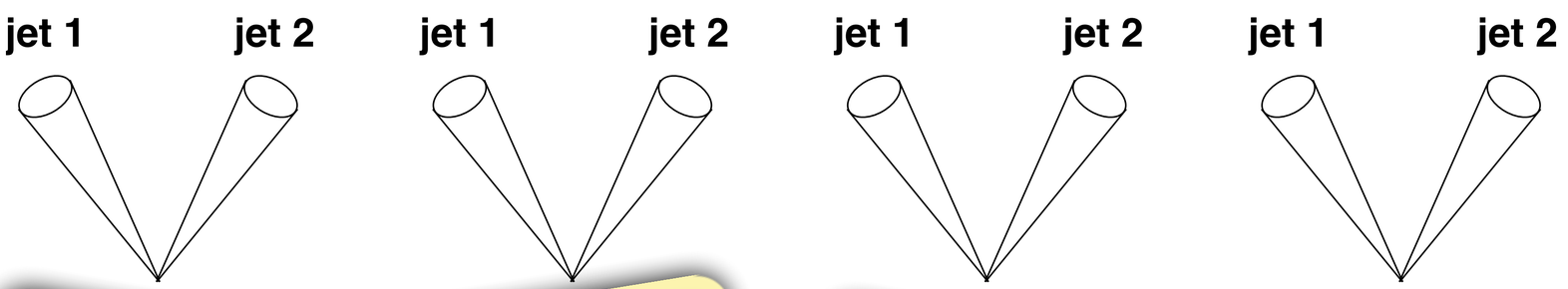


hadron level





Jet | Def^{n} Jet | Def^{n} Jet | Def^{n} Jet | Def^{n}



Jets

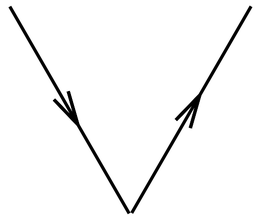
- signatures of quarks & gluons from hard scattering processes
- result from fragmentation (perturbative) & hadronisation (non-perturbative) of original partons

Jets

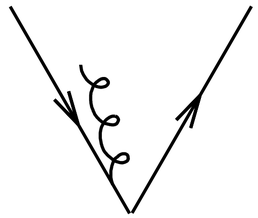
- are collimated bunches (sprays) of particles
- reflect kinematics & topology of partons
- really are what the algorithm defines them to be!

Jet Reconstruction

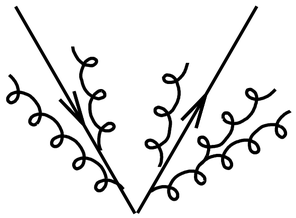
Diagram from Gavin Salam



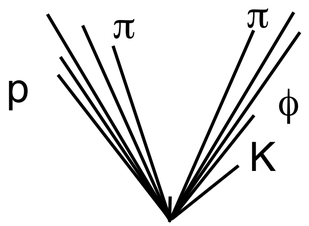
LO partons



NLO partons

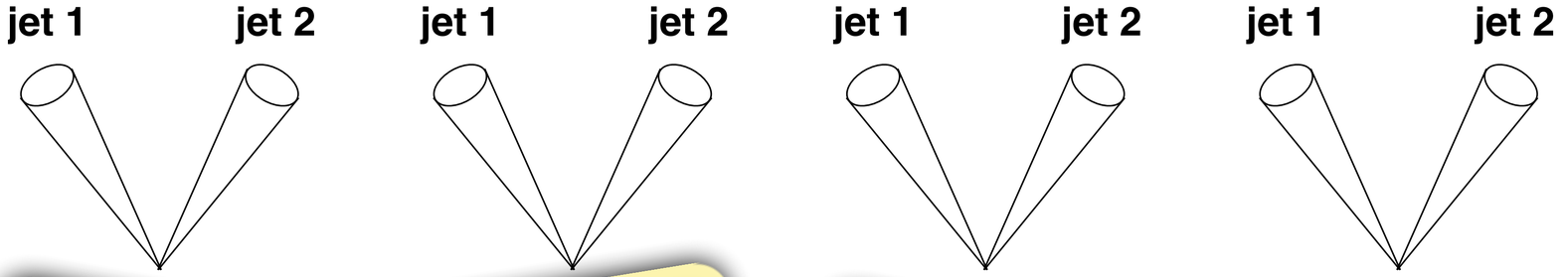


parton shower



hadron level

Jet | Def^n

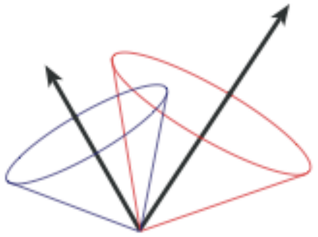


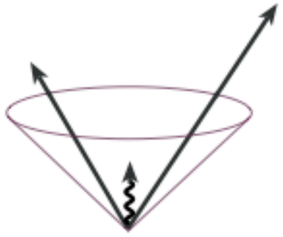
Jets

- signatures of quarks & gluons from hard scattering processes
- result from fragmentation (perturbative) & hadronisation (non-perturbative) of original partons

Jets

- are collimated bunches (sprays) of particles
- reflect kinematics & topology of partons
- really are what the algorithm defines them to be!



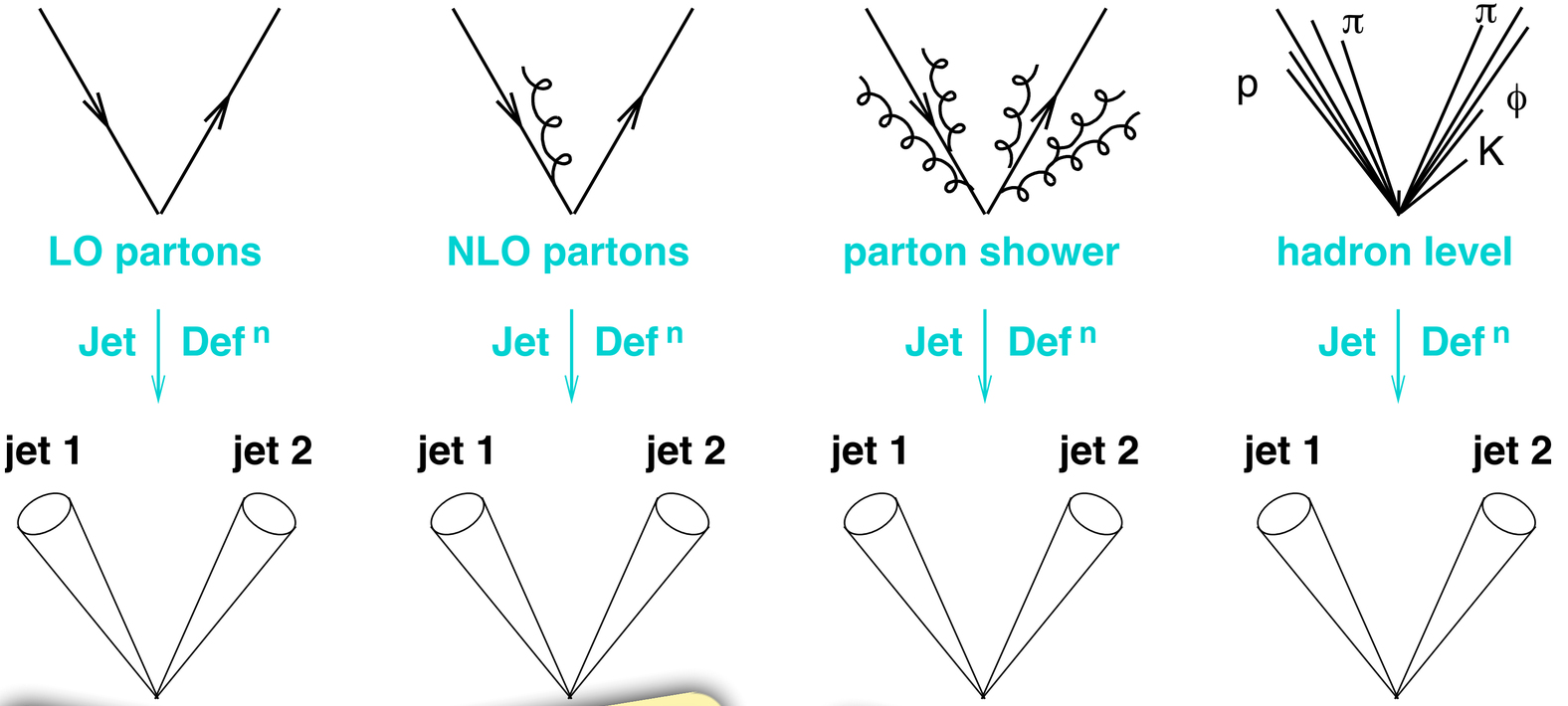


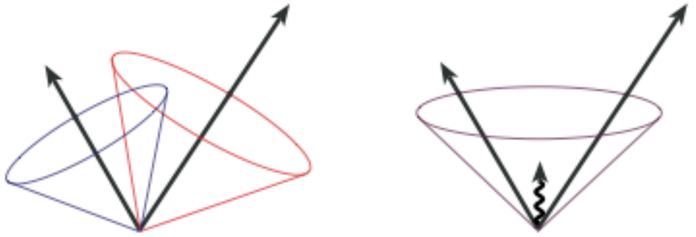
Infrared safety

- stable against adding soft parton emission

Jet Reconstruction

Diagram from Gavin Salam





Jets

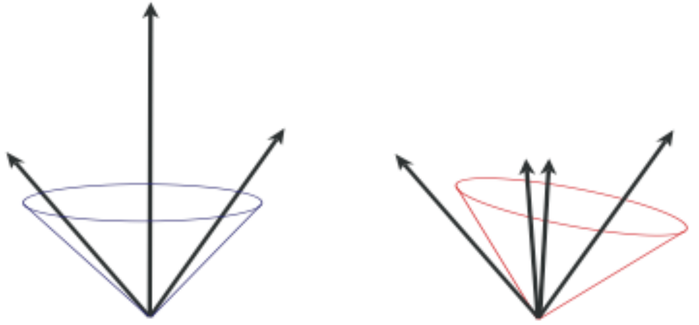
- signatures of quarks & gluons from hard scattering processes
- result from fragmentation (perturbative) & hadronisation (non-perturbative) of original partons

Jets

- are collimated bunches (sprays) of particles
- reflect kinematics & topology of partons
- really are what the algorithm defines them to be!

Infrared safety

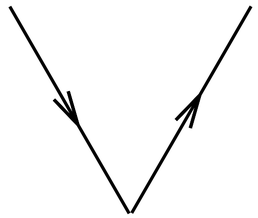
- stable against adding soft parton emission



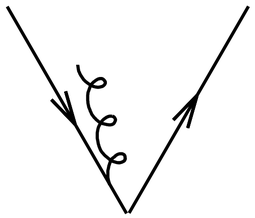
Richard Cavanaugh, Fermilab/UIC, HCP Summer School 31 July 2024

Jet Reconstruction

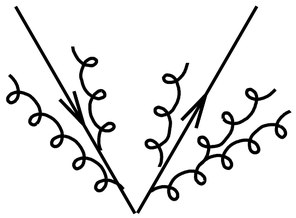
Diagram from Gavin Salam



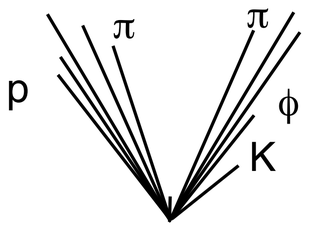
LO partons



NLO partons

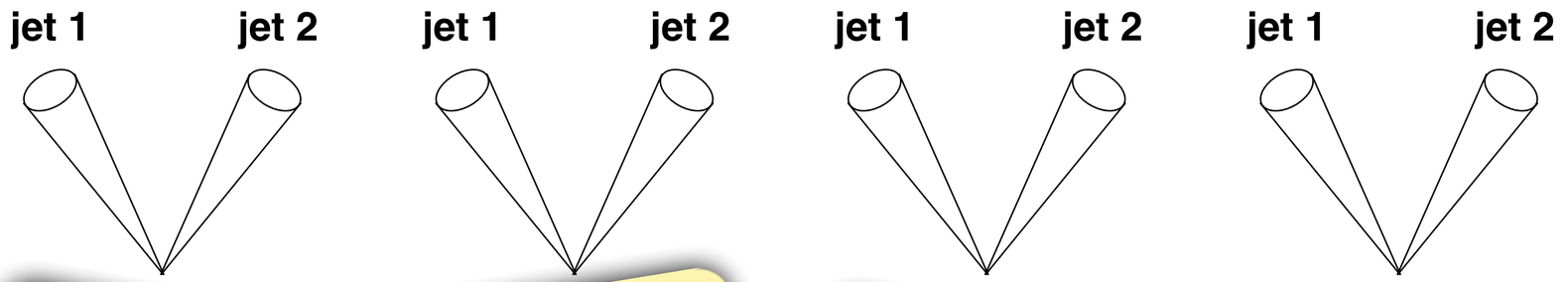


parton shower



hadron level

Jet | Def^n

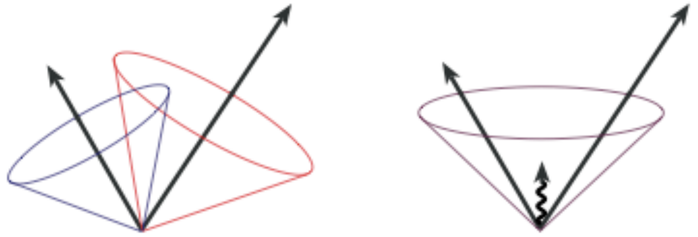


Jets

- signatures of quarks & gluons from hard scattering processes
- result from fragmentation (perturbative) & hadronisation (non-perturbative) of original partons

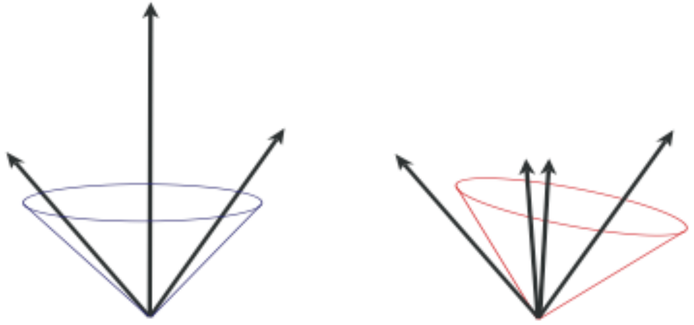
Jets

- are collimated bunches (sprays) of particles
- reflect kinematics & topology of partons
- really are what the algorithm defines them to be!



Infrared safety

- stable against adding soft parton emission



Collinear safety

- stable against replacing any parton with collinear pair

Jet Reconstruction

Richard Cavanaugh, Fermilab/UIC, HCP Summer School 31 July 2024

Jet Reconstruction

Iterative Cone Algorithms

- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Jet Reconstruction

Iterative Cone Algorithms

- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Sequential Clustering Algorithms

- Historically used (before LHC) at lepton colliders
- Now standard at LHC

Jet Reconstruction

Iterative Cone Algorithms

- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Sequential Clustering Algorithms

- Historically used (before LHC) at lepton colliders
- Now standard at LHC

Experimental Input

- Fundamentally, just a collection of 4-momenta, not more not less
- Historically
  - Reconstructed calo cells/clusters
  - Reconstructed tracks
- LHC (& LEP) Era(s)
  - Reconstructed particles $\{e^{\pm}, \gamma, \mu^{\pm}, h^{\pm}, h^{0}\}$

Jet Reconstruction

Iterative Cone Algorithms

- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Jet Reconstruction

Iterative Cone Algorithms

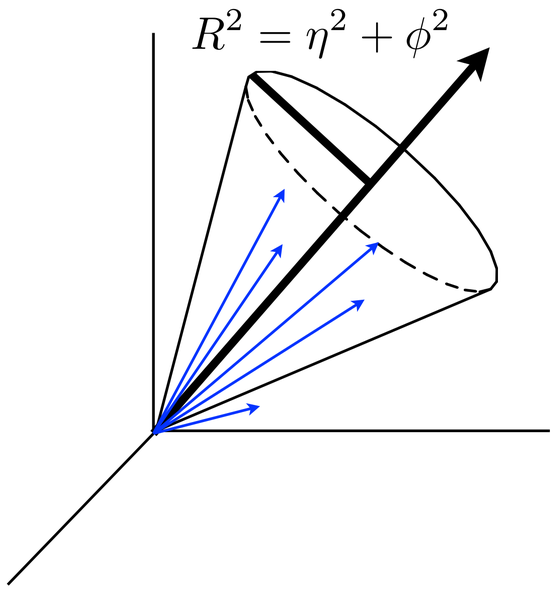
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges



Jet Reconstruction

Iterative Cone Algorithms

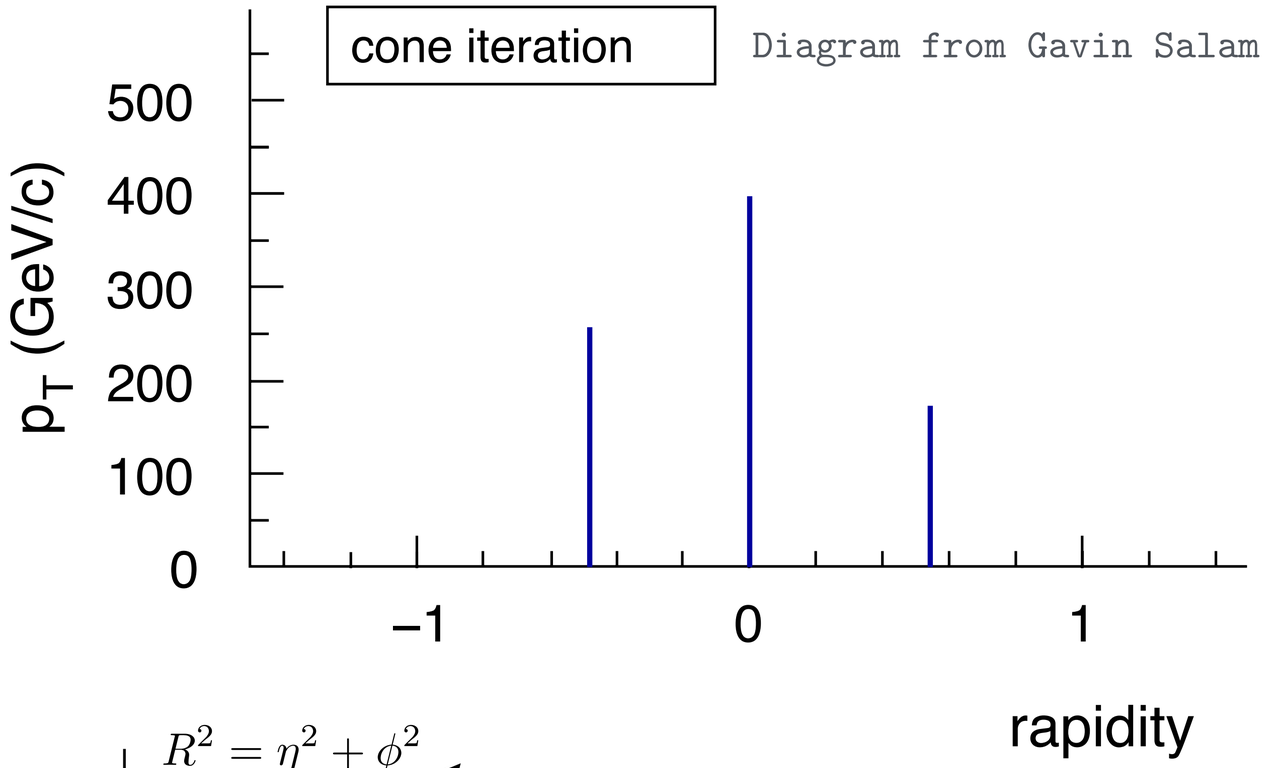
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

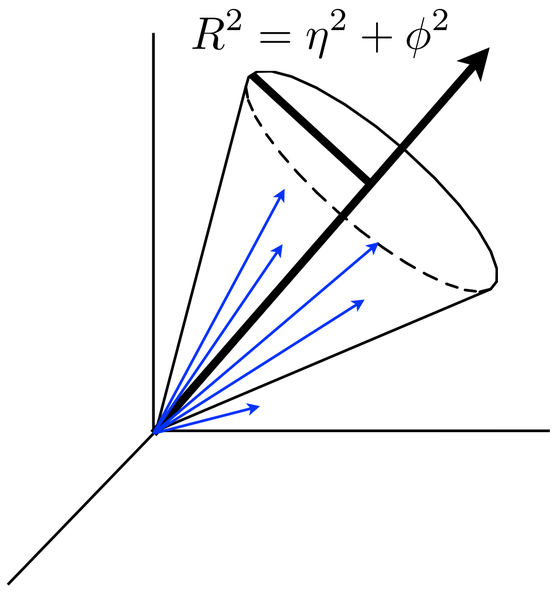
Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges





Jet Reconstruction

Iterative Cone Algorithms

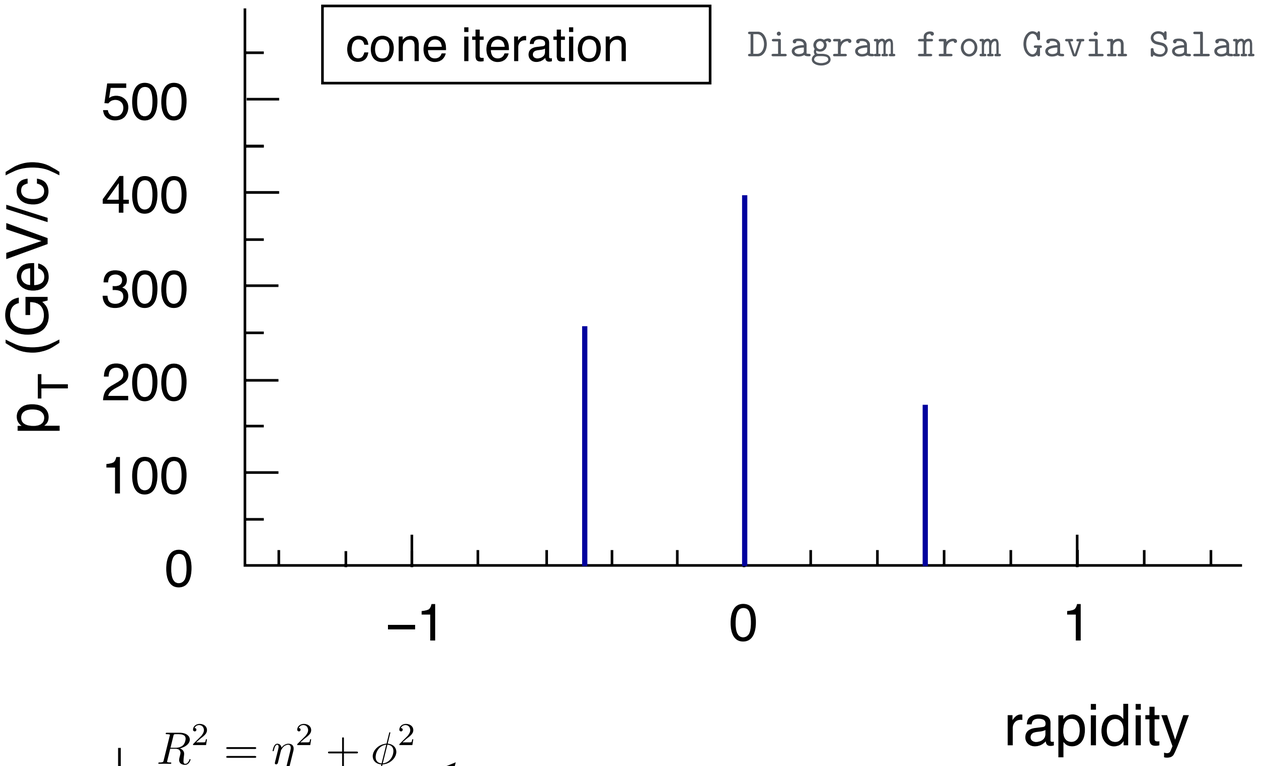
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

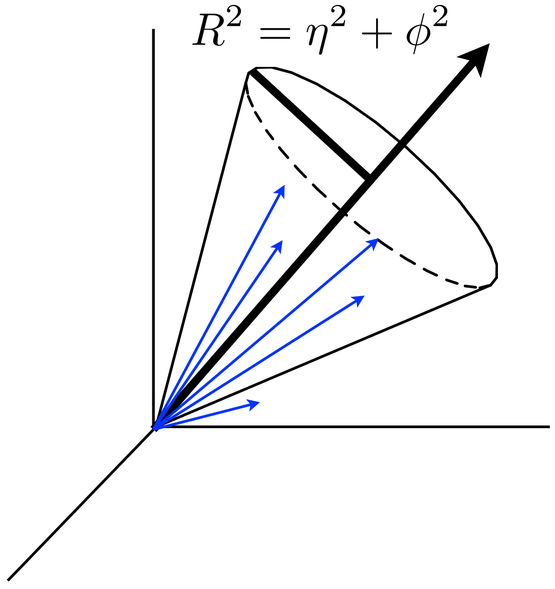
Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges





Scenario 1:

Jet Reconstruction

Iterative Cone Algorithms

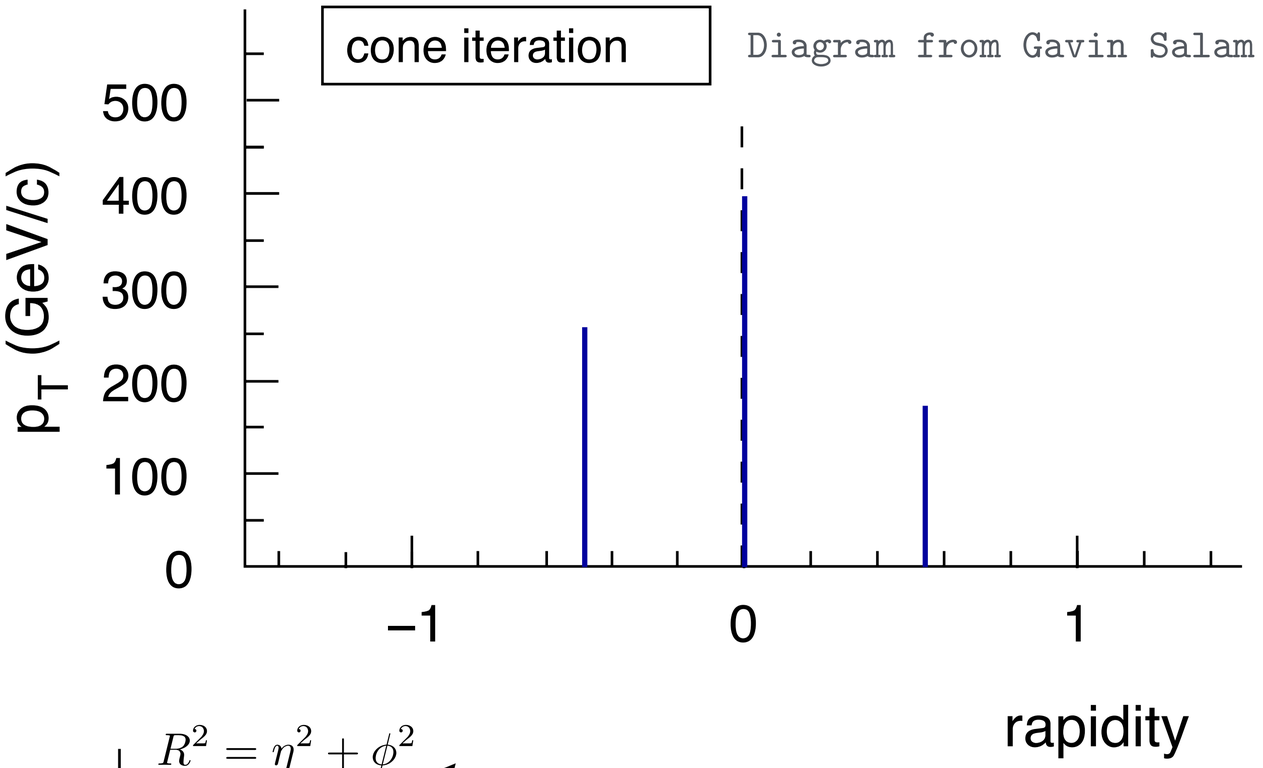
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

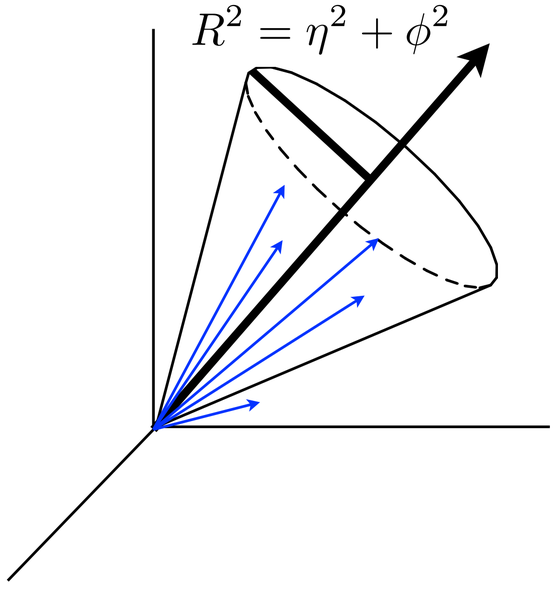
Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges





Scenario 1:

Jet Reconstruction

Iterative Cone Algorithms

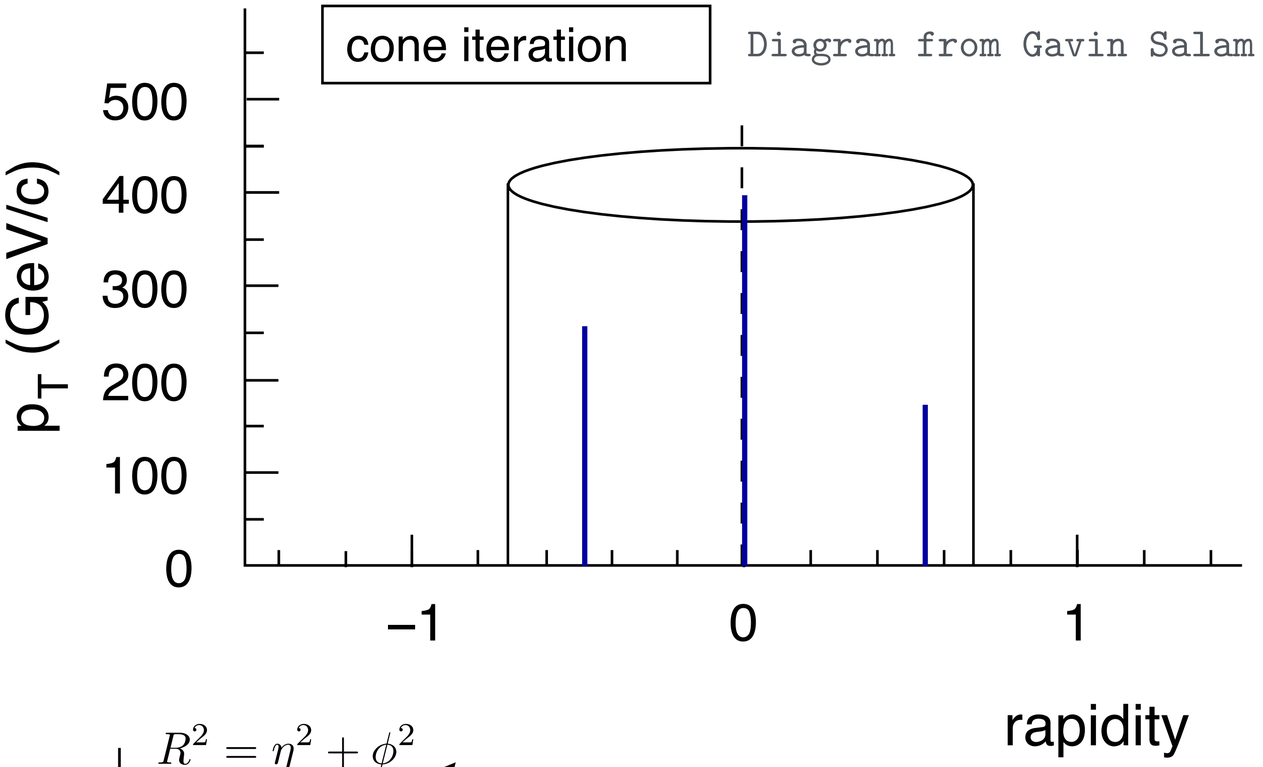
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

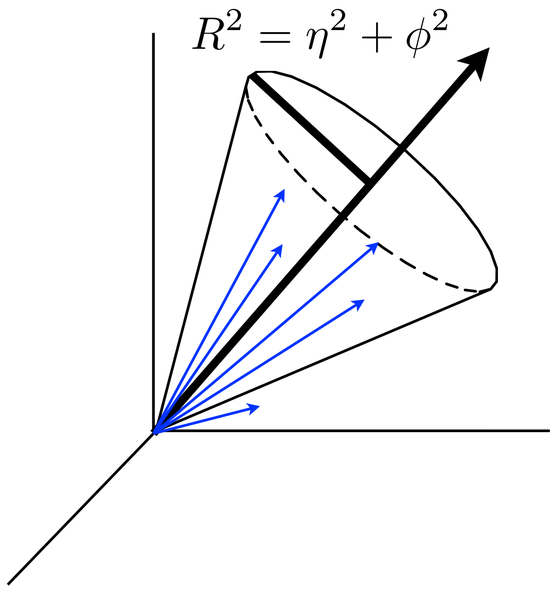
Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges





Scenario 1:

Jet Reconstruction

Iterative Cone Algorithms

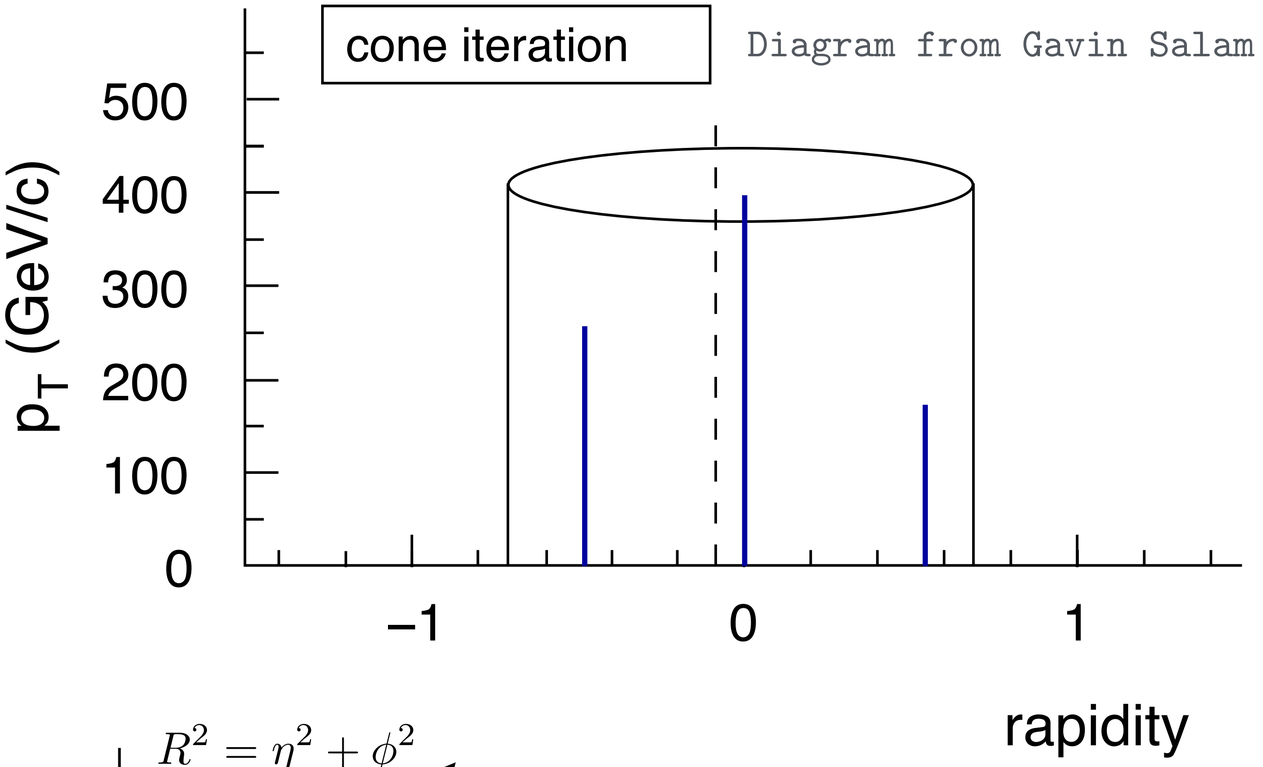
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

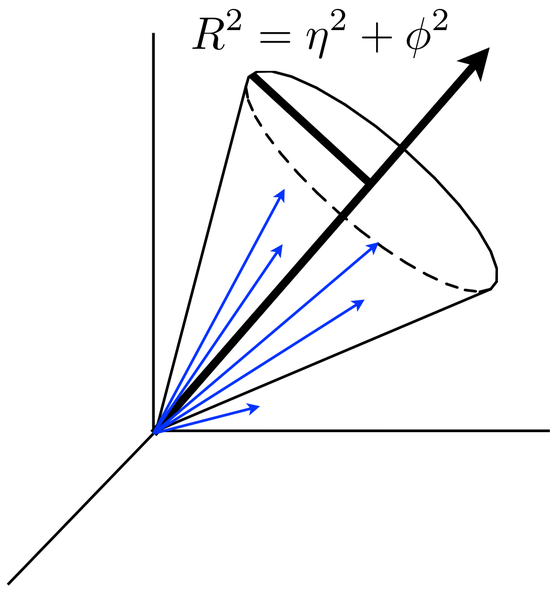
Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges





Scenario 1:

Jet Reconstruction

Iterative Cone Algorithms

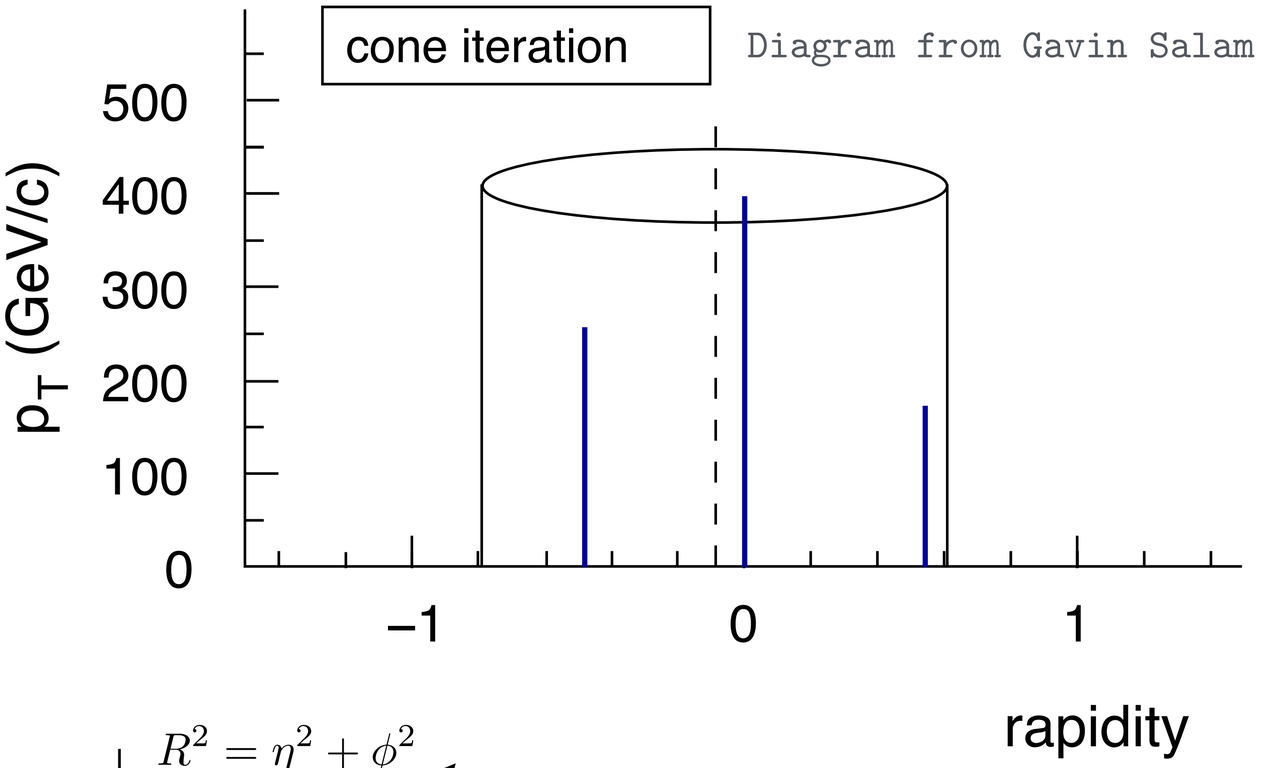
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

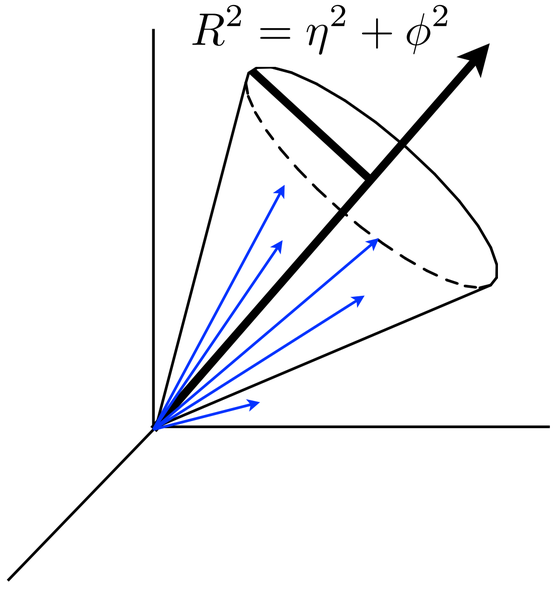
Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges





Scenario 1:

Jet Reconstruction

Iterative Cone Algorithms

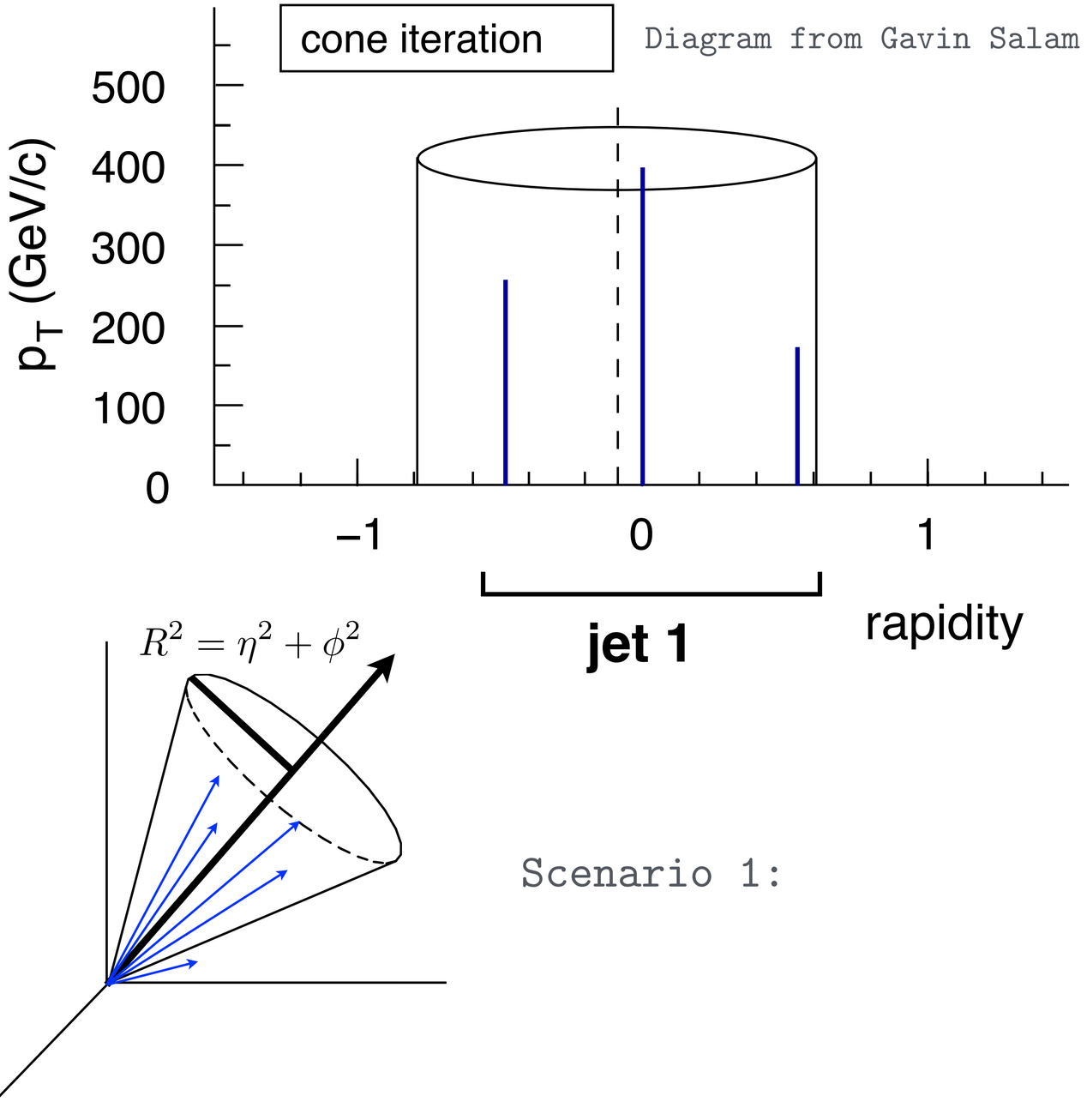
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges



Scenario 1:

Jet Reconstruction

Iterative Cone Algorithms

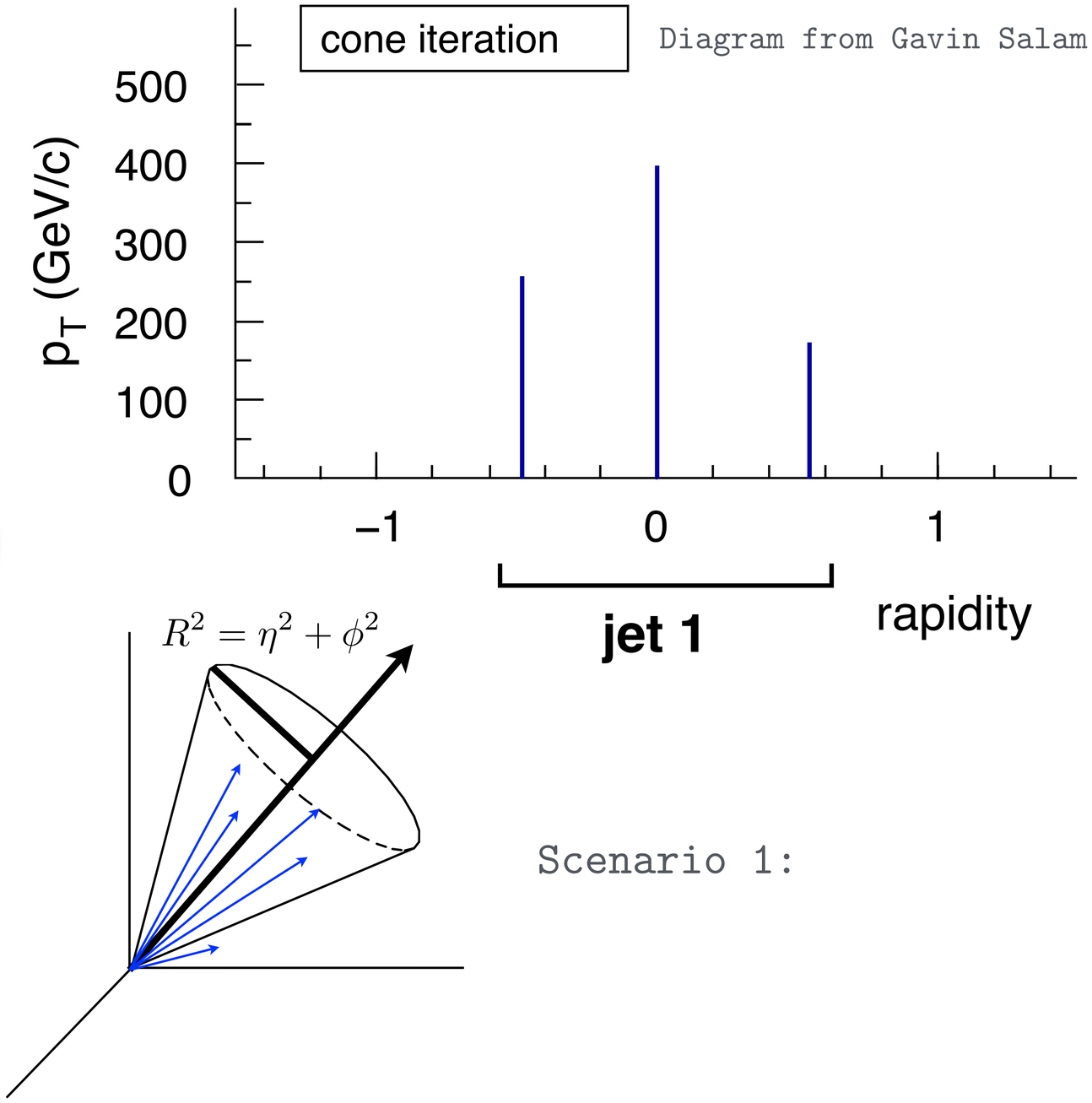
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges



Scenario 1:

Jet Reconstruction

Iterative Cone Algorithms

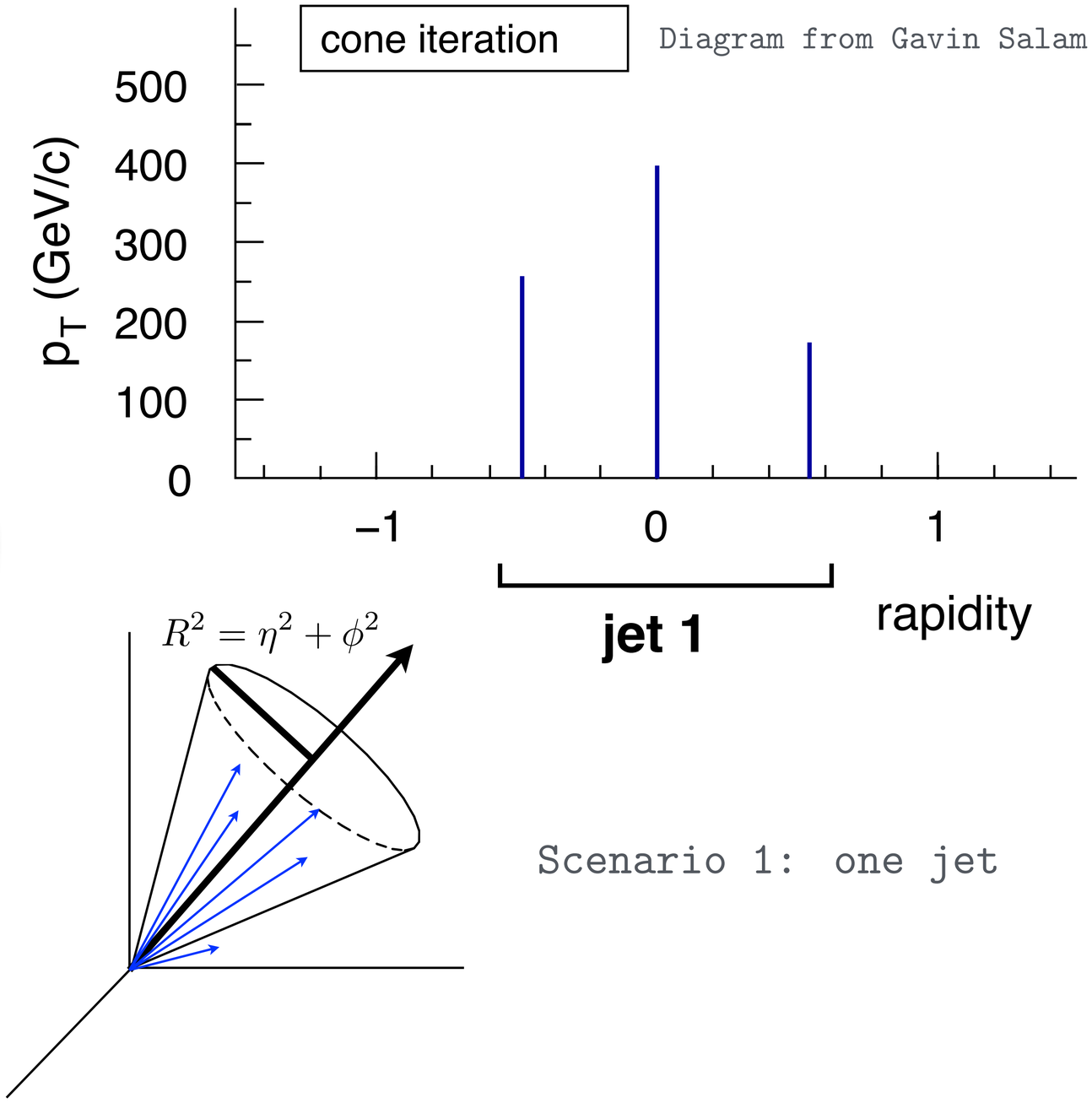
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges



Scenario 1: one jet

Jet Reconstruction

Iterative Cone Algorithms

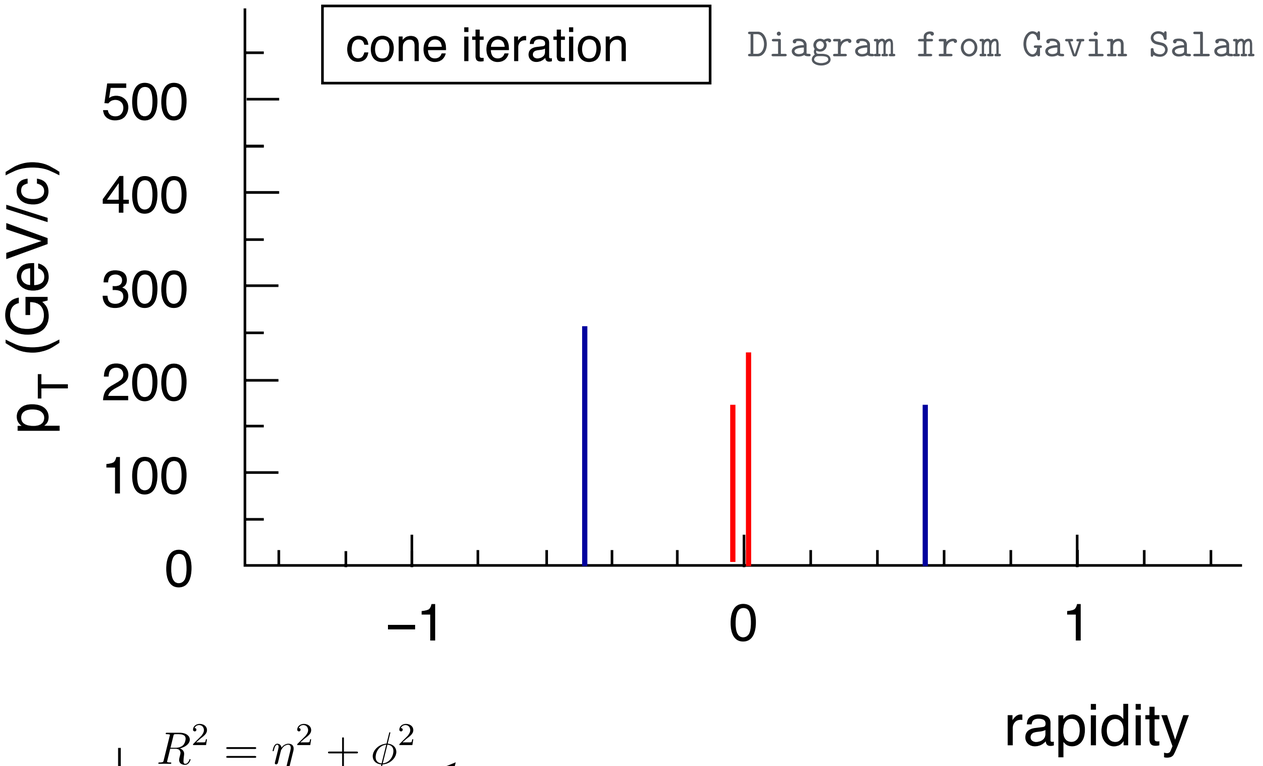
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

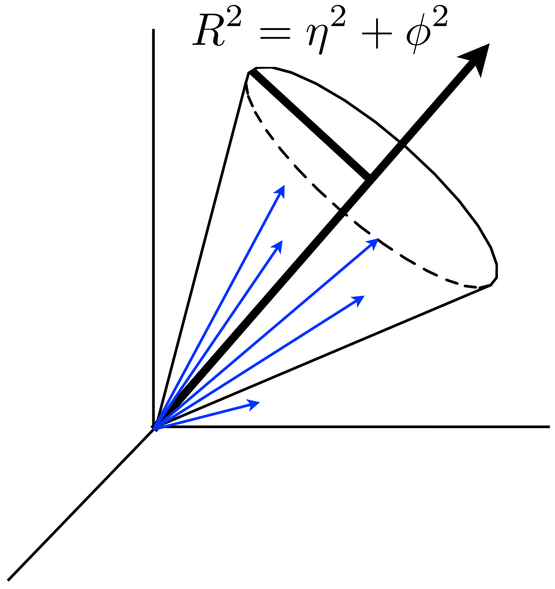
Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges





Scenario 1: one jet

Jet Reconstruction

Iterative Cone Algorithms

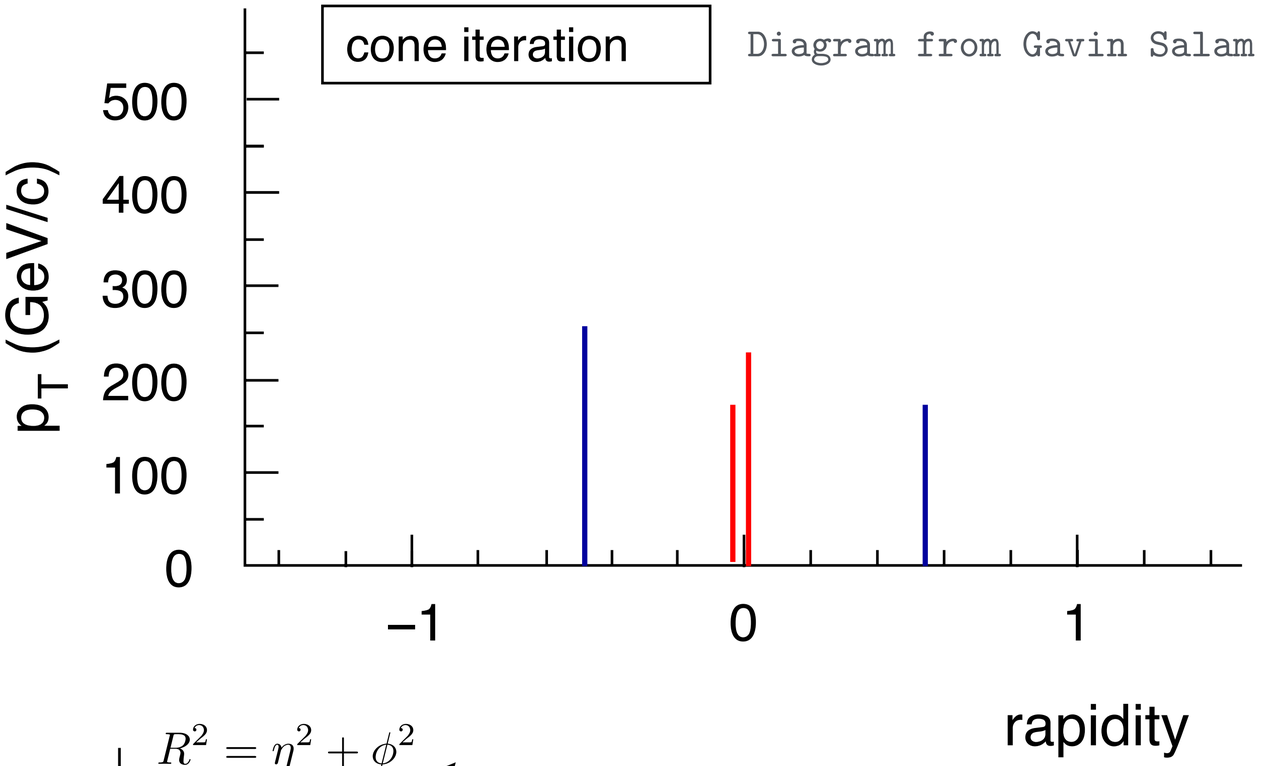
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

