



Stage de L3 sur l'expérience SHiP

Reconstruction dans FairShip de la désintégration

$$HNL \longrightarrow e^{\pm} + \rho^{\mp}$$

Michel Fabrice Serret

LPNHE, Université Pierre et Marie Curie, 4 place Jussieu 75015 France

Contents

Contents	2
1 Remerciements	3
2 Introduction	3
2.1 SHIP - Search for Hidden Particles	3
3 Methode de Reconstruction	6
3.1 Coupures	6
3.2 Canal $\mu\pi$	6
3.3 Canal $e\rho$	6
3.3.1 Identification des particules chargées	6
3.3.2 Vertex de désintégration et Pion neutre	8
3.3.3 Reconstruction ρ et HNL	8
4 Coupure spécifique	10
5 Code	11
5.1 Philosophie du Code	11
5.1.1 Objet Python	12
6 Conclusion	15
References	15

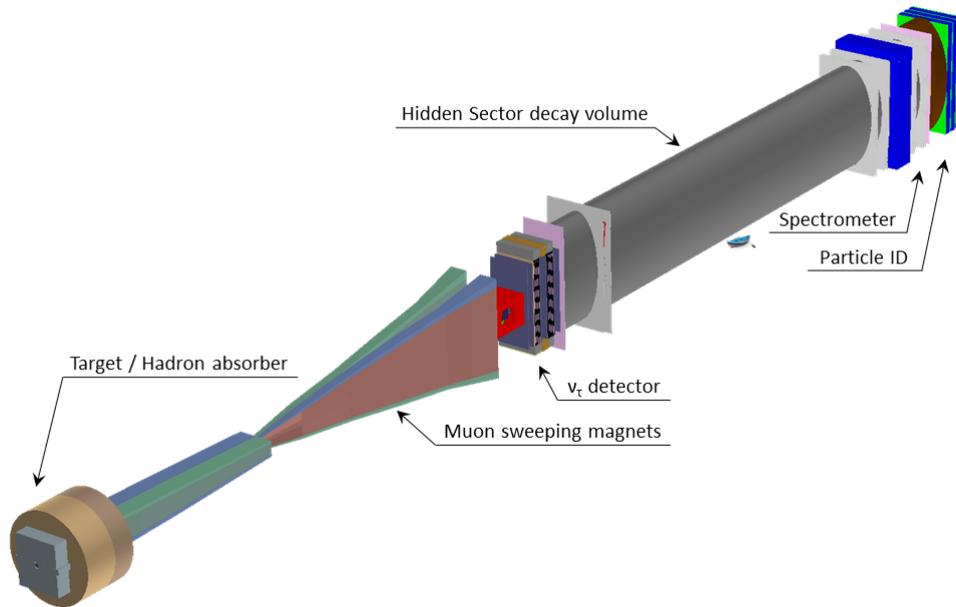


Figure 1: Schéma de l'expérience SHIP

1 Remerciements

Je tiens à remercier mon maître de stage M. Jacques Chauveau, ainsi que M. Bertrand Laforge pour toute son aide.

2 Introduction

2.1 SHIP - Search for Hidden Particles

Le proposition de l'expérience SHIP vise à explorer des domaines de la physique ayant été peu étudié par les expériences précédentes, notamment le domaine de la physique de haute intensité et basse énergie, permettant d'observer des événements rares de basse énergie. De la physique au-delà du modèle standard pourrait s'y trouver, et cette expérience servirait à tester de nombreux modèles ayant des secteurs cachés, notamment des modèles expliquant la matière noire par un neutrino lourd 'stérile' de Majorana le Heavy Neutral Lepton (HNL) créé à partir de la collision des protons avec une cible.[1][4]

Afin de faire cela le projet consisterait à faire impacter un faisceau de protons de 400 GeV provenant du SPS sur une cible fixe lourde, un *beam dump* visant à ralentir les particules chargées tel que les pions et les kaons afin que leur désintégration produisant des neutrinos soit isotrope, et donc non focalisée sur le détecteur, diminuant ainsi le bruit de fond neutrino, l'idée de rajouter un champ magnétique au beam dump est en étude actuellement.[2][3] Ces impacts créeraient potentiellement différentes particules au-delà du modèle standard. Ces particules interagissant très faiblement et ayant de longues durées de vie, l'idée du projet est de mettre une chambre de désintégration sous vide (ou potentiellement remplie d'hélium) après le dump et un dispositif permettant au préalable d'éliminer le plus de particules possibles, en particulier les muons.

