Introduction

- I proposed a change for two values in the electron ID — specifically for the variables max profile discrepancy and max residual E/p (slides from Aug.18 with more detail!)
- In these slides I'm looking into how this proposed change would impact pion reconstruction
  - This is the electron id
- Proposal: change maxProfileDiscrepancy from 0.6 to 0.8 and maxResidualEOverP from 0.2 to 0.4

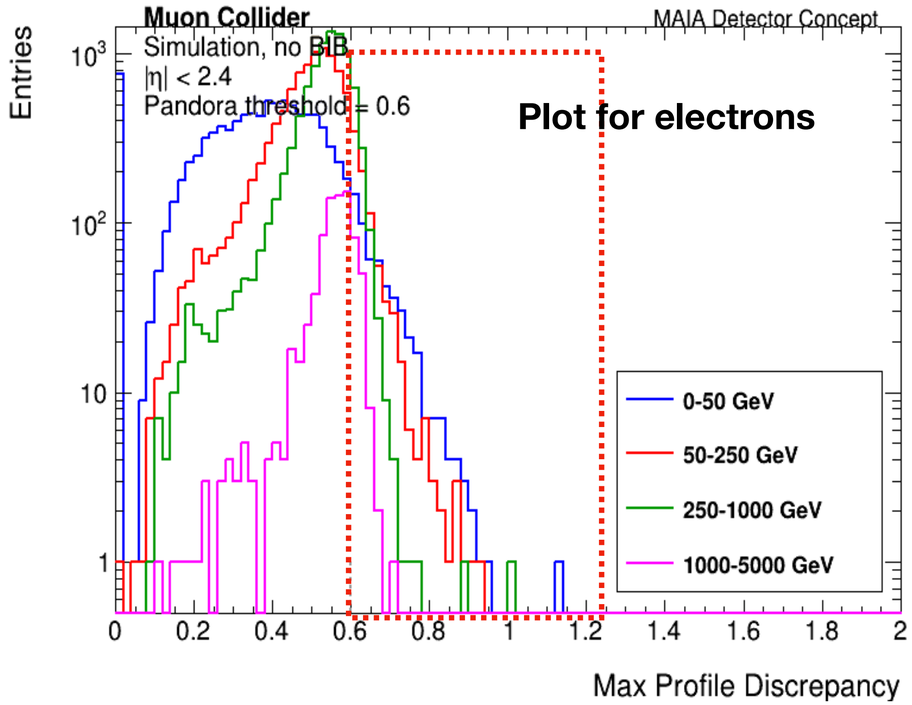
```
LCParticleIdPlugins::LCElectronId::LCElectronId() :
    m_maxInnerLayer(4),
    m_maxEnergy(5.f),
    m_maxProfileStart(4.5f),
    m_maxProfileDiscrepancy(0.6f),
    m_profileDiscrepancyForAutoId(0.5f),
    m_maxResidualEOverP(0.2f)
{
}
```


J. Watts

Profile Discrepancy and Pions

Proposed change to profile discrepancy is from 0.6 to 0.8

- The max profile discrepancy variable describes how much the particles EM shower deviates from a generated/expected EM shower
- The current value is 0.6 in the electron ID. This causes low electron reconstruction efficiency
- Most of the PFOs not identified as electrons are identified as pions!