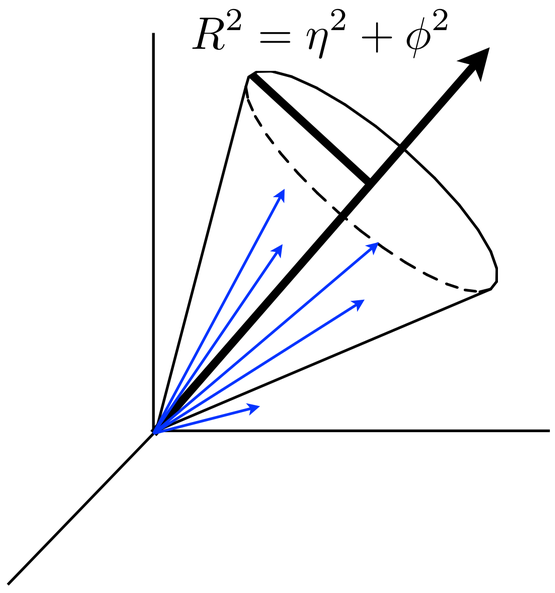
Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges





Scenario 1: one jet

Scenario 2:

Jet Reconstruction

Iterative Cone Algorithms

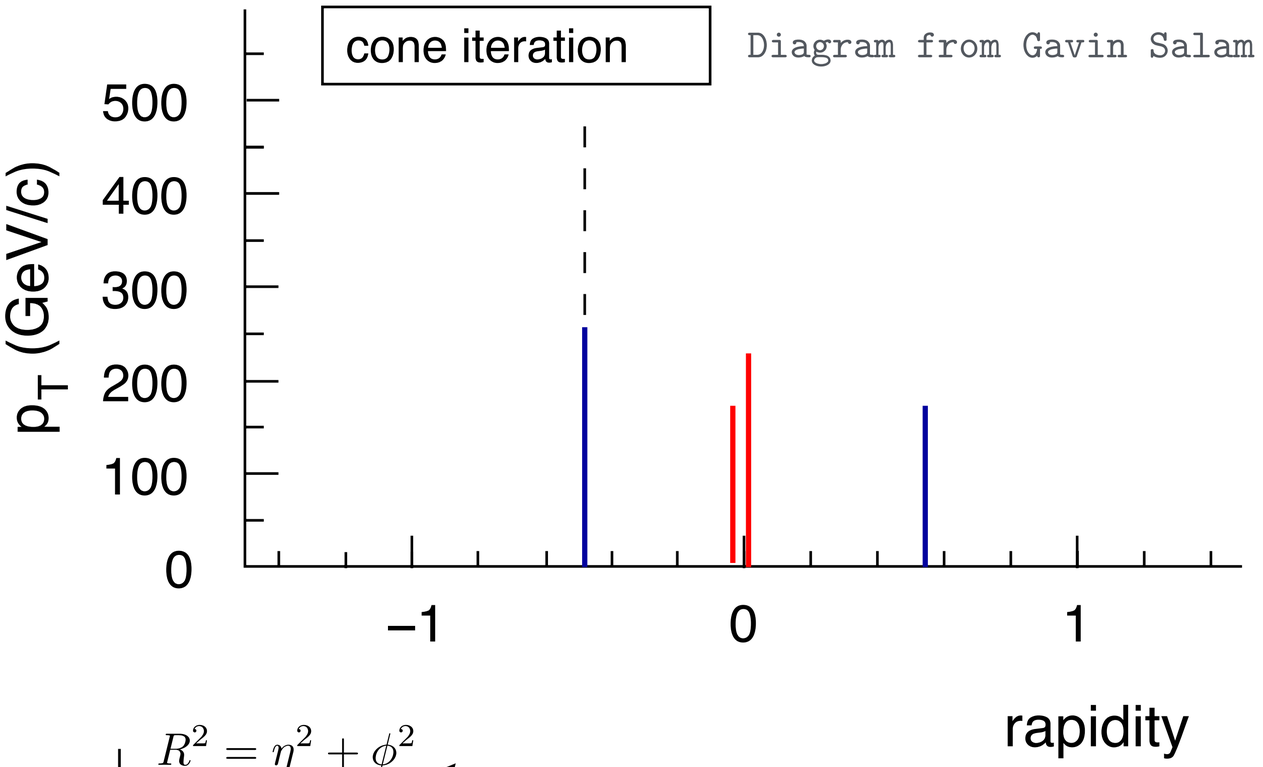
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

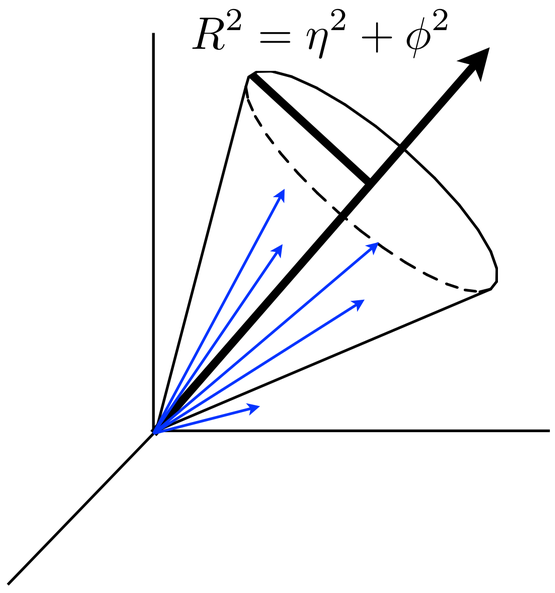
Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges





Scenario 1: one jet

Scenario 2:

Jet Reconstruction

Iterative Cone Algorithms

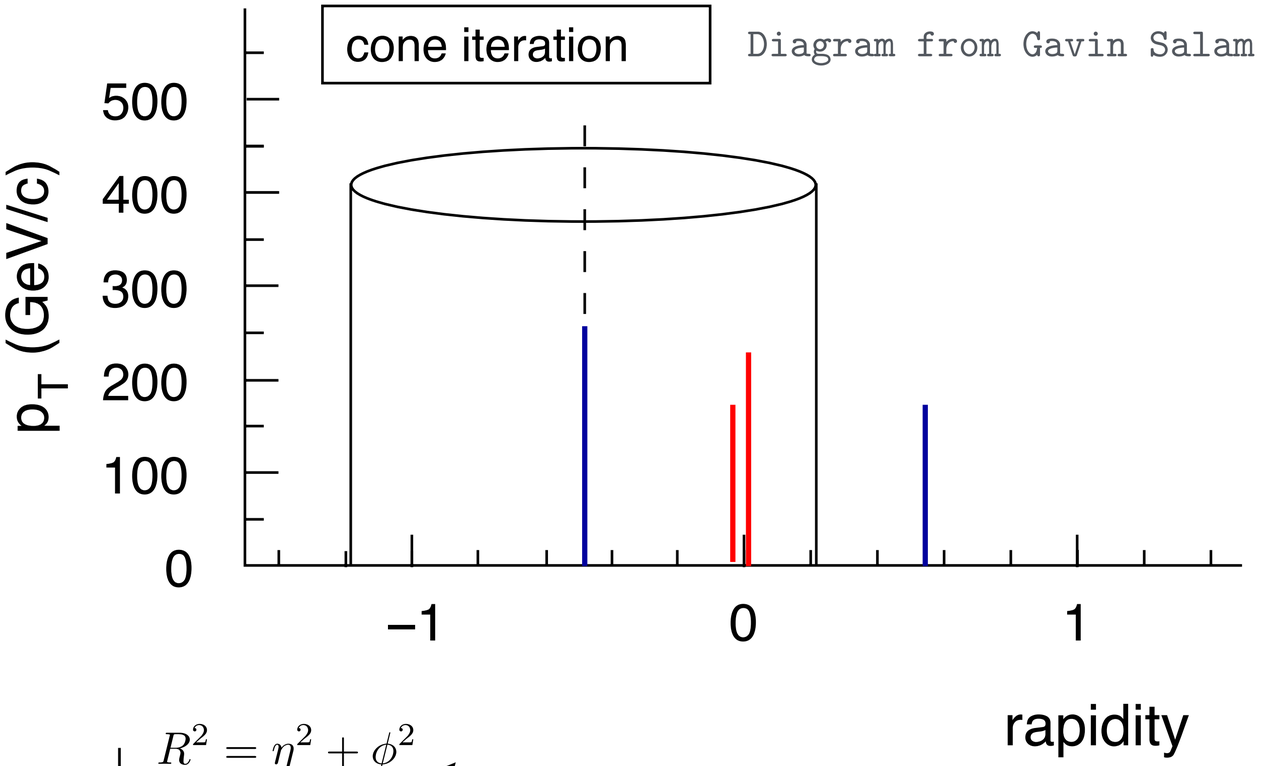
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

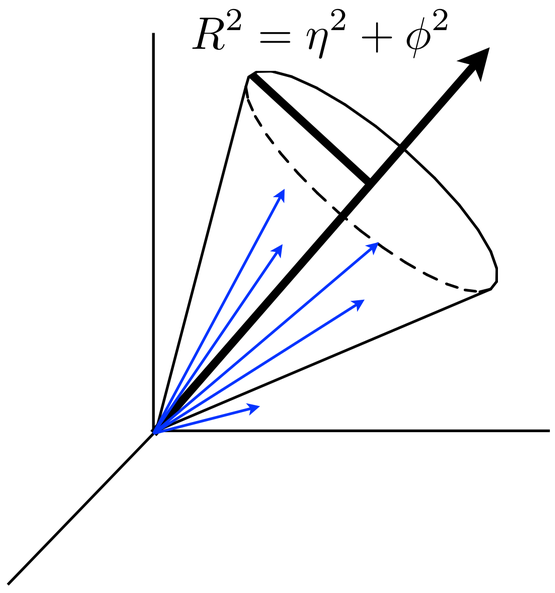
Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges





Scenario 1: one jet

Scenario 2:

Jet Reconstruction

Iterative Cone Algorithms

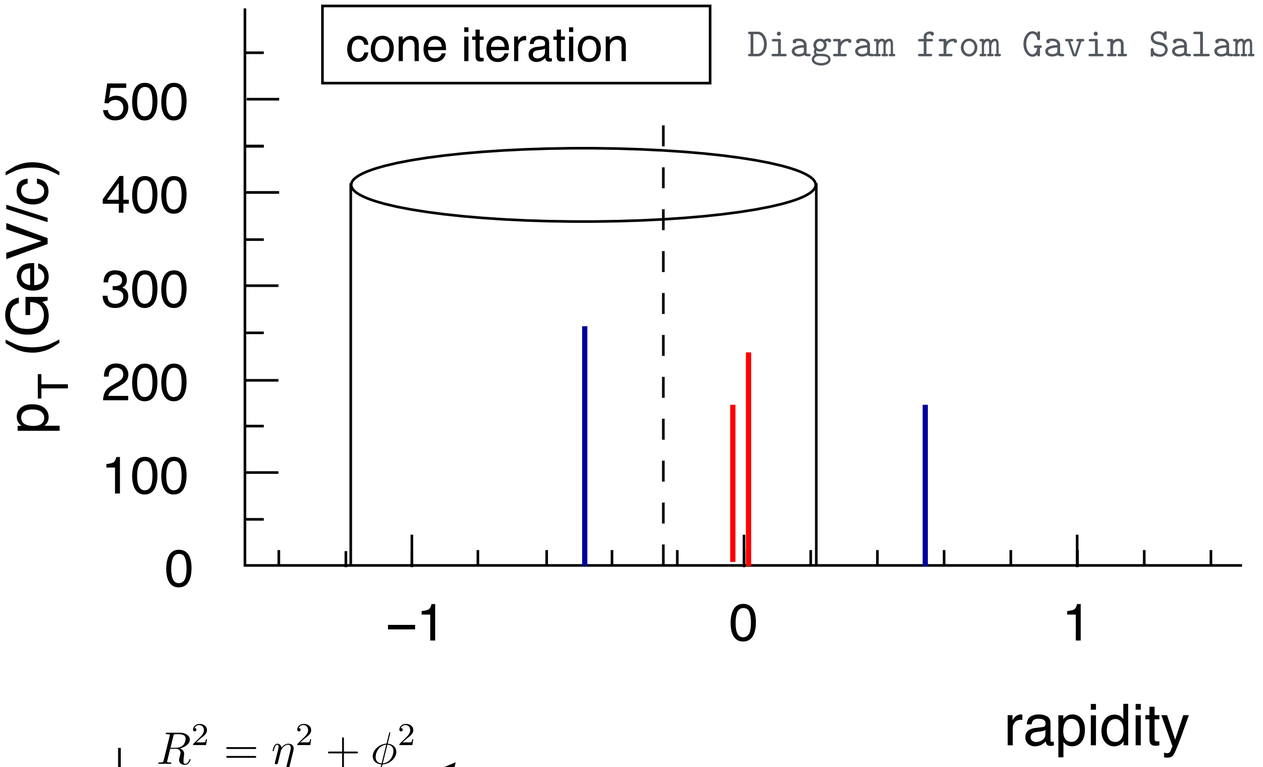
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

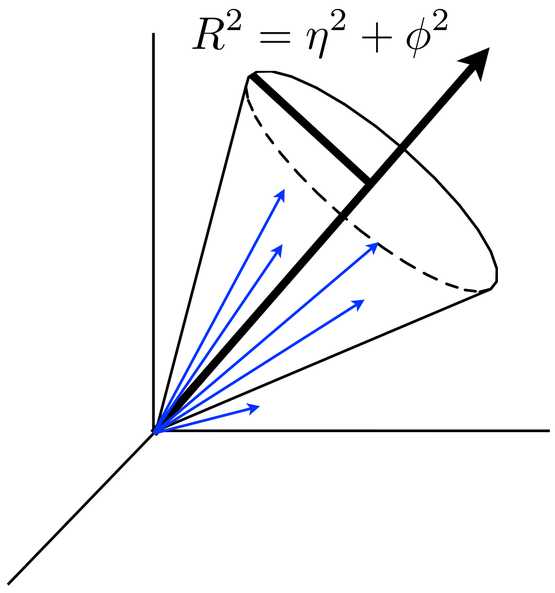
Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges





Scenario 1: one jet

Scenario 2:

Jet Reconstruction

Iterative Cone Algorithms

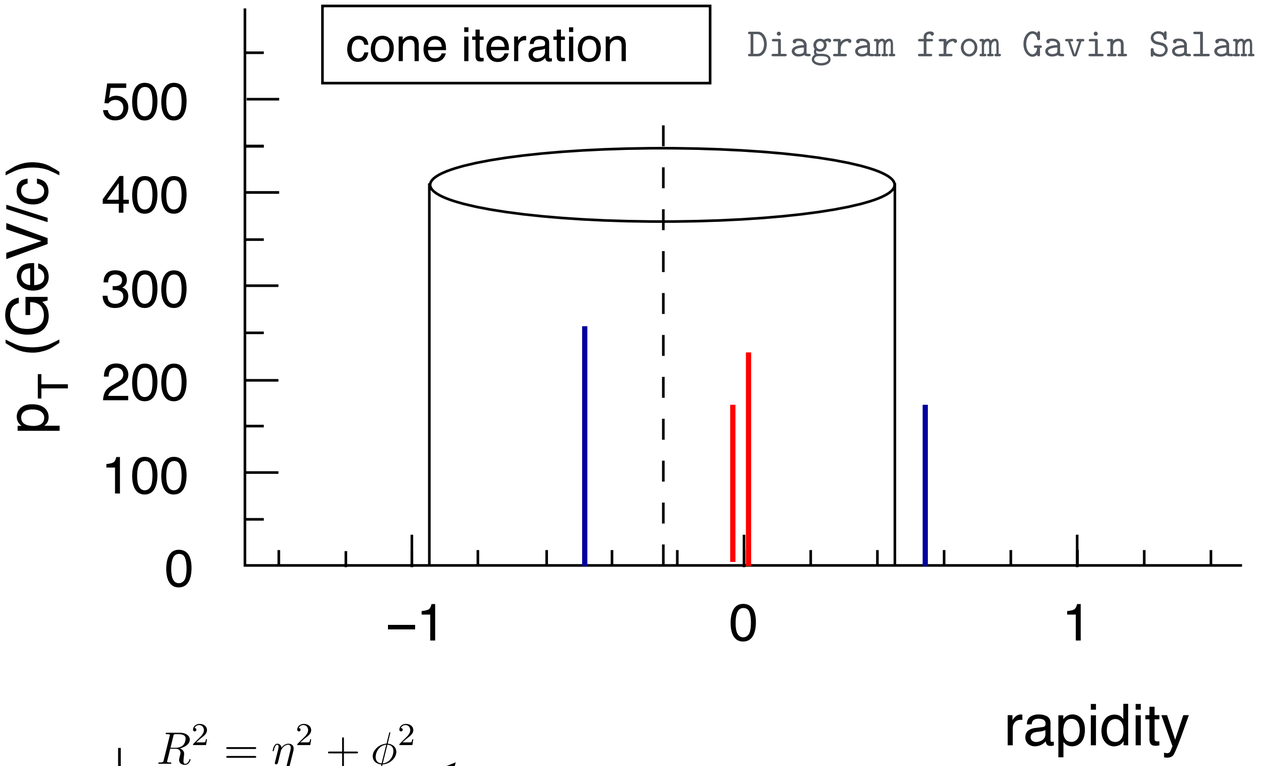
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

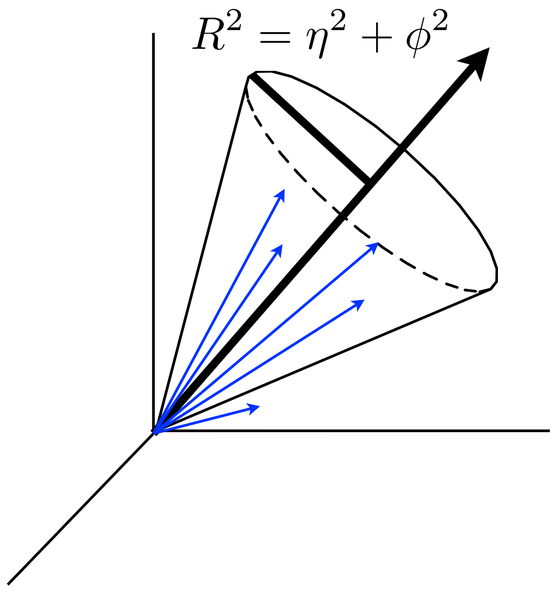
Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges





Scenario 1: one jet

Scenario 2:

Jet Reconstruction

Iterative Cone Algorithms

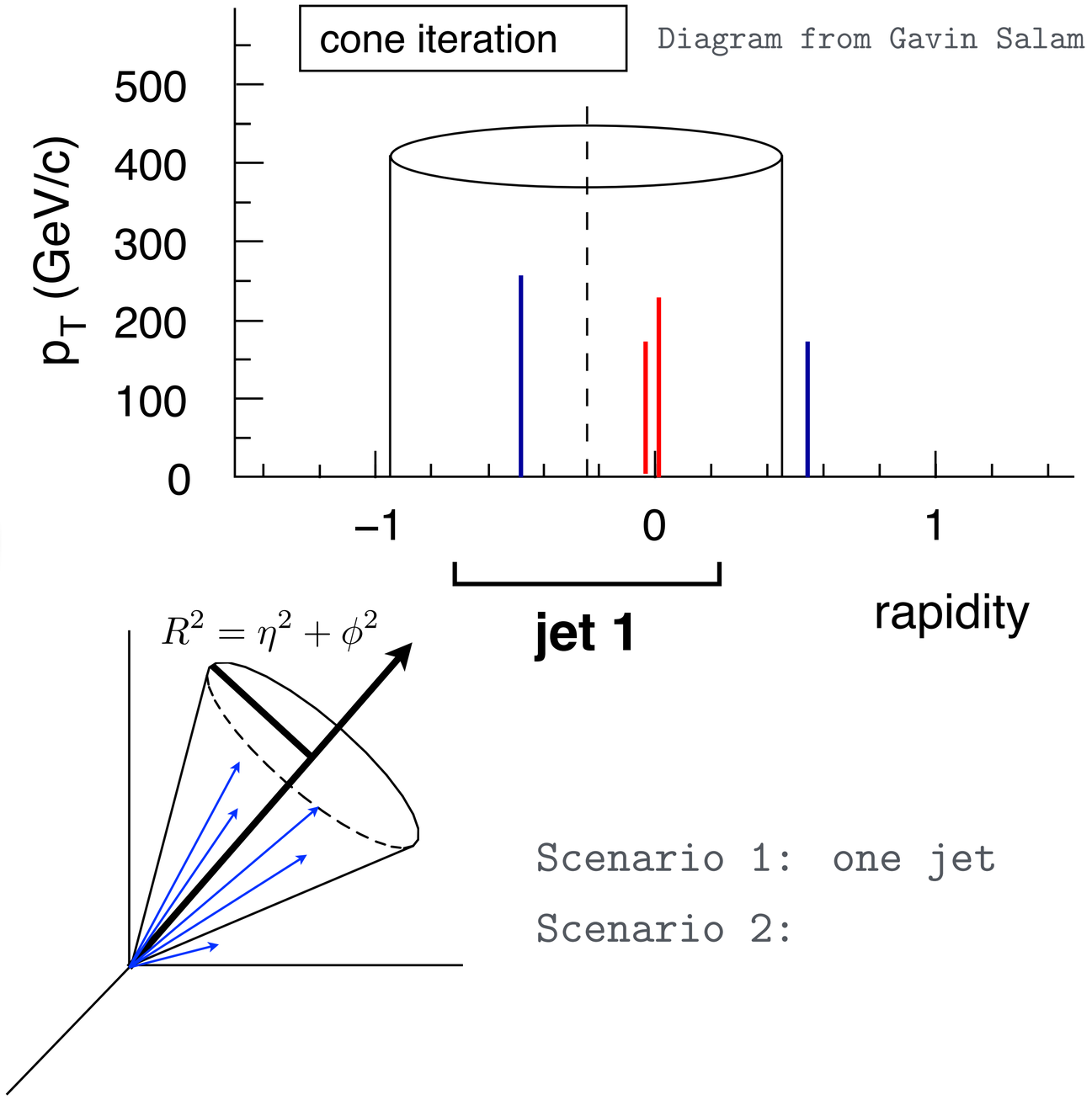
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges



Scenario 1: one jet

Scenario 2:

Jet Reconstruction

Iterative Cone Algorithms

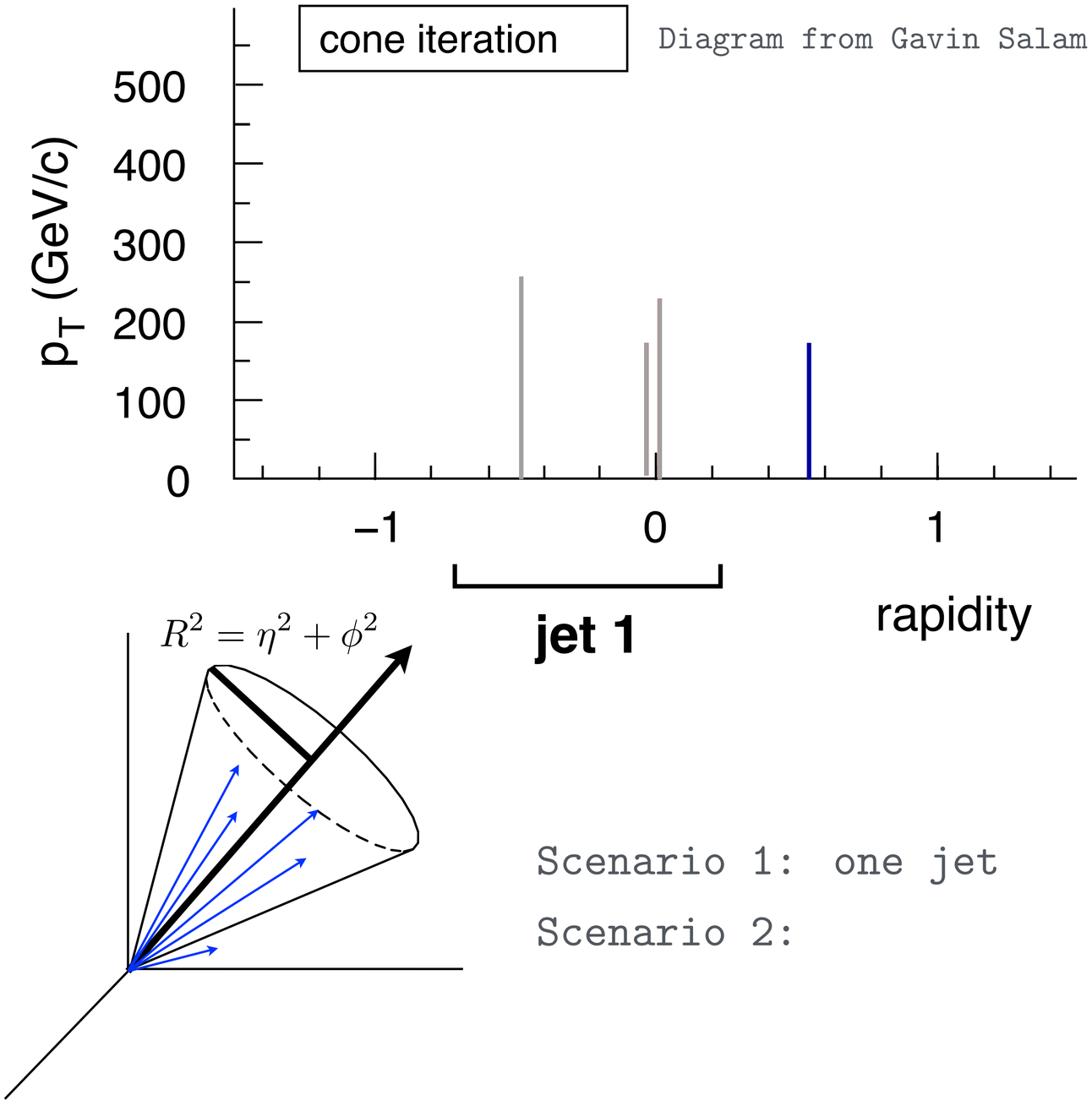
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges



Scenario 1: one jet

Scenario 2:

Jet Reconstruction

Iterative Cone Algorithms

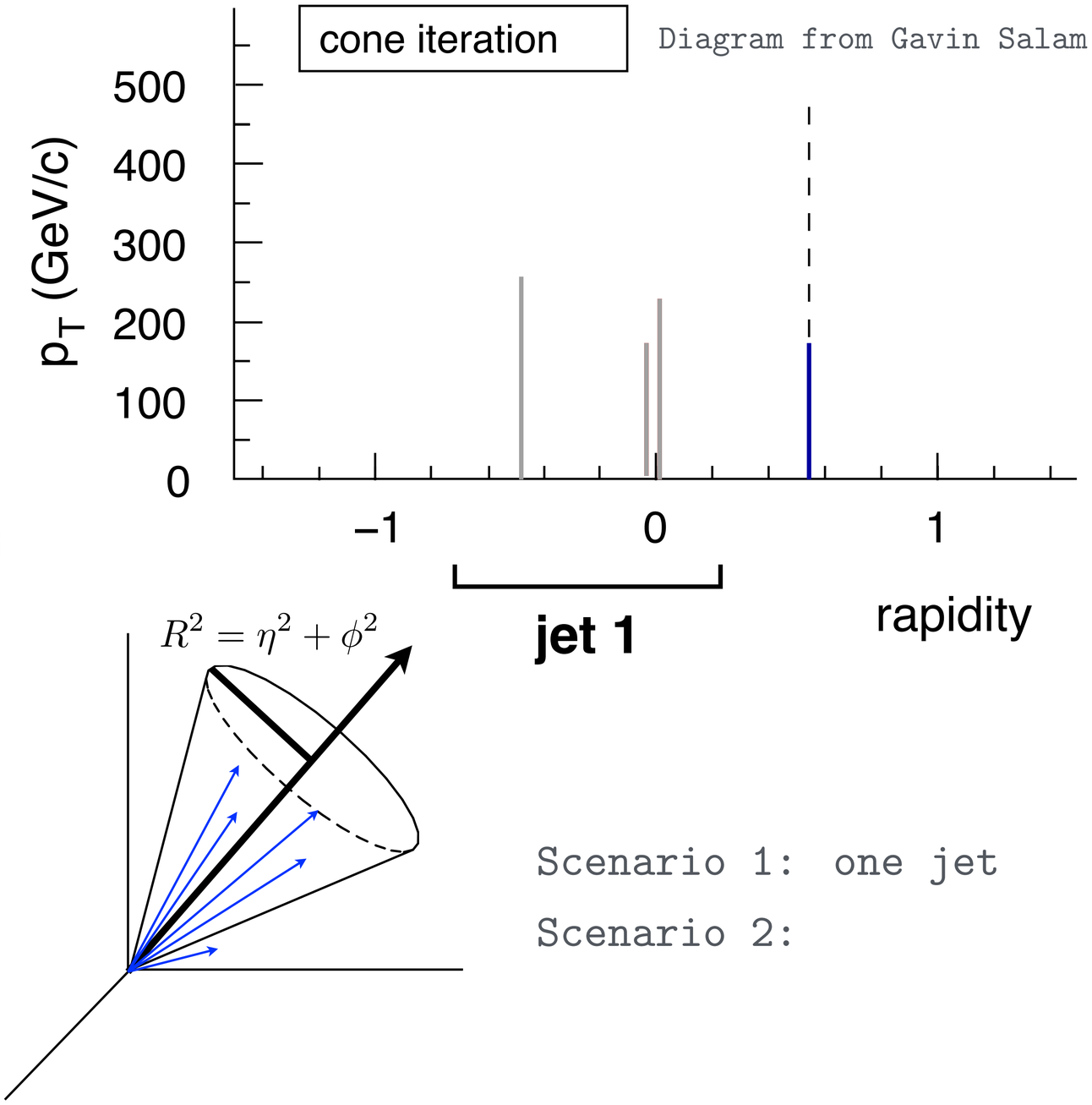
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges



Scenario 1: one jet

Scenario 2:

Jet Reconstruction

Iterative Cone Algorithms

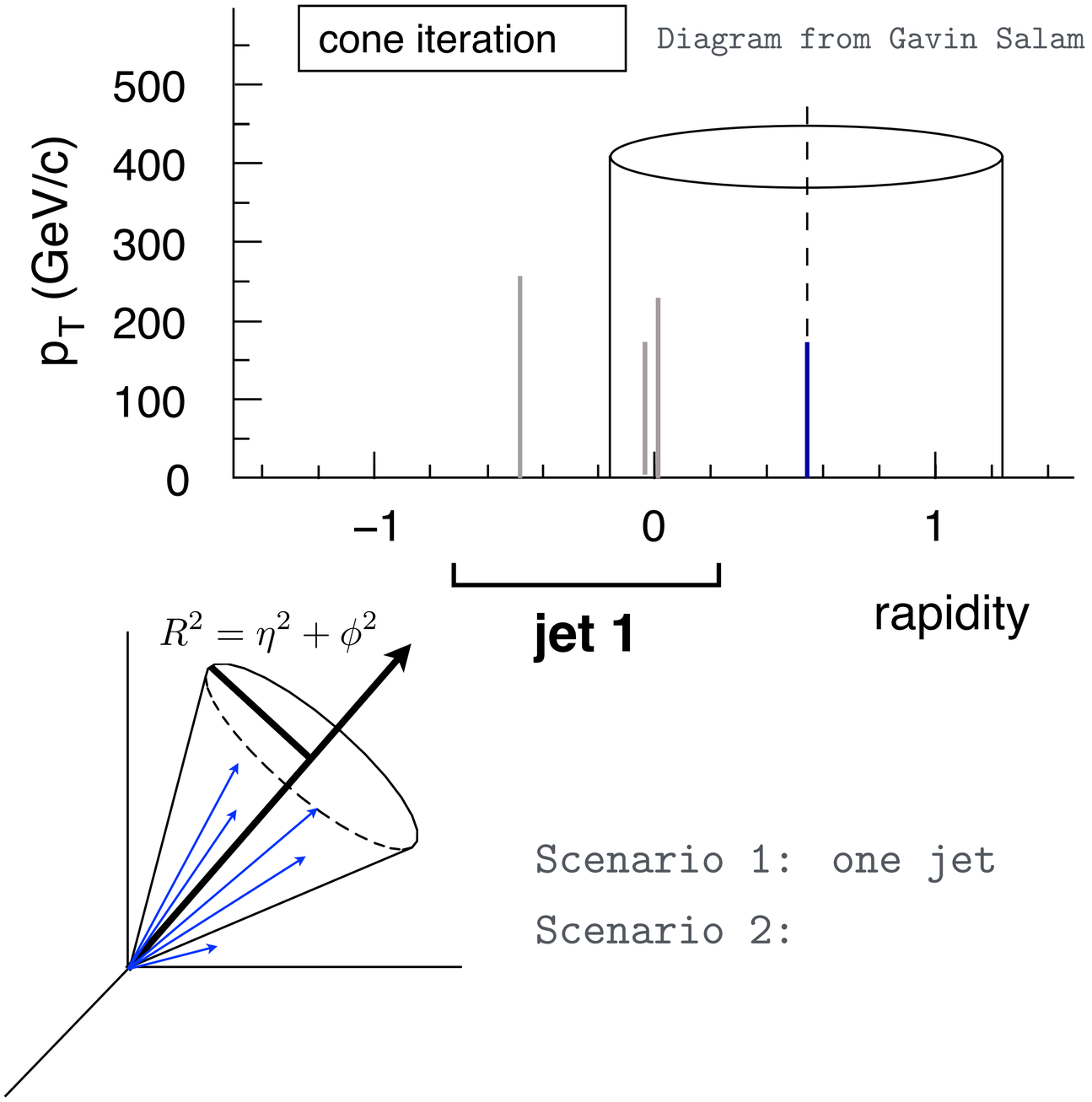
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges



Scenario 1: one jet

Scenario 2:

Jet Reconstruction

Iterative Cone Algorithms

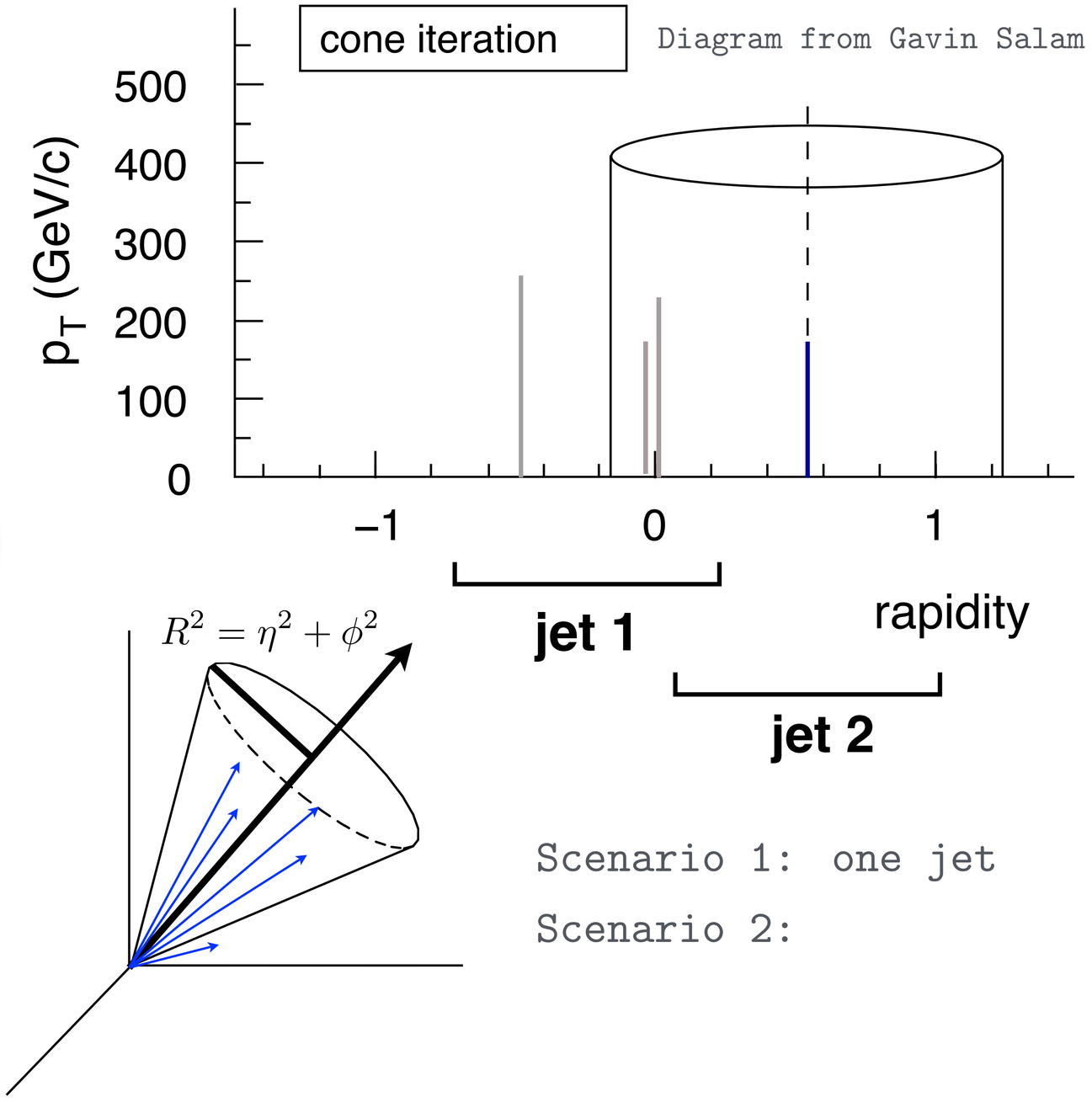
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges



Scenario 1: one jet

Scenario 2:

Jet Reconstruction

Iterative Cone Algorithms

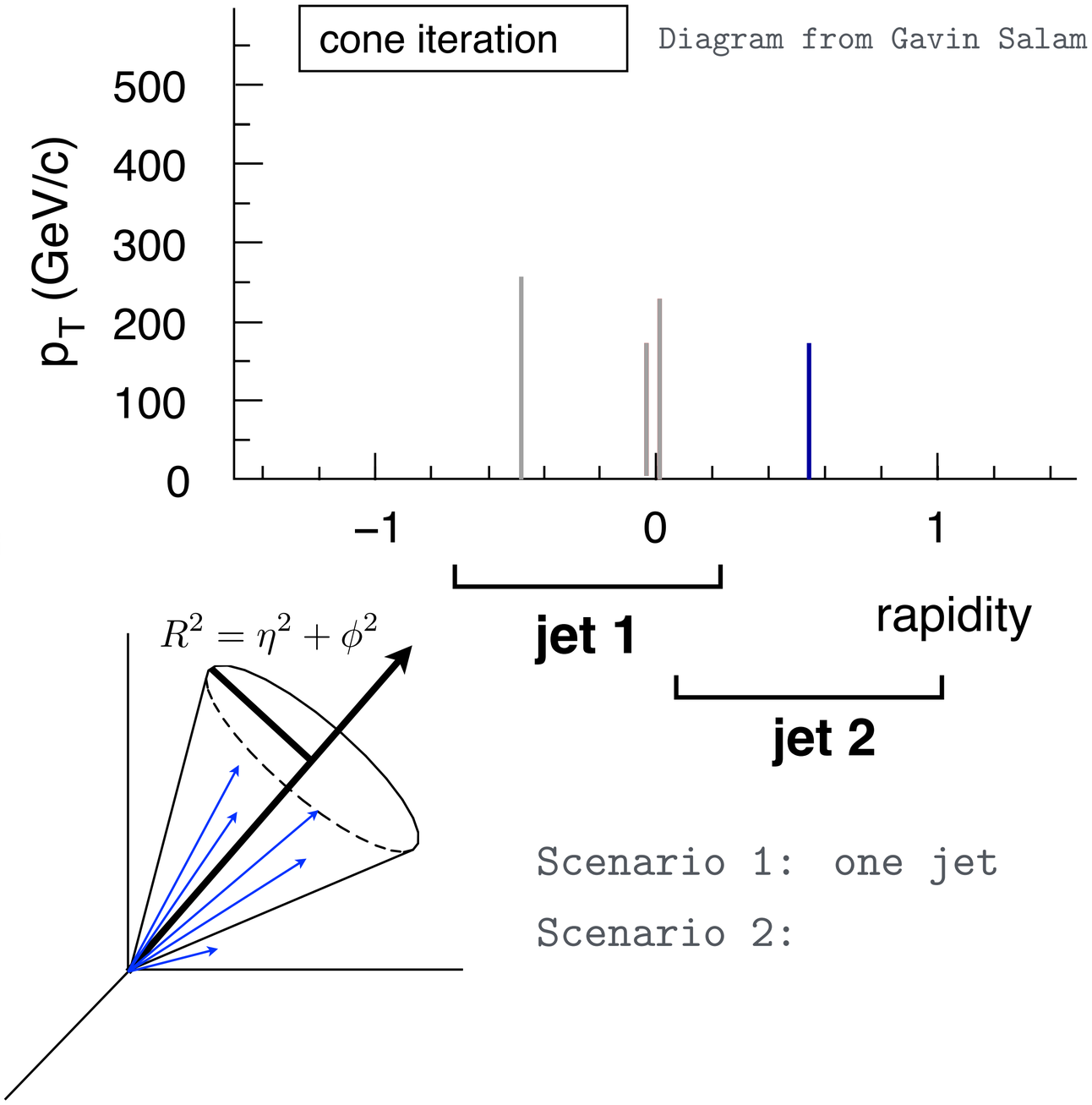
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges



Scenario 1: one jet

Scenario 2:

Jet Reconstruction

Iterative Cone Algorithms

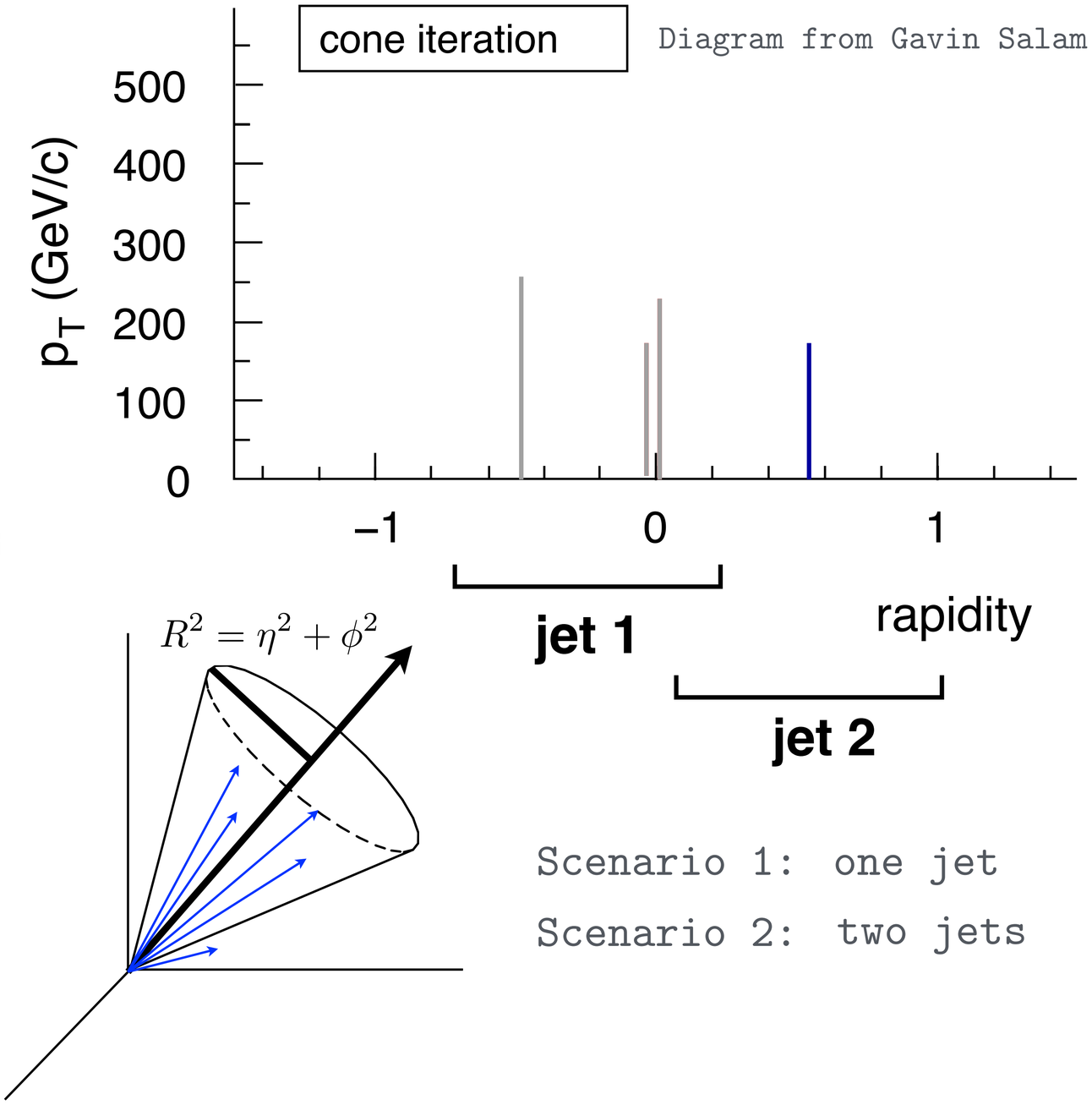
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges



Jet Reconstruction

Iterative Cone Algorithms

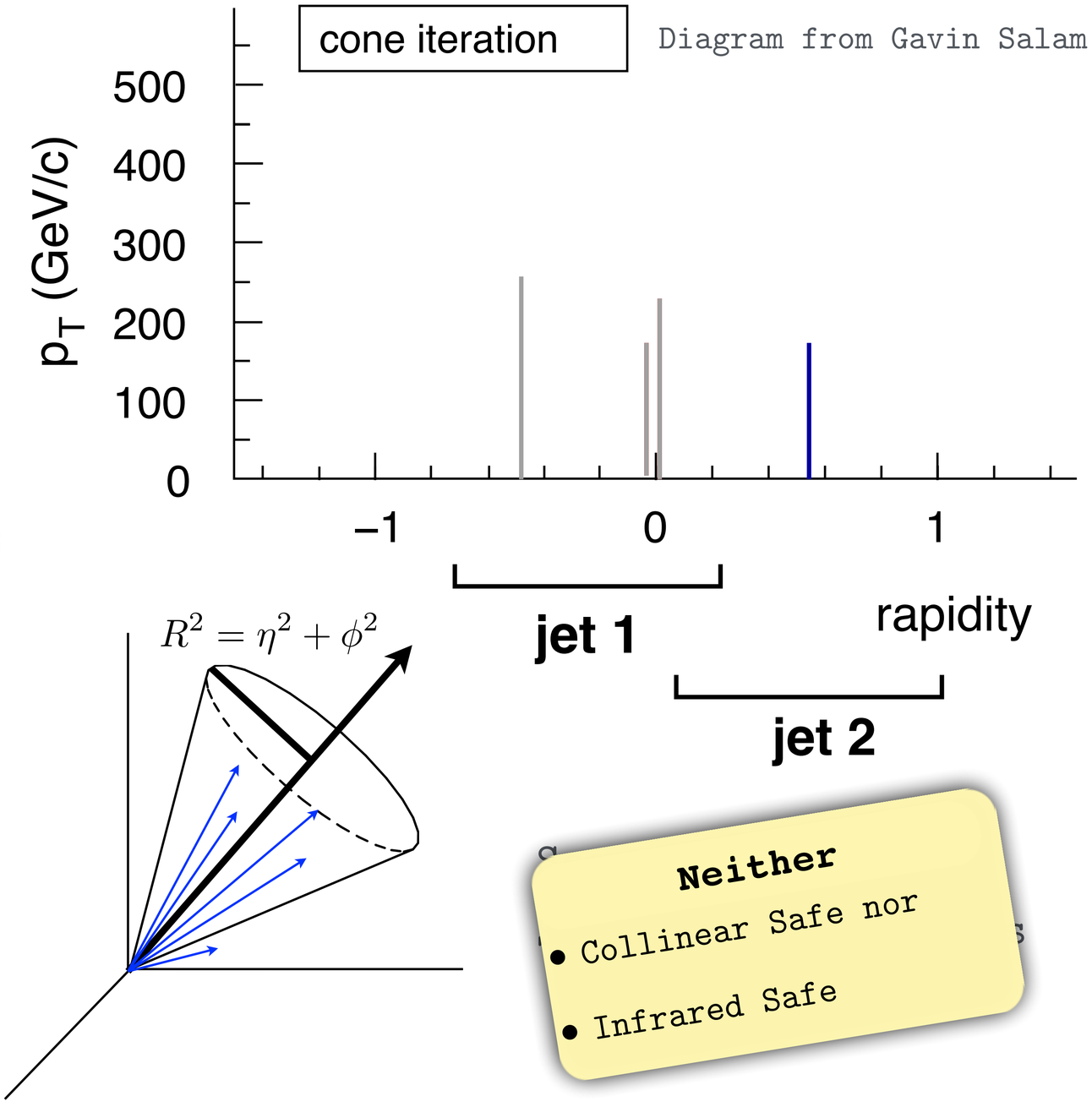
- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Iterative Cone Algorithm

- Associate particles to jet if within a cone of radius R in (η, ϕ) space
- Start with trial seed axis
  - calc centroid over all particles in cone

$$\eta_C = \frac{\sum_i p_{T,i}\eta_i}{\sum_i p_{T,i}} \; ; \; \phi_C = \frac{\sum_i p_{T,i}\phi_i}{\sum_i p_{T,i}}$$

- Throw new trial cone about centroid
  - iterate until centroid converges



Neither

- Collinear Safe nor
- Infrared Safe

Jet Reconstruction

Iterative Cone Algorithms

- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Jet Reconstruction

Iterative Cone Algorithms

- Historically used (before LHC) at hadron colliders
- But also tried at lepton colliders

Sequential Clustering Algorithms

- Historically used (before LHC) at lepton colliders
- Now standard at LHC

Jet Reconstruction

Sequential Clustering Algorithms

- Historically used (before LHC) at lepton colliders
- Now standard at LHC

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy $y_{ij} > y_{\text{cut}}$

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

Jet Reconstruction

JADE

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2

Jet Reconstruction

Sequential Clustering Algorithms

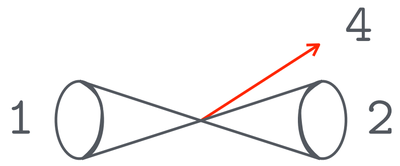
- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2



Jet Reconstruction

Sequential Clustering Algorithms

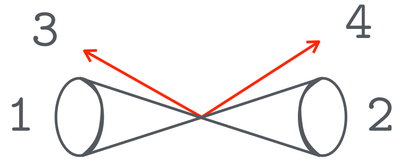
- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2



Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

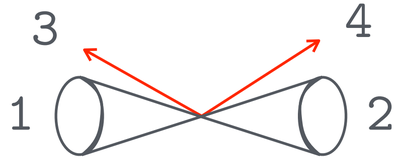
Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2

$y_{13} < y_{34}$; $y_{24} < y_{34}$



Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

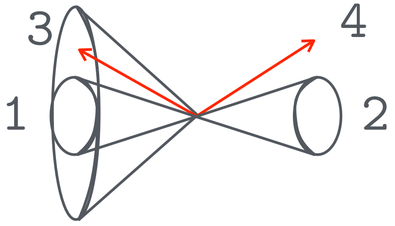
Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2

$y_{13} < y_{34}$; $y_{24} < y_{34}$



Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

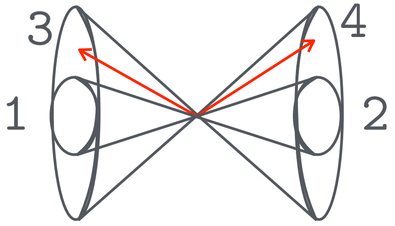
Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2

$y_{13} < y_{34}$; $y_{24} < y_{34}$



Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

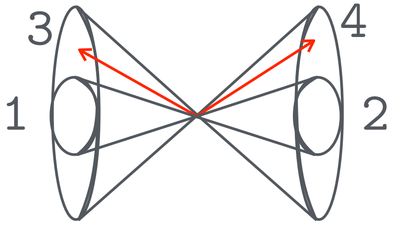
Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2

$y_{13} < y_{34}$; $y_{24} < y_{34}$



2 jets

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

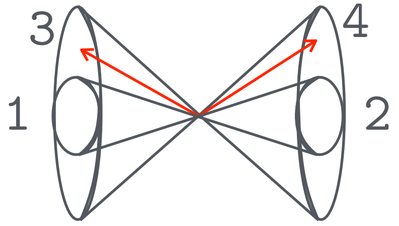
Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2

$y_{13} < y_{34}$; $y_{24} < y_{34}$



2 jets

Jet Reconstruction

Sequential Clustering Algorithms

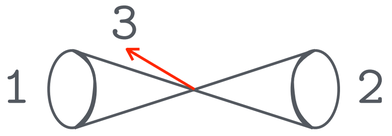
- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

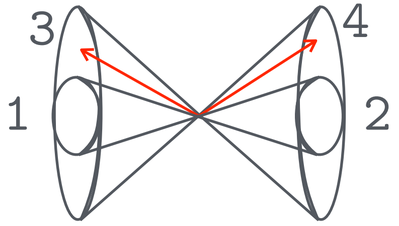
- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2



$y_{13} < y_{34}$; $y_{24} < y_{34}$



2 jets

Jet Reconstruction

Sequential Clustering Algorithms

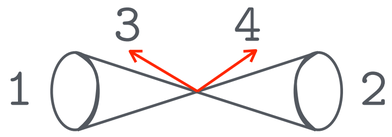
- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

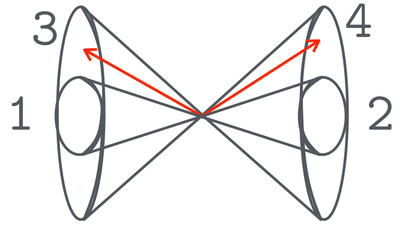
- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2



$y_{13} < y_{34}$; $y_{24} < y_{34}$



2 jets

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

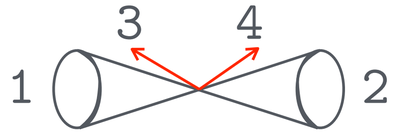
Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

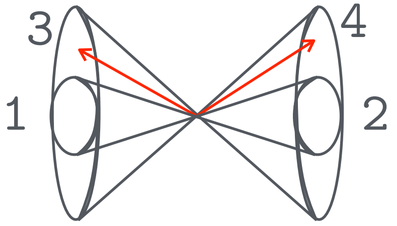
JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2

$y_{34} < y_{13}$; $y_{34} < y_{24}$



$y_{13} < y_{34}$; $y_{24} < y_{34}$



2 jets

Jet Reconstruction

Sequential Clustering Algorithms

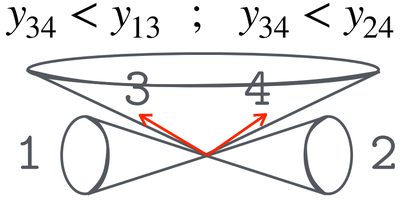
- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

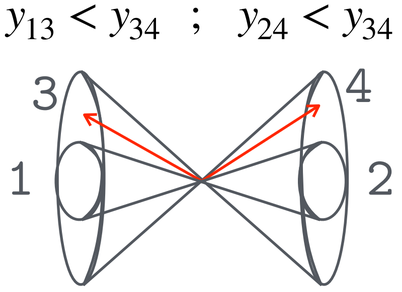
Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2





2 jets

Jet Reconstruction

Sequential Clustering Algorithms

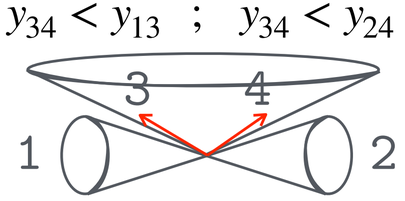
- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

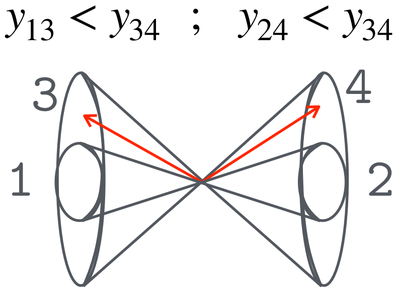
- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

JADE

- The metric is $y_{ij} = \dfrac{2E_iE_j(1-\cos\theta_{ij})}{E_{\text{cm}}^2} = \dfrac{m_{ij}^2}{E_{\text{cm}}^2}$
- Two particles are combined if $y_{ij} < y_{\text{cut}}$
- Possible to reconstruct "ghost jets"
- Example: two soft gluons emitted close to q1 and q2; the gg invariant mass, y_{34}, can be smaller than that of any qg pairs and hence 3 jets can be reconstructed instead of 2



3rd "ghost" jet



2 jets

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

Jet Reconstruction

DURHAM

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

DURHAM

- The metric is

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

DURHAM

- The metric is $y_{ij} = \dfrac{2\min(E_i^2, E_j^2)(1 - \cos\theta_{ij})}{E_{\text{cm}}^2}$

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

DURHAM

- The metric is $y_{ij} = \dfrac{2\min(E_i^2, E_j^2)(1-\cos\theta_{ij})}{E_{\text{cm}}^2}$
- Reduces sensitivity to "ghost" jets

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

DURHAM

- The metric is $y_{ij} = \dfrac{2\min(E_i^2, E_j^2)(1-\cos\theta_{ij})}{E_{\text{cm}}^2}$
- Reduces sensitivity to "ghost" jets
- For small emission angles,

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

DURHAM

- The metric is $y_{ij} = \dfrac{2\min(E_i^2, E_j^2)(1-\cos\theta_{ij})}{E_{\text{cm}}^2}$
- Reduces sensitivity to "ghost" jets
- For small emission angles,

$$\theta_{ij} \to 0$$

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

DURHAM

- The metric is $y_{ij} = \dfrac{2\min(E_i^2, E_j^2)(1-\cos\theta_{ij})}{E_{\text{cm}}^2}$
- Reduces sensitivity to "ghost" jets
- For small emission angles,

$$\theta_{ij} \to 0 \implies y_{ij} = \frac{2\min(E_i^2, E_j^2)[1-(1-\theta_{ij}^2/2+\ldots)]}{E_{\text{cm}}^2}$$

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

DURHAM

- The metric is $y_{ij} = \dfrac{2\min(E_i^2, E_j^2)(1-\cos\theta_{ij})}{E_{\text{cm}}^2}$
- Reduces sensitivity to "ghost" jets
- For small emission angles,

$$\theta_{ij} \to 0 \implies y_{ij} = \frac{2\min(E_i^2, E_j^2)[1-(1-\theta_{ij}^2/2+\ldots)]}{E_{\text{cm}}^2} \approx \frac{\min(E_i^2, E_j^2)\theta_{ij}^2}{E_{\text{cm}}^2}$$

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i, j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

DURHAM

- The metric is $y_{ij} = \dfrac{2\min(E_i^2, E_j^2)(1-\cos\theta_{ij})}{E_{\text{cm}}^2}$
- Reduces sensitivity to "ghost" jets
- For small emission angles,

$$\theta_{ij} \to 0 \implies y_{ij} = \frac{2\min(E_i^2, E_j^2)[1-(1-\theta_{ij}^2/2+\ldots)]}{E_{\text{cm}}^2}$$

$$\approx \frac{\min(E_i^2, E_j^2)\theta_{ij}^2}{E_{\text{cm}}^2}$$

$$= \frac{\min(k_{T,i}^2, k_{T,j}^2)}{E_{\text{cm}}^2}$$

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

DURHAM

- The metric is $y_{ij} = \dfrac{2\min(E_i^2, E_j^2)(1-\cos\theta_{ij})}{E_{\text{cm}}^2}$
- Reduces sensitivity to "ghost" jets
- For small emission angles,

$$\theta_{ij} \to 0 \implies y_{ij} = \frac{2\min(E_i^2, E_j^2)[1-(1-\theta_{ij}^2/2+...)]}{E_{\text{cm}}^2}$$

$$\approx \frac{\min(E_i^2, E_j^2)\theta_{ij}^2}{E_{\text{cm}}^2}$$

$$= \frac{\min(k_{T,i}^2, k_{T,j}^2)}{E_{\text{cm}}^2}$$

- metric becomes the minimum kT of one particle w.r.t. the other

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

DURHAM

- The metric is $y_{ij} = \dfrac{2\min(E_i^2, E_j^2)(1-\cos\theta_{ij})}{E_{\text{cm}}^2}$
- Reduces sensitivity to "ghost" jets
- For small emission angles,

$$\theta_{ij} \to 0 \implies y_{ij} = \frac{2\min(E_i^2, E_j^2)[1-(1-\theta_{ij}^2/2+...)]}{E_{\text{cm}}^2}$$

$$\approx \frac{\min(E_i^2, E_j^2)\theta_{ij}^2}{E_{\text{cm}}^2}$$

$$= \frac{\min(k_{T,i}^2, k_{T,j}^2)}{E_{\text{cm}}^2}$$

- metric becomes the minimum kT of one particle w.r.t. the other
- Soft, collinear radiation associated to quark jets

