Comparison of track reconstruction performance in the barrel Straw Tracker in SpdRoot 4.1.1 (2021) and 4.1.5 (2023) tags

2023-06-13

SPD Physics Weekly Meeting

Ekaterina Mosolova

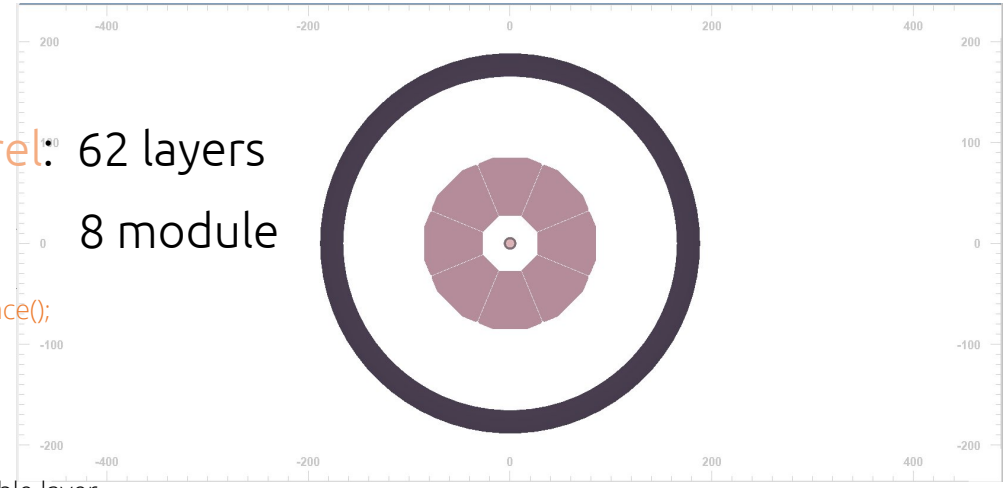
Geometry: updated number of layers in 4.1.5

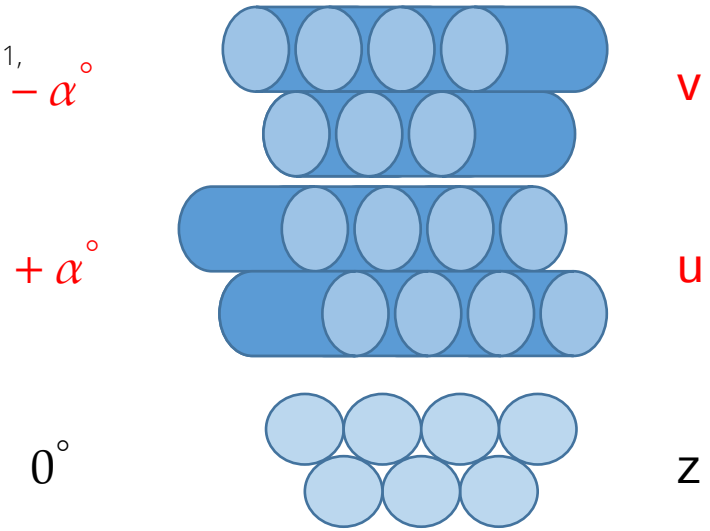
```
void CustomTsB(Double_t deg)
{
  if (deg < 0 || deg > 90) return;

  SpdTsTBGeoMapper* mapper = SpdTsTBGeoMapper::Instance();

  Double_t d = 1.0;    // straw diameter;
  Double_t gap1 = 0.;  // gap between straws in the sublayer
  Double_t gap2 = 0.;  // gap between the sublayers in the double layer
  Double_t gap3 = 0.;  // gap between the double layers in the module

  Double_t w = SpdTsTBGeoMapper::GetLayerWidthFromStrawDiameter(d, gap1,
gap2);
  mapper->SetStrawModulePars(1, w, gap3);

  mapper->SetStrawLayerPars(1,   0., d, gap1, gap2);   // z layer
  mapper->SetStrawLayerPars(1, +deg, d, gap1, gap2);   // u layer
  mapper->SetStrawLayerPars(1, -deg, d, gap1, gap2);   // v layer
```

see “Update of the straw tracker geometry description in SpdRoot” by Ruslan Akhunzyanov

Barrel: 62 layers

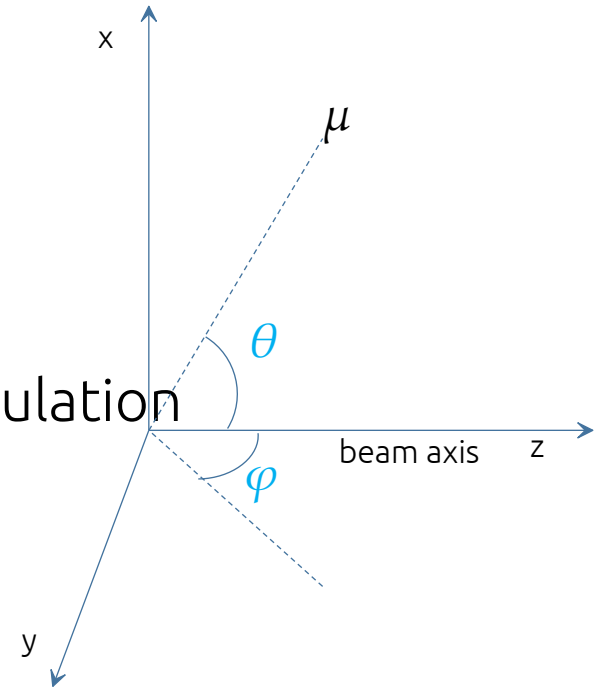
8 module





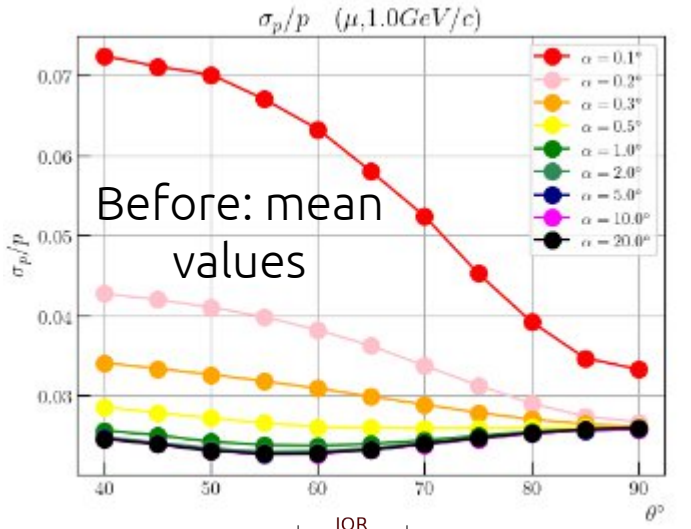
Was generated 10^4 events (μ)

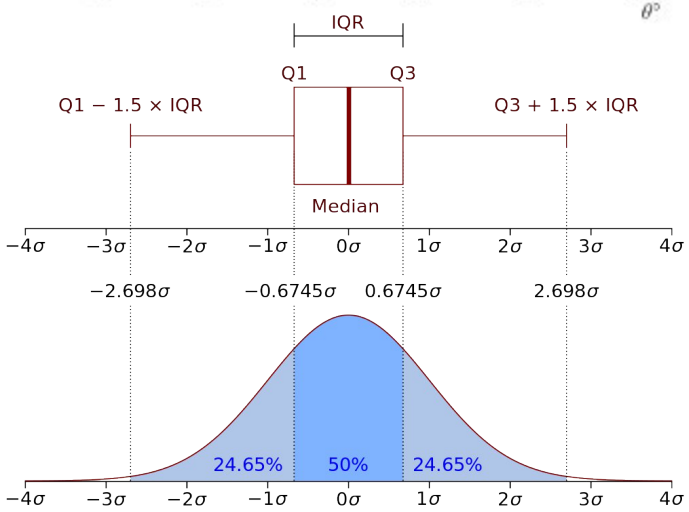
- Version spdroot is tag 4.1.5 (from docker)
- Detectors: ts barrel + ts endcaps
- run->AddModule(Pipe);
- run->AddModule(Magnet);
- Barrel: ~62 layers
- Endcaps: 16 layers
- No vertex detector
- 100 sample and 100 particles in each simulation
- Range of α (CustomTsB):
  - 0.1°, 0.2°, 0.3°, 0.5°, 1°, 2°, 5°, 10°, 20°
- Range of θ (SpdIsotropicGenerator):
  - From 40° to 90° with step 10°
- Range of φ:
  - From 0° to 360°

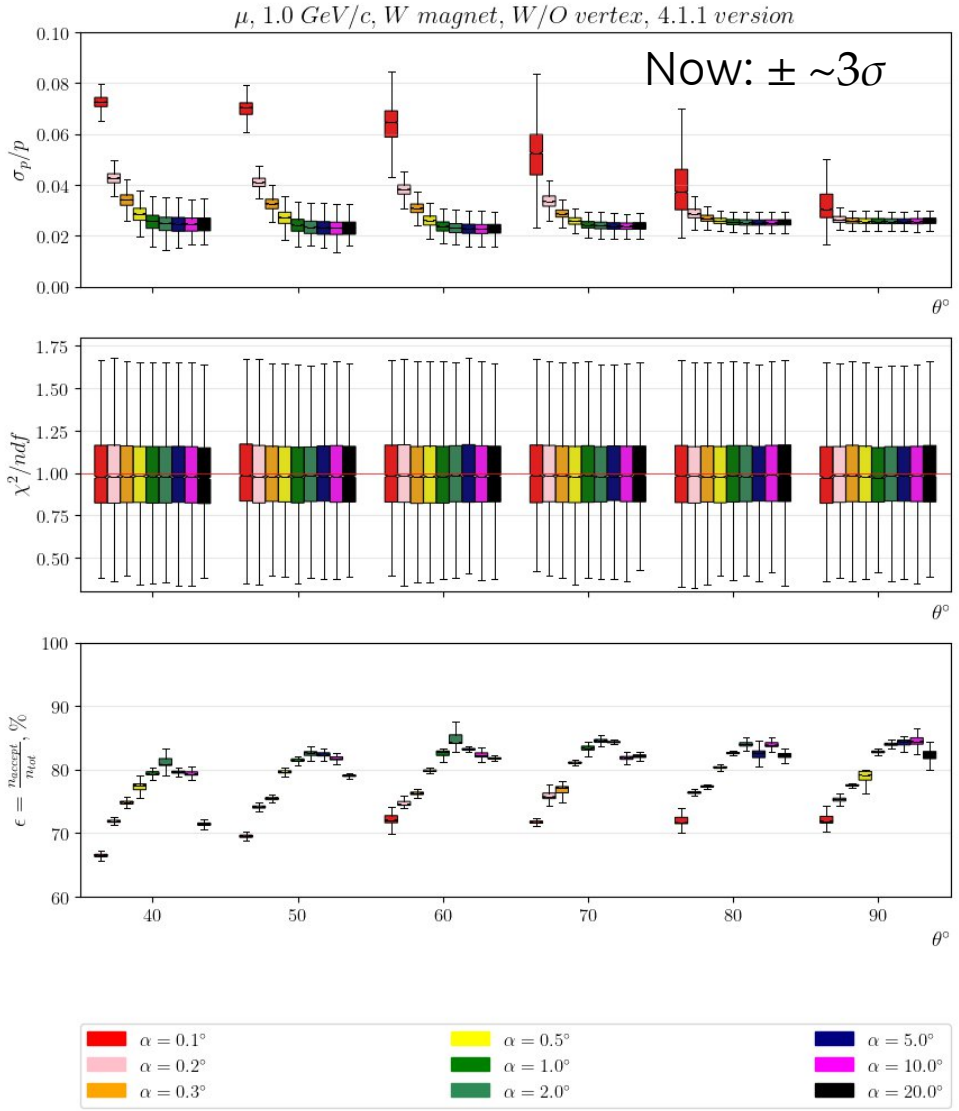


Files:
SimuQslPy8.c
RecoEventFull.c

Whiskers shows ± 1,5 IQR (difference between the 1st and 3rd quartile of the data)