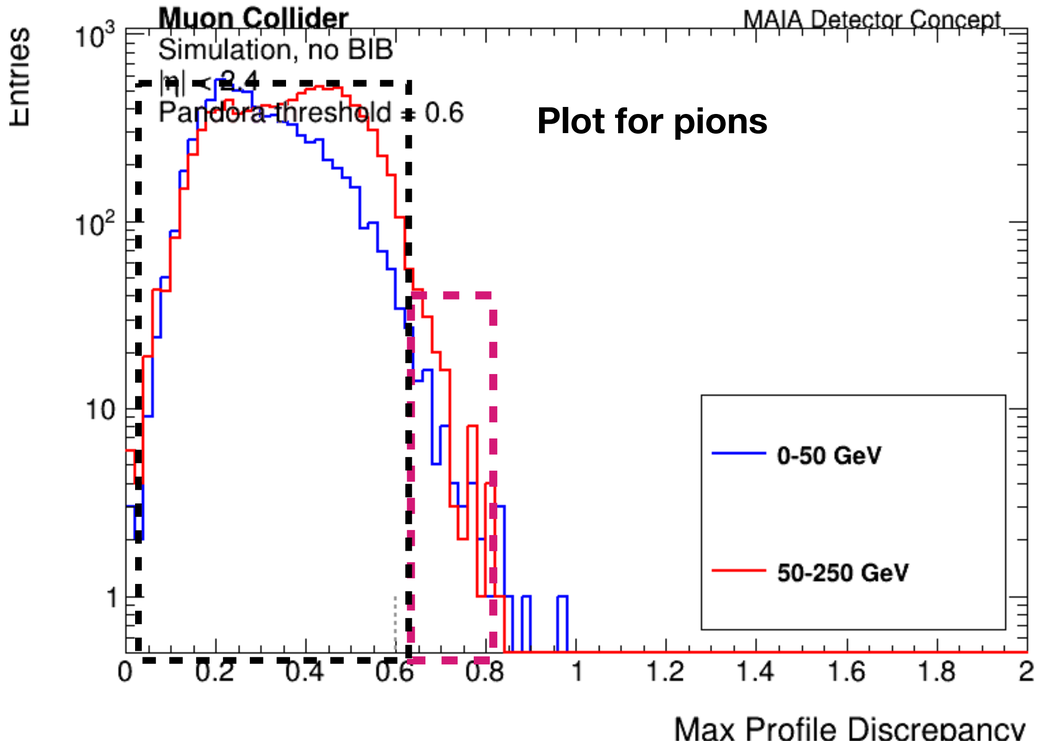


Plot for electrons: Electrons inside of the red box are being thrown away, contributing to low reco efficiency

How pions would be affected by updated max profile discrepancy value

The proposed max profile discrepancy value for electrons is 0.8

- Within the black box are all the values accepted as an electron with the current max profile discrepancy value
  - Even though these are pfos from a pion gun!
- The values in the pink box are pions that would now also get identified (along with values in black box) as electrons with the updated profile discrepancy value
- This will result in a drop in pion reconstruction, but not by a significant amount