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

DURHAM

- The metric is $y_{ij} = \dfrac{2\min(E_i^2, E_j^2)(1-\cos\theta_{ij})}{E_{\text{cm}}^2}$
- Reduces sensitivity to "ghost" jets
- For small emission angles,

$$\theta_{ij} \to 0 \implies y_{ij} = \frac{2\min(E_i^2, E_j^2)[1-(1-\theta_{ij}^2/2+...)]}{E_{\text{cm}}^2}$$

$$\approx \frac{\min(E_i^2, E_j^2)\theta_{ij}^2}{E_{\text{cm}}^2}$$

$$= \frac{\min(k_{T,i}^2, k_{T,j}^2)}{E_{\text{cm}}^2}$$

- metric becomes the minimum kT of one particle w.r.t. the other
- Soft, collinear radiation associated to quark jets

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

DURHAM

- The metric is $y_{ij} = \dfrac{2\min(E_i^2, E_j^2)(1-\cos\theta_{ij})}{E_{\text{cm}}^2}$
- Reduces sensitivity to "ghost" jets
- For small emission angles,

$$\theta_{ij} \to 0 \implies y_{ij} = \frac{2\min(E_i^2, E_j^2)[1-(1-\theta_{ij}^2/2+\ldots)]}{E_{\text{cm}}^2} \approx \frac{\min(E_i^2, E_j^2)\theta_{ij}^2}{E_{\text{cm}}^2} = \frac{\min(k_{T,i}^2, k_{T,j}^2)}{E_{\text{cm}}^2}$$

- metric becomes the minimum kT of one particle w.r.t. the other
- Soft, collinear radiation associated to quark jets

Jet Reconstruction

Sequential Clustering Algorithms

- To "cluster" implies a distance metric:
  - For every pair of particles (i,j) compute a distance measure y_{ij} (metric defined by algorithm)
- Cluster according to a resolution parameter called y_{cut}
  - if $\min(y_{ij}) < y_{\text{cut}}$, then particles i and j are merged (according to a recombination scheme) and become new object k
- Iterate until all particle/object pairs satisfy y

Advantages

- Collinear Safe AND Infrared Safe
- Num jets related to resolution param y_{cut}

Disadvantage

- Underlying beam remnant particles can obscure soft particles emitted from the hard scatter partons

DURHAM

- The metric is $y_{ij} = \dfrac{2\min(E_i^2, E_j^2)(1-\cos\theta_{ij})}{E_{\text{cm}}^2}$
- Reduces sensitivity to "ghost" jets
- For small emission angles,

$$\theta_{ij} \to 0 \implies y_{ij} = \frac{2\min(E_i^2, E_j^2)[1-(1-\theta_{ij}^2/2+\ldots)]}{E_{\text{cm}}^2} \approx \frac{\min(E_i^2, E_j^2)\theta_{ij}^2}{E_{\text{cm}}^2} = \frac{\min(k_{T,i}^2, k_{T,j}^2)}{E_{\text{cm}}^2}$$

becomes the minimum kT of one
e w.r.t. the other
llinear radiation associated to
ts

Jet Reconstruction

Richard Cavanaugh, Fermilab/UIC, HCP Summer School 31 July 2024

Jet Reconstruction

kT Algorithm

Jet Reconstruction

kT Algorithm

- List of particles with

Jet Reconstruction

kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2$

Jet Reconstruction

kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2} \ln \frac{E + p}{E - p}$

Jet Reconstruction

kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2} \ln \frac{E + p}{E - p}$
- For each particle, define the distance from the beam

Jet Reconstruction

kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2} \ln \frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$

Jet Reconstruction

kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

Jet Reconstruction

kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Jet Reconstruction

kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:

$R = \{0.4, 0.8, 1.0\}$

Jet Reconstruction

kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j

Jet Reconstruction

kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k

Jet Reconstruction

kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets

Jet Reconstruction

kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Jet Reconstruction

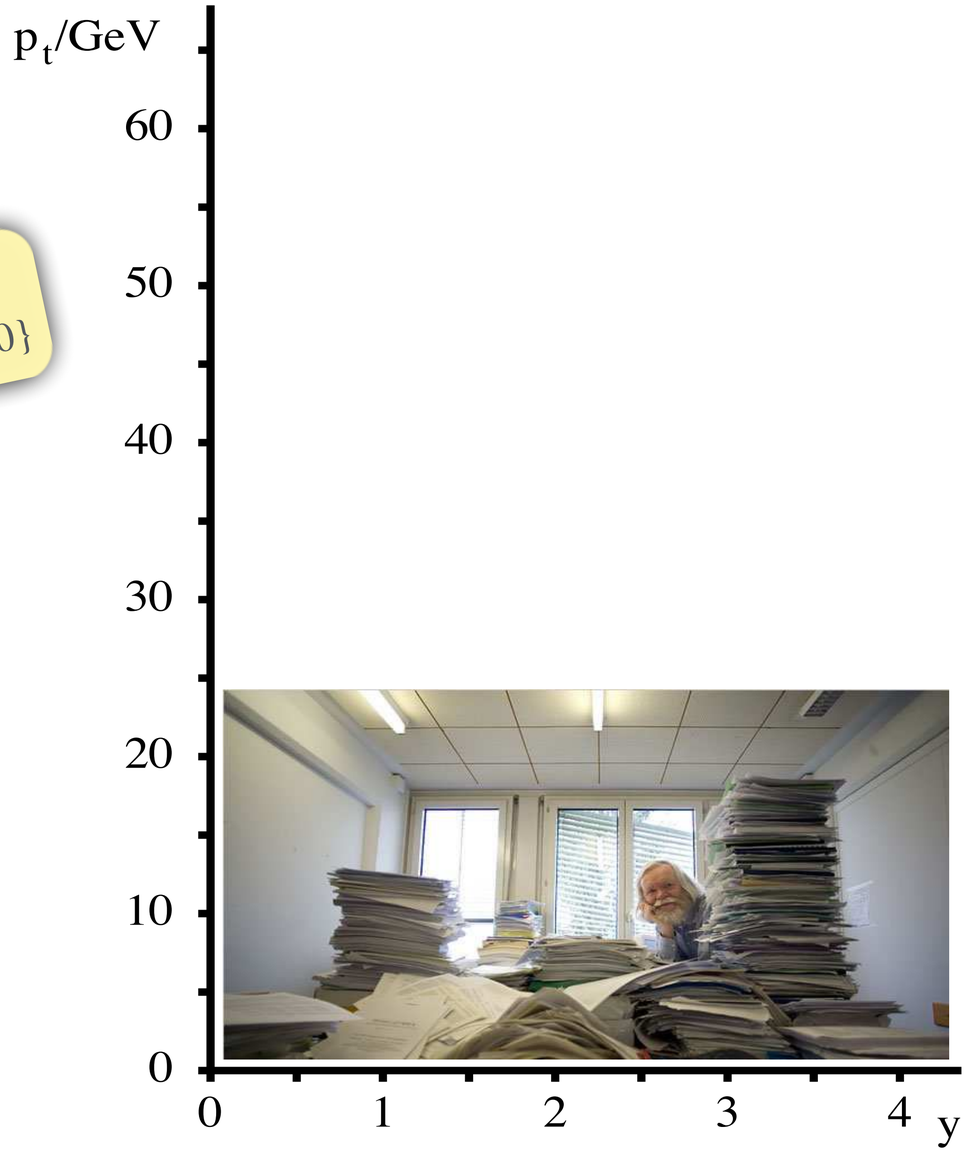
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Jet Reconstruction

kT Algorithm

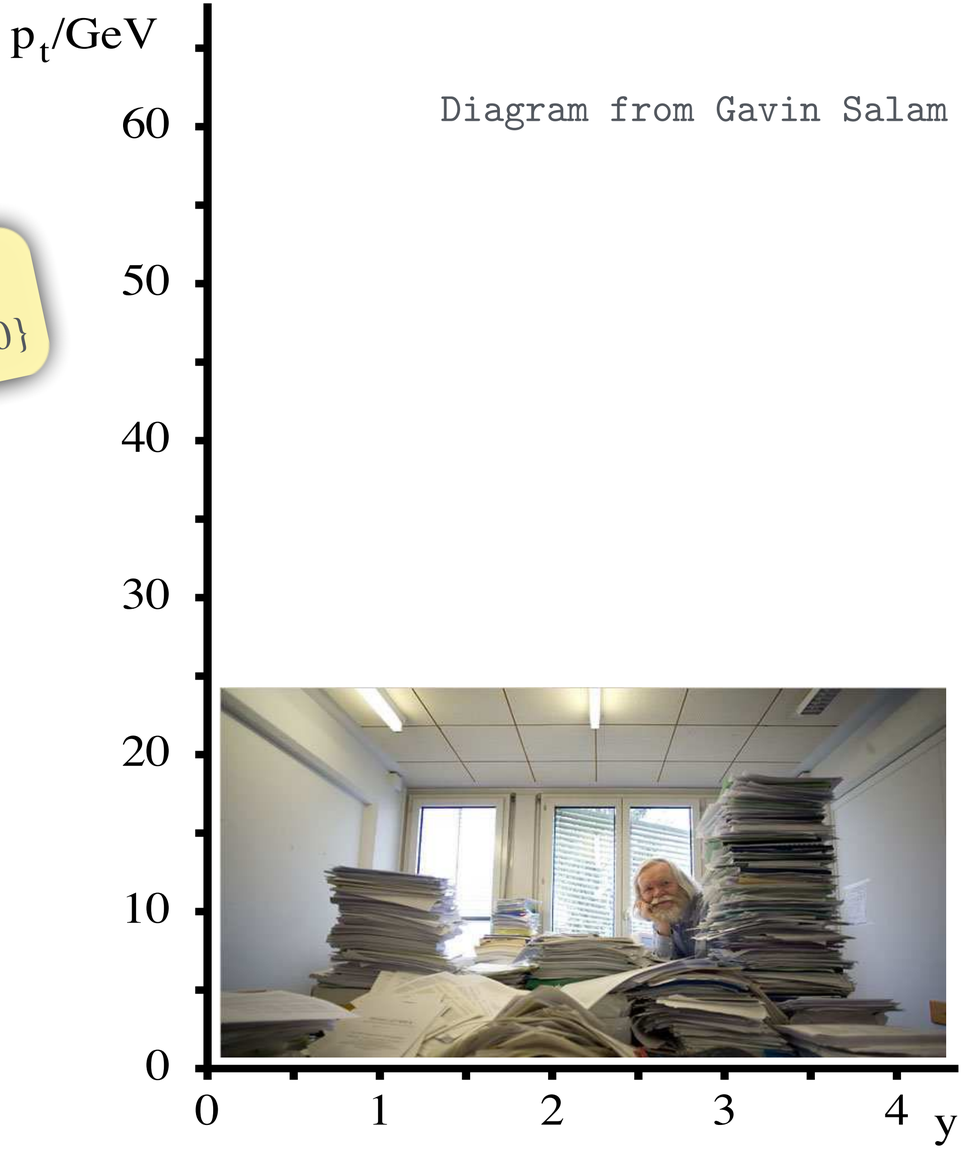
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Diagram from Gavin Salam



Jet Reconstruction

kT Algorithm

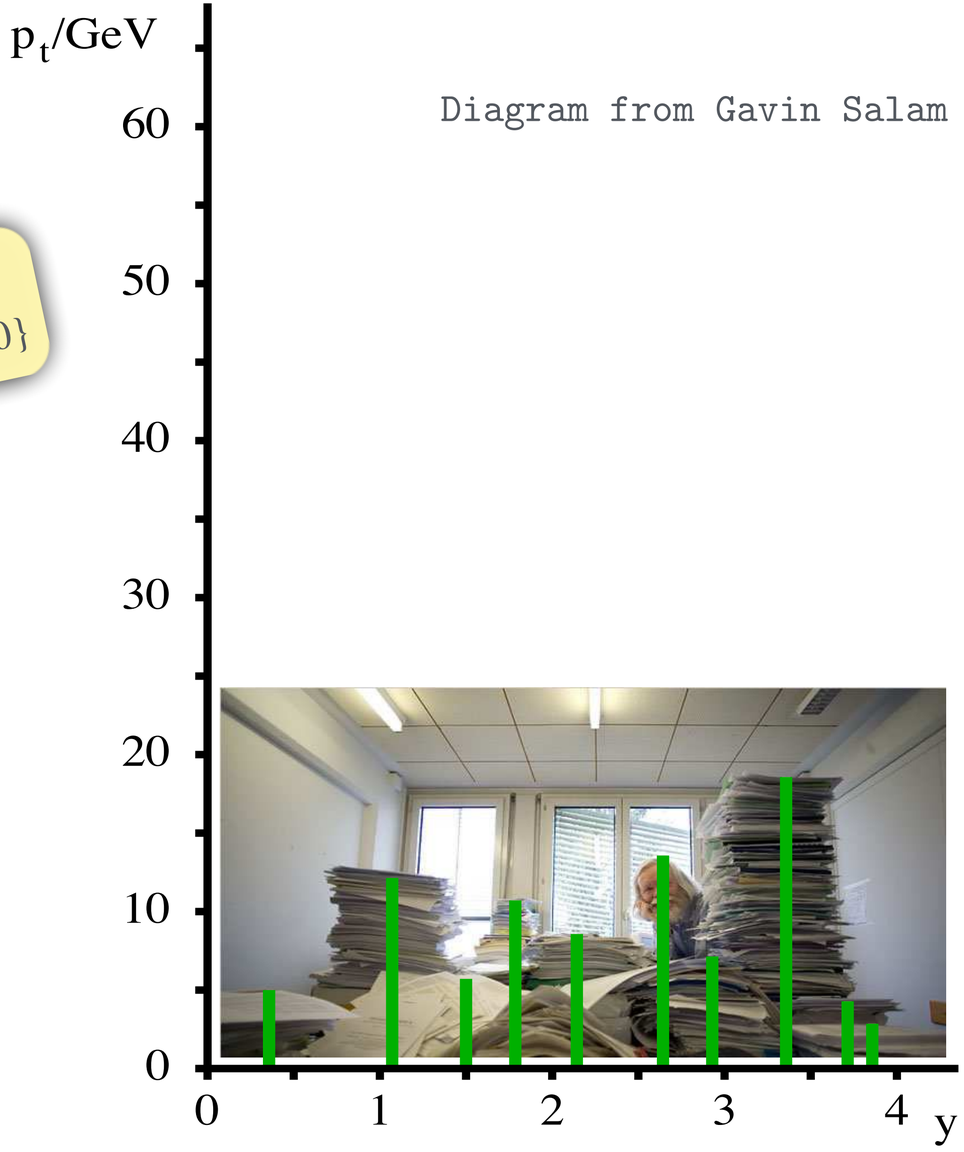
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Diagram from Gavin Salam



Jet Reconstruction

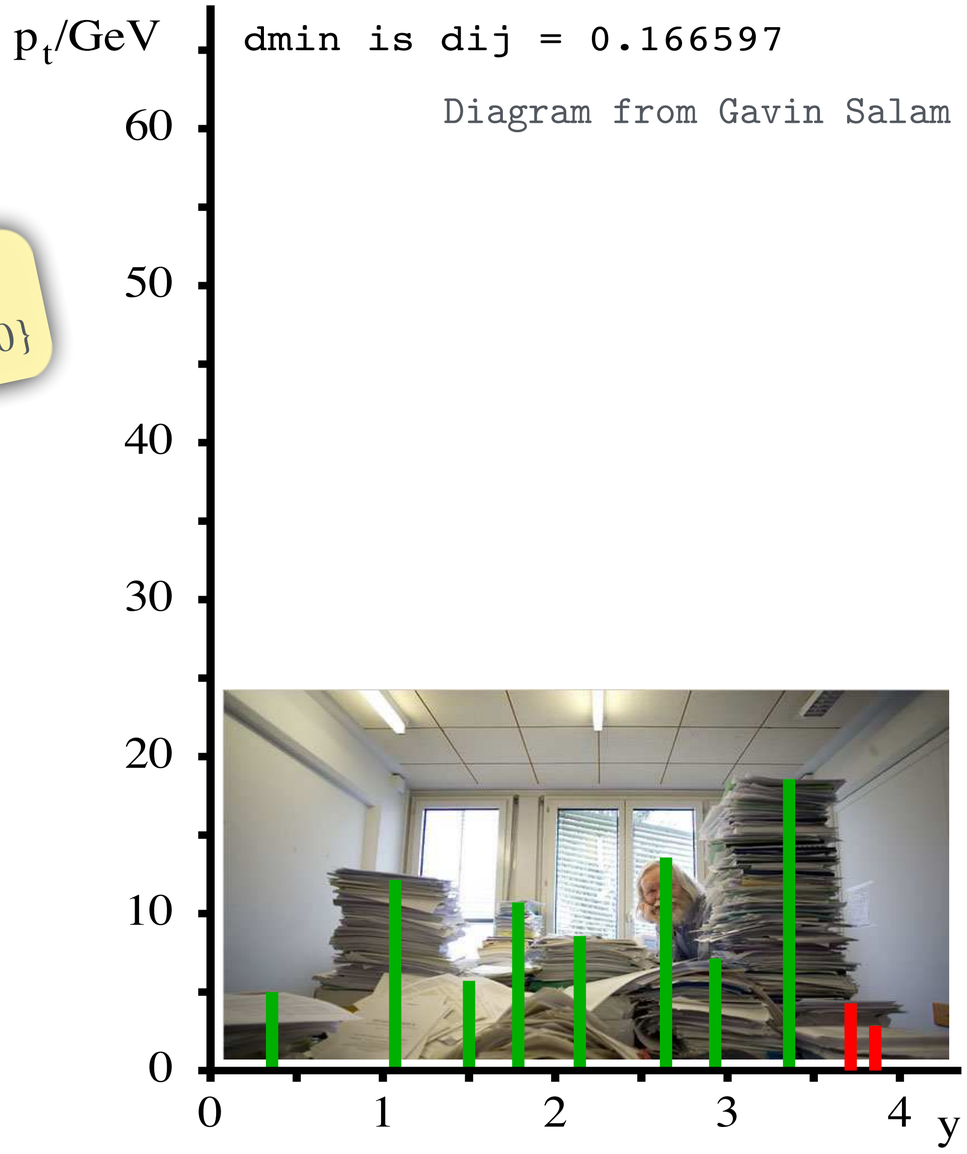
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Jet Reconstruction

kT Algorithm

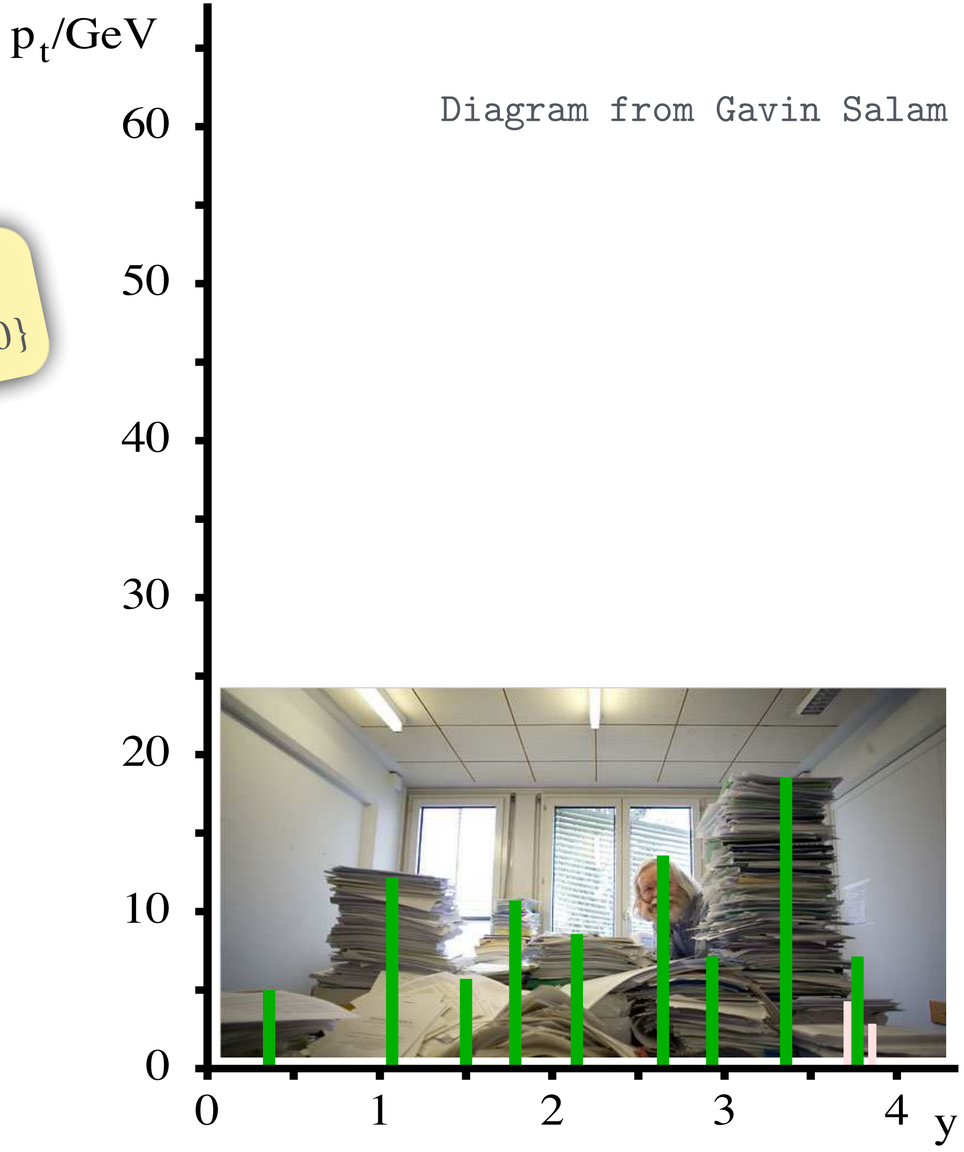
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Diagram from Gavin Salam



Jet Reconstruction

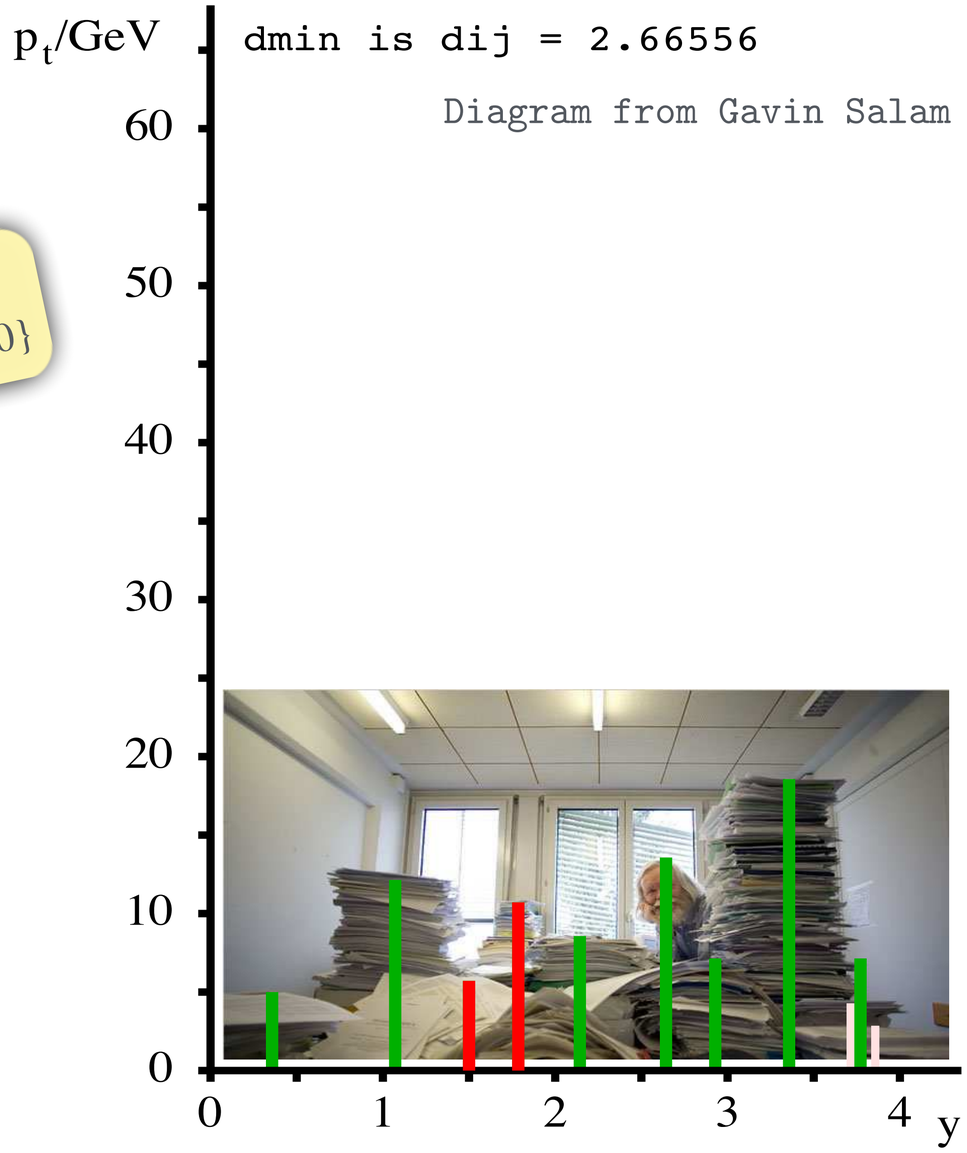
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Jet Reconstruction

kT Algorithm

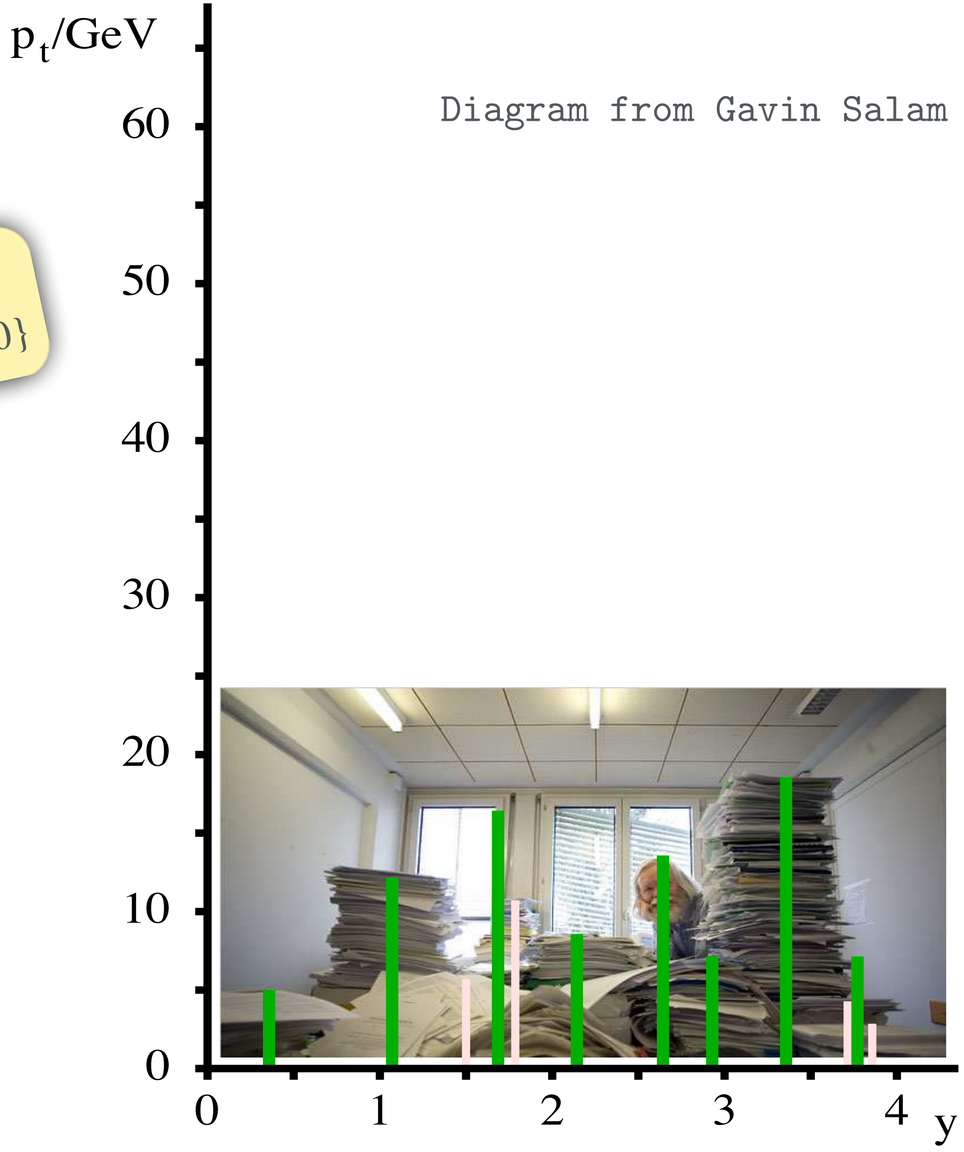
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Diagram from Gavin Salam



Jet Reconstruction

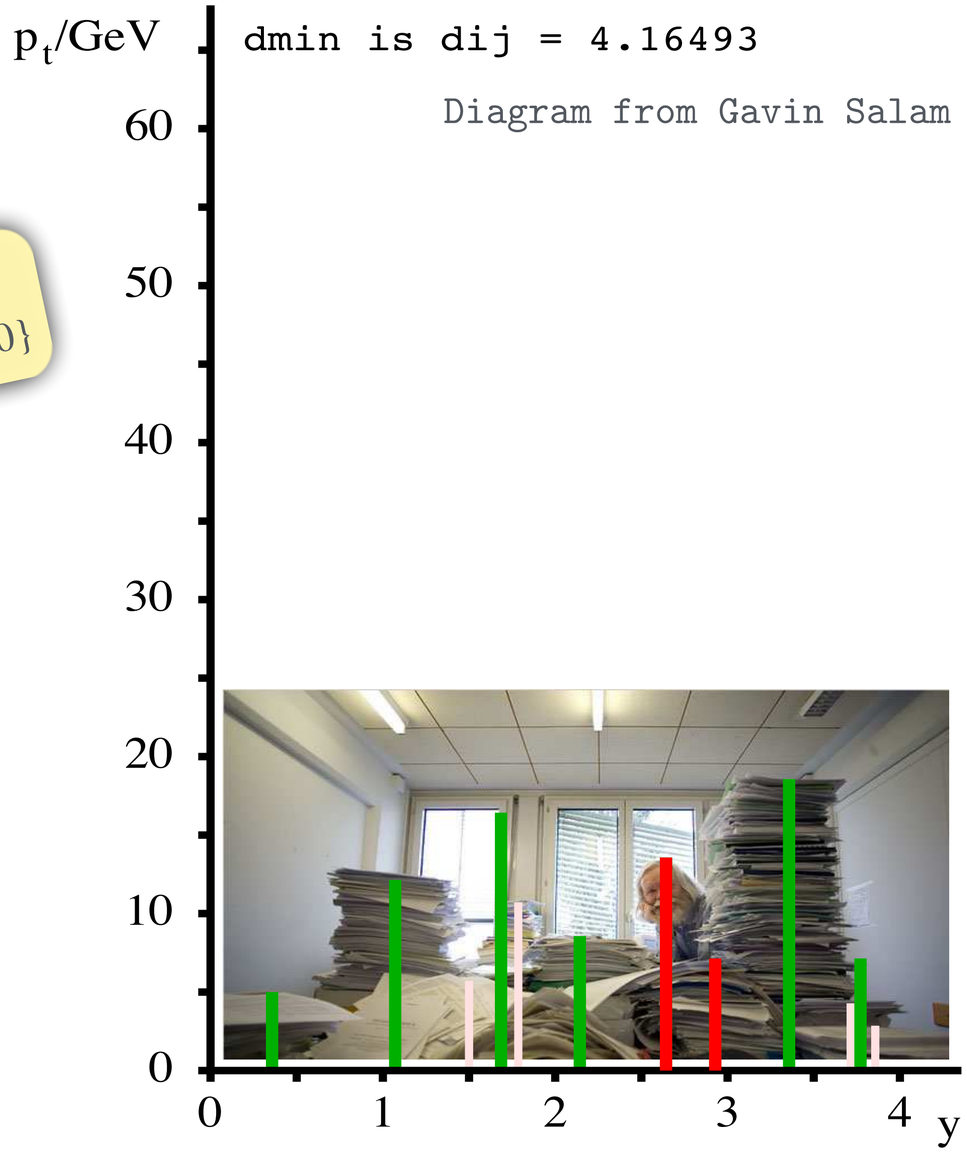
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Jet Reconstruction

kT Algorithm

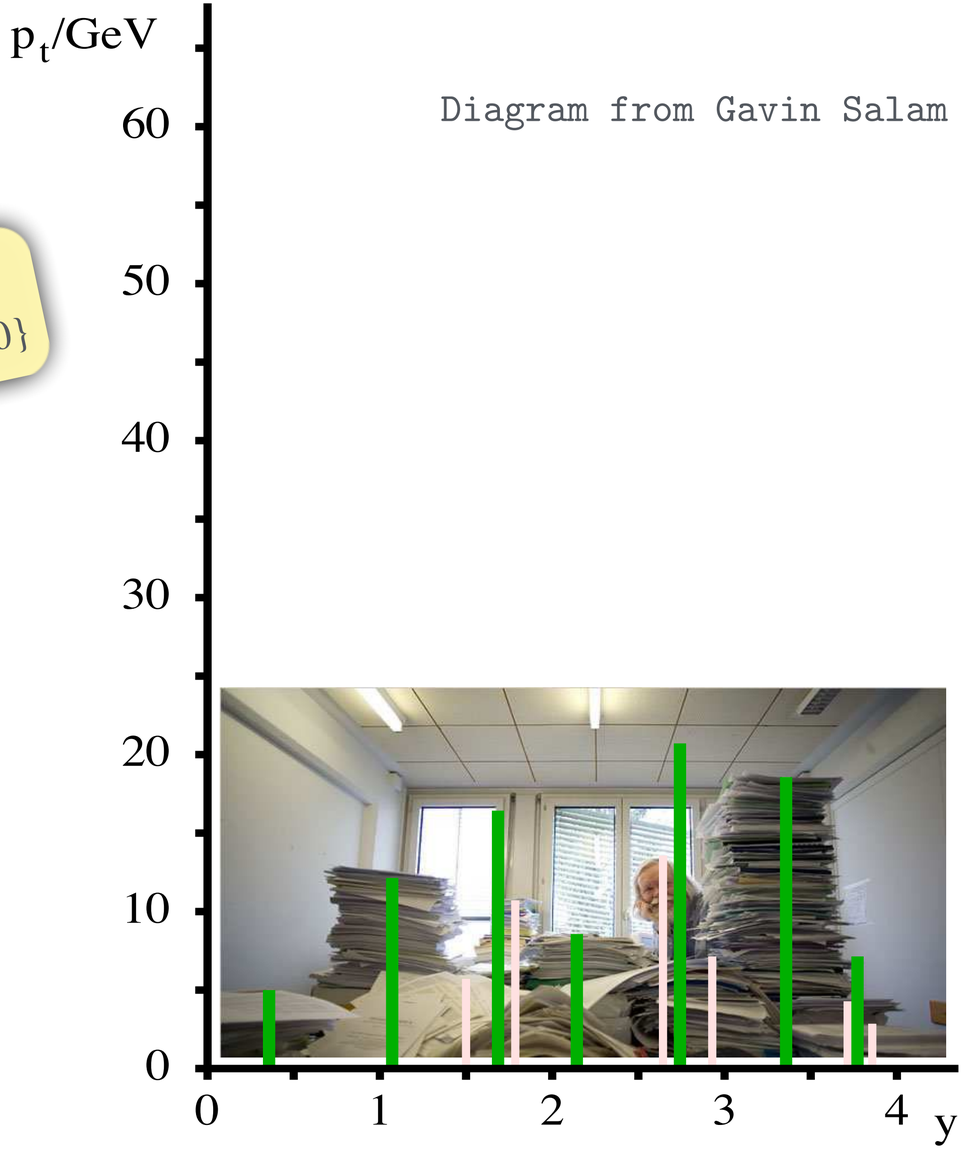
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Diagram from Gavin Salam



Jet Reconstruction

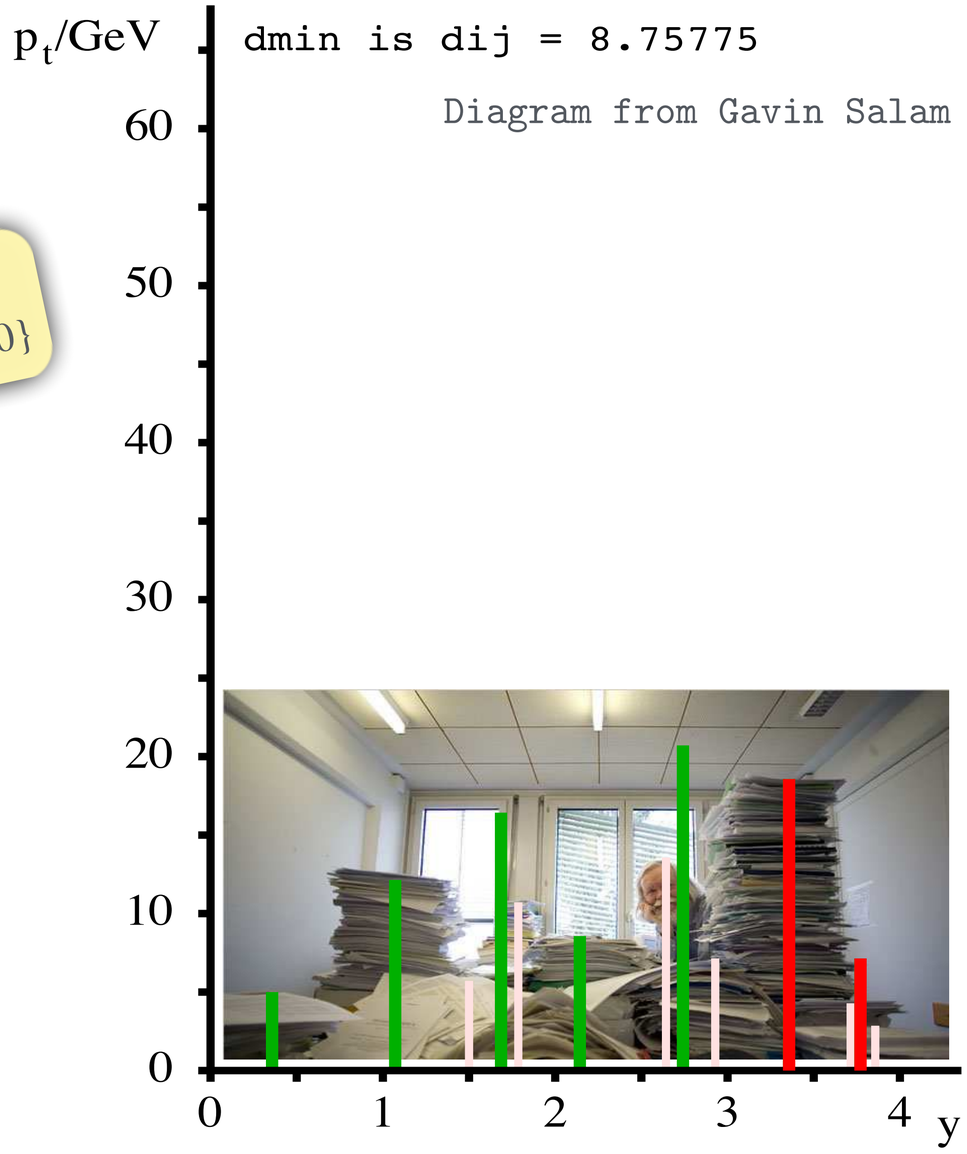
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Jet Reconstruction

kT Algorithm

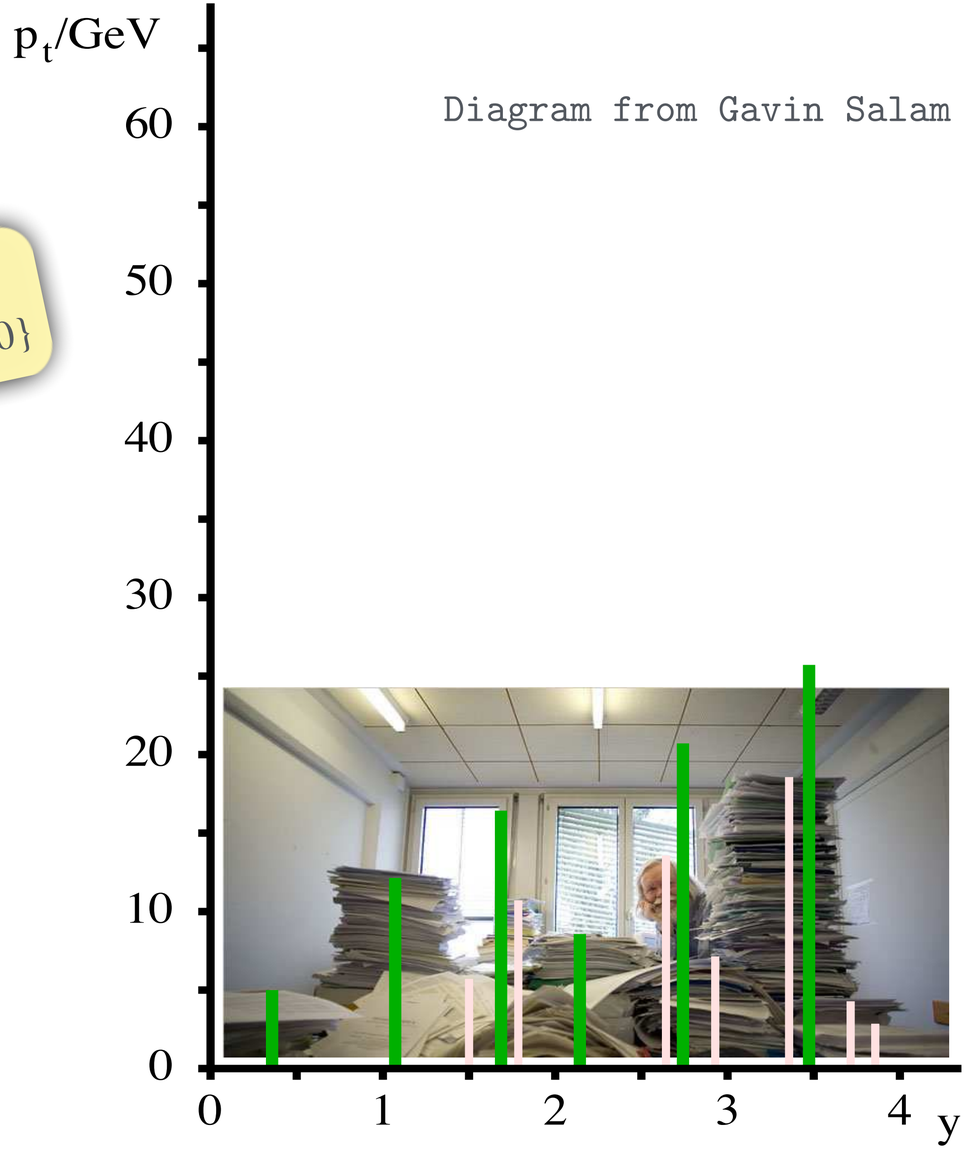
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Diagram from Gavin Salam



Jet Reconstruction

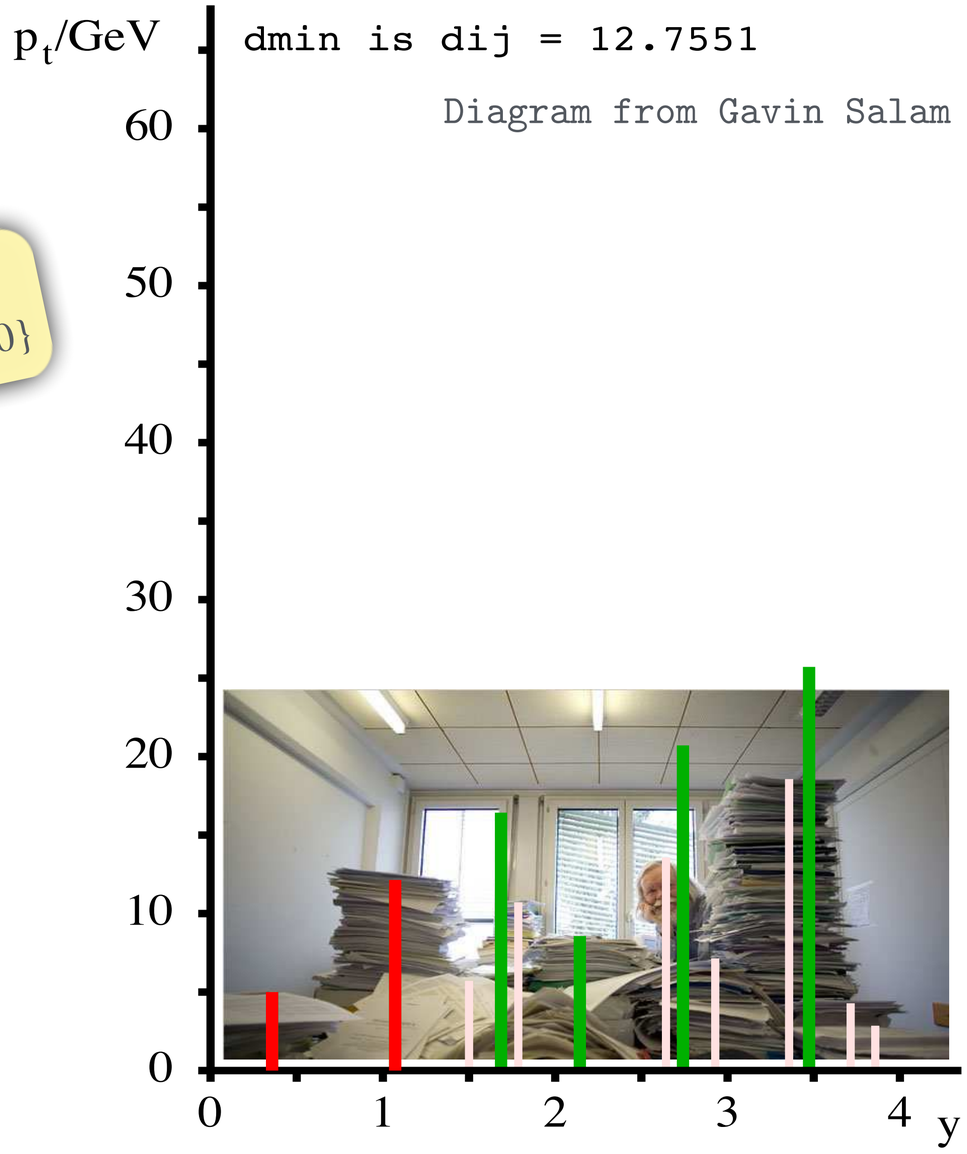
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Jet Reconstruction

kT Algorithm

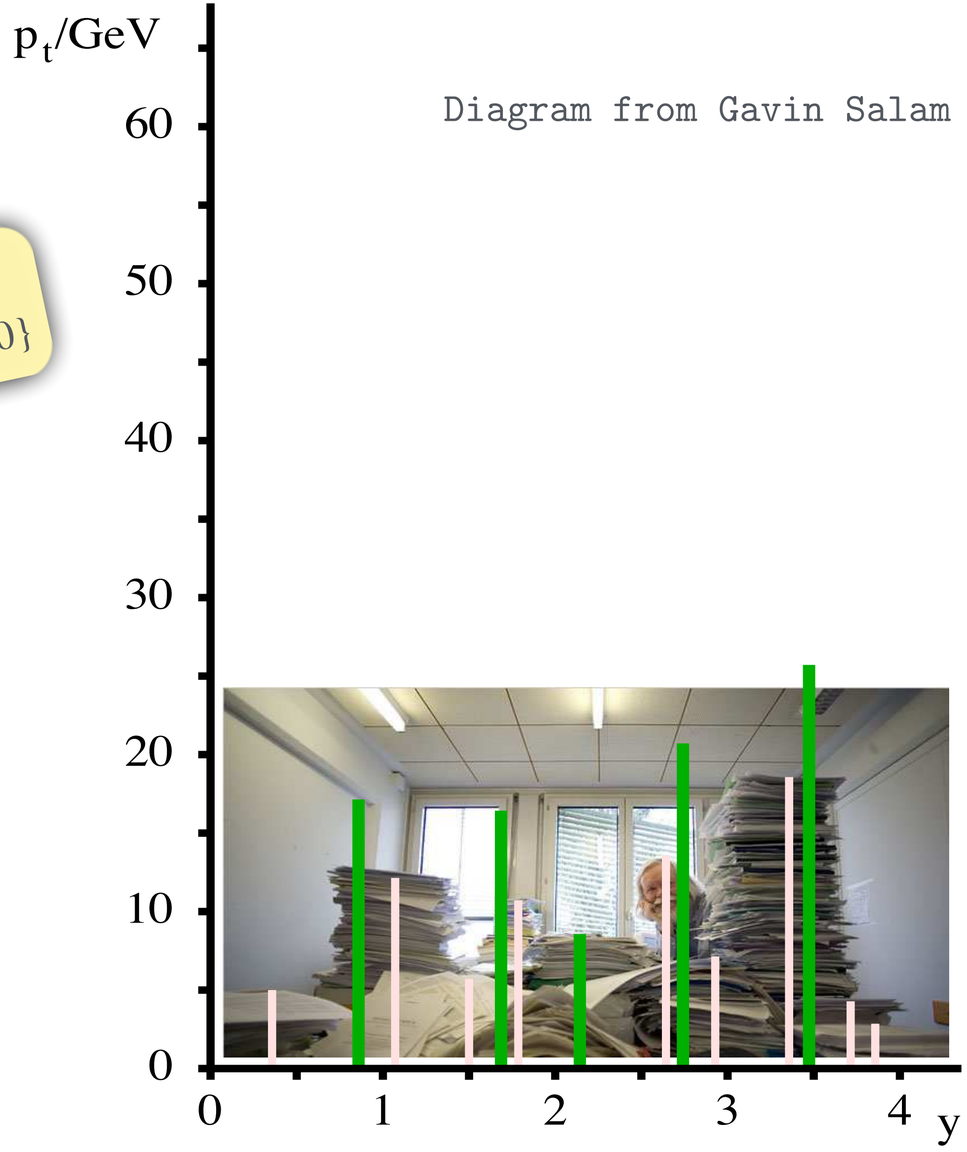
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Diagram from Gavin Salam



Jet Reconstruction

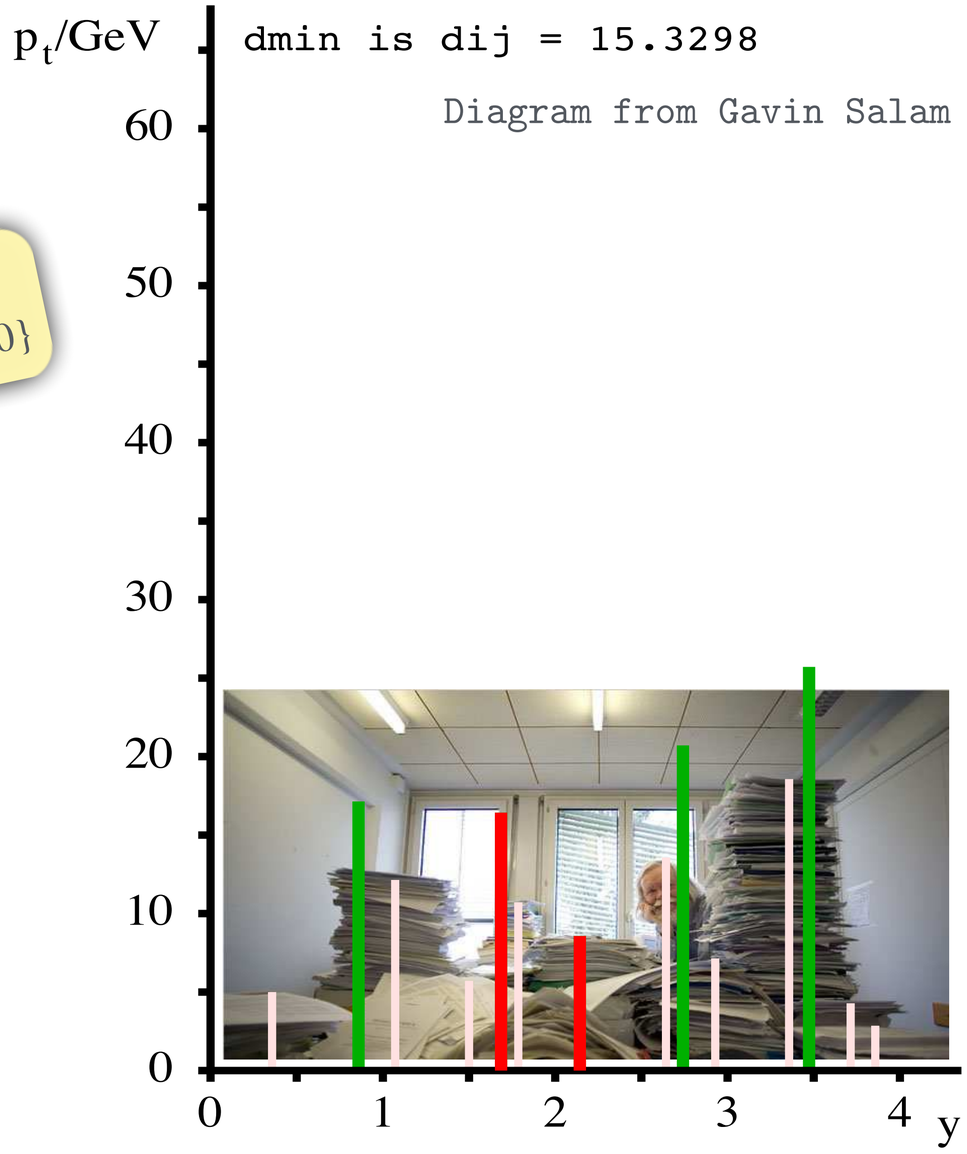
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Jet Reconstruction

kT Algorithm

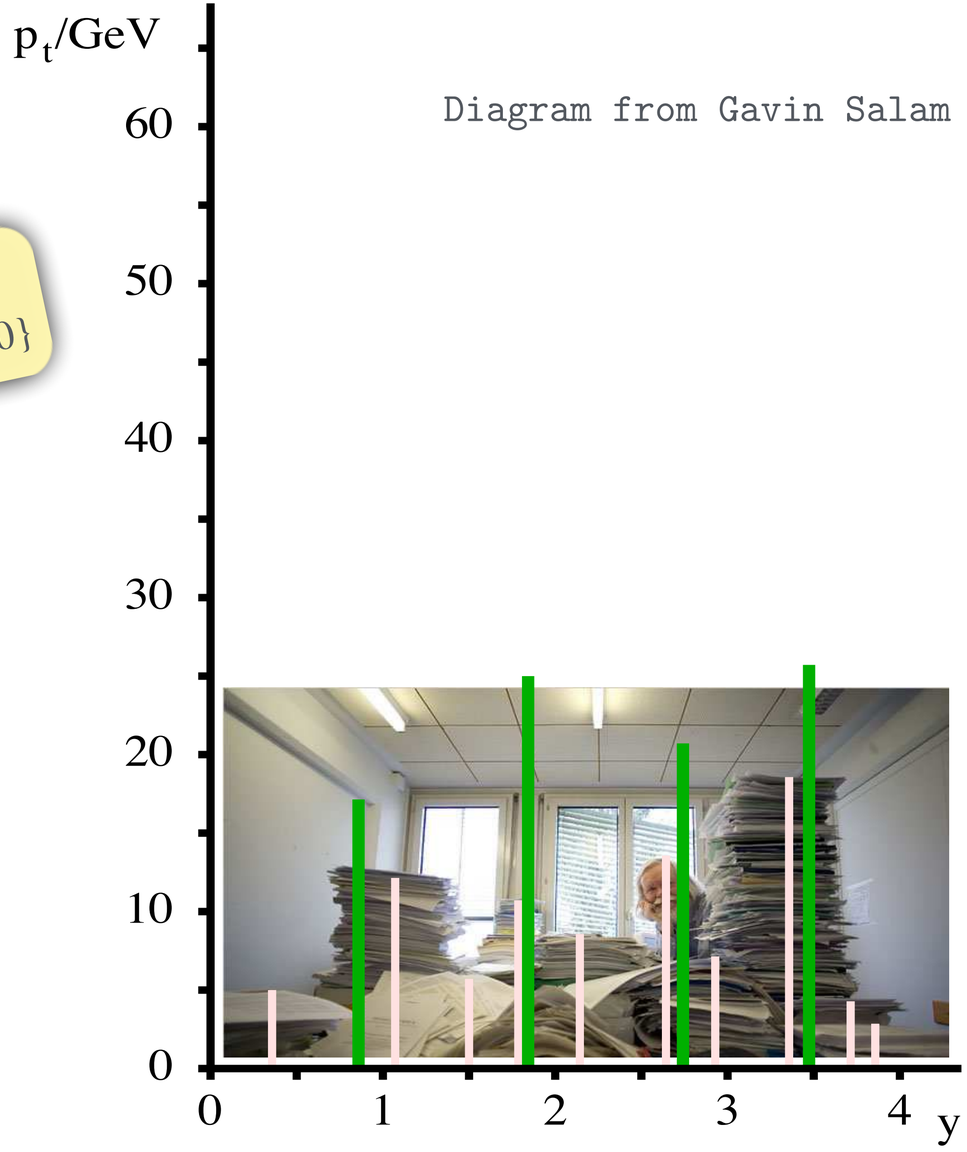
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Diagram from Gavin Salam



Jet Reconstruction

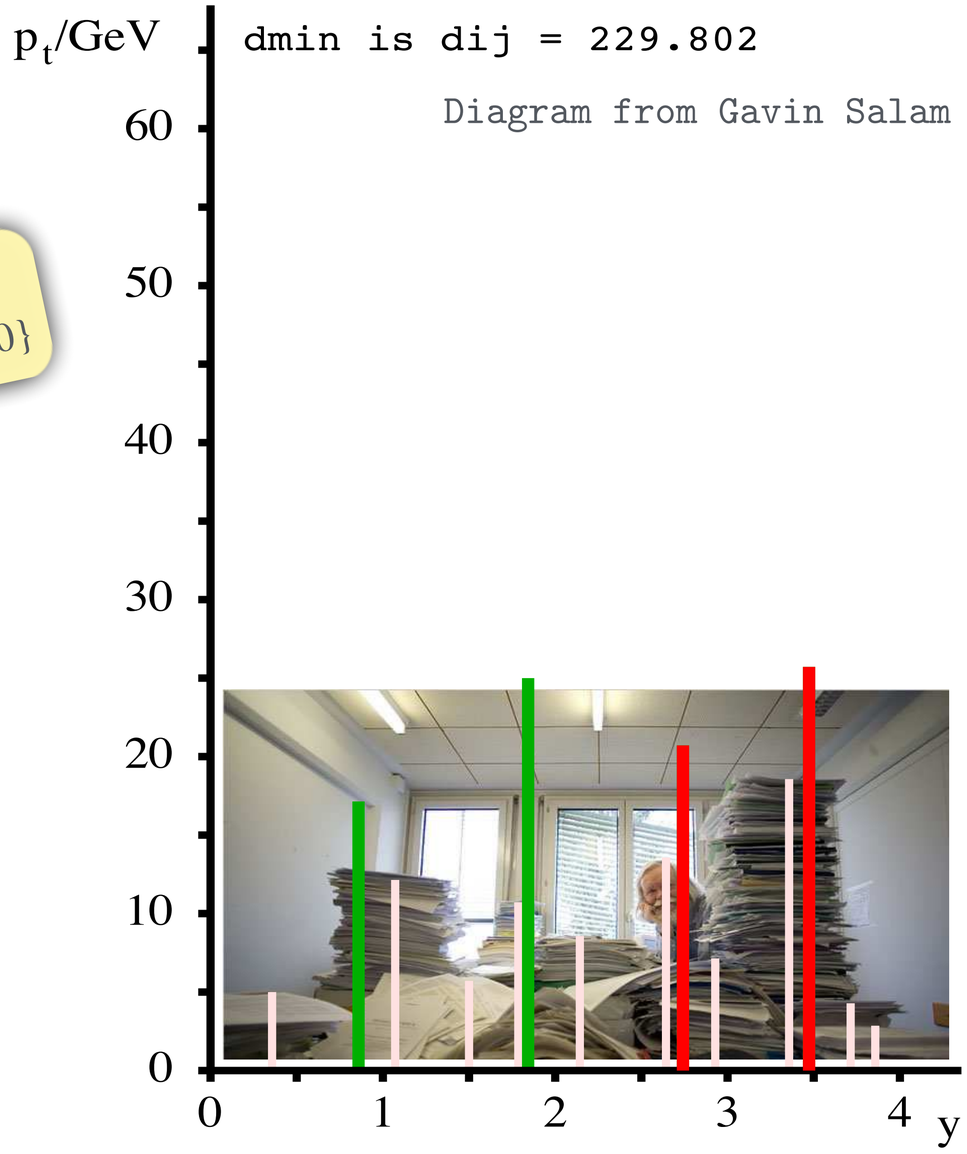
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Jet Reconstruction

kT Algorithm

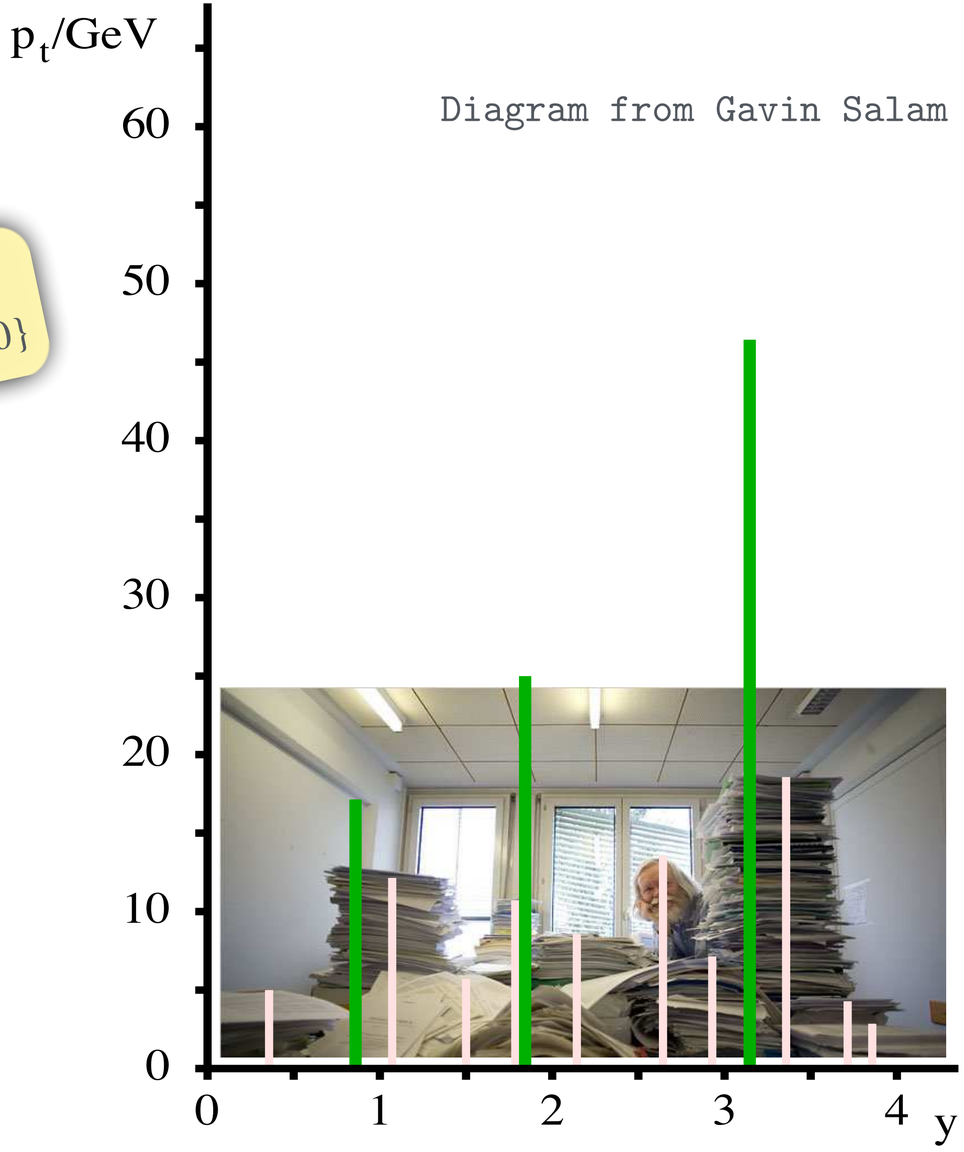
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Diagram from Gavin Salam