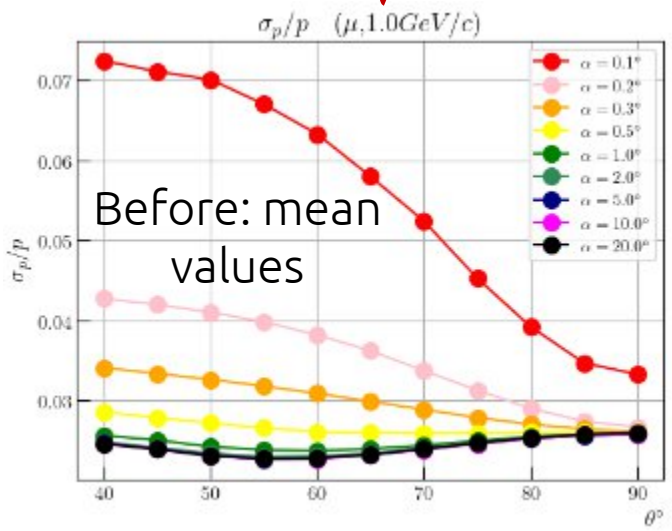


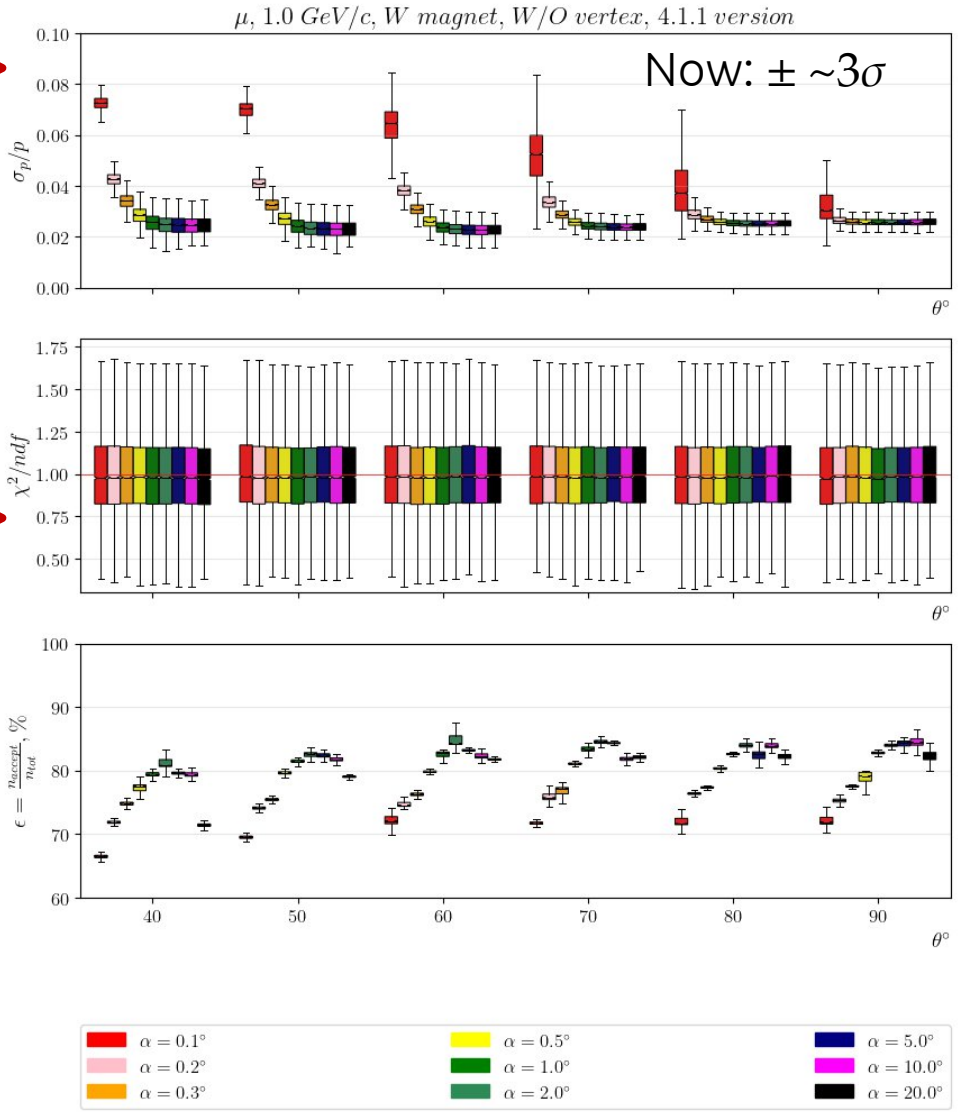




Used boxplot for better visibility

Whiskers shows ± 1,5 IQR (difference between the 1st and 3rd quartile of the data)





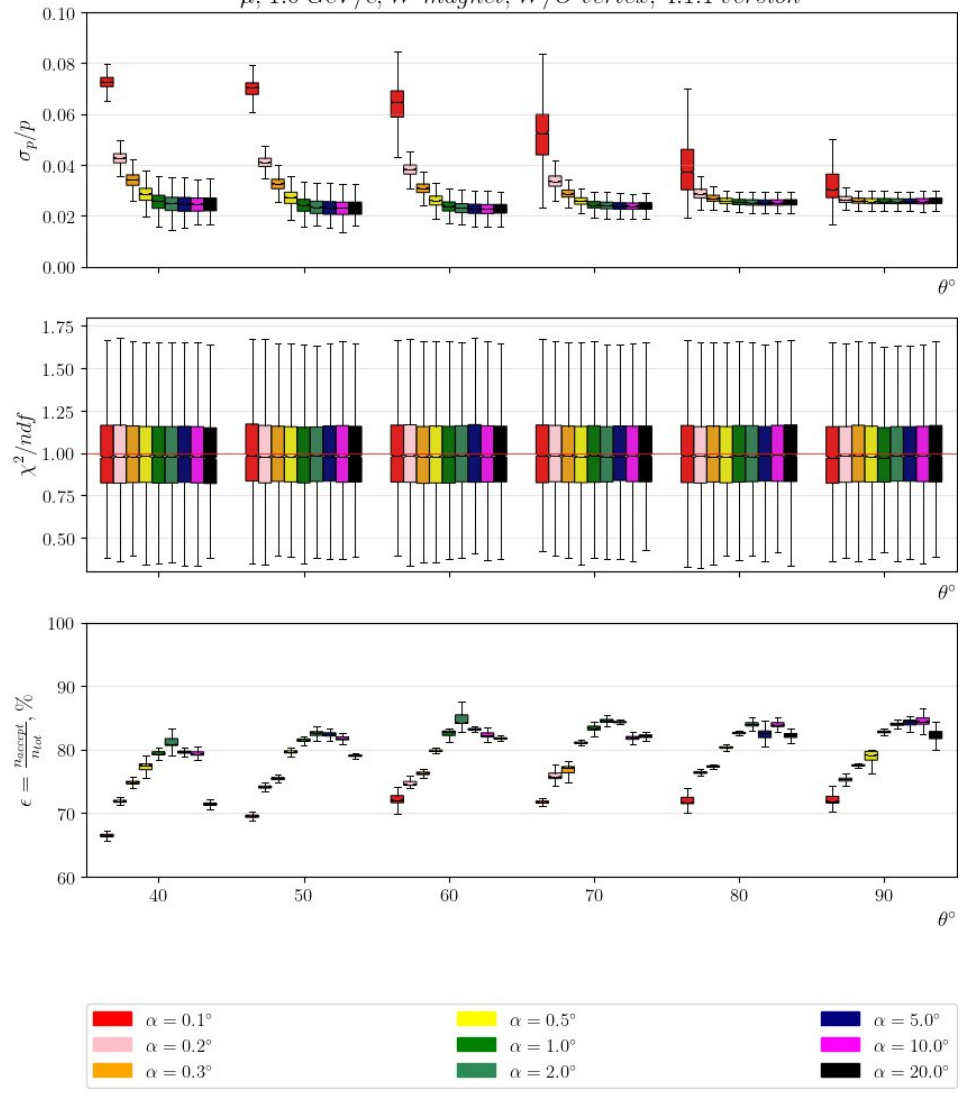
Added χ^2/ndf

Added efficiency: accepted/total

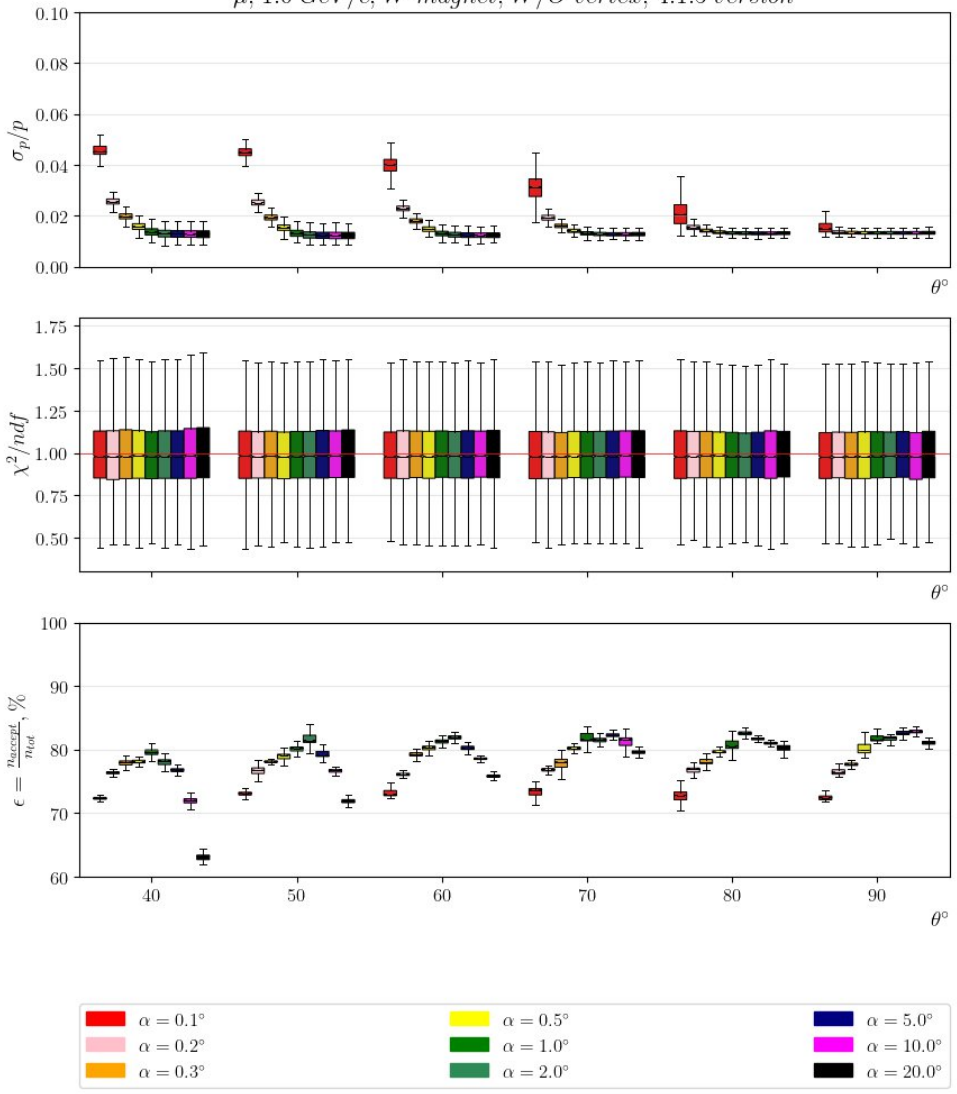
'accepted' i. g. w/o error track and cut = 50 layers

