~~Uniquness Tracking~~
EVENT COUNTING

Benedikt Zihlmann

September 22, 2020

Introduction

Consider EVENT COUNTING for cross section purposes!
NO Dalitz analyses and the likes, where intermediate states are an important part of the analysis!

Get the basics right first.

Introduction

The issue of uniqueness of a Final State (FS) within an event.
Definition of a Final State as asked by an analysis:

- Number of Charged Tracks N_c^{FS}
- Number of Neutral Showers N_s^{FS} (FS = Reaction)

Introduction

The issue of uniqueness of a Final State (FS) within an event.
Definition of a Final State as asked by an analysis:

- Number of Charged Tracks N_c^{FS}
- Number of Neutral Showers N_s^{FS} (FS = Reaction)

Reconstructed Event:

- Number of Charged Tracks N_c^{event}
- Number of Neutral Showers N_s^{event}

Introduction

The issue of uniqueness of a Final State (FS) within an event.
Definition of a Final State as asked by an analysis:

- Number of Charged Tracks N_c^{FS}
- Number of Neutral Showers N_s^{FS} (FS = Reaction)

Reconstructed Event:

- Number of Charged Tracks N_c^{event}
- Number of Neutral Showers N_s^{event}
- $N_c^{event} \geq N_c^{FS}$
- $N_s^{event} \geq N_s^{FS}$

Introduction

The issue of uniqueness of a Final State (FS) within an event.
Definition of a Final State as asked by an analysis:

- Number of Charged Tracks N_c^{FS}
- Number of Neutral Showers N_s^{FS} (FS = Reaction)

Reconstructed Event:

- Number of Charged Tracks N_c^{event}
- Number of Neutral Showers N_s^{event}
- $N_c^{event} \geq N_c^{FS}$
- $N_s^{event} \geq N_s^{FS}$

good combo: Set of Final State particles in combination with a beam photon that satisfies all analysis cuts.

Introduction

The issue of uniqueness of a Final State (FS) within an event.
Definition of a Final State as asked by an analysis:

- Number of Charged Tracks N_c^{FS}
- Number of Neutral Showers N_s^{FS} (FS = Reaction)

Reconstructed Event:

- Number of Charged Tracks N_c^{event}
- Number of Neutral Showers N_s^{event}
- $N_c^{event} \geq N_c^{FS}$
- $N_s^{event} \geq N_s^{FS}$

good combo: Set of Final State particles in combination with a beam photon that satisfies all analysis cuts.

- **prompt** beam photon: **EVENT** NO MASS CONSTRAINT!

Introduction

The issue of uniqueness of a Final State (FS) within an event.
Definition of a Final State as asked by an analysis:

- Number of Charged Tracks N_c^{FS}
- Number of Neutral Showers N_s^{FS} (FS = Reaction)

Reconstructed Event:

- Number of Charged Tracks N_c^{event}
- Number of Neutral Showers N_s^{event}
- $N_c^{event} \geq N_c^{FS}$
- $N_s^{event} \geq N_s^{FS}$

good combo: Set of Final State particles in combination with a beam photon that satisfies all analysis cuts.

- **prompt** beam photon: **EVENT** NO MASS CONSTRAINT!
- **out-of-time** beam photon: **ACCIDENTAL**

Introduction

The issue of uniqueness of a Final State (FS) within an event.
Definition of a Final State as asked by an analysis:

- Number of Charged Tracks N_c^{FS}
- Number of Neutral Showers N_s^{FS} (FS = Reaction)

Reconstructed Event:

- Number of Charged Tracks N_c^{event}
- Number of Neutral Showers N_s^{event}
- $N_c^{event} \geq N_c^{FS}$
- $N_s^{event} \geq N_s^{FS}$

good combo: Set of Final State particles in combination with a beam photon that satisfies all analysis cuts.

- **prompt** beam photon: **EVENT** NO MASS CONSTRAINT!
- **out-of-time** beam photon: **ACCIDENTAL**
- **different FS particles OR mass constraints**

Examples 1

- $\gamma + p \to \eta + p$
  $\to \pi^0 + \pi^0 + \pi^0 + p$
  $\to \gamma + \gamma + \gamma + \gamma + \gamma + \gamma + p$ with m_{π^0} NOT constrained.