Jet Reconstruction

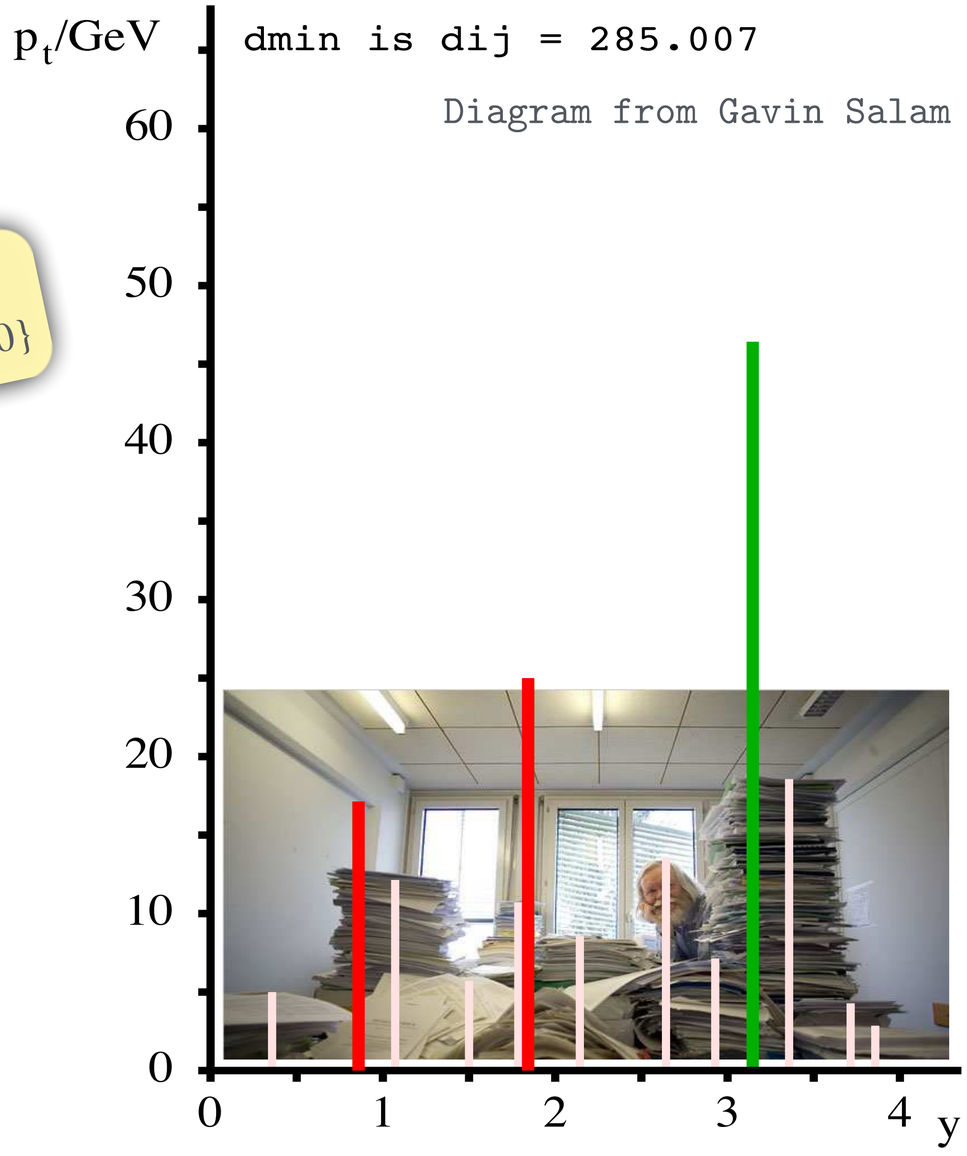
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Jet Reconstruction

kT Algorithm

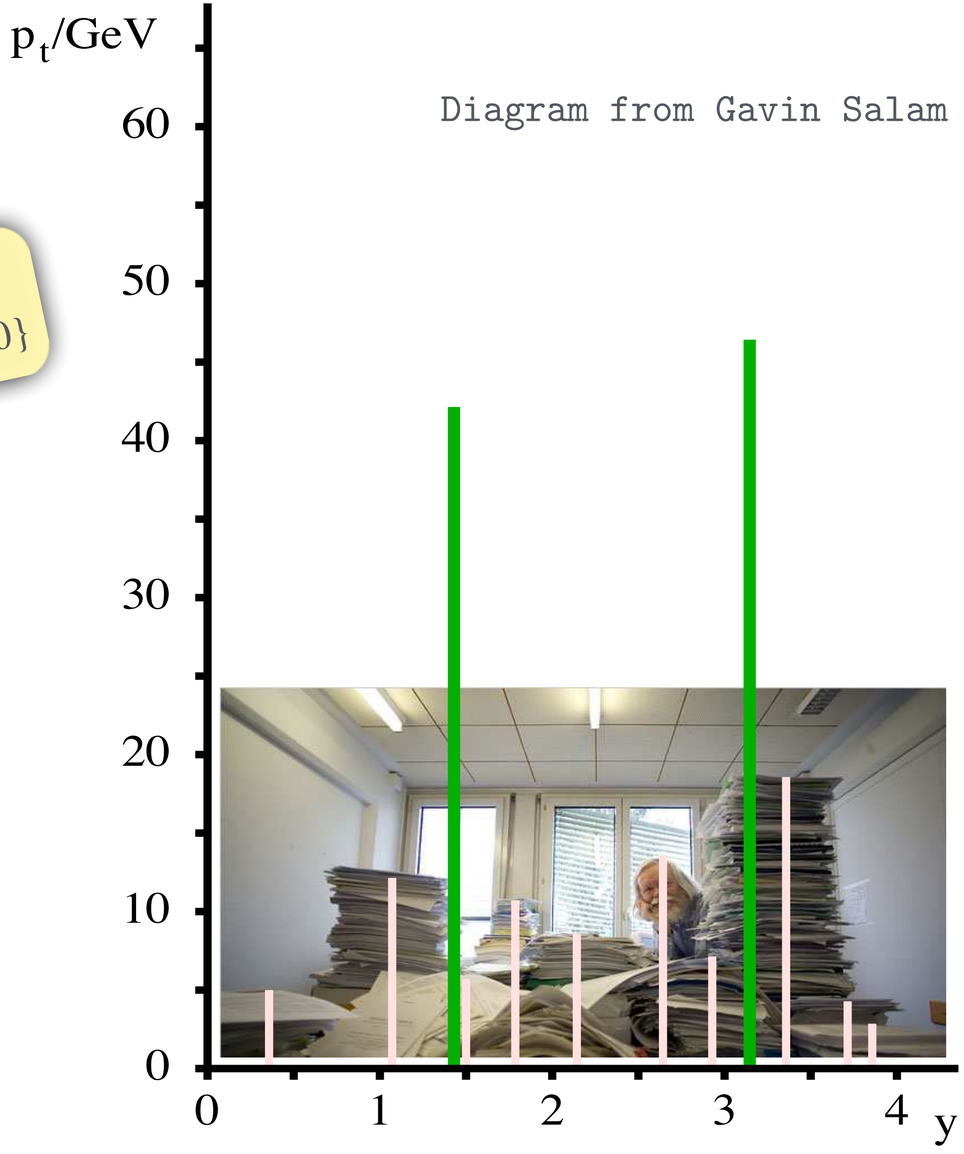
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Diagram from Gavin Salam



Jet Reconstruction

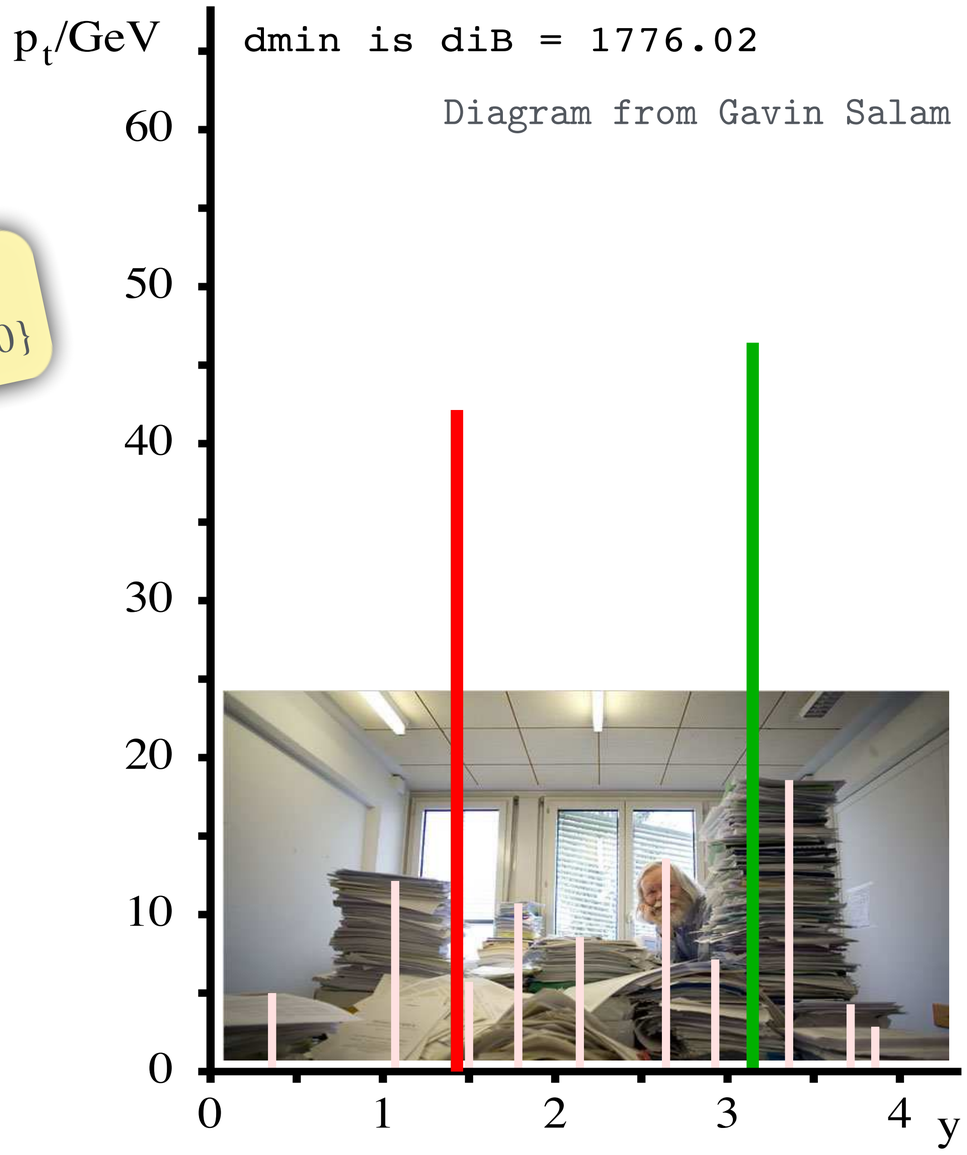
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Jet Reconstruction

kT Algorithm

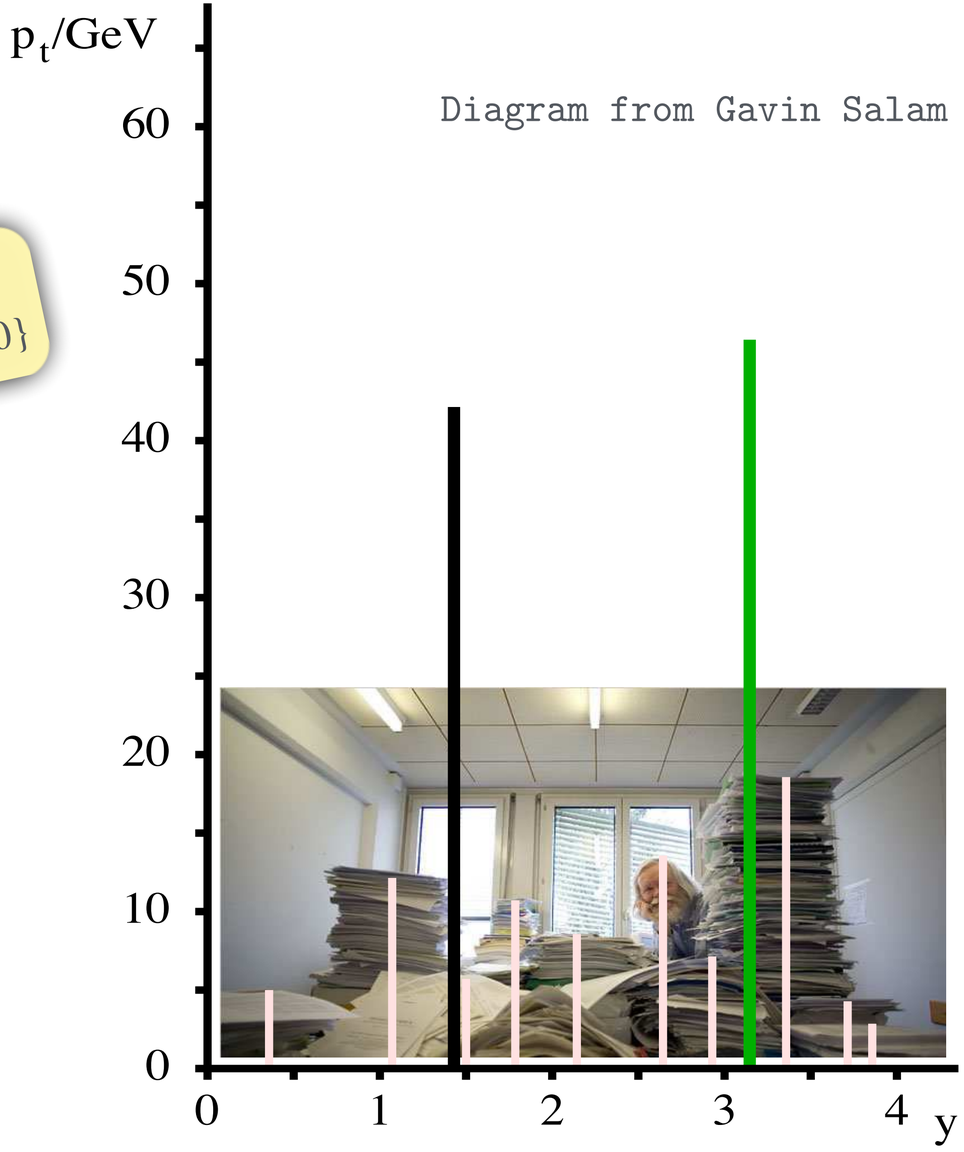
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Diagram from Gavin Salam



Jet Reconstruction

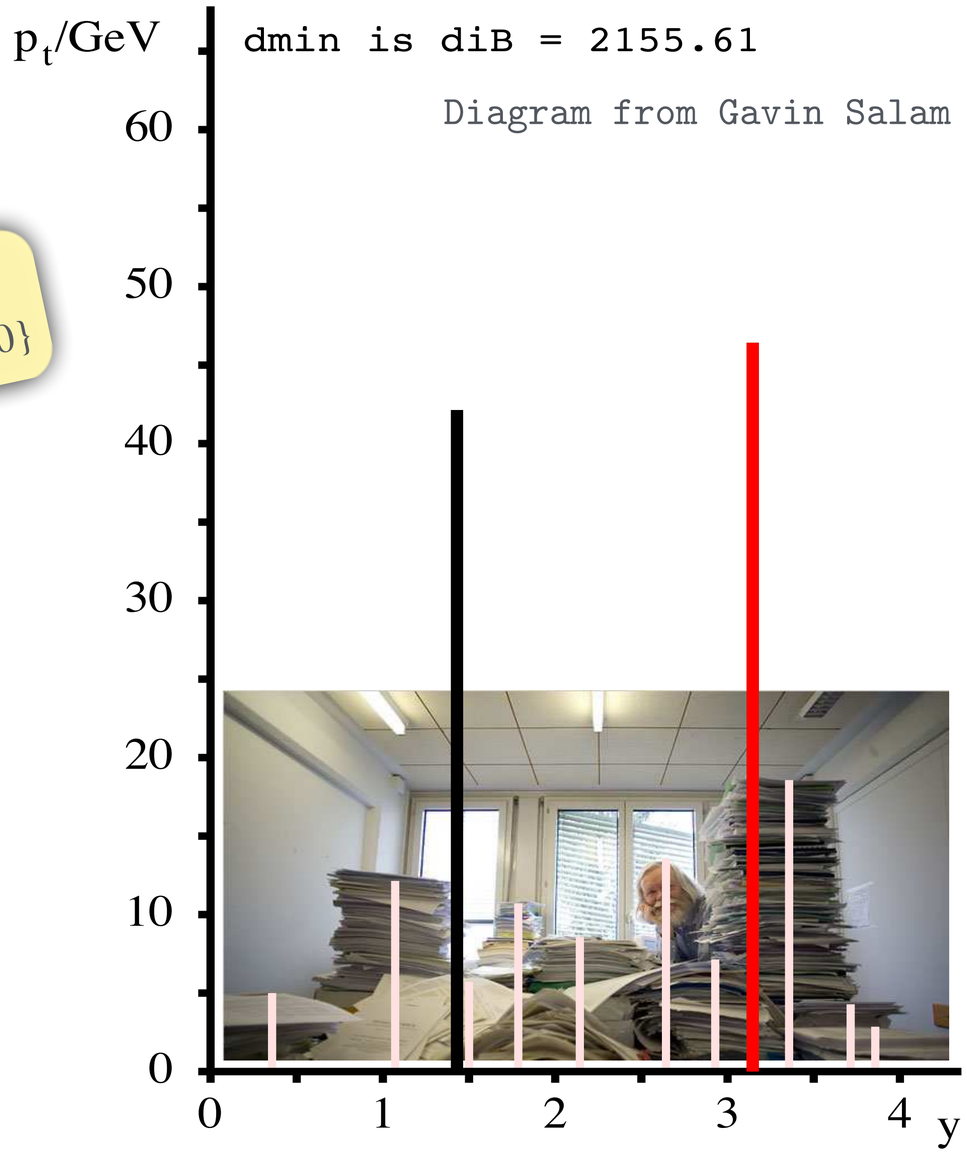
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Jet Reconstruction

kT Algorithm

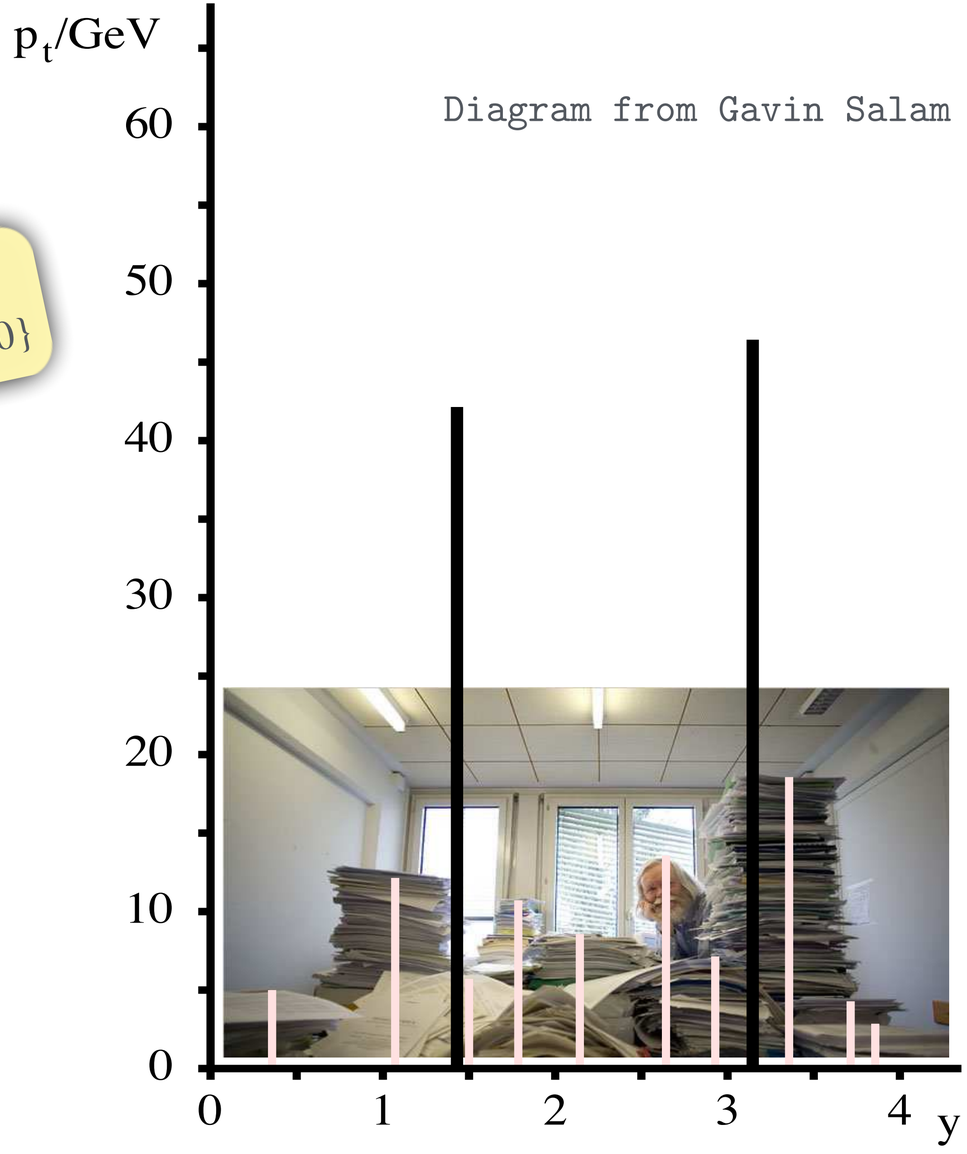
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Diagram from Gavin Salam



Jet Reconstruction

kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Jet Reconstruction

kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Diagrams from Gavin Salam

Jet Reconstruction

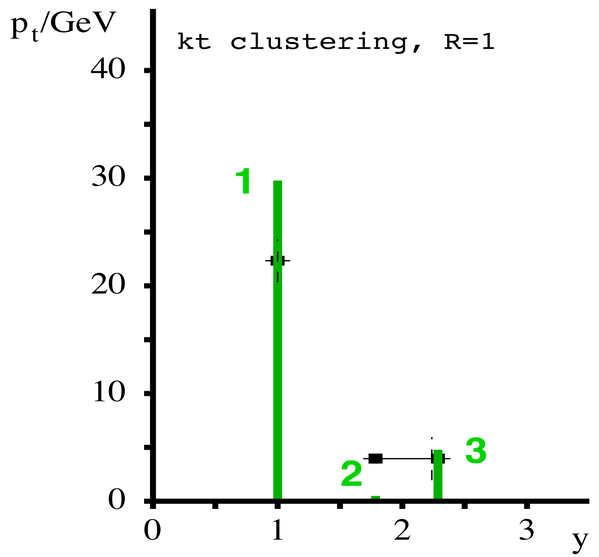
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Diagrams from Gavin Salam

Jet Reconstruction

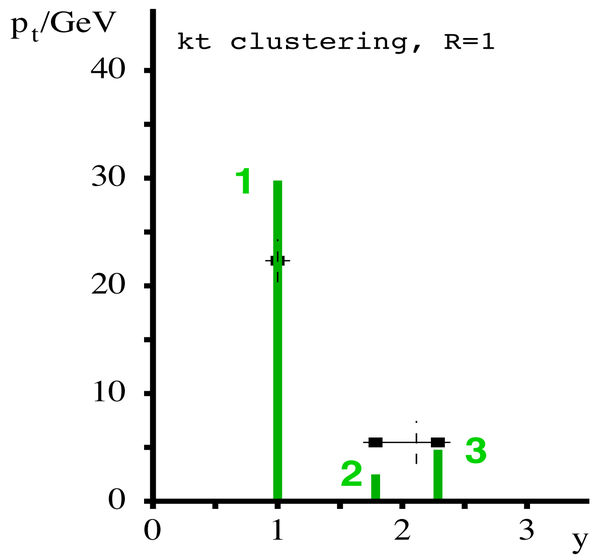
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Diagrams from Gavin Salam

Jet Reconstruction

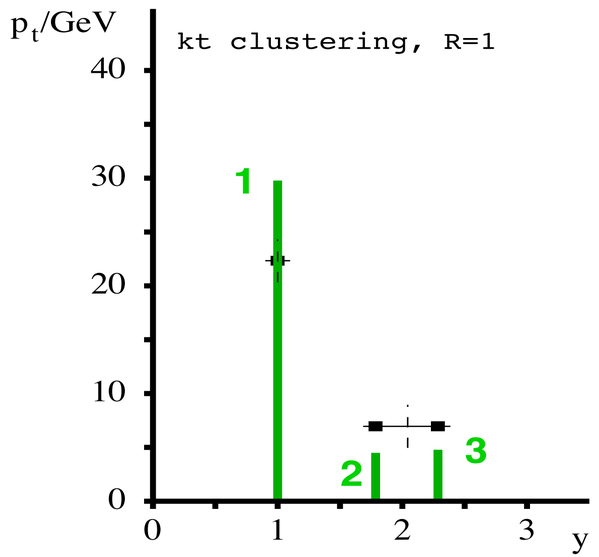
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Diagrams from Gavin Salam

Jet Reconstruction

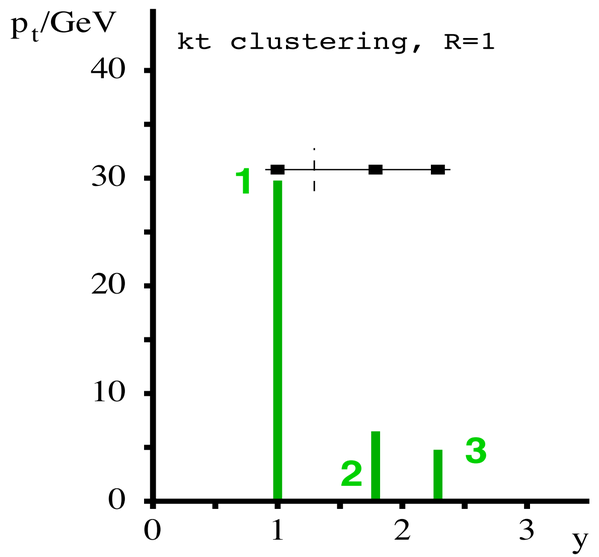
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Diagrams from Gavin Salam

Jet Reconstruction

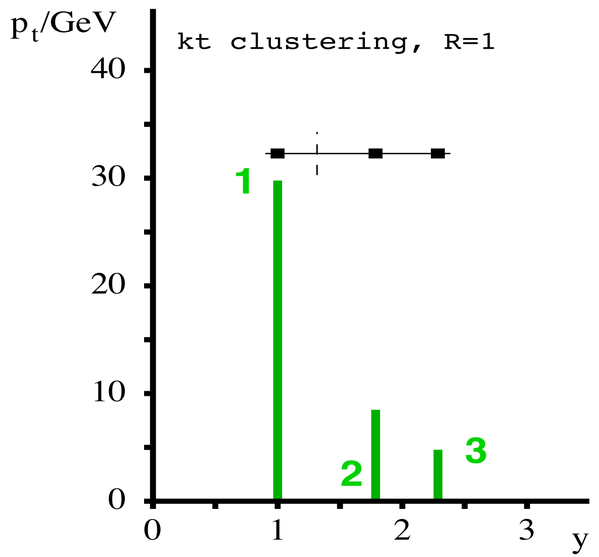
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Diagrams from Gavin Salam

Jet Reconstruction

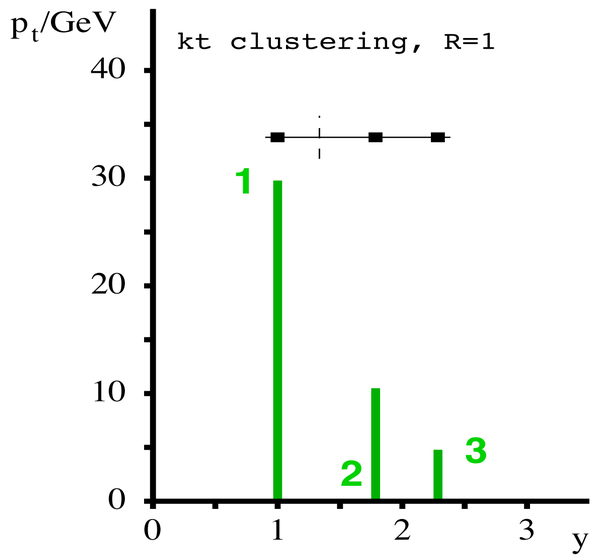
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Diagrams from Gavin Salam

Jet Reconstruction

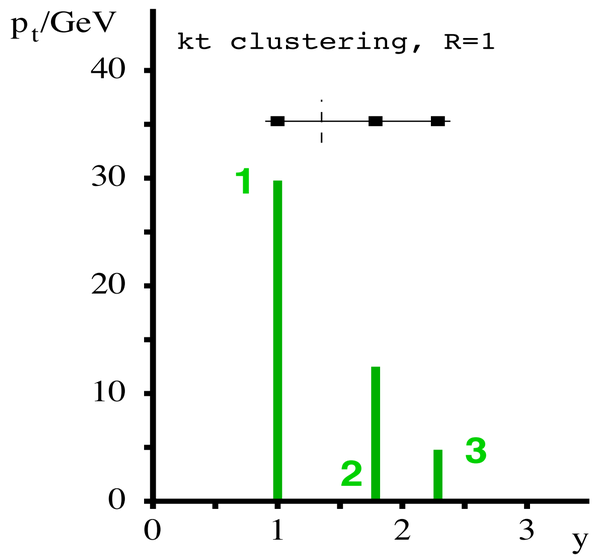
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Diagrams from Gavin Salam

Jet Reconstruction

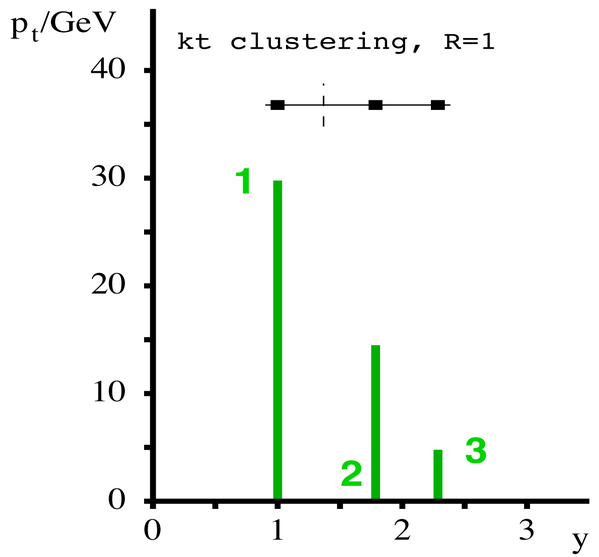
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Diagrams from Gavin Salam

Jet Reconstruction

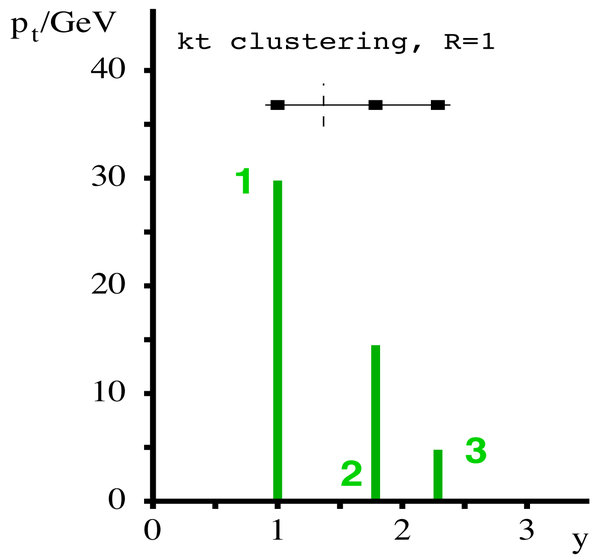
kT Algorithm

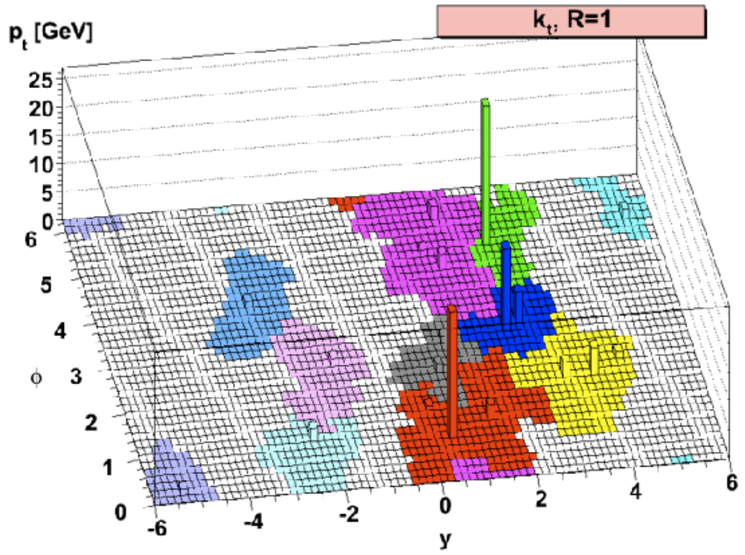
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}





Diagrams from Gavin Salam

Jet Reconstruction

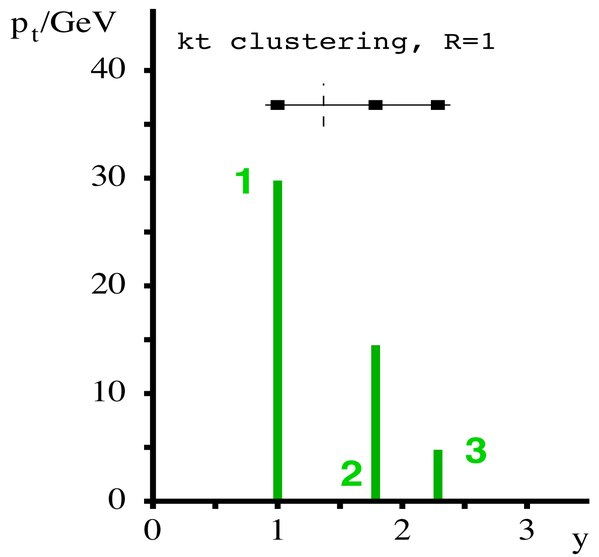
kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Diagrams from Gavin Salam

Jet Reconstruction

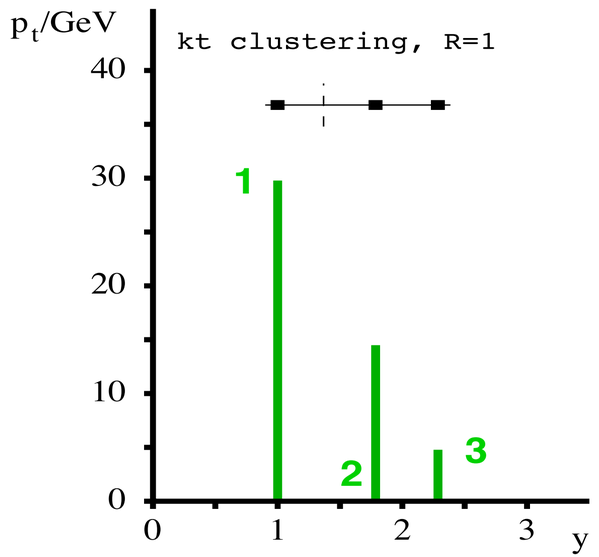
anti-kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^2, k_{\mathrm{T},j}^2)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Diagrams from Gavin Salam

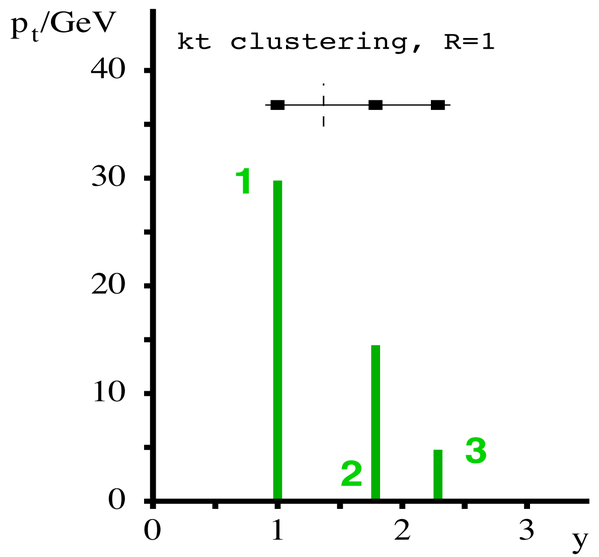
Jet Reconstruction anti-kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Diagrams from Gavin Salam

Jet Reconstruction anti-kT Algorithm

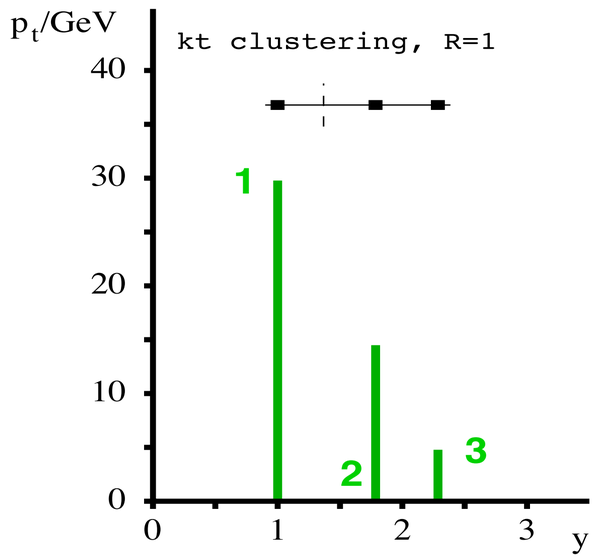
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = -1$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}



Diagrams from Gavin Salam

Jet Reconstruction

anti-kT Algorithm

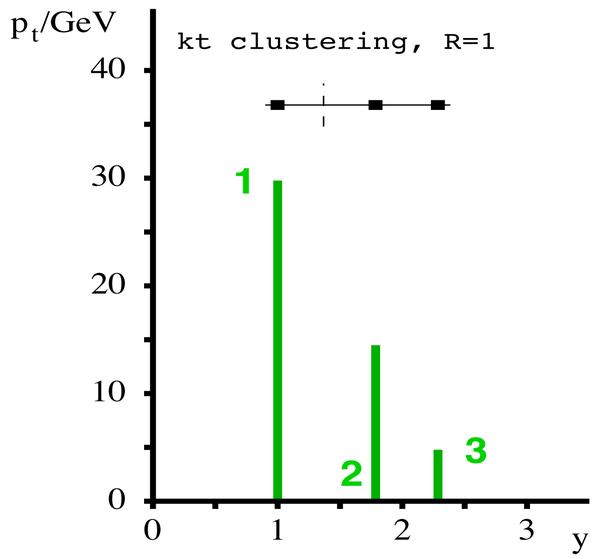
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

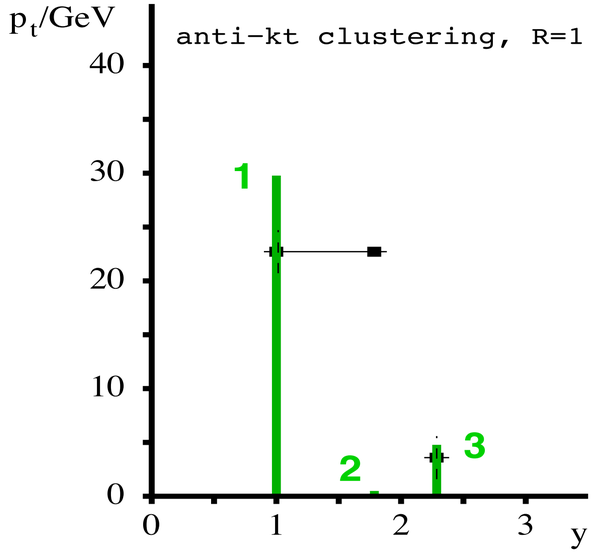
$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = -1$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}





Diagrams from Gavin Salam

Jet Reconstruction

anti-kT Algorithm

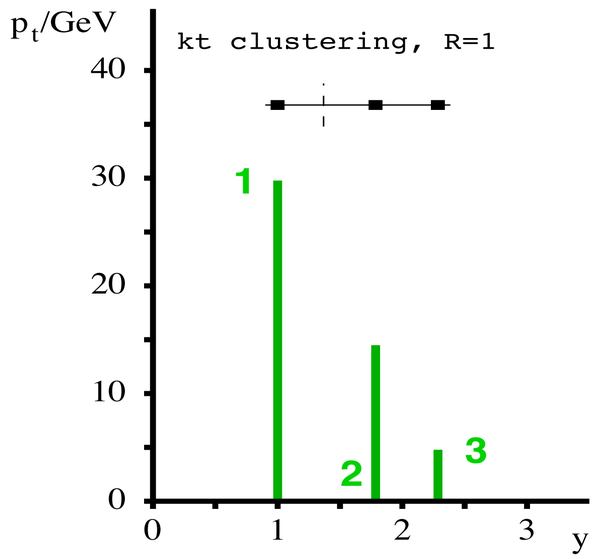
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

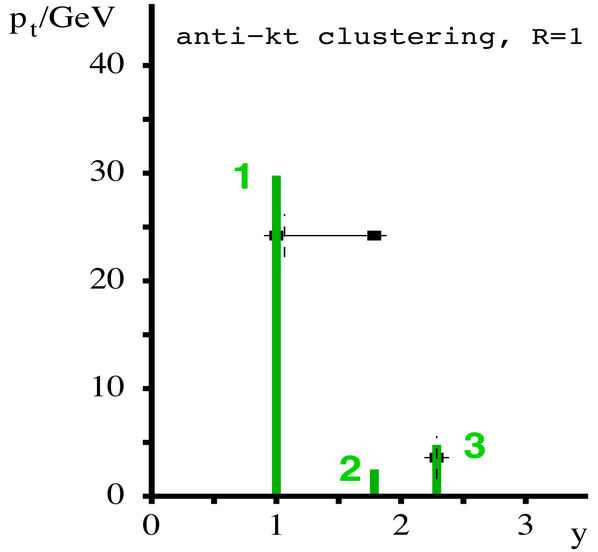
$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = -1$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}





Diagrams from Gavin Salam

Jet Reconstruction

anti-kT Algorithm

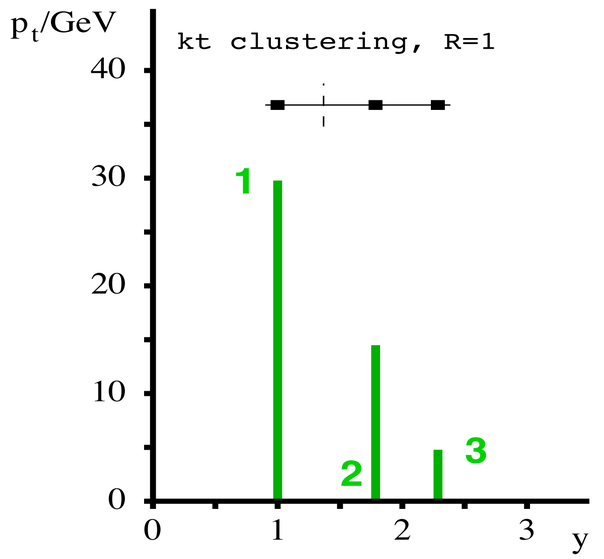
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

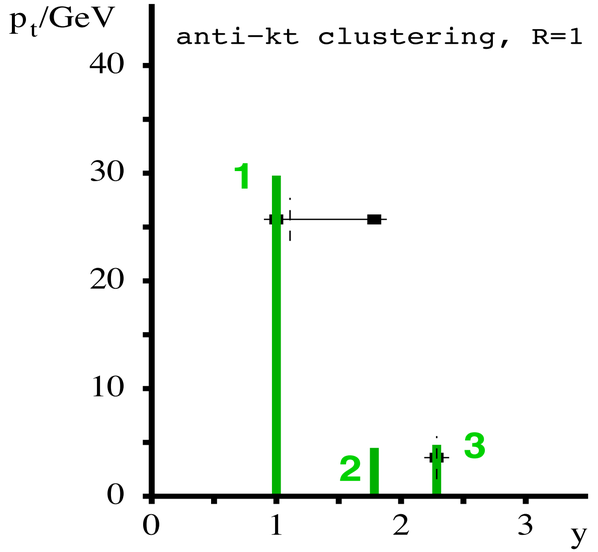
$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = -1$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}





Diagrams from Gavin Salam

Jet Reconstruction

anti-kT Algorithm

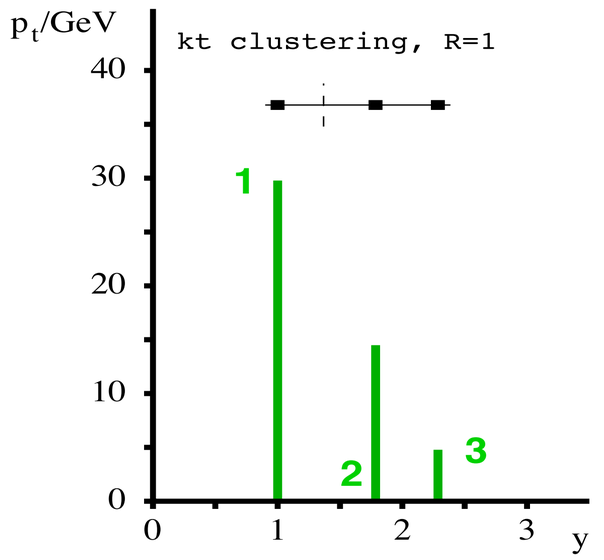
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

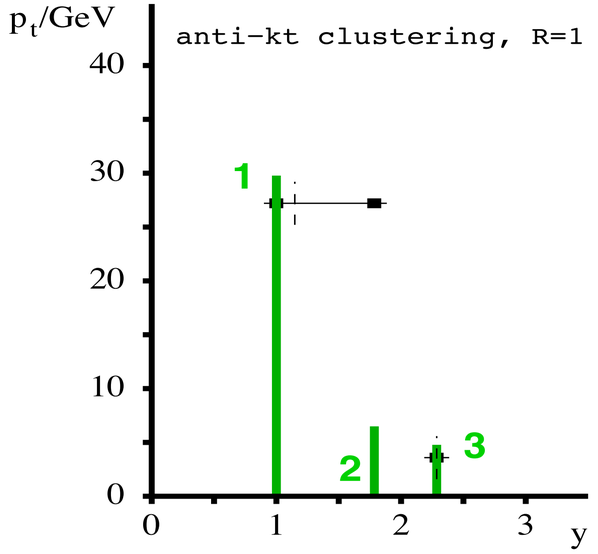
$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = -1$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}





Diagrams from Gavin Salam

Jet Reconstruction

anti-kT Algorithm

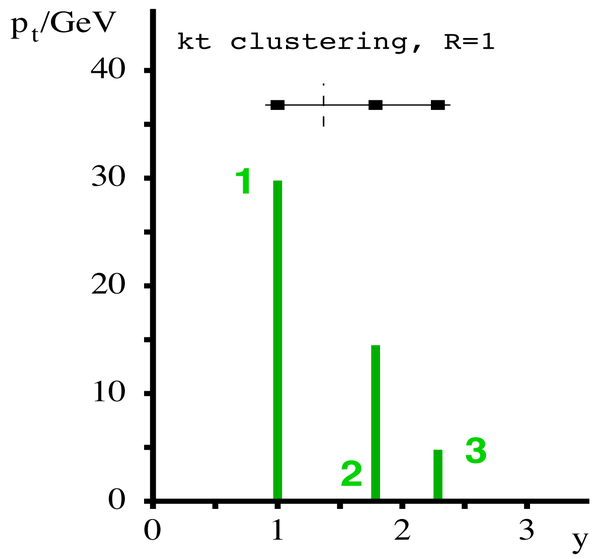
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

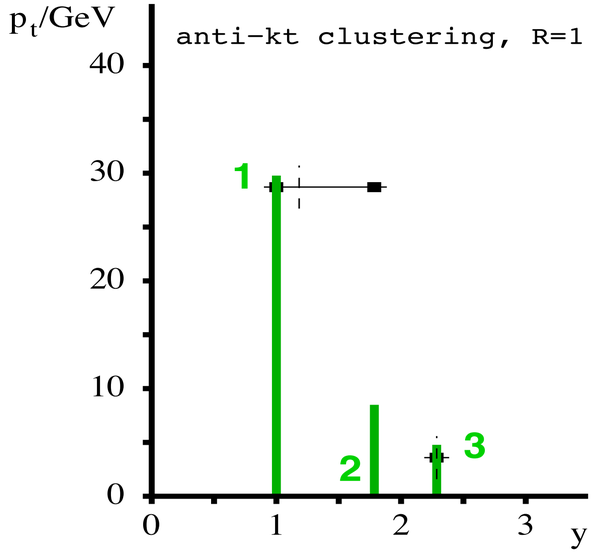
$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = -1$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}





Diagrams from Gavin Salam

Jet Reconstruction

anti-kT Algorithm

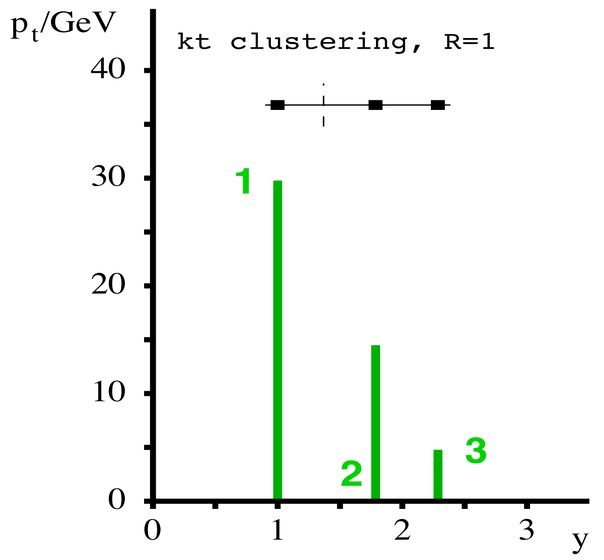
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

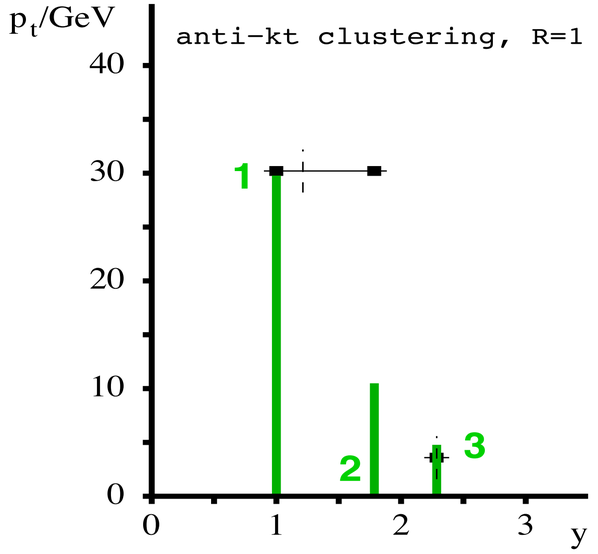
$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = -1$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}





Diagrams from Gavin Salam

Jet Reconstruction

anti-kT Algorithm

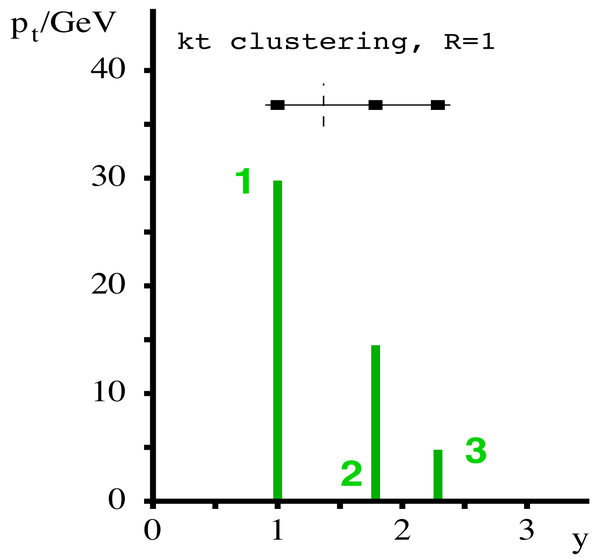
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

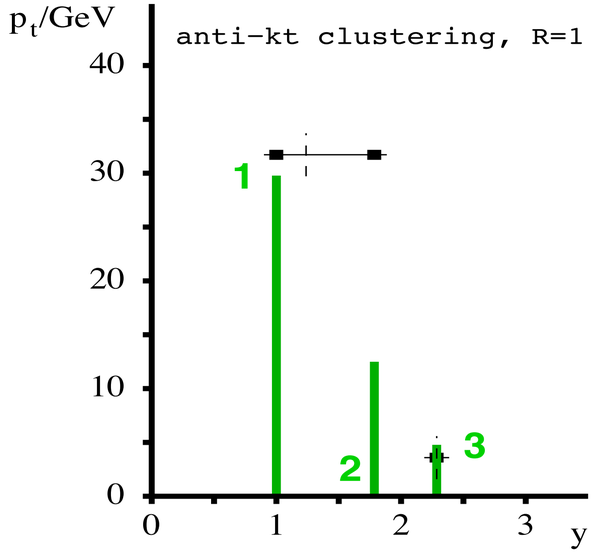
$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = -1$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}





Diagrams from Gavin Salam

Jet Reconstruction

anti-kT Algorithm

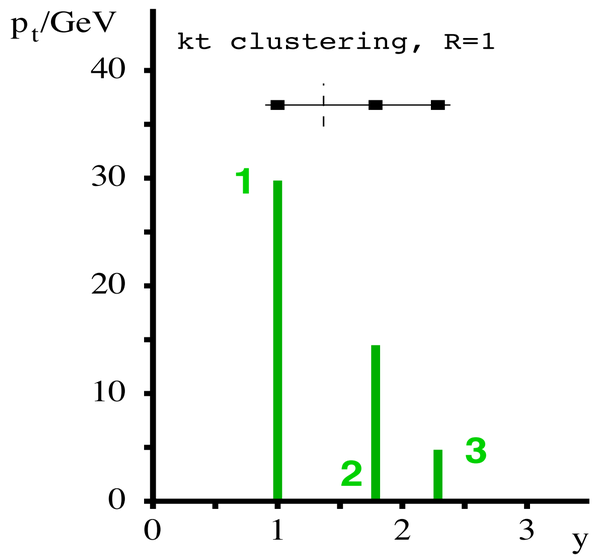
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

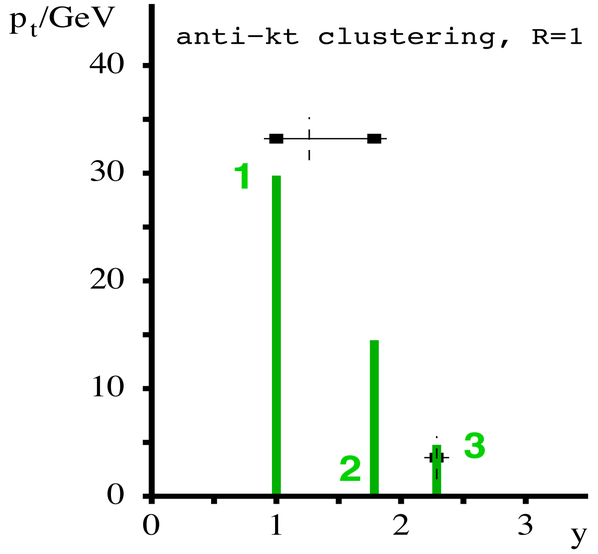
$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = -1$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}





Diagrams from Gavin Salam

Jet Reconstruction anti-kT Algorithm

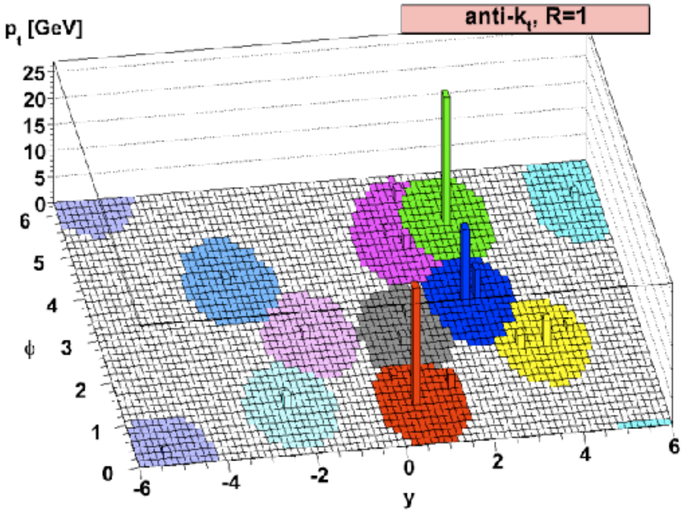
- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

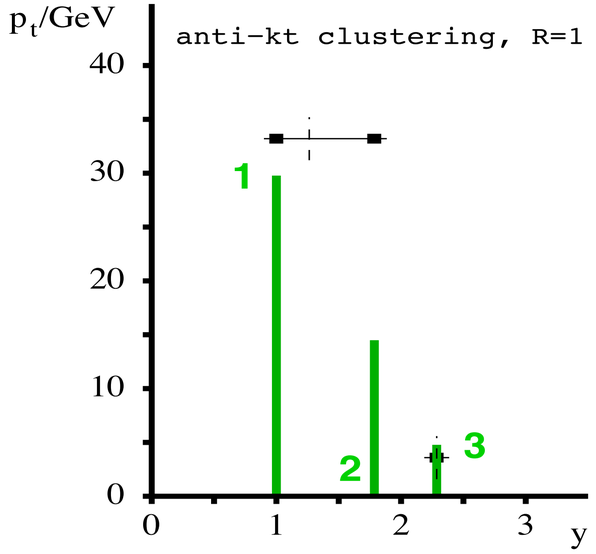
$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = -1$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}





Diagrams from Gavin Salam

Jet Reconstruction anti-kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = -1$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Jet Reconstruction anti-kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Jet Reconstruction anti-kT Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = 0$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Jet Reconstruction

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = 0$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Jet Reconstruction Cambridge-Aachen Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

$p = 0$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Jet Reconstruction Cambridge-Aachen Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

$$p = 0$$

Typical values:
$R = \{0.4, 0.8, 1.0\}$

CA Algorithm:
Often used in jet sub-structure

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

Jet Reconstruction Cambridge-Aachen Algorithm

- List of particles with $k_{\mathrm{T}}^2 = p_x^2 + p_y^2 \quad y = \frac{1}{2}\ln\frac{E+p}{E-p}$
- For each particle, define the distance from the beam $d_{iB} = k_{\mathrm{T},i}^2$
- For each pair (i,j) of different particles, define

$$d_{ij} = \min(k_{\mathrm{T},i}^{2p}, k_{\mathrm{T},j}^{2p})\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{R^2}$$

Typical values:

$R = \{0.4, 0.8, 1.0\}$

$p = 0$

- Determine $\min(d_{ij}, d_{iB})$ over all i and j
  - if d_{ij} → cluster objects i and j, form new object jet k
  - if d_{iB} → add object to list of jets
- Stop clustering only d_{iB}

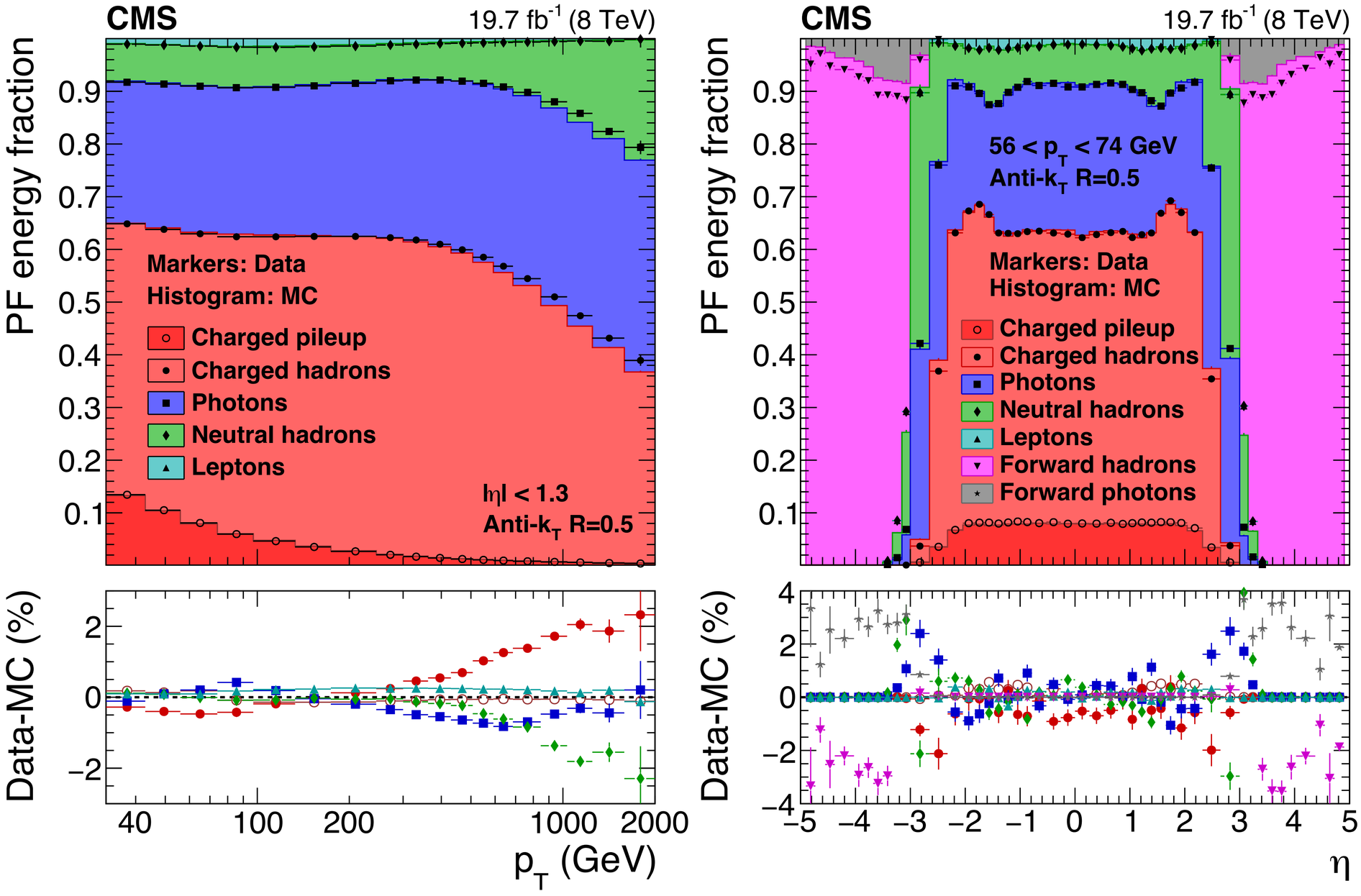
CA Algorithm:

Often used in jet sub-structure

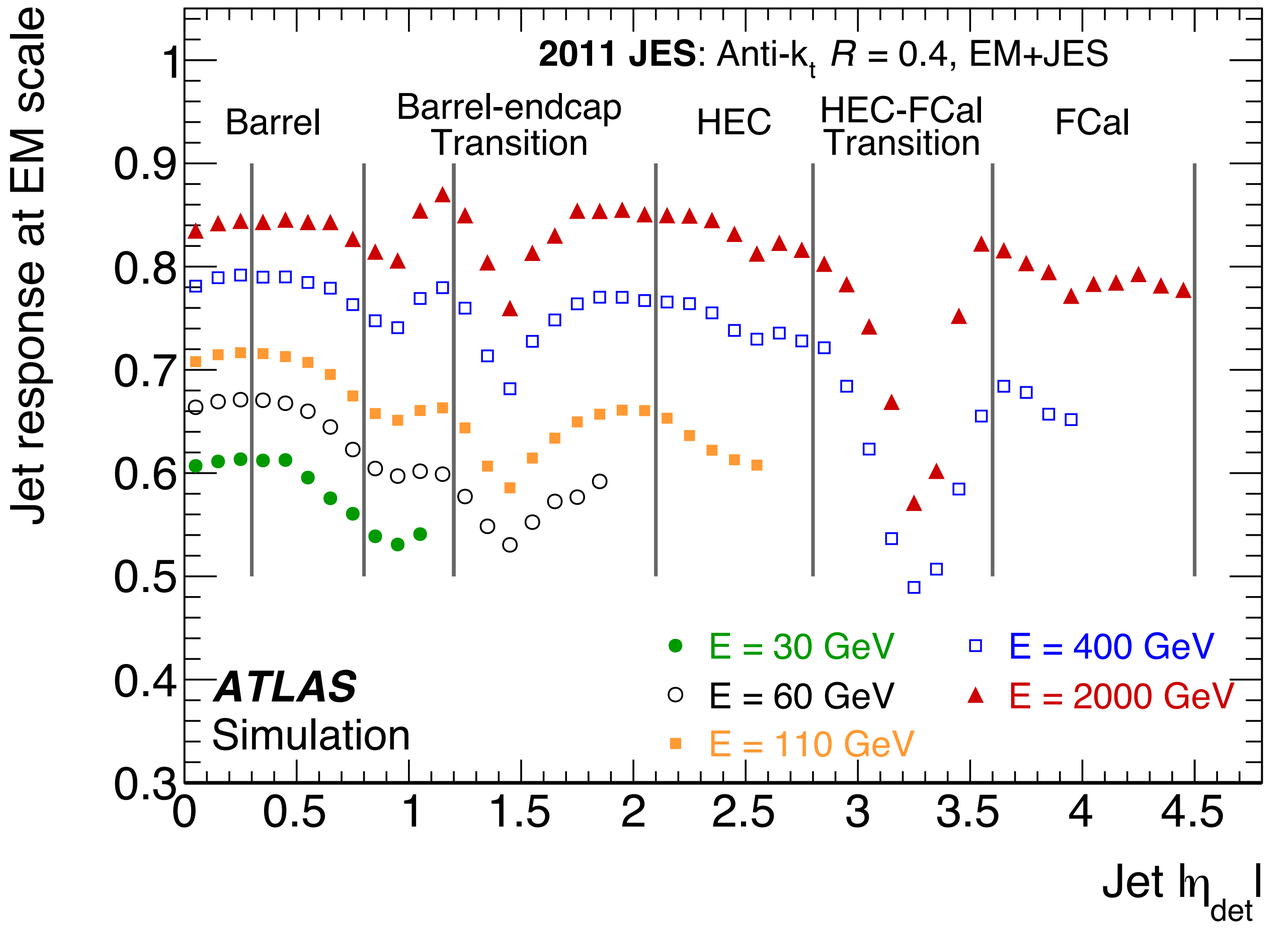
I am so very sorry!!