We have smaller momentum variance for each theta value in 4.1.5

4.1.1 (2021)



4.1.5 (2023)



New version (4.1.5) didn't improve efficiency, which is still below 85%

Only particles meeting the criteria shown below were selected (In order of application):

- **GetNHitsTsEC()** = 0
- **GetNHitsTsB()** > cut (Number of layers with hits)
- **GetGeneration()** <= 1

  Only primary particle
- **GetFitPars()** = True

  (spddata/reco/vnt/SpdTrackMC.h)

  Get (or create) track fit parameters
- **GetFirstState()** = True

  (spddata/reco/vnt/SpdTrackFitPar.h

  Get track parameters at first hit position
- **GetIsGood()** = True

  (spddata/reco/vnt/SpdTrackFitPar.h)

  Has'n Error Flag

  Has'n error message

  GenFit fully converged
- **GetIsAcceptable()** = True

  Has'n Error Flag

  Has'n error message

  NDF ≥ 3

  $\chi^2/NDF \leq 2$
- **GetChi2overNDF()** ≤ 2

```
inline Bool_t SpdTrackFitPar::GetIsGood() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fConvergencyGF != 1) return false;
    return true;
}

inline Bool_t SpdTrackFitPar::GetIsAcceptable() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fNDF < 3) return false;
    if (GetChi2overNDF() < 2) return true;
    return false;
}
```

5% events were lost due to getIsGood filter
An additional 15% events were lost due to getAccptable filter

- **GetNHitsTsEC()** = 0
- **GetNHitsTsB()** > cut (Number of layers with hits)
- **GetGeneration()** <= 1

  Only primary particle
- **GetFitPars()** = True

  (spddata/reco/vnt/SpdTrackMC.h)

  Get (or create) track fit parameters
- **GetFirstState()** = True

  (spddata/reco/vnt/SpdTrackFitPar.h

  Get track parameters at first hit position
- **GetIsGood()** = True

  (spddata/reco/vnt/SpdTrackFitPar.h)

  Has'n Error Flag

  Has'n error message

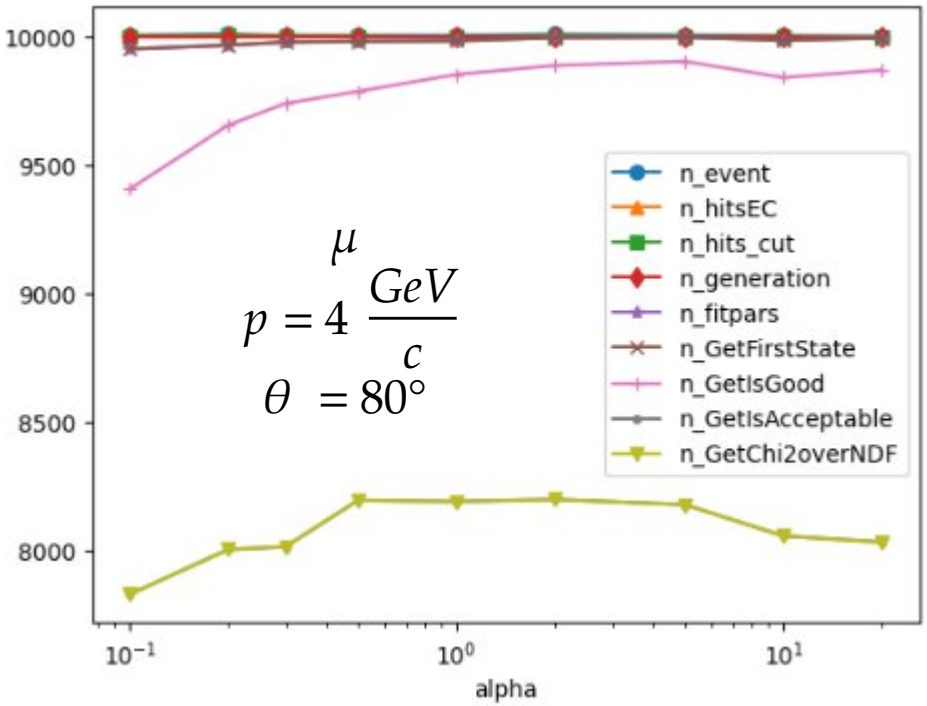
  GenFit fully converged
- **GetIsAcceptable()** = True

  Has'n Error Flag

  Has'n error message

  NDF ≥ 3

  $\chi^2/NDF \leq 2$
- **GetChi2overNDF()** ≤ 2



α — angle between straw tubes

θ — angle between beam axis and direction of particle velocity vector

Application of the getIsGood filter decrease tracks amount by 5%

- **GetNHitsTsEC()** = 0
- **GetNHitsTsB()** > cut (Number of layers with hits)
- **GetGeneration()** <= 1

  Only primary particle
- **GetFitPars()** = True

  (spddata/reco/vnt/SpdTrackMC.h)

  Get (or create) track fit parameters
- **GetFirstState()** = True

  (spddata/reco/vnt/SpdTrackFitPar.h

  Get track parameters at first hit position
- **GetIsGood()** = True

  (spddata/reco/vnt/SpdTrackFitPar.h)

  Has'n Error Flag

  Has'n error message

  GenFit fully converged
- **GetIsAcceptable()** = True

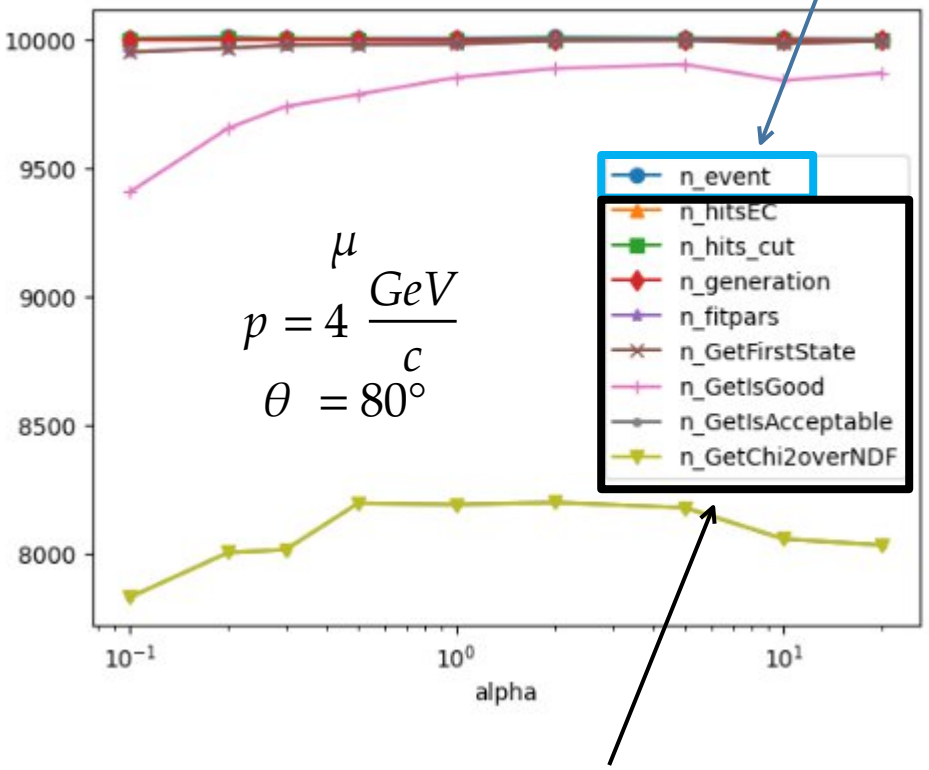
  Has'n Error Flag

  Has'n error message

  NDF ≥ 3

  $\chi^2/NDF \leq 2$
- **GetChi2overNDF()** ≤ 2

All events

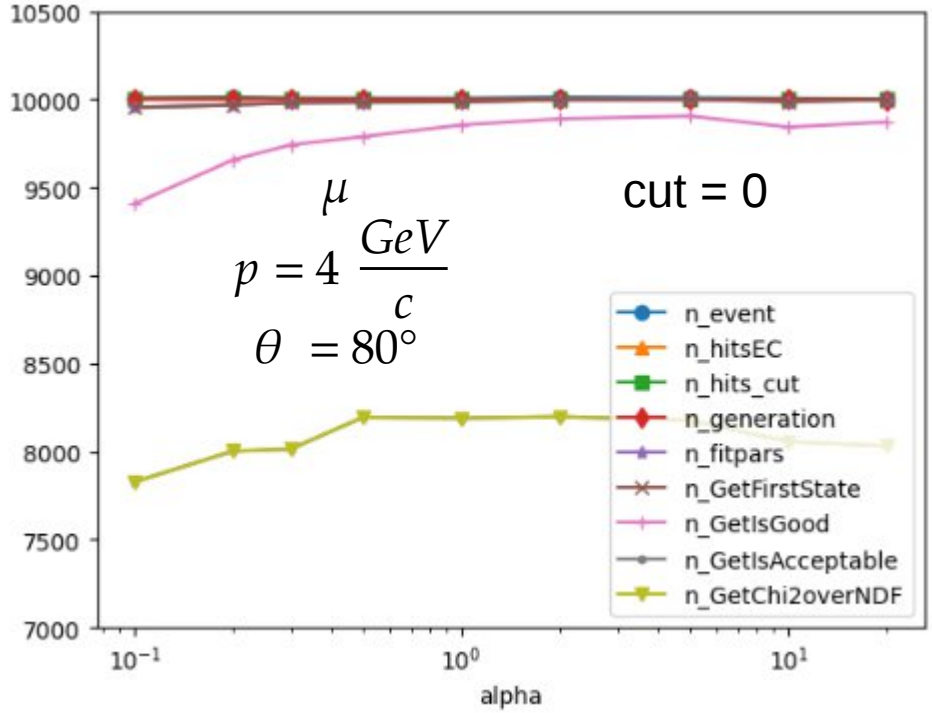


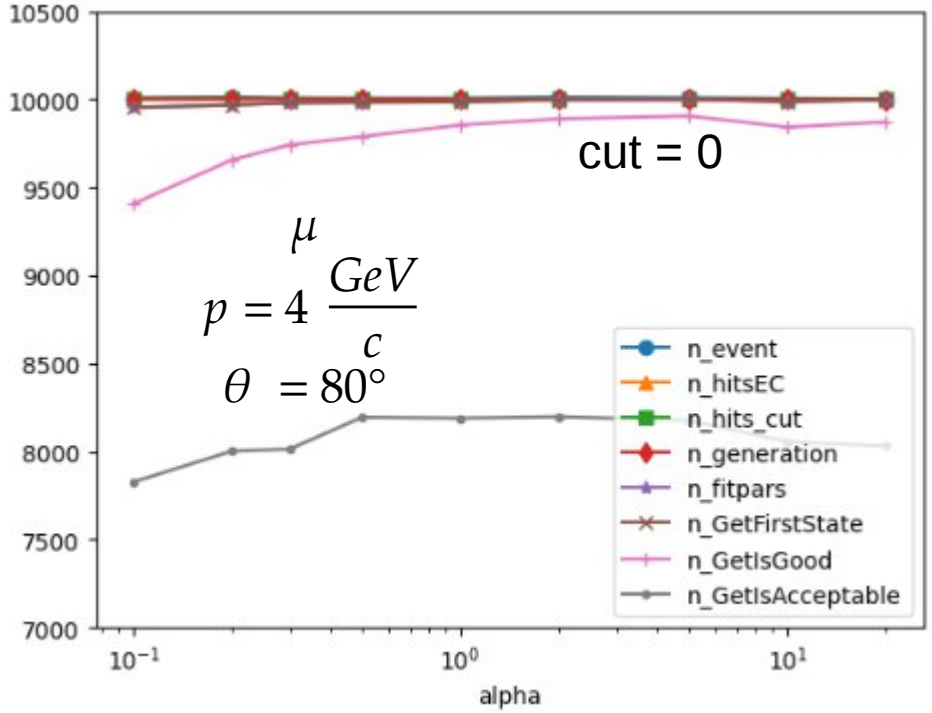
The number of particles **after** the condition