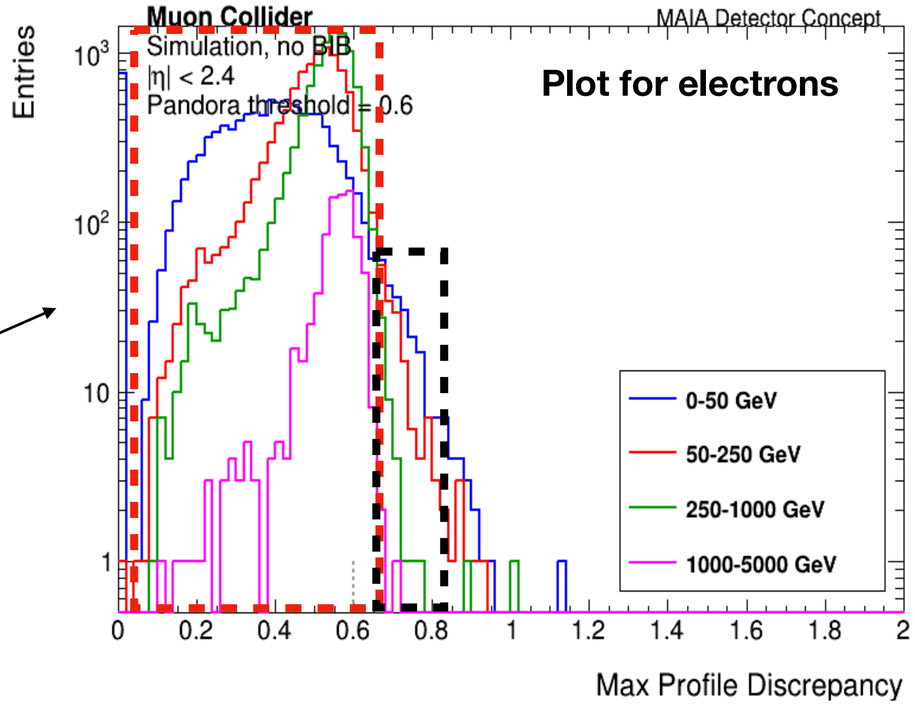


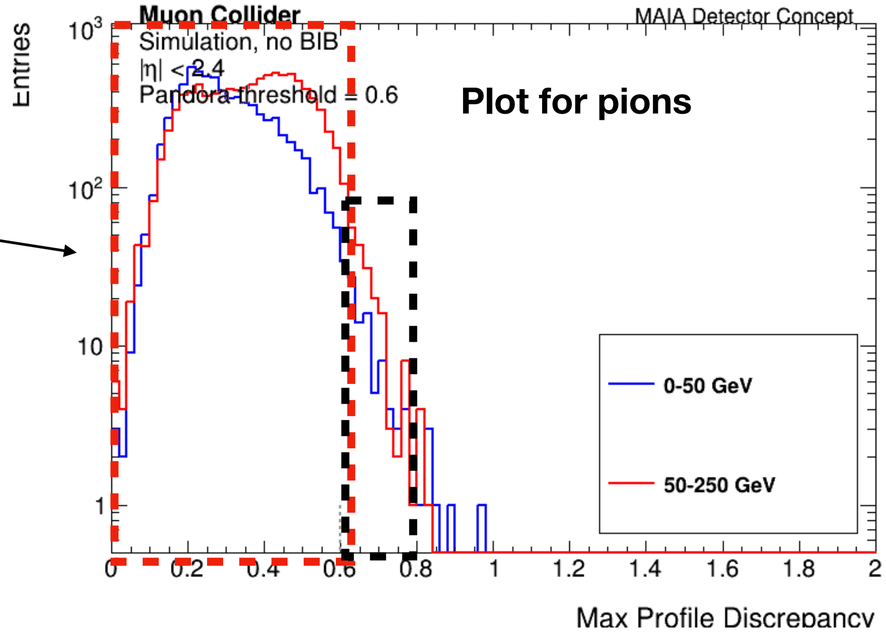
J. Watts

Conclusion

Profile Discrepancy

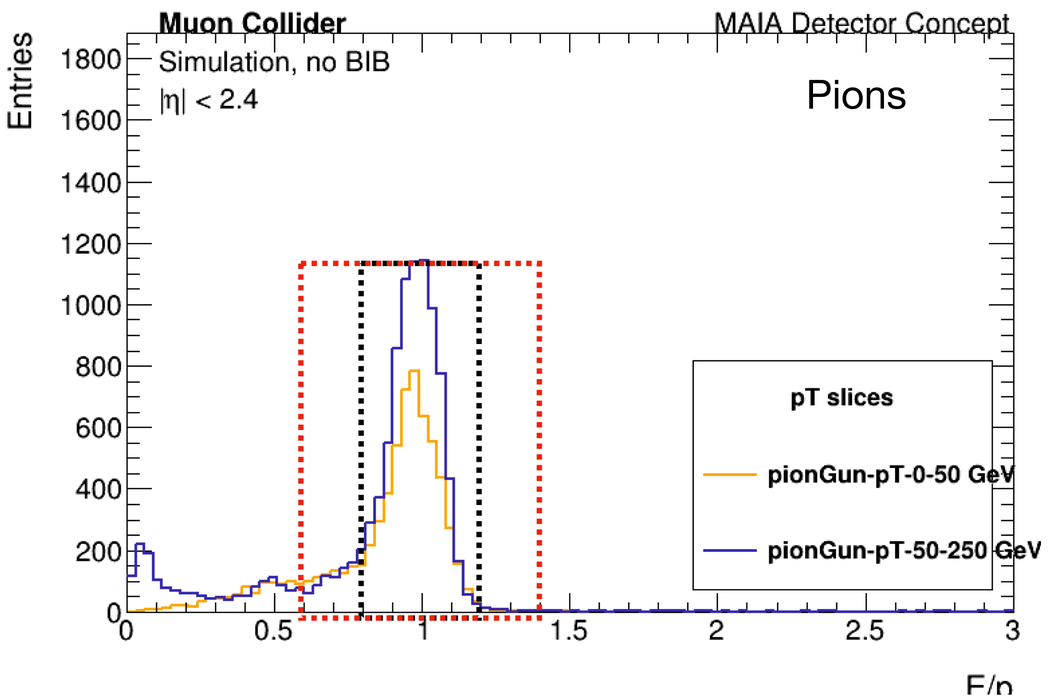
- With the updated maxProfileDiscrepancy variable, the pfos identified as electrons will include values in the red and black box, rather than just the red box
  - Yay for higher electron reco efficiency!
- This will lower the pion reco efficiency but not by a significant amount
- The pions identified as electrons would include variables in the red and black box rather than just the red box (which is with the current value for maxProfileDiscrepancy)
- So we can see that **Updating the max profile discrepancy value in the electron ID to 0.8 would negativity affect our pion reconstruction, but its a negligible amount compared to the increase in electron reco efficiency!**

