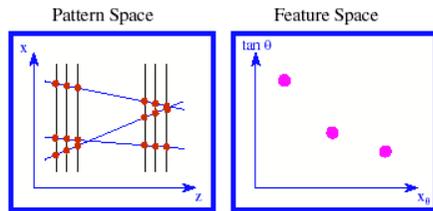


Pattern recognition

Le particelle in un evento posseggono caratteristiche che possono essere descritte da un insieme dei parametri (*features*).

Nel rivelatore si misurano dei segnali disposti in modo ordinato (*pattern*).



Lo scopo della *pattern recognition* è quello di identificare le forme e misurarle per risalire ai parametri (*features*).

PGI2006 lect_11 1

Pattern recognition è usata *on-line* principalmente per ricerca di tracce e configurazione di sciami.

Tipicamente si affronta un problema algoritmico la cui complessità cresce col numero N di "informazioni" come $N!$

(Molto di quanto detto per le tracce si applica anche ad altre ricerche di configurazioni)

Metodi globali

Tutte le informazioni entrano nell'algoritmo

- La decisione è presa quando tutta l'informazione è conosciuta
- Può applicarsi direttamente ai dati senza manipolazioni intermedie
- Ha bisogno di un buon modello per le tracce

Algoritmi globali

- Trasformazioni conformi, per esempio $z' = 1/z$
- Reti neurali
- Template matching <---> Lookup table
- Trasformata di Hough ==> metodo degli istogrammi

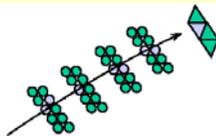
PGI2006 lect_11 2

Example of pattern recognition by look-up tables

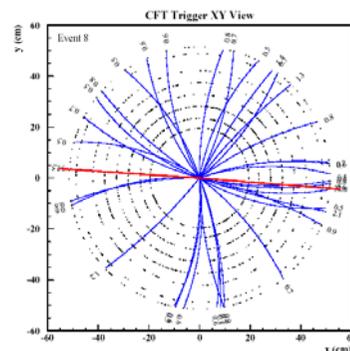


Central Tracking Trigger

- Feed all axial fibers plus preshower into gate arrays
- Analyze 2-layer doublet patterns
- A track if a fiber combination is consistent with $P_T > (1.5, 3, 6, 8)$ GeV ; $Q = \pm 1$
- Hundreds of patterns at 1.5 GeV
- Tag categories (incl. CPS info): track, isolated track, electron, ...
- Send 64 physics bits to L1 Framework
 - from menu of combinations of:
 - track counts, p_T , tags, charge, occupancy, φ sector, total P_T ...



Trigger response for $Z \rightarrow ee$ with 4 min.bias



PGI2006 lect_11 3

DØ upgrade:

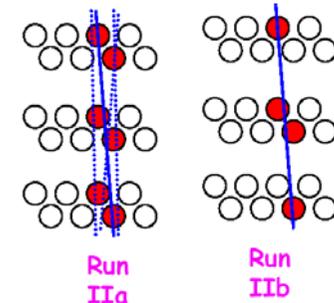
Run IIa ==> Run IIb

Migliorare la risoluzione:
aumentare il numero di combinazioni
passando da

- due fibre per doppio strato
- a
- una o due fibre per doppio strato.

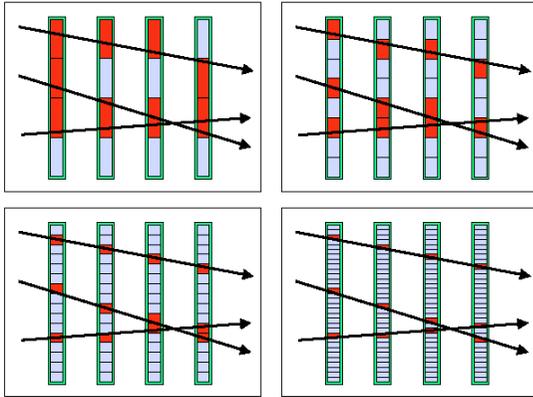
Richiede:

