

A.a. 2014-2015

Simulazione di uno spettrometro per il decadimento $K_S^0 \rightarrow \pi^+ \pi^-$

Studente:

Federico MASSA

Docente:

Sergio GIUDICI

1. Introduzione

Il presente progetto consiste nella simulazione di uno spettrometro per la ricostruzione dell'impulso di un fascio puro di K^0 di impulso $100 \pm 5 \text{ GeV}/c$, distribuito gaussianamente.

!!!!!!!!!!!!NOTA: NON CONSIDERIAMO L'IMPURITÀ DI KL, MA QUANTO AVREBBE INFLUITO ALLA DISTANZA Z1 SCELTA?!!!!!!!!!!!!!!!!!!!!!! MAGARI SCRIVERE NELLE CONCLUSIONI, COME LIMITI DELLA SIMULAZIONE

Scopo del progetto è quello di misurare la risoluzione del suddetto spettrometro al variare della tecnica di ricostruzione utilizzata e del numero di rivelatori utilizzati nell'apparato sperimentale (!!!!!!!!!!!!!!!!E DEL RAGGIO INTERNO?!!!!!!!!!!!!!!). Ai fini della simulazione è stato considerato solamente il decadimento del $K_S^0 \rightarrow \pi^+\pi^-$.

Il codice utilizzato è stato scritto interamente in C++, con l'ausilio delle librerie ROOT del CERN per la visualizzazione e il salvataggio in memoria dei dati.

2. Apparato sperimentale simulato

L'apparato simulato, nella sua versione normale, appare come in Fig. 1, 2.

Ogni piano di rivelatore è modellizzato da una corona circolare di raggio interno R_{int} e raggio esterno R_{ext} , con risoluzione fissa sugli assi x e y di 1 mm ¹. Sebbene quest'ultimo sia importante a livello pratico nel disegno del rivelatore, ai fini di questa simulazione questo sarà sempre considerabile infinito, poiché scelto in modo tale da contenere l'interesse dei decadimenti. La scelta del raggio interno è invece soggetta a differenti necessità, che saranno discusse più avanti (Sez. 5.1!!!!!!!!!!!!!!). Il campo magnetico è situato in una zona di 0.35 m al centro tra i piani 2 e 3, che distano tra loro 3 m e vale $B = 1 \text{ T}$. Il p_{kick} corrispondente vale quindi circa $p_k = qB\Delta L = 105 \text{ MeV}/c$. La modellizzazione dell'effetto di un magnete come un kick effettuato nel punto medio tra i rivelatori sarà giustificata in sez.4 !!!!!!!!!!! In Tab.1 sono riassunte le caratteristiche scelte per l'apparato e la lunghezza media di decadimento, con $\langle \lambda \rangle = \frac{\langle p \rangle}{M_K} c\tau$, $\tau = 8.954 \cdot 10^{-11} \text{ s}$.

¹Come si vedrà in Sez.5, la risposta del rivelatore è assunta gaussiana e la risoluzione indicata è dunque uguale alla deviazione standard di questa distribuzione.

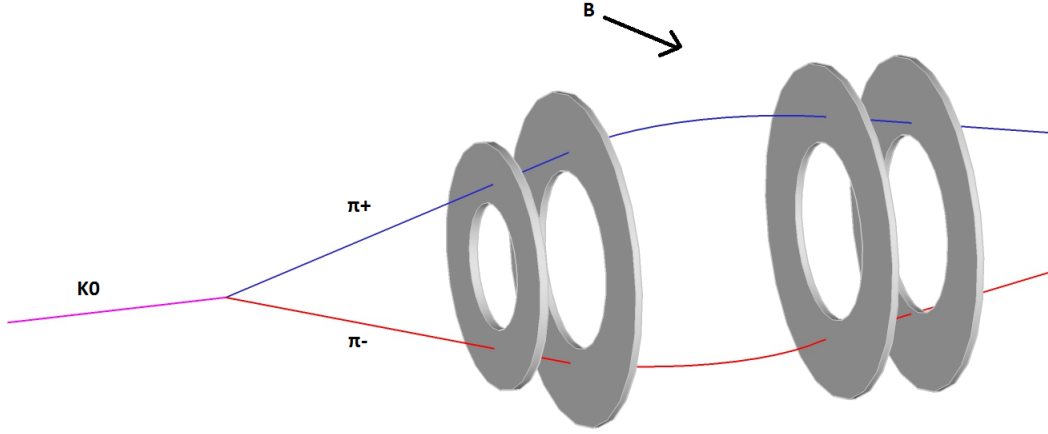


Figure 1: Riproduzione 3D dell'apparato sperimentale: con \vec{B} è indicato il vettore del campo magnetico, nella direzione positiva delle x .

$\langle \lambda \rangle (\text{m})$	5.4
$z_1 (\text{m})$	50
$z_2 (\text{m})$	60
$z_3 (\text{m})$	63
$z_4 (\text{m})$	73
$R_{\text{int}} (\text{m})$	0.1
$p_{\text{K}} (\text{MeV}/c)$	105
$B (\text{T})$	1
$\Delta L (\text{m})$	0.35
$\sigma_x (\text{m})$	0.001

Table 1: Tabella riassuntiva delle caratteristiche dell'apparato sperimentale.

3. Struttura del codice

Il codice, scritto in C++, presenta una struttura modulare rappresentata in un diagramma in Fig.3.

Esso si basa sull'uso di una classe di ROOT, *TNtupleD*, alle cui istanze ci riferiremo con il nome *ntuple*, che permette di salvare in un file tutte le caratteristiche desiderate per un evento in forma di variabili in doppia precisione.

Il codice consiste di tre moduli:

- *Generation*: grazie all'uso di un generatore di numeri pseudocasuali (*TRandom3*), si serve di una classe chiamata *KGen* per generare le caratteristiche di un singolo evento.

