

**GSI**

GSI-93-11  
REPORT  
FEBRUAR 1993  
ISSN 0171-4546

**TRACKING MIT DER HOUGH-TRANSFORMATION FÜR DIE  
ZENTRALE DRIFTKAMMER DES GSI- $4\pi$ -EXPERIMENTS**

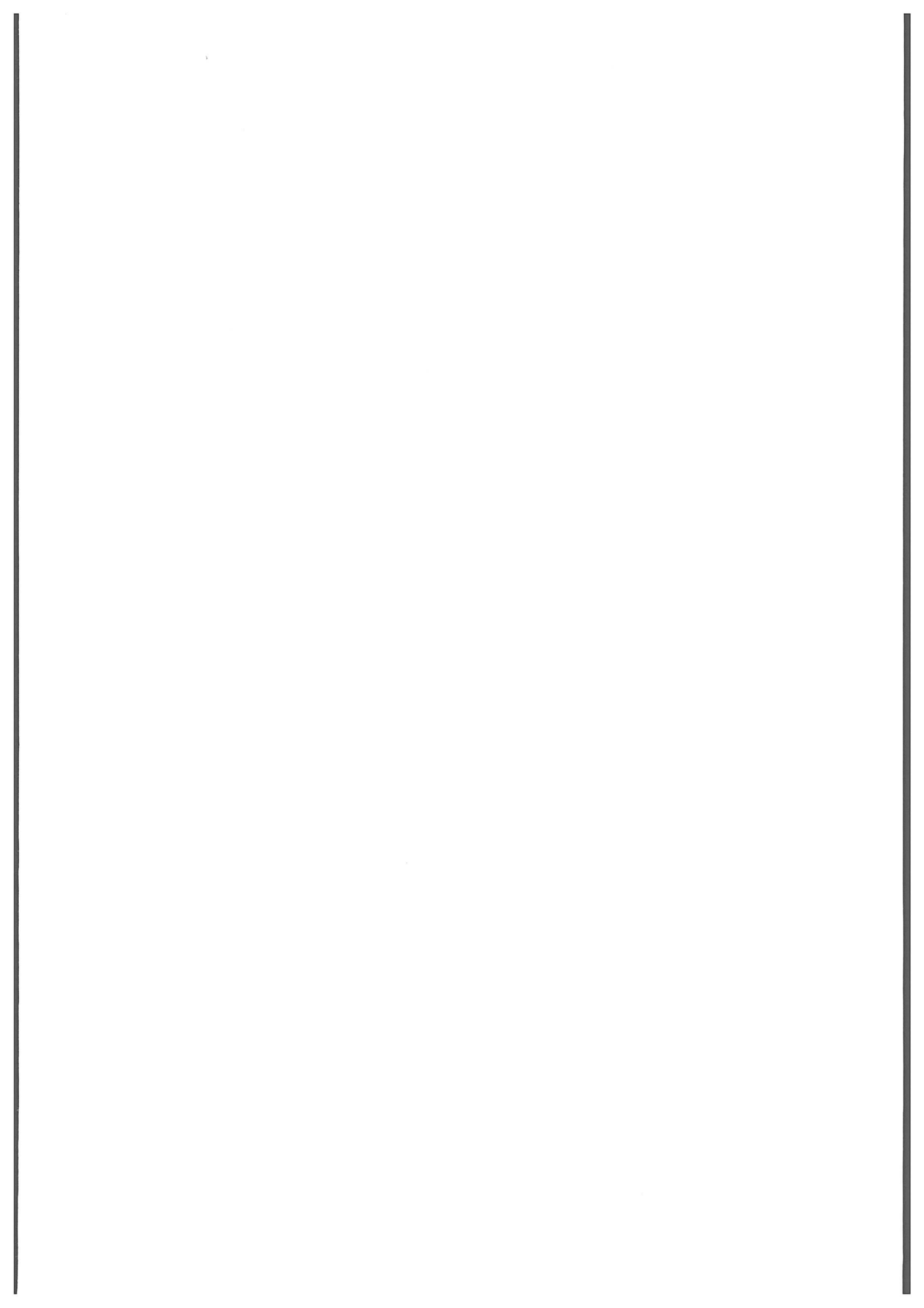
D. BEST

(Diplomarbeit, Universität Mainz)

Tracking  
mit der  
Hough-Transformation  
für die Zentrale Driftkammer  
des GSI- $4\pi$ -Experiments

Diplomarbeit  
von Dieter Best

Universität Mainz  
August 1992



# Inhaltsverzeichnis

<b>1</b>	<b>Einführung – Pattern Recognition in der Physik</b>	<b>8</b>
<b>2</b>	<b>Die Physik</b>	<b>12</b>
<b>3</b>	<b>Der <math>4\pi</math>-Detektor</b>	<b>18</b>
<b>4</b>	<b>Methoden der Spurerkennung</b>	<b>28</b>
4.1	Die Aufgabe der Spurerkennung . . . . .	28
4.2	Methoden für die Spurerkennung . . . . .	29
4.3	Die kombinatorische Methode . . . . .	30
4.4	Klassifizierung der Methoden . . . . .	31
4.5	Lokale Methoden . . . . .	32
4.5.1	Die Track-Following-Methode . . . . .	32
4.5.2	Die Track-Road-Methode . . . . .	33
4.5.3	Die Track-Element-Methode . . . . .	34
4.6	Globale Methoden . . . . .	35
4.6.1	Die Histogramm-Methode . . . . .	35
4.6.2	Das Template Matching . . . . .	37
4.6.3	Der Minimum-Spanning-Tree . . . . .	39
<b>5</b>	<b>Einführung der Hough-Transformation</b>	<b>41</b>
<b>6</b>	<b>Die adaptive Hough-Transformation</b>	<b>51</b>
6.1	Die Idee der adaptiven HT . . . . .	51
6.2	Flußdiagramm der AHT . . . . .	53
6.3	Das Binning . . . . .	53
6.4	Das Clusterfinding mit dem Connected Components Algorithmus . . . . .	54
6.5	Der neue Parameterbereich . . . . .	61
6.6	Die Komplexität der AHT . . . . .	63
6.7	Die Strategie der Spurerkennung . . . . .	64

6.7.1	Die Vorselektion der Daten . . . . .	64
6.7.2	Das Backmapping . . . . .	66
<b>7</b>	<b>Das FOPI Simulationsprogramm</b>	<b>69</b>
<b>8</b>	<b>Test mit Simulationen</b>	<b>74</b>
8.1	Effizienz und Zeitaufwand . . . . .	74
8.2	Die Doppelspurauflösung . . . . .	84
<b>9</b>	<b>Track-Fitting</b>	<b>86</b>
9.1	Einführung . . . . .	86
9.2	Konzepte . . . . .	87
9.3	Das Track-Modell . . . . .	90
9.4	Globale Methode der kleinsten Quadrate . . . . .	91
9.5	Track-Fit für die SIS $4\pi$ CDC . . . . .	95
9.5.1	Parametrisierung der Helix . . . . .	95
9.5.2	Die Track-Fit-Strategie . . . . .	97
9.5.3	Ein schneller Kreisfit . . . . .	99
9.5.4	Der Helix-Fit (CALL CFTHHEL) . . . . .	101
9.5.5	Helixfit mit Vielfachstreuung (CALL CMSHEL) . . . . .	102
9.5.6	Resultate des Trackfits . . . . .	104
<b>10</b>	<b>Vertex-Fitting</b>	<b>110</b>
10.1	Konzepte . . . . .	110
10.2	Vertex-Fit für die SIS $4\pi$ CDC . . . . .	110
10.3	Resultate des Vertex-Fits . . . . .	114
<b>11</b>	<b>Test mit experimentellen Daten</b>	<b>118</b>
11.1	Die Strahlzeit . . . . .	118
11.2	Das Verhalten der Driftkammer . . . . .	119
11.3	Kalibrierung der Driftkammer . . . . .	121
11.4	Ergebnisse der Strahlzeit . . . . .	123
<b>12</b>	<b>Zusammenfassung</b>	<b>127</b>
<b>A</b>	<b>Formeln für den Track-Fit</b>	<b>131</b>
<b>B</b>	<b>Formeln für den Vertex-Fit</b>	<b>133</b>

# Abbildungsverzeichnis

1.1	Streamer-kammerbild . . . . .	9
1.2	Typisches Bild einer Driftkammer . . . . .	10
2.1	Phasendiagramm für Kernmaterie . . . . .	13
2.2	Verlauf der nuklearen Zustandsgleichung . . . . .	14
2.3	Bounce-Off und Side-Splash . . . . .	15
2.4	Raum-Zeit-Verhalten einer Schwerionenkollision . . . . .	16
3.1	Seitenansicht des $4\pi$ -Detektors . . . . .	19
3.2	Querschnitt vom Aufbau des $4\pi$ -Detektors . . . . .	20
3.3	Spur und Spiegelspur in der CDC . . . . .	22
3.4	3D-Ansicht vom Aufbau des $4\pi$ -Detektors . . . . .	25
3.5	Testaufbau der zentralen Driftkammer . . . . .	26
4.1	Ca+Ca Ereignis bei 1GeV/u . . . . .	29
4.2	Die Track-Following-Methode . . . . .	32
4.3	Die Track-Element-Methode . . . . .	35
4.4	Konforme Abbildung . . . . .	36
4.5	$\phi$ -Histogramm . . . . .	37
4.6	Ein spanning tree . . . . .	40
5.1	Die Hough-Transformation für ein Geradenstück . . . . .	42
5.2	Parameterraum für ein Geradenstück . . . . .	43
5.3	Die Hough-Transformation, um Kreise zu finden . . . . .	44
5.4	rz-Projektion eines Ereignisses in der CDC . . . . .	46
5.5	xy-Projektion eines Ereignisses in der CDC . . . . .	47
5.6	Akkumulator für das Hit-Pattern . . . . .	48
5.7	Verteilung der Schnittpunktefehler . . . . .	49
6.1	Flußdiagramm für die AHT . . . . .	52
6.2	Binning für die AHT . . . . .	54
6.3	Binary Input Image . . . . .	55

6.4	Connected Components Image . . . . .	56
6.5	Typisches Aussehen des Hough-Feldes . . . . .	61
6.6	Herausgefilterter Peak . . . . .	62
6.7	rz-Projektion eines Ereignisses in der CDC . . . . .	64
6.8	$\theta$ -Histogramm . . . . .	65
6.9	Flußdiagramm der Spurrekonstruktion . . . . .	67
7.1	Struktur der Bank CHIT . . . . .	70
7.2	Struktur der Bank CPOS . . . . .	71
7.3	Struktur der Track-Bank CTRK . . . . .	72
8.1	Einzeltrack-Effizienz . . . . .	75
8.2	Effizienz der Spurrekonstruktion für Ca+Ca . . . . .	76
8.3	Transversale Impulsverteilung für Ca+Ca bei 1 GeV/u . . . . .	77
8.4	CPU-time für Ca+Ca . . . . .	78
8.5	CPU-time für Au+Au . . . . .	79
8.6	Effizienz der Spurrekonstruktion für Au+Au . . . . .	80
8.7	xy-Projektion eines Ereignisses in der CDC . . . . .	81
8.8	rz-Projektion eines Ereignisses . . . . .	82
8.9	Doppelspur-Auflösung der Hough-Transformation . . . . .	84
9.1	Input und Methode beim Track-Fit . . . . .	88
9.2	Effekt der Vielfachstreuung auf den Track . . . . .	90
9.3	Der Track-Fit als Projektion auf die constraint surface . . . . .	93
9.4	Wahl der Helixparameter in der $s_{xy}z$ -Ebene . . . . .	97
9.5	Wahl der Helixparameter in der xy-Ebene . . . . .	98
9.6	$\chi^2$ -Verteilung und -Wahrscheinlichkeit des Trackfits . . . . .	103
9.7	Pulls und normalisierte Pulls der Trackhits . . . . .	105
9.8	Transversale Impulsauflösung des Trackfits . . . . .	106
9.9	Auflösungsvermögen des Trackfits . . . . .	108
9.10	Normalisierte Pulls und Fehlerverteilung des Trackfits . . . . .	109
10.1	Meßwerte und Parameter beim Vertex-Fit . . . . .	111
10.2	Wahl der Parameter für den Vertex-Fit in der $s_{xy}z$ -Ebene . . . . .	112
10.3	Wahl der Parameter für den Vertex-Fit in der xy-Ebene . . . . .	113
10.4	Pull- und Fehlerverteilung beim Vertex-Fit . . . . .	115
10.5	$\chi^2$ -Verteilung und -Wahrscheinlichkeit für den Vertex-Fit . . . . .	116
11.1	Ein Ereignis aus der Strahlzeit vom April 92 . . . . .	119
11.2	$p_T$ -Verteilung der Strahlzeit von Anfang April 1992 . . . . .	120
11.3	dE/dx-Plot im Experiment . . . . .	121