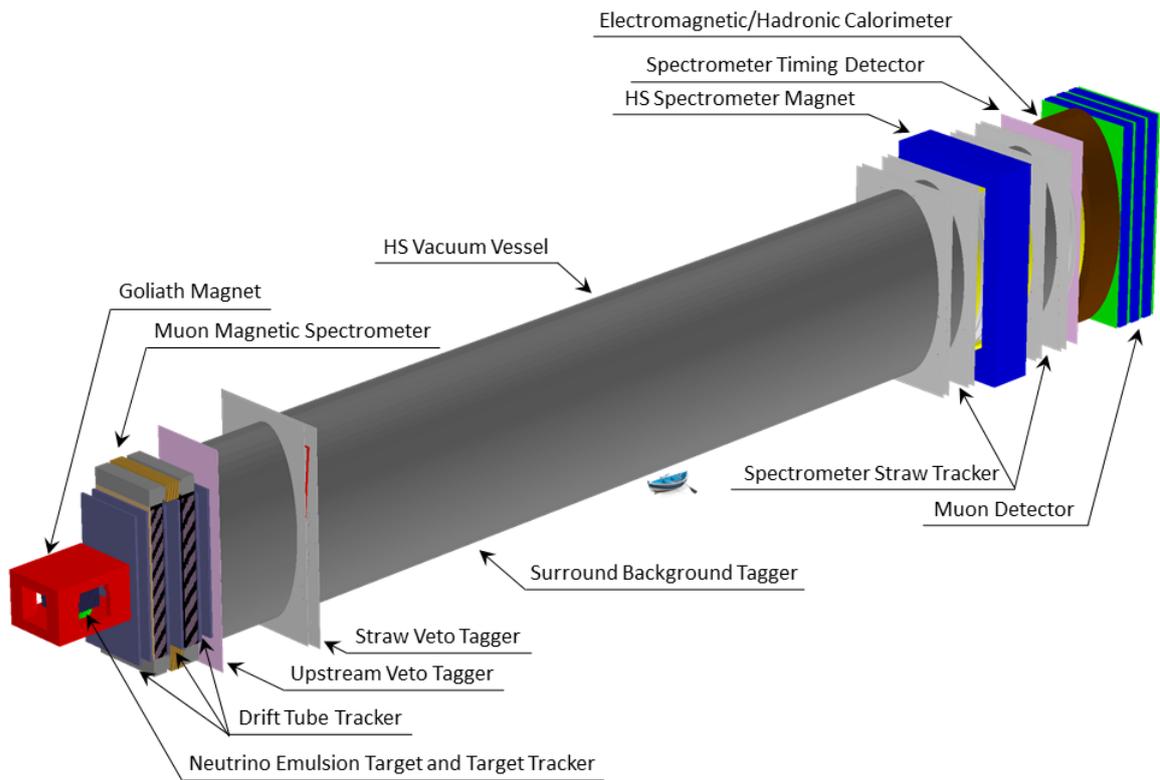


Figure 2: Schéma du Détecteur SHIP

Pour détecter et mesurer ces événements, le dispositif expérimental envisagé est composé de détecteurs à pailles (Straw Tracker) et de deux calorimètres (Électromagnétique et Hadronique), d'un détecteur de muons entre autres, comme dans le schéma précédent.

La conception du détecteur doit être optimisée afin d'obtenir des performances optimales tout en minimisant les coûts. C'est dans cette optique que le sujet de mon stage s'inscrit. Il s'agit d'étudier l'identification de particules, en particulier la séparation entre mesons π chargés, électrons et photons.

La méthode d'analyse du détecteur proposé se fait de manière entièrement numérique (en utilisant Pythia8, Geant4 et ROOT). D'abord, une simulation Monte Carlo des événements ainsi que de leurs interactions avec le détecteur et la cible est faite. Des données générées par cette simulation, un programme de reconstruction nous renvoie les mesures des détecteurs, ainsi que les 'objets' reconstruits tels les traces de particules chargées ou les 'clusters' d'énergie déposée dans le calorimètre électromagnétique. Enfin, les tâches du dernier programme consistent à analyser les données de la reconstruction et à identifier les événements et finalement à les comparer aux données de la simulation (la vérité Monte Carlo ou "*MC Truth*") afin de déterminer l'efficacité de la reconstruction. On parle donc d'optimisation des détecteurs afin de les rendre les plus effectifs possible afin d'identifier différentes particules, chargées ou non. L'optimisation est donc faite en comparant les simulations numériques de différents détecteurs possibles. Ceci nécessite donc des méthodes de reconstruction pour les différents canaux de désintégration qui nous permettent d'analyser différents détecteurs.

Pour les études d'identification, on choisit un canal comportant e^\pm , π^\pm , et γ dans l'état final, notamment le canal $e\rho$. Le travail effectué est l'étude de la réponse du dispositif expérimental à ces événements. Le résultat servira de base aux comparaisons entre le détecteur actuel et des dispositifs optimisés à concevoir.

$$HNL \longrightarrow e^\pm + \rho^\mp$$

fut choisi car il contient en fin de course 2 photons, et deux particules chargées différentes, un électron et un pi chargé.