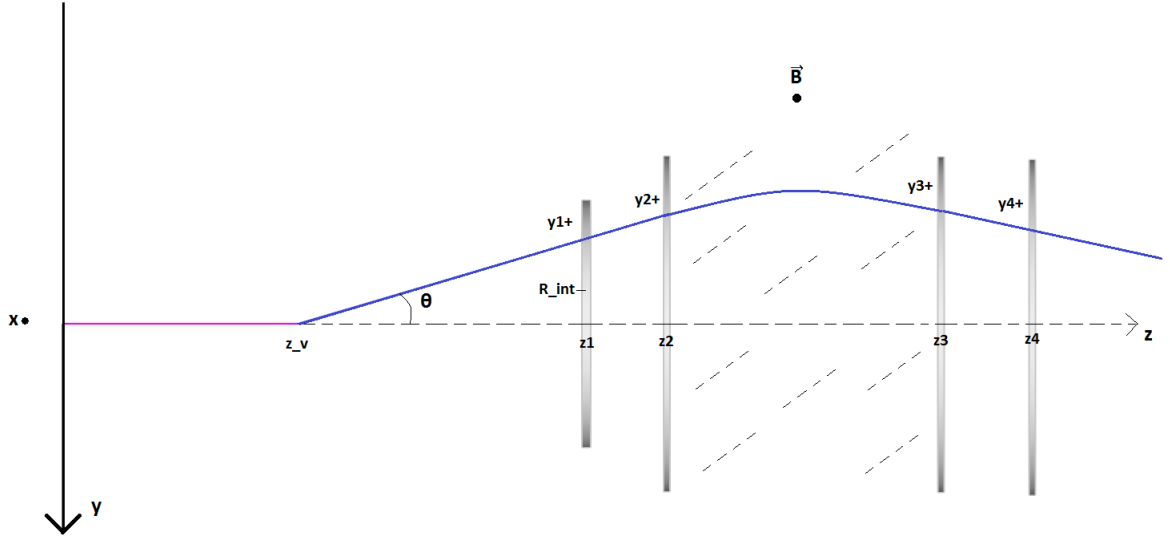


Figure 2: Sezione dell'apparato sperimentale: in viola lungo l'asse z , la traccia del K^0 genitore, in blu e rosso rispettivamente le tracce delle particelle decadute π^+ e π^- . Il campo magnetico, nella direzione positiva delle x , è uscente perpendicolarmente dal piano della figura. Con z_i sono indicate le coordinate z dei quattro piani di rivelatore. Con R_{int} , invece, il raggio interno del rivelatore più vicino.

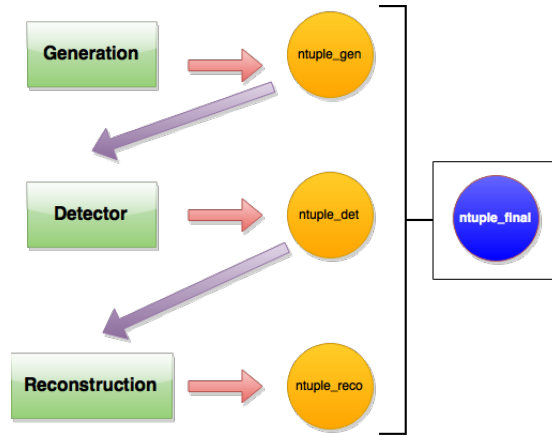


Figure 3: Diagramma di flusso del funzionamento del codice. Ogni modulo riceve in input e/o produce in output una *ntuple*.

Iterando questo procedimento, vengono generati tutti gli eventi e salvati in una ntupla chiamata *ntuple_gen*.

- *Detector*: utilizzando come input *ntuple_gen*, si occupa di trovare i punti di passaggio dei prodotti di decadimento nei rivelatori e di effettuare uno *smeering* gaussiano per simulare la risposta del rivelatore. Si serve della classe *B_event* per simulare l'effetto del campo magnetico. Gli hit trovati vengono salvati nella ntupla *ntuple_det*.
- *Reconstruction*: utilizzando come input *ntuple_det*, che contiene solo le informazioni fruibili allo sperimentatore, si occupa di effettuare un fit cinematico con vincoli e parametri non misurati per trovare delle misure migliorate, che vengono scritte in *ntuple_reco*.

Le informazioni provenienti dalle tre ntuple vengono infine unite in una sola, chiamata *ntuple_final*, che permette di studiare la risoluzione dello spettrometro confrontando la misura migliorata dell'impulso con quella generata. Nelle prossime sezioni analizzeremo nel dettaglio i tre moduli.

4. Generazione dell'evento

Il modulo *Generation* si basa sulla classe *KGen*, che genera i singoli eventi tramite il metodo *Generate* e su un main, che richiama un numero predeterminato di volte questo metodo e ne salva i risultati volta per volta su *ntuple_gen*. Il main ha anche a disposizione l'opzione per generare eventi con punto di decadimento costante. Da questo momento le variabili contrassegnate da * si intendono nel sistema del centro di massa, mentre quelle senza in quello del laboratorio. Nel decadimento preso in esame la particella madre, il K^0 , è scalare, per cui non vi può essere alcuna direzione privilegiata nel decadimento nel sistema del centro di massa. Questo comporta che i prodotti di decadimento presentino una distribuzione angolare piatta in $\cos\theta^*$ e ϕ^* , dove il sistema di coordinate scelto è quello polare con polo nel punto di decadimento. θ^* è dunque calcolato come $\cos^{-1}(\cos\theta^*)_{gen}$ ed è dunque compreso tra 0 e π , mentre ϕ è generato uniformemente tra 0 e 2π . Una volta generata la direzione del primo prodotto, il secondo è automaticamente determinato dalla legge di conservazione dell'impulso. Risulta:

$$\theta_2^* = \pi - \theta_1^*$$

$$\phi_2^* = \phi_1^* + \pi$$

dove a ϕ_2^* si sottrae 2π se $\phi_2^* > 2\pi$.