11.4 Ortsauflösung der Driftkammer im Experiment . . . . .	122
11.5 Point-Removal-Efficiency im Experiment . . . . .	123
11.6 Vertex-Verteilung im Experiment . . . . .	124
11.7 Zeitaufwand im Experiment . . . . .	125

# Tabellenverzeichnis

4.1	Ein Beispiel für template matching . . . . .	38
6.1	Labelzuordnung . . . . .	58
6.2	Variablendeklaration für den CCA . . . . .	59
9.1	Eigenschaften der Methode der kleinsten Quadrate . . . . .	87
9.2	Konfidenzintervalle für normalverteilte Größen . . . . .	94
9.3	Parametrisierung der Teilchenspur . . . . .	95

*Als Zarathustra dreißig Jahre alt war, verließ er seine Heimat und den See seiner Heimat und ging in das Gebirge. Hier genoß er seines Geistes und seiner Einsamkeit und wurde dessen zehn Jahre nicht müde. Endlich aber verwandelte sich sein Herz, – und eines Morgens stand er mit der Morgenröte auf, trat vor die Sonne hin und sprach zur ihr also:*

*Du großes Gestirn! Was wäre dein Glück, wenn du nicht die hättest, welchen du leuchtest!*

*Zehn Jahre kamst du hier herauf zu meiner Höhle: du würdest deines Lichtes und dieses Weges satt geworden sein, ohne mich, meinen Adler und meiner Schlange.*

*Aber wir warteten deiner an jedem Morgen, nahmen dir deinen Überfluß ab und segneten dich dafür.*

*Siehe! Ich bin meiner Weisheit überdrüssig, wie die Biene, die des Honigs zu viel gesammelt hat, ich bedarf der Hände, die sich ausstrecken.*

– FRIEDRICH NIETZSCHE, ALSO SPRACH ZARATHUSTRA,  
ERSTER TEIL, ZARATHUSTRAS VORREDE

# Kapitel 1

## Einführung – Pattern Recognition in der Physik

*Die Zeit ist da, sagte das Walroß, um über viele Dinge zu sprechen.*

— L. CARROL

Seit den Tagen der Wilsonschen Nebelkammer müssen Teilchenspuren, sogenannte *Tracks*, in Gaskammern erkannt und gemessen werden. Dieser Prozeß wird als *Pattern Recognition* bezeichnet.

Anfangs war das Aufzeichnungsmedium ein Film. Filme werden auch heute noch verwendet bei sogenannten *optischen* Detektoren, wie zum Beispiel Blaskammern, optischen Sparkkammern und optischen Streamerkammern (siehe Abbildung 1.1), bei denen geladene Teilchen sichtbare Tracks hinterlassen, die fotografiert werden können.

Seit etwa 1960 werden auch Computer dafür eingesetzt. Dies setzt voraus, daß die Daten in digitaler Form vorliegen. Drei verschiedene Methoden wurden dafür entwickelt:

1. Fotos werden digitalisiert, d.h. sie werden in kleine Zellen aufgeteilt, und der *Grauwert* jeder Zelle wird aufgenommen.
2. Anstatt das Bild zu fotografieren, wird es mit einer TV Kamera aufgenommen. Das so erhaltene Analogsignal, eine elektrische Spannung, wird digitalisiert.
3. Sogenannte elektronische Detektoren wurden entwickelt. Diese basieren auf anderen Effekten als der Lichtemission, um den Durchgang von

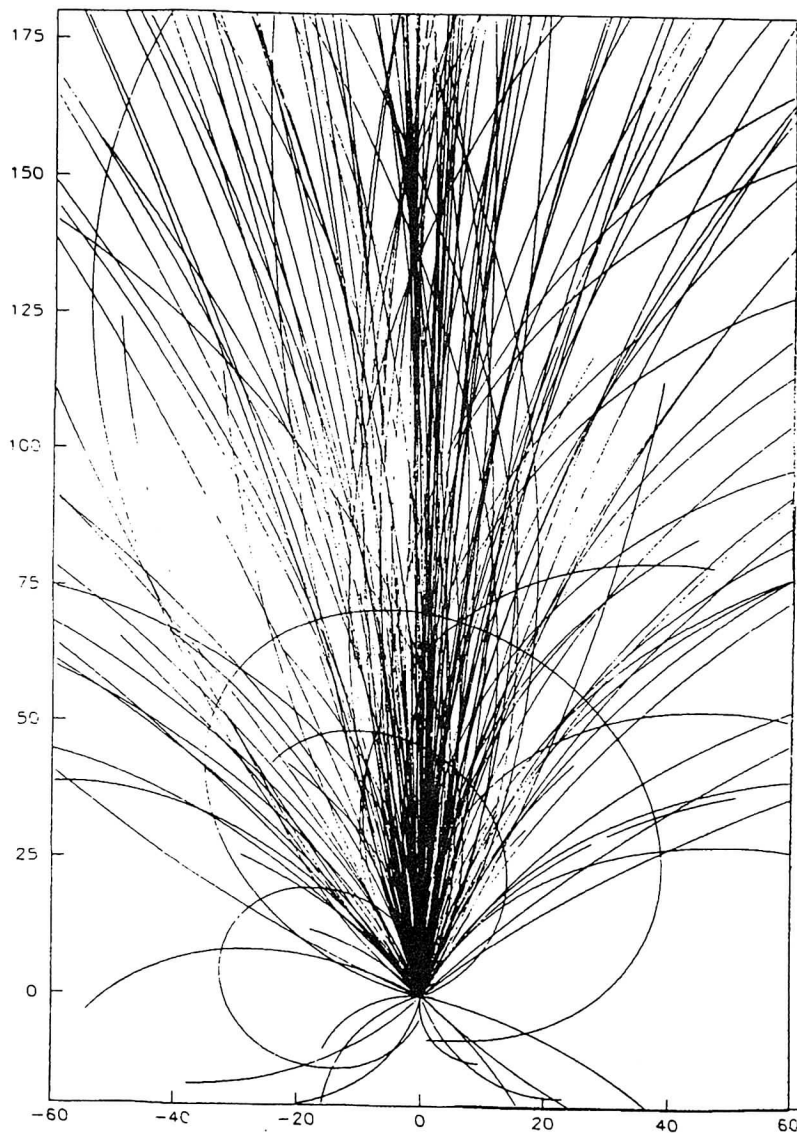


Abbildung 1.1: Simuliertes Streamerkammerbild für eine zentrale Kollision von  $^{16}\text{O}+\text{Xe}$  bei  $200\text{ GeV/u}$ , generiert als Superposition von 16  $p+\text{Xe}$  Reaktionen aufgenommen im Experiment NA35 [STO85].

---

Teilchen festzustellen. Sie liefern entweder eine direkte digitalisierte Information, beispielsweise die Drahtnummer, oder sie produzieren ein Analogsignal, das dann digitalisiert wird.

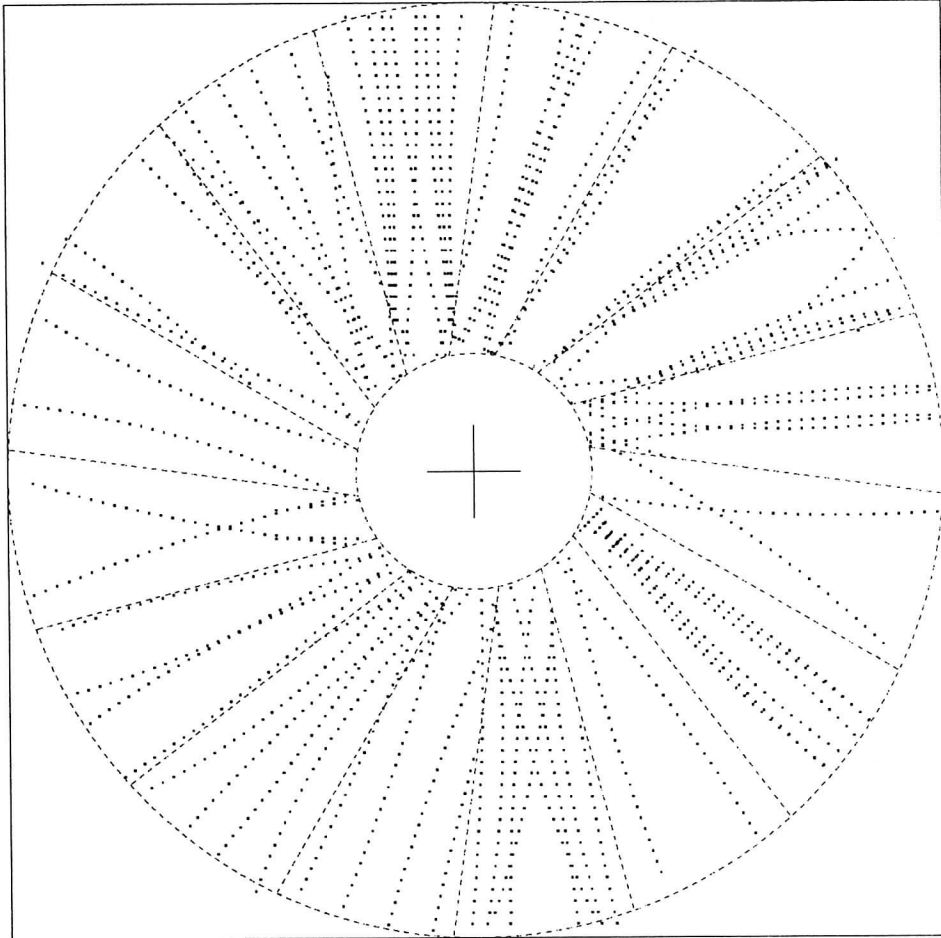


Abbildung 1.2: Projektion einer simulierten Au+Au-Reaktion bei 1 GeV/u in der SIS-4 $\pi$ -CDC (zentrale Driftkammer) senkrecht zur Strahlachse. Die Kammer befindet sich in einem solenoidalen Magnetfeld parallel zur Strahlachse. Das Kreuz zeigt die Targetposition an.

Elektronische Detektoren (Abb. 1.2) haben heute fast vollständig Blasenkammern oder andere optische Detektoren verdrängt, hauptsächlich weil man mit ihnen besser triggern kann, weil sie eine höhere Auflösung und eine kürzere Totzeit haben.

Die automatische Verarbeitung der Daten ist nun die Regel. Die Software, um diese Aufgabe zu erfüllen, ist ein integraler Bestandteil des Experiments. Darunter ist die *Spurerkennung*, das *Track-Finding*, von entscheidender Bedeutung für die meisten Experimente. Dies geht so weit,

daß beim Design moderner Detektoren die Track-Finding-Methode, die angewandt werden soll, mit berücksichtigt wird [GRO87]. Bei elektronischen Detektoren, die typischerweise Bilder wie in Abbildung 1.2 liefern, besteht das Problem der Spurerkennung darin, aus den gemessenen Koordinaten, den sogenannten *Hits*, die Teilchenspur räumlich zu rekonstruieren. So erhält man die Richtung und über die Krümmung der Spur im Magnetfeld den Betrag des Teilchenimpulses.

Ziel dieser Arbeit ist es, die Leistungsfähigkeit einer neuen Methode zur Spurerkennung in Driftkammern, der *Hough-Transformation* [HOU62], zu zeigen. Als Einführung in die Problematik soll im folgenden die relevante Physik und der Detektor, mit dem diese Physik untersucht wird, beschrieben werden. Daran schließt sich ein Überblick der vorhandenen Track-Finding-Methoden an. Dann wird die Hough-Transformation und deren Möglichkeiten beschrieben. Es folgen darauf zwei Kapitel über Spuranpassung, dem sogenannten *Track-Fitting*, und die Rekonstruktion des primären Vertex, dem *Vertex-Fitting*, die für eine vollständige Analyse nötig sind und einen Vergleich mit den simulierten Ereignissen erst ermöglichen beziehungsweise es erlauben, physikalische Informationen zu extrahieren.