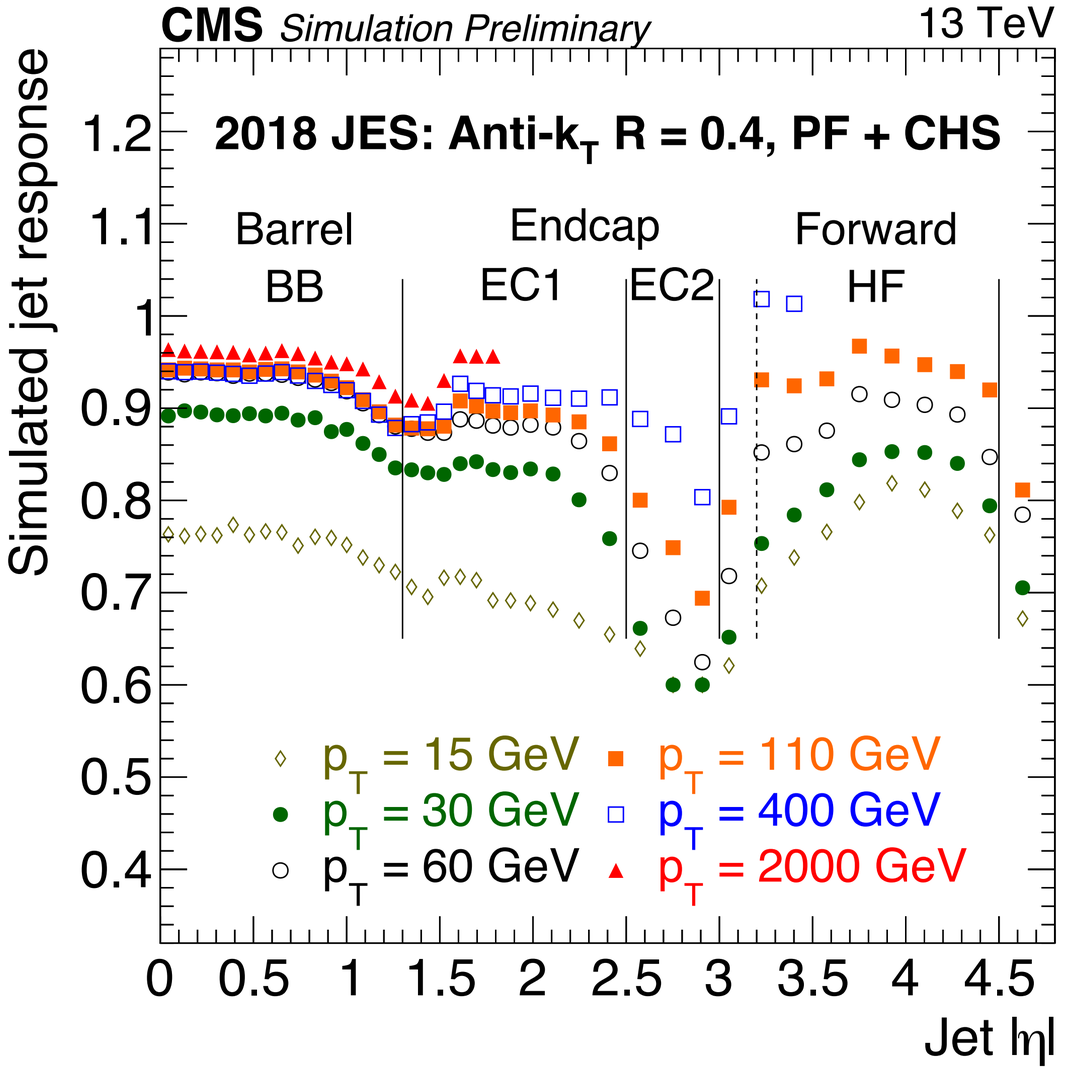
Just not enough time to discuss jet sub-structure -- but wow is it interesting!

Actual particle composition of jets



Done finding jets...



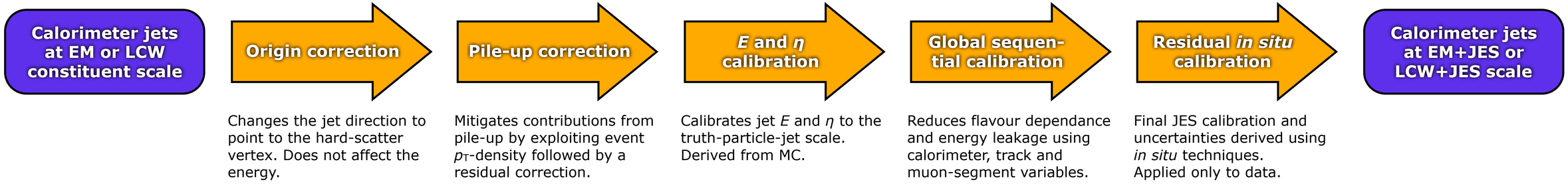


...must now calibrate !

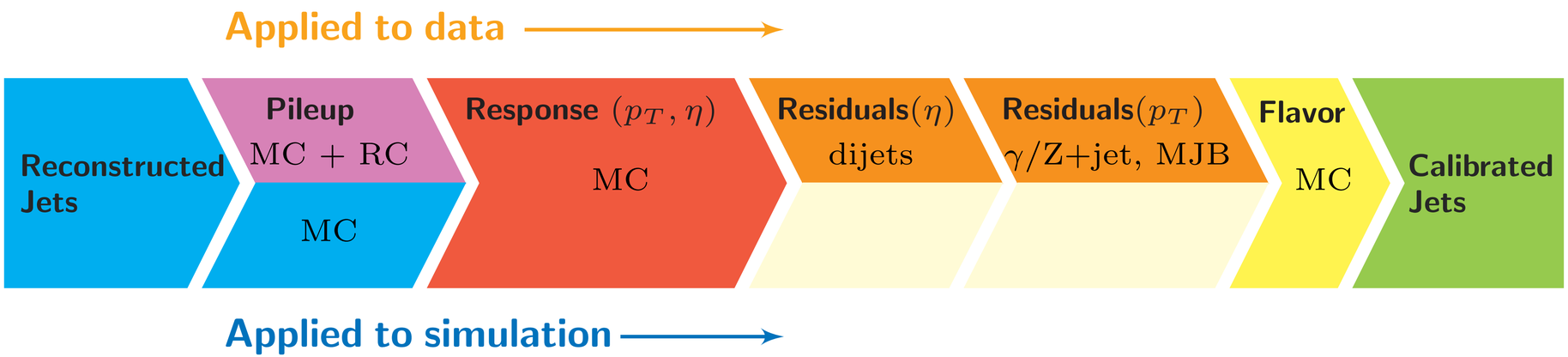
Jet Momentum Corrections

ATLAS

Anti-k_t $R = 0.4$ and $R = 0.6$



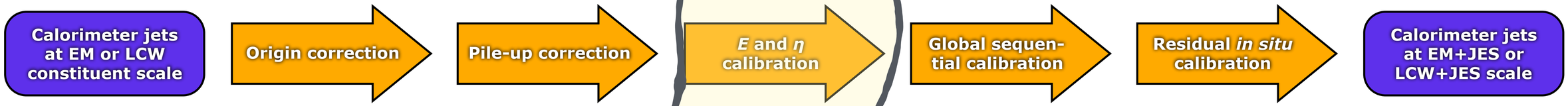
CMS



Jet Momentum Corrections

ATLAS

Anti-k_t $R = 0.4$ and $R = 0.6$



Changes the jet direction to point to the hard-scatter vertex. Does not affect the energy.

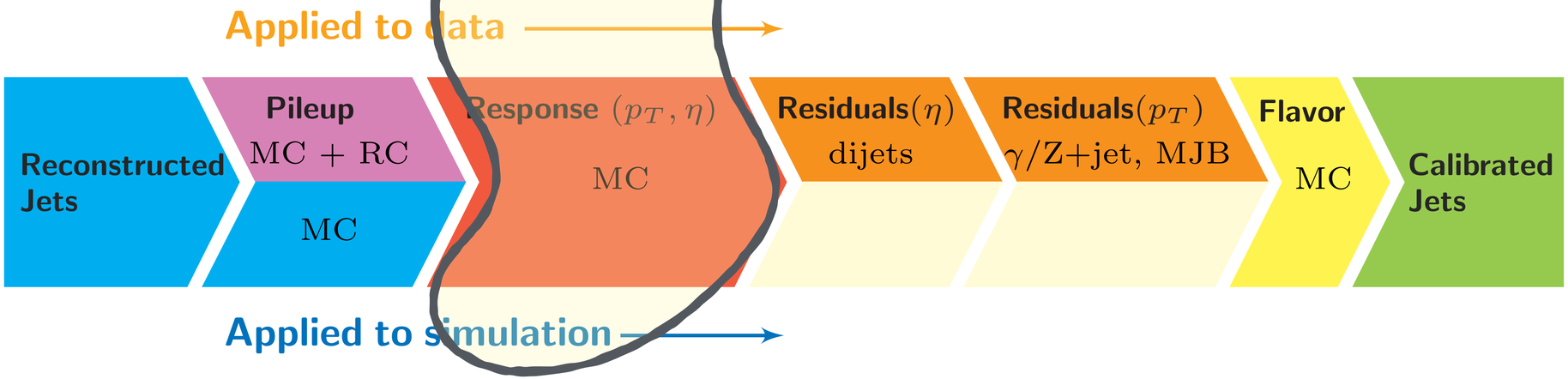
Mitigates contributions from pile-up by exploiting event p_T-density followed by a residual correction.

Calibrates jet E and η to the truth-particle-jet scale. Derived from MC.

Reduces flavour dependance and energy leakage using calorimeter, track and muon-segment variables.

Final JES calibration and uncertainties derived using *in situ* techniques. Applied only to data.

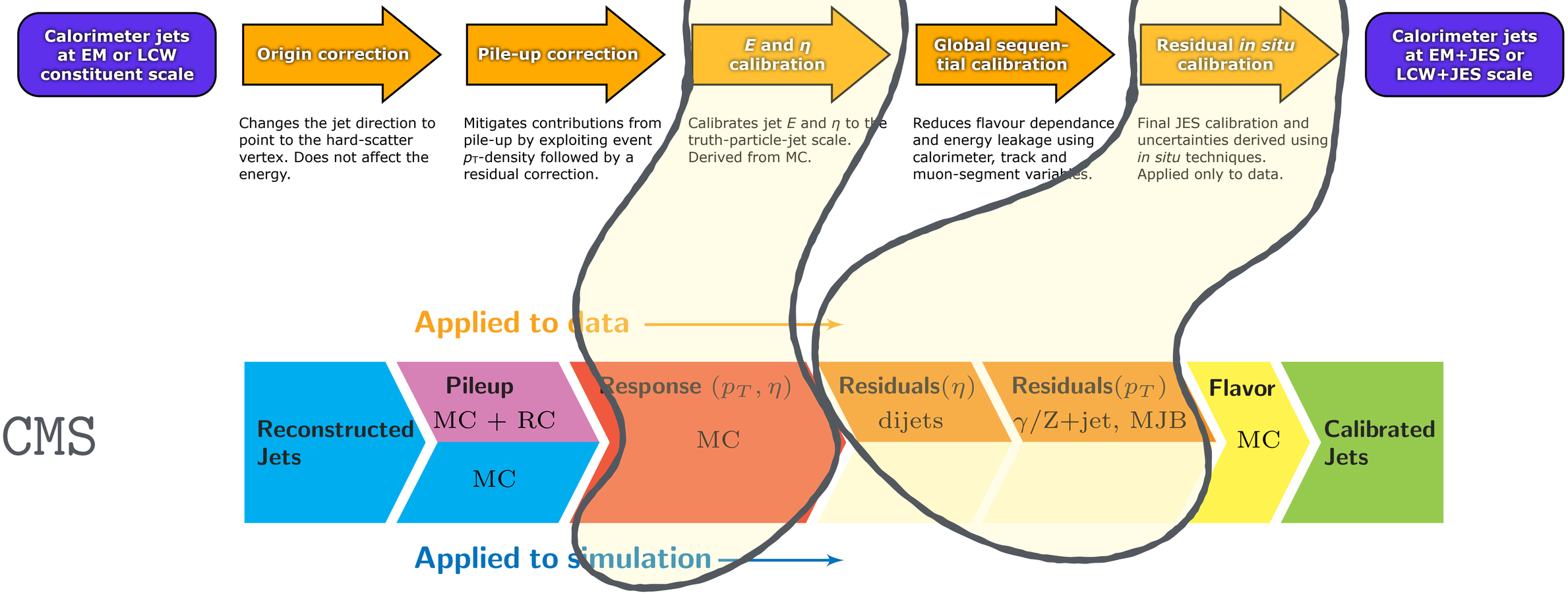
CMS



Jet Momentum Corrections

ATLAS

Anti-k_t $R = 0.4$ and $R = 0.6$

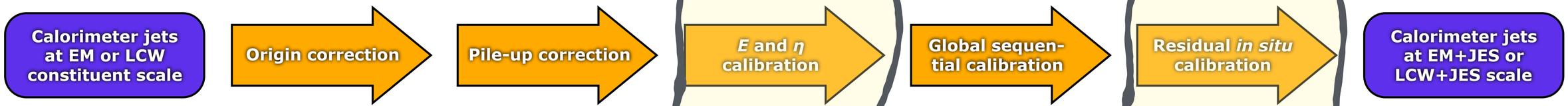


CMS

Jet Momentum Corrections

ATLAS

Anti-k_t $R = 0.4$ and $R = 0.6$



Changes the jet direction to point to the hard-scatter vertex. Does not affect the energy.

Mitigates contributions from pile-up by exploiting event p_T-density followed by a residual correction.

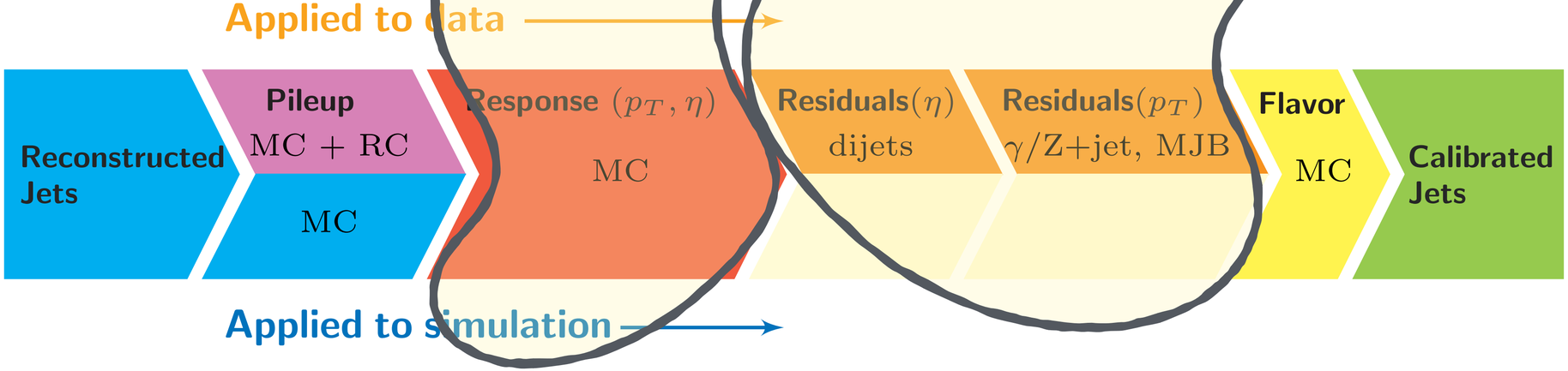
Calibrates jet E and η to the truth-particle-jet scale. Derived from MC.

Reduces flavour dependance and energy leakage using calorimeter, track and muon-segment variables.

Final JES calibration and uncertainties derived using *in situ* techniques. Applied only to data.

Discuss these (briefly) first

CMS



Jet Momentum Corrections

ATLAS

Anti-k_t $R = 0.4$ and $R = 0.6$

Discuss these (briefly) first

Calorimeter jets at EM or LCW constituent scale

Origin correction
Changes the jet direction to point to the hard-scatter vertex. Does not affect the energy.

Pile-up correction
Mitigates contributions from pile-up by exploiting event p_T-density followed by a residual correction.

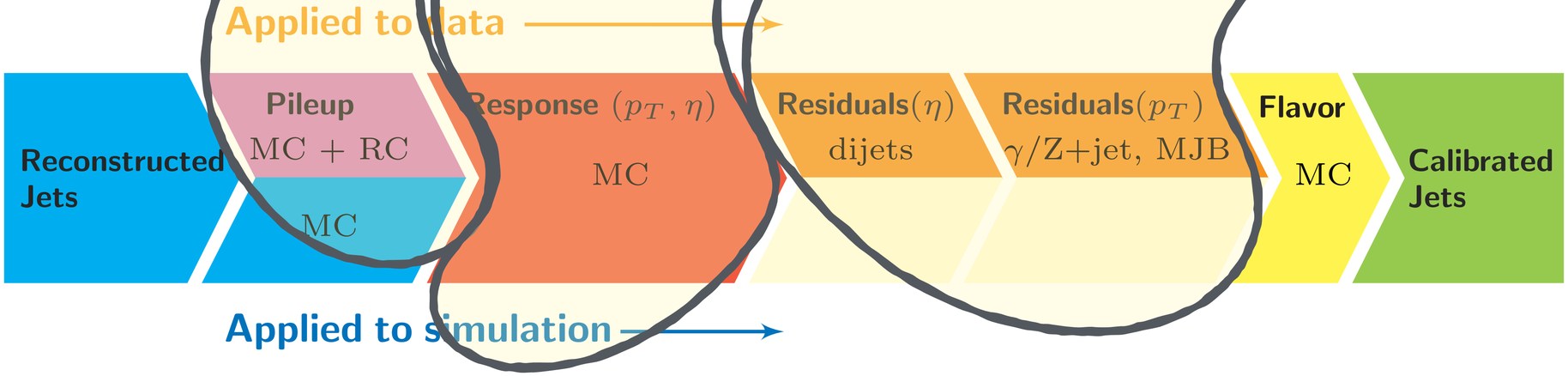
***E* and *η* calibration**
Calibrates jet *E* and *η* to the truth-particle-jet scale. Derived from MC.

Global sequential calibration
Reduces flavour dependance and energy leakage using calorimeter, track and muon-segment variables.

Residual *in situ* calibration
Final JES calibration and uncertainties derived using *in situ* techniques. Applied only to data.

Calorimeter jets at EM+JES or LCW+JES scale

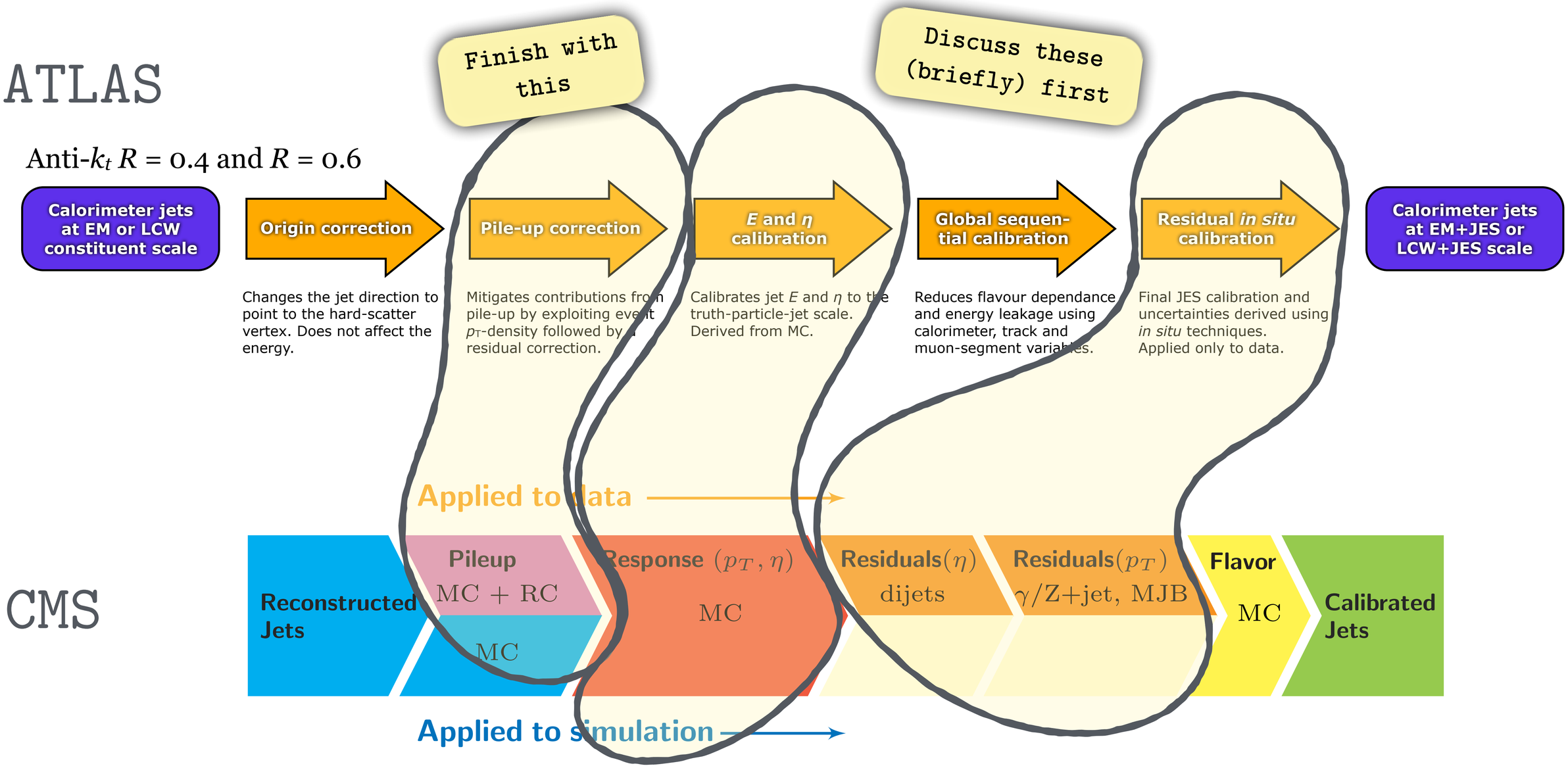
CMS



Jet Momentum Corrections

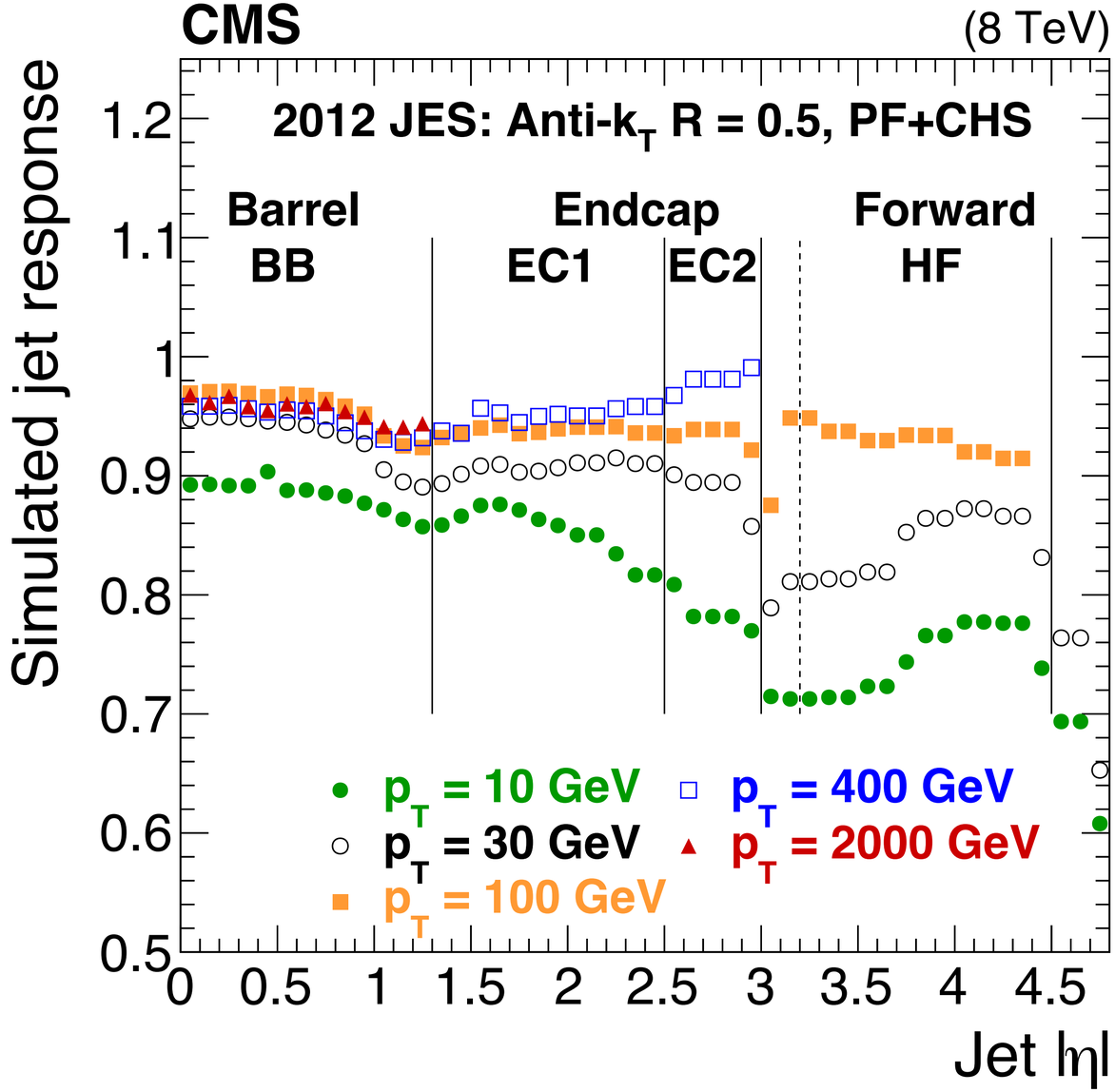
ATLAS

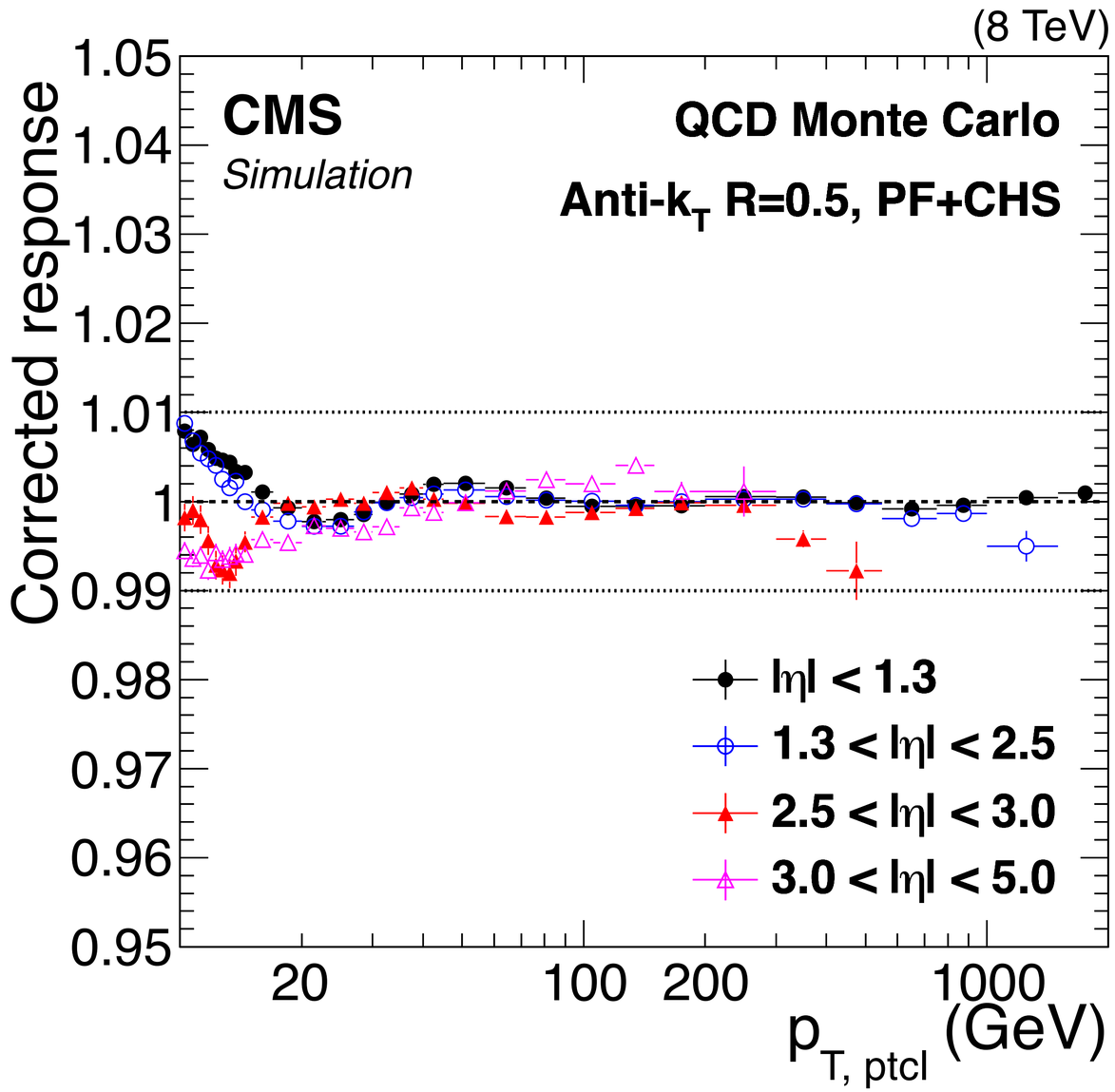
Anti-k_t $R = 0.4$ and $R = 0.6$



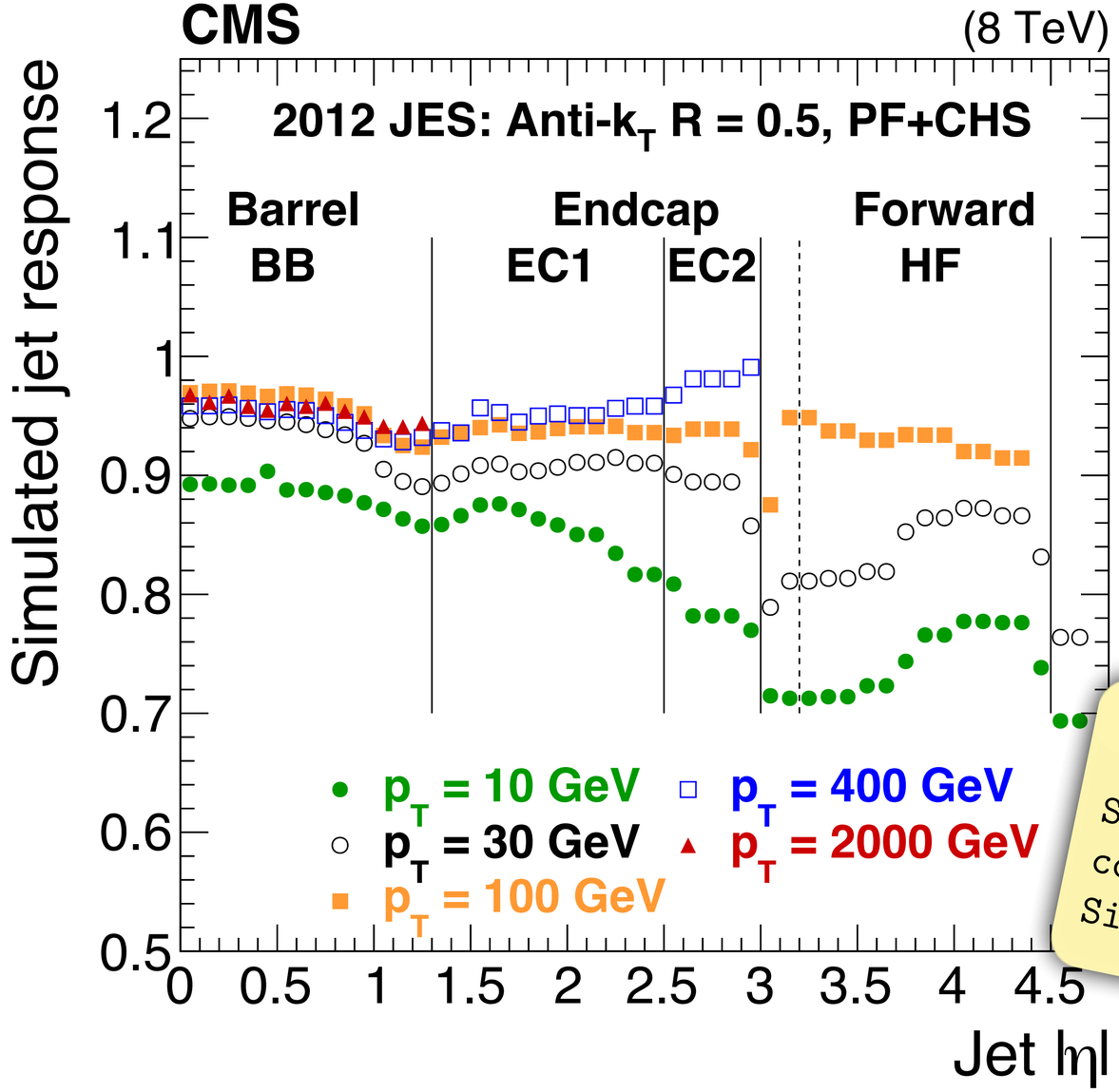
CMS

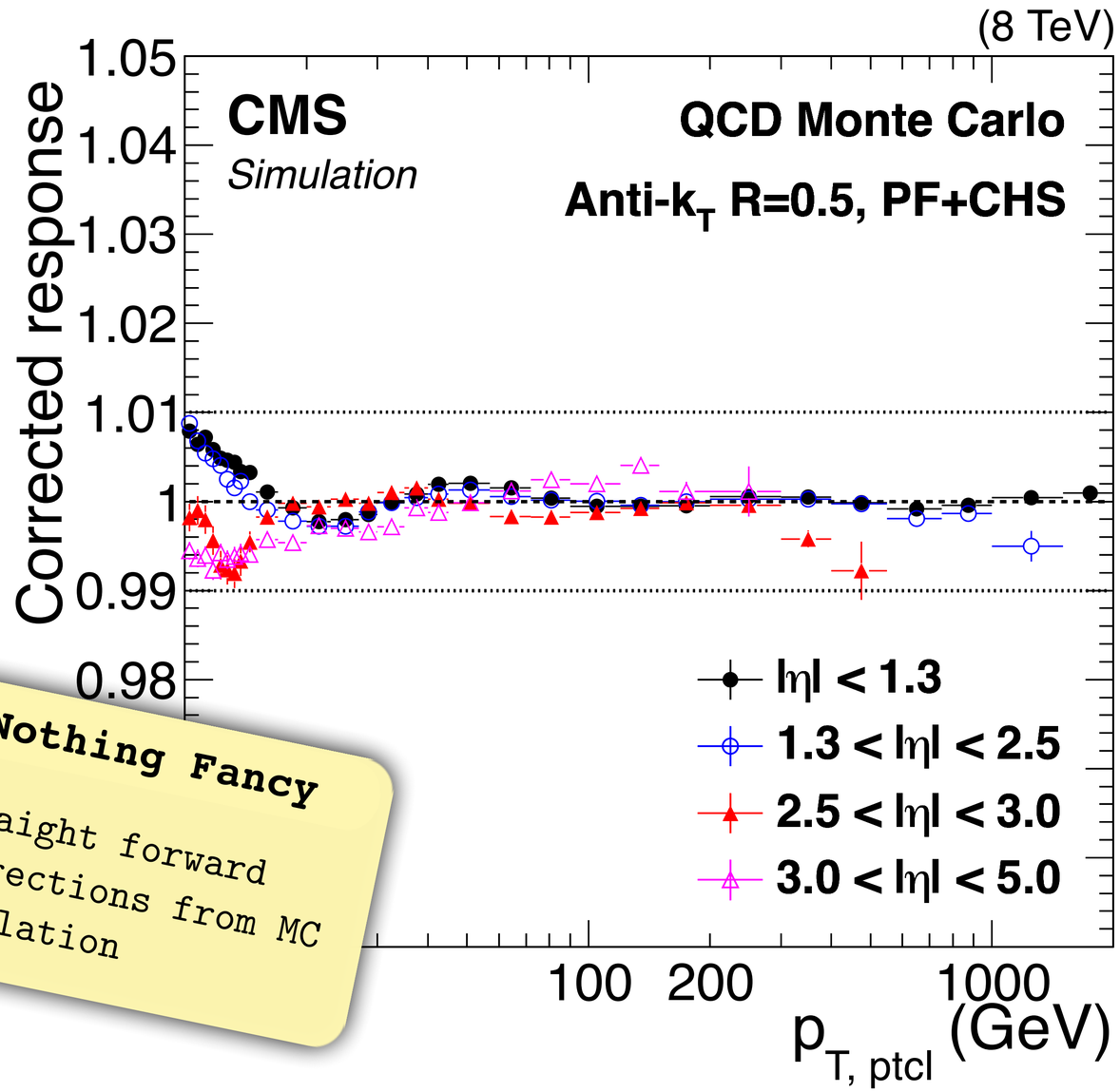
Step 1: Correct gross response using MC Simulation





Step 1: Correct gross response using MC Simulation



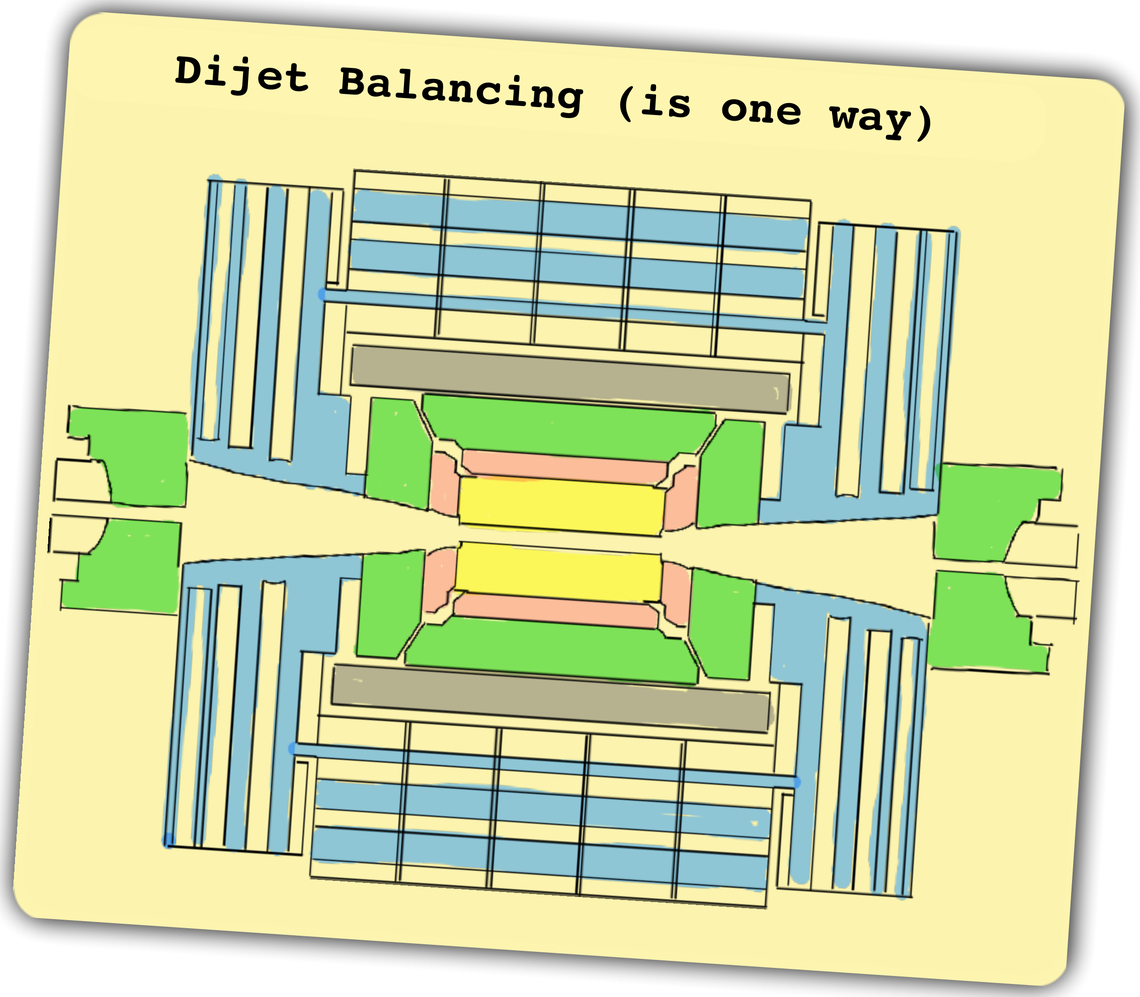


Nothing Fancy

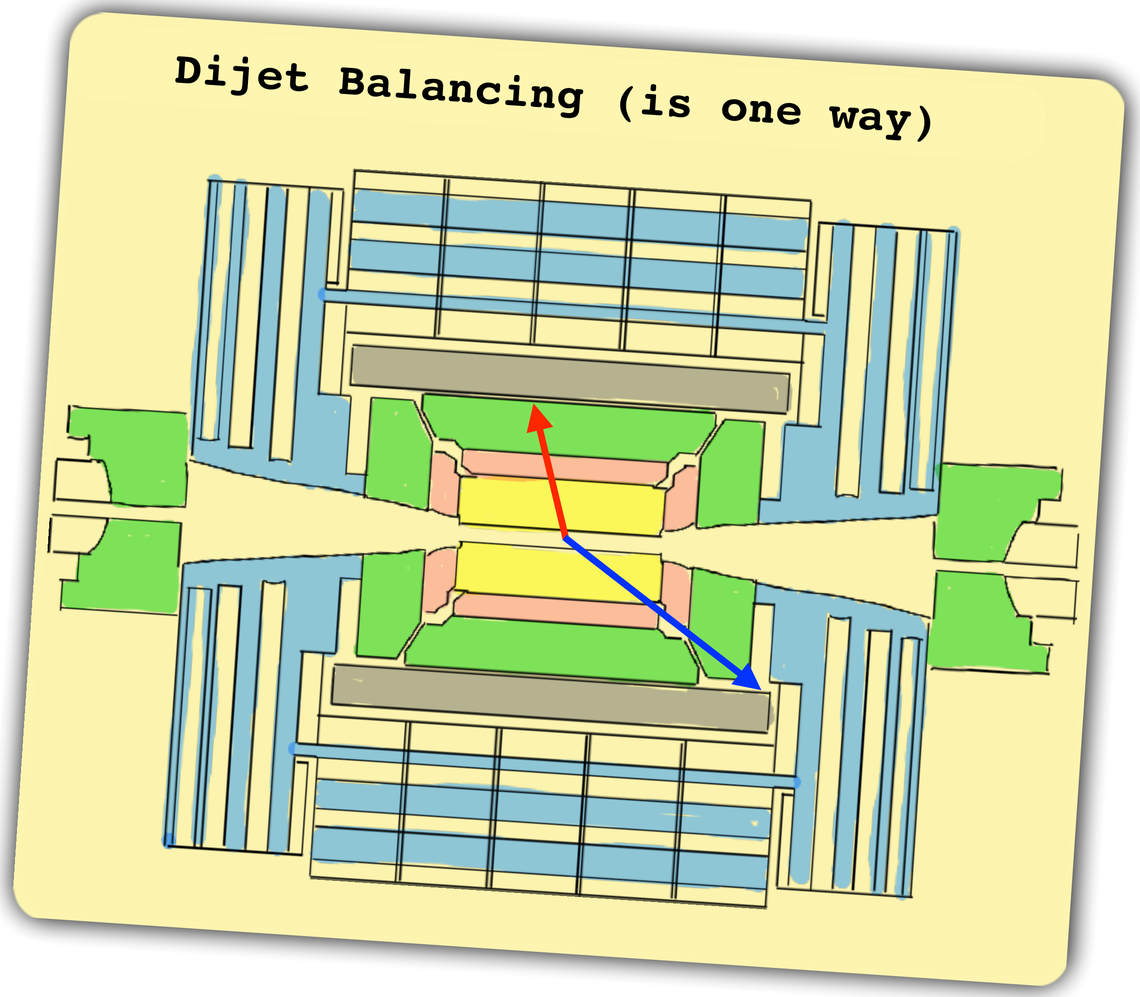
Straight forward corrections from MC Simulation

Step 2: Relative η Residual-Correction using data

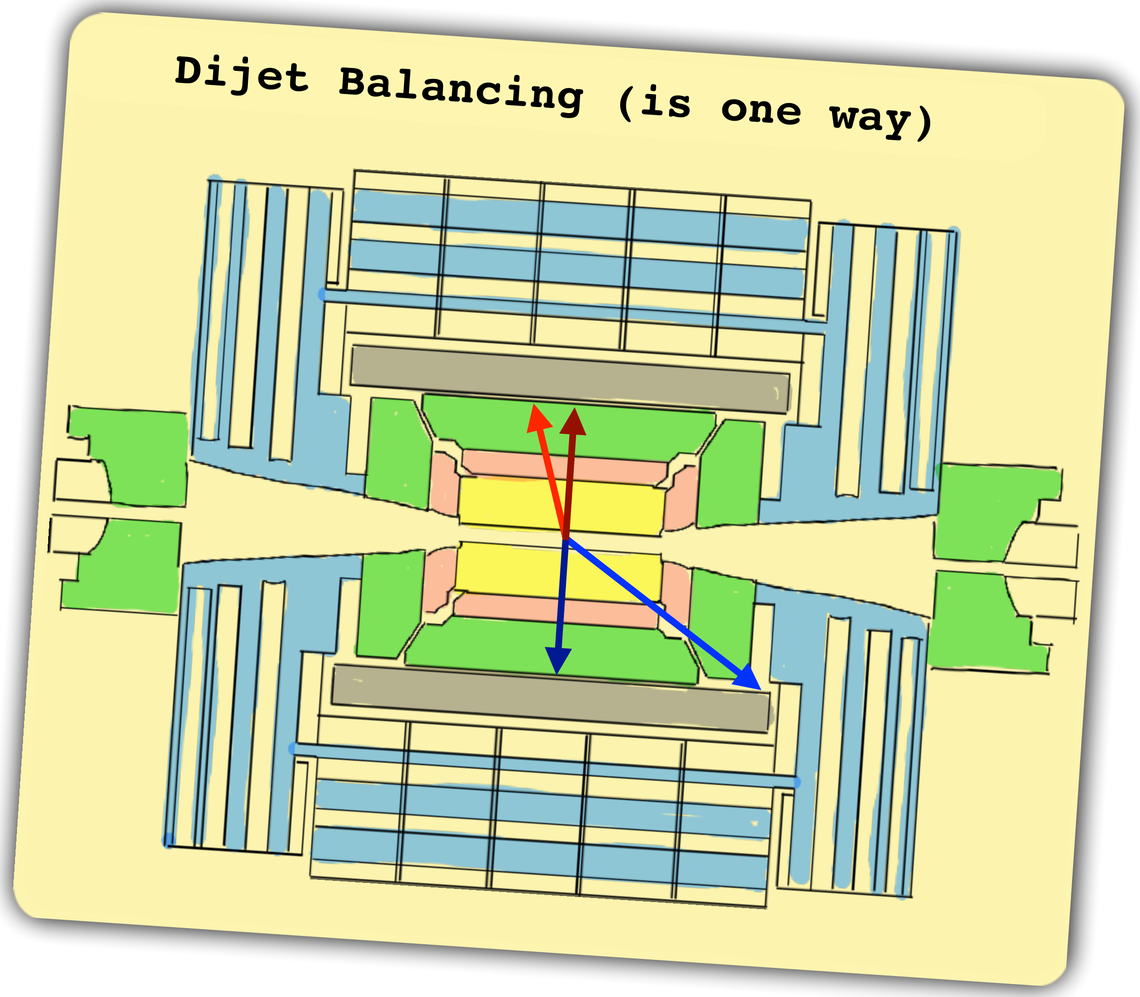
Step 2: Relative η Residual-Correction using data



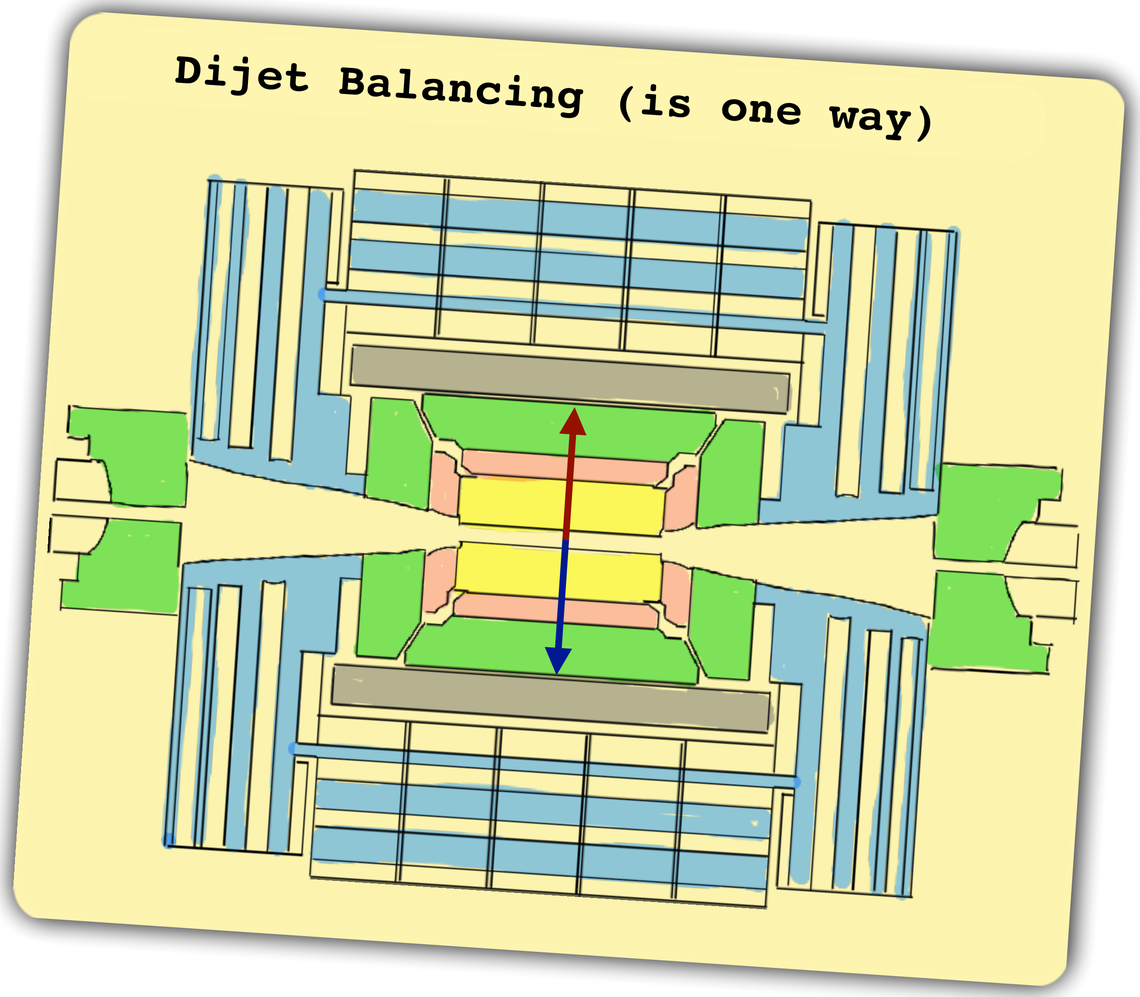
Step 2: Relative η Residual-Correction using data