Examples 1

- $\gamma + p \to \eta + p$
  $\to \pi^0 + \pi^0 + \pi^0 + p$
  $\to \gamma + \gamma + \gamma + \gamma + \gamma + \gamma + p$ with m_{π^0} NOT constrained.
- Reaction Filter Tree: $N_c^{event} \geq 1$ and $N_s^{event} \geq 6$. ($N_c^{FS} = 1$, $N_s^{FS} = 6$)

Examples 1

- $\gamma + p \to \eta + p$
  $\to \pi^0 + \pi^0 + \pi^0 + p$
  $\to \gamma + \gamma + \gamma + \gamma + \gamma + \gamma + p$ with m_{π^0} NOT constrained.
- Reaction Filter Tree: $N_c^{event} \geq 1$ and $N_s^{event} \geq 6$. ($N_c^{FS} = 1$, $N_s^{FS} = 6$)
- The fact that the γs come from π^0 is irrelevant for **event counting!**

Examples 1

- $\gamma + p \to \eta + p$
  $\to \pi^0 + \pi^0 + \pi^0 + p$
  $\to \gamma + \gamma + \gamma + \gamma + \gamma + \gamma + p$ with m_{π^0} NOT constrained.
- Reaction Filter Tree: $N_c^{event} \geq 1$ and $N_s^{event} \geq 6$. ($N_c^{FS} = 1$, $N_s^{FS} = 6$)
- The fact that the γs come from π^0 is irrelevant for **event counting!**
- Many ways to combine the 6γs can make up 3π^0s. Each such combo is in the tree if the KinFit converged. (same χ^2 if M unconstraint.)
  However, this is irrelevant for the task of event counting.

Examples 1

- $\gamma + p \rightarrow \eta + p$
  $\rightarrow \pi^0 + \pi^0 + \pi^0 + p$
  $\rightarrow \gamma + \gamma + \gamma + \gamma + \gamma + \gamma + p$ with m_{π^0} NOT constrained.
- Reaction Filter Tree: $N_c^{event} \geq 1$ and $N_s^{event} \geq 6$. ($N_c^{FS} = 1$, $N_s^{FS} = 6$)
- The fact that the γs come from π^0 is irrelevant for **event counting!**
- Many ways to combine the 6γs can make up 3π^0s. Each such combo is in the tree if the KinFit converged. (same χ^2 if M unconstraint.)
  However, this is irrelevant for the task of event counting.
- Case of $N_c^{event} = 1$ and $N_s^{event} = 6$: Only one **good in-time** combo per event unless there is more than one **prompt** beam photon! What to do then?

good combo: Combo of FS particles with beam photon that survives all analysis cuts, including KinFit!

Examples 1

- $\gamma + p \rightarrow \eta + p$
  $\rightarrow \pi^0 + \pi^0 + \pi^0 + p$
  $\rightarrow \gamma + \gamma + \gamma + \gamma + \gamma + \gamma + p$ with m_{π^0} NOT constrained.
- Reaction Filter Tree: $N_c^{event} \geq 1$ and $N_s^{event} \geq 6$. ($N_c^{FS} = 1$, $N_s^{FS} = 6$)
- The fact that the γs come from π^0 is irrelevant for **event counting!**
- Many ways to combine the 6γs can make up 3π^0s. Each such combo is in the tree if the KinFit converged. (same χ^2 if M unconstraint.)
  However, this is irrelevant for the task of event counting.
- Case of $N_c^{event} = 1$ and $N_s^{event} = 6$: Only one **good in-time** combo per event unless there is more than one **prompt** beam photon! What to do then?

  Nothing other than accidental subtraction!

Examples 1

- $\gamma + p \rightarrow \eta + p$
  $\rightarrow \pi^0 + \pi^0 + \pi^0 + p$
  $\rightarrow \gamma + \gamma + \gamma + \gamma + \gamma + \gamma + p$ with m_{π^0} NOT constrained.
- Reaction Filter Tree: $N_c^{event} \geq 1$ and $N_s^{event} \geq 6$. ($N_c^{FS} = 1$, $N_s^{FS} = 6$)
- The fact that the γs come from π^0 is irrelevant for **event counting!**
- Many ways to combine the 6γs can make up 3π^0s. Each such combo is in the tree if the KinFit converged. (same χ^2 if M unconstraint.)
  However, this is irrelevant for the task of event counting.
- Case of $N_c^{event} = 1$ and $N_s^{event} = 6$: Only one **good in-time** combo per event unless there is more than one **prompt** beam photon! What to do then?