# Kapitel 2

## Die Physik

*One never notices what has been done;  
one can only see what remains to be  
done.*

– MARIE CURIE

Geht man davon aus, daß die Kernmaterie, die uns umgibt, sich thermodynamisch beschreiben läßt, so kann man ihr unter anderem folgende Eigenschaften zuordnen:

- Die Dichte  $\rho$  ist etwa  $\rho_0 \approx 0.17$  Nukleonen/ $fm^3$ .
- Die Temperatur  $T$  ist im Grundzustand 0.
- Der Druck  $P$  ist 0.

Nun ist es einmal von grundlegendem kernphysikalischem Interesse, wie der Druck sich in Abhängigkeit von der Temperatur und Dichte verhält, also die *Zustandsgleichung der Kernmaterie* zu bestimmen:

$$P = P(\rho, T).$$

Eine Fragestellung ist zum Beispiel, ob es Phasenübergänge gibt.

Abbildung 2.1 zeigt ein Phasendiagramm für Kernmaterie in den Größen  $\rho$  und  $T$ . Mehrere neue Phasen von Kernmaterie werden in diesem Diagramm erwartet. In der Quantenfeldtheorie, respektive QCD, erwartet man zum Beispiel einen Phasenübergang zum Quark–Gluon–Plasma.

Eine zentrale Rolle spielt die nukleare Zustandsgleichung aber auch in der Astrophysik für das Verständnis der Struktur und Dynamik von Sternen, insbesondere der Dynamik von Super–Nova–Explosionen, und

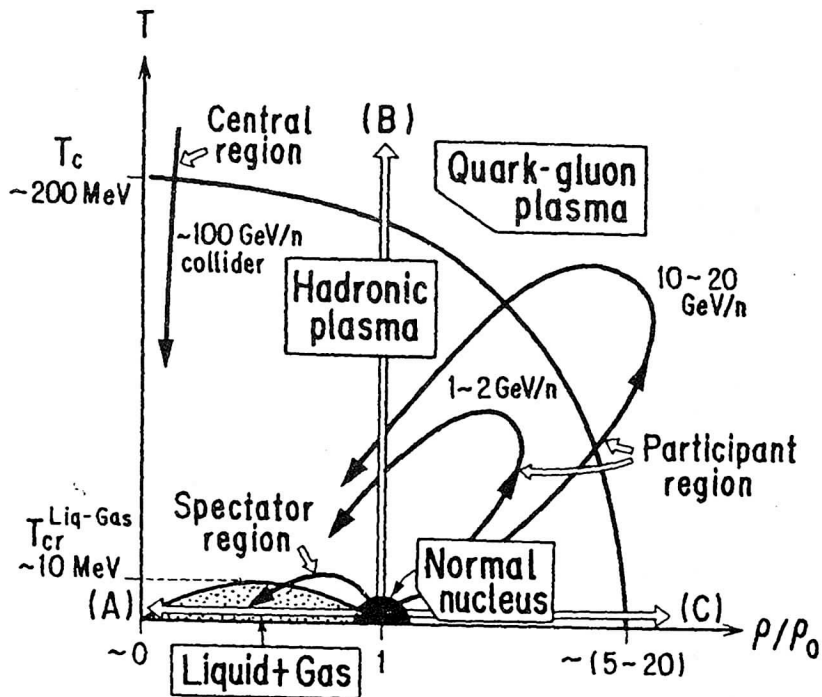


Abbildung 2.1: Erwartete neue Formen von Kernmaterie in der  $(\rho, T)$ -Ebene [NAG84]. Mögliche Pfade während eines Schwerionenstoßes sind mit durchgezogenen Linien angezeigt. (A) Übergang vom flüssigen in den gasförmigen Zustand. (B) Übergang von normaler Kernmaterie zu hadronischem Plasma und schließlich Quark-Gluon-Plasma wenn  $\rho$  konstant ist und  $T$  nur erhöht wird. (C) Übergang von normaler Kernmaterie zu Quark-Gluon-Plasma bei konstanter Temperatur  $T$  und erhöhter Dichte  $\rho$ .

der Stabilität von Neutronensternen. Diese Prozesse hängen stark von der Kompressibilität der Kernmaterie ab. Ebenso werden die Verhältnisse Bruchteile von Sekunden nach dem Urknall entscheidend beeinflusst von den Vorgängen, die in diesem Zusammenhang von Interesse sind.

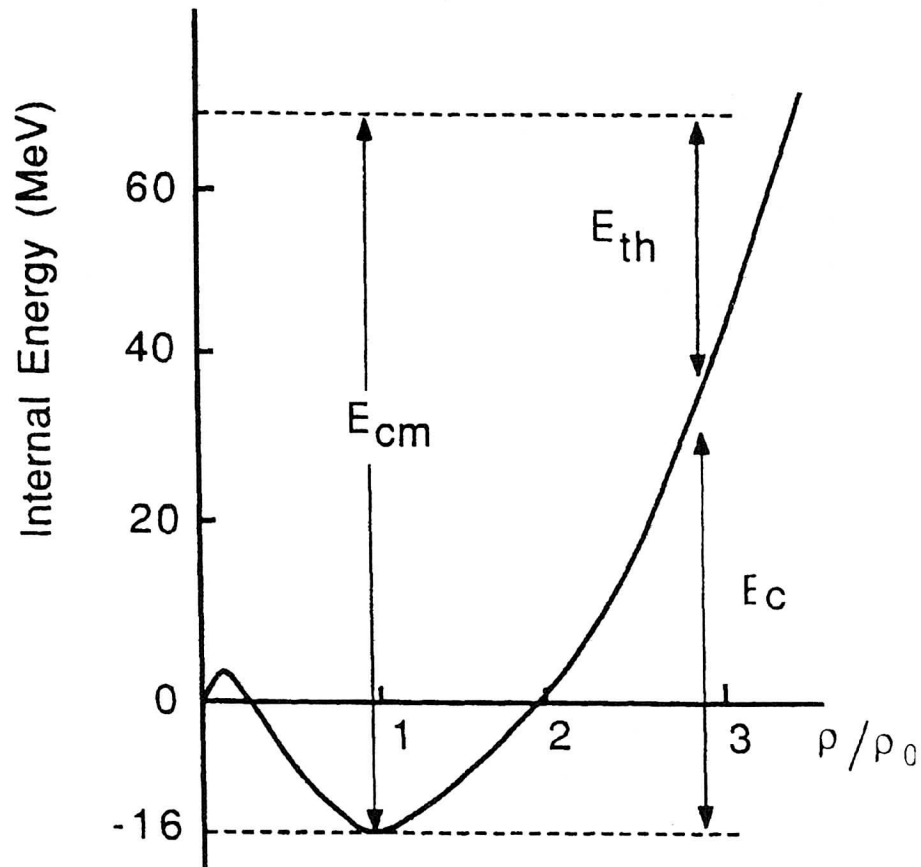


Abbildung 2.2: Erwarteter Verlauf der nuklearen Zustandsgleichung [STO86]: Die Skizze zeigt die Funktion  $W(\rho, T = 0)$ , die die Energie pro Baryon in Kernmaterie mit der Dichte in Verbindung bringt. Die Dichte  $\rho$  wird dabei in Einheiten von  $\rho_0 = 0.17$  Nukleonen/ $fm^3$  gemessen. Die Reaktion fängt am tiefsten Punkt an (vor der Reaktion), dann kommt die Kompressionsphase und schließlich die Expansionsphase (Expansion bis zum freeze-out). Während der zweiten Phase wird die Energie aufgeteilt auf thermische Energie ( $E_T$ ) und Kompressionsenergie ( $E_c$ ).

Diese astrophysikalischen Vorgänge sind aber nicht oder nur sehr schwer einer Beobachtung zugänglich. Durch Stöße schwerer Kerne mit relativistischen Energien erreicht man aber im Labor Zustände hochangeregter und verdichteter Kernmaterie.

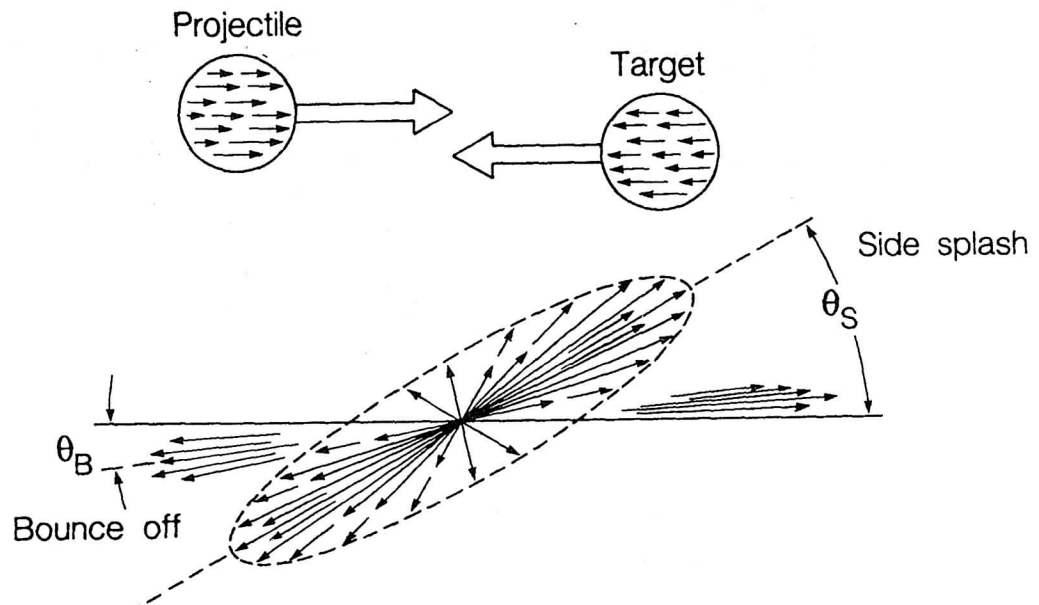


Abbildung 2.3: Schematisierte Ansicht des Side-Splash und Bounce-Off Effektes in relativistischen Schwerionenstößen [RIT85]

Während einer Schwerionenkollision beschreibt man im Phasendiagramm einen Pfad. Die Zeitentwicklung einer solchen Kollision hängt entscheidend von der nuklearen Zustandsgleichung ab.

Abbildung 2.2 zeigt den erwarteten Verlauf der nuklearen Zustandsgleichung in den Variablen interne Energie  $W$  pro Nukleon gegen Kompression. Die Kurve  $W(\rho/\rho_0, T = 0)$ , mit einer Parabelform um  $\rho_0$ , gibt die Grundzustandsenergie pro Nukleon von ausgedehnter Kernmaterie wieder: Das ist die Funktion, die man von den Daten zu extrahieren hofft. Sie wird als Zustandsgleichung der Kernmaterie bezeichnet.

Eines der ersten Modelle zur Beschreibung relativistischer Schwerionenstöße war das sogenannte "Feuerball"-("fireball"-) Modell. Abbildung 2.4 zeigt die Modellvorstellung einer Schwerionenkollision. Durch den Stoßparameter der Reaktion wird bestimmt ("clean cut geometry"), wieviele Nukleonen miteinander wechselwirken ("participants") und wieviele nicht auf die Reaktionszone treffen ("spectators").

Dynamische Modelle für Schwerionenstöße bei relativistischen Energien sagen eine Kompressions-Expansions-Abfolge der Participants voraus. Die Reaktion fängt mit der Grundzustandsenergie von  $-16 \text{ MeV/A}$

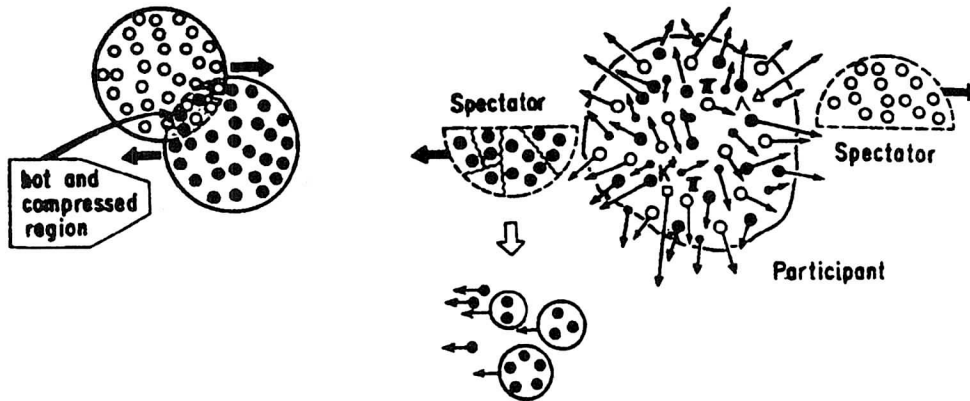


Abbildung 2.4: Erwartetes Raum-Zeit-Verhalten einer Schwerionenkollision bei hohen Energien [NAG84]

an, die relative CM-Energie ist durch die Strahlenergie gegeben. Darauf folgt die Kompressionsphase. Nach der maximalen Kompression beginnt die Expansion des Systems bis zum freeze-out [STO86], der dadurch definiert ist, daß es im System keine Stöße mehr gibt.