Essendo un decadimento a due corpi identici in massa, inoltre, l'energia dei prodotti ha un valore ben preciso ed uguale per i due prodotti, così come quello degli impulsi:

$$E_1^* = E_2^* = \frac{M_K}{2}$$

$$p_i^* = \sqrt{E_i^{2*} - M_\pi^2}$$

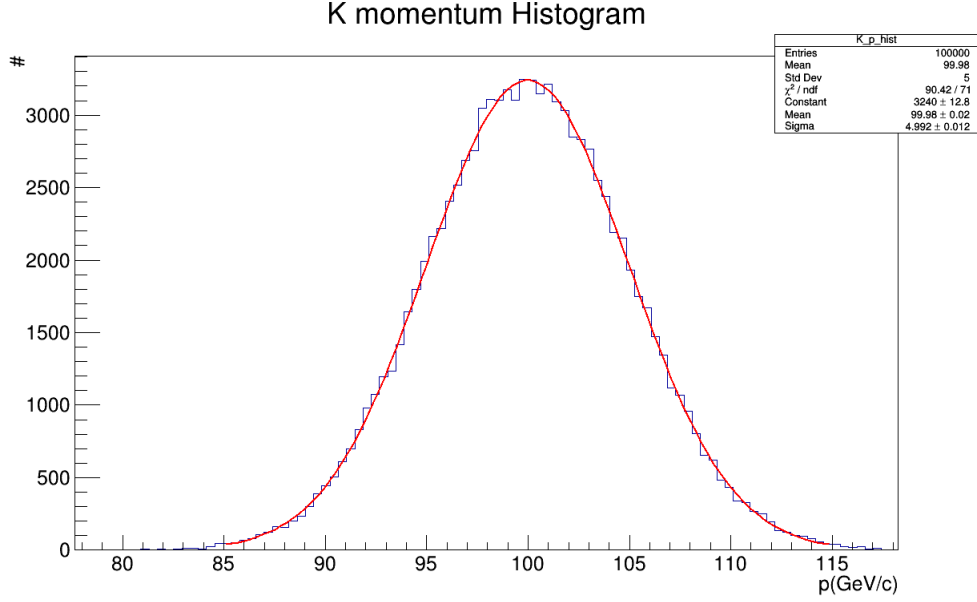


Figure 4: Distribuzione dell'impulso generato del K^0 e fit gaussiano.

Si procede dunque a generare l'impulso del K secondo una distribuzione gaussiana con media $100 \text{ GeV}/c$ e deviazione standard $5 \text{ GeV}/c$, e si calcola la sua β come

$$\beta = p_K / E_K \approx 1 - \frac{1}{2} \frac{M_K^2}{p_K^2}$$

Con questo valore si possono trovare gli impulsi nel sistema del laboratorio con un boost lungo z con $\beta_{boost} = -\beta$. Per quanto riguarda gli angoli, sono calcolati tramite:

$$\cos \theta_i = \frac{p_{zi}}{p_i}$$

$$\phi_i = \text{atan2}(p_{ix}, p_{iy}) = \phi_i^*$$

Per quanto riguarda la z di decadimento, viene generata con distribuzione esponenziale e lunghezza di decadimento dipendente dall'energia del K^0 generato:

$$z_v \sim \text{Exp}(z_v | \lambda = \beta \gamma c \tau)$$

Queste funzionalità sono accessibili dal main tramite i metodi pubblici *GetTheta*, *GetP*, ... della classe *KGen*.

In Fig. 4, 5, 6, 7, 8, 9, 10 sono rappresentate alcune distribuzioni degli eventi generati.

Le curve ricalcano il comportamento atteso e i fit restituiscono un valore di χ^2 accettabile con una fiducia del 1%. I parametri ottenuti dai fit sono compatibili con quelli generati.

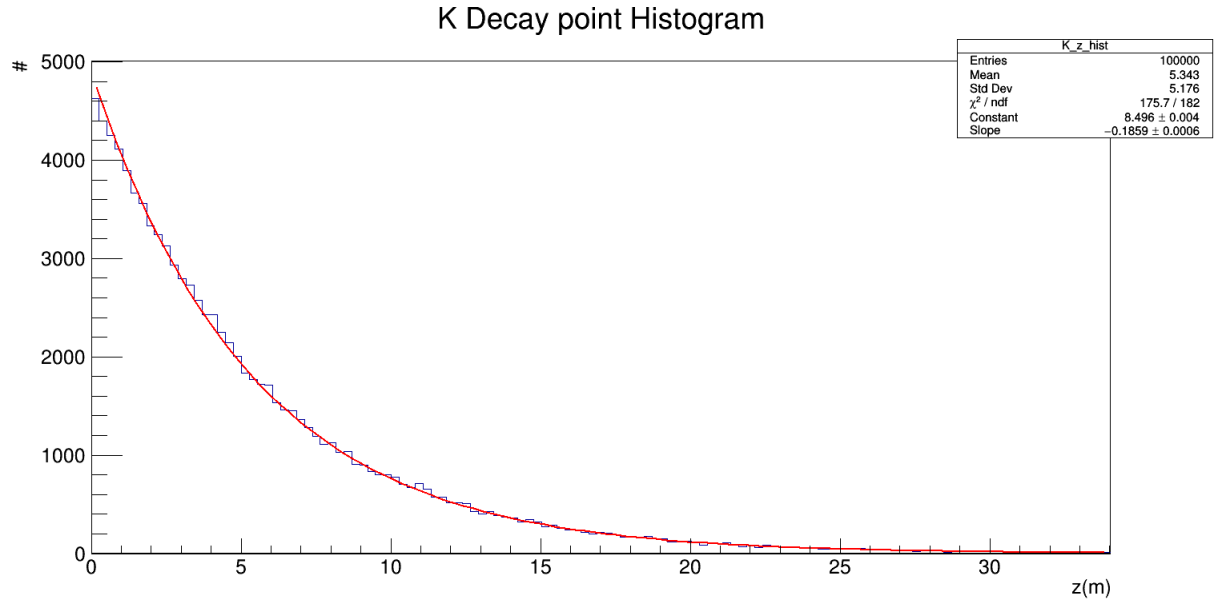


Figure 5: Distribuzione della z di decadimento e fit esponenziale. In realtà la distribuzione vera si discosta da un'esponenziale a causa della distribuzione dell'impulso del K^0 .