- gate arrays più grandi
- e
- processori più veloci



PGI2006 lect_11 4

Tree search in template matching

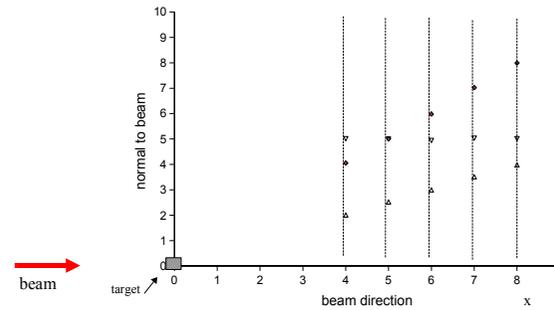


PGI2006 lect_11 5

Esempio elementare di trasformata di Hough

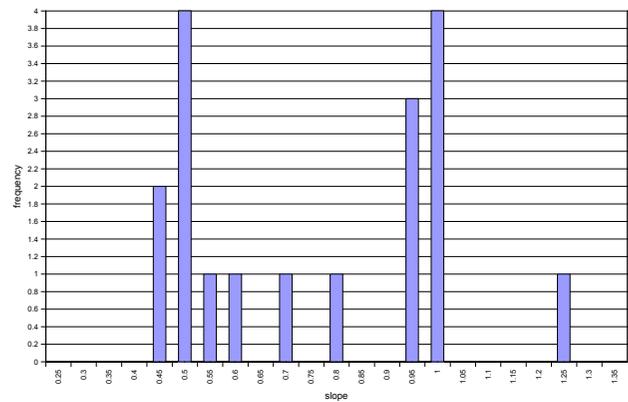
Nello spazio fisico a due dimensioni (x, y) il modello di traccia è una retta che passa per l'origine (*target*). Alcuni rivelatori misurano dei punti a valle. L'equazione della retta generica è $y = px$.

Se i punti sperimentali x_i, y_i appartengono ad una retta che passa per l'origine soddisfano all'equazione.



PGI2006 lect_11 6

Nello spazio dei parametri p , ad una sola dimensione, i valori sperimentali di p sono $p_i = y_i/x_i$. Essi si accumulano intorno ai valori di p che corrispondono a rette che passano per l'origine nello spazio fisico.



I2006 lect_11 7

L'equazione generale di una retta $y = px + q$ nel piano è determinata da due parametri p e q : si ha quindi un istogramma a due dimensioni. Per evitare valori divergenti di p per linee verticali, si preferisce rappresentare la retta in funzione della sua distanza r dall'origine e dell'angolo θ che la sua normale forma coll'asse delle ascisse:

$$x \cos \theta + y \sin \theta = r$$

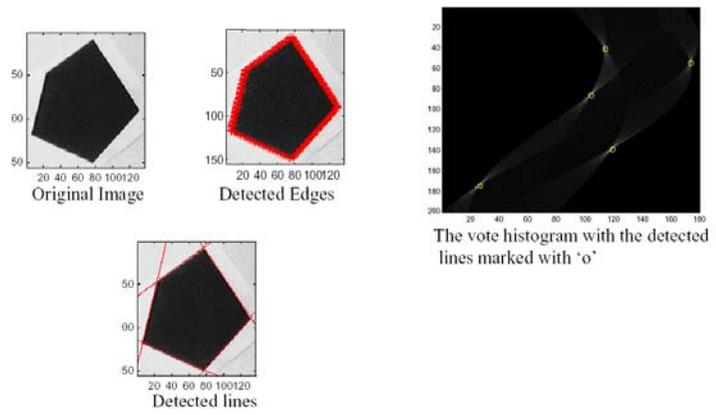
L'istogramma si costruisce ora nel piano dei parametri r e θ usando le coppie di coordinate misurate (x_i, y_i) .