Aufschluß über die Kompressionseffekte liefern die experimentelle Beobachtung des sogenannten "side splash" und "squeeze out" (Abbildung 2.3).

Gegenwärtig ist man aber noch weit davon entfernt, die dynamischen Prozesse bei einer Schwerionenkollision im Detail bestimmen zu können. Mit Bose-Einstein-Interferometrie, dem sog. Hanbury-Brown/Twiss-Effekt, kann man Aufschlüsse über die beteiligten Wechselwirkungsvolumina bzw. Zeitskalen erhalten und daraus auf die Dichte schließen.

Pionen und  $K^+$ -Mesonen ergeben verschiedene Dichten, da die gemessenen  $K^+$ -Mesonen aufgrund der größeren mittleren freien Weglänge von Kaonen aus einer früheren Phase der Kollision stammen als die Pionen, die mit viel größerer Wahrscheinlichkeit absorbiert werden.

Aufschluß über die Temperatur erhält man unter gewissen Bedingungen aus den gemessenen Energieverteilungen von beispielsweise Pionen unter einem bestimmten Winkel. Man wählt einen Winkel von  $90^\circ$  im Schwerpunktsystem, weil dort der Spectator-Beitrag am kleinsten ist. Die Form des Spektrums ist etwa exponentiell:

$$E(d^3\sigma/dp^3) \propto \exp(-E^{cm}/E_0).$$

Die Größe  $E_0$  kann für ein thermodynamisches System im globalen

Gleichgewicht mit einer Temperatur identifiziert werden, da die Form des Spektrum von einer thermalisierten Quelle durch die Boltzmann-Verteilung gegeben ist.

Auf diese Weise kann man versuchen, über den Vergleich mit dynamischen Theorien (beispielsweise BUU, QMD, hydrodynamische Modelle) für jede emittierte Teilchenspezies experimentelle Punkte in der  $(\rho, T)$ -Ebene zu erhalten. Indem man diese Punkte miteinander verbindet, kann man etwas über die Zeitentwicklung einer Schwerionenkollision lernen.

“Inklusive“ (Einteilchen-)Spektren zeigen zwar thermische beziehungsweise statistische Eigenschaften, doch ihre Interpretation ist durch die Vielzahl der möglichen Beiträge nicht eindeutig. Die möglichen Reaktionsmechanismen lassen sich aber durch den Nachweis möglichst aller Teilchen und eine konsistente Beschreibung der verschiedenen Teilchenspezies einschränken. Dies strebt man durch den Aufbau eines “exklusiven“ ( $4\pi$ -) Experiments an.

# Kapitel 3

## Der $4\pi$ -Detektor

*When choosing between two evils, I always like to try the one I've never tried before.*

– MAE WEST

Durch einen möglichst vollständigen Nachweis aller Teilchen, die in einem Kernstoß erzeugt werden, will man Informationen über die sogenannte Zustandsgleichung der Kernmaterie erhalten. Daher wurde an der GSI in Darmstadt begonnen, ein  $4\pi$ -Experiment aufzubauen, das Detektoren mit hoher Effizienz zum Nachweis geladener Teilchen, wie Protonen, leichte Kernbruchstücke und Pionen enthält.

Der  $4\pi$ -Detektor ist ein hochmodulares Detektorsystem mit mehreren Detektormodulen [GOB87, GSI1189, GSI0491]. Abbildung 3.1 zeigt eine Seitenansicht des Detektors. Der Detektor besteht aus folgenden Elementen:

- äußere Plastikwand und innere Plastikwand (0-Grad Detektor)
- Ionisationskammern (Parabola) und Rosace
- Zentrale Driftkammer (CDC)
- Helitron
- Plastik Barrel und Čerenkov Barrel
- supraleitender Solenoid

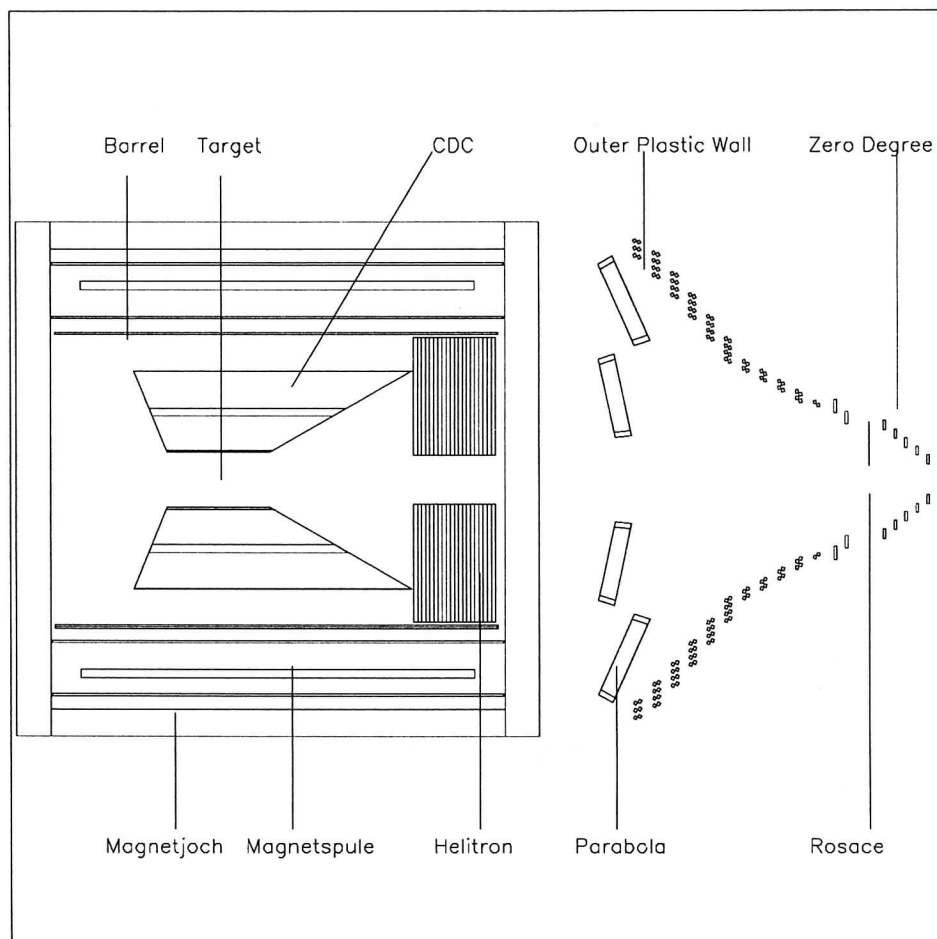


Abbildung 3.1: Seitenansicht des  $4\pi$ -Detektors

Der Detektor wird in zwei Schritten aufgebaut. Phase 1 besteht aus der **Vorwärtswand**, die den vollen Azimut und einen polaren Winkelbereich von  $1^\circ$  bis  $30^\circ$  abdeckt. Die Vorwärtswand besteht aus zwei Teilen:

- **Plastikwand,**
- **Cluster Detector Shell.**

Sie besteht aus insgesamt 764 Szintillator-Streifen. Diese messen die Flugzeit der Reaktionsprodukte relativ zu einem in-beam START Detektor. Die Plastikwand liefert einen schnellen Online Trigger für die Multiplizität geladener Teilchen genauso wie für deren azimutalen oder polaren Verteilung.

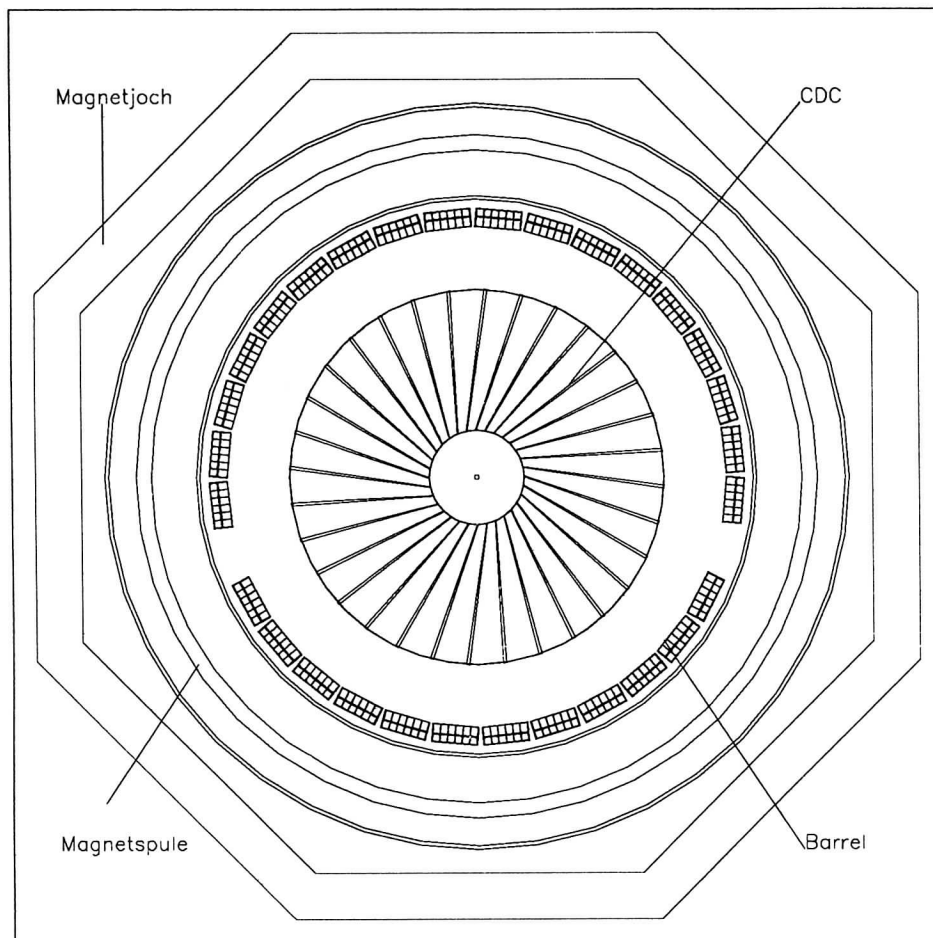


Abbildung 3.2: Querschnitt vom Aufbau des  $4\pi$ -Detektors

Die beiden Teile sind im einzelnen:

- **Äußere Plastikwand:** Die äußere Plastikwand, die den polaren Winkelbereich von  $7^\circ$  bis  $30^\circ$  abdeckt, besteht aus 512 Szintillator-Streifen in 8 Sektoren. Sie werden beidseitig ausgelesen. Die gemessene Position auf dem Streifen liefert den Azimut, den polaren Winkel erhält man durch die Position des Streifens.
- **Innere Plastikwand (0-Grad Detektor):** Der innere Bereich der Vorwärtswand besteht aus 252 trapezförmigen Szintillatoren, die nur einseitig ausgelesen werden. Sie sind in 7 konzentrischen

Ringen um die Strahlachse angeordnet, die den polaren Winkelbereich von  $1^\circ$  bis  $7^\circ$  abdecken.

Die Identifizierung der Elemente beruht auf einer  $\Delta E - TOF$ -Messung. Sie liefert die Ladung  $Z$  des Fragments, falls sie nicht im Szintillator gestoppt werden. Dies führt zu einer Minimalenergie für die Teilchen: 45 MeV für Protonen und etwa 85 MeV/u für Kohlenstoffkerne. Eine der Hauptanforderungen an den Detektor ist aber die Messung von schweren Fragmenten ("clusters") als auch von schnellen geladenen Teilchen. Daher gibt es vor der Vorwärtswand noch eine dünnere  $\Delta E$ -Shell.