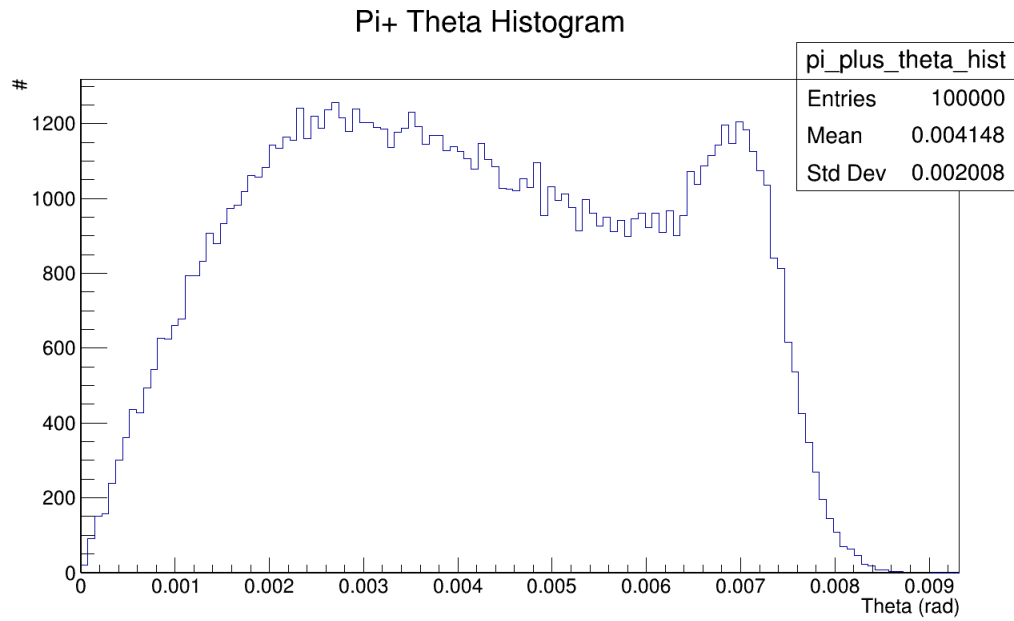


Figure 6: Distribuzione dell'angolo θ nel caso del π^+ . La distribuzione è limitata superiormente ad un valore inferiore a π a causa del boost. Nel caso del π^- la distribuzione è analoga.

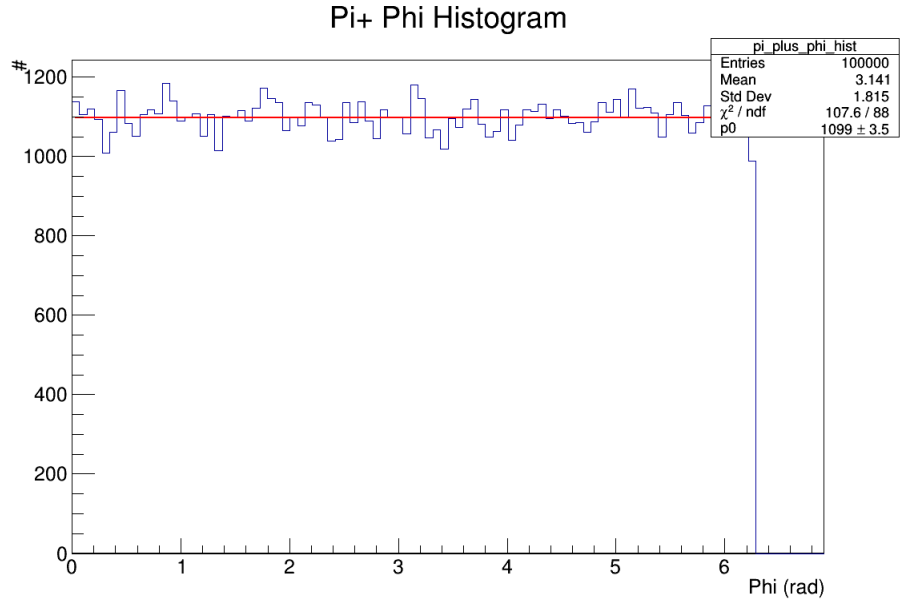


Figure 7: Distribuzione dell'angolo π nel caso del π^+ . Essendo quest'angolo invariato dopo il boost, la distribuzione risulta uniforme tra 0 e 2π . In figura è mostrato il fit della funzione costante. Nel caso del π^- la distribuzione è analoga.

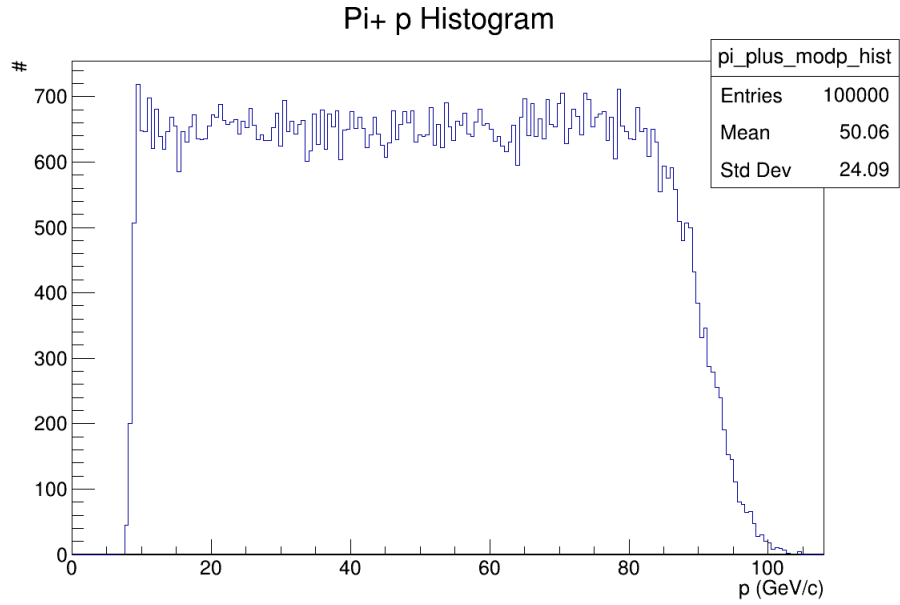


Figure 8: Distribuzione del modulo dell'impulso del π^+ . La distribuzione è limitata inferiormente e superiormente per ragioni cinematiche. Nel caso del π^- la distribuzione è analoga.