  Nothing other than accidental subtraction!
- All combos will have the same KinFit χ^2 and all Momenta are the same.

Examples 1

- $\gamma + p \to \eta + p$
  $\to \pi^0 + \pi^0 + \pi^0 + p$
  $\to \gamma + \gamma + \gamma + \gamma + \gamma + \gamma + p$ with m_{π^0} NOT constrained.
- Reaction Filter Tree: $N_c^{event} \geq 1$ and $N_s^{event} \geq 6$. ($N_c^{FS} = 1$, $N_s^{FS} = 6$)
- The fact that the γs come from π^0 is irrelevant for **event counting!**
- Many ways to combine the 6γs can make up 3π^0s. Each such combo is in the tree if the KinFit converged. (same χ^2 if M unconstraint.)
  However, this is irrelevant for the task of event counting.
- Case of $N_c^{event} = 1$ and $N_s^{event} = 6$: Only one **good in-time** combo per event unless there is more than one **prompt** beam photon! What to do then?

  Nothing other than accidental subtraction!
- All combos will have the same KinFit χ^2 and all Momenta are the same.
- Case of $N_c^{event} > 1$ and/or $N_s^{event} > 6$: More than one combo of 6γs with **the same prompt** beam photon makes a **good** combo! (Not all 6γs are the same)
  What to do then?
  this is the issue to discuss here!

Examples 2

- $\gamma + p \to \eta\prime + p + \pi^{+} + \pi^{-}$
  $\to \eta + \pi^{0} + p + \pi^{+} + \pi^{-} \to 4\gamma + p + \pi^{+} + \pi^{-}$

Examples 2

- $\gamma + p \rightarrow \eta\prime + p + \pi^+ + \pi^-$
  $\rightarrow \eta + \pi^0 + p + \pi^+ + \pi^- \rightarrow 4\gamma + p + \pi^+ + \pi^-$
- $N_s^{FS} = 4$, with two different intermediate states π^0, η

Examples 2

- $\gamma + p \rightarrow \eta\prime + p + \pi^+ + \pi^-$
  $\rightarrow \eta + \pi^0 + p + \pi^+ + \pi^- \rightarrow 4\gamma + p + \pi^+ + \pi^-$
- $N_s^{FS} = 4$, with two different intermediate states π^0, η
- If masses are unconstrained, intermediate states combos, π^0 and η do not matter. All four γs will have the same 4-momenta for any combo, BUT IF ....

Examples 2

- $\gamma + p \rightarrow \eta\prime + p + \pi^+ + \pi^-$
  $\rightarrow \eta + \pi^0 + p + \pi^+ + \pi^- \rightarrow 4\gamma + p + \pi^+ + \pi^-$
- $N_s^{FS} = 4$, with two different intermediate states π^0, η
- If masses are unconstrained, intermediate states combos, π^0 and η do not matter. All four γs will have the same 4-momenta for any combo, BUT IF ....
- $N_c^{FS} = 3 > 1$: role of p and π^+ can switch, if NO PID! Different χ^2 (Mass Constraint)!

Examples 2

- $\gamma + p \rightarrow \eta\prime + p + \pi^+ + \pi^-$
  $\rightarrow \eta + \pi^0 + p + \pi^+ + \pi^- \rightarrow 4\gamma + p + \pi^+ + \pi^-$
- $N_s^{FS} = 4$, with two different intermediate states π^0, η
- If masses are unconstrained, intermediate states combos, π^0 and η do not matter. All four γs will have the same 4-momenta for any combo, BUT IF ....
- $N_c^{FS} = 3 > 1$: role of p and π^+ can switch, if NO PID! Different χ^2 (Mass Constraint)!
- Possible double counting even at $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ because p and π^+ can switch roles

Examples 2

- $\gamma + p \rightarrow \eta\prime + p + \pi^+ + \pi^-$
  $\rightarrow \eta + \pi^0 + p + \pi^+ + \pi^- \rightarrow 4\gamma + p + \pi^+ + \pi^-$
- $N_s^{FS} = 4$, with two different intermediate states π^0, η
- If masses are unconstrained, intermediate states combos, π^0 and η do not matter. All four γs will have the same 4-momenta for any combo, BUT IF ....
- $N_c^{FS} = 3 > 1$: role of p and π^+ can switch, if NO PID! Different χ^2 (Mass Constraint)!
- Possible double counting even at $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ because p and π^+ can switch roles
- This is true for any Final State with $N_c^{FS} > 1$ and same charge.
  (p,π^+,K^+) or ($\bar{p}$, π^-,K^-)