Dieser Cluster-Detektor bedeckt den gleichen Winkelbereich von  $1^\circ$  bis  $30^\circ$ . Aus technischen Gründen wurden für die beiden polaren Winkelbereiche verschiedene Lösungen verwirklicht:

- **Ionisationskammern (Parabola):** Über  $6^\circ$  wird eine Wand aus 16 Ionisationskammern benutzt, die in zwei konzentrischen Ringen angeordnet sind und den Outer Plastic Wall abdecken. Jede Kammer enthält 8 Anoden. Die resultierende Granularität von 128 Elementen wird als ausreichend für die erwartete Multiplizität schwerer Fragmente angesehen.
- **Rosace:** Für den Winkelbereich von  $1^\circ$  bis  $6^\circ$  wird ein Array von 60 dünnen (2mm) Szintillator-Paddles für die  $\Delta E$ -Messung benutzt.

Phase 2 vervollständigt den Aufbau durch Detektoren zur Aufnahme von Teilchenspuren, sogenannten *Tracking-Detektoren*:

- **Zentrale Driftkammer:** Die zentrale Driftkammer [GOB88] ist eine zylindrische Driftkammer mit einem äußeren Radius von etwa 80 cm und einem inneren von etwa 20 cm.

Der Vorwärtskonus hat einen Öffnungswinkel von  $30^\circ$ . Er zeigt genau auf die Targetposition und gibt damit den Teilchen, die mit kleineren polaren Winkeln emittiert werden, den Weg frei. Sie werden dann von den Detektoren in der Vorwärtsrichtung registriert.

Der Rückwärtskonus hat einen Öffnungswinkel von  $60^\circ$ . Er war ursprünglich nicht vorgesehen, mußte aber aus Gründen der mechanischen Stabilität eingeführt werden.

Damit wird der volle Azimut und ein polarer Winkelbereich von  $30^\circ$  bis etwa  $140^\circ$  von der zentralen Driftkammer (CDC) abgedeckt.

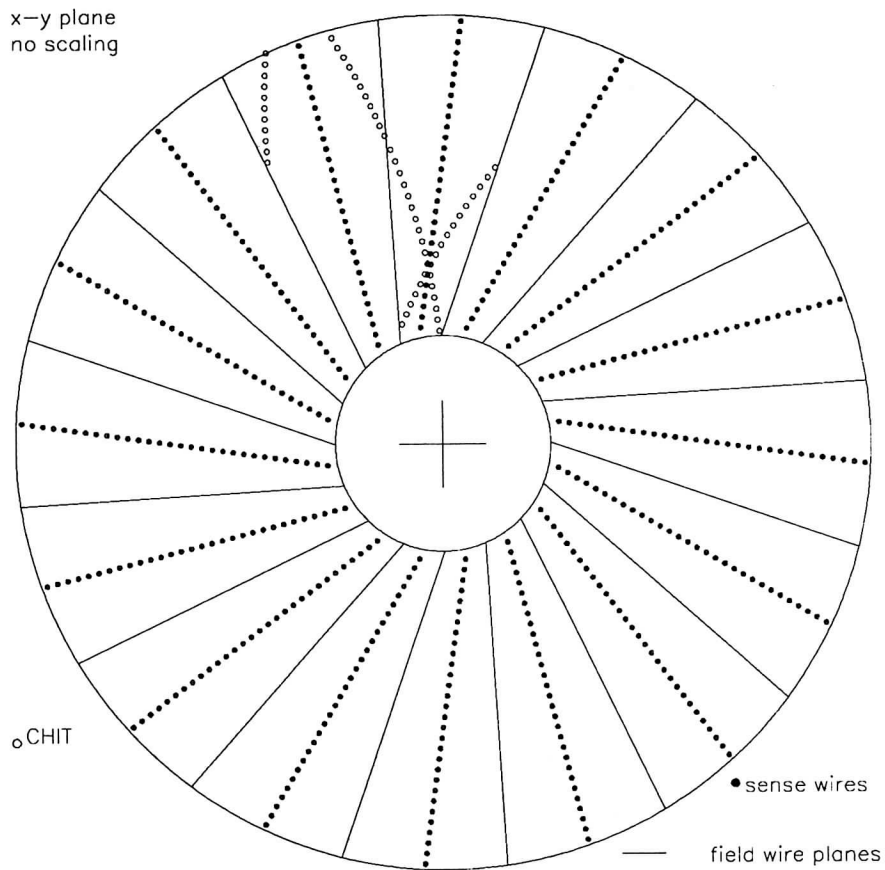


Abbildung 3.3: Spur und Spiegelspur in der CDC: Ein Querschnitt senkrecht zur Strahlachse zeigt die Segmentierung der zylindrischen Driftkammer. Sie besteht aus 16 radialen Sektoren mit 60 Signaldrähten. Sie laufen parallel zur Strahlrichtung und sind in einer Ebene angeordnet. Allerdings werden nur 30 davon ausgelesen (volle Punkte). Die durchgehende Spur von Kreisen zeigt eine Teilchenspur an, die Spiegelspur wird einmal zerrissen, wenn das Teilchen von einem Sektor in einen anderen geht, und außerdem kommt sie aufgrund der geneigten Sektoren nicht vom Target (Kreuz in der Mitte).

Die innere Länge beträgt etwa 86 cm die äußere 190 cm. Die Driftkammer ist aus einer EPOXY-Struktur aufgebaut. Der Innenmantel besteht aber aus einer dünnen Mylarfolie, um die Vielfachstreu-

ung so klein wie möglich zu halten.

Der Querschnitt senkrecht zur Strahlachse des Detektor-Systems in Abbildung 3.2 zeigt die Segmentierung der Driftkammer. Die zentrale Driftkammer ist vom Typ einer Jet-Kammer [KLE87] und ähnlich der JADE-Driftkammer am DESY [DRU80] oder der DIOGENE-Driftkammer [ALA87] in Saclay aufgebaut:

Das zylindrische Volumen der CDC ist in 16 radiale Sektoren aufgeteilt. In jedem Sektor befinden sich 60 Signaldrähte mit einem Abstand von 10 mm. Sie verlaufen parallel zur Strahlachse und sind in einer Ebene angeordnet (Abbildung 3.3), die von innen nach außen zeigt. In dieser Ebene sind die Drähte leicht gegeneinander versetzt, sie weisen ein sogenanntes *Staggering* auf, um die elektrostatische Stabilität der Drahtanordnung zu erhöhen. Ein Charakteristikum der  $4\pi$ -Driftkammer ist, daß die Sektoren nicht auf das Target ausgerichtet sind, vielmehr sind sie, wie Abbildung 3.2 zeigt, mit einem *Tilt-Winkel* von  $8^\circ$  verdreht.

Die zentrale Driftkammer enthält ein Gasgemisch von 88% Argon, 10% Methan und 2% Isobutan und arbeitet unter Atmosphärendruck. Die Elektronen, die von den durchfliegenden Teilchen im Kammergas freigesetzt werden, haben eine Driftgeschwindigkeit von etwa  $40 \mu\text{m}/\text{ns}$ .

Das elektrische Feld in der CDC steht orthogonal auf der Ebene der Signaldrähte. Die Driftkammer befindet sich im solenoidalen Magnetfeld eines supraleitenden Magneten mit einer Feldstärke von maximal 6 kG. Die Magnetfeldlinien verlaufen parallel zur Strahlachse und senkrecht zu den elektrischen Feldlinien. Aufgrund der langen Driftwege (maximal 15 cm) macht sich die Lorentzkraft bemerkbar. Die Driftzellen, also die Bereiche, aus denen die Elektronen auf einen Signaldraht driften, sind daher gegen die Drahtebene um einen Lorentzwinkel von etwa  $20^\circ$  verdreht.

Die Signaldrähte werden an beiden Enden ausgelesen. Aus den Ankunftszeiten wird die Driftzeit bestimmt und daraus die x- und y-Koordinate. Da man nicht weiß, von welcher Seite der Driftzelle die Elektronen auf den Draht gekommen sind, gibt es zu jedem Punkt noch einen sogenannten *Spiegelpunkt* und daher zu jeder Spur noch eine *Spiegelspur*. Aber aufgrund des Tiltwinkels bietet sich eine einfache Möglichkeit, Spiegelspuren in der Phase der Spurerkennung zu verwerfen, da die Spiegelspur nicht aus der Targetregion kommt.

Das ist in Abbildung 3.3 illustriert. Die  $z$ -Koordinate wird über Ladungsteilung berechnet.

Um die Kosten für die Elektronik zu reduzieren, werden von den 60 Signaldrähten aber nur 30 benutzt, so daß im allgemeinen 30 Punkte pro Track über eine Strecke von mehr als 60 cm gemessen werden. Auf diese Weise erhält man außerdem eine Ausweichmöglichkeit für eventuell gerissenen Drähte. Zusätzlich kann man versuchen, eine Gruppierung der auszulesenden Drähte zu finden, die eine bessere Impulsauflösung ermöglicht.

Die Ortsauflösung soll in der XY-Ebene  $\sigma_{r\phi} = 300\mu\text{m}$  und in  $z$ -Richtung  $\sigma_z = 20\text{mm}$  betragen ( $\approx 1\%$  der Drahtlänge). Angestrebt wird eine Doppelspurauflösung von 5-10mm.

Die zentrale Driftkammer erlaubt aber nicht nur die räumliche Rekonstruktion von gekrümmten Teilchenspuren und damit eine Impulsmessung, sondern bietet auch die Möglichkeit, den differentiellen Energieverlust  $dE/dx$  zu bestimmen. Die  $dE/dx$ -Auflösung sollte 5-10% betragen. Indem man  $dE/dx$  gegen den Impuls aufträgt ist eine Teilchenidentifizierung möglich.

- **Helitron:** Die HELITRON-Driftkammer erfüllt die gleichen Aufgaben wie die zentrale Driftkammer für den polaren Winkelbereich von  $7^\circ$  bis  $30^\circ$ . Es werden Helix-Trajektorien für geladene Teilchen aufgenommen, die in Vorwärtsrichtung emittiert und vom Magnetfeld abgelenkt werden. Eine Impulsauflösung von  $\Delta p/p \leq 4\%$  für Teilchen-Korrelation-Messungen wird angestrebt. Mit Hilfe der Flugzeitinformation von der Vorwärtswand kann man eine Teilchenidentifikation durchführen, insbesondere hofft man  $\pi^\pm$ ,  $p$ ,  $d$ ,  $t$ ,  $^3\text{He}$  und  $^4\text{He}$  trennen zu können.

Teilchen, die mit polaren Winkeln größer als  $30^\circ$  emittiert werden, sind hauptsächlich Isotope von H und He, oder es handelt sich um Pionen. Aufgrund der speziellen Eigenschaften von Kaonen (strangeness, lange mittlere freie Weglänge), ist es von großem Interesse, Kaonen zu messen. Da diese aber bei SIS-Energien selten sind, müssen zusätzliche Anstrengungen unternommen werden. Daher wird die CDC umgeben von einem Barrel-Detektor, der wiederum aus zwei Teilen besteht:

- **Plastik Barrel:** Der Plastik Barrel ist eine TOF-Detector-Shell aus 180 Plastik-Szintillator-Streifen. Sie sind 240 cm lang, 3.2 cm breit, 4 cm tief und werden an beiden Enden ausgelesen. Diese