La désintégration totale est:

$$\begin{aligned} HNL &\longrightarrow e^\pm + \rho^\mp \\ \rho^\mp &\longrightarrow \pi^0 + \pi^\mp \\ \pi^0 &\longrightarrow \gamma + \gamma \end{aligned}$$

on a donc:

$$HNL \longrightarrow e^\pm + \pi^\mp + \gamma + \gamma$$

L'objectif de mon stage est donc de mettre en place une méthode de reconstruction pour les événements de ce canal afin de pouvoir étudier les différents dispositifs possibles.

3 Méthode de Reconstruction

La reconstruction consiste à: • Identifier les évènements utiles avec une efficacité maximale • rejeter les évènements de bruits de fond avec un taux de rejet maximum

3.1 Coupures

Sur le nombre d'évènement créés, on va trier nos évènements par rapport à certains critères, notamment ceux de volume fiduciel, ceux sur la cinématique de l'état final, ainsi que des critères de veto.

Afin de reconstruire notre évènement, nous avons utilisé des méthodes déjà mises au point pour la reconstruction d'évènements $\mu\pi$

3.2 Canal $\mu\pi$

$$HNL \longrightarrow \mu^\pm + \pi^\mp$$

Ces évènements ne comprennent que deux particules chargées dans l'état final, les traces de ces particules après identification (détecteur de muon) ont été utilisées afin de reconstruire le vertex de la désintégration à l'endroit où les deux traces étaient les plus proches. A partir des traces et de l'identification des particules, on peut reconstruire les quadrivecteurs des deux particules et les ajouter pour récupérer le quadrivecteur et donc la masse de notre HNL ainsi que son paramètre d'impact par rapport au centre de la cible.

La méthode que nous avons réutilisée est celle qui nous permet à partir de deux traces chargées de recréer un vertex de désintégration.

3.3 Canal $e\rho$

$$HNL \longrightarrow e^\pm + \rho^\mp \longrightarrow e^\pm + \pi^\mp + (\pi^0 \longrightarrow \gamma + \gamma)$$

Pour cette désintégration, la tâche est plus complexe car nous devons nous préoccuper de l'identification des électrons à l'aide du calorimètre et nous avons deux particules neutres. Cette reconstruction se fait donc à partir des données du calorimètre électromagnétique, ainsi que des deux traces chargées reconstruites à partir des détecteurs à pailles. Les interactions des particules dans le calorimètre électromagnétique sont détectées dans plusieurs cellules de mesure qui constituent des amas ou *clusters*. Leurs caractéristiques diffèrent selon que la particule est neutre ou chargée.

3.3.1 Identification des particules chargées

Nous allons identifier l'électron et le pion en faisant usage de la propriété du pion d'être un MIP (Minimum Ionizing Particle), c'est à dire que le pion contrairement aux électrons et aux photons ne produit pas de gerbe dans le calorimètre et ne perd donc que peu de son énergie. D'abord, nous allons associer nos clusters (dans le calorimètre) à nos traces en prenant la trace la plus proche de notre cluster et en imposant que cette distance soit inférieure à 15 cm, valeur qui, comme on le voit sur la figure 1, est supérieure à 3σ (résolution spatiale). Nous ne prenons pas de distance de coupure inférieure pour maximiser les associations et parce que la multiplicité des traces est petite. Il y a cependant une incertitude sur l'extrapolation de la trace chargée au plan du calorimètre car nous prenons la face avant du calorimètre pour l'extrapolation.