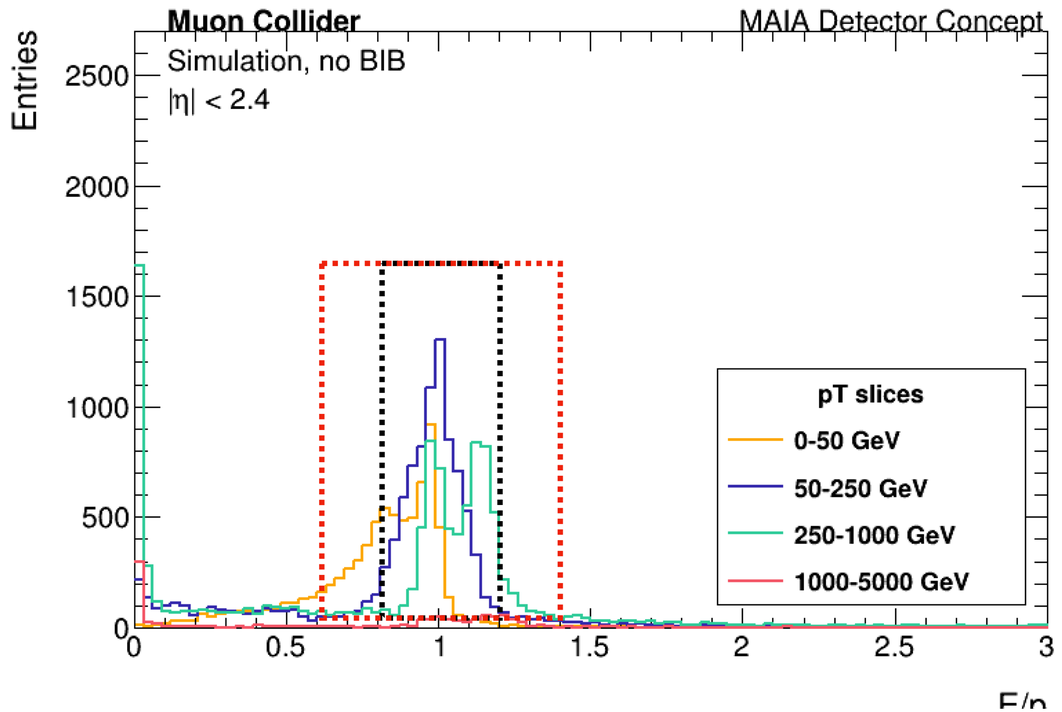




J. Watts

Conclusion





Residual E/p

Red box shows what values will be accepted as electrons with new residual E/p cut (proposed change from 0.2 to 0.4). Black box shows originally accepted values with Pandoras default of 0.2.