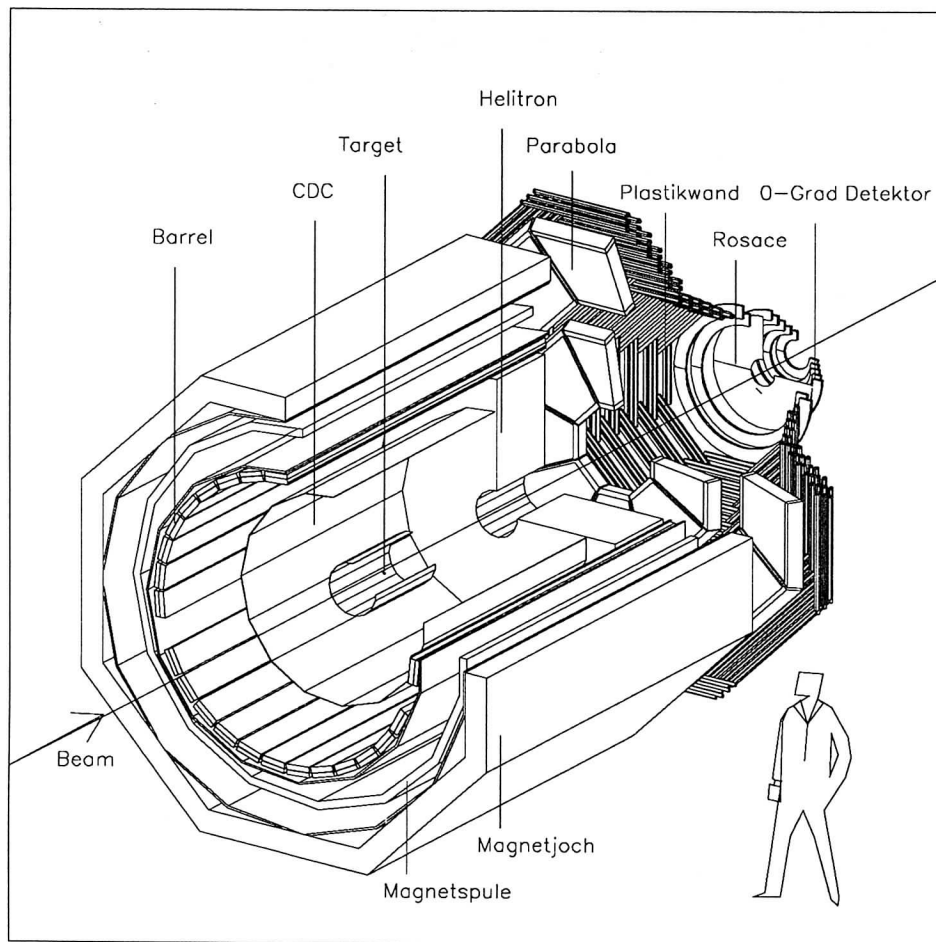


Abbildung 3.4: 3D-Ansicht vom Aufbau des  $4\pi$ -Detektors

TOF-Information kann zusätzlich zum differentiellen Energieverlust der CDC benutzt werden, um die Teilchenidentifikation zuverlässiger zu machen.

- **Čerenkov Barrel:** Der TOF-Detektor wird von einer zweiten Schale umgeben. Diese besteht aus einer gleichen Anzahl von wassergefüllten Čerenkov-Zählern, die 3cm dick sind. Ein spezielles Salz dient als Wellenlängenschieber. Die Schwelle für Kaonen ist 560 MeV/c. Die einzelnen Elemente beider Detektoren sind zu 30 Modulen gruppiert, wovon jedes 6 TOF- und 6 Čerenkov-Zellen enthält.

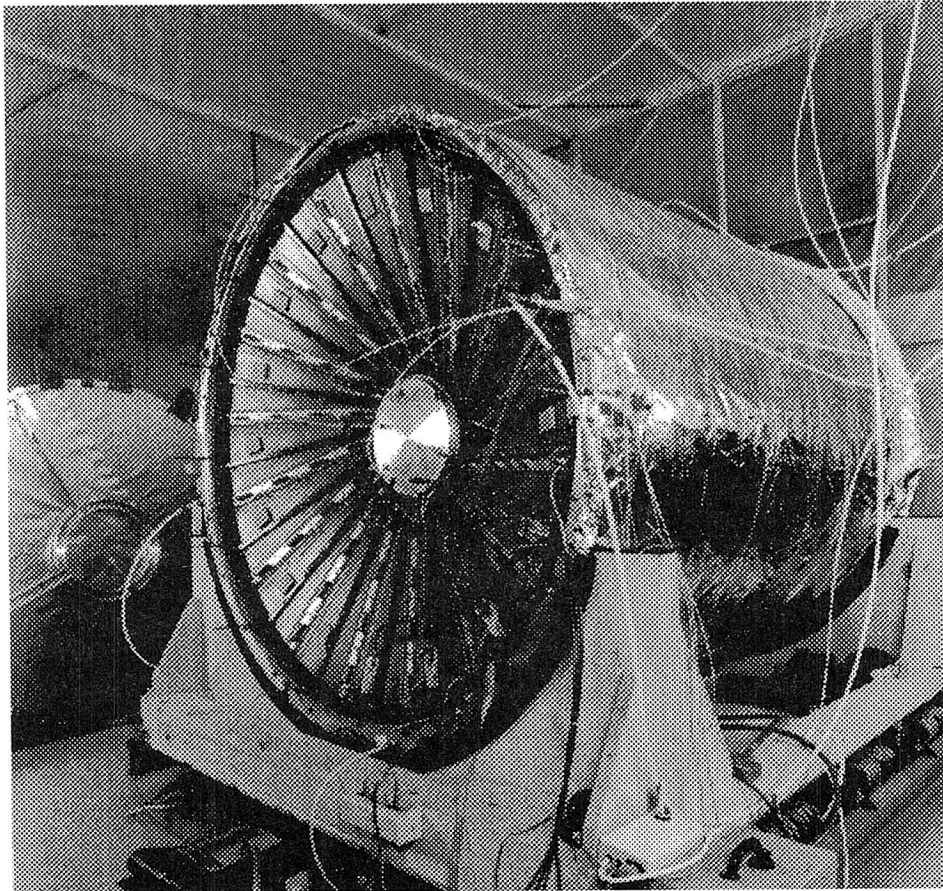


Abbildung 3.5: Testaufbau der zentralen Driftkammer in einem Zelt in der Targethalle der GSI.

---

Alle diese Detektoren arbeiten in einem Magnetfeld von etwa 0.6 T, das von einem **supraleitenden Solenoiden** erzeugt wird. Das Magnetfeld ist im Bereich der CDC auf 2% homogen, so daß man in guter Näherung von Helix-Trajektorien für die geladenen Teilchen ausgehen kann. Die dreidimensionale Abbildung 3.4 verschafft einen besseren Eindruck vom ganzen Detektor, insbesondere von der Anordnung der Barrel-Module.

Mit dem Aufbau von Phase 2 und der Einführung von Tracking-Detektoren wurde eine neue Form von Analyse-Software nötig – Pattern Recognition für die Driftkammern.

Im allgemeinen wird Pattern Recognition als schwieriges Problem betrachtet in dem Sinne, daß es keine allgemeinen und in irgendeiner Weise

optimalen Strategien gibt, um ein gegebenes Problem zu lösen.

Dies trifft auch für die zentrale Driftkammer des  $4\pi$ -Experimentes an der GSI zu, obgleich bereits viel Erfahrung mit Pattern Recognition für Driftkammern in der Hochenergiephysik gesammelt wurde. Das folgende Kapitel stellt die wichtigsten Ansätze zur Spurerkennung vor. Diese haben sich für die bis dahin erreichten Trackmultiplizitäten in Driftkammern als erfolgreich erwiesen.

Bei hadronischen Reaktionen und insbesondere Schwerionenkollisionen wird man aber mit viel höheren Trackmultiplizitäten als bisher konfrontiert. Daher haben sich die Ansprüche an das Auflösungsvermögen und den dynamischen Bereich erhöht. Gleichzeitig wurden die Detektoren in den letzten Jahren verbessert.

Damit wurde das Argument, Drahtkammern liefern entweder zu wenige oder zu ungenaue Meßwerte von gekrümmten Trajektorien, um globale Pattern Recognition Methoden anwenden zu können, obsolet:

In dieser Arbeit wird die Leistungsfähigkeit der sogenannten *Hough-Transformation* für das Pattern Recognition in Jet-Kammern vorgestellt. Damit werden die Tracking-Möglichkeiten auf viel höhere Track-Dichten erweitert, als das mit den bisher zur Verfügung stehenden Methoden möglich war.

Die Stärke der in dieser Arbeit entwickelten Implementierung der Hough-Transformation, die sogenannte *Adaptive Hough-Transformation*, besteht darin, daß sie die vom Detektor zur Verfügung gestellte globale Information ausnutzt, um Tracks mit kleinem Abstand im Phasenraum aufzulösen, eine wichtige Eigenschaft für Anwendungen zum Beispiel in der Pionen-Interferometrie. Diese Eigenschaften bleiben selbst bei 100% Untergrund erhalten. Dies wird einmal in einem Test mit Simulationen gezeigt und dann durch die Anwendung auf die ersten experimentellen Daten bestätigt.

# Kapitel 4

## Methoden der Spurerkennung

*The golden rule is that there is no golden rule.*

– GEORGE BERNARD SHAW

### 4.1 Die Aufgabe der Spurerkennung

Ist eine Menge von Ortsmessungen in einem Detektor gegeben, so ist es die Aufgabe der Spurerkennung, des *Track-Finding*, diese Menge in Teilmengen, sogenannte Klassen, aufzuspalten, so daß

1. jede Klasse Meßwerte enthält, die von einem Teilchen stammen könnten, und
2. eine Klasse alle Meßwerte enthält, die nicht einem bestimmten Teilchen mit gewisser Sicherheit zugeordnet werden konnten.

Diese Definition gibt ein realistisches Ziel vor. Sie reduziert Track-Finding zu einem *Cluster-Analyse-Problem*. Dabei ist ein *Cluster* folgendermaßen definiert [AND72]:

*A cluster is loosely defined as a collection of vectors or points that are close together.*

Daher ist man in der *Clusteranalyse* hauptsächlich damit beschäftigt, passende Abstandsdefinitionen zwischen Paaren von Objekten zu entwickeln und Algorithmen aufzustellen, die mit dieser Definition arbeiten.

Der *minimum spanning tree* Algorithmus gehört hierher und wird später noch genauer beschrieben, da er eine der wenigen Beispiele ist, eine nicht-heuristische Methode auf *real-life-Track-Finding* anzuwenden.

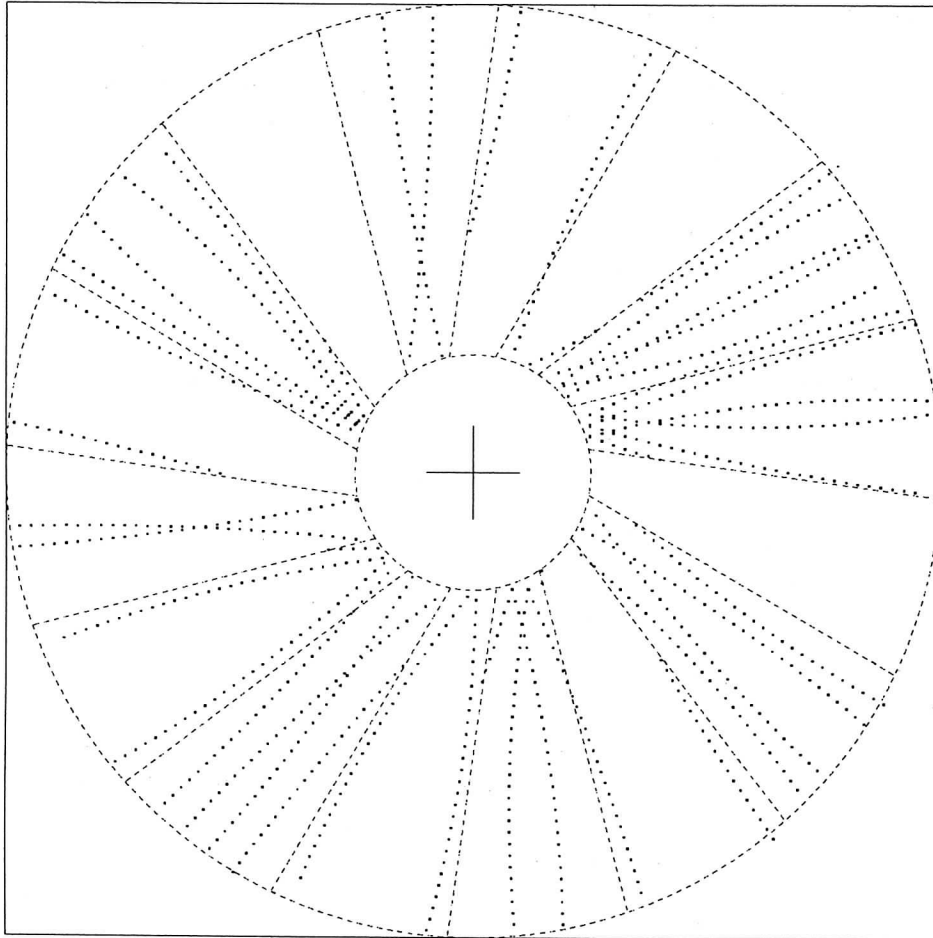


Abbildung 4.1: Beispiel für ein Ereignis niedriger Multiplizität: Ca+Ca Ereignis bei 1GeV/u. Die Tracks sind Kreisbögen, die durch den Koordinatenursprung gehen.