La trasformata di Hough si applica a tracce elicoidali, come quelle in un solenoide ai *colliders*. Nella proiezione su un piano perpendicolare ai fasci queste tracce sono degli archi di cerchio che passano dal punto di interazione. Possono essere parametrizzate in coordinate polari ρ e φ in funzione della curvatura $1/R$ e di un angolo di emissione ψ misurato rispetto a una direzione fissa:

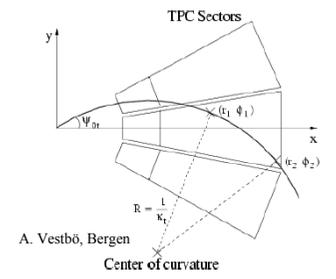
$$\rho = 2R \sin(\varphi - \psi)$$

L'istogramma si costruisce ora nel piano dei parametri $1/R$ e ψ usando le coppie di coordinate misurate (ρ_i, φ_i) .

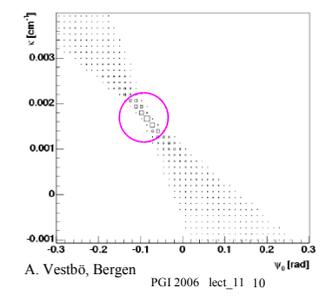
PGI2006 lect_11 8



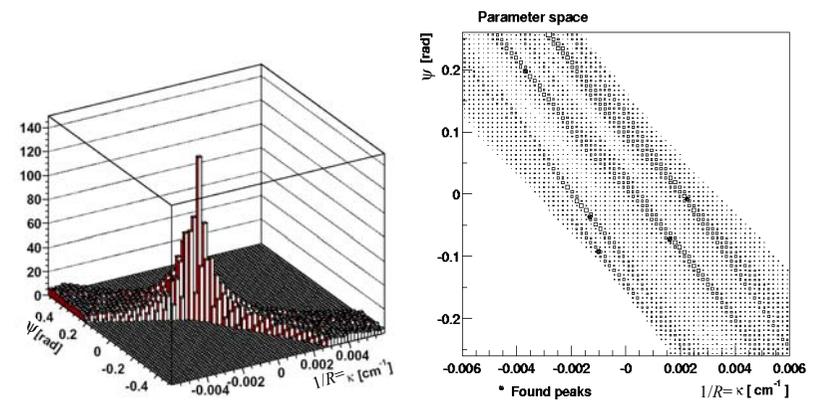
I parametri della traccia $1/R$ e ψ sono rappresentati in figura, ove appare anche la traccia proiettata su alcuni settori della TPC di ALICE.



I picchi dell'istogramma, costruito usando come informazione la carica sulle *pads* com'è misurata dall'ADC senza nessun trattamento per trovare punti (*clusters*), rappresentano tracce.

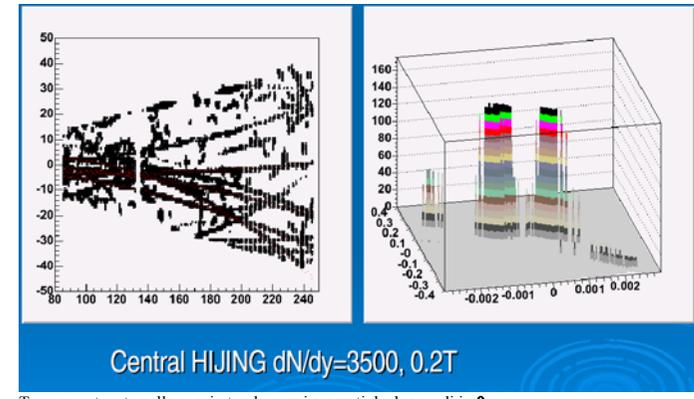


ALICE Simulation



Tracce contenute nello spazio tra due coni separati da due gradi in θ

ALICE Simulation



Tracce contenute nello spazio tra due coni separati da due gradi in θ

I picchi nell'istogramma sono difficili da trovare principalmente a causa della correlazione tra le variabili $1/R$ e ψ

Definiamo un cambiamento di variabili:

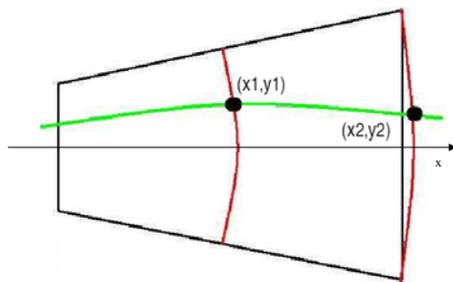
$$x' = \frac{x}{x^2 + y^2} \quad y' = \frac{y}{x^2 + y^2}$$

In questo riferimento i cerchi che passano per l'origine del piano (x, y) diventano rette.

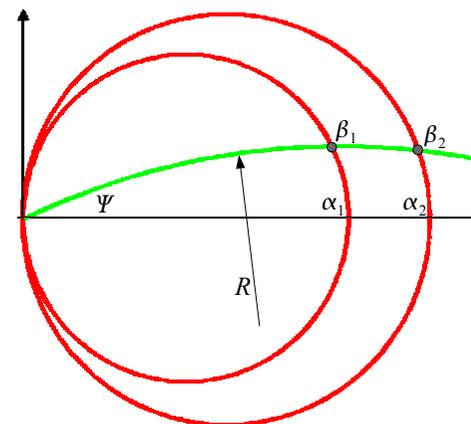
Prendiamo due curve **fisse** su un settore di TPC come in figura (rosse) definite da $x' = \cos t$.

Esse sono cerchi che passano per il punto di interazione, origine del piano (x, y) .

Una traccia che passa per l'origine del piano (x, y) è completamente identificata dalle sue **intersezioni** coi due cerchi.

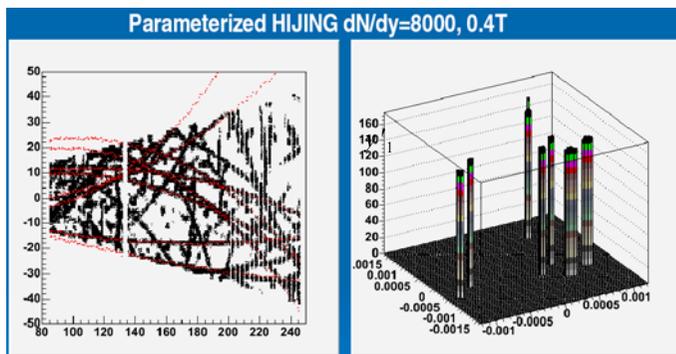


PGI2006 lect_11 13



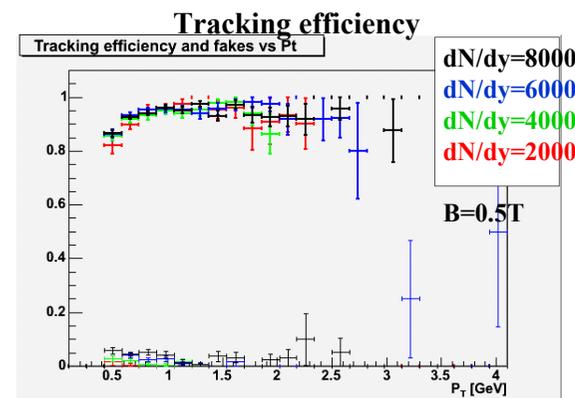
PGI2006 lect_11 14

L'equazione di una traccia qualsiasi è lineare nelle due variabili y'_1 e y'_2 ; se β_1 e β_2 sono i valori che y'_1 e y'_2 assumono quando la traccia incrocia i due cerchi, β_1 e β_2 possono essere usati come parametri (invece di $1/R$ e ψ) nell'istogramma della trasformata di Hough. Grazie alla trasformazione di coordinate i picchi sono ora ben separati.