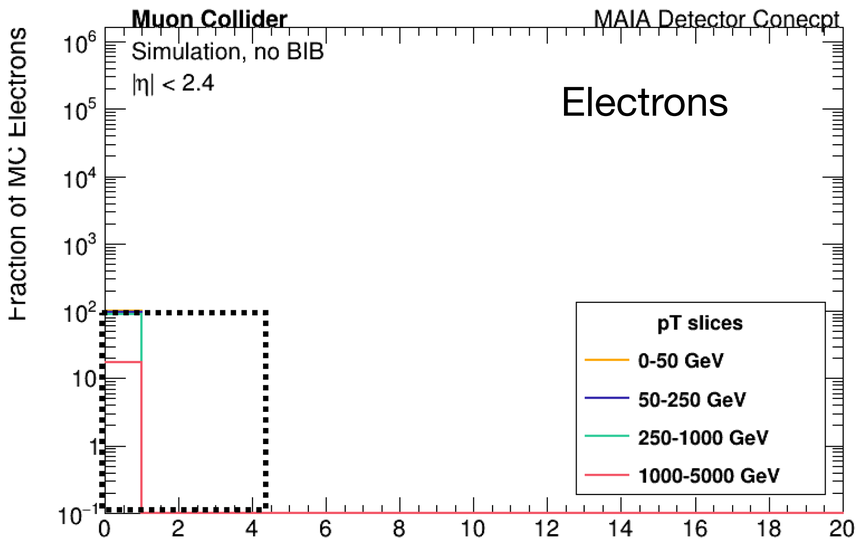
Updating the max residual E/p value to 0.4 from 0.2 would negatively affect our pion reconstruction, but by a negligible amount compared to the increase in electron efficiency

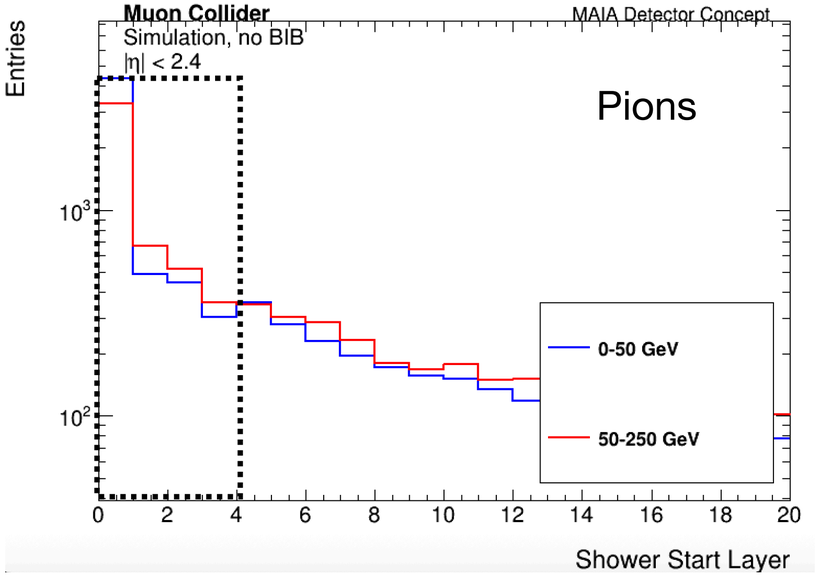
Pion vs electrons in electron ID variables

Backup

maxInnerLayer()

*No change proposed





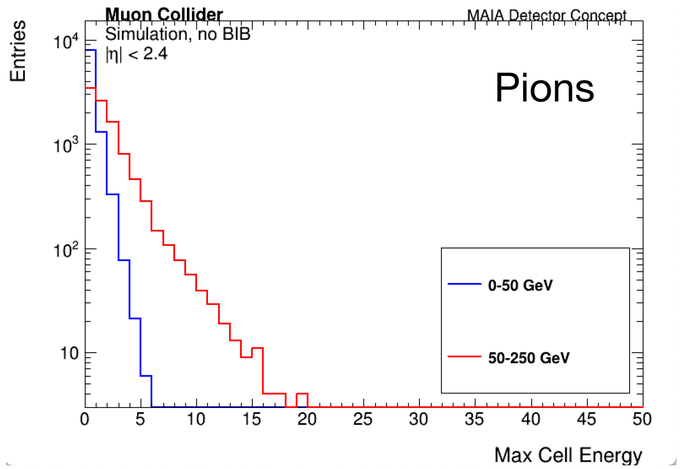
Black box shows values accepted as electrons