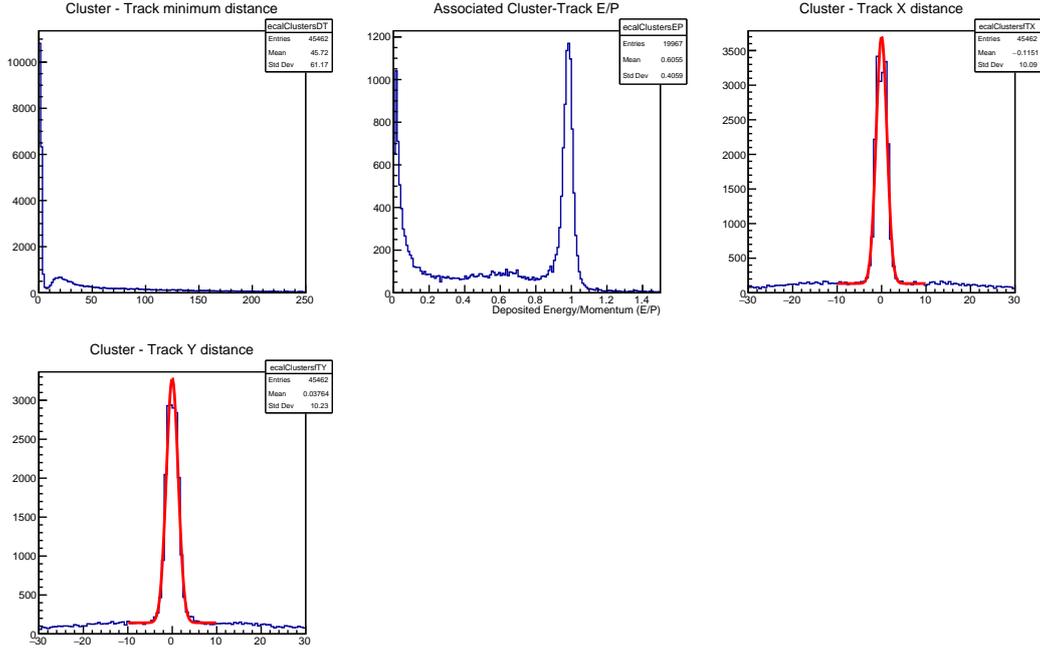


Figure 3: *Propriétés des Couple Trace-Cluster Reconstit*

Ensuite nous regardons l'énergie déposée dans le calorimètre pour notre cluster divisée par la norme de la quantité de mouvements de la trace associée, (E/P). Les électrons et les pions ayant des distributions de E/P largement différentes, comme on peut le voir sur la figure 2, on peut donc les identifier grâce à ceci en posant que si $E/P > 0.9$ alors le couple Trace-Cluster est associé à un électron, et à un pion chargé sinon. On définit ainsi les candidats électrons et les candidats pions chargés.

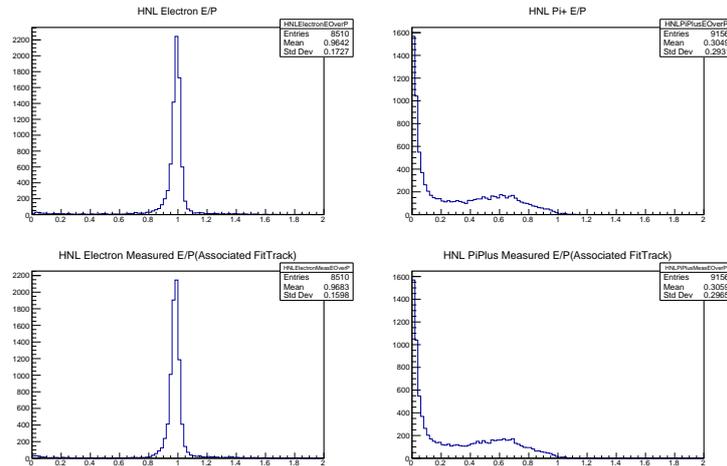


Figure 4: *Comparaison des distributions reconstruites de E/P pour les deux particules aux distributions de la simulation*

3.3.2 Vertex de désintégration et Pion neutre

Nous avons maintenant nos candidats électrons et pions chargés, nous pouvons donc reconstruire le vertex de désintégration comme pour l'événement $\mu\pi$. Il nous reste maintenant à reconstituer les particules neutres. A partir des clusters non associés à des traces, nous allons donc créer un quadrivecteur quantité de mouvement en posant que la quantité de mouvement d'un photon est :

$$\vec{P} = \|E\| \cdot \vec{u}_{dir}$$

et en posant que le vecteur de direction unitaire :

$$\vec{u}_{dir} = \frac{\vec{Cluster}_{Pos} - \vec{Vertex}_{Pos}}{\|\vec{Cluster}_{Pos} - \vec{Vertex}_{Pos}\|}$$

Pour toutes les paires de clusters non associés considérés comme des photons nous calculons la masse invariante à partir la somme des deux quadrivecteurs. Les paires ayant une masse invariante proche du π^0 (à mieux que 3σ en Δm) sont gardées en mémoire comme 'Candidats π^0 ', les photons utilisés sont retirés de notre liste de photons et on répète ceci jusqu'à avoir une liste de candidats π^0 .

3.3.3 Reconstruction ρ et HNL

Nous avons maintenant reconstruit toutes les particules de la désintégration étudiée. En sommant les quadrivecteurs de nos deux pions, nous obtenons le quadrivecteur de notre particule ρ et donc sa masse et en sommant tous les quadrivecteurs, nous obtenons le quadrivecteur de notre particule HNL et donc sa masse, et son paramètre d'impact par rapport au centre de la cible (cf. figure 5 & 6).