α — angle between straw tubes

θ — angle between beam axis and direction of particle velocity vector

GetIsAcceptable and GetChi2overNDF is similar. Using two is overkill





GetIsAcceptable()

Has'n Error Flag

Has'n error message

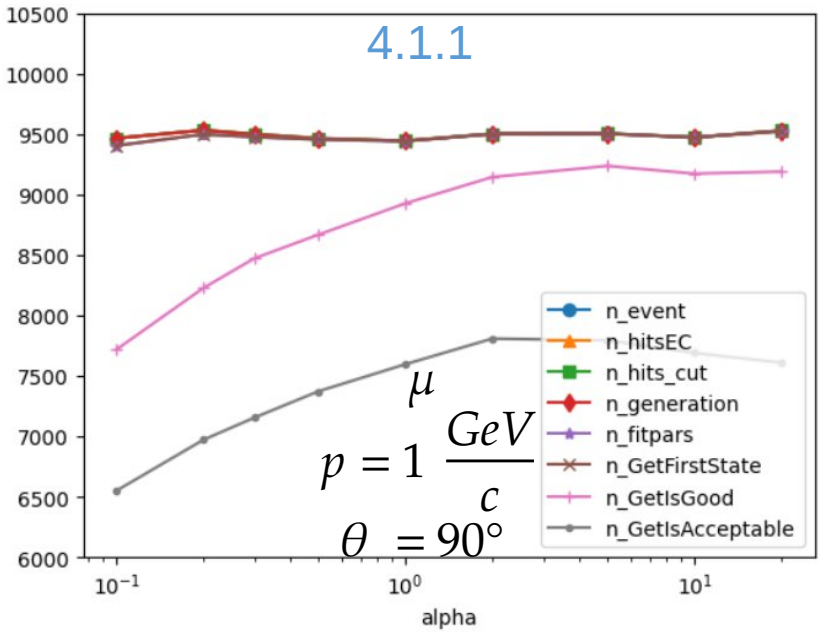
NDF ≥ 3

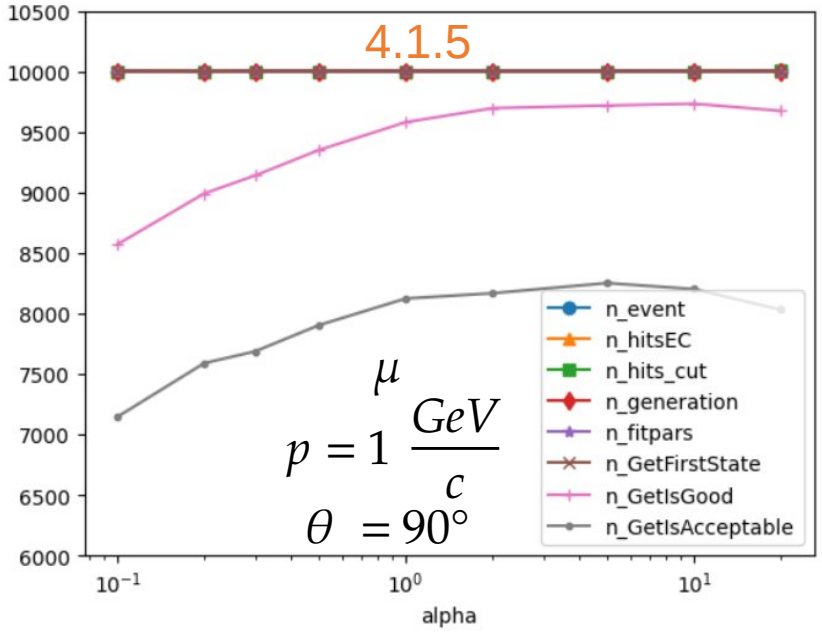
χ^2/NDF ≤ 2

GetChi2overNDF()

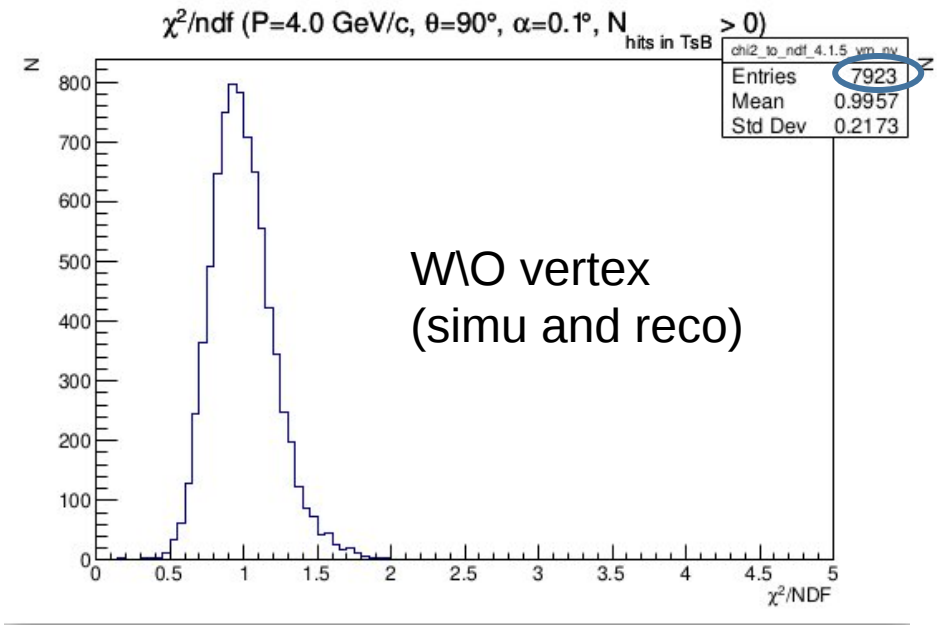
χ^2/NDF ≤ 2

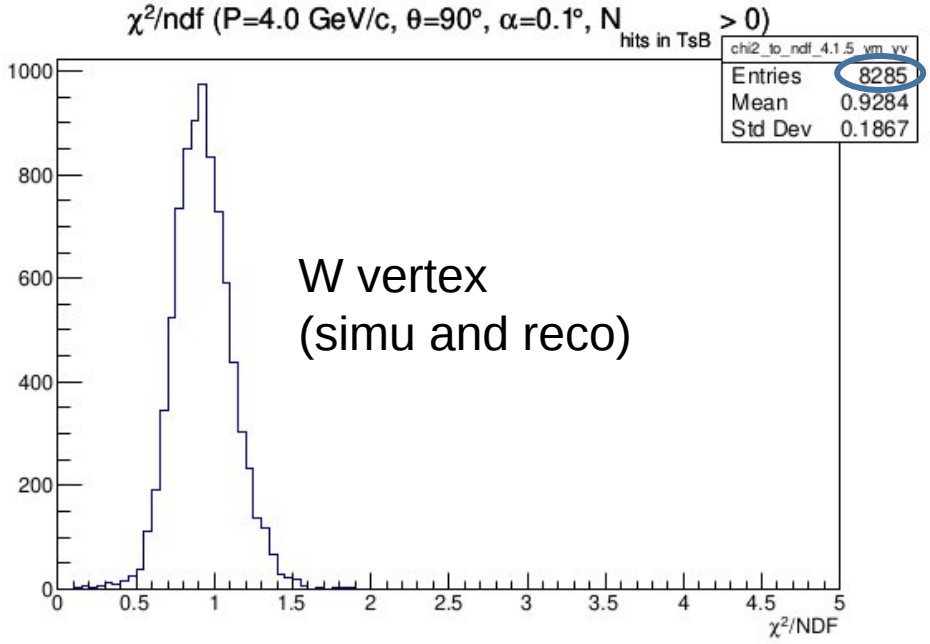
In 4.1.1 (2021) 5% of events are lost during the simulation phase





Statistics with vertex detector are richer by less than 5%





Previously, there were reports that such parameters filter out 20% of the tracks

Cuts on tracks used in the analysis

- Only tracks from primary vertex.
- Fit parameters exist.
- `GetIsAcceptable()`
  (no fit error flags, ndf ≥ 3, $\chi^2/\text{ndf} \leq 2$)
- [No hits in endcaps.]

About 80% of tracks pass these cuts.

Source: Ruslan Akhunzyanov SPD Physics & MC meeting, Nov 10, 2021

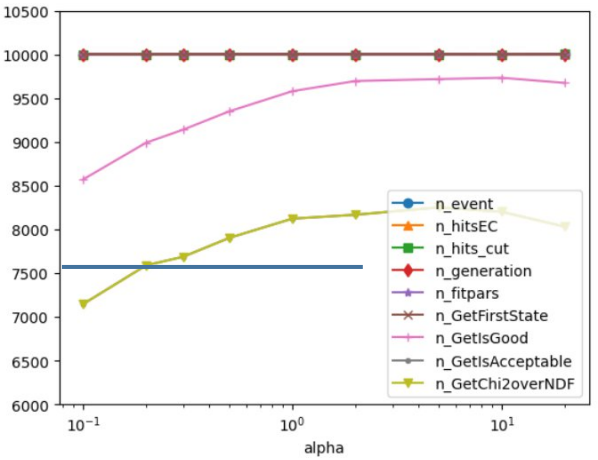
Does the $\chi^2/ndf \leq 2$ condition affect the number of events?