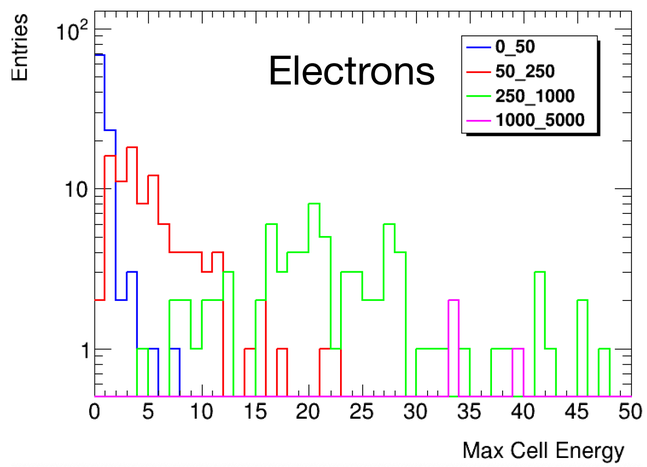
Pion vs electrons in electron ID variables

Backup

maxEnergy()

*No change proposed





These plots show the max energy of the cells from a cluster. The max energy variable is set to 5 GeV, and this is the max energy of the cluster. Multiple cells make up a cluster, and this variable and shower start layer variable are only used if the cluster does not pass the isEMshower() variable. Since one cell is already > the max energy allowed, and our samples go to 5000 GeV, future work could determine if this variable is needed for the MAIA geometry