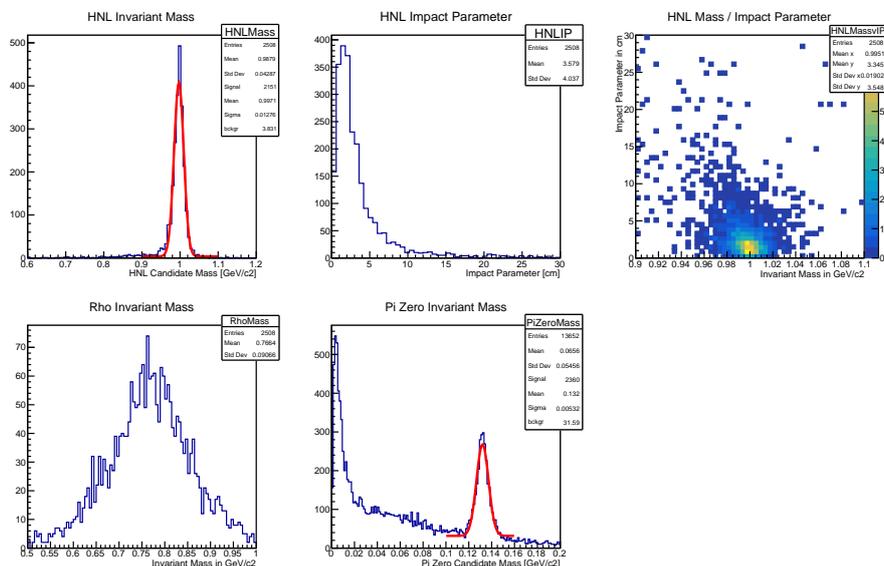


Figure 5: Propriétés des HNL, ρ et π^0 Reconstituit

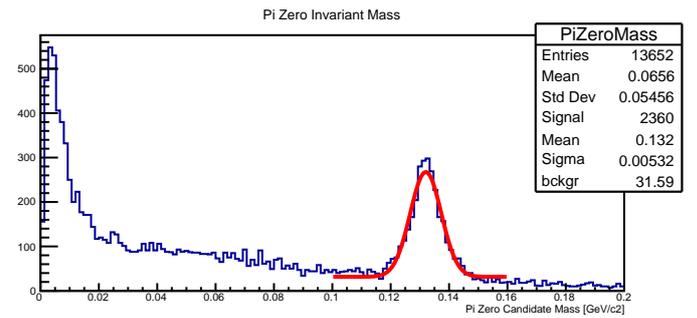
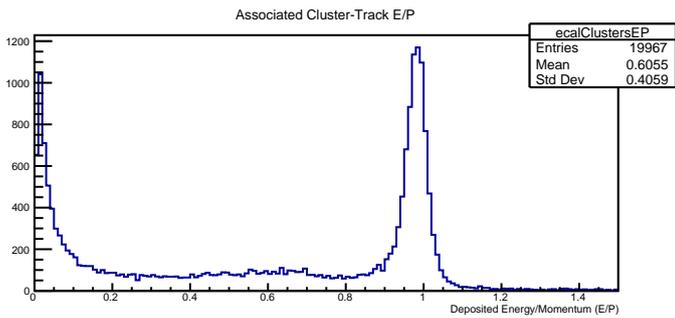
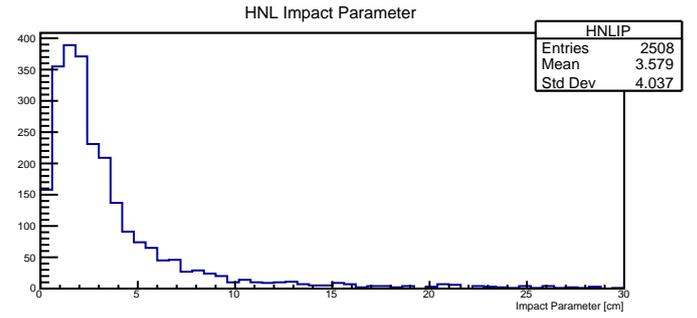
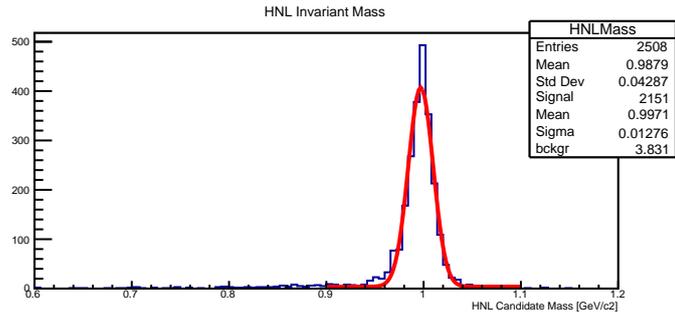
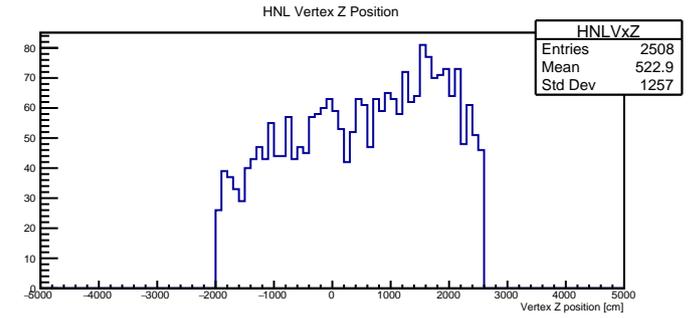
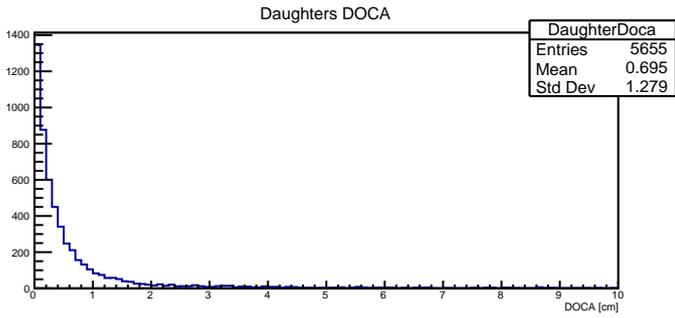
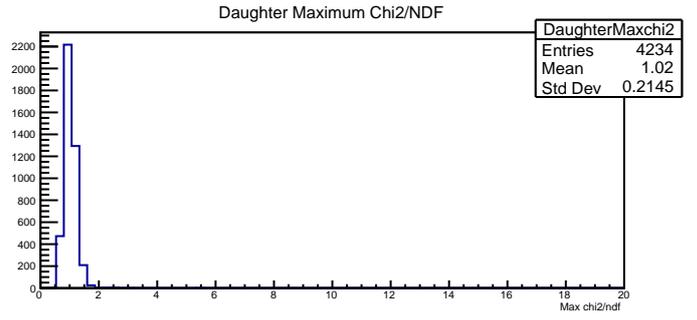
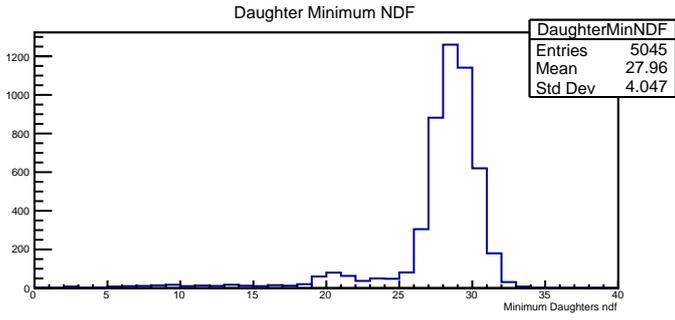