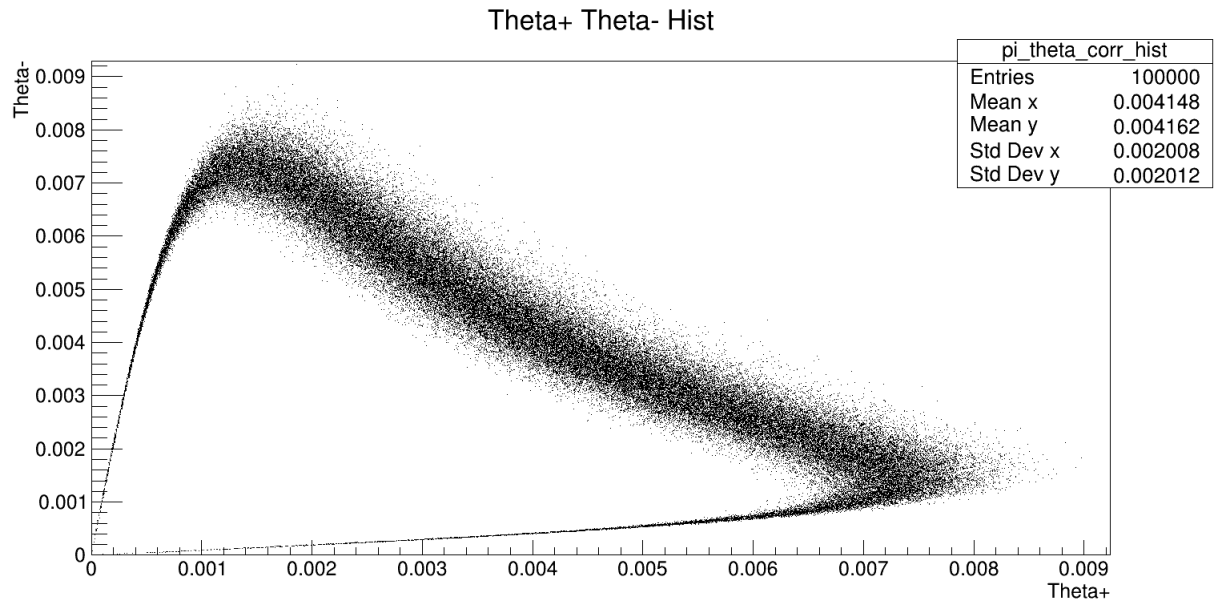


Figure 9: Correlazione tra gli angoli θ delle due particelle. Si nota che quando uno è massimo l'altro è piccolo e viceversa.

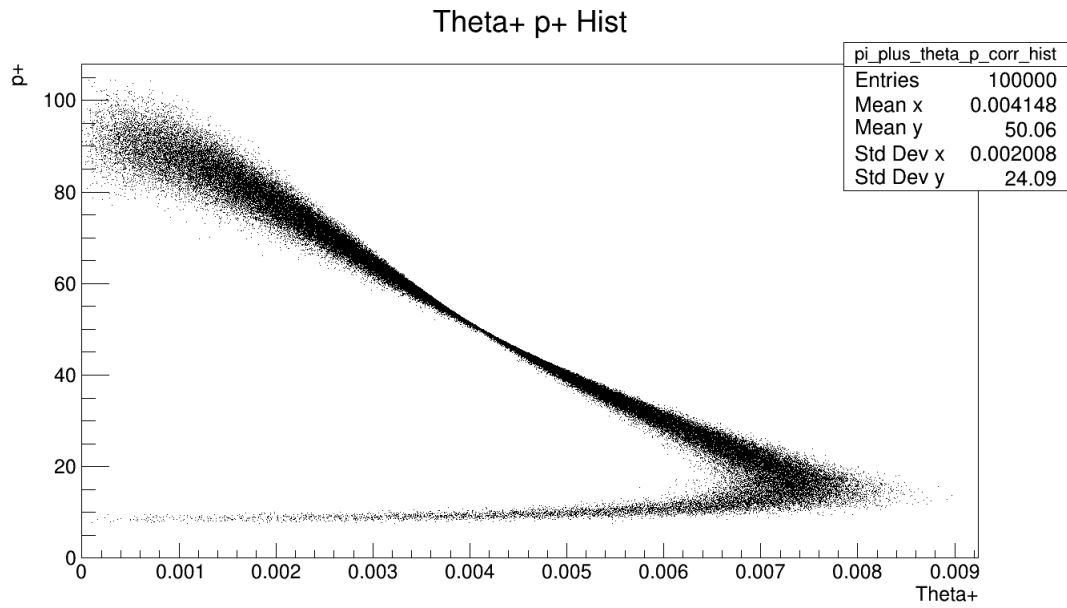


Figure 10: Correlazione tra l'angolo θ del π^+ e il suo impulso.

5. Simulazione della risposta dei rivelatori

Il secondo modulo (*Detector*) calcola i punti di passaggio della particella nel rivelatore, escludendo dal salvataggio in memoria quelli che non superano il taglio geometrico imposto dalla struttura del rivelatore. Questo modulo fa uso della classe *B_event* per simulare l'effetto del campo magnetico e calcolare i punti di passaggio nei piani di rivelatore 3 e 4.

5.1 Raggio interno

La scelta del raggio interno si basa su tre aspetti principali:

- *Larghezza del fascio*;
- *Risoluzione e algoritmi di ricostruzione*;
- *Accettanza*.

Per quanto riguarda la **larghezza del fascio**, essa non è stata presa in considerazione nella simulazione. Ciò nonostante, affinché quest'ultima abbia validità, occorre specificare un raggio interno minimo ragionevole, che collochiamo sui 10 *cm*.