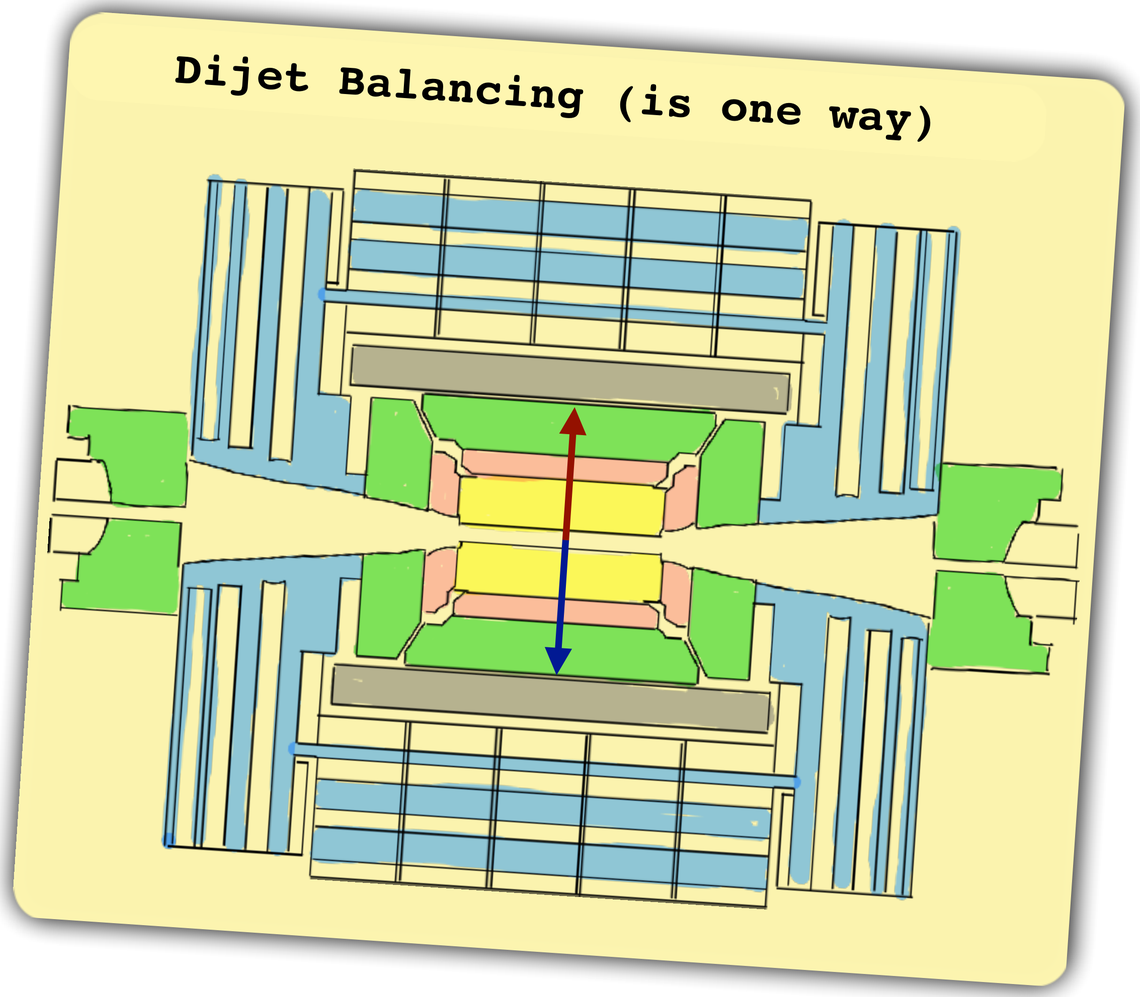
Step 2: Relative η Residual-Correction using data

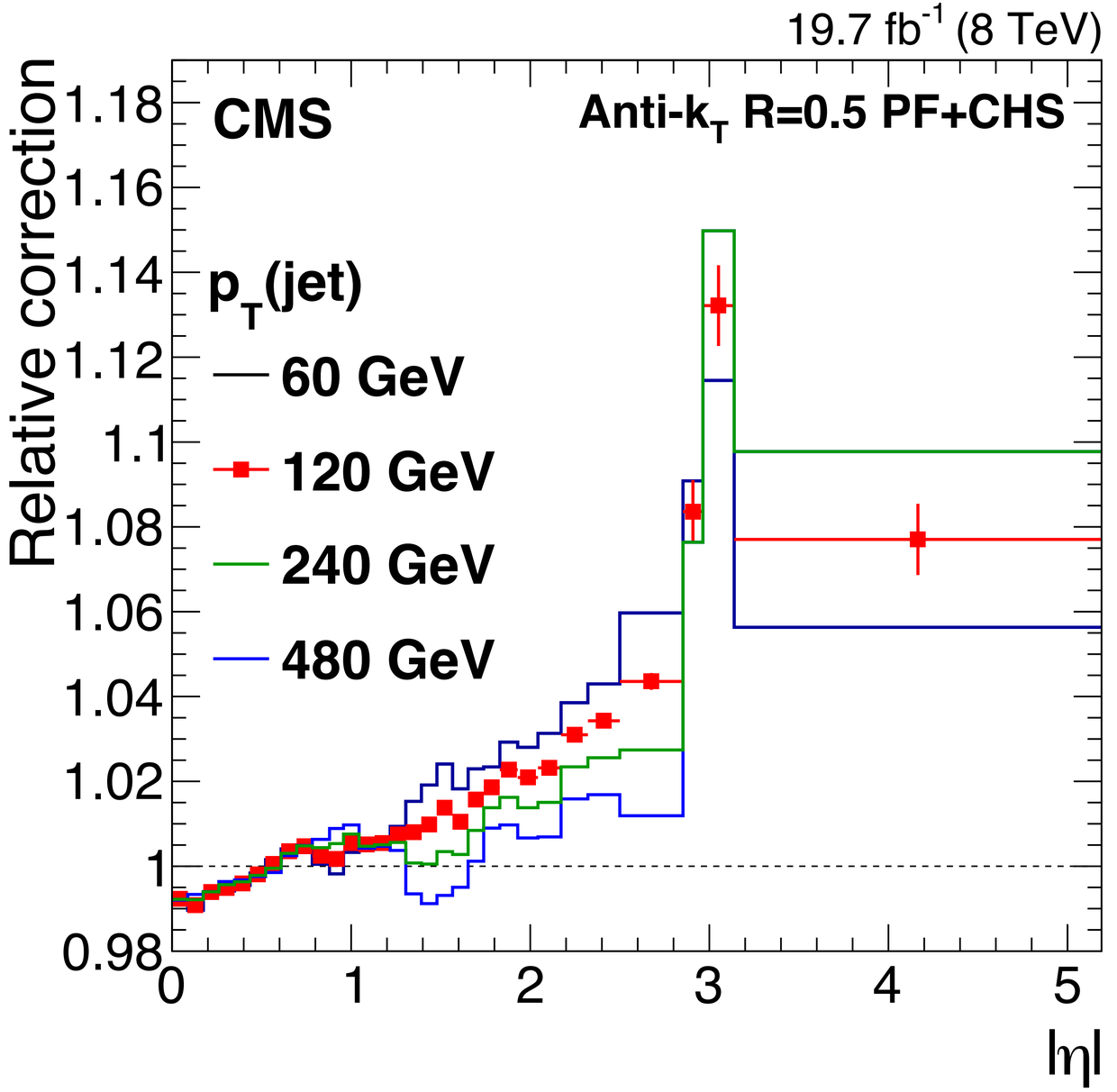


Step 2: Relative η Residual-Correction using data



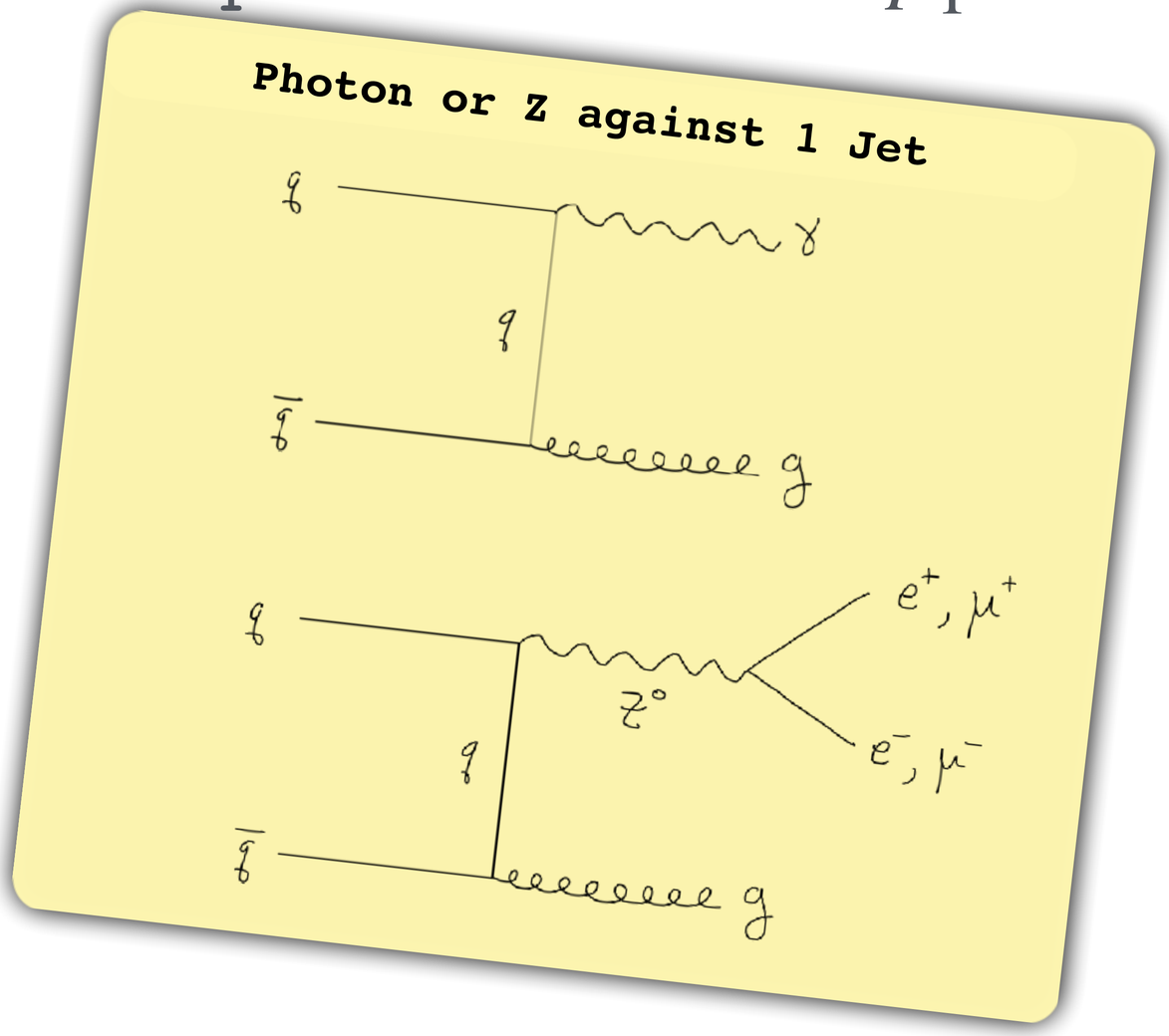
Step 2: Relative η Residual-Correction using data



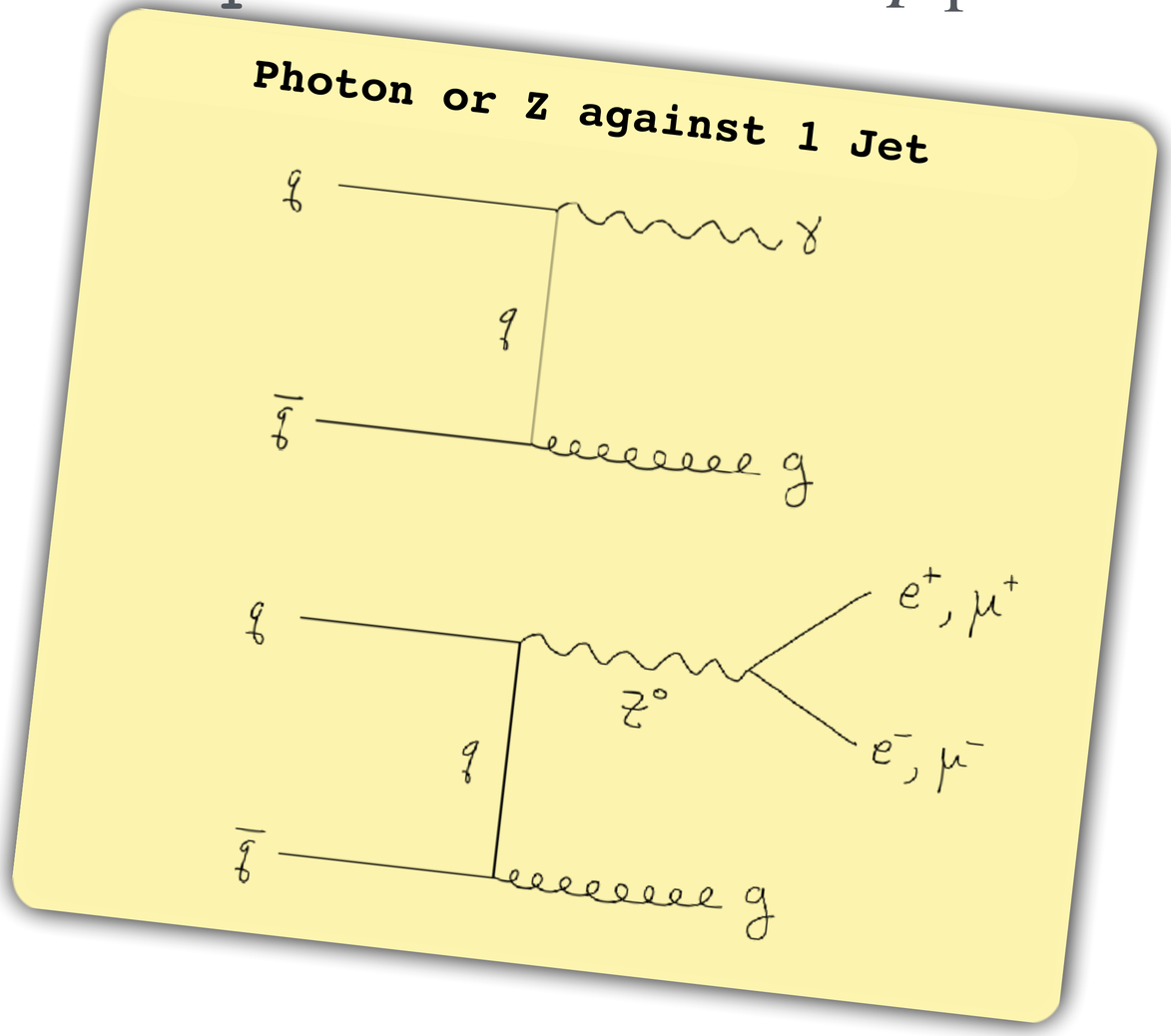


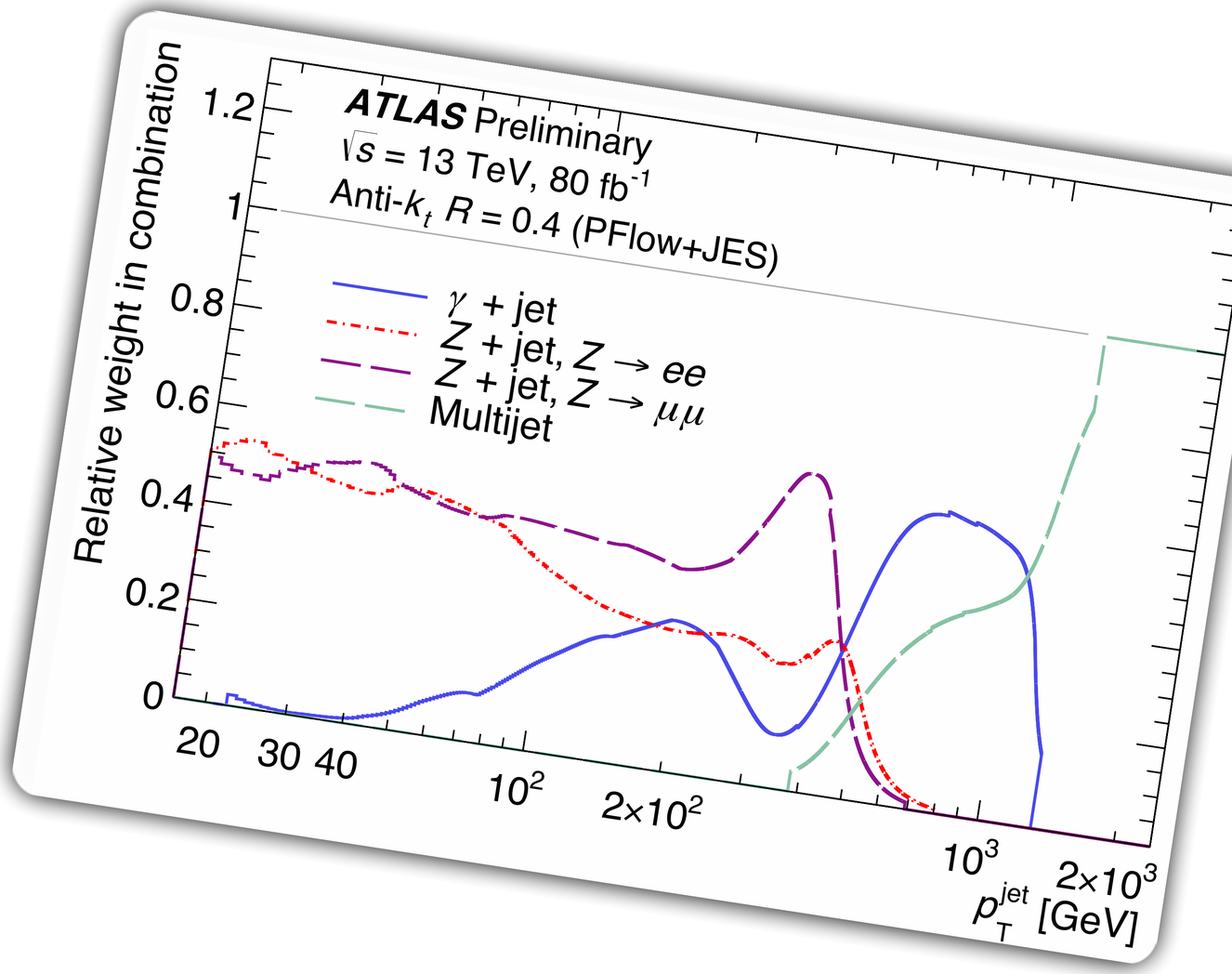
Step 3: Absolute p_{T} Residual-Correction using data

Step 3: Absolute p_{T} Residual-Correction using data

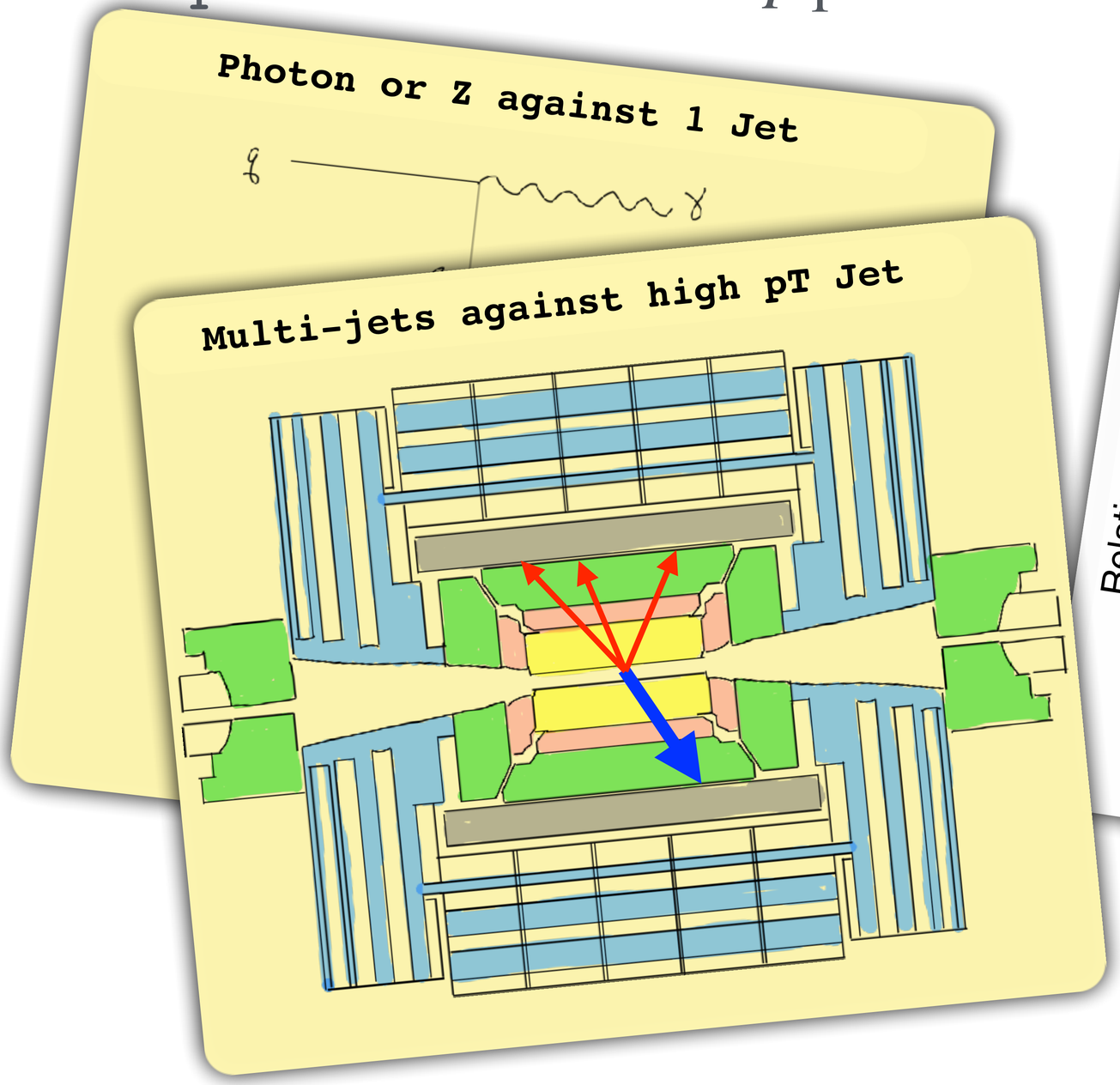


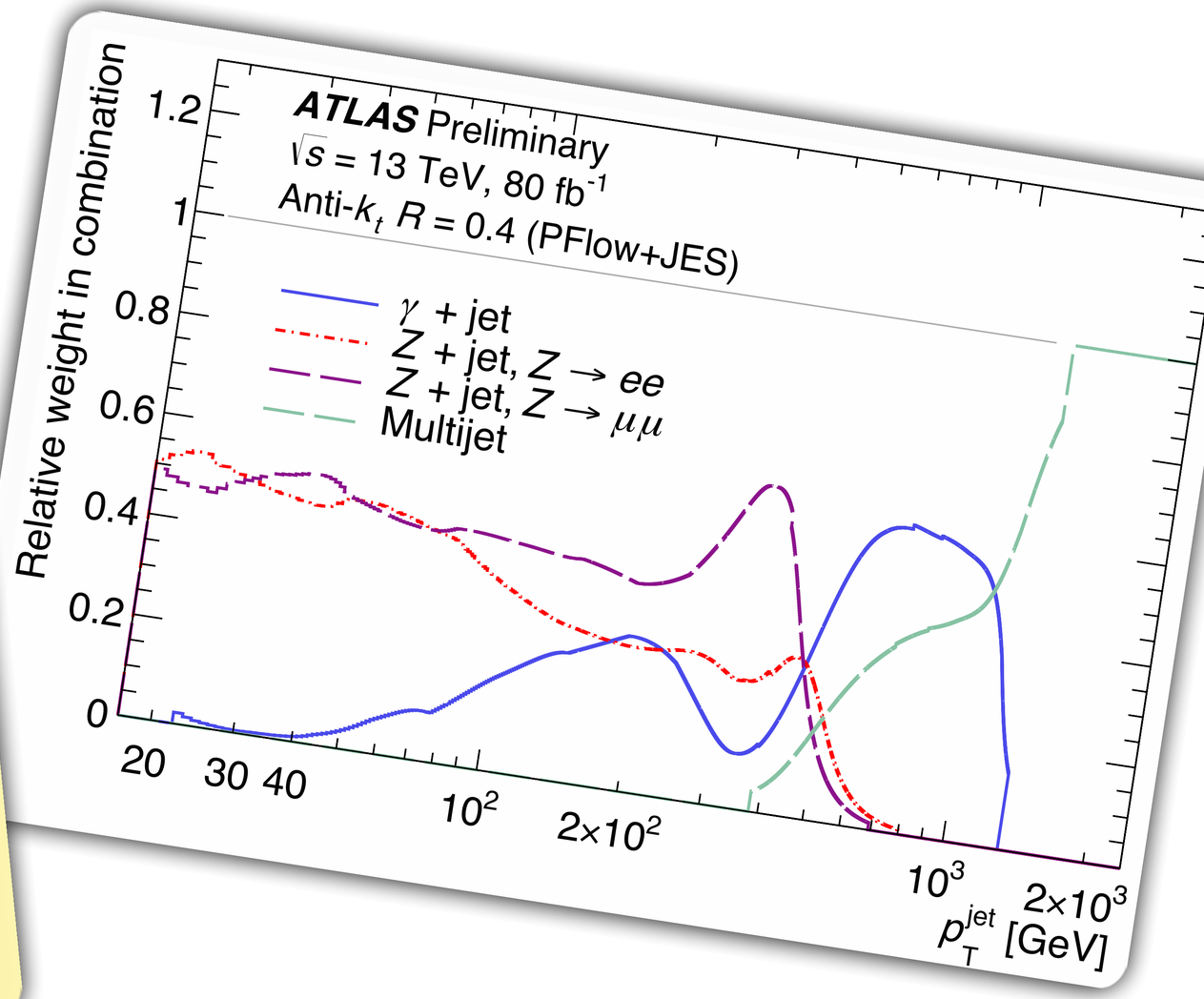
Step 3: Absolute p_T Residual-Correction using data



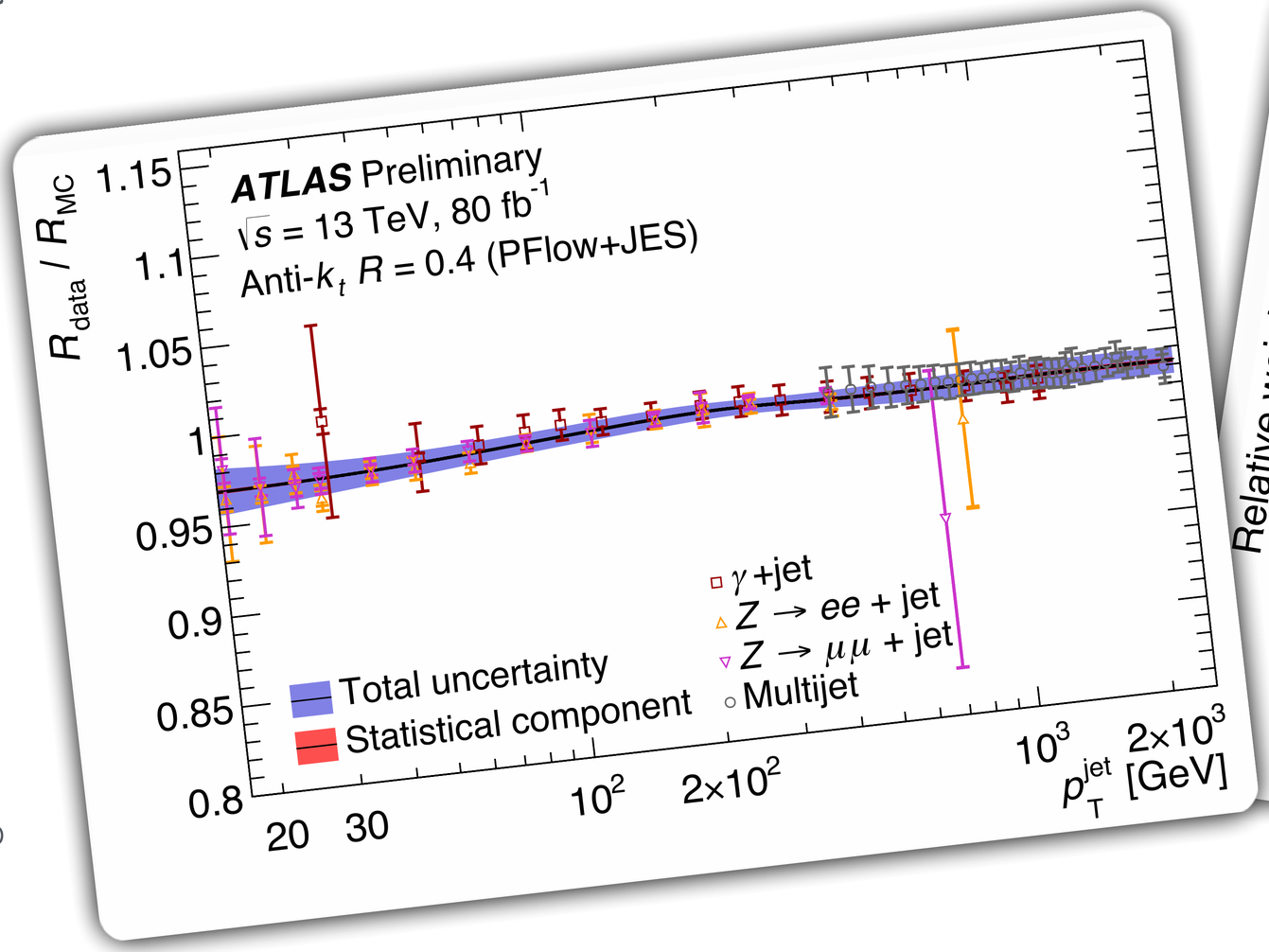


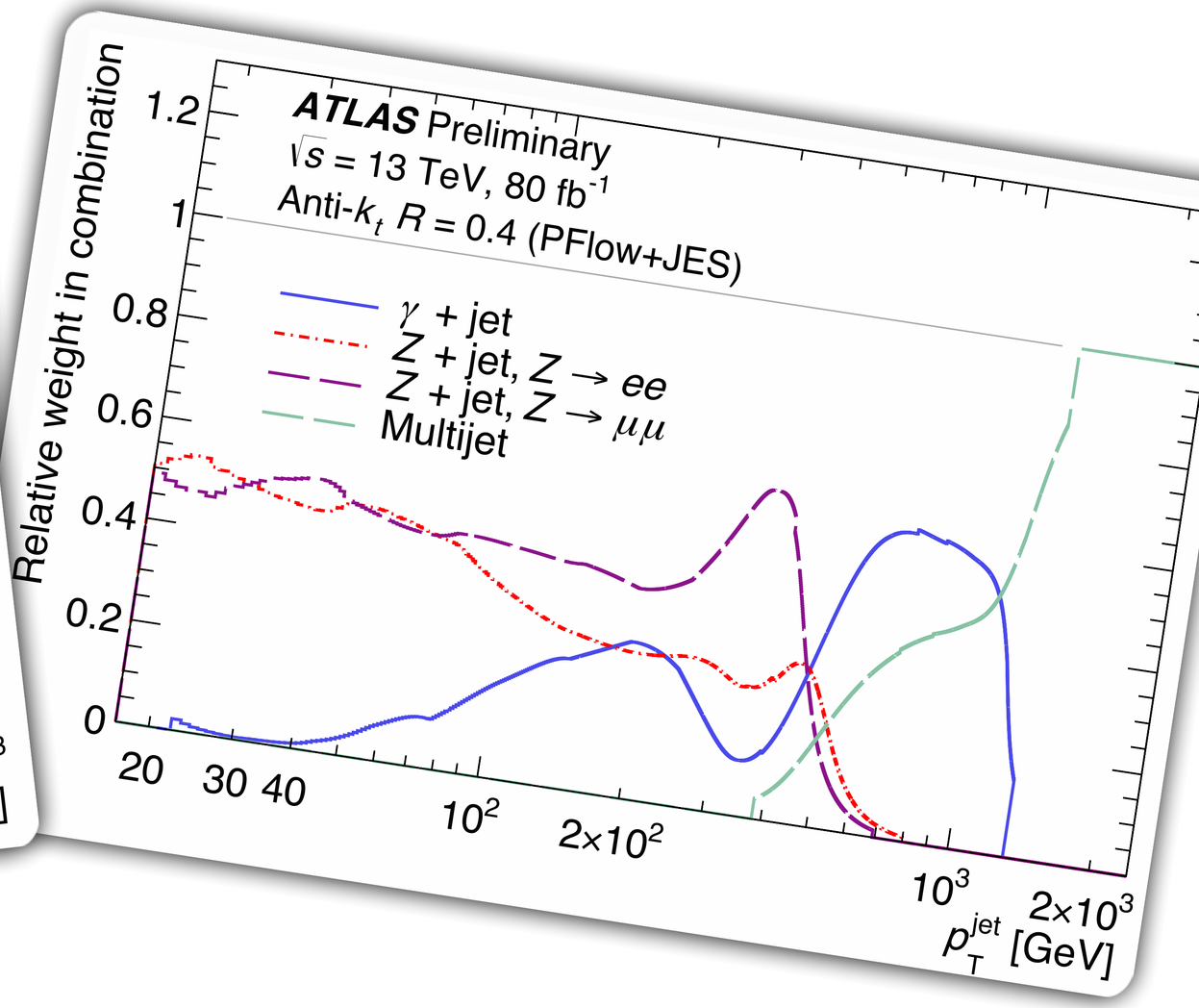
Step 3: Absolute p_T Residual-Correction using data