Examples 2

- $\gamma + p \rightarrow \eta\prime + p + \pi^+ + \pi^-$
  $\rightarrow \eta + \pi^0 + p + \pi^+ + \pi^- \rightarrow 4\gamma + p + \pi^+ + \pi^-$
- $N_s^{FS} = 4$, with two different intermediate states π^0, η
- If masses are unconstrained, intermediate states combos, π^0 and η do not matter. All four γs will have the same 4-momenta for any combo, BUT IF ....
- $N_c^{FS} = 3 > 1$: role of p and π^+ can switch, if NO PID! Different χ^2 (Mass Constraint)!
- Possible double counting even at $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ because p and π^+ can switch roles
- This is true for any Final State with $N_c^{FS} > 1$ and same charge.
  (p,π^+,K^+) or ($\bar{p}$, π^-,K^-)
- This will happen even with using `std::set` and `std::map` in DSelector doing uniqueness tracking following the example code. (explicit differentiation between same-charged particle types)

Examples 2

- $\gamma + p \rightarrow \eta\prime + p + \pi^+ + \pi^-$
  $\rightarrow \eta + \pi^0 + p + \pi^+ + \pi^- \rightarrow 4\gamma + p + \pi^+ + \pi^-$
- $N_s^{FS} = 4$, with two different intermediate states π^0, η
- If masses are unconstrained, intermediate states combos, π^0 and η do not matter. All four γs will have the same 4-momenta for any combo, BUT IF ....
- $N_c^{FS} = 3 > 1$: role of p and π^+ can switch, if NO PID! Different χ^2 (Mass Constraint)!
- Possible double counting even at $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ because p and π^+ can switch roles
- This is true for any Final State with $N_c^{FS} > 1$ and same charge. (p,π^+,K^+) or ($\bar{p}$, π^-,K^-)
- This will happen even with using `std::set` and `std::map` in DSelector doing uniqueness tracking following the example code. (explicit differentiation between same-charged particle types)
- This effect is reaction dependent. (Peter Pauli sees level of 1.5%)

Examples 2

- $\gamma + p \rightarrow \eta\prime + p + \pi^+ + \pi^-$
  $\rightarrow \eta + \pi^0 + p + \pi^+ + \pi^- \rightarrow 4\gamma + p + \pi^+ + \pi^-$
- $N_s^{FS} = 4$, with two different intermediate states π^0, η
- If masses are unconstrained, intermediate states combos, π^0 and η do not matter. All four γs will have the same 4-momenta for any combo, BUT IF ....
- $N_c^{FS} = 3 > 1$: role of p and π^+ can switch, if NO PID! Different χ^2 (Mass Constraint)!
- Possible double counting even at $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ because p and π^+ can switch roles
- This is true for any Final State with $N_c^{FS} > 1$ and same charge. (p,π^+,K^+) or ($\bar{p}$, π^-,K^-)
- This will happen even with using `std::set` and `std::map` in DSelector doing uniqueness tracking following the example code. (explicit differentiation between same-charged particle types)
- This effect is reaction dependent. (Peter Pauli sees level of 1.5%)
- Additional photons, same issue as example 1

Some Comments On Counting

Cases Of More Than One Unique FS **good** combo per event:

- $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ PLUS more than one **prompt** beam photon
  This is taken care off by accidental subtraction, UNLESS ($N_{c+or-}^{FS} > 1$)

Some Comments On Counting

Cases Of More Than One Unique FS **good** combo per event:

- $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ PLUS more than one **prompt** beam photon
  This is taken care off by accidental subtraction, UNLESS ($N_{c+or-}^{FS} > 1$)
- $N_c^{event} > N_c^{FS}$ or $N_s^{event} > N_s^{FS}$ and no cuts on unused energy and additional charged tracks.