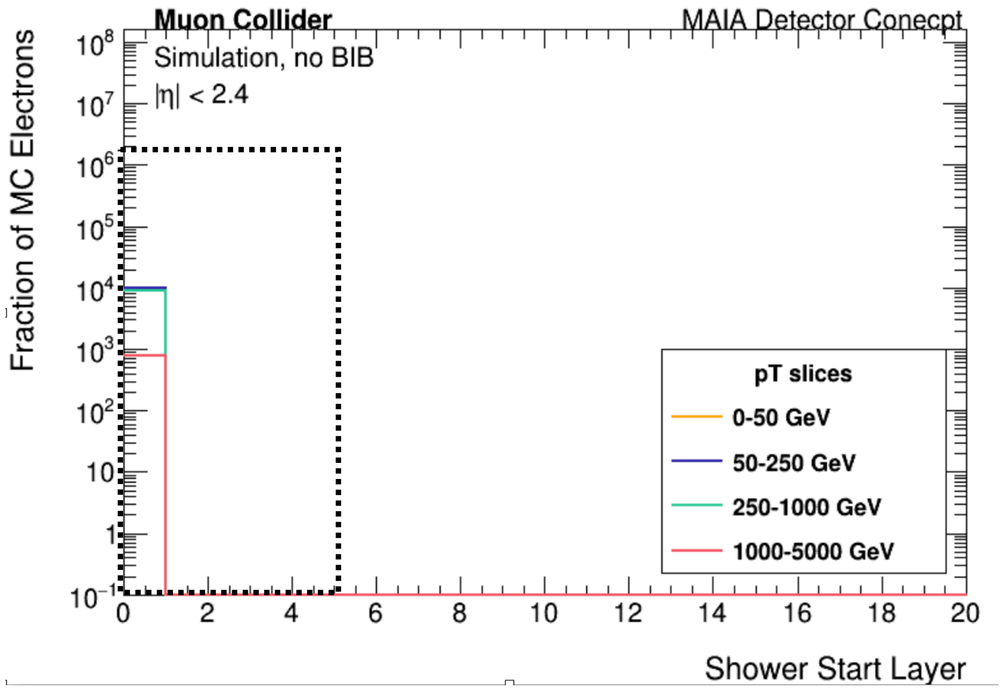
J. Watts

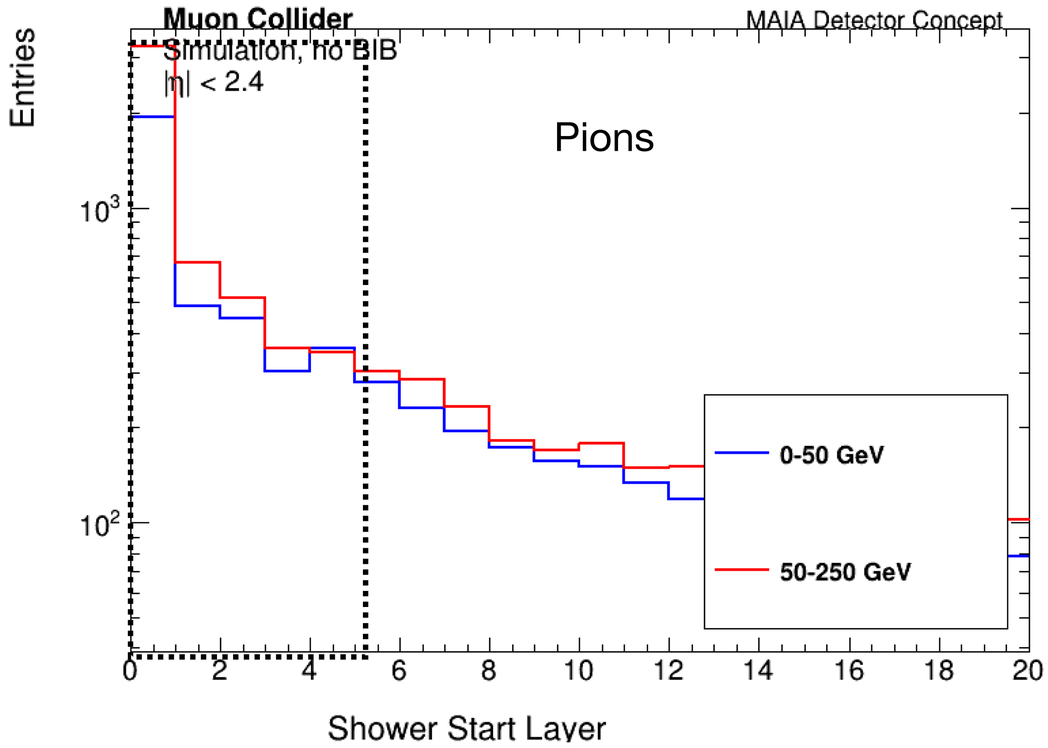
Pion vs electrons in electron ID variables

Backup

maxProfileStart()