Tracce contenute nello spazio tra due coni separati da due gradi in θ

PGI2006 lect_11 15



PGI2006 lect_11 16

Metodi locali

- Scelgono un candidato di traccia alla volta e predicono punti successivi che dovrebbero appartenere alla stessa traccia, estrapolando punto dopo punto
- La decisione è presa passo a passo, quando solo una parte dell'informazione è disponibile o usata
- Il risultato può dipendere dall'ordine in cui i dati sono impiegati
- Ha bisogno di solito di essere innescato (*seeding*)
- Non c'è bisogno di un modello generale per le tracce

Algoritmi locali

Track road

Track element

Track following ==> *Kalmán filter*

PGI2006 lect_11 17

Le caratteristiche della traccia che attraversa la superficie k sono determinate da un vettore \vec{p} a 5 componenti, per esempio due componenti di posizione e tre di momento.

Usando il linguaggio dei filtri digitali lineari, nei quali tanto i dati d'ingresso x_i quanto i risultati y_i sono sequenze di scalari

$$y_k = K_k x_k + b_k y_{k-1}$$

dove y_{k-1} descrive la traccia sulla superficie $k-1$

x_k sono le misure sulla superficie k

Ci sono due azioni distinte:

$$y_k = K_k (x_k - H_k y_{k-1}) + (b_k + K_k H_k) y_{k-1}$$

filtra **estrapolazione**

In realtà le misure sono un vettore \vec{x}_k a 2 dimensioni, che rappresenta il punto sulla k -esima superficie e i risultati sono un vettore \vec{y}_k a 5 dimensioni che definisce la traccia sulla k -esima superficie.

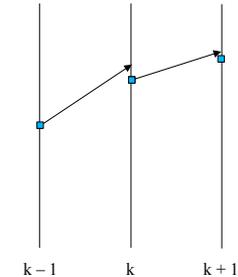
PGI2006 lect_11 19

Filtro di Kalmán

Nel metodo di *track following* si estrapolano i parametri della traccia da una "superficie" alla seguente, dove si troverà un punto, se esiste, a una certa distanza (errore) dall'estrapolazione.

Nel filtro di Kalmán i criteri di accettazione del nuovo punto e l'uso che se ne fa sono più elaborati. L'estrapolazione tiene conto di *multiple scattering*, curvatura etc.

Il nuovo punto è accettato ed associato alla posizione del punto estrapolato usando un filtro digitale, le cui caratteristiche sono determinate col metodo dei minimi quadrati applicato ad una funzione che contiene tutta l'informazione disponibile su quella "superficie", compresi gli errori.



PGI2006 lect_11 18

Quindi tutte le "costanti" sono matrici o funzioni matriciali; inoltre la funzione di estrapolazione può non essere lineare:

$$\vec{p}_k = K_k (\vec{x}_k - H_k f_k(\vec{p}_{k-1})) + c_k f_k(\vec{p}_{k-1})$$

Il coefficiente del filtro contiene l'informazione relativa alla matrice degli errori del passo precedente $k-1$.

Vantaggi

- *track finding* e *track fitting* simultaneamente
- non si devono invertire grosse matrici
- connessione facile ai rivelatori adiacenti
- non è necessario un modello globale della traccia
- si può tener conto di processi aleatori

Svantaggi

- è necessario costruire dei "punti" (*cluster finding*)
- si deve partire da elementi di traccia noti (*seeds*)

PGI2006 lect_11 20

Per pattern recognition in generale:

R. Frühwirth, M. Regler, R. K. Bock, H. Grote and D. Notz,
Data Analysis Techniques for High-Energy Physics, 2nd ed., Cambridge
University Press 2000

R Mankel, Pattern recognition and event reconstruction in
particle physics experiments, Rep. Prog. Phys. 67 (2004) 553-622
http://arxiv.org/PS_cache/physics/pdf/0402/0402039.pdf

Karel Šafarik, Tracking at the LHC, CERN Academic Training
2000/01

<http://home.cern.ch/~karel/TRKLecture.ppt>

<http://www.wlap.org/cern/lectures/academ/2001/safarik/>

Per la trasformata di Hough (e tante altre cose):

R. K. Bock and W. Krischer:
<http://rkb.home.cern.ch/rkb/titleA.html>