Figure 6: Propriétés du HNL

4 Coupure spécifique

A travers la reconstruction, nous faisons de nombreuses coupures sur la qualité de l' événement et sur des critères de sélections.

Coupures pour 50000 événements $e\rho$	
Contient un HNL:	50000
Trace ≥ 2 :	8248
Cluster ≥ 4 :	5888
Événements Reconstructibles:	3487
Vertex et deux Candidats Photons:	5322
Vertex \subset Volume de désintégration:	4880
Traces \subset Volume Fiduciel:	4880
Degrés de liberté des traces ≥ 25 :	4436
Distance d'approche minimale (DOCA) $< 1\text{cm}$:	4178
$\chi^2/\text{NDoF} < 5$:	4177
$P_{\text{ParticulesChargées}} \geq 1 \text{ GeV}$:	4102
Vertex Associé aux Particules chargées:	3691
$N_{e^-} > 0$:	3620
$N_{\pi^-} > 0$:	3175
$N_{\pi^0} > 0$:	2500
Paramètre d'impact < 10 :	2275
Veto déclenché:	2186
Événement == Monte Carlo:	2001

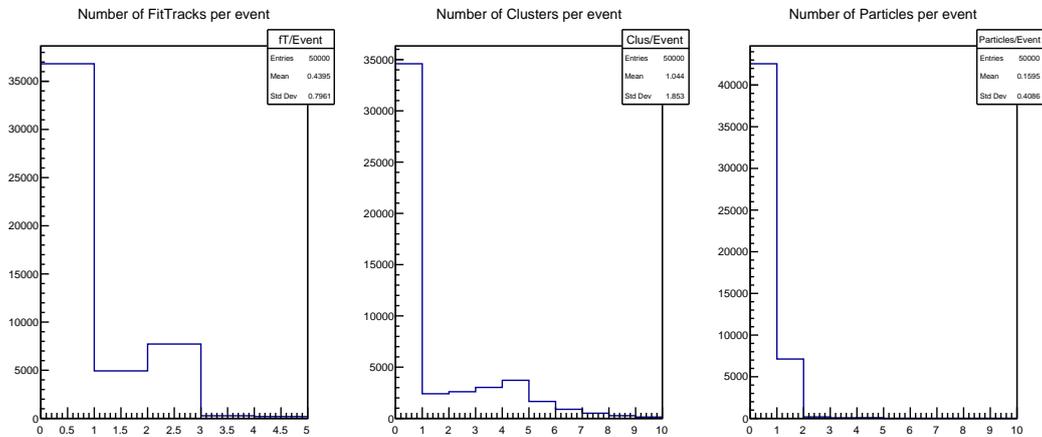


Figure 7: Propriétés des événements (ici Particle représente une paire de traces chargées)