## 4.2 Methoden für die Spurerkennung

Alle Methoden bestehen aus zwei Schritten:

- Im ersten Schritt wird eine Teilmenge von Meßwerten gewählt, ein sogenannter *Track-Kandidat* (eigentliche Spurerkennung).
- Im zweiten Schritt wird untersucht, ob die Teilmenge ein akzeptabler Track ist, unter Umständen wird eine Spuranpassung (*Fit*) durchgeführt.

Beides zusammen wird als *Tracking* bezeichnet.

Diese Zwei-Schritte-Prozedur wird oft noch in weitere Schritte unterteilt, um das Verfahren zu beschleunigen. In einer Initialisierungsphase wird eine bestimmte Anzahl von Meßwerten ausgewählt und entweder akzeptiert oder verworfen. Werden sie akzeptiert, so werden weitere Meßwerte dazugenommen und diese neue Gesamtmenge untersucht. Diese Vorgehensweise wird wiederholt, bis man am Schluß einen Endtest auf den vollständigen Track anwendet.

Der Geschwindigkeitsvorteil gegenüber der einfachen Zwei-Schritt-Methode, die erst den vollständigen Trackkandidaten auswählt und dann die Entscheidungsfunktion anwendet, beruht auf zwei Gründen:

- Erstens hat die Reduktion der Meßwerte pro Trackkandidat eine beträchtliche Reduktion der Anzahl dieser Kandidaten zur Folge.
- Zweitens kann die Anwendung der endgültigen Entscheidungsfunktion ziemlich zeitaufwendig sein, zum Beispiel, wenn ein iterativer Fit mit einem Trackmodell ausgeführt werden soll.

*Eine Folge von Entscheidungsfunktionen, angefangen mit einer einfachen aber schnellen, bis hin zu einer präzisen, aber langsamen Methode, am Ende, verbessert die Geschwindigkeit, wenn in jedem Zwischenschritt ein Großteil der falschen Trackkandidaten verworfen wird. Dabei muß darauf geachtet werden, daß kein guter Trackkandidat durch irgendeinen der einfachen approximativen Tests verworfen wird.*

Wenn für eine gegebene Methode alle möglichen Tracks gefunden wurden, bleibt für gewöhnlich eine Anzahl von Punkten zurück, die keinem Track zugeordnet werden konnten.

Wenn verschiedene Methoden angewendet werden, oder dieselbe Methode in einer anderen Art und Weise, ist ein gutes Maß für die *Effizienz* notwendig. Ein solches Maß muß von Experiment zu Experiment neu gewählt werden.

### 4.3 Die kombinatorische Methode

Die einfachste Methode, in einer Punktmenge Teilchenspuren zu finden, besteht in der vollständig kombinatorischen Methode. Diese Methode arbeitet folgendermaßen:

1. teile die Menge aller Ortsmessungen in alle möglichen Teilmengen auf und

2. passe jede Teilmenge (=Trackkandidat) an ein Trackmodell an, nenne diese einen Track, falls der Fit gut ist.

Obwohl diese Methode in einigen einfachen Fällen, bei denen man es mit wenigen Punkten zu tun hat, angewendet werden kann, ist sie für die meisten praktischen Anwendungen zu zeitaufwendig.

Eine der Aufgaben ist es aber, den Zeitaufwand so gering wie möglich, dabei jedoch die Effizienz so hoch wie möglich zu halten. Das ist nicht immer einfach.

## 4.4 Klassifizierung der Methoden

Außer der vollständig kombinatorischen Methode, können alle weiteren Methoden zur Spurerkennung als *lokal* oder *global* klassifiziert werden.

Eine Methode heißt *global*, wenn alle Punkte durch den Algorithmus in der gleichen Art und Weise behandelt werden. Der Algorithmus produziert eine Liste von Tracks, oder wenigstens eine Liste, in der die Tracks einfacher gefunden werden können als unter den Originaldaten. Der Algorithmus kann deshalb als eine Transformation aller Koordinaten betrachtet werden. Der Zeitaufwand für eine globale Methode ist daher im Prinzip proportional zur Anzahl  $N$  der Punkte in einem Ereignis.

Im Gegensatz dazu wird bei einer *lokalen Methode* ein Trackkandidat nach dem anderen behandelt. Man fängt mit einigen wenigen Punkten an. Das ist die Track-Kandidat-Initialisierung. Dann sammelt man weitere Punkte, zum Beispiel durch Interpolation oder Extrapolation des benutzten Trackmodells. Werden zusätzliche Punkte gefunden, so werden diese zum Trackkandidaten hinzugenommen, sonst wird der Trackkandidat nach einer bestimmten Anzahl von Versuchen verworfen. Da lokale Methoden immer fruchtlose Versuche machen müssen, um einen Trackkandidaten zu finden, und so die gleichen Punkte in verschiedenen Kombinationen immer wieder benutzen, wächst der Zeitaufwand schneller als linear mit der Anzahl der Punkte.

Globale Methoden sind unabhängig davon, in welcher Ordnung die Punkte in den Algorithmus eingehen. Das trifft für lokale Methoden nicht zu, da die Behandlung jedes Punktes von der Initialisierung abhängt und vom bisherigen Verlauf der Spurerkennung.

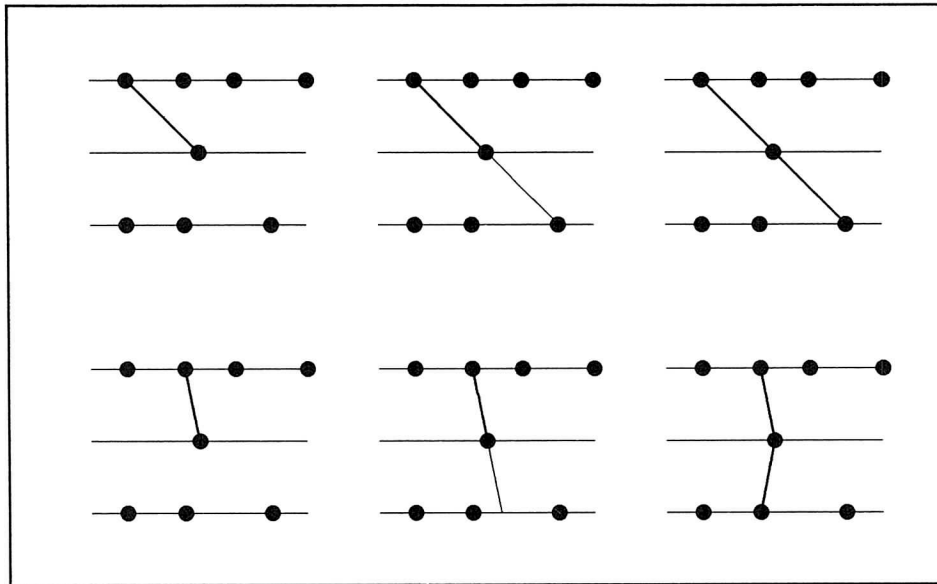


Abbildung 4.2: Die Track-Following-Methode: Die Track-Following-Methoden benutzen eine kombinatorische Initialisierung der Track-Kandidaten um die ersten drei Punkte eines Tracks zu finden. In der obigen Abbildung werden zwei Beispiele gezeigt (erste und zweite Reihe), die mit linearer Extrapolation arbeiten.

## 4.5 Lokale Methoden

### 4.5.1 Die Track-Following-Methode

Bei dieser Methode wird am Anfang ein Track-Segment, ein sogenanntes *tracklet*, gewählt, das aus einigen wenigen Punkten, drei oder vier, besteht. Dieses Tracklet wird so entfernt wie möglich von der primären Wechselwirkungszone gewählt, weil dort die Tracks im allgemeinen weitaus separierter sind als anderswo.

Im nächsten Schritt wird durch Extrapolation in die Richtung der primären Wechselwirkungszone ein weiterer Punkt gesucht. Diese Extrapolation kann von nullter Ordnung sein wie in Abbildung 4.2, indem man einfach den nächsten Nachbarpunkt wählt, von erster Ordnung, indem man mit einem Geradenstück weitergeht, oder von zweiter Ordnung, indem man eine parabelförmige Fortsetzung wählt, oder irgendeine höhere Ordnung, oder indem man irgendein Trackmodell benutzt, zum Beispiel

einen Kreis oder eine Helix. Das Ziel ist, in jedem Fall einen schnellen Algorithmus zu haben, um weitere Punkte zu finden. Die Parabelextrapolation reicht in vielen Fällen mit Magnetfeld aus, da sie das Vorzeichen der Krümmung erhält und so das Verhalten einer richtigen Teilchenspur wiedergibt. Außerdem ist diese Methode sehr schnell, da eine Parabel durch drei Punkte geschrieben werden kann als

$$y = a_1(x)y_1 + a_2(x)y_2 + a_3(x)y_3. \quad (4.1)$$

Dabei hängen die Koeffizienten nur von den Drahtpositionen in der Kammer ab, so daß sie vorher nur einmal berechnet werden müssen.

Dieser als *Track-Following-Methode* bezeichnete Ansatz benötigt nur ein lokales Trackmodell, da er nur nach den nächsten Punkten sucht, wobei die bisher gefundenen dazu benutzt werden, das Trackmodell zu extrapolieren. Diese Vorgehensweise erlaubt ein einfaches und daher schnelles Trackmodell. Werden aber andererseits die Abstände zwischen den Drähten zu groß, so ist das approximative Modell nicht präzise genug, und wegen den Meßfehlern wird selbst ein absolut korrektes Tracking, das mit bereits gefundenen Punkten arbeitet, problematisch sein.

Die Track Following Methode benutzt eine kombinatorische Initialisierung der Trackkandidaten. Wenn dann einmal einige Punkte zu diesem Kandidaten hinzugefügt wurden, so ist dies dann schon sehr wahrscheinlich ein guter Track und der *computational overhead* ist ziemlich gering. Dementsprechend ist der Zeitaufwand proportional einer Zahl zwischen  $N$  und  $N^2$ <sup>1</sup>.

Beispiele für die Track-Following-Methode findet man beim ALEPH am CERN [NOR88], beim DIOGENE in Saclay [POI83], oder auch [TRZ88].

## 4.5.2 Die Track-Road-Methode

Bei der sogenannten *Track-Road-Methode* wird keine Extrapolation wie bei der *Track-Following-Methode* benutzt, sondern die viel präzisere Interpolation, um weitere Punkte für einen Track zu finden. Man wählt Anfangspunkte an beiden Enden des Tracks und einen in der Mitte im Falle gekrümmter Tracks. Ein einfaches Trackmodell wird dann benutzt, um weitere Punkte für diesen Track zu finden, indem man eine *Road* gewisser Breite um das Trackmodell legt. Dieses Trackmodell kann exakt sein, wie

---

<sup>1</sup> $N$  = Anzahl der Punkte

zum Beispiel ein Kreis im Falle eines homogenen solenoidalen Magnetfeldes, oder eine Gerade in einem feldfreien Raum, oder es kann approximativ sein. In diesem Fall muß man die Breite der Road mit Monte-Carlo-Tracks bestimmen. Im Prinzip kann die Road um so enger sein, je besser das Trackmodell ist. In der Praxis kann man allerdings aufgrund systematischer Fehler in den Ortsmessungen und wegen des Untergrundes nicht die theoretische Breite von  $3\sigma$  der Detektoraufösung wählen. Die Track-Road-Methode ist langsamer als die Track-Following-Methode, aber manchmal ist sie die einzige Methode, die angewendet werden kann, besonders im Falle räumlich weit auseinanderliegender Detektoren. Die meisten modernen Detektoren liefern allerdings eine genügend hohe Dichte an Meßwerten, so daß eine schnellere Methode angewendet werden kann.

Da in einem Magnetfeld drei Punkte für die Initialisierung gebraucht werden, muß man wegen Detektorineffizienzen die verschiedensten Kombinationen benutzen. Die meisten Kombinationen werden dabei falsch sein, so daß der Zeitaufwand dieser Methode typischerweise proportional einem Faktor zwischen  $N^2$  und  $N^3$  ist.

Diese Methode wurde auch beim DIOGENE in Saclay verwendet [POI83].