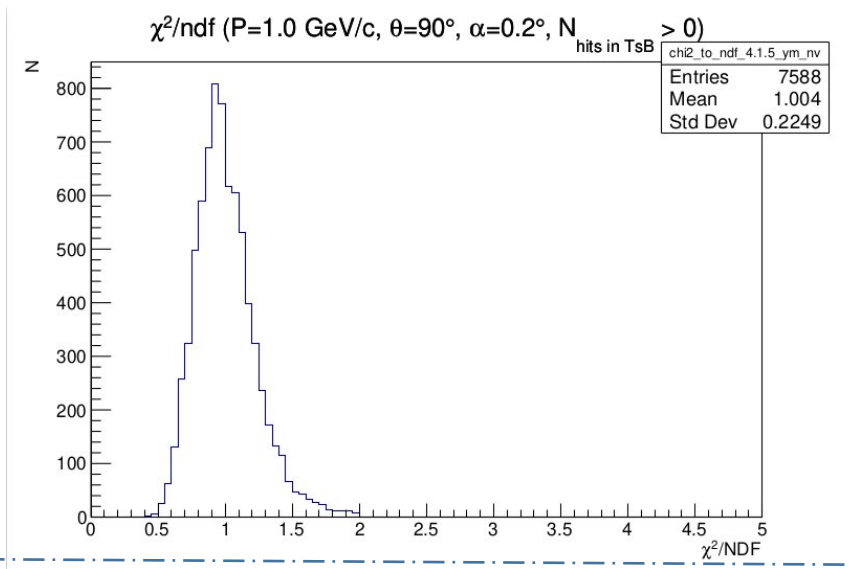
GetIsGood() == True

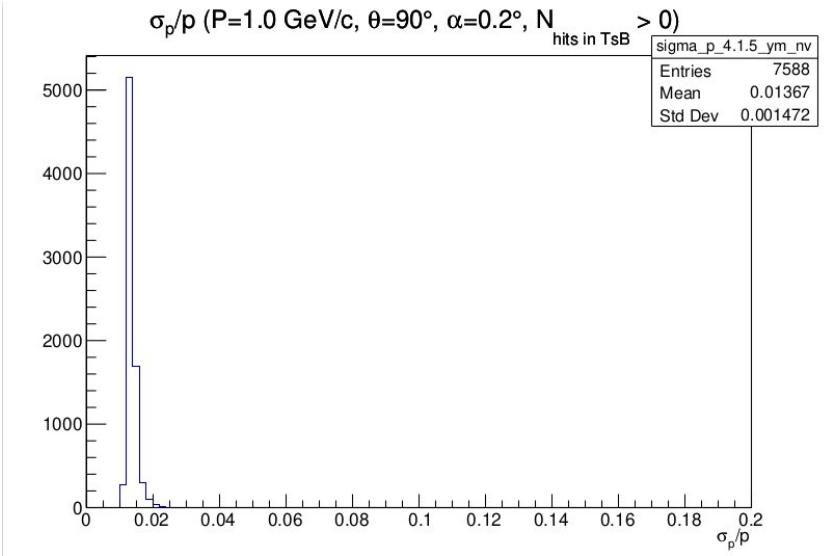
GetIsAcceptable() == True

GetChi2overNDF() <= 2

you are here








```
inline Bool_t SpdTrackFitPar::GetIsGood() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fConvergencyGF != 1) return false;
    return true;
}
```

```
inline Bool_t SpdTrackFitPar::GetIsAcceptable() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fNDF < 3) return false;
    if (GetChi2overNDF() < 2) return true;
    return false;
}
```

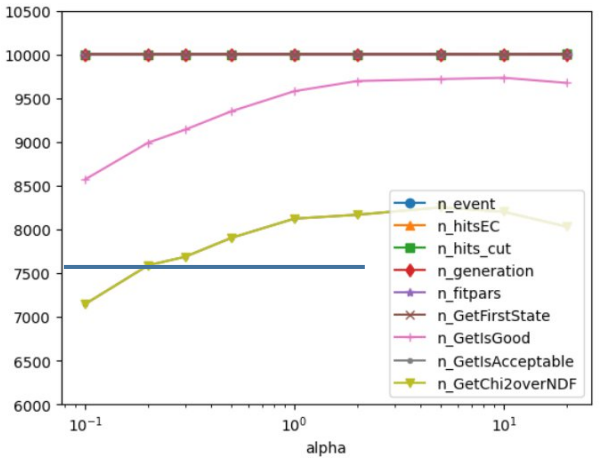
Loss of events due to a condition χ^2/ndf in GetAcceptable

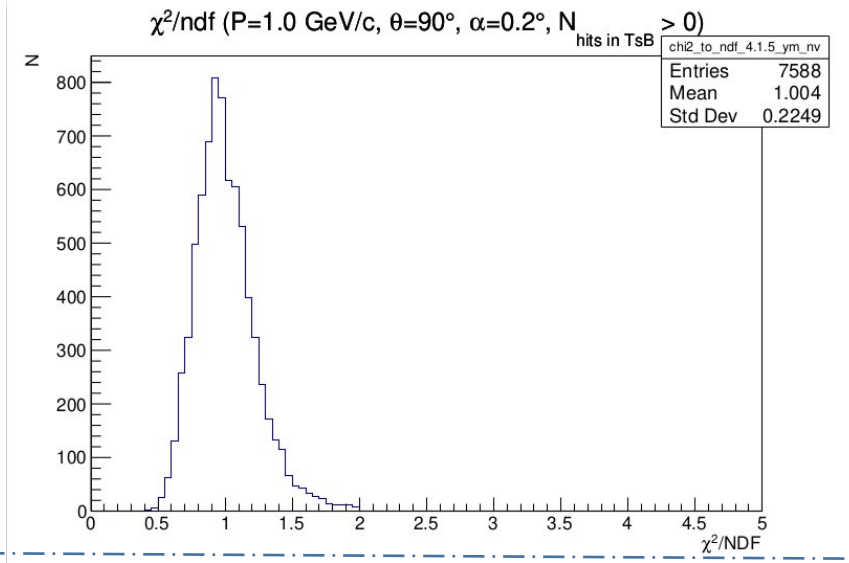
GetIsGood() == True

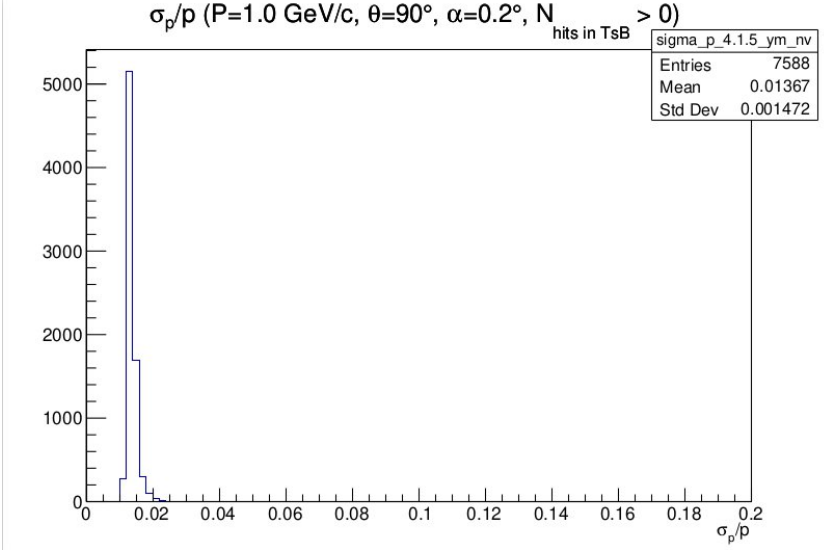
GetIsAcceptable() == True

GetChi2overNDF() <= 2

you are here








```
inline Bool_t SpdTrackFitPar::GetIsGood() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fConvergencyGF != 1) return false;
    return true;
}
```

```
inline Bool_t SpdTrackFitPar::GetIsAcceptable() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fNDF < 3) return false;
    if (GetChi2overNDF() < 2) return true;
    return false;
}
```

Conclusion

- Why do we get such a small efficiency with GetAcceptable condition?
- In 4.1.5 (2023) we stopped losing particles in the simulation
- The number of events with the vertex detector are richer by less than 5%
- The low efficiency issue was known before
- We need to take a closer look at how χ^2/ndf is calculated

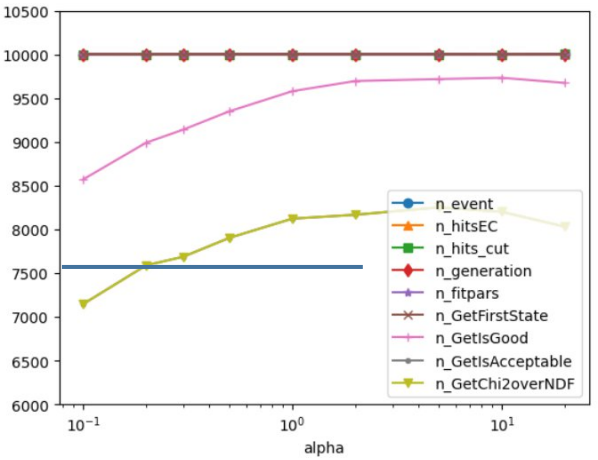
Backup slides

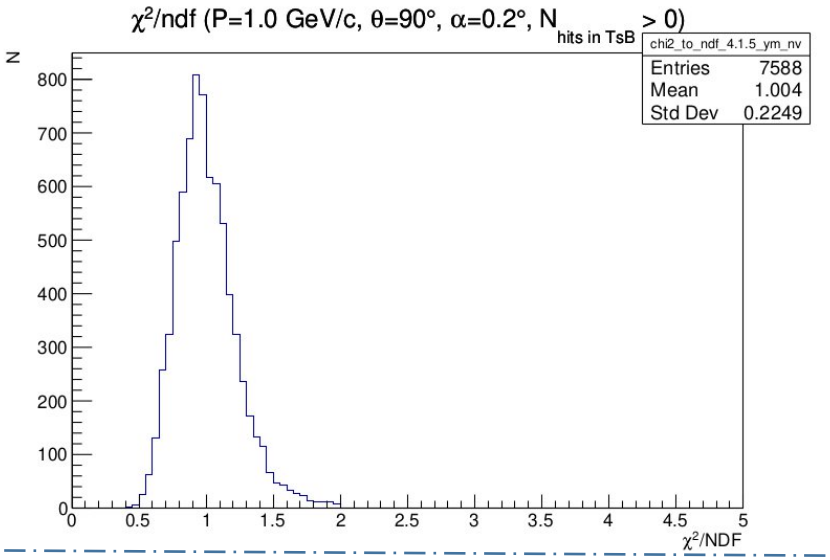
GetIsGood() == True

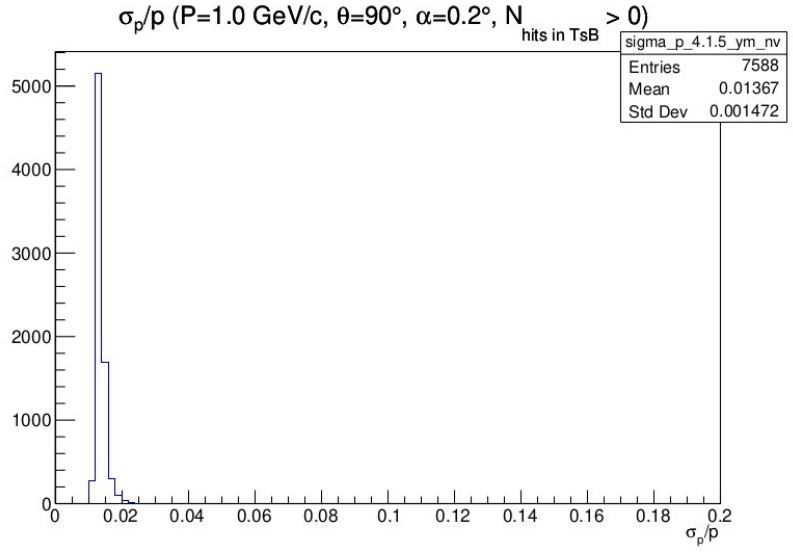
GetIsAcceptable() == True

GetChi2overNDF() <= 2

you are here








```
inline Bool_t SpdTrackFitPar::GetIsGood() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fConvergencyGF != 1) return false;
    return true;
}
```