5 Code

5.1 Philosophie du Code

La méthode de reconstitution comporte trois grandes étapes :

1. Association des traces et des clusters • Boucle sur les Clusters du Calorimètre électromagnétique dont on retient les mesures de la position et de l'énergie perdue (déposée) par la particule E.

- Boucle interne sur les traces chargées fittées à partir des impacts sur les chambres à pailles du spectromètre :

- Détermination de la position extrapolée de la trace dans le plan du calorimètre et ainsi de la distance Cluster-Trace (Objet: TrackClus), on trouve pour chaque cluster la trace avec laquelle la distance est minimale d.

- Les associations pour lesquelles d est suffisamment petite ($d < 15\text{cm}$) sont rangées dans des Objets ChargedCluster que l'on trie selon le rapport E/P où E est l'énergie du cluster et P la quantité de mouvement mesurée par les détecteurs à paille. On a vu plus haut comment cette quantité permet de discriminer les électrons des MIP. Le 'ChargedCluster' est gardé en mémoire comme Candidat tron (liste: ElectronCandidate) si $E > P$ dépasse 0.9 ou comme Candidat Pion (liste: PiPCandidate) sinon.
- Les clusters n'ayant pas été associés à des traces ($d > 15\text{ cm}$) sont considérés comme des clusters possibles pour les photons émis par la désintégration et sont gardés en mémoire.

2. Boucles sur les Couples Candidats Électron, Candidats Pi Chargés

- Ici nous réutilisons le code écrit pour les événements $\mu\pi$ où les couples de traces chargées, 'Particules', sont utilisés pour déterminer les vertex de désintégration. Comme le π^0 et le ρ ont des durées de vie extrêmement courtes nous faisons l'approximation que les vertex des désintégrations successives sont confondus.

- On cherche la 'Particule' (Objet copié du code $\mu\pi$) associée à nos deux ElectronCandidate et PiPCandidate et on récupère le vertex de désintégration ainsi que le quadrivecteur quantité de mouvement de la 'Particule', le tout dans un Objet ElPiPCandidate.

- Boucles sur les Clusters non associés :

- On suppose que l'énergie totale du photon a été absorbée par le calorimètre et on utilise cette énergie afin de calculer, à partir du vertex, de cette énergie et de la position du cluster, le quadrivecteur P du candidat photon.

- Pour les couples de photons on somme les quadrivecteurs reconstruits et on garde ceux ayant une masse invariante à moins de 3σ de la masse du π^0 (objet:PiZeroCandidate) . On garde ensuite en mémoire autant d'Objets ElPiPPiZero contenant les objets ElPiPCandidate et PiZeroCandidate, qu'il y a d'objets PiZeroCandidate.

3. Dans nos ElPiPPiZero, on va reconstruire le HNL et calculer le quadrivecteur P ainsi que paramètre d'impact du HNL par rapport au centre de la cible.

5.1.1 Objet Python

Afin de simplifier l'écriture du programme, nous avons créé des objets Python.

1. TrackClus

L'objet TrackClus permet de calculer la distance Trace-Cluster, afin de faire l'association.

Attributs:

aClus == Objet Cluster provenant de EcalReconstructed

cl == indice de l'objet aClus

fT == Objet FitTrack provenant de sTree.FitTracks

tr == indice de l'objet fT

Fonctions:

-distance()

Créer les attributs et les ajouter aux histogrammes:

distx == distance Trace-Cluster sur l'axe des x

disty == distance Trace-Cluster sur l'axe des y

dist == distance Trace-Cluster

2. ChargedCluster

Couple Trace-Cluster associé, permet de différencier Pion et Électron grâce à son attribut E/P.

Attributs:

aClus == Objet Cluster provenant de EcalReconstructed

cl == indice de l'objet aClus

fT == Objet FitTrack provenant de sTree.FitTracks

tr == indice de l'objet fT

distx == distance Trace-Cluster sur l'axe des x

disty == distance Trace-Cluster sur l'axe des y

dist == distance Trace-Cluster

EOverP == Énergie déposée dans le Cluster diviser par quantité de mouvement calculer dans FitTracks.getFittedState()

HNLMother == Vérité Monte Carlo, la particule a telle pour mère un HNL

HNLGMother == Vérité Monte Carlo, la particule a telle pour grand-mère un HNL

Fonctions:

-CheckMC()

Vérifie les vérité Monte Carlo:

MCHNLEl == Vrai si la particule Monte Carlo associée à l'objet est un électron provenant du HNL

MCHNLPiP == Vrai si la particule Monte Carlo associée à l'objet est un Pion Chargé provenant du HNL

-FourMom(m):

Calcule le quadrivecteur P en imposant la masse de la particule == m et en utilisant la quantité de mouvement du FitTrack.