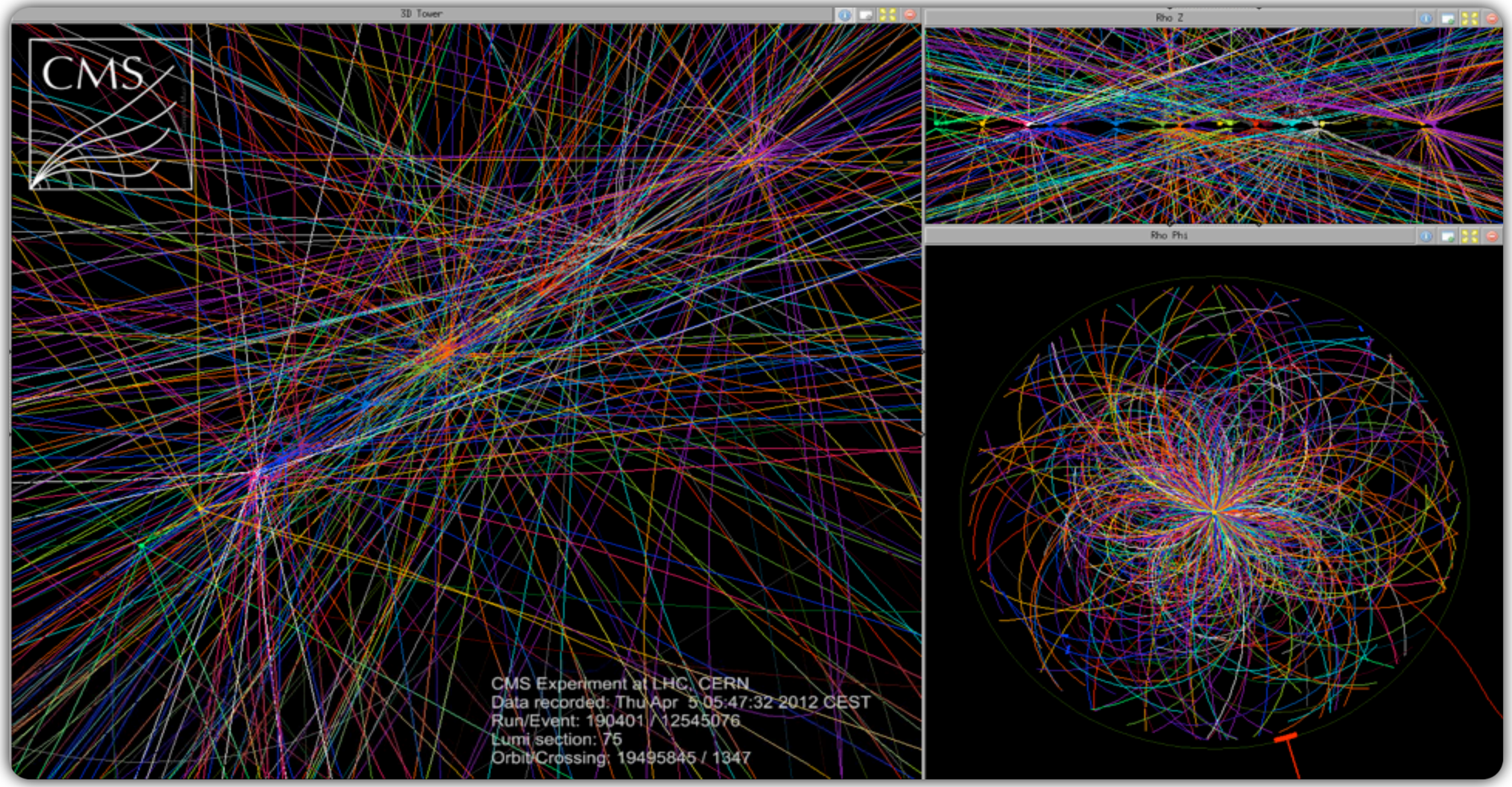


Step 3: Absolute p_{T} Residual-Correction using data



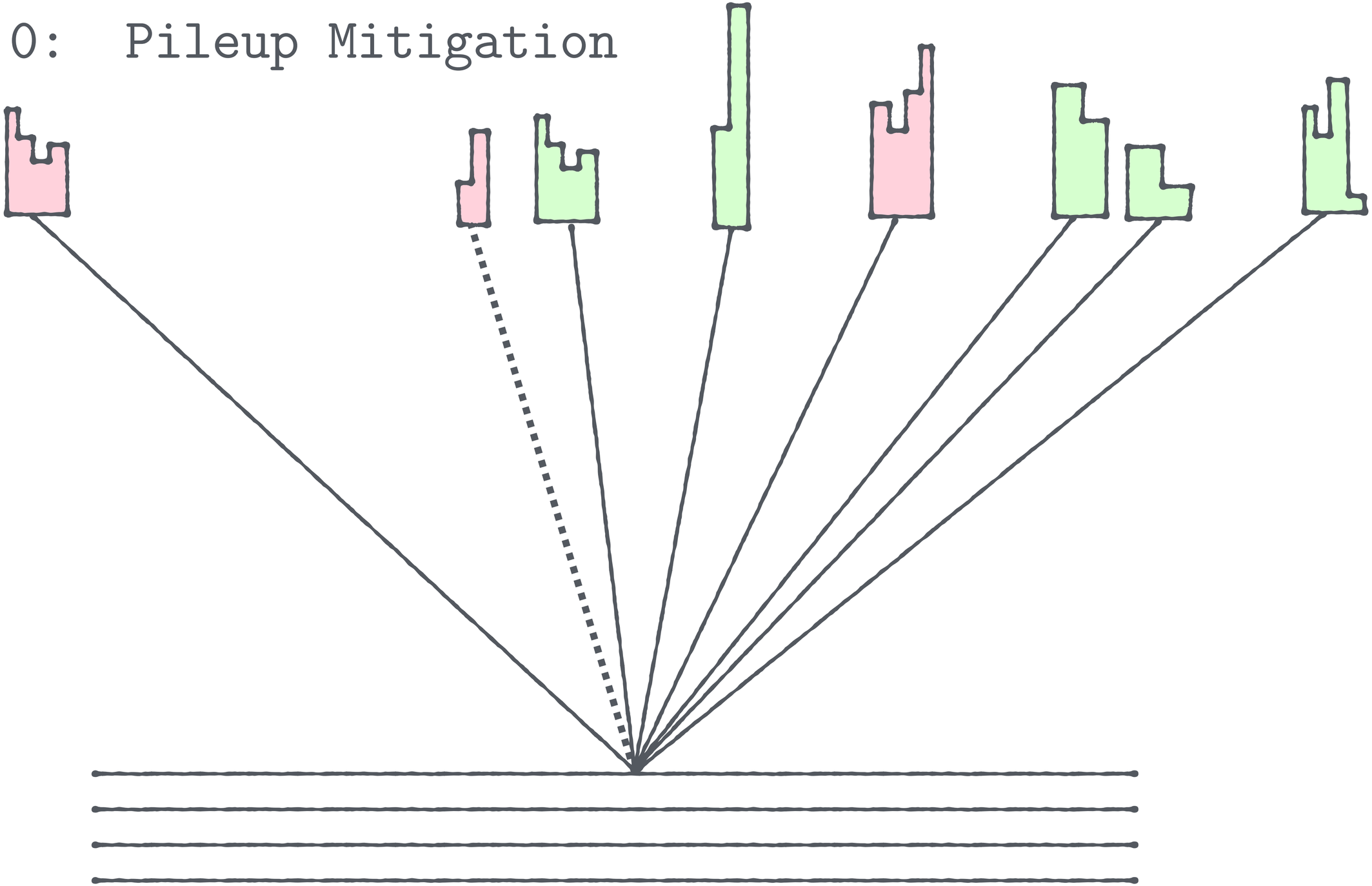


Step 0: Pileup Mitigation

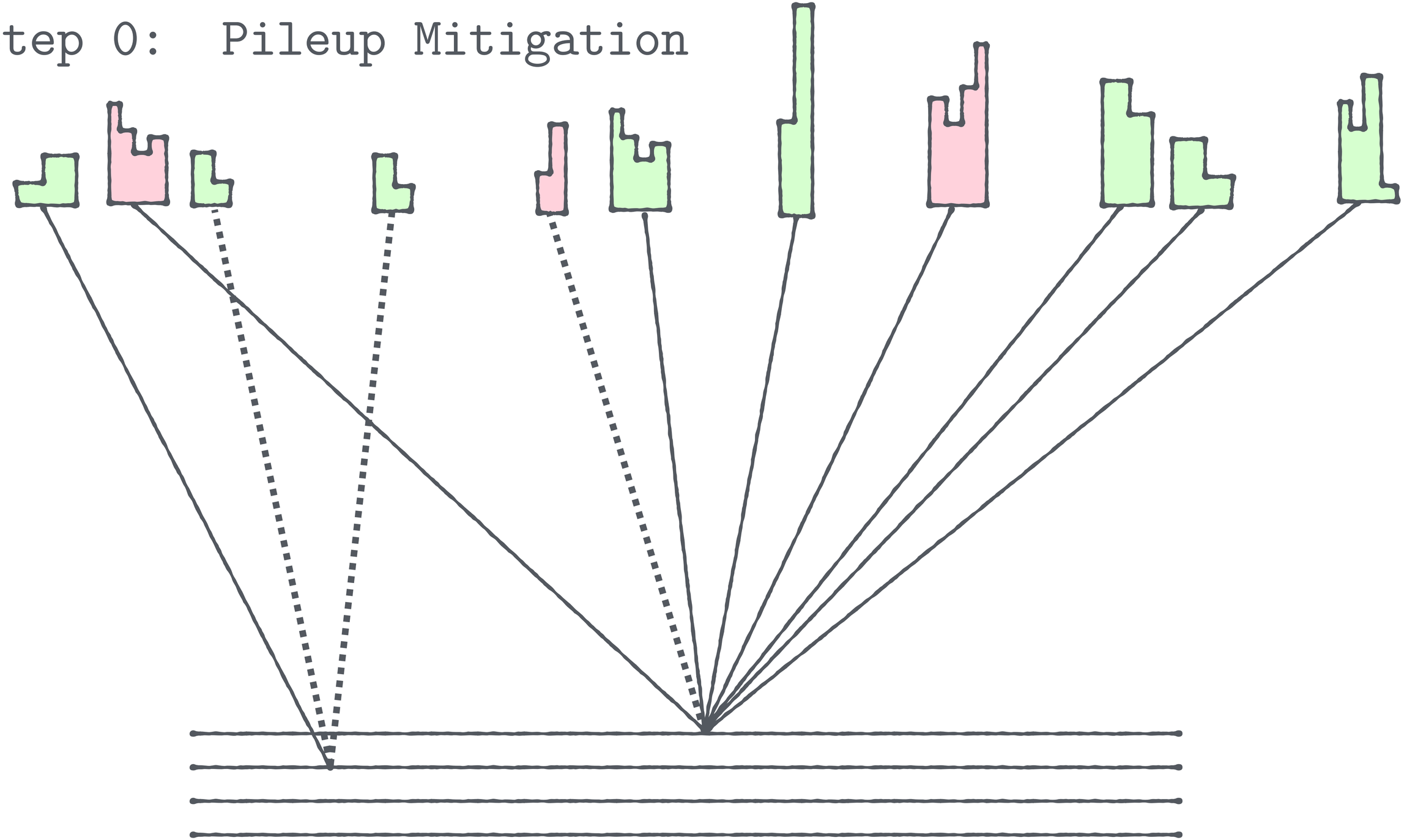


Step 0: Pileup Mitigation

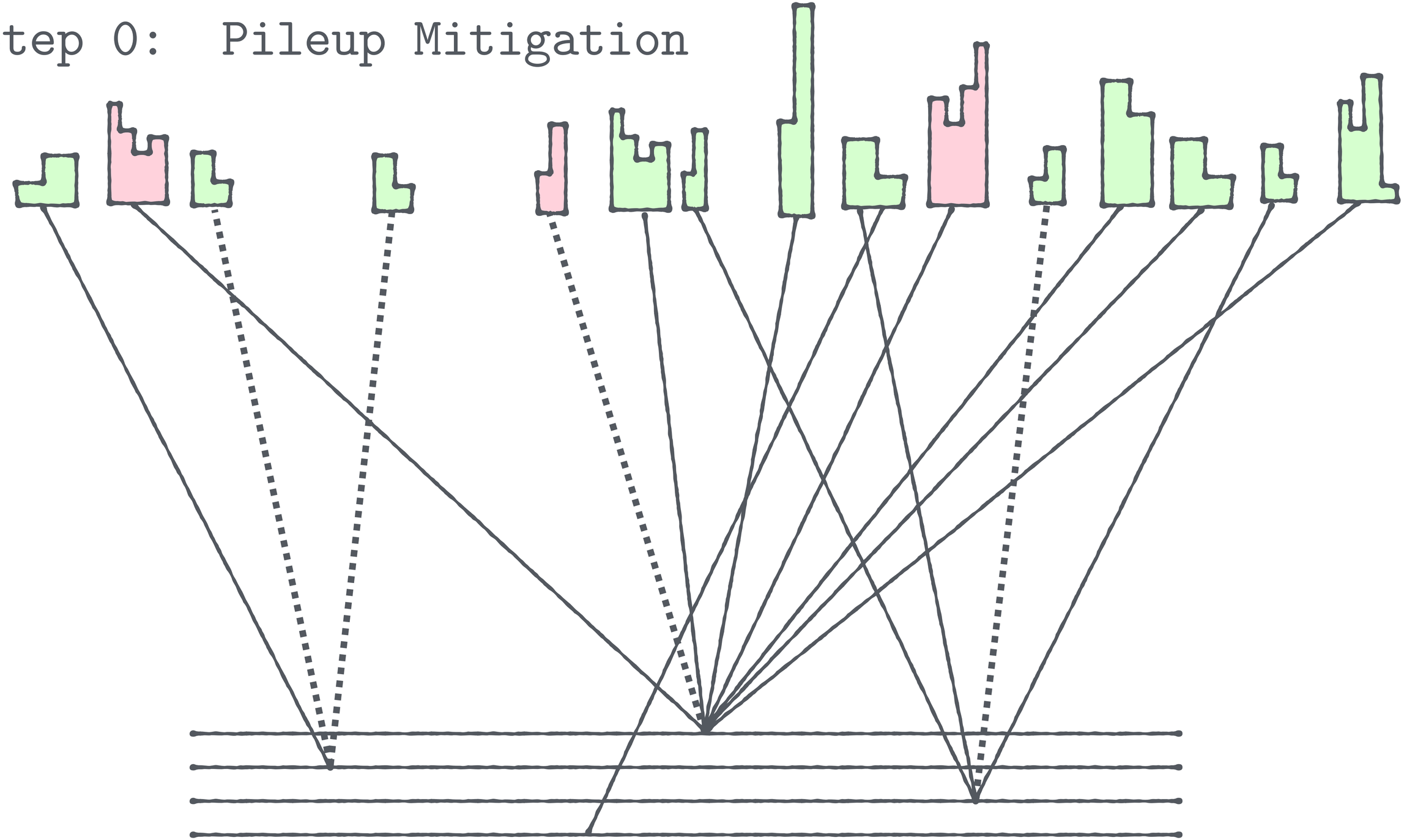
Step 0: Pileup Mitigation



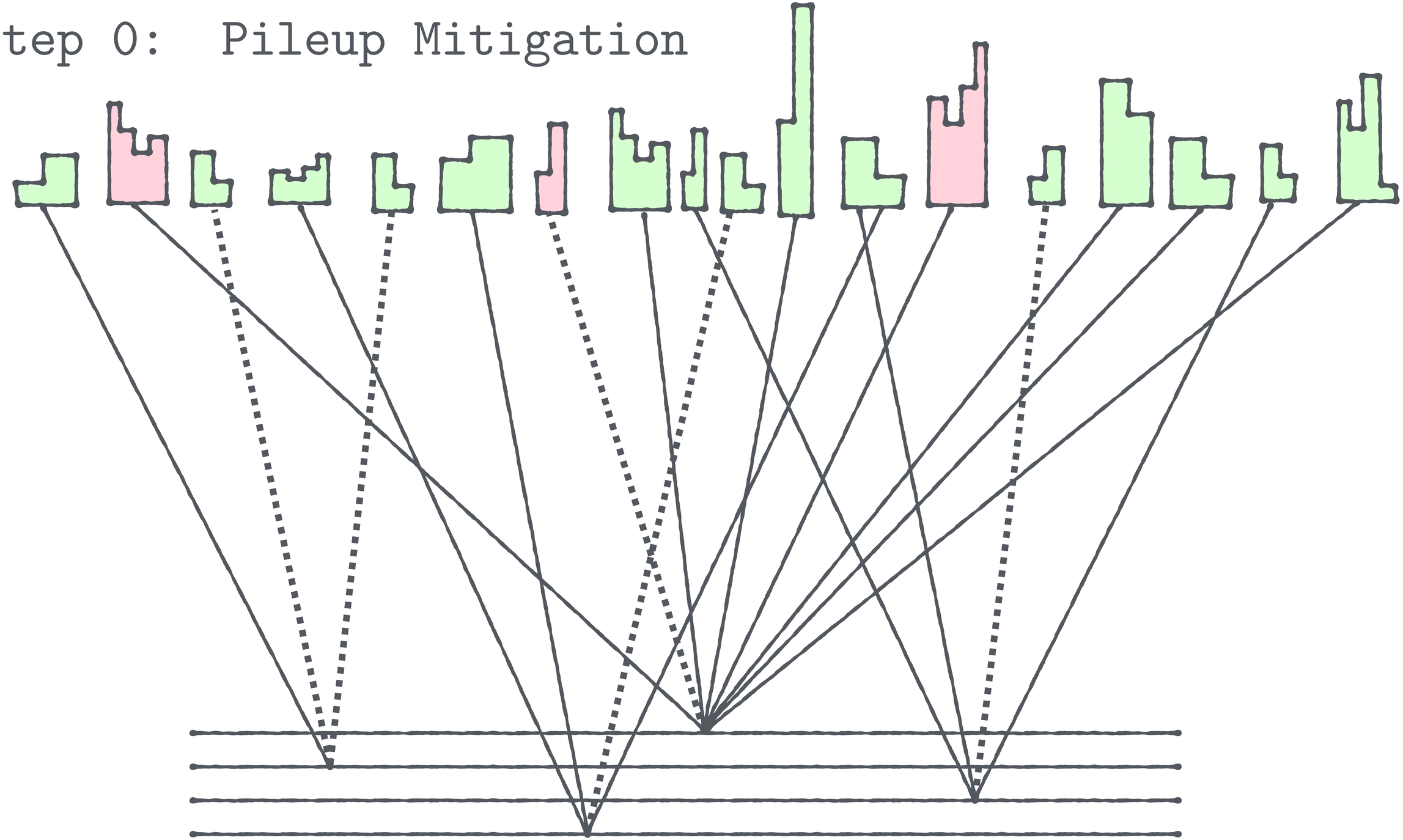
Step 0: Pileup Mitigation



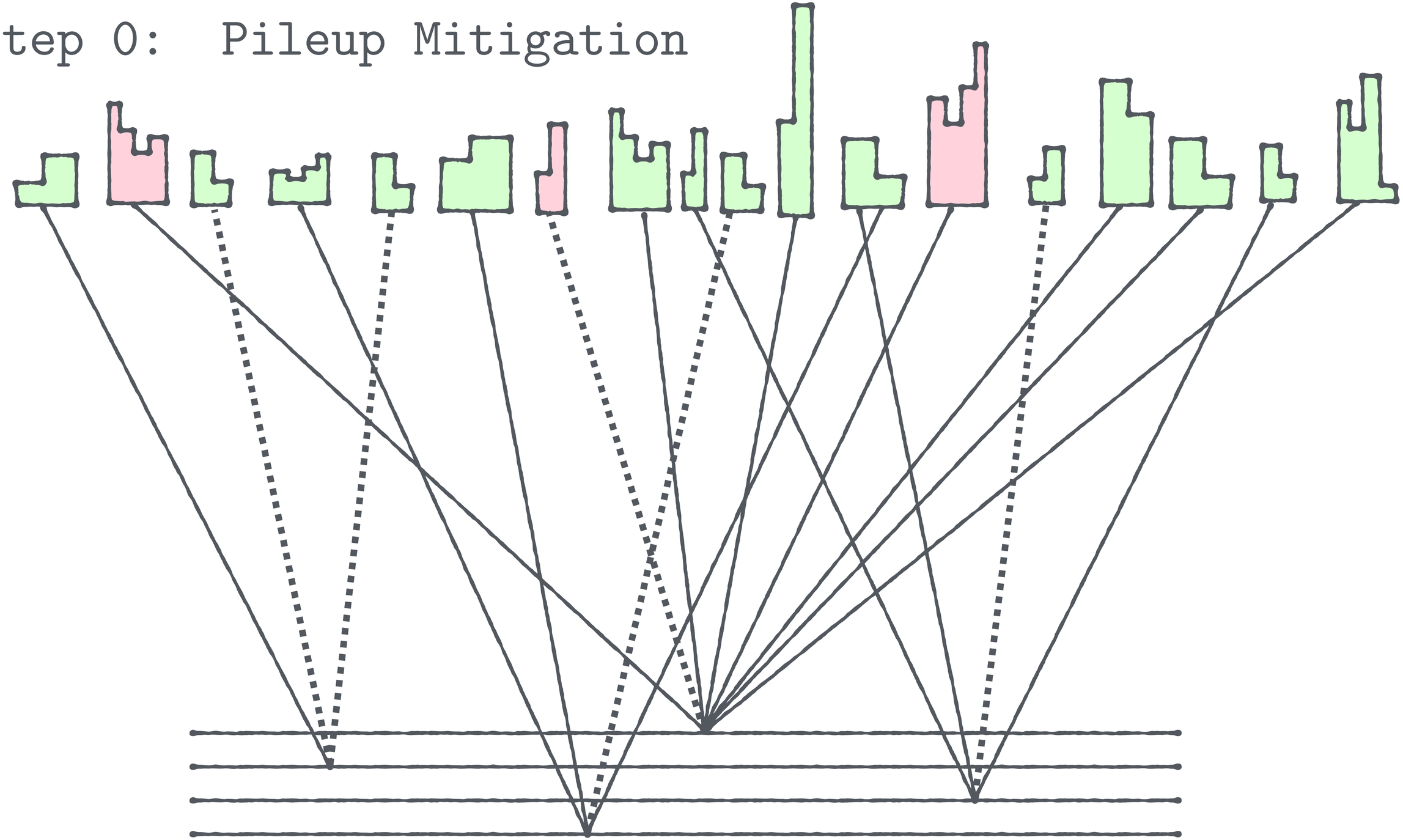
Step 0: Pileup Mitigation



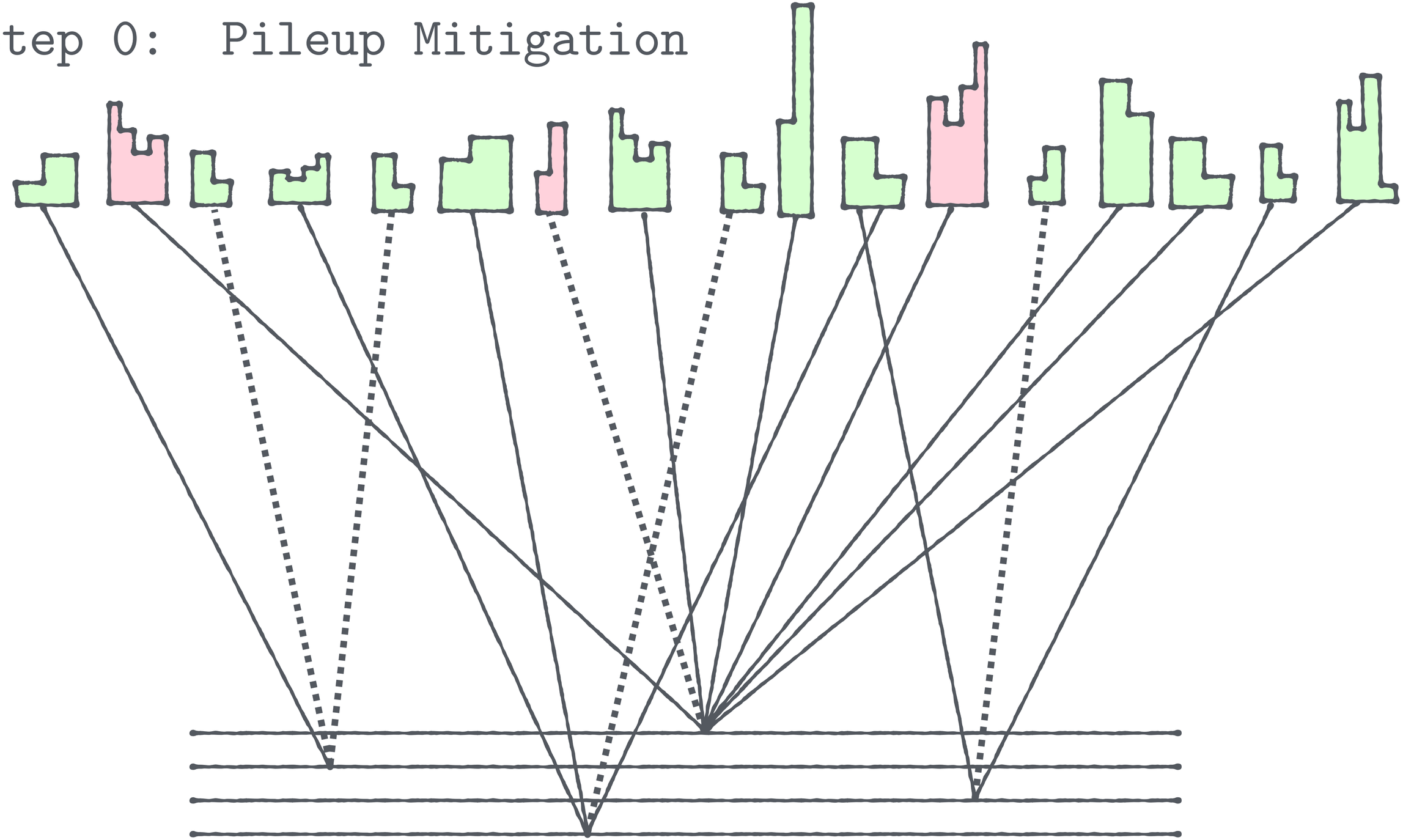
Step 0: Pileup Mitigation



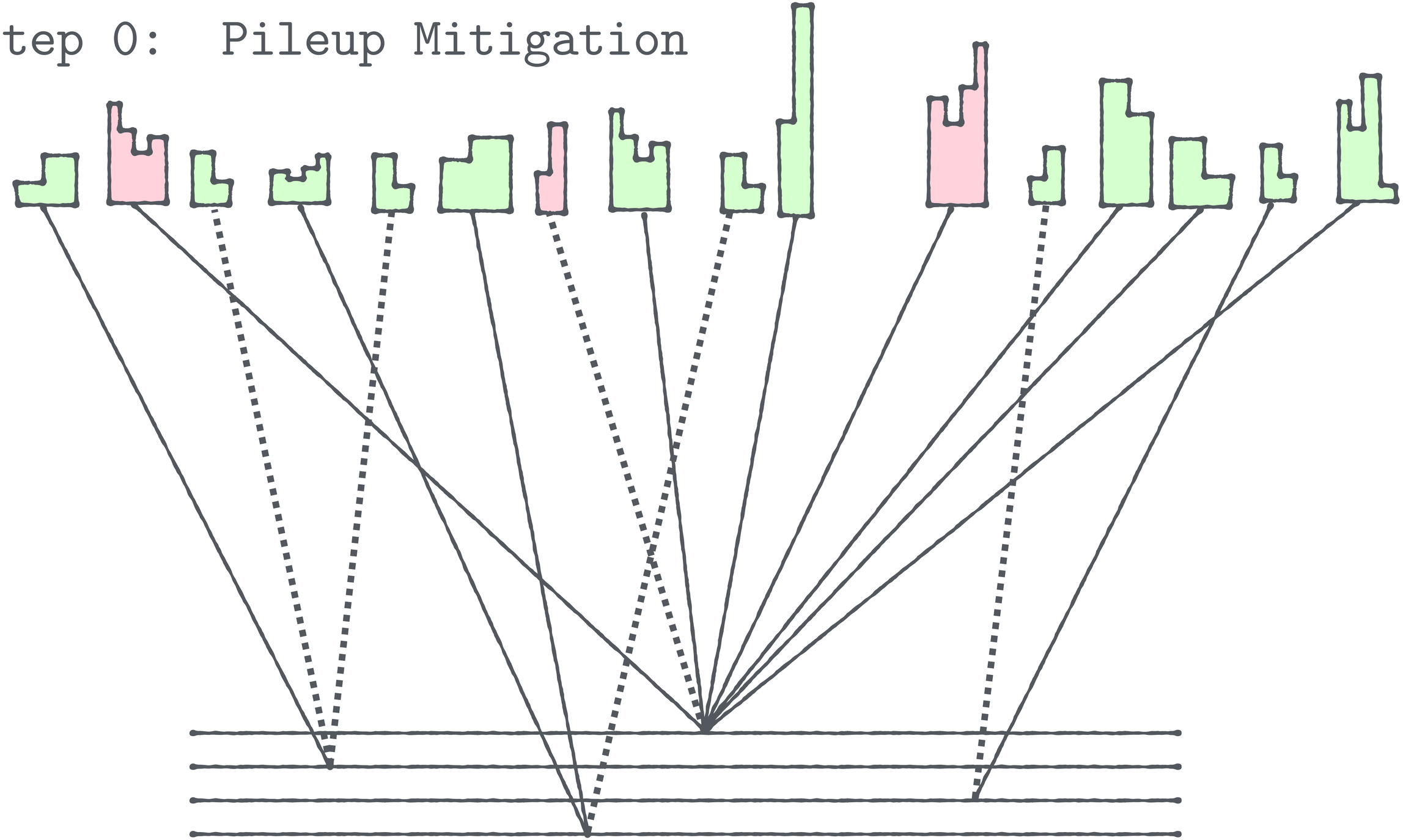
Step 0: Pileup Mitigation



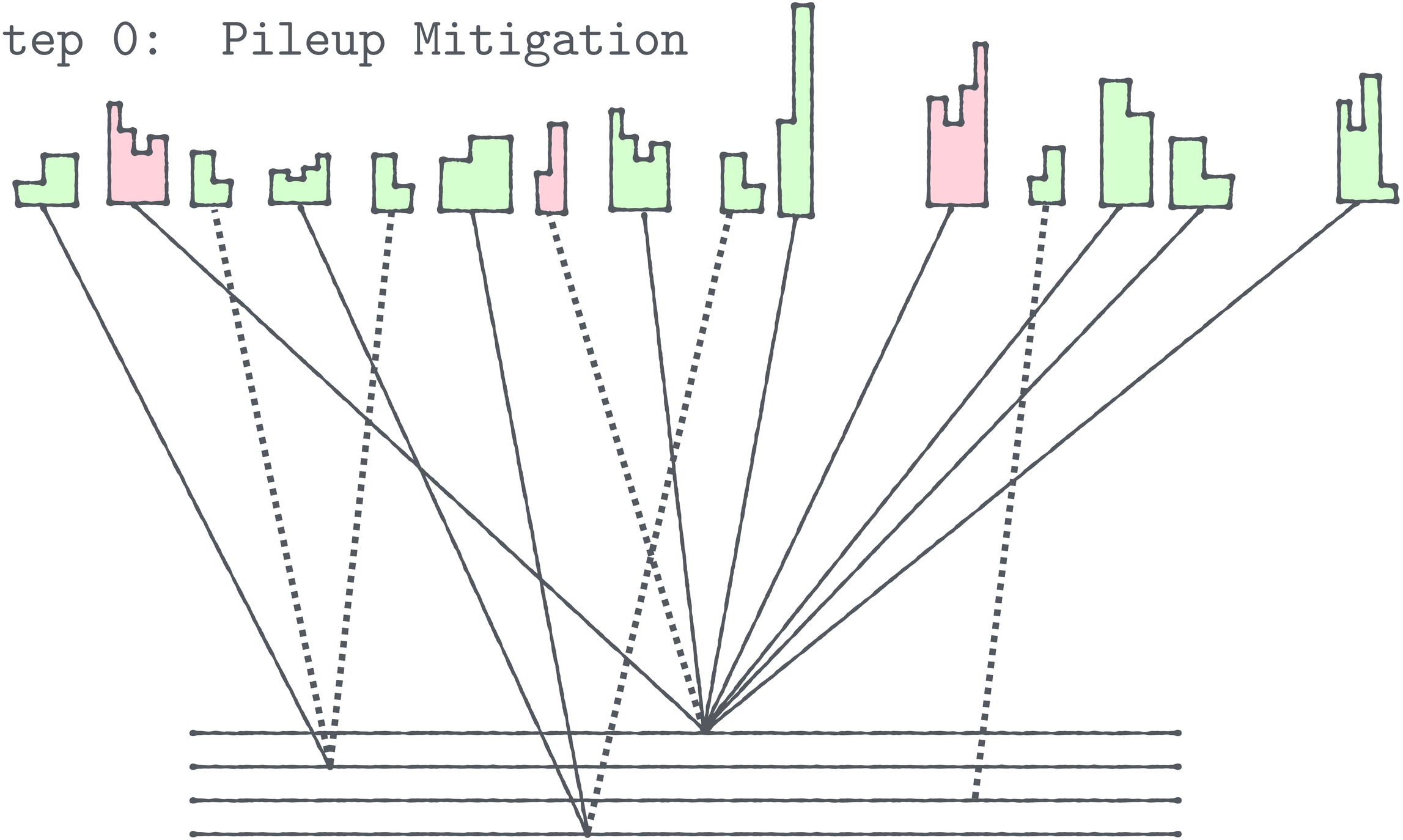
Step 0: Pileup Mitigation



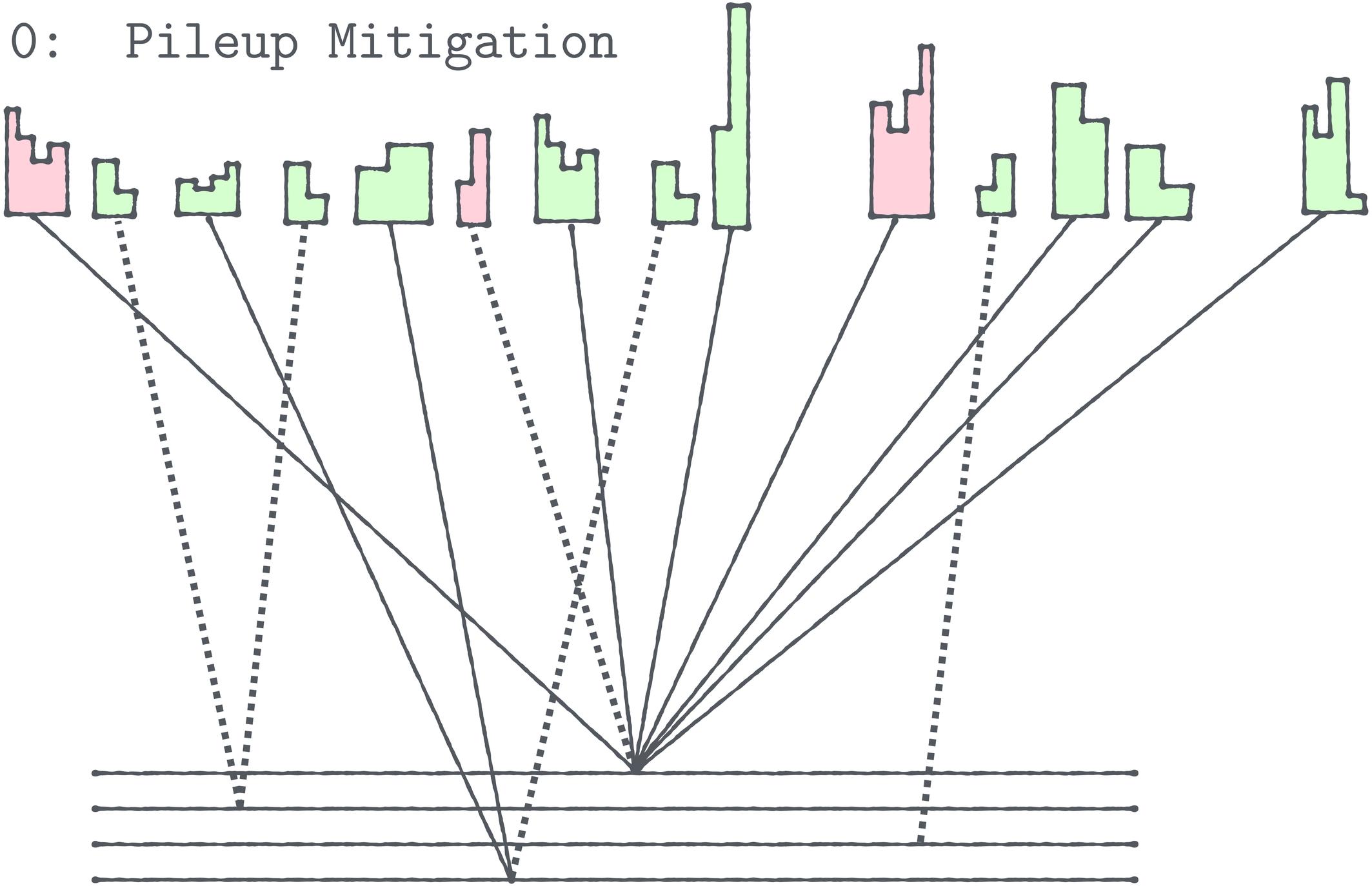
Step 0: Pileup Mitigation



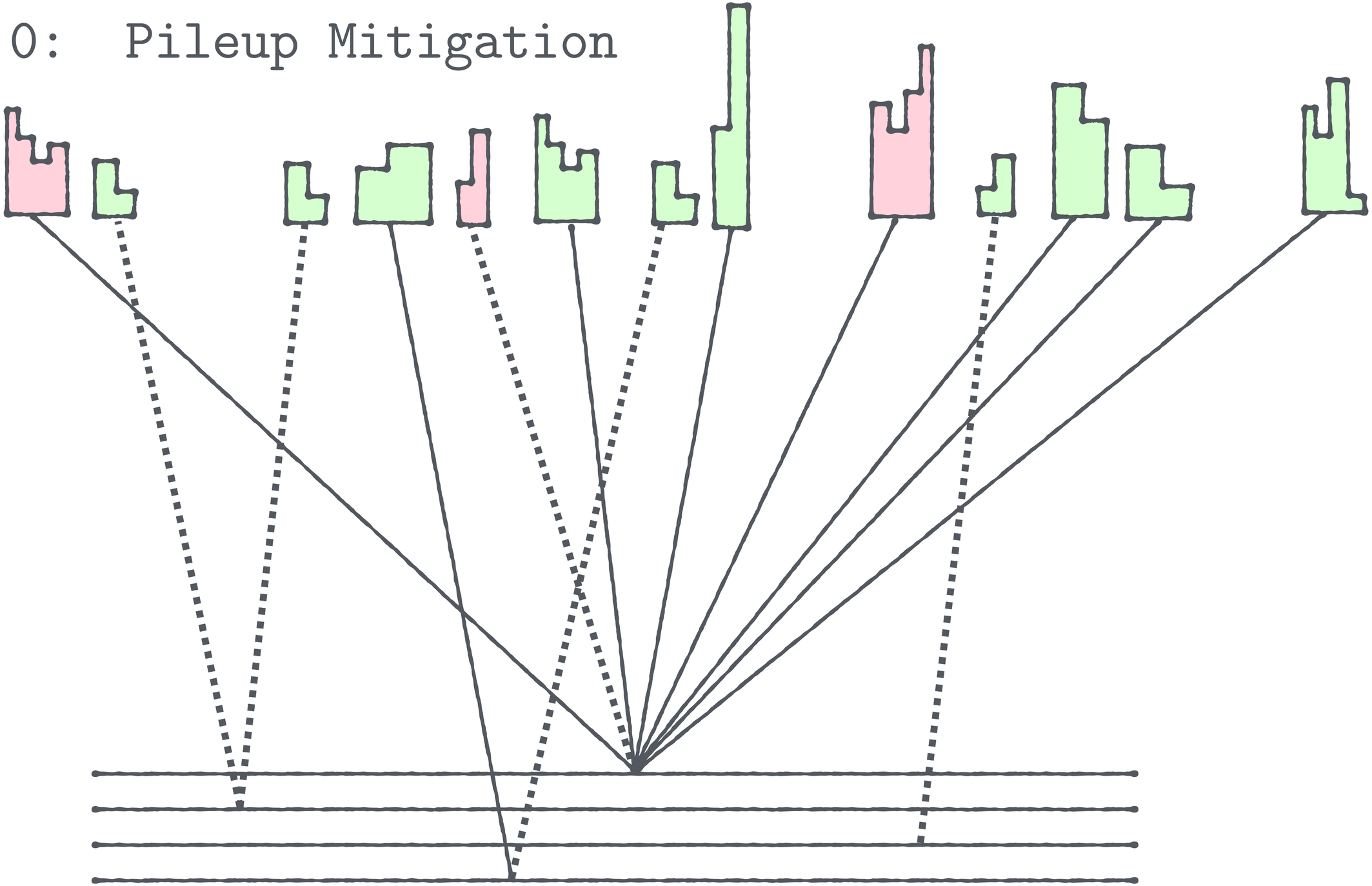
Step 0: Pileup Mitigation



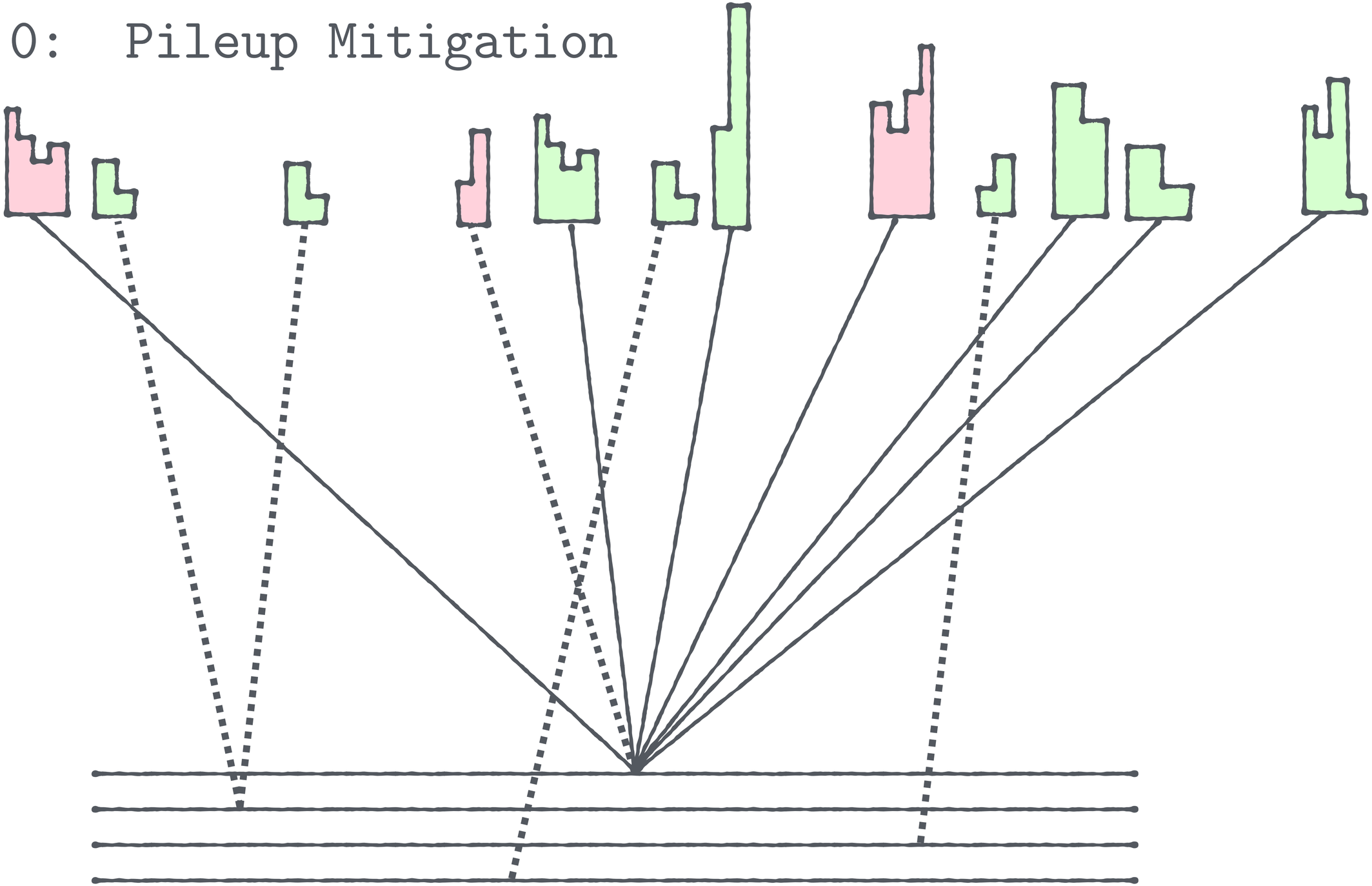
Step 0: Pileup Mitigation



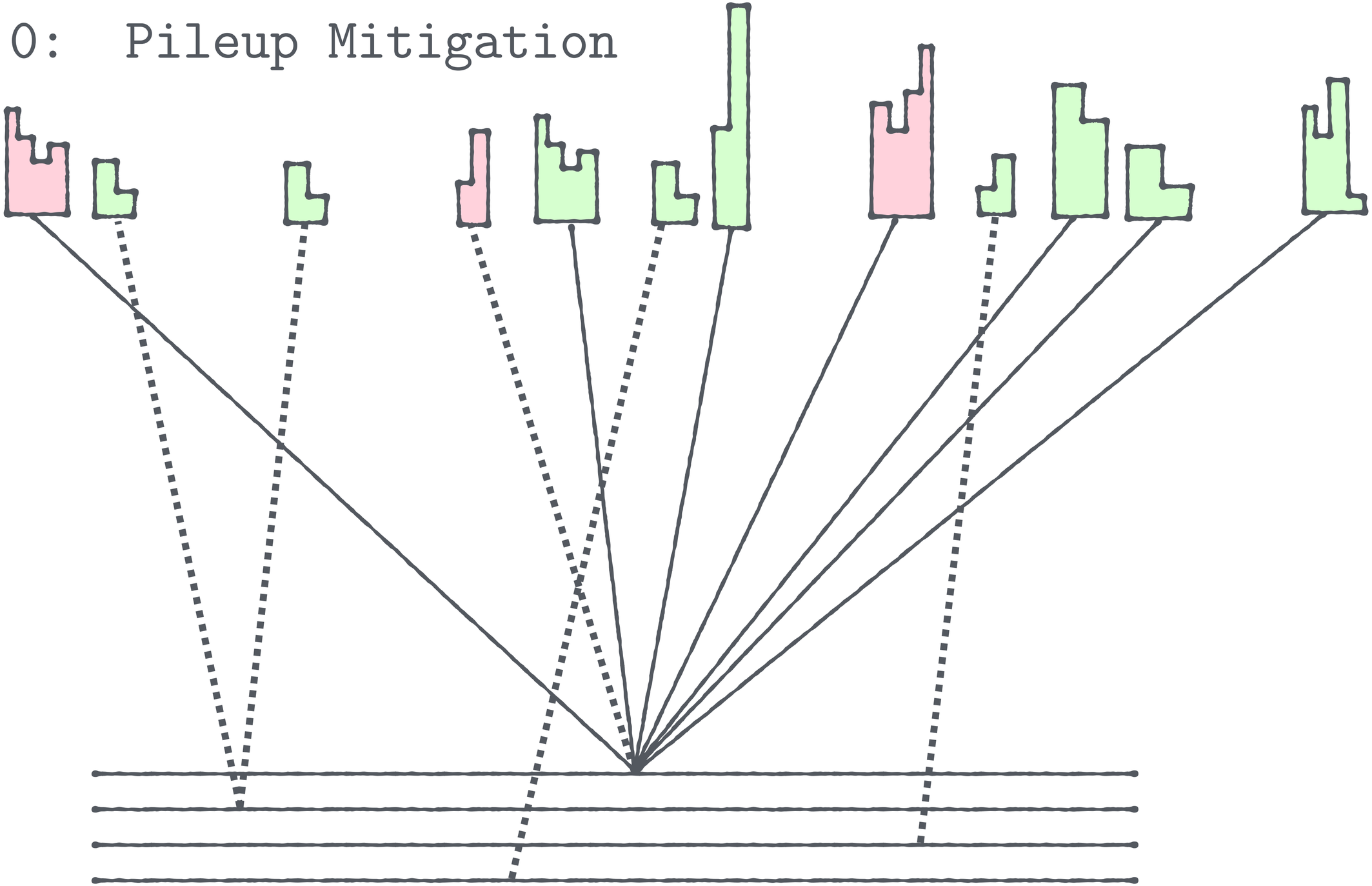
Step 0: Pileup Mitigation



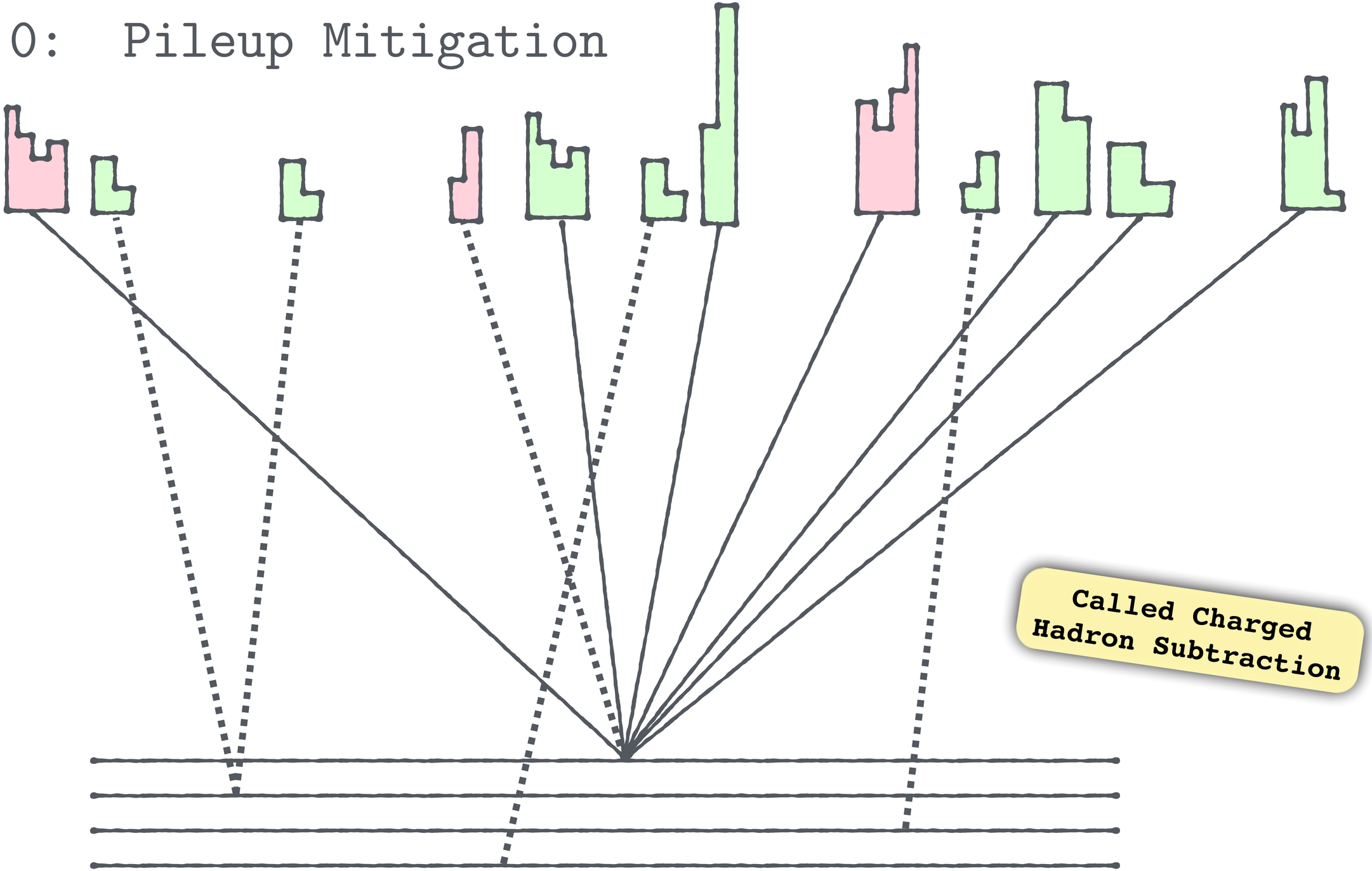
Step 0: Pileup Mitigation



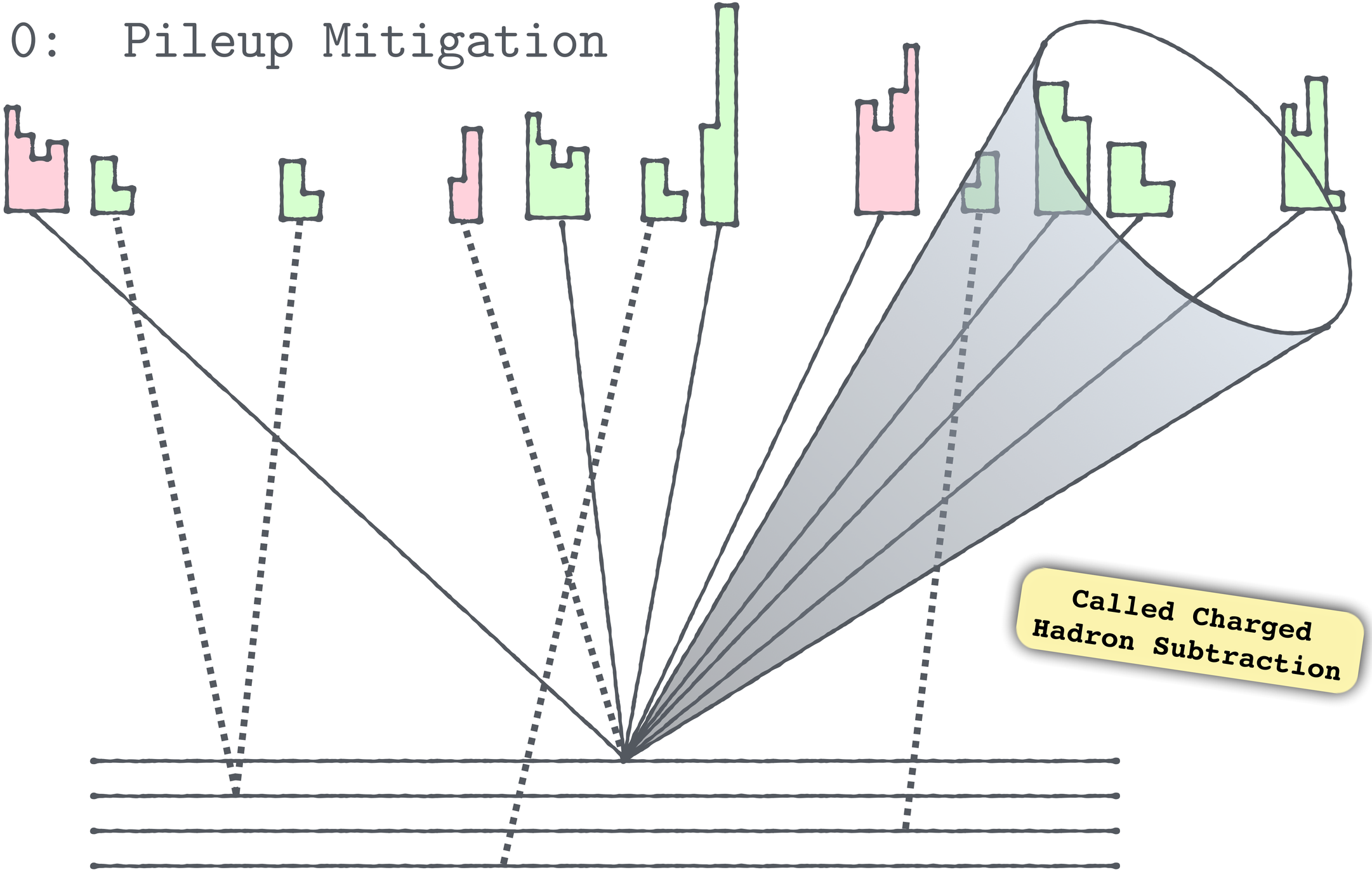
Step 0: Pileup Mitigation



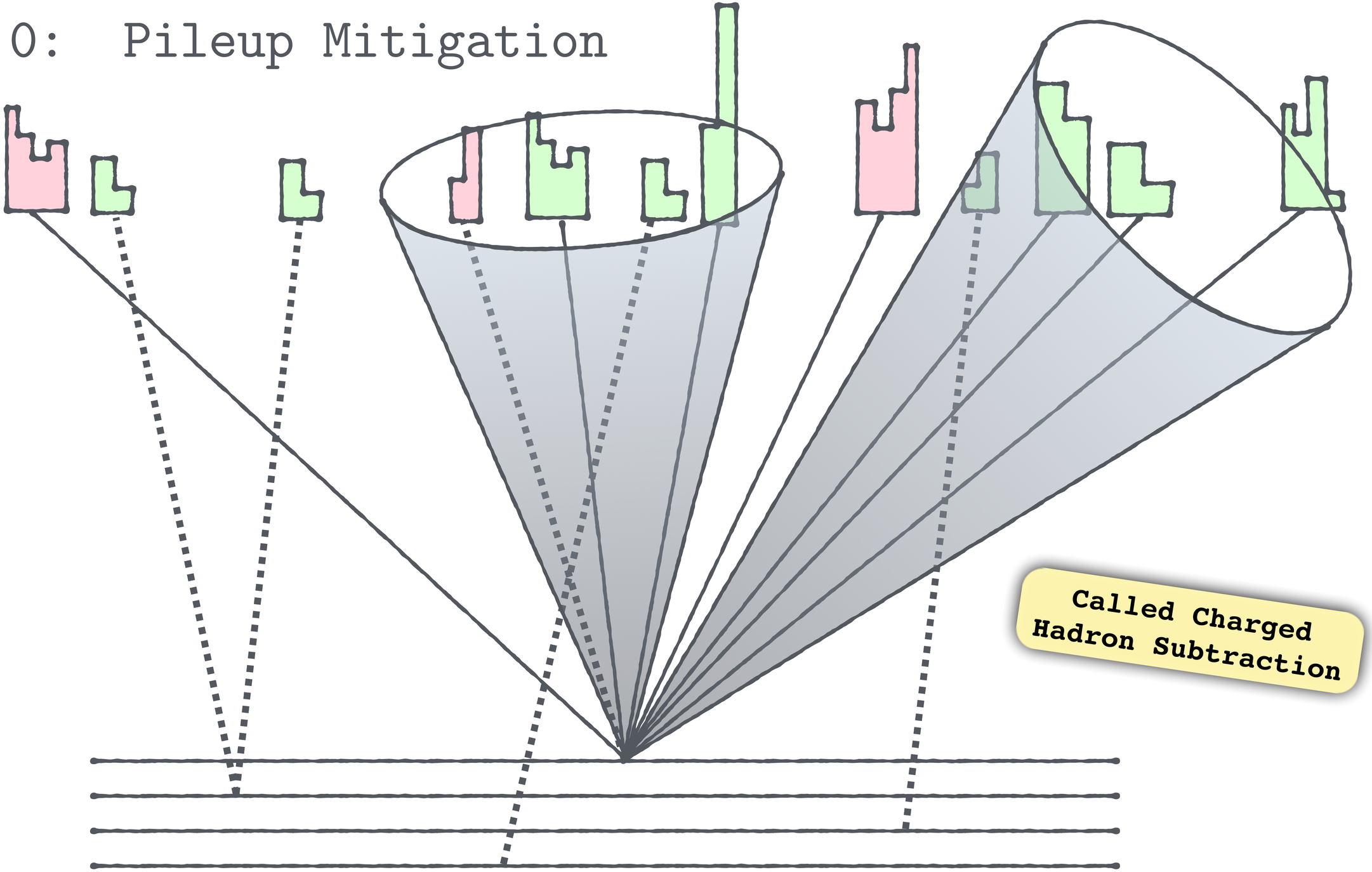
Step 0: Pileup Mitigation



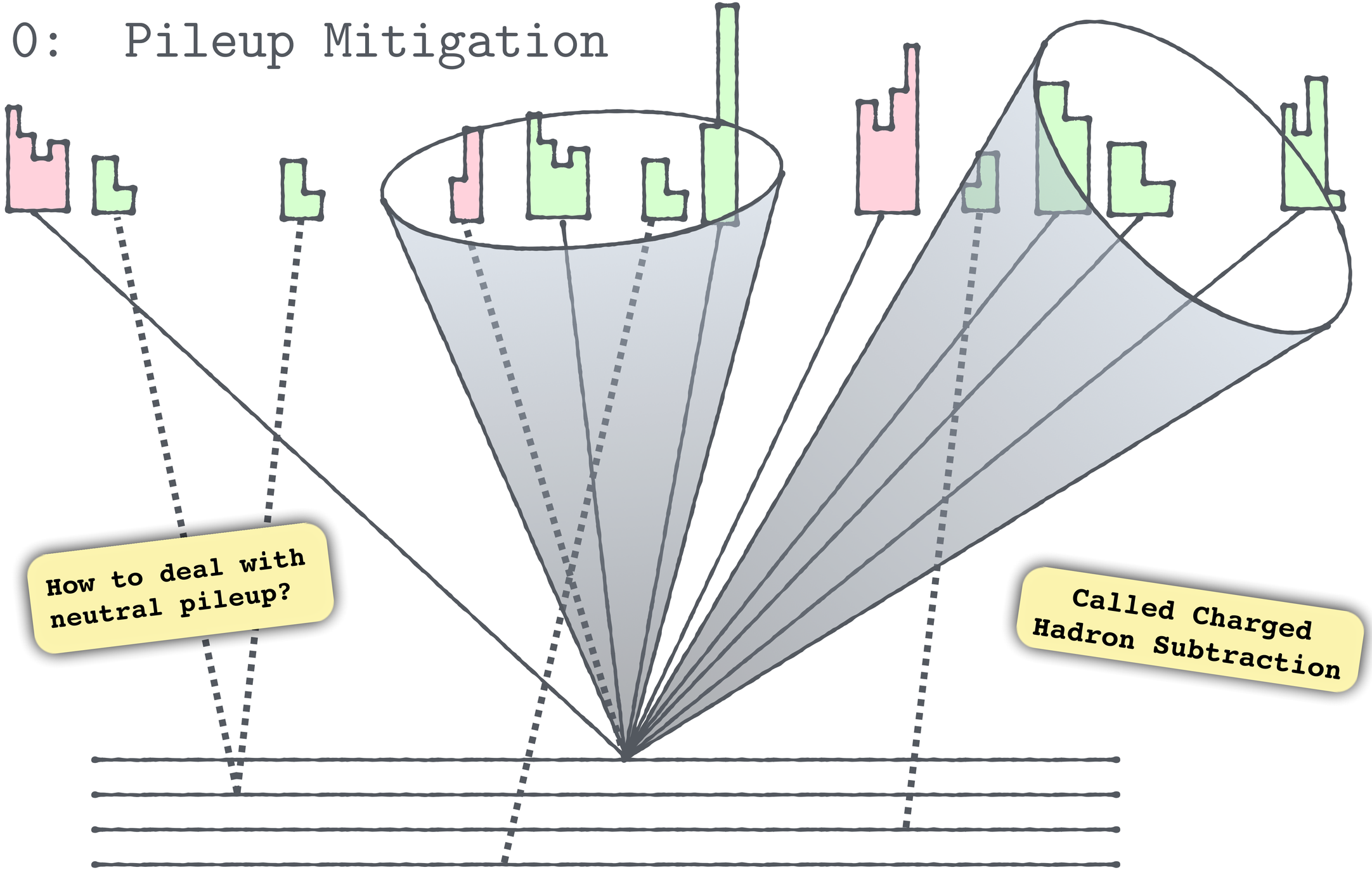
Step 0: Pileup Mitigation



Step 0: Pileup Mitigation



Step 0: Pileup Mitigation



Pile Up Per Particle Identification

Pile Up Per Particle Identification

Richard Cavanaugh, Fermilab/UIC, HCP Summer School 31 July 2024

Idea in a nutshell

- Particles from the leading vertex originate from a hard scattering process
- QCD Confinement: hadrons are produced in jets, which are collimated
  - leading vertex hadrons tend to be hard and not isolated
- Particles from pileup are soft and uncorrelated with the leading vertex hard process
  - PU particles tend to be soft and isolated

Idea in a nutshell

- Particles from the leading vertex originate from a hard scattering process
- QCD Confinement: hadrons are produced in jets, which are collimated
  - leading vertex hadrons tend to be hard and not isolated
- Particles from pileup are soft and uncorrelated with the leading vertex hard process
  - PU particles tend to be soft and isolated

Consider the following

$$\epsilon_{ij} \equiv \frac{p_{\mathrm{T},i}}{\Delta R_{ij}}$$

Encapsulates the basic idea:

(1) hard vs soft

(2) nearby vs far-away

Pile Up Per Particle Identification

Consider the following

$$\epsilon_{ij} \equiv \frac{p_{\mathrm{T},i}}{\Delta R_{ij}}$$

Encapsulates the basic idea:

(1) hard vs soft

(2) nearby vs far-away

Pile Up Per Particle Identification

Form a shape variable

$$\alpha_i \equiv \sum_j \frac{p_{\mathrm{T},i}}{\Delta R_{ij}} \Theta(R_{\mathrm{min}} < \Delta R_{ij} < R_0)$$

Consider the following

$$\epsilon_{ij} \equiv \frac{p_{\mathrm{T},i}}{\Delta R_{ij}}$$

Encapsulates the basic idea:

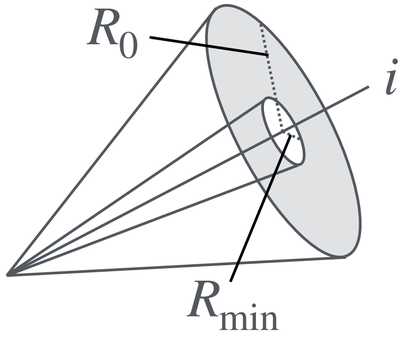
(1) hard vs soft

(2) nearby vs far-away

Pile Up Per Particle Identification

Form a shape variable

$$\alpha_i \equiv \sum_j \frac{p_{\mathrm{T},i}}{\Delta R_{ij}} \Theta(R_{\mathrm{min}} < \Delta R_{ij} < R_0)$$



Consider the following

$$\epsilon_{ij} \equiv \frac{p_{\mathrm{T},i}}{\Delta R_{ij}}$$

Encapsulates the basic idea:

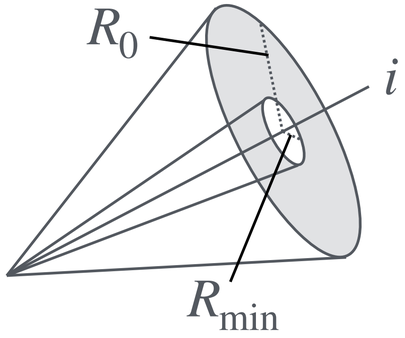
(1) hard vs soft

(2) nearby vs far-away

Pile Up Per Particle Identification

Form a shape variable

$$\alpha_i \equiv \sum_j \frac{p_{\mathrm{T},i}}{\Delta R_{ij}} \Theta(R_{\mathrm{min}} < \Delta R_{ij} < R_0)$$



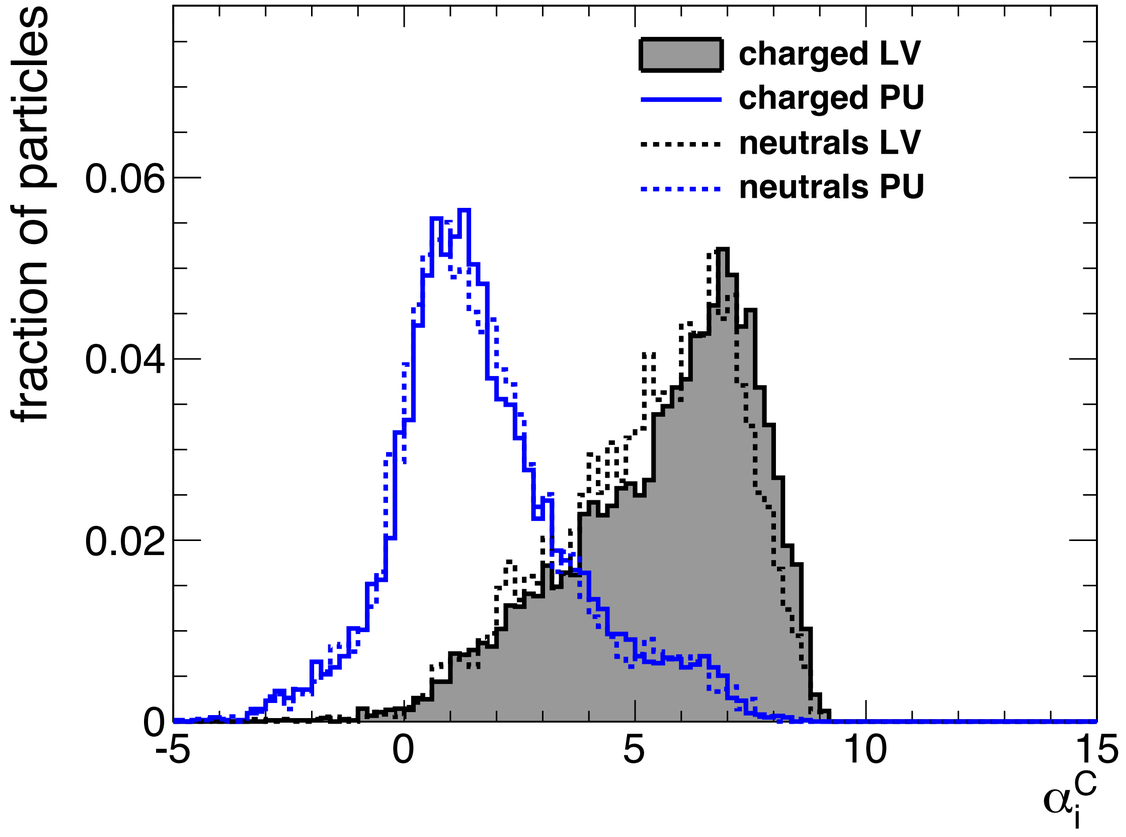
Consider the following

$$\epsilon_{ij} \equiv \frac{p_{\mathrm{T},i}}{\Delta R_{ij}}$$

Encapsulates the basic idea:

(1) hard vs soft

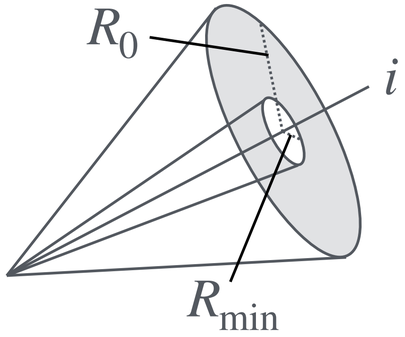
(2) nearby vs far-away

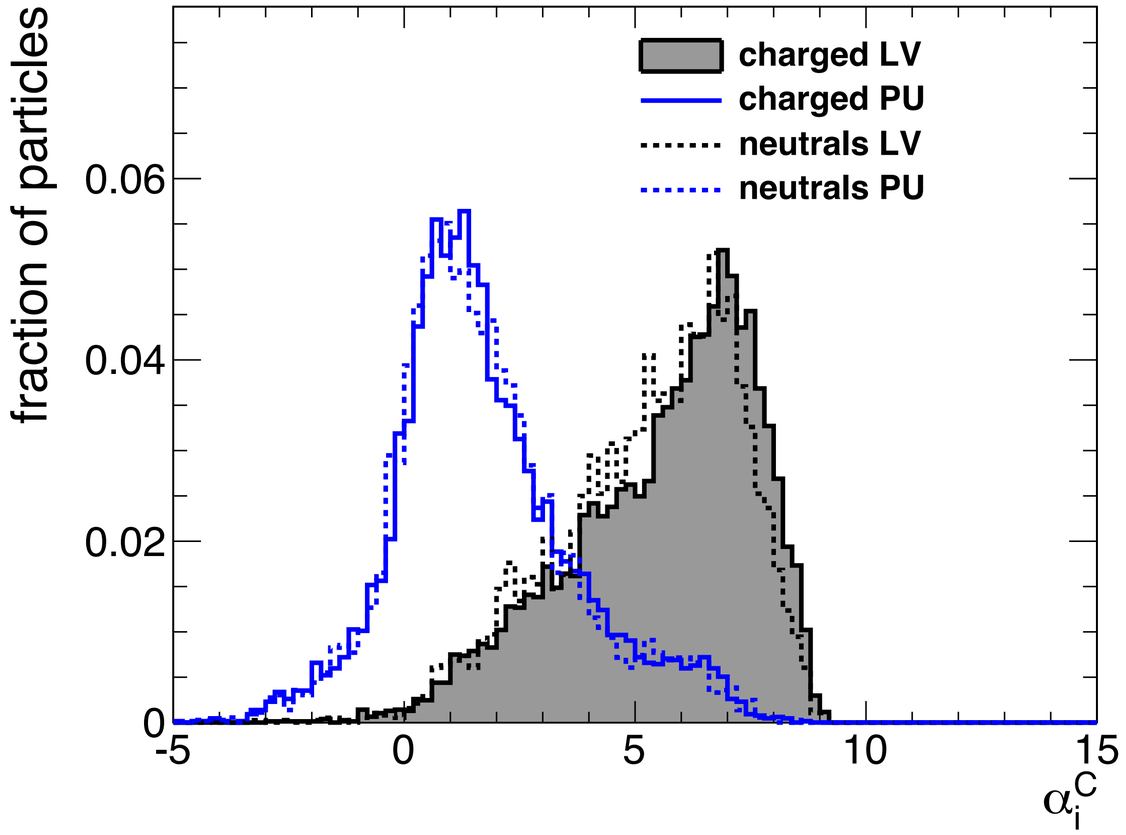


Pile Up Per Particle Identification

Form a shape variable

$$\alpha_i \equiv \sum_j \frac{p_{\mathrm{T},i}}{\Delta R_{ij}} \Theta(R_{\mathrm{min}} < \Delta R_{ij} < R_0)$$

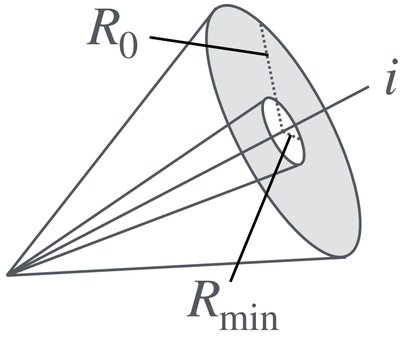


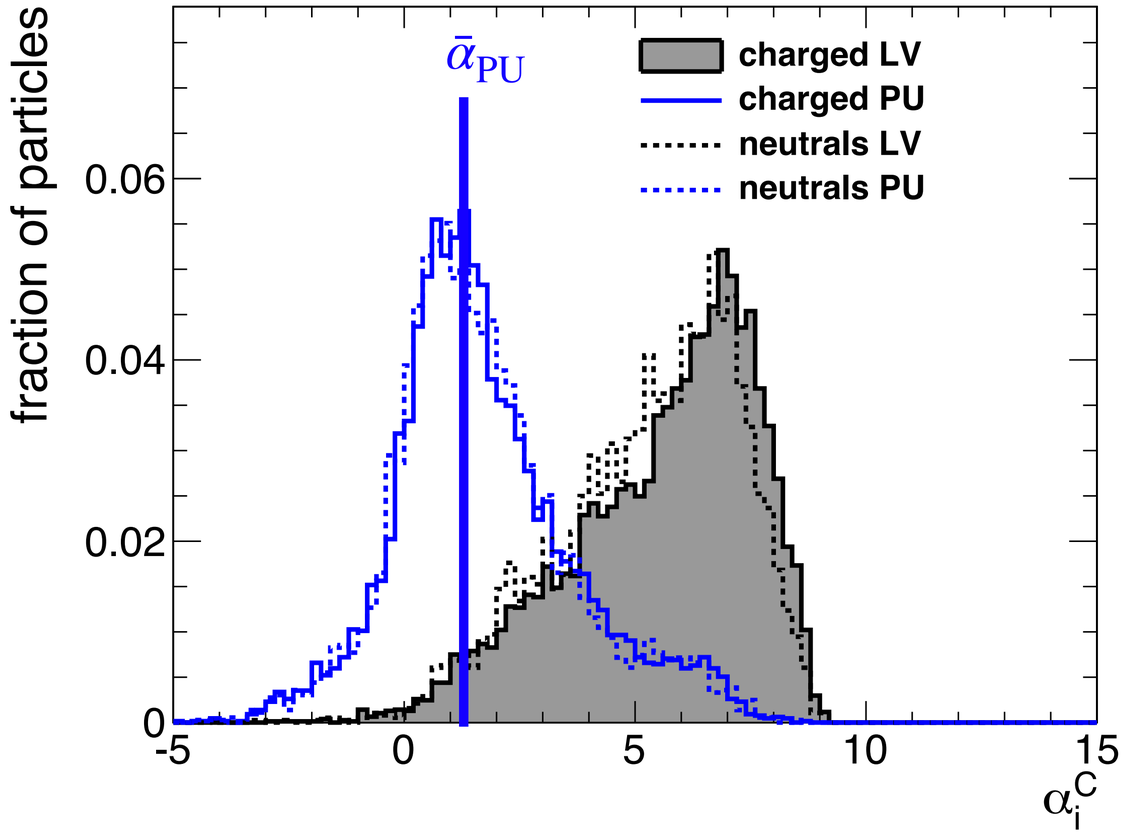


Pile Up Per Particle Identification

Form a shape variable

$$\alpha_i \equiv \sum_j \frac{p_{\mathrm{T},i}}{\Delta R_{ij}} \Theta(R_{\mathrm{min}} < \Delta R_{ij} < R_0)$$

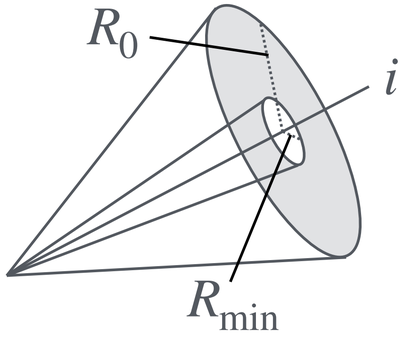


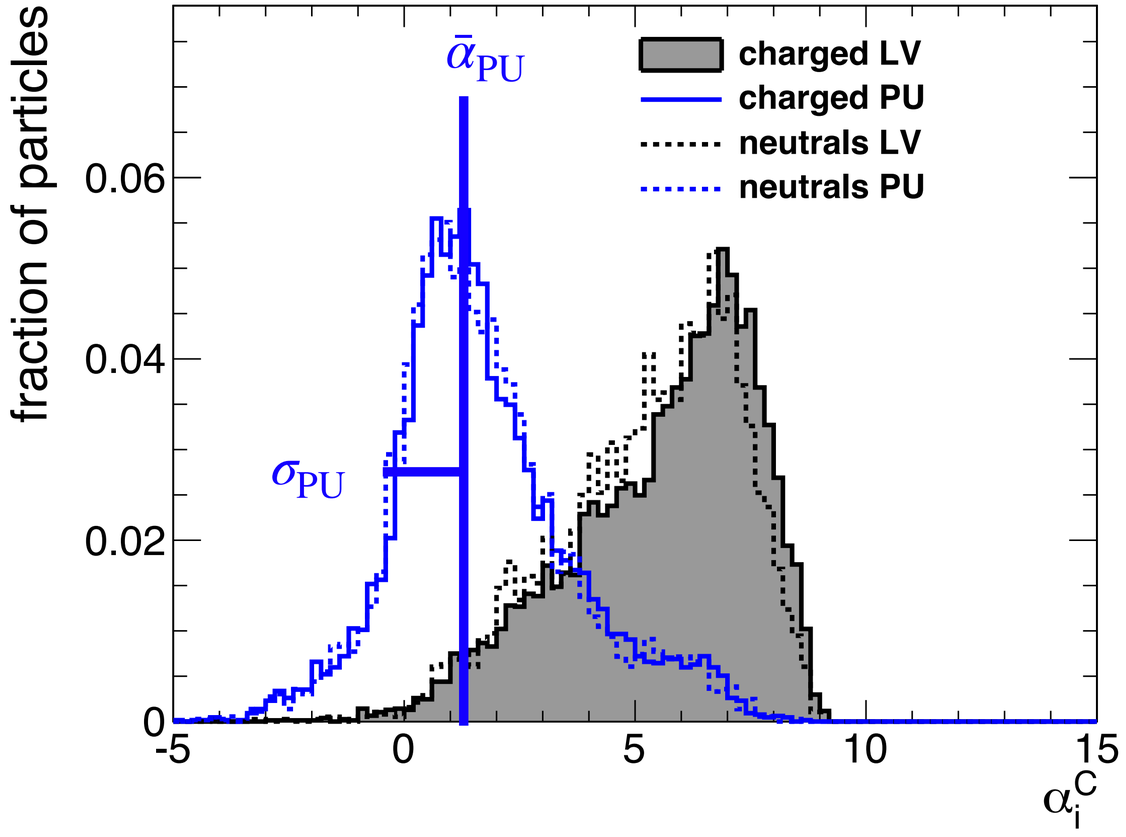


Pile Up Per Particle Identification

Form a shape variable

$$\alpha_i \equiv \sum_j \frac{p_{\mathrm{T},i}}{\Delta R_{ij}} \Theta(R_{\mathrm{min}} < \Delta R_{ij} < R_0)$$

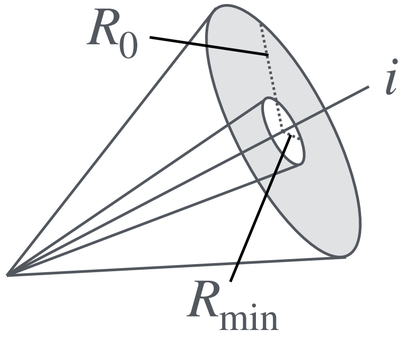




Pile Up Per Particle Identification

Form a shape variable

$$\alpha_i \equiv \sum_j \frac{p_{\mathrm{T},i}}{\Delta R_{ij}} \Theta(R_{\mathrm{min}} < \Delta R_{ij} < R_0)$$

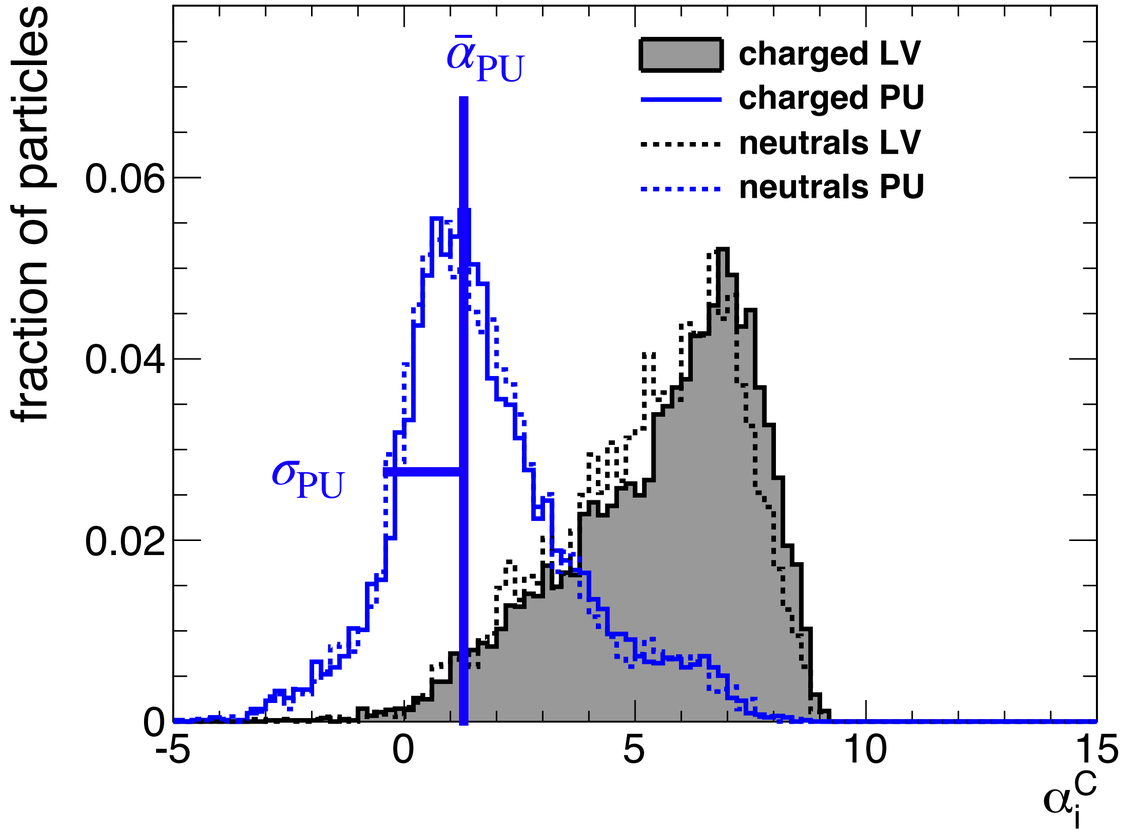


Now form a test statistic

$$\chi_i^2 \equiv \Theta(\alpha_i - \bar{\alpha}_{\mathrm{PU}}) \frac{(\alpha_i - \bar{\alpha}_{\mathrm{PU}})^2}{\sigma_{\mathrm{PU}}^2}$$

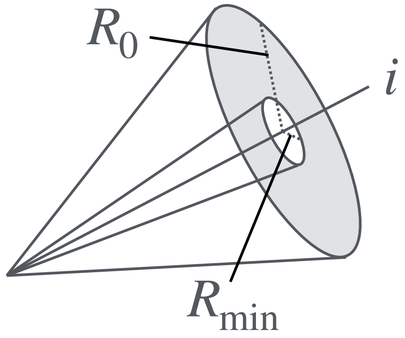
and assign a weight as

$$w_i \equiv F_{\chi^2,\mathrm{NDF}=1}(\chi_i^2)$$



Form a shape variable

$$\alpha_i \equiv \sum_j \frac{p_{\mathrm{T},i}}{\Delta R_{ij}} \Theta(R_{\mathrm{min}} < \Delta R_{ij} < R_0)$$