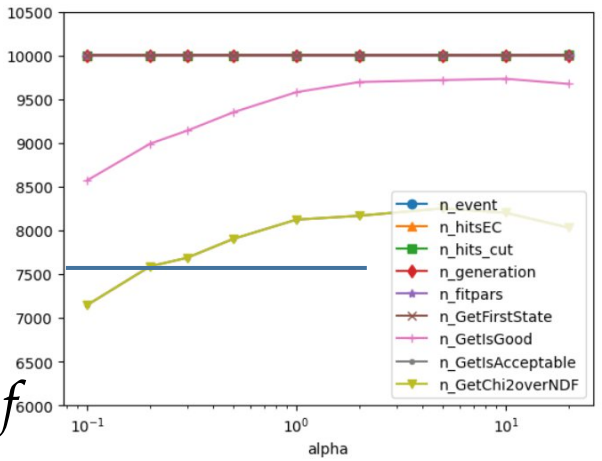
```
inline Bool_t SpdTrackFitPar::GetIsAcceptable() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fNDF < 3) return false;
    if (GetChi2overNDF() < 2) return true;
    return false;
}
```

GetIsGood() == True

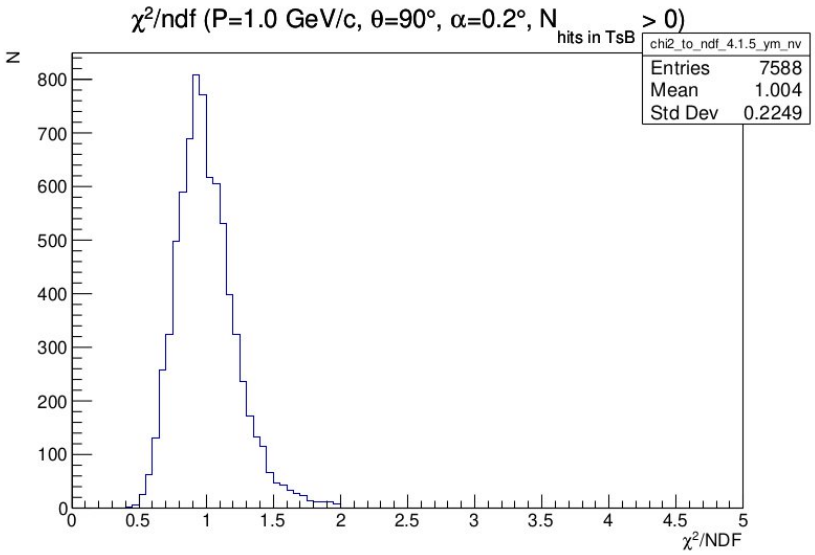
GetIsAcceptable() == True

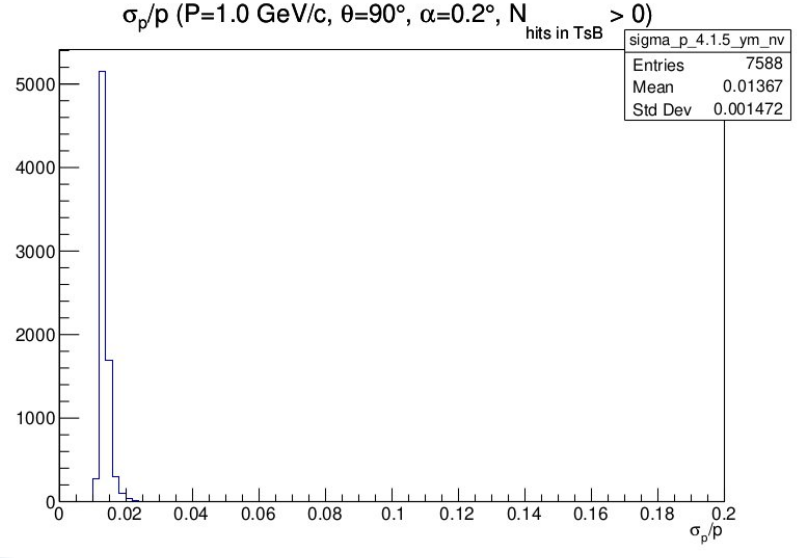
GetChi2overNDF() <= 2

you are here



Loss of events due to a condition χ^2/ndf






```
inline Bool_t SpdTrackFitPar::GetIsGood() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fConvergencyGF != 1) return false;
    return true;
}
```

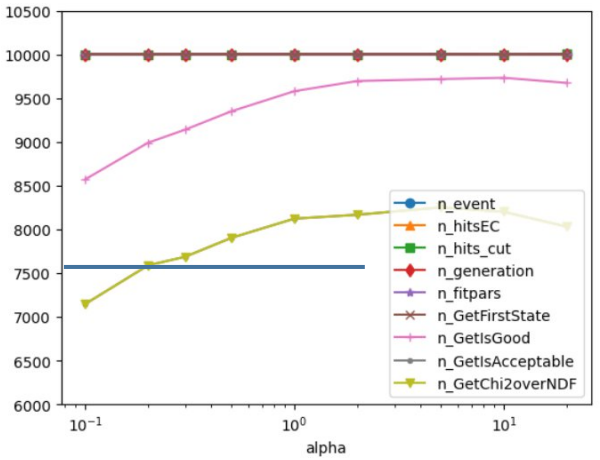
```
inline Bool_t SpdTrackFitPar::GetIsAcceptable() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fNDF < 3) return false;
    if (GetChi2overNDF() < 2) return true;
    return false;
}
```

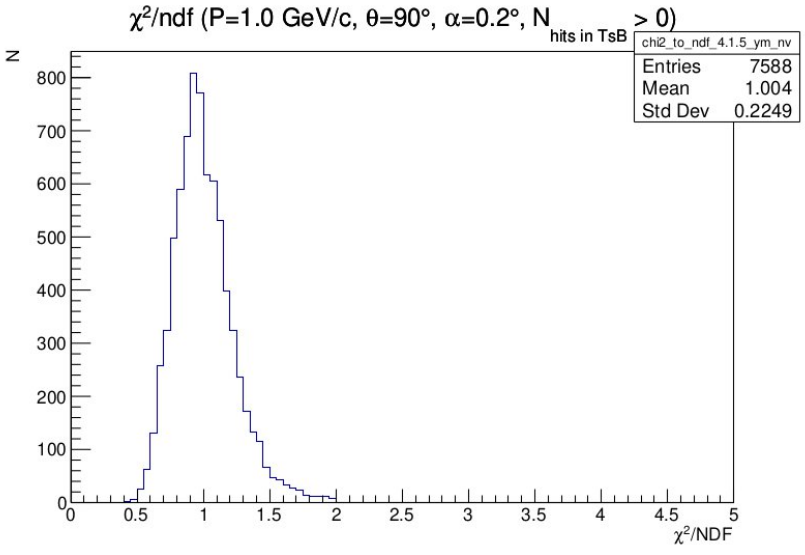
GetIsGood() == True

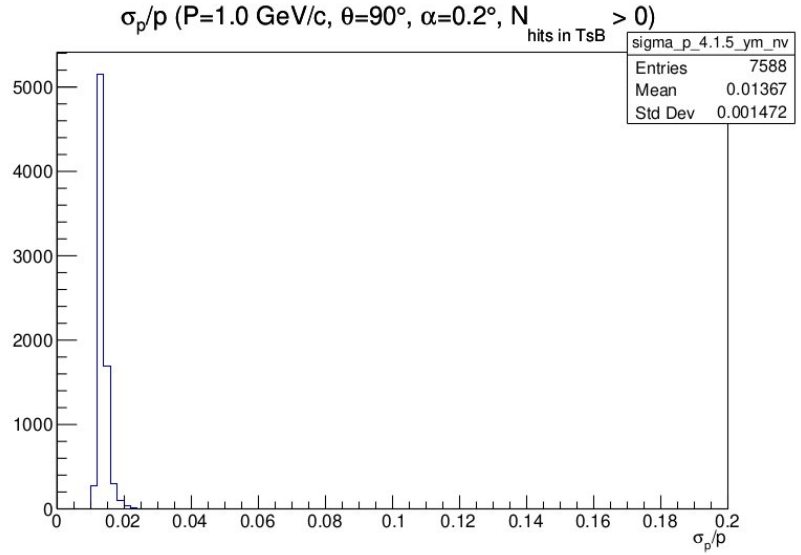
GetIsAcceptable() == True

GetChi2overNDF() <= 2

you are here








```
inline Bool_t SpdTrackFitPar::GetIsGood() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fConvergencyGF != 1) return false;
    return true;
}
```

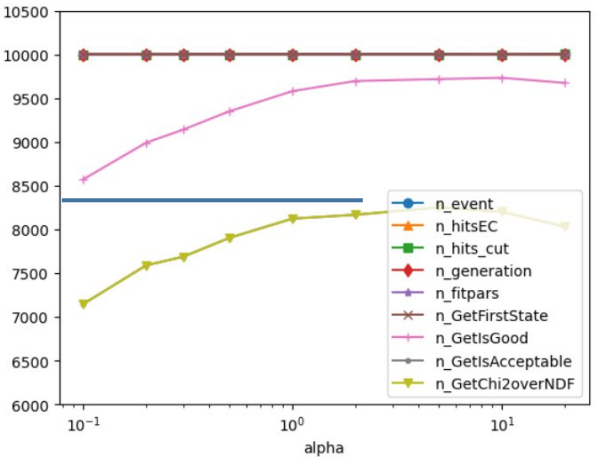
```
inline Bool_t SpdTrackFitPar::GetIsAcceptable() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fNDF < 3) return false;
    if (GetChi2overNDF() < 2) return true;
    return false;
}
```

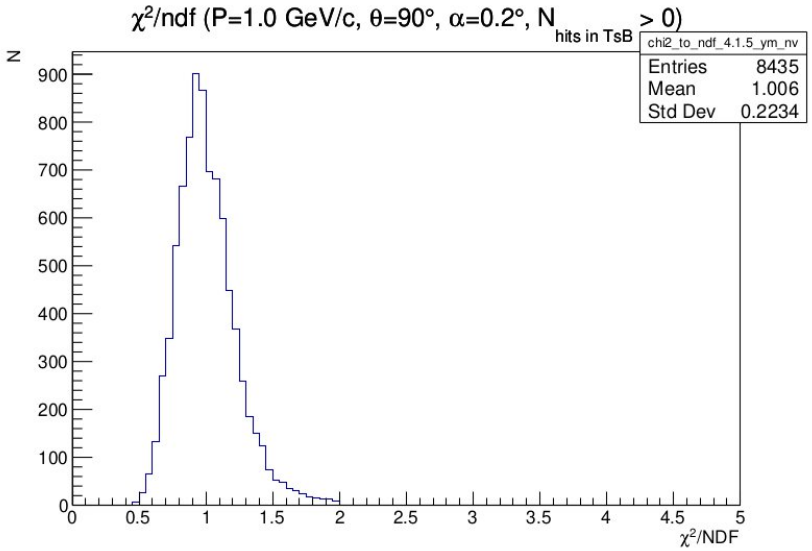
GetIsGood() == True

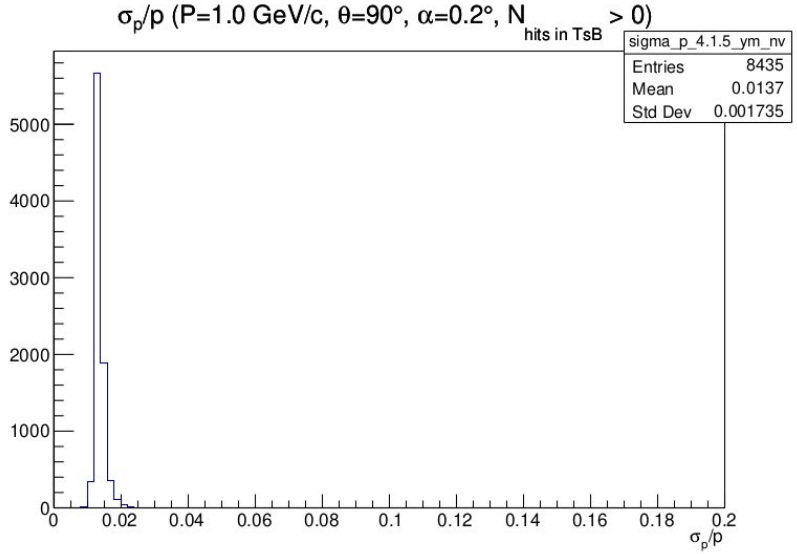
GetIsAcceptable() == True

GetChi2overNDF() <= 2

you are here








```
inline Bool_t SpdTrackFitPar::GetIsGood() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fConvergencyGF != 1) return false;
    return true;
}
```