P == Objet TLorentzVector() quantité de mouvement

-MCGraphs():

Créer des graphiques si les particules sont associées aux électron et pion HNL

3. EIPiPCandidate

Couple Electron-Pion qui sera associé à la 'Particule' provenant de $\mu\pi$, et utilisé pour calculer le vertex.

Attributs:

PiP == Objet ChargedCluster associer au Pion Chargé

El == Objet ChargedCluster associer à L'Électron

Fonctions:

-ParticleAssoc(Particle)

Associe la particule et calcule le vertex, quadrivecteur quantité de mouvement et le doca:

doca == 'distance of closest approach' des deux traces

vertex == Objet TVector3() associer à la position du vertex

P == Objet TLorentzVector() quantité de mouvement

4. GammaCandidate

Candidats Photons

Attributs:

aClus == Objet Cluster provenant de EcalReconstructed

cl == indice de l'objet aClus

MCHNLGamma == vérité Monte Carlo, Vrai si le gamma provient du HNL

Fonctions:

-VertexAssoc(vertex)

Associer le cluster à un vertex et crée le quadrivecteur correspondant à son énergie dans la direction Vertex \rightarrow Cluster

vertex == vertex associé

dir == TVector3() unitaire dans la direction Vertex \rightarrow Cluster

P == Objet TLorentzVector() quantité de mouvement reconstruite à partir de aClus.RecoE() et dir

5. PiZeroCandidate

Candidat π^0 créé à partir de couple de candidats photons.

Attributs:

Gamma1 == Objet GammaCandidate

Gamma2 == Objet GammaCandidate

vertex == Vertex associé aux Gamma

Indices == liste des indices des Gamma1 et Gamma2 dans la liste 'GammaCandidates'

P == Objet TLorentzVector() quantité de mouvement = Gamma1.P + Gamma2.P

MCHNLPiZero == verite Monte Carlo, Vrai si le PiZero est associé au HNL

Fonctions:

-Pi0Fit()

Créer des attributs des quantités de mouvements refittée pour rapprocher la masse du PiZero:

FitP == Objet TLorentzVector() quantité de mouvement refittée

FitChi2 == Chi2 du Fit

6. EIPiPPiZero

Objet utilisé pour recréer le HNL et contenant le couple candidats pion chargé et électrons, et le candidat PiZero.

Attributs:

EIPiP == Objet EIPiPCandidate associé au candidats Particules Chargées

PiZero == Objet PiZeroCandidate associé au candidat PiZero

P == Objet TLorentzVector() quantité de mouvement associée au candidat HNL

RhoP == Objet TLorentzVector() quantité de mouvement associée au candidat Rhô

HNLMass == Masse invariante du candidat HNL

RhoMass == Masse invariant du candidat Rhô

IP == paramètre d'impact du candidat HNL

MCHNL == vérité Monte Carlo, Vrai si le HNL est associé au HNL MonteCarlo

Fonctions:

-HNLStats()

Fais les statistiques des HNLs/Rhô

-VertexResol()

Fait les graphiques de la résolution des vertex

6 Conclusion

Les détecteurs du dispositif expérimental étant en phase d'optimisation, la reconstruction des événements simulés du canal $e\rho$ entre autres permettra de déterminer l'efficacité des différents modèles de Calorimètre électromagnétique. La prochaine étape que nous avons effleurée à la fin du stage est celle de l'effet du bruit de fond neutrino sur nos reconstructions, nous avons lancé la reconstruction sur un petit nombre d'événements 10000 et nous n'avons vu aucune reconstruction erronée. Dans le futur, il faudra analyser ceci sur un nombre plus élevé d'événements de bruit de fond.

References

- [1] SHiP Collaboration. A facility to Search for Hidden Particles at the CERN SPS: the SHiP physics case. arXiv:1504.04855 [hep-ph], 2015.
- [2] SHiP Collaboration. A facility to Search for Hidden Particles (SHiP) at the CERN SPS. arXiv:1504.04956 [physics.ins-det], 2015.
- [3] SHiP Collaboration. A Facility to Search for Hidden Particles (SHiP) at the CERN SPS, Addendum to Technical Proposal. CERN-SPSC-2015-040 / SPSC-P-350-ADD-2, 2015.
- [4] E. Graverini. SHiP sensitivity to Heavy Neutral Leptons. CERN-SHiP-NOTE-2016-CDS March 13, 2016.