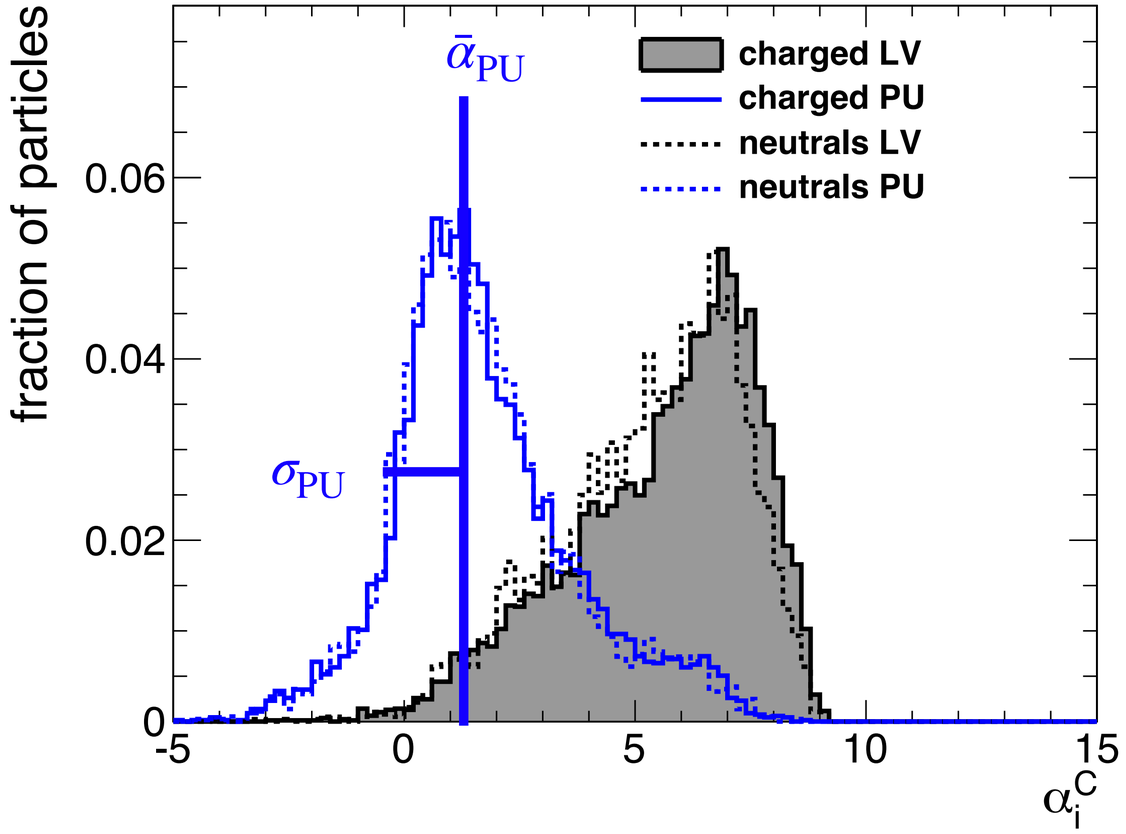


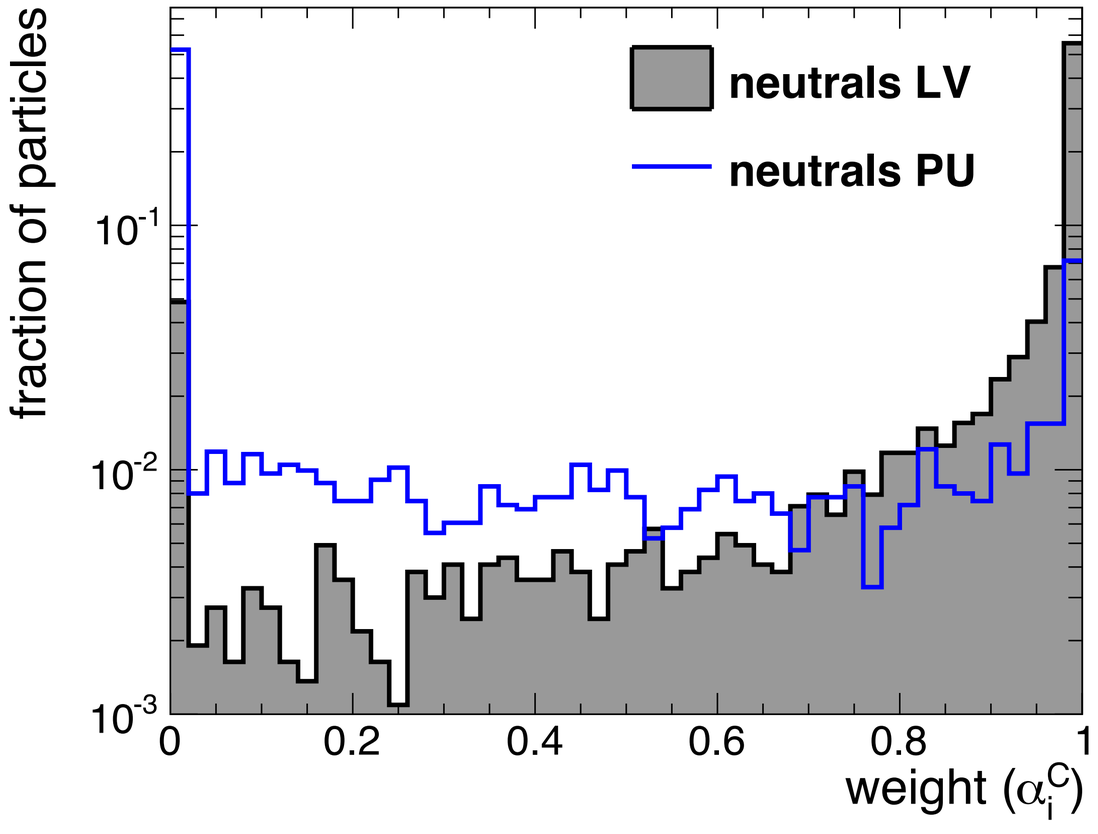
Now form a test statistic

$$\chi_i^2 \equiv \Theta(\alpha_i - \bar{\alpha}_{\mathrm{PU}}) \frac{(\alpha_i - \bar{\alpha}_{\mathrm{PU}})^2}{\sigma_{\mathrm{PU}}^2}$$

and assign a weight as

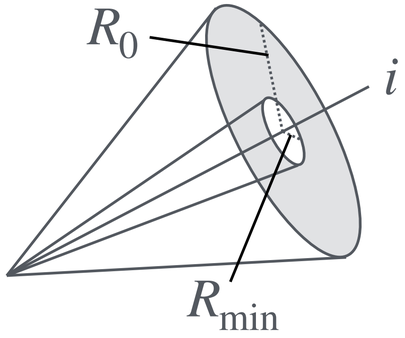
$$w_i \equiv F_{\chi^2,\mathrm{NDF}=1}(\chi_i^2)$$





Form a shape variable

$$\alpha_i \equiv \sum_j \frac{p_{\mathrm{T},i}}{\Delta R_{ij}} \Theta(R_{\mathrm{min}} < \Delta R_{ij} < R_0)$$



Now form a test statistic

$$\chi^2_i \equiv \Theta(\alpha_i - \bar{\alpha}_{\mathrm{PU}}) \frac{(\alpha_i - \bar{\alpha}_{\mathrm{PU}})^2}{\sigma^2_{\mathrm{PU}}}$$

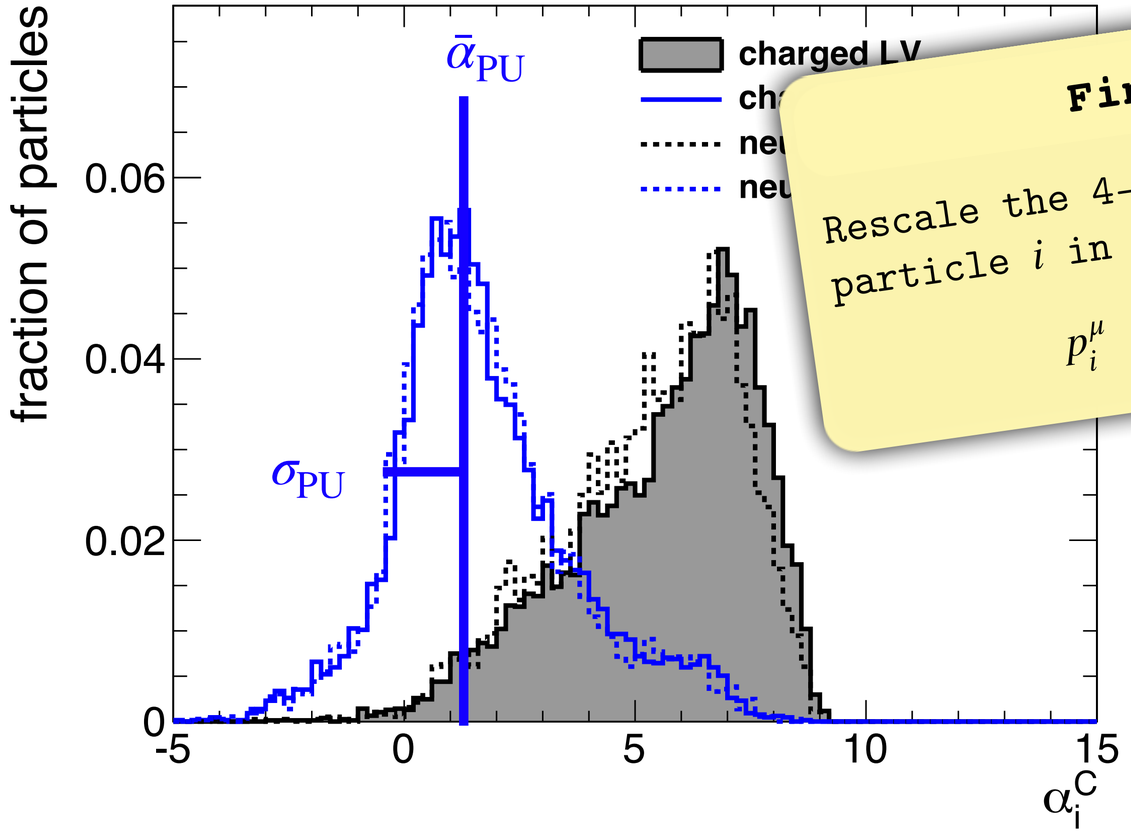
and assign a weight as

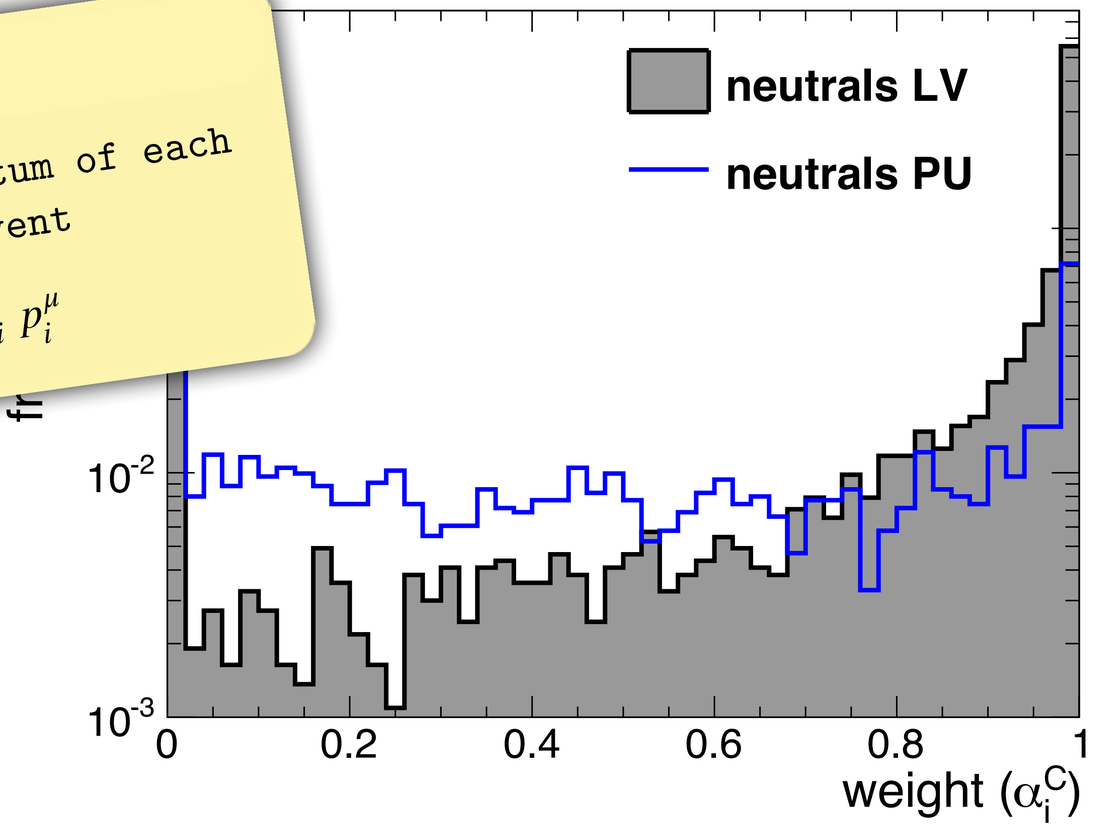
$$w_i \equiv F_{\chi^2,\mathrm{NDF}=1}(\chi^2_i)$$

Finally

Rescale the 4-momentum of each particle i in the event

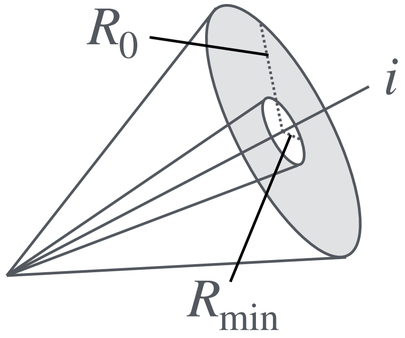
$$p^\mu_i \rightarrow w_i \, p^\mu_i$$





Form a shape variable

$$\alpha_i \equiv \sum_j \frac{p_{\mathrm{T},i}}{\Delta R_{ij}} \Theta(R_{\mathrm{min}} < \Delta R_{ij} < R_0)$$

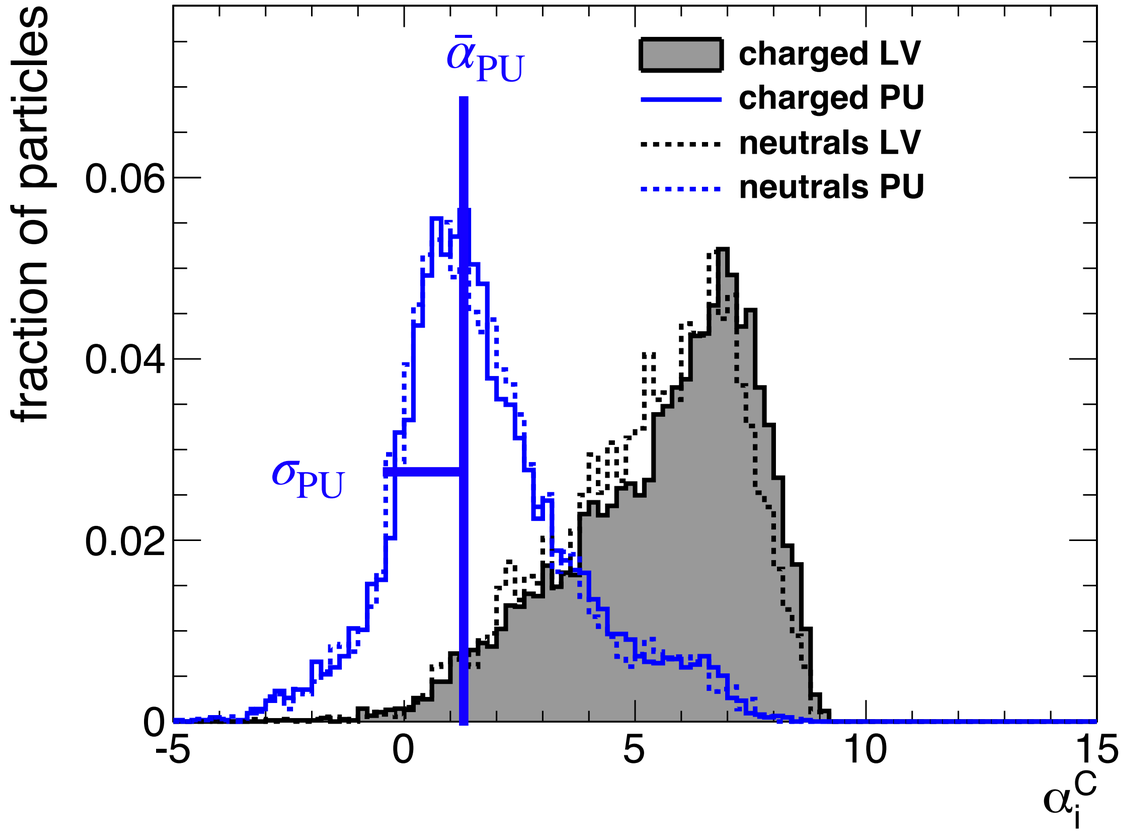


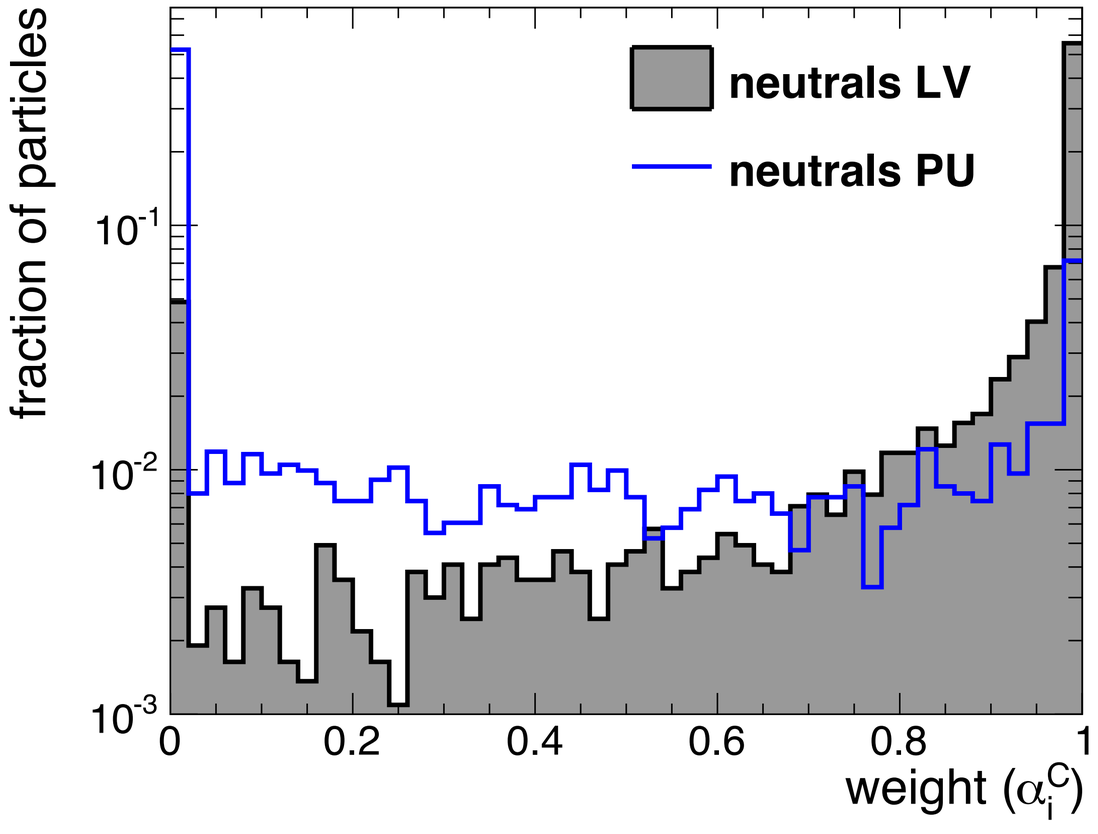
Now form a test statistic

$$\chi_i^2 \equiv \Theta(\alpha_i - \bar{\alpha}_{\mathrm{PU}}) \frac{(\alpha_i - \bar{\alpha}_{\mathrm{PU}})^2}{\sigma_{\mathrm{PU}}^2}$$

and assign a weight as

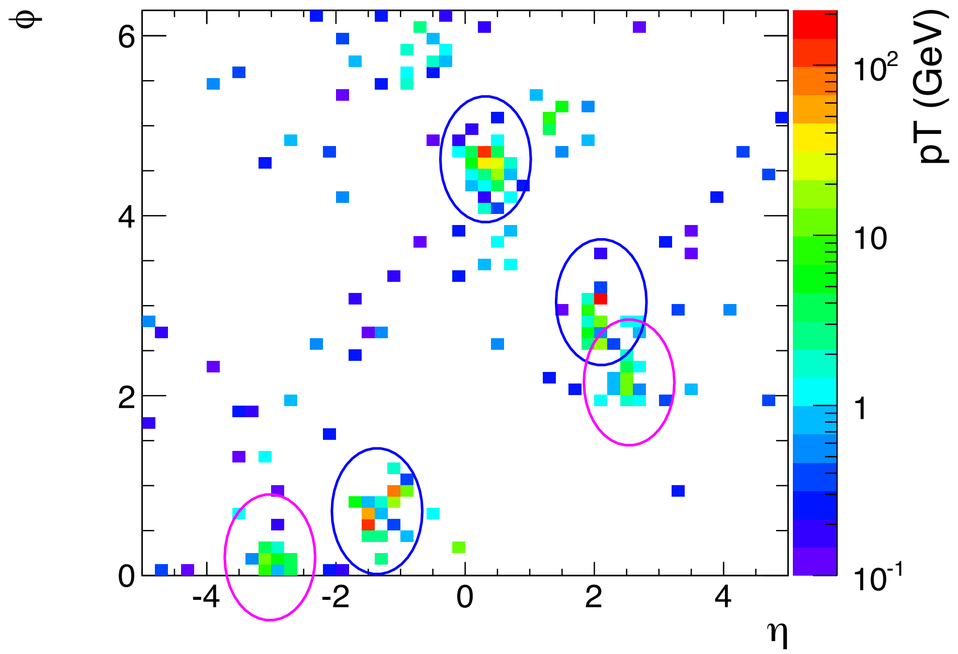
$$w_i \equiv F_{\chi^2,\mathrm{NDF}=1}(\chi_i^2)$$



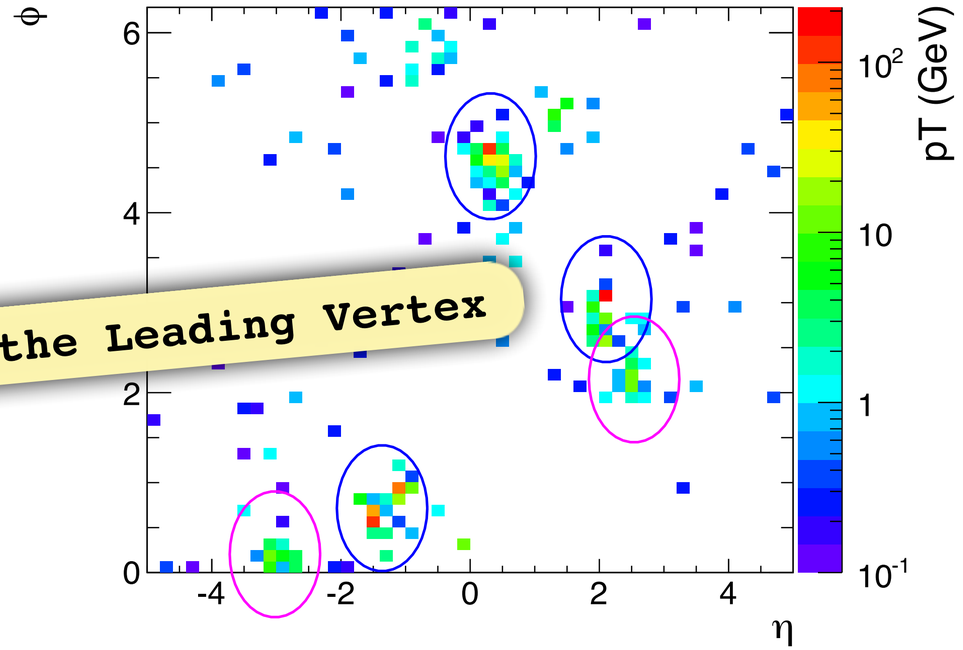


Pile Up Per Particle Identification

Pile Up Per Particle Identification

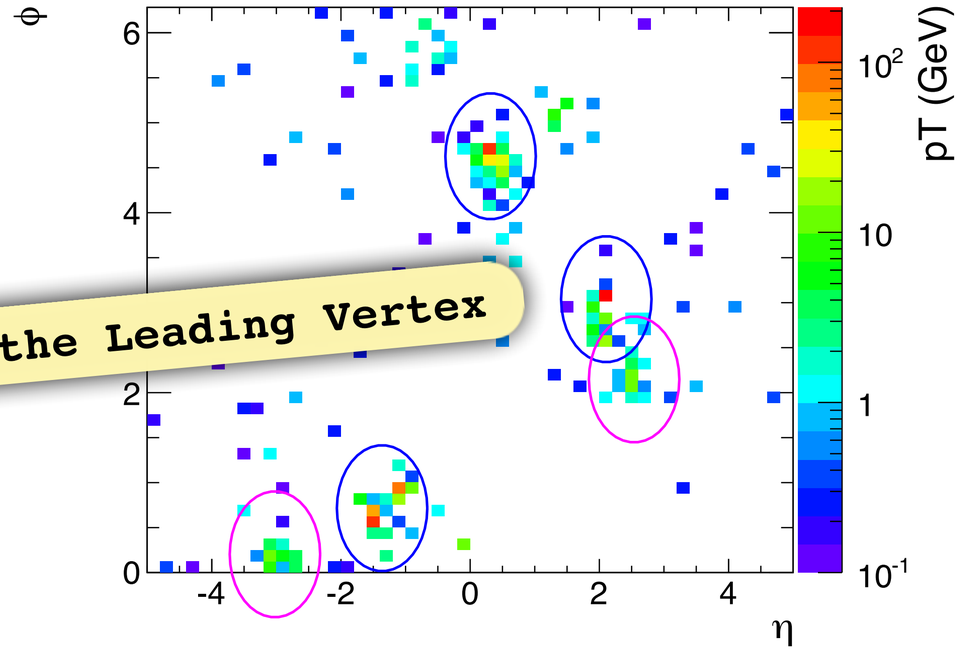


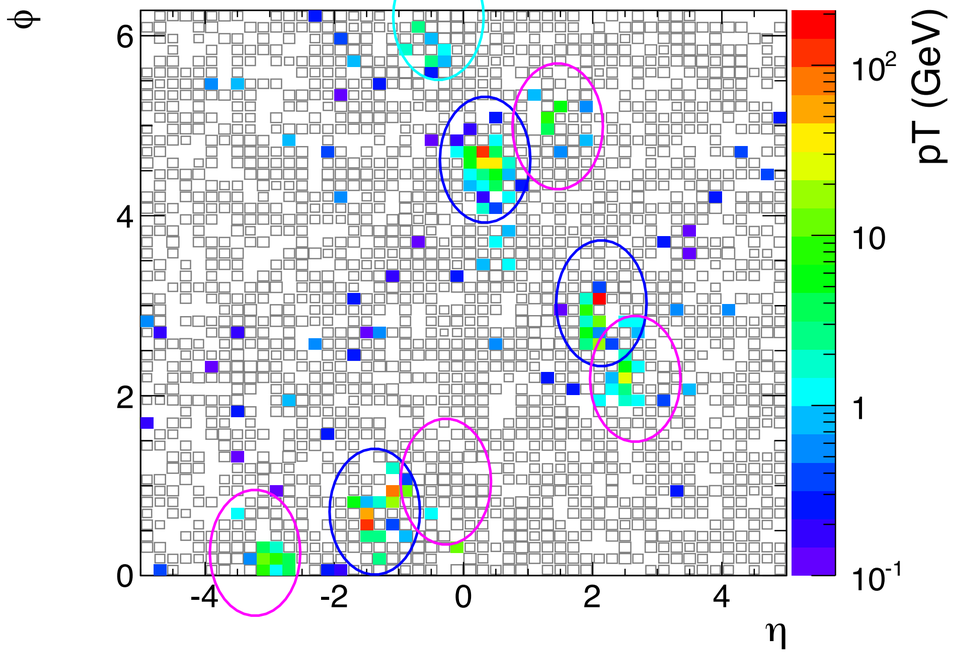
Pile Up Per Particle Identification



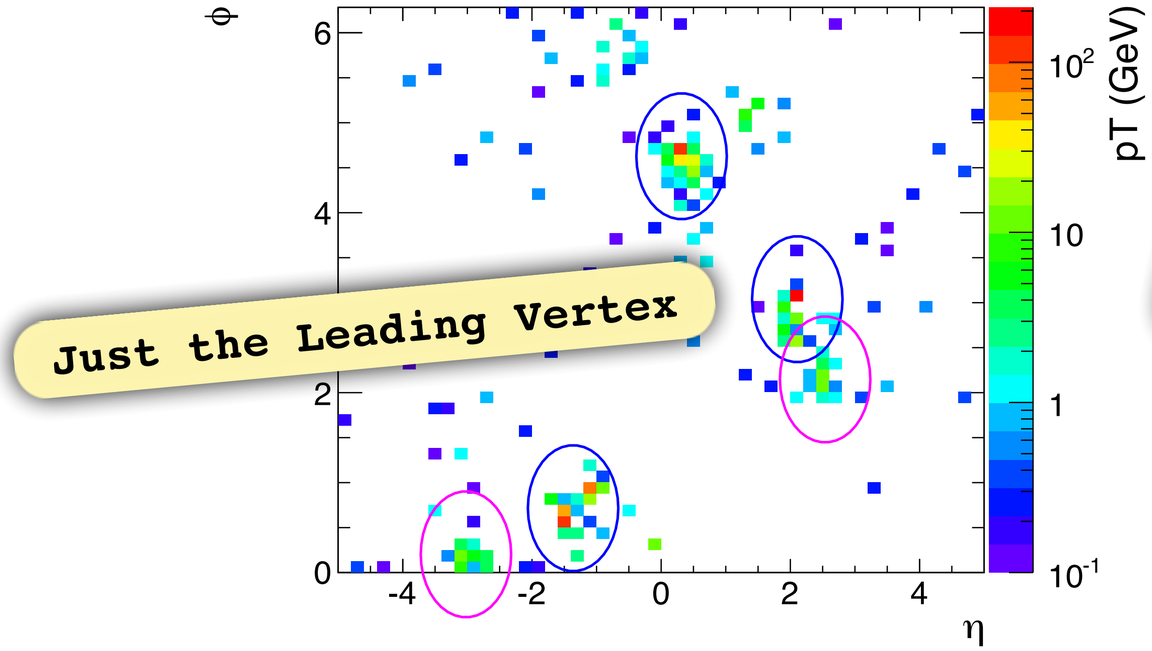
Just the Leading Vertex

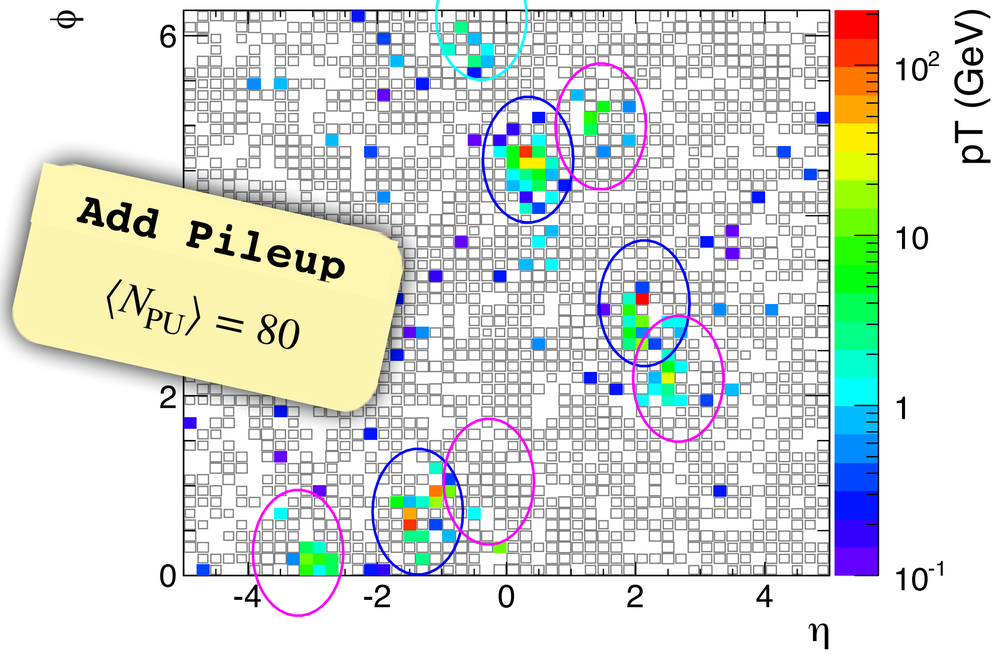
Pile Up Per Particle Identification



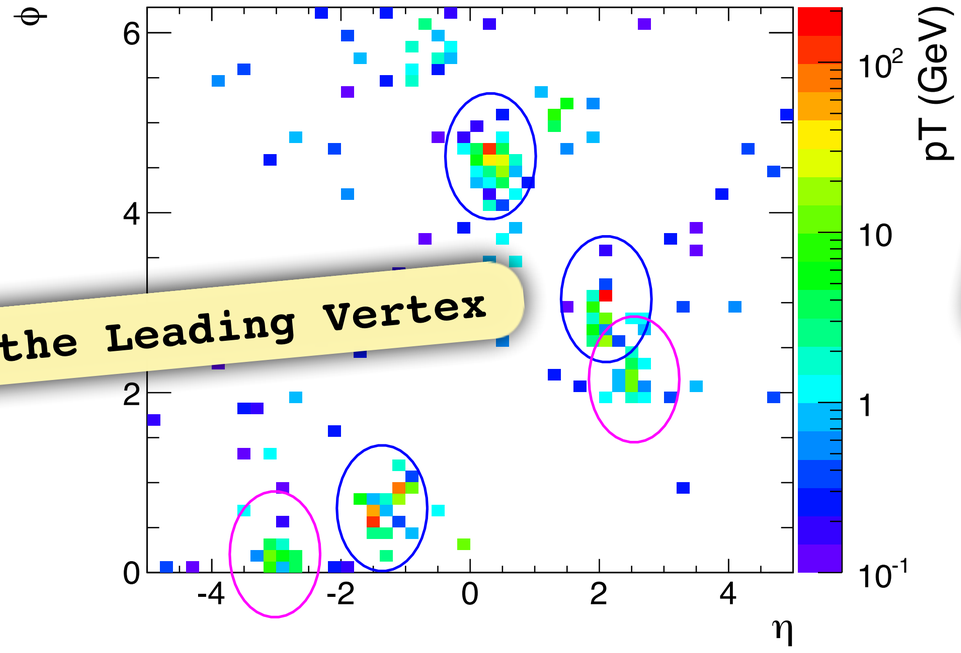


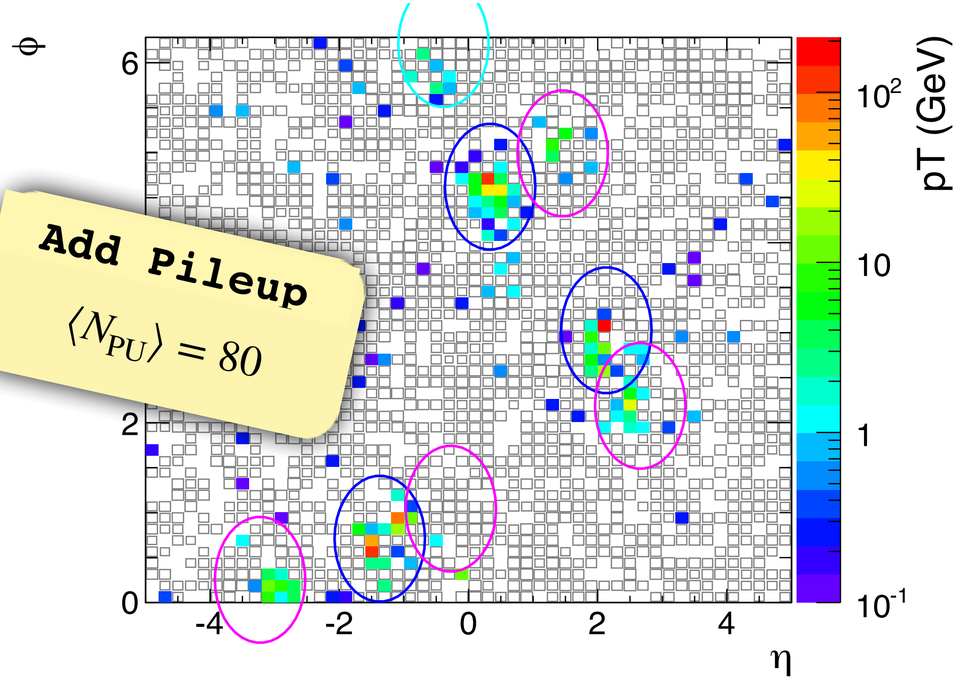
Pile Up Per Particle Identification

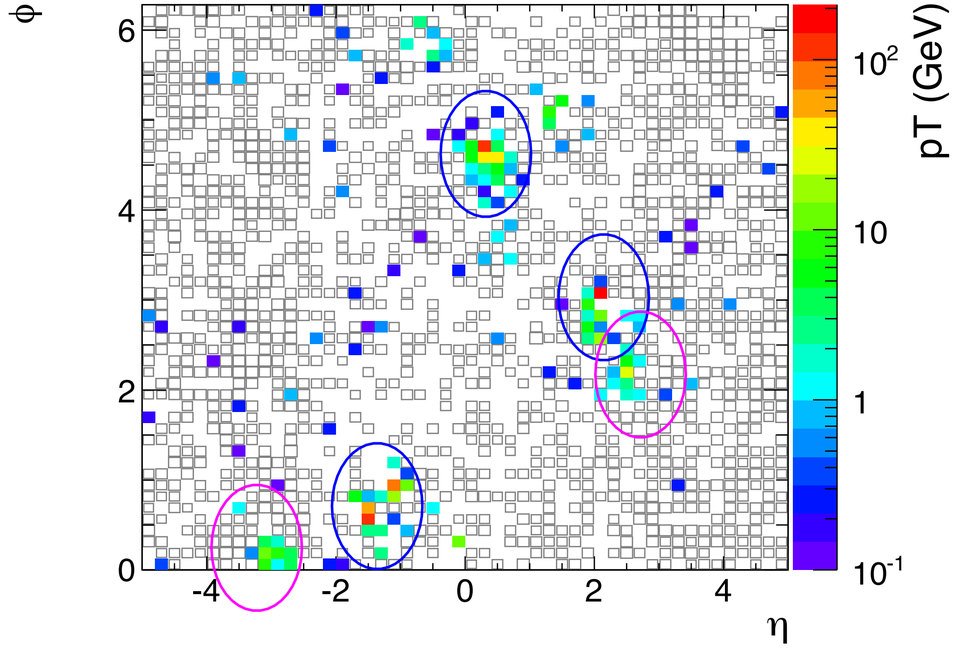




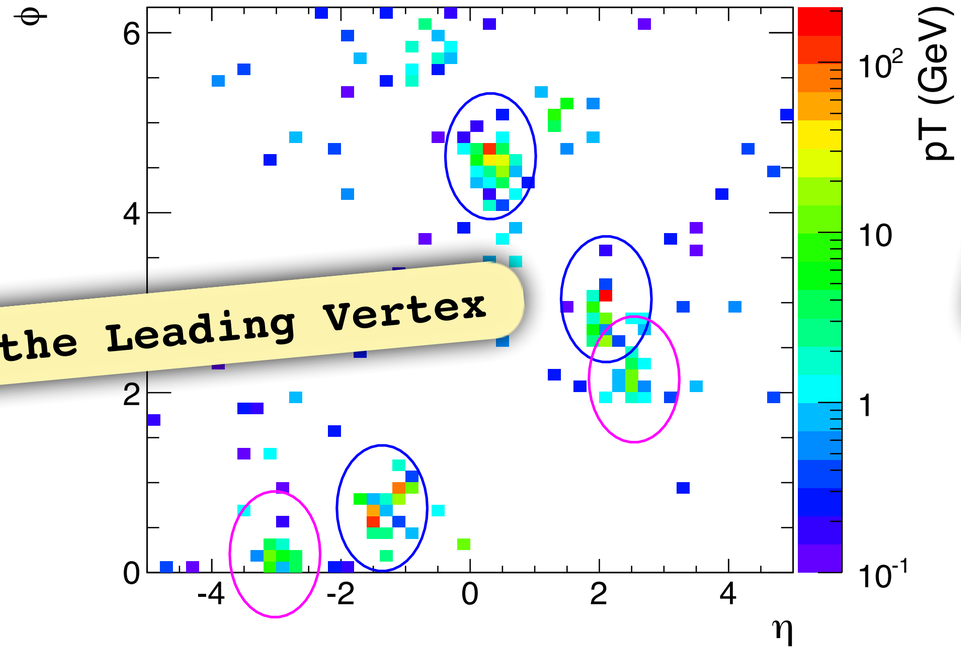
Pile Up Per Particle Identification

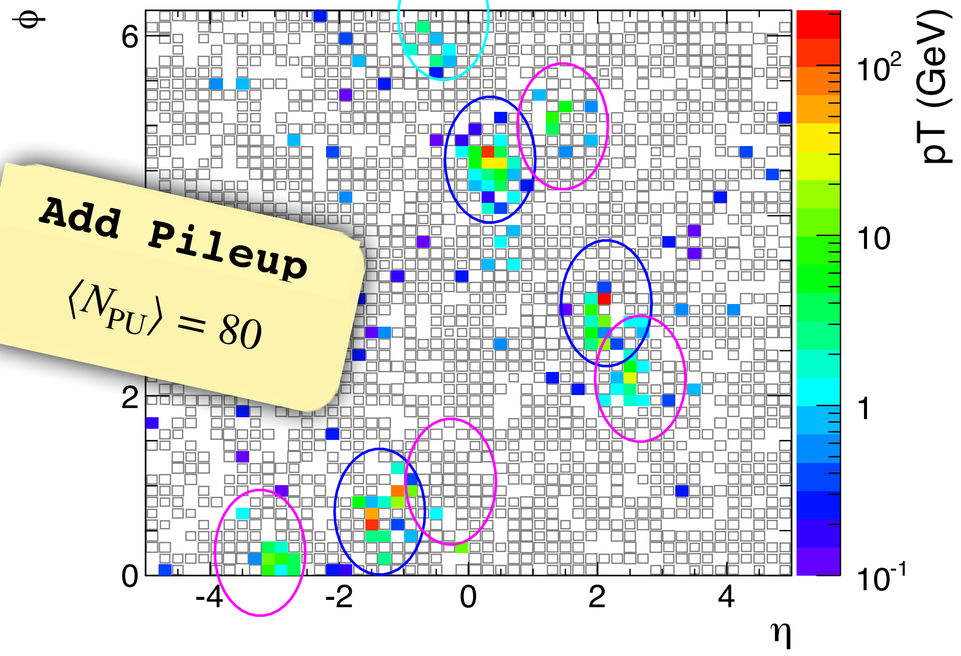


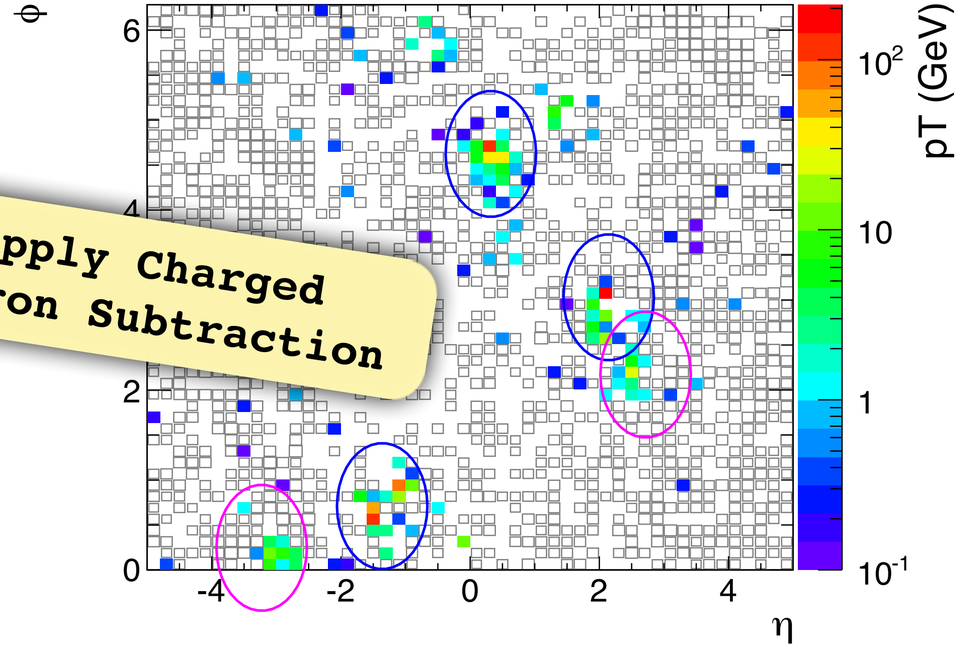




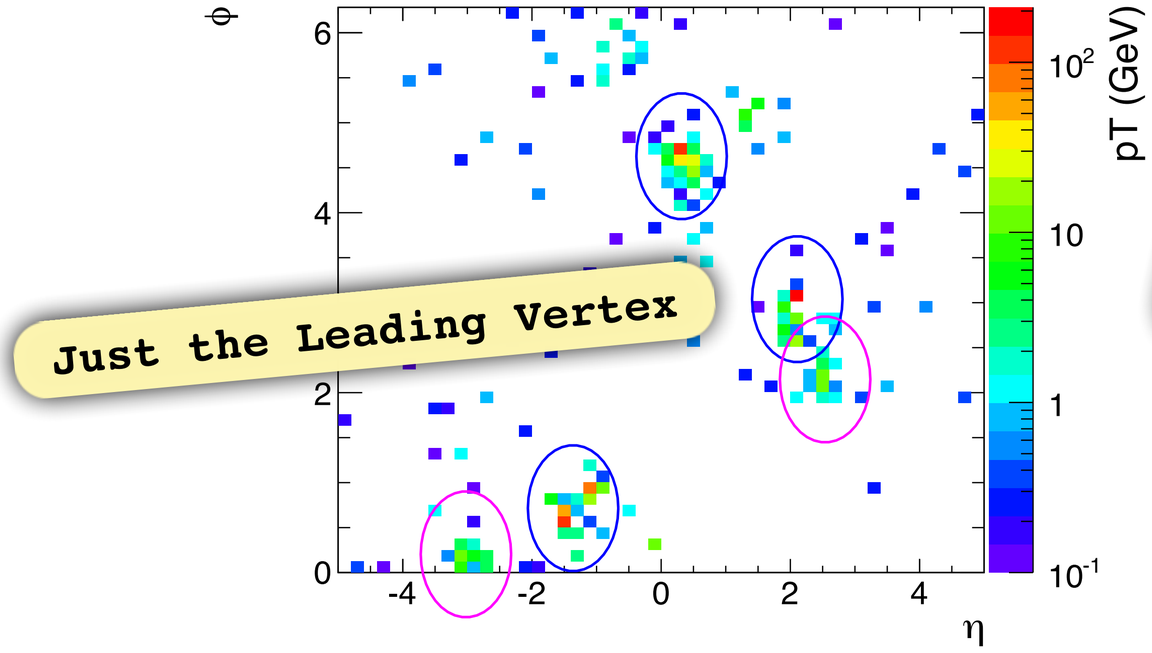
Pile Up Per Particle Identification

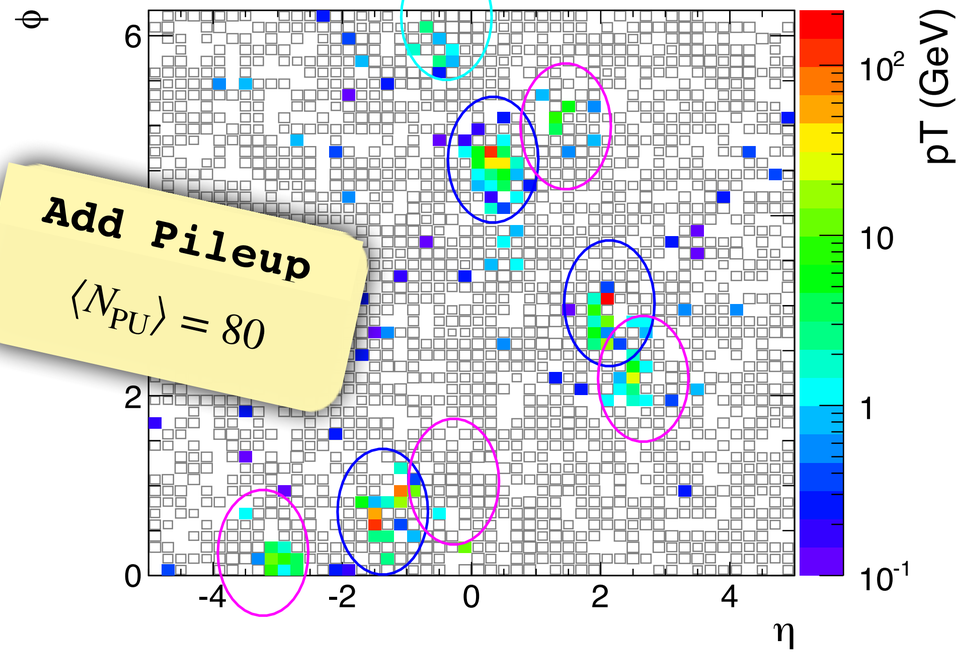


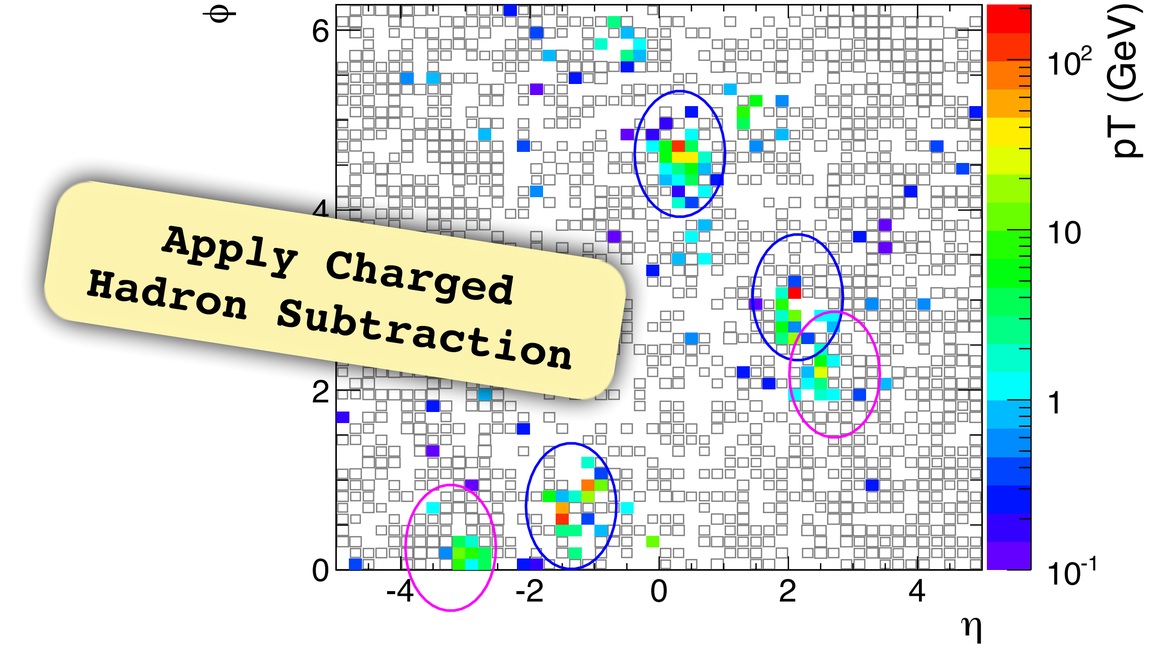


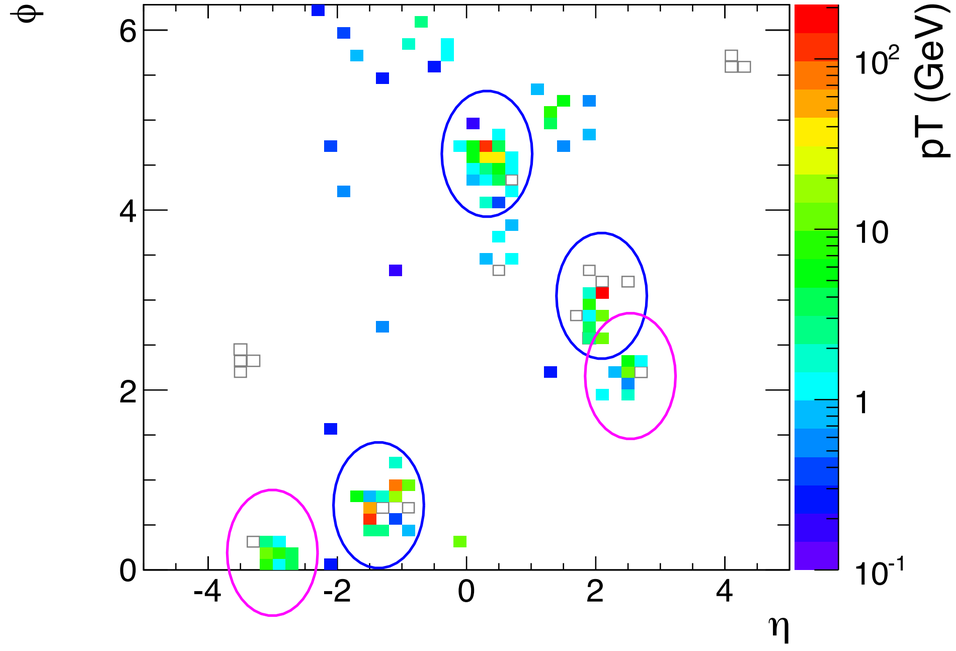


Pile Up Per Particle Identification

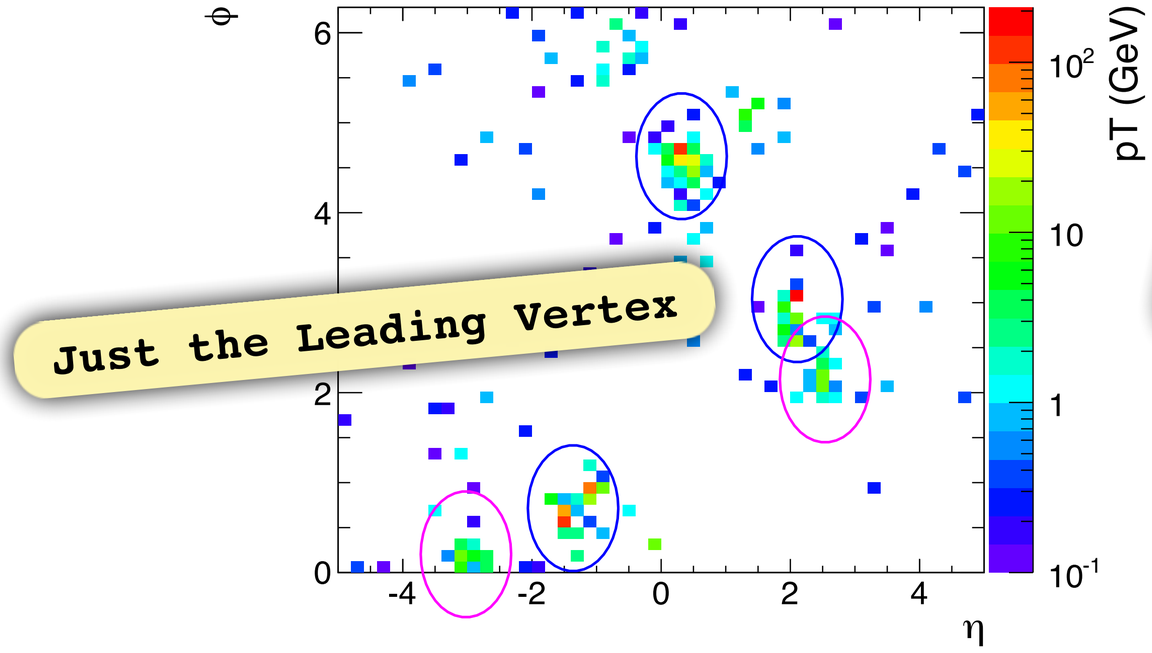


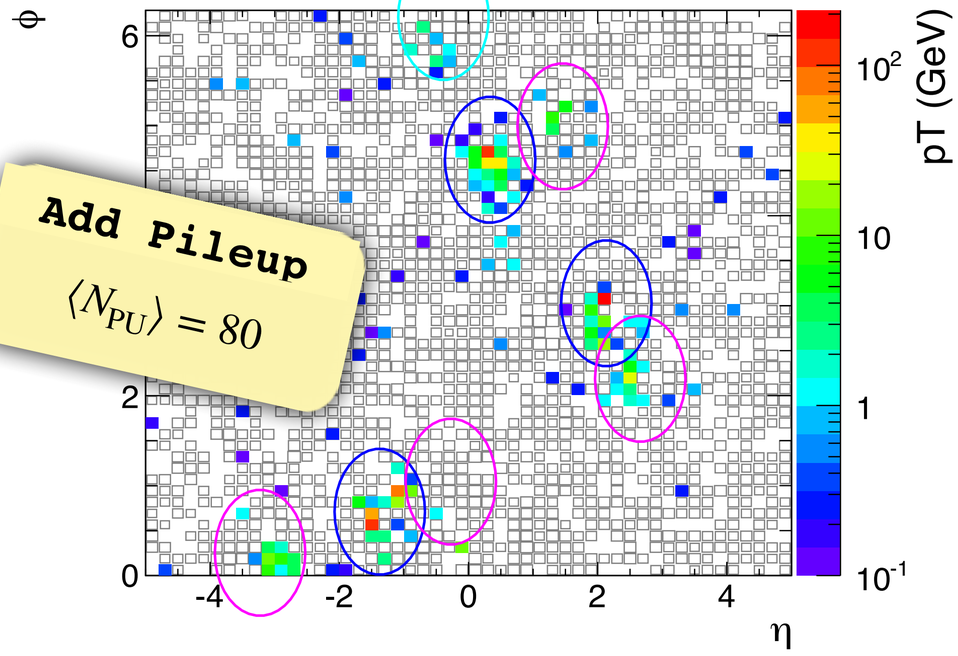


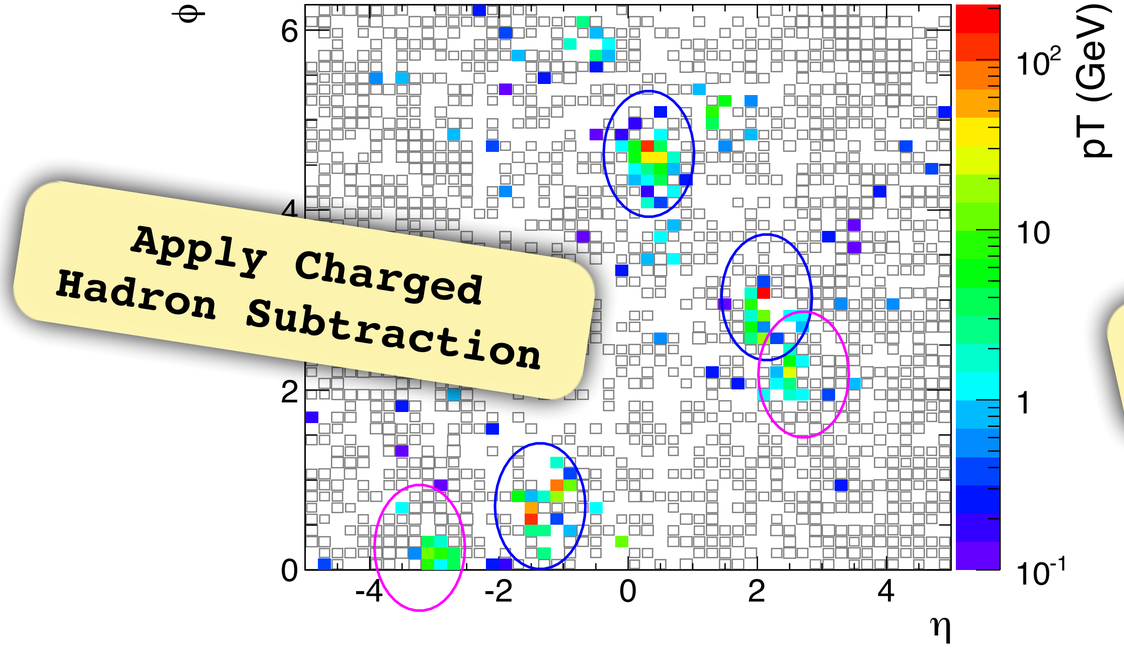


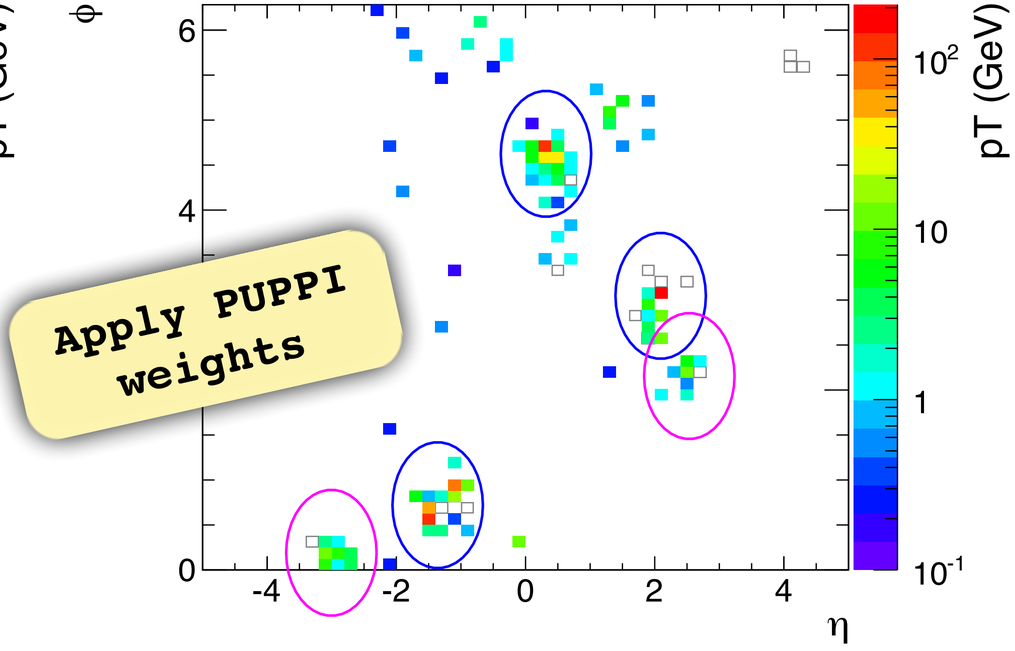


Pile Up Per Particle Identification









OK...I'll stop here