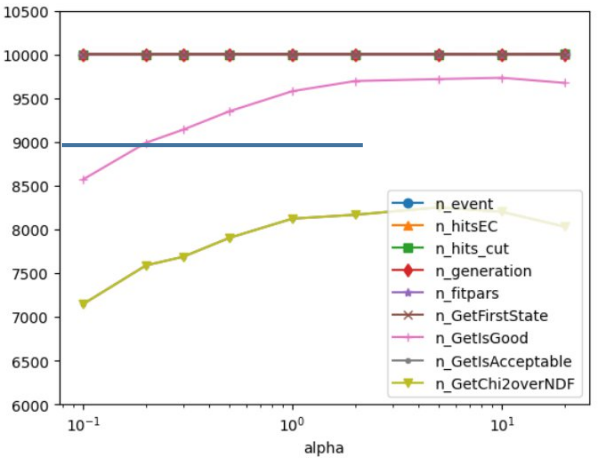
```
inline Bool_t SpdTrackFitPar::GetIsAcceptable() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fNDF < 3) return false;
    if (GetChi2overNDF() < 2) return true;
    return false;
}
```

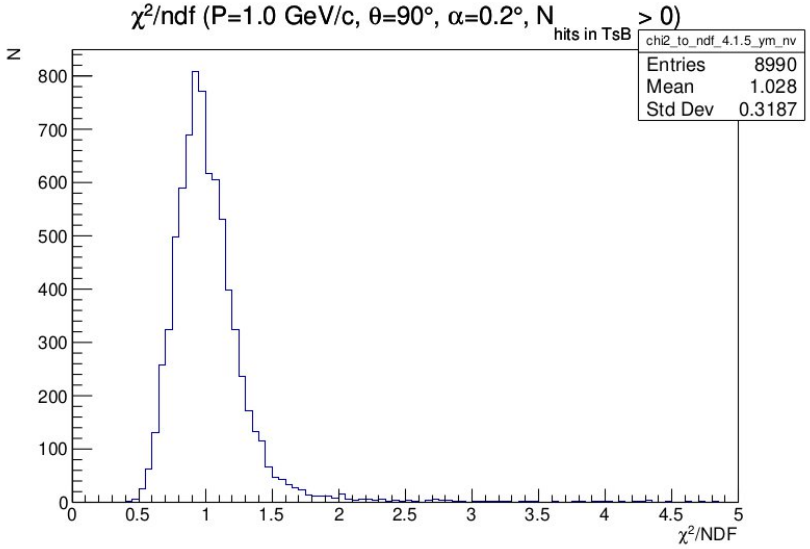
GetIsGood() == True

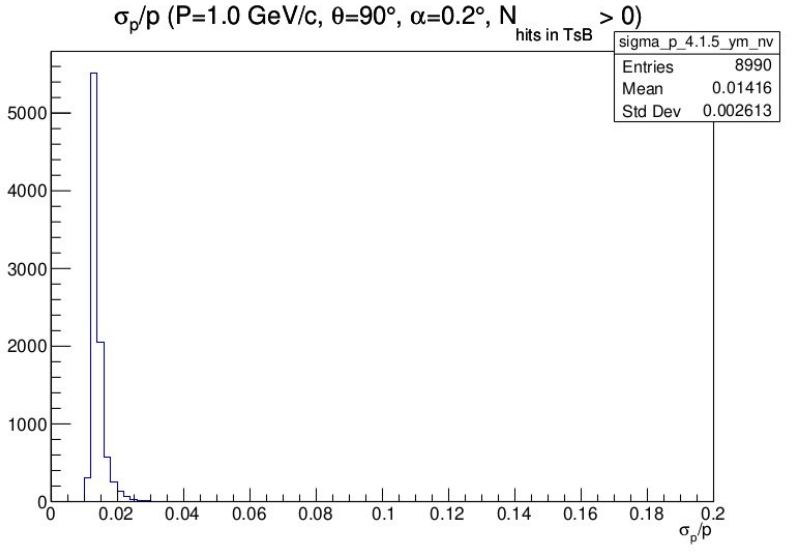
GetIsAcceptable() == True

GetChi2overNDF() <= 2

you are here








```
inline Bool_t SpdTrackFitPar::GetIsGood() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fConvergencyGF != 1) return false;
    return true;
}
```

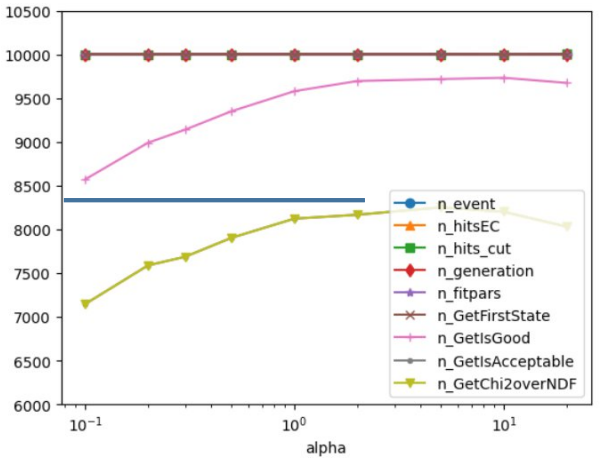
```
inline Bool_t SpdTrackFitPar::GetIsAcceptable() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fNDF < 3) return false;
    if (GetChi2overNDF() < 2) return true;
    return false;
}
```

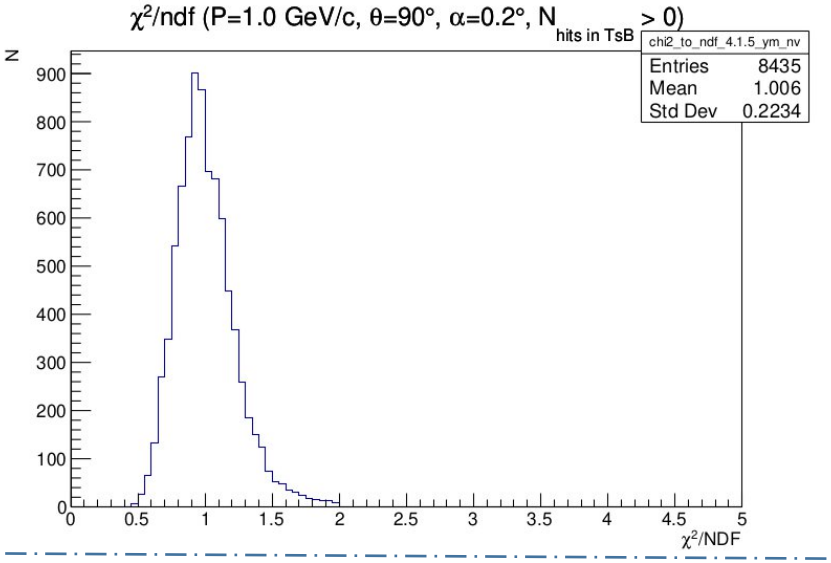
GetIsGood() == True

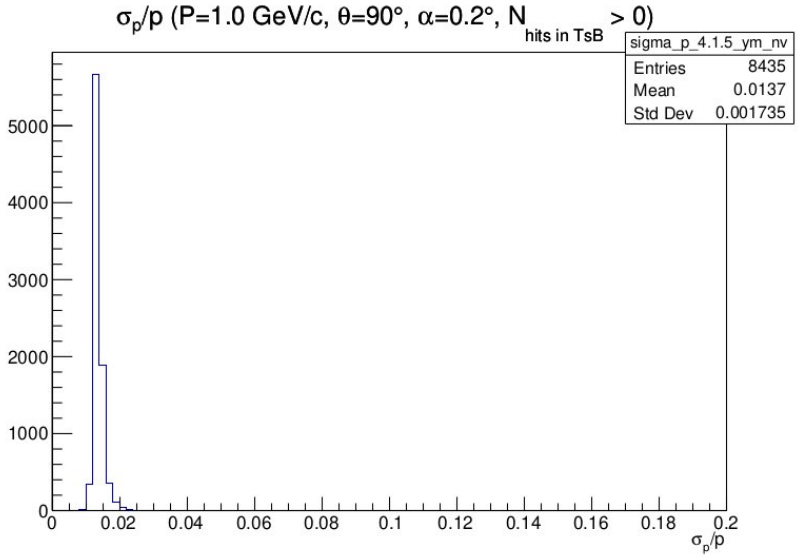
GetIsAcceptable() == True

GetChi2overNDF() <= 2

you are here








```
inline Bool_t SpdTrackFitPar::GetIsGood() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fConvergencyGF != 1) return false;
    return true;
}
```

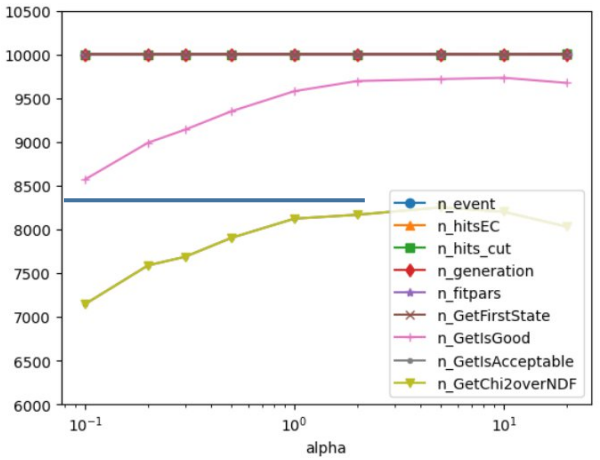
```
inline Bool_t SpdTrackFitPar::GetIsAcceptable() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fNDF < 3) return false;
    if (GetChi2overNDF() < 2) return true;
    return false;
}
```