Il secondo aspetto riguarda la **risoluzione** e gli **algoritmi di ricostruzione**. Per quanto il fit cinematico utilizzato non faccia uso esclusivo della curvatura della traccia prima e dopo il campo magnetico, nel calcolo dei parametri e nella loro inizializzazione appaiono spesso al denominatore differenze tra le coordinate di due rivelatori. Questo si riflette su un peggioramento del calcolo dei parametri nei casi in cui la differenza delle coordinate sia prossima allo zero, ovvero quando l'angolo θ delle tracce è piccolo. Prevedere un raggio interno non nullo permette quindi di escludere dall'analisi eventi troppo piatti che comporterebbero un peggioramento complessivo della risoluzione dello spettrometro e/o un eventuale fallimento del fit per tali eventi.

Il raggio interno non può tuttavia approssimare indefinitamente il massimo teorico (quello per cui solo i decadimenti prodotti nell'origine possono essere rivelati) poiché l'**accettanza** del sistema tende a zero. Tipicamente infatti si può cercare un compromesso tra la risoluzione e il tempo di misura necessario per ricostruire i parametri dello spettro (μ e σ di una gaussiana in questo caso) con la precisione desiderata.

L'accettanza è definita come la probabilità che un evento venga accettato dal sistema rivelante, dato un avvenuto decadimento. In formule, nel nostro caso:

$$a = P(x_1^{2+} + y_1^{2+} > R_{int}^2 \cap x_1^{2-} + y_1^{2-} > R_{int}^2 \mid \text{avvenuto decadimento})$$

Questa quantità, nota la configurazione sperimentale, potrebbe essere calcolata teoricamente. Tuttavia, molto raramente questo è possibile e si preferisce misurarla dai dati simulati. Essendo il numero di eventi accettati r distribuiti come una binomiale con numero di prove uguali a N_{gen} (numero di eventi generati) e probabilità di successo a , un possibile stimatore per l'accettanza è:

$$\hat{a} = \frac{r}{N_{gen}}$$

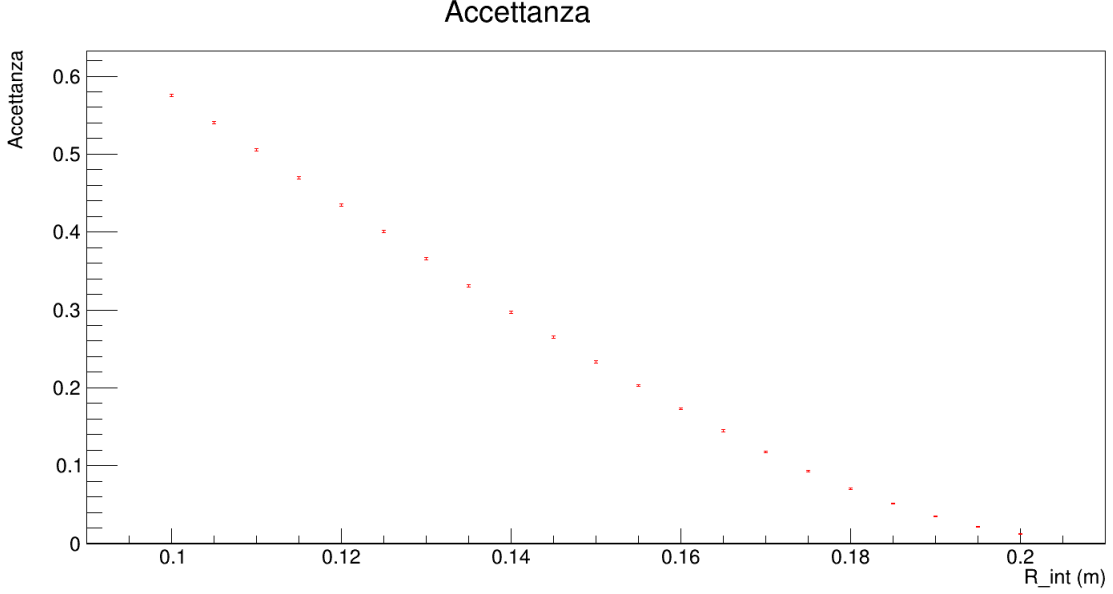


Figure 11: Accettanza del sistema in funzione del raggio interno del rivelatore 1.

$$Var(\hat{a}) = \frac{a(1-a)}{N_{gen}} \approx \frac{\hat{a}(1-\hat{a})}{N_{gen}}$$

che risulta essere *unbiased*, di massima verosimiglianza ed efficiente. La sovrastante approssimazione è buona fino a che il numero di eventi accettati è significativamente diverso da zero (o da N_{gen}).

In Fig.11 è mostrata la curva di accettanza per il rivelatore, fissate le dimensioni longitudinali a quelle descritte in Sez.2 e al variare del raggio interno.

5.2 Coordinate degli hit: linearità e approssimazione di kick

Noti gli angoli e il punto di decadimento è possibile calcolare facilmente i punti di arrivo delle tracce sui rivelatori 1 e 2. Con riferimento a fig.2, si ricava:

$$x_{1,2}^{\pm} = (z_{1,2} - z_v) \tan(\theta^{\pm}) \cos(\phi^{\pm}) \quad (1)$$

$$y_{1,2}^{\pm} = (z_{1,2} - z_v) \tan(\theta^{\pm}) \sin(\phi^{\pm}) \quad (2)$$

Per la simulazione dell'effetto del campo magnetico si utilizza un'approssimazione che risulta buona quando $p_k = qB\Delta z \ll p$ e quando $\cos(\theta) \approx 1$. Nel nostro caso (riferimento a fig.8)

$$|p_k| = 105 \text{ MeV}/c \ll p_{min} \approx 15 \text{ GeV}/c$$

In queste condizioni, per una particella incidente a $\theta = 0$, vale

$$(\vec{p}_{after} - \vec{p}_{before})^2 = (\Delta p)^2 = (F\Delta t)^2 \approx (qB\Delta z)^2$$

dove *before* e *after* si riferiscono al valore assunto prima e dopo il campo magnetico. La precedente relazione, sviluppata usando le approssimazioni porta a

$$\theta = \frac{p_K}{p}$$

se θ non è zero, ma è piccolo, è sufficiente sostituire $\theta \rightarrow \Delta\theta$.