*No change proposed





Black box shows values accepted as electrons

J. Watts

PFO id

Backup

```
Electron Reconstruction Efficiency Summary:

0-50 GeV:
  Total electrons: 8481
  Reconstructed as electrons: 4057 (0.478)
  Missed electrons:
    → Reconstructed as photons: 849 (0.100 of total)
    → Reconstructed as charged pions: 3484 (0.411 of total)
    → Reconstructed as neutral pions: 0 (0.000 of total)
    → Reconstructed as neutrons: 7 (0.001 of total)
    → Reconstructed as other: 0 (0.000 of total)
    → Not reconstructed: 84 (0.010 of total)

50-250 GeV:
  Total electrons: 9831
  Reconstructed as electrons: 4405 (0.448)
  Missed electrons:
    → Reconstructed as photons: 896 (0.091 of total)
    → Reconstructed as charged pions: 4453 (0.453 of total)
    → Reconstructed as neutral pions: 0 (0.000 of total)
    → Reconstructed as neutrons: 23 (0.002 of total)
    → Reconstructed as other: 0 (0.000 of total)
    → Not reconstructed: 54 (0.005 of total)

250-1000 GeV:
  Total electrons: 8875
  Reconstructed as electrons: 3467 (0.391)
  Missed electrons:
    → Reconstructed as photons: 1494 (0.168 of total)
    → Reconstructed as charged pions: 3876 (0.437 of total)
    → Reconstructed as neutral pions: 0 (0.000 of total)
    → Reconstructed as neutrons: 4 (0.000 of total)
    → Reconstructed as other: 0 (0.000 of total)
    → Not reconstructed: 34 (0.004 of total)

1000-5000 GeV:
  Total electrons: 799
  Reconstructed as electrons: 239 (0.299)
  Missed electrons:
    → Reconstructed as photons: 280 (0.350 of total)
    → Reconstructed as charged pions: 279 (0.349 of total)
    → Reconstructed as neutral pions: 0 (0.000 of total)
    → Reconstructed as neutrons: 1 (0.001 of total)
    → Reconstructed as other: 0 (0.000 of total)
    → Not reconstructed: 0 (0.000 of total)
```

summary statistics from electron reco

```
=== SUMMARY (combined) ===
Total true charged pions: 18161
Reconstructed as charged pions: 16111  (eff = 0.887)
Missed as electrons:      1205
Missed as photons:        159
Missed as neutral pions:  0
Missed as neutrons:       593
Missed as other:          29
Not reconstructed:        64
```

```
=== SUMMARY (per slice) ===
0-50 GeV:
  True charged pions:        8441
  Reco as charged pions:     7345 (eff = 0.870)
  Missed as electrons:       786
  Missed as photons:         38
  Missed as neutral pions:   0
  Missed as neutrons:        196
  Missed as other:           22
  Not reconstructed:         54
50-250 GeV:
  True charged pions:        9720
  Reco as charged pions:     8766 (eff = 0.902)
  Missed as electrons:       419
  Missed as photons:         121
  Missed as neutral pions:   0
  Missed as neutrons:        397
  Missed as other:           7
  Not reconstructed:         10
```

summary statistics from pion reco

Confusion Matrices

0-50 GeV	Reco electron	Reco pion
Total electrons	~48%	~41%
Total pions	~9%	~87%

50-250 GeV	Reco electron	Reco pion
Total electrons	~45%	~45%
Total pions	~4%	~90%

```
bool LCParticleIdPlugins::LCElectronId::IsMatch(const Cluster *const pCluster) const
{
    const TrackList &associatedTrackList(pCluster->GetAssociatedTrackList());

    if (associatedTrackList.empty())
        return false;

    const float electromagneticEnergy(pCluster->GetElectromagneticEnergy());

    if (!this->GetPandora().GetPlugins()->GetParticleId()->IsEmShower(pCluster) &&
        ((pCluster->GetInnerPseudoLayer() > m_maxInnerLayer) || (electromagneticEnergy > m_maxEnergy)))
    {
        return false;
    }

    const float showerProfileStart(pCluster->GetShowerProfileStart(this->GetPandora()));
    const float showerProfileDiscrepancy(pCluster->GetShowerProfileDiscrepancy(this->GetPandora()));

    if ((showerProfileStart > m_maxProfileStart) || (showerProfileDiscrepancy > m_maxProfileDiscrepancy))
        return false;

    if (showerProfileDiscrepancy < m_profileDiscrepancyForAutoId)
        return true;

    for (TrackList::const_iterator iter = associatedTrackList.begin(), iterEnd = associatedTrackList.end(); iter != iterEnd; ++iter)
    {
        const float momentumAtDca((*iter)->GetMomentumAtDca().GetMagnitude());

        if (momentumAtDca < std::numeric_limits<float>::epsilon())
            throw StatusCodeException(STATUS_CODE_FAILURE);

        const float eOverP(electromagneticEnergy / momentumAtDca);

        if (std::fabs(eOverP - 1.f) < m_maxResidualEOverP)
            return true;
    }

    return false;
```