Some Comments On Counting

Cases Of More Than One Unique FS **good** combo per event:

- $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ PLUS more than one **prompt** beam photon
  This is taken care off by accidental subtraction, UNLESS ($N_{c+or-}^{FS} > 1$)
- $N_c^{event} > N_c^{FS}$ or $N_s^{event} > N_s^{FS}$ and no cuts on unused energy and additional charged tracks.
- Combination of 1 and 2

Some Comments On Counting

Cases Of More Than One Unique FS **good** combo per event:

- $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ PLUS more than one **prompt** beam photon
  This is taken care off by accidental subtraction, UNLESS ($N_{c+or-}^{FS} > 1$)
- $N_c^{event} > N_c^{FS}$ or $N_s^{event} > N_s^{FS}$ and no cuts on unused energy and additional charged tracks.
- Combination of 1 and 2
- These instances will lead to multiple counting even when using the methods of `std::set` and `std::map` in DSelector

Some Comments On Counting

Cases Of More Than One Unique FS **good** combo per event:

- $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ PLUS more than one **prompt** beam photon
  This is taken care off by accidental subtraction, UNLESS ($N_{c+or-}^{FS} > 1$)
- $N_c^{event} > N_c^{FS}$ or $N_s^{event} > N_s^{FS}$ and no cuts on unused energy and additional charged tracks.
- Combination of 1 and 2
- These instances will lead to multiple counting even when using the methods of `std::set` and `std::map` in DSelector

What to do in these cases?

Some Comments On Counting

Cases Of More Than One Unique FS **good** combo per event:

- $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ PLUS more than one **prompt** beam photon
  This is taken care off by accidental subtraction, UNLESS ($N_{c+or-}^{FS} > 1$)
- $N_c^{event} > N_c^{FS}$ or $N_s^{event} > N_s^{FS}$ and no cuts on unused energy and additional charged tracks.
- Combination of 1 and 2
- These instances will lead to multiple counting even when using the methods of `std::set` and `std::map` in DSelector

What to do in these cases?

A) Select the good combo with the best χ^2/df ?

Some Comments On Counting

Cases Of More Than One Unique FS **good** combo per event:

- $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ PLUS more than one **prompt** beam photon
  This is taken care off by accidental subtraction, UNLESS ($N_{c+or-}^{FS} > 1$)
- $N_c^{event} > N_c^{FS}$ or $N_s^{event} > N_s^{FS}$ and no cuts on unused energy and additional charged tracks.
- Combination of 1 and 2
- These instances will lead to multiple counting even when using the methods of `std::set` and `std::map` in DSelector

What to do in these cases?

A) Select the good combo with the best χ^2/df ?

B) Count all good combos and weight them by $1/\sum(goodFScombo)$. If you do accidental subtraction this weight has to be multiplied to 1/N_{acc-p} ?

Some Comments On Counting

Cases Of More Than One Unique FS **good** combo per event:

- $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ PLUS more than one **prompt** beam photon
  This is taken care off by accidental subtraction, UNLESS ($N_{c+or-}^{FS} > 1$)
- $N_c^{event} > N_c^{FS}$ or $N_s^{event} > N_s^{FS}$ and no cuts on unused energy and additional charged tracks.
- Combination of 1 and 2
- These instances will lead to multiple counting even when using the methods of `std::set` and `std::map` in DSelector

What to do in these cases?

A) Select the good combo with the best χ^2/df ?

B) Count all good combos and weight them by $1/\sum(goodFScombo)$. If you do accidental subtraction this weight has to be multiplied to $1/N_{acc-p}$?

C) combination of A) and B): weight each good combo "i" with $\chi_i^2/\sum_{i=1}^{i=N}\chi_i^2$

Some Comments On Counting

Cases Of More Than One Unique FS **good** combo per event:

- $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ PLUS more than one **prompt** beam photon
  This is taken care off by accidental subtraction, UNLESS ($N_{c+or-}^{FS} > 1$)
- $N_c^{event} > N_c^{FS}$ or $N_s^{event} > N_s^{FS}$ and no cuts on unused energy and additional charged tracks.
- Combination of 1 and 2
- These instances will lead to multiple counting even when using the methods of `std::set` and `std::map` in DSelector

What to do in these cases?

A) Select the good combo with the best χ^2/df ?

B) Count all good combos and weight them by $1/\sum(goodFScombo)$. If you do accidental subtraction this weight has to be multiplied to 1/N$_{acc-p}$?