Usando infine $\Delta\theta \approx \Delta \tan \theta$ si ottiene:

$$\frac{\partial y}{\partial z}|_{after} = \frac{\partial y}{\partial z}|_{before} + \text{sign}(q) \frac{|p_k|}{p}$$

ovvero

$$\frac{y_4^\pm - y_3^\pm}{z_4 - z_3} = \frac{y_2^\pm - y_1^\pm}{z_2 - z_1} + \text{sign}(q) \frac{|p_k|}{p} \quad (3)$$

mentre le coordinate x rimangono lineari entro queste approssimazioni:

$$x_{3,4}^\pm = (z_{3,4} - z_v) \tan(\theta^\pm) \cos(\phi^\pm) \quad (4)$$

In queste condizioni l'effetto del campo magnetico è quello di far variare bruscamente la pendenza della traccia con un *kick*, che supponiamo dato nel punto medio tra i rivelatori, dove si colloca il magnete. Imponendo anche che le tracce entrante e uscente si incontrino nel punto medio si ricava:

$$y_3^\pm = y_1^\pm + (y_2^\pm - y_1^\pm) \frac{z_3 - z_1}{z_2 - z_1} + \frac{z_3 - z_2}{2} \text{sign}(q) \frac{|p_k|}{p} \quad (5)$$

5.2.1 Applicazione dello smearing del rivelatore

Una volta calcolati i punti di passaggio veri della particella, è stato effettuato uno smearing gaussiano di deviazione standard σ_x attorno al punto vero (tab.1, sez.2).

I risultati sono stati salvati sulla ntupla *ntuple_{det}*, che contiene le informazioni fruibili allo sperimentatore.

6. Ricostruzione dell'evento

La ricostruzione dell'evento può avvenire su vari livelli di sofisticatezza. Una ricostruzione molto rozza della z di decadimento, per esempio, può essere quella ottenuta invertendo la relazione:

$$\frac{x_4^\pm}{z_4 - z_v} = \frac{x_1^\pm}{z_1 - z_v}$$

dove sono stati scelti i piani 1 e 4 per ridurre l'errore.

Una ricostruzione un po' più sofisticata è rappresentata dalla *Closest Distance of Approach*, ovvero la z corrispondente al punto di minima distanza tra la congiungente degli hit dei rivelatori 1 e 2 e l'asse di decadimento. Si ricava in questo caso:

$$z_{CDA}^\pm = z_1 - (z_2 - z_1) \frac{x_1^\pm(x_2^\pm - x_1^\pm) + y_1^\pm(y_2^\pm - y_1^\pm)}{(x_2^\pm - x_1^\pm)^2 - (y_2^\pm - y_1^\pm)^2}$$

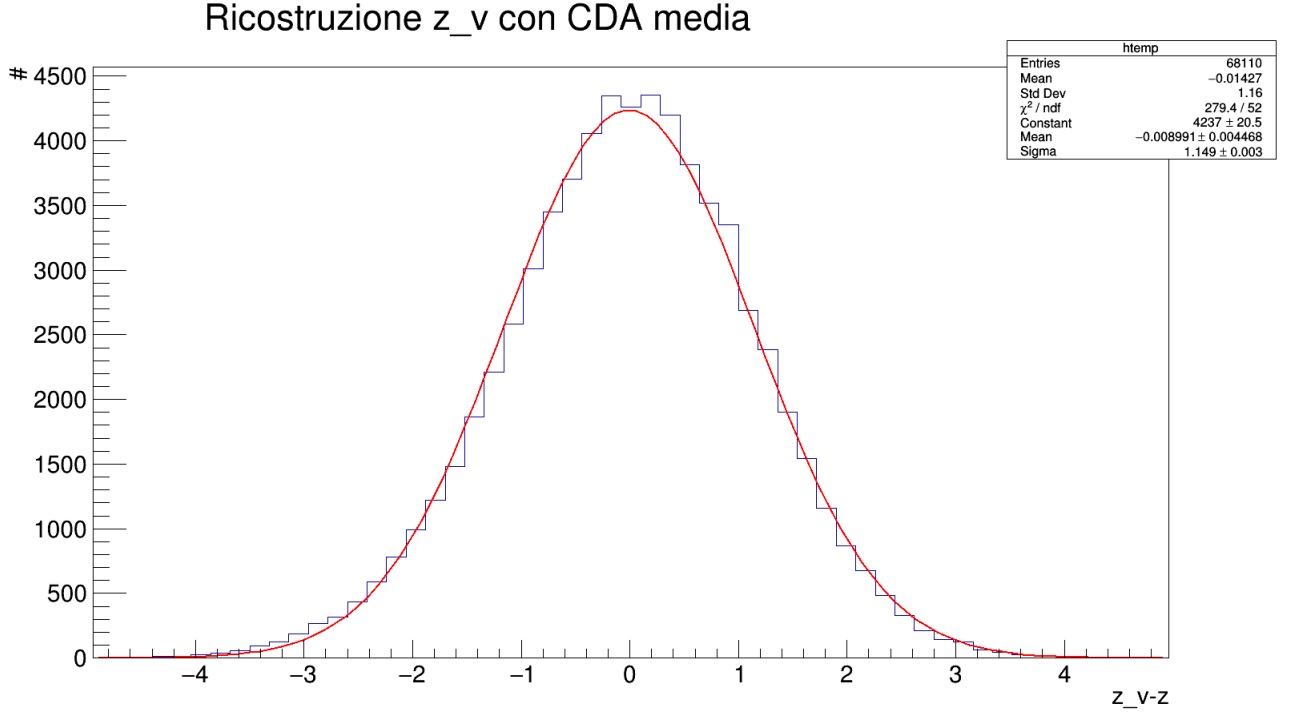


Figure 12: Ricostruzione di z_v con la media delle CDA e fit gaussiano.