### 4.5.3 Die Track-Element-Methode

Bei der *Track-Element-Methode* wird der Track in zwei Schritten konstruiert. Zuerst werden mit wenigen Punkten kurze Trackelemente (Tracklets) gebildet. Diese werden bei modular aufgebauten Detektoren in natürlichen Untereinheiten des Detektors konstruiert, bei einer Driftkammer zum Beispiel in einer Driftzelle. Von diesem Trackkandidaten ausgehend wird eine Extrapolation oder Interpolation nullter Ordnung (nächster Nachbar), erster Ordnung (Geradenstück) oder zweiter Ordnung (Parabelextrapolation) ausgeführt, um Trackelemente zu finden. In einem zweiten Schritt werden dann diese Elemente mit der Track-Following-Methode oder irgendeiner anderen Methode zur Spurerkennung zu einem vollständigen Track kombiniert (Abbildung 4.3).

Der große Vorteil dieser Methode ist ihre Geschwindigkeit im Vergleich zu Methoden, die alle Punkte mehr oder weniger direkt verwenden. Sie ist deshalb geeignet für Detektoren mit einer hohen Punktdichte. Außerdem kann bei Driftkammern das Problem der Spiegelpunkte auf der Trackelement-Ebene gelöst werden. Ursprünglich wurde sie für die Analyse von Blaskammerbildern verwendet. Anwendung fand sie bei der

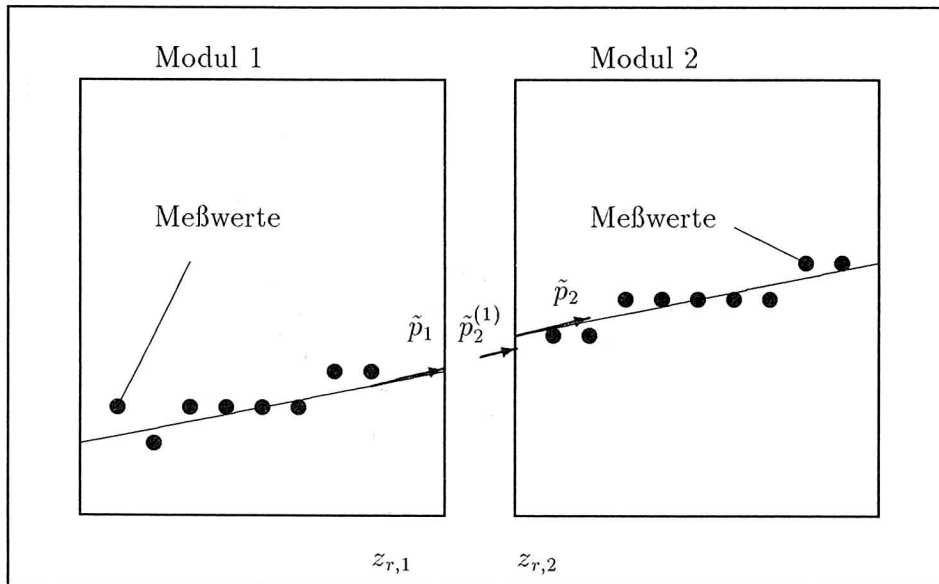


Abbildung 4.3: Die Track-Element-Methode: Die Track-Parameter  $\tilde{p}_1$  werden zu der neuen Referenzfläche propagiert.  $\tilde{p}_1^{(1)}$  und  $\tilde{p}_2$  werden als direkte Messungen der wahren Trackparameter an der Referenzfläche  $z_{r,2}$  betrachtet.

JADE Driftkammer am DESY [OLS80]. Eine ähnliche Methode wurde beim UA1 Detektor am CERN benutzt [PIM85].

## 4.6 Globale Methoden

Einen ganz anderen Weg gehen die globalen Methoden, die auch auf dem größeren Gebiet der *Cluster-Analyse* Verwendung finden. In diesem Fall werden alle Punkte in der gleichen Art und Weise behandelt.

### 4.6.1 Die Histogramm-Methode

Bei den Histogramm-Methoden benutzt man Funktionen der Punktkoordinaten. Die Werte dieser Funktionen werden in ein Histogramm eingetragen. Die Tracks sollten dann *Cluster* oder *Peaks* in diesen Histogrammen bilden. Diese sollten dann einfacher zu finden sein als die Tracks in

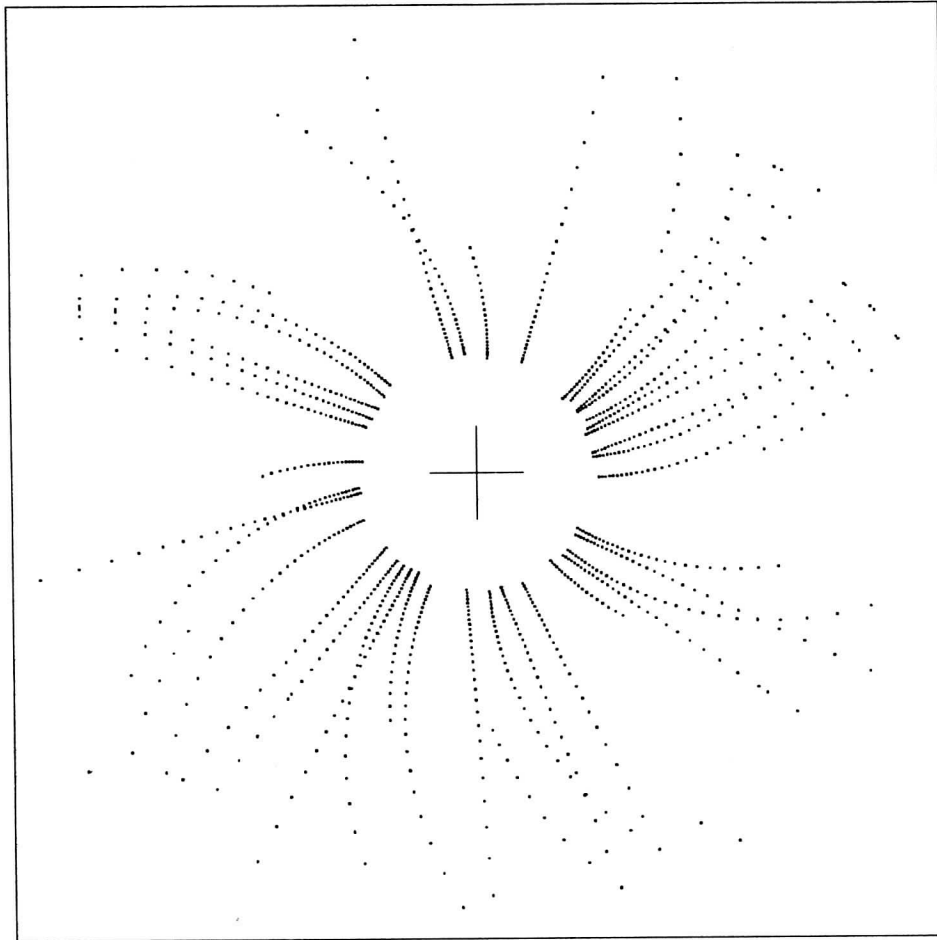


Abbildung 4.4: Die Kreisbögen nach der konformen Abbildung: Die ursprünglichen Kreisbögen sind in Abbildung 4.1 zu sehen. Der Abstand einer Geraden zum Koordinatenursprung ist proportional zur Krümmung des ursprünglichen Kreisbogens.

dem ursprünglichen Punktemuster, dem *Hit-Pattern*.

Ein Beispiel für eine solche Methode ist die *inverse konforme Abbildung* [HAN87, HAN88], die man anwenden kann, um kreisförmige Teilchenspuren, die alle durch den Ursprung (= primärer Wechselwirkungspunkt) verlaufen, zu finden:

$$u = \frac{x}{x^2 + y^2}, \quad (4.2)$$

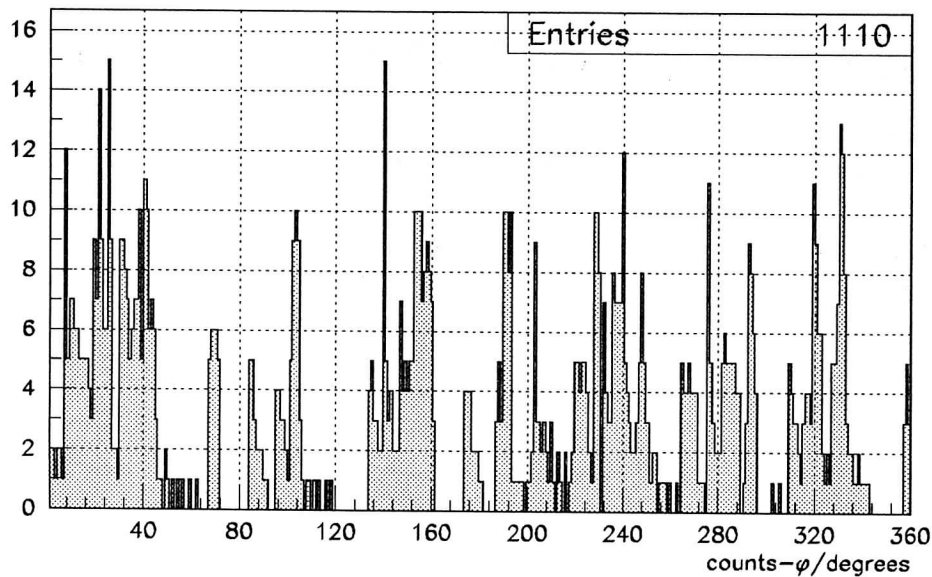


Abbildung 4.5:  $\phi$ -Histogramm der Tracks in Abbildung 4.4. Die Spuren mit hohem Transversalimpuls  $p_T$  bilden *Cluster* oder *Peaks*, die leicht gefunden werden können.

$$v = \frac{y}{x^2 + y^2}. \quad (4.3)$$

Diese Transformation erzeugt Geradenstücke in der  $uv$ -Ebene (Abbildung 4.4). Der kürzeste Abstand vom Ursprung ist dabei gegeben durch

$$d = \frac{1}{2R}, \quad (4.4)$$

mit  $R = \text{Trackradius}$ . Die Histogramm-Methode, angewendet auf die konforme Abbildung, toleriert einen gewissen Abstand der Spur vom Ursprung und wird die Spuren immer noch korrekt auffinden (Abbildung 4.5). Diese Methode wurde am CERN im Experiment R807 angewendet [DAH79].

#### 4.6.2 Das Template Matching

Diese Methode erfordert eine Liste aller möglichen Tracks und kann daher nur angewendet werden, wenn diese Zahl nicht zu hoch wird. In diesem Fall kann eine binäre Suche oder ein Hashalgorithmus sehr schnelles

000000000	000000000	000000000	000000000
0001111000	0000001000	0001111000	0001111000
0010000100	0000011000	0010000100	0010000100
0100000010	0000101000	0100000100	0000000100
0100000010	0001001000	0000000100	0000000100
0100000010	0010001000	0000000100	0000000100
0100000010	0000001000	0000001000	0000001000
0100000010	0000001000	0000010000	0000001000
0100000010	0000001000	0000100000	0000000100
0100000010	0000001000	0001000000	0000000100
0100000010	0000001000	0010000000	0000000100
0010000100	0000001000	0111111100	0010000100
0001111000	0000001000	0000000000	0001111000
0000000000	0000000000	0000000000	0000000000

Tabelle 4.1: Ein Beispiel für template matching: Templates für die vier Ziffern 0, 1, 2 und 3. Sie werden dargestellt durch eine Verteilung der Einsen in einem 10 mal 14 Gitter mit Nullen.

Trackfinding liefern. Diese Methode wurde für den MARK III Detektor am SPEAR [BEC85] verwendet.

Jede Kombination von Drähten, die gefeuert haben, erzeugt ein bestimmtes Muster (*template*). Die Unterteilung des Detektors in Zellen führt zu einer kleinen Zahl von vernünftigen Kombinationen. Bereiche, wo Tracks nahe zusammenkommen, führen zu Konfusion und müssen mit *klassischen* Methoden (Kombinatorik zum Beispiel) aufgelöst werden.

Diese Methode ist besonders geeignet für zylindrische Driftkammern, da die Einteilung des Detektors in Zellen eine natürliche Basis für diesen Algorithmus bietet.

Im Falle des MARK III Detektors funktioniert die Methode im wesentlichen folgendermaßen: Zu