C) combination of A) and B): weight each good combo "i" with $\chi_i^2/\sum_{i=1}^{i=N}\chi_i^2$

D) Other approach ?

Some Comments On Counting

Cases Of More Than One Unique FS **good** combo per event:

- $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ PLUS more than one **prompt** beam photon
  This is taken care off by accidental subtraction, UNLESS ($N^{FS}_{c+or-} > 1$)
- $N_c^{event} > N_c^{FS}$ or $N_s^{event} > N_s^{FS}$ and no cuts on unused energy and additional charged tracks.
- Combination of 1 and 2
- These instances will lead to multiple counting even when using the methods of `std::set` and `std::map` in DSelector

What to do in these cases?

A) Select the good combo with the best χ^2/df ?

B) Count all good combos and weight them by $1/\sum(goodFScombo)$. If you do accidental subtraction this weight has to be multiplied to 1/N$_{acc-p}$?

C) combination of A) and B): weight each good combo "i" with $\chi_i^2/\sum_{i=1}^{i=N}\chi_i^2$

D) Other approach ?

$\rightarrow$ Requires a recommendation by the group and example of how to do it.

Some Comments On Counting

Cases Of More Than One Unique FS **good** combo per event:

- $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ PLUS more than one **prompt** beam photon
  This is taken care off by accidental subtraction, UNLESS ($N_{c+or-}^{FS} > 1$)
- $N_c^{event} > N_c^{FS}$ or $N_s^{event} > N_s^{FS}$ and no cuts on unused energy and additional charged tracks.
- Combination of 1 and 2
- These instances will lead to multiple counting even when using the methods of `std::set` and `std::map` in DSelector

What to do in these cases?

A) Select the good combo with the best χ^2/df ?

B) Count all good combos and weight them by $1/\sum(goodFScombo)$. If you do accidental subtraction this weight has to be multiplied to $1/N_{acc-p}$?

C) combination of A) and B): weight each good combo "i" with $\chi_i^2/\sum_{i=1}^{i=N}\chi_i^2$

D) Other approach ?

→ Requires a recommendation by the group and example of how to do it.

- This does NOT address the issue of background subtraction!

Some Comments On Counting

Cases Of More Than One Unique FS **good** combo per event:

- $N_c^{event} = N_c^{FS}$ and $N_s^{event} = N_s^{FS}$ PLUS more than one **prompt** beam photon
  This is taken care off by accidental subtraction, UNLESS ($N_{c+or-}^{FS} > 1$)
- $N_c^{event} > N_c^{FS}$ or $N_s^{event} > N_s^{FS}$ and no cuts on unused energy and additional charged tracks.
- Combination of 1 and 2
- These instances will lead to multiple counting even when using the methods of `std::set` and `std::map` in DSelector

What to do in these cases?

A) Select the good combo with the best χ^2/df ?
B) Count all good combos and weight them by $1/\sum(goodFScombo)$. If you do accidental subtraction this weight has to be multiplied to 1/N$_{acc-p}$?
C) combination of A) and B): weight each good combo "i" with $\chi_i^2/\sum_{i=1}^{i=N}\chi_i^2$
D) Other approach ?

→ Requires a recommendation by the group and example of how to do it.

- This does NOT address the issue of background subtraction!
- This does NOT address the issue of intermediate final states (e.q. Dalitz plot analysis)

Lessons learned so far

- Outline of the problem needs carefull language!

Lessons learned so far

- Outline of the problem needs carefull language!
- Do not ask for intermediate state (η, π^0) mass constraints! (Need KinFit in DSelector) **The task at hand is event counting only**.

Lessons learned so far

- Outline of the problem needs carefull language!
- Do not ask for intermediate state (η, π^0) mass constraints! (Need KinFit in DSelector) **The task at hand is event counting only**.
- PID choice (charged particle) is like mass constraint for neutrals.

Lessons learned so far

- Outline of the problem needs carefull language!
- Do not ask for intermediate state (η, π^0) mass constraints! (Need KinFit in DSelector) **The task at hand is event counting only**.
- PID choice (charged particle) is like mass constraint for neutrals.
- MC data: It can happen that the reconstructed event-combo matched to the thrown particles does NOT converge in the KinFit! But another combo DOES!