Una ricostruzione migliorata di un fattore $\sqrt{2}$ si può ottenere facilmente semplicemente usando come z_v la media aritmetica dei valori ottenuti per le due tracce.

Per quanto riguarda l'impulso, una prima ricostruzione può essere ottenuta invertendo l'eq.3 per trovare p . In fig.12,13, 14 sono mostrate le distribuzioni delle variabili ricostruite meno il loro valor vero. Il valore del χ^2 mostra che la distribuzione scelta per il fit non è compatibile, ma questo non è un problema in quanto non c'è nessun motivo statistico per cui la variabile sia distribuita gaussianamente e il fit serve solo ad estrarre dei parametri sensati. Si noti che la ricostruzione dell'impulso del K^0 con la sola curvatura ha portato ad una risoluzione in impulso di circa $9 \text{ GeV}/c$, che nel nostro caso non sono sufficienti a descrivere la forma dello spettro che ha una larghezza pari a circa la metà della risoluzione trovata.

6.1 La classe IterKinFitP

Il modo scelto per ricostruire in modo più efficace l'impulso si basa su un metodo iterativo di minimizzazione di un X^2 vincolato.

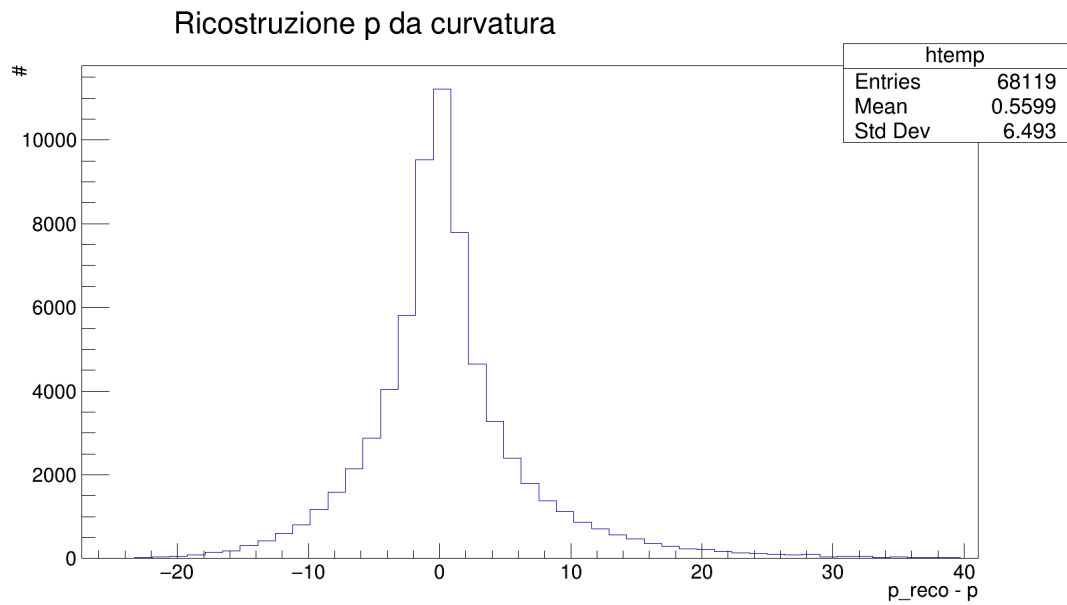


Figure 13: Ricostruzione dell'impulso della singola particella con la curvatura.

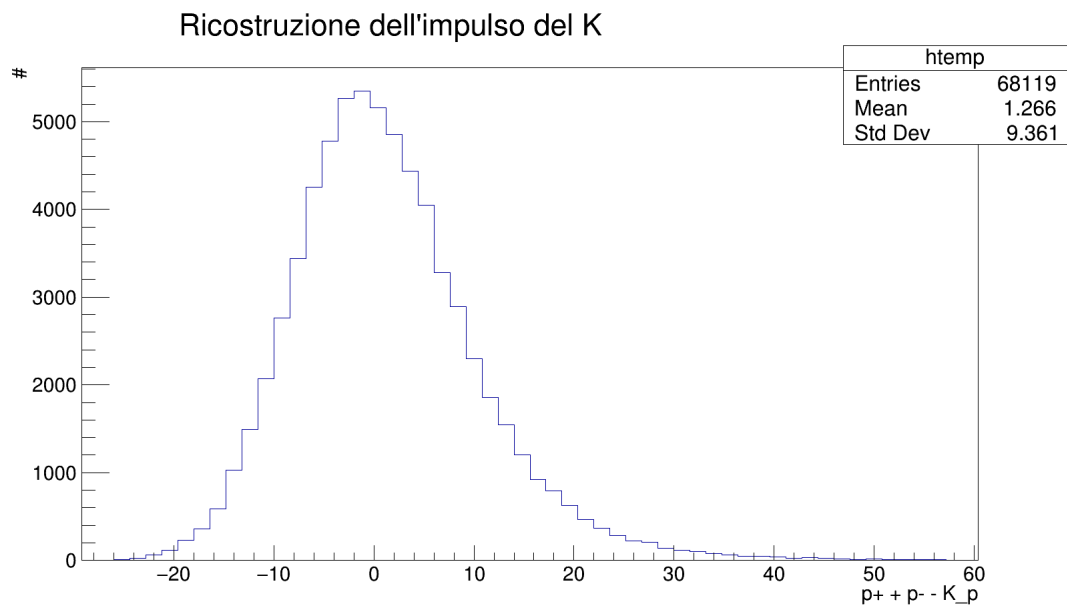


Figure 14: Ricostruzione dell'impulso del K con la curvatura.

6.2 Identificazione della carica

6.3 Problema combinatoriale

6.4 Set di vincoli scelti

6.5 Misura della risoluzione con quattro rivelatori

6.6 Misura della risoluzione con tre rivelatori

7. Conclusioni

8. Appendice: valori numerici utilizzati

- $c = 299792458 \text{ m/s}$
- $M_K = 497.614 \text{ MeV}/c^2$
- $M_\pi = 139.57018 \text{ MeV}/c^2$
- $\tau = 8.954 \cdot 10^{-11} \text{ s}$