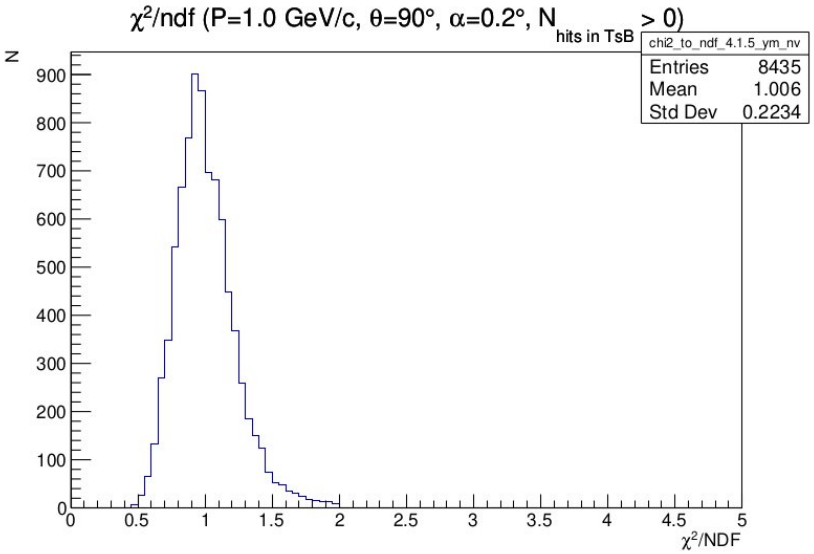
GetIsGood() == True

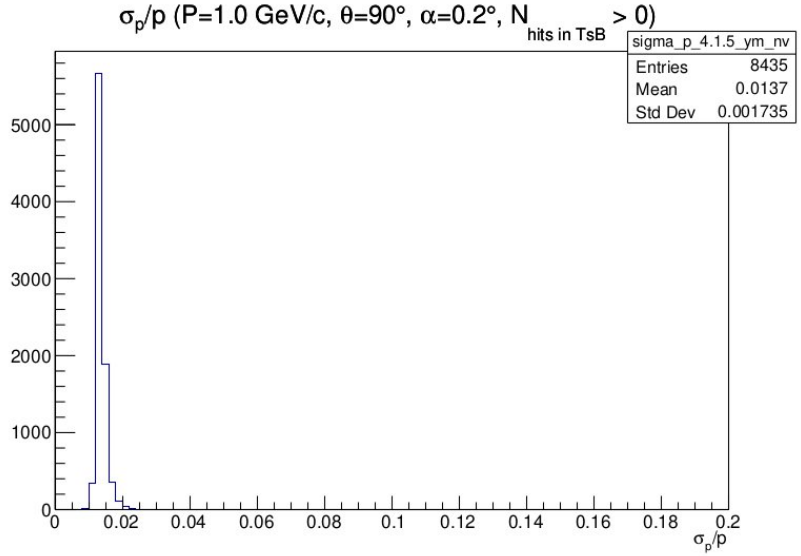
GetIsAcceptable() == True

GetChi2overNDF() <= 2

you are here



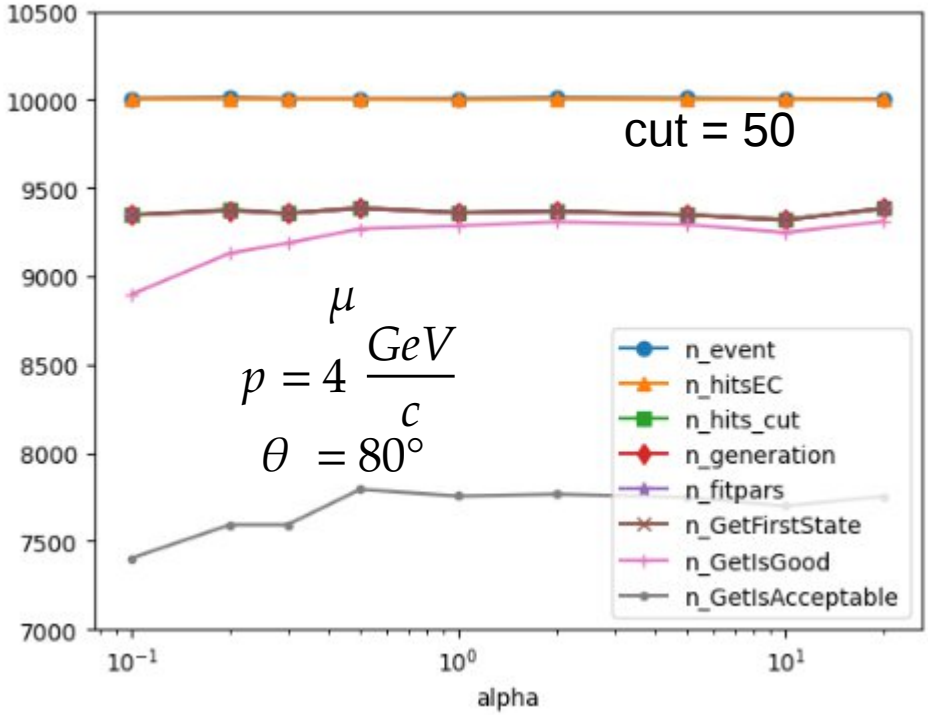


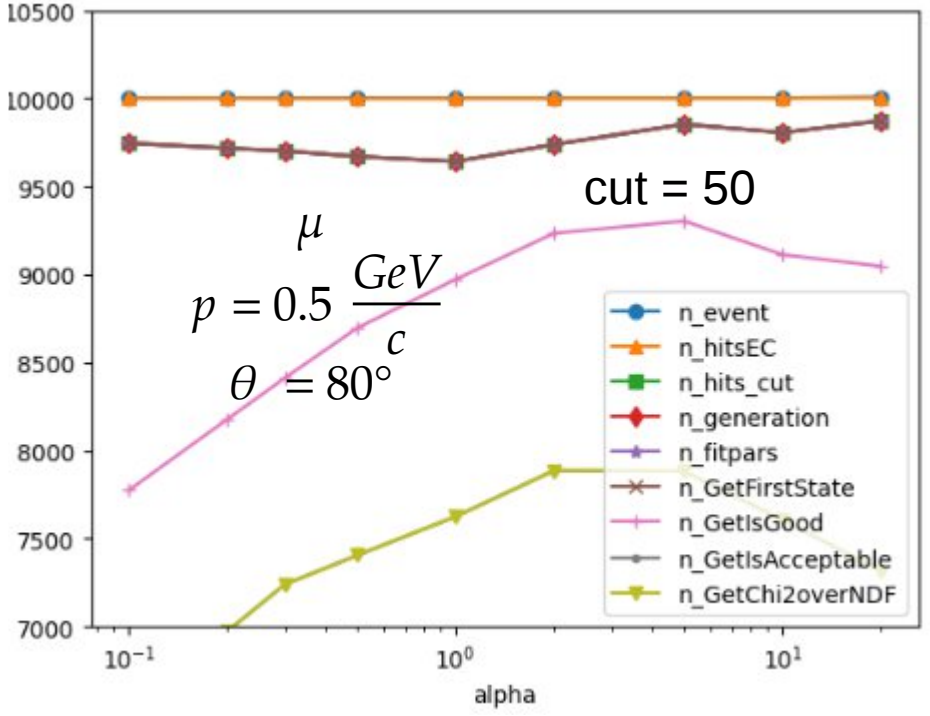



```
inline Bool_t SpdTrackFitPar::GetIsGood() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fConvergencyGF != 1) return false;
    return true;
}
```

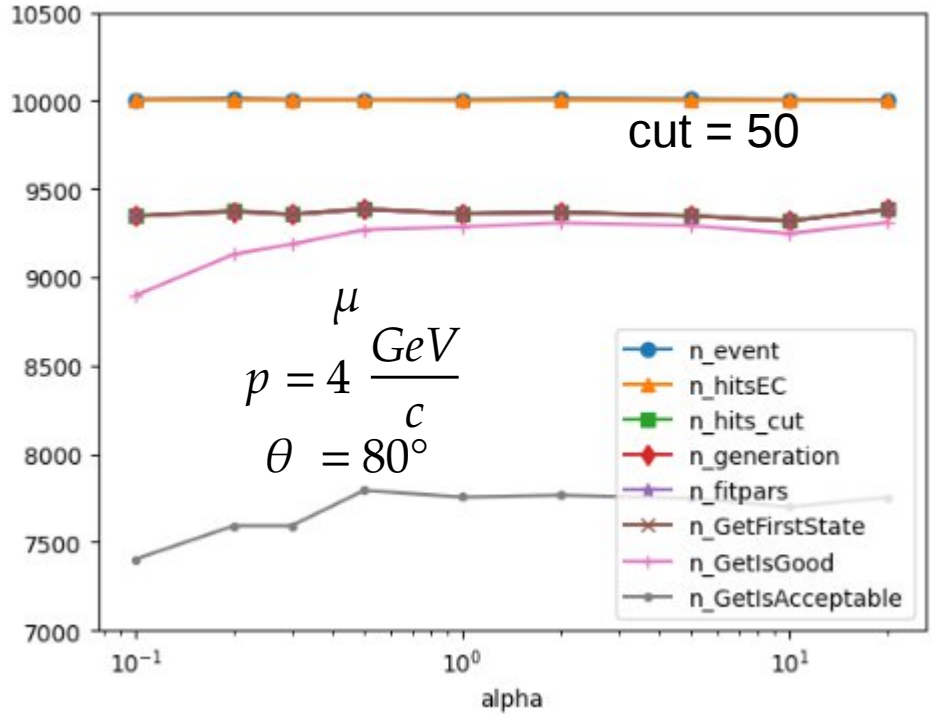
```
inline Bool_t SpdTrackFitPar::GetIsAcceptable() const
{
    if (fErrorFlag != 0) return false;
    if (HasErrorMesg()) return false;
    //if (fNFailedHits > 0) return false;
    if (fNDF < 3) return false;
    if (GetChi2overNDF() < 2) return true;
    return false;
}
```

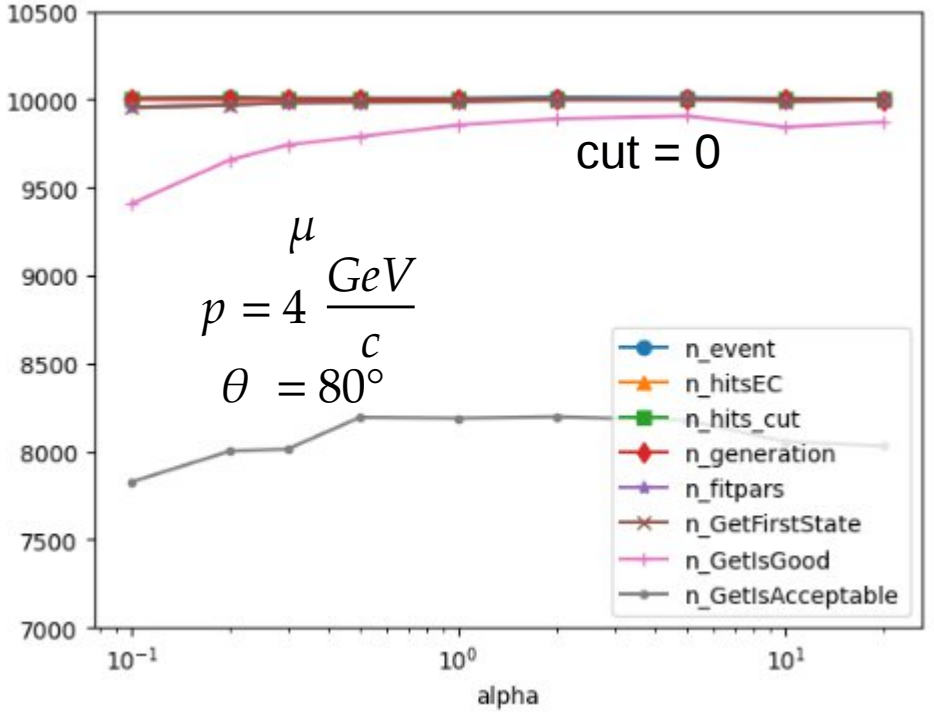
Efficiency drops as energy increases (this is strange)





Reconstruction efficiency at cut = 50 is 94%





GetIsAcceptable()

Has'n Error Flag

Has'n error message

NDF ≥ 3

χ/NDF ≤ 2

GetChi2overNDF()

χ/NDF ≤ 2