Lessons learned so far

- Outline of the problem needs carefull language!
- Do not ask for intermediate state (η, π^0) mass constraints! (Need KinFit in DSelector) **The task at hand is event counting only**.
- PID choice (charged particle) is like mass constraint for neutrals.
- MC data: It can happen that the reconstructed event-combo matched to the thrown particles does NOT converge in the KinFit! But another combo DOES!

  This happens easily 10 to 20% of the time, but is reaction dependent!

Lessons learned so far

- Outline of the problem needs carefull language!
- Do not ask for intermediate state (η, π^0) mass constraints! (Need KinFit in DSelector) **The task at hand is event counting only**.
- PID choice (charged particle) is like mass constraint for neutrals.
- MC data: It can happen that the reconstructed event-combo matched to the thrown particles does NOT converge in the KinFit! But another combo DOES!

  This happens easily 10 to 20% of the time, but is reaction dependent!

  Cannot simply use matched-MC final states as normalization! Interpretation of MC data reconstruction is not straight forward.

Lessons learned so far

- Outline of the problem needs carefull language!
- Do not ask for intermediate state (η, π^0) mass constraints! (Need KinFit in DSelector) **The task at hand is event counting only**.
- PID choice (charged particle) is like mass constraint for neutrals.
- MC data: It can happen that the reconstructed event-combo matched to the thrown particles does NOT converge in the KinFit! But another combo DOES!

  This happens easily 10 to 20% of the time, but is reaction dependent!

  Cannot simply use matched-MC final states as normalization! Interpretation of MC data reconstruction is not straight forward.
- Using unused energy may not be worthwhile the trouble! Big source of background. (obviously reaction dependent)

Lessons learned so far

- Outline of the problem needs carefull language!
- Do not ask for intermediate state (η, π^0) mass constraints! (Need KinFit in DSelector) **The task at hand is event counting only**.
- PID choice (charged particle) is like mass constraint for neutrals.
- MC data: It can happen that the reconstructed event-combo matched to the thrown particles does NOT converge in the KinFit! But another combo DOES!

  This happens easily 10 to 20% of the time, but is reaction dependent!

  Cannot simply use matched-MC final states as normalization! Interpretation of MC data reconstruction is not straight forward.
- Using unused energy may not be worthwhile the trouble! Big source of background. (obviously reaction dependent)
- Same with additional charged tracks in the event.

Lessons learned so far

- Outline of the problem needs carefull language!
- Do not ask for intermediate state (η, π^0) mass constraints! (Need KinFit in DSelector) **The task at hand is event counting only**.
- PID choice (charged particle) is like mass constraint for neutrals.
- MC data: It can happen that the reconstructed event-combo matched to the thrown particles does NOT converge in the KinFit! But another combo DOES!

  This happens easily 10 to 20% of the time, but is reaction dependent!

  Cannot simply use matched-MC final states as normalization! Interpretation of MC data reconstruction is not straight forward.
- Using unused energy may not be worthwhile the trouble! Big source of background. (obviously reaction dependent)
- Same with additional charged tracks in the event.
- Shower quality cut, not clear of its benefits, clearly reaction dependent as well.

Lessons learned so far

- Outline of the problem needs carefull language!
- Do not ask for intermediate state (η, π^0) mass constraints! (Need KinFit in DSelector) **The task at hand is event counting only**.
- PID choice (charged particle) is like mass constraint for neutrals.
- MC data: It can happen that the reconstructed event-combo matched to the thrown particles does NOT converge in the KinFit! But another combo DOES!

  This happens easily 10 to 20% of the time, but is reaction dependent!

  Cannot simply use matched-MC final states as normalization! Interpretation of MC data reconstruction is not straight forward.
- Using unused energy may not be worthwhile the trouble! Big source of background. (obviously reaction dependent)
- Same with additional charged tracks in the event.
- Shower quality cut, not clear of its benefits, clearly reaction dependent as well.

That is all at this point.

Simplest Final State Example

Simplest Final State: 1 Charted track, 2 photons
Examples: $\gamma + p \rightarrow p + \pi^0$ or $\gamma + p \rightarrow p + \eta$ with $\pi^0(\eta) \rightarrow \gamma\gamma$

Simplest Final State Example

Simplest Final State: 1 Charted track, 2 photons
Examples: $\gamma + p \rightarrow p + \pi^0$ or $\gamma + p \rightarrow p + \eta$ with $\pi^0(\eta) \rightarrow \gamma\gamma$

- Events with only 1 charged track and 2 neutrals. **DONE.** There is only one Final State combo! At this point all beam photons in combination with the 3 FS particles are either in the "prompt" peak or in the side peaks "accidentals". Most of the time there is only one prompt beam photon, but some times you will have more than one and usually they do not have the same energy. One of them is the correct beam photon that initiated the event (most likely, modulo efficiency) and all others are "accidentals". These are the accidentals underneath the prompt peak that need to be subtracted. And this subtraction is done by using the accidental beam photons, those that are in the side peaks. This assumes that the number of accidentals underneath the prompt peak is the same amount as in the side peaks, which we know is almost true but not quite, hence the accidental scaling factor.

Simplest Final State Example

Simplest Final State: 1 Charted track, 2 photons
Examples: $\gamma + p \rightarrow p + \pi^0$ or $\gamma + p \rightarrow p + \eta$ with $\pi^0(\eta) \rightarrow \gamma\gamma$

- Events with only 1 charged track and 2 neutrals. **DONE.** There is only one Final State combo! At this point all beam photons in combination with the 3 FS particles are either in the "prompt" peak or in the side peaks "accidentals".
- Events with more than one charged track and/or more than 2 neutrals. **Additional charged tracks and/or unused energy in the tree for the event!** In This case there is a potential to have **more than one UNIQUE FS** combo that survives with a prompt beam photon all analysis cuts. In this case we have to count all these FS combos (=N) in the event and weight all combos that surve all cuts with an additional facctor 1/N. (This is one way to handle it! This is OPEN for debate, this is our JOB to give a recommendation of what to do!)

Event Statistics

Final State 3 charged tracks and 6 neutrals:

$\gamma + p \rightarrow p + \pi^{+} + \pi^{-} + 3\pi^{0}$

EVEN IF THE EVENT HAS EXACTLY THE SAME NUMBER AND TYPE OF PARTICLES AS THE FS THERE IS POTENTIAL FOR MORE THAN ONE FS COMOBO!

Event Statistics

Final State 3 charged tracks and 6 neutrals:
$\gamma + p \to p + \pi^+ + \pi^- + 3\pi^0$

- Events with exactly 6 FS γs: 44.8%

Event Statistics

Final State 3 charged tracks and 6 neutrals:
$\gamma + p \rightarrow p + \pi^{+} + \pi^{-} + 3\pi^{0}$

- Events with exactly 6 FS γs: 44.8%
- and with exactly 3 charged tracks: 95.3% of those
- of those: 86% have exactly one prompt beam photon
  12.2% have two prompt beam photons
  1.6% have tree prompt beam photons
- 0nly 0.15% have more than one Q combo

Event Statistics

Final State 3 charged tracks and 6 neutrals:
$\gamma + p \rightarrow p + \pi^{+} + \pi^{-} + 3\pi^{0}$

- Events with exactly 6 FS γs: 44.8%
- and with exactly 3 charged tracks: 95.3% of those
- of those: 86% have exactly one prompt beam photon
  12.2% have two prompt beam photons
  1.6% have tree prompt beam photons
- 0nly 0.15% have more than one Q combo
- Events with more than 6 FS γs: 55.2%

Event Statistics

Final State 3 charged tracks and 6 neutrals:
$\gamma + p \rightarrow p + \pi^{+} + \pi^{-} + 3\pi^{0}$

- Events with exactly 6 FS γs: 44.8%
- and with exactly 3 charged tracks: 95.3% of those
- of those: 86% have exactly one prompt beam photon
  12.2% have two prompt beam photons
  1.6% have tree prompt beam photons
- 0nly 0.15% have more than one Q combo
- Events with more than 6 FS γs: 55.2%
- the mean Neutral combos is 2.5

Event Statistics

Final State 3 charged tracks and 6 neutrals:
$\gamma + p \to p + \pi^+ + \pi^- + 3\pi^0$

- Events with exactly 6 FS γs: 44.8%
- and with exactly 3 charged tracks: 95.3% of those
- of those: 86% have exactly one prompt beam photon
  12.2% have two prompt beam photons
  1.6% have tree prompt beam photons
- 0nly 0.15% have more than one Q combo
- Events with more than 6 FS γs: 55.2%
- the mean Neutral combos is 2.5
- only 0.5% have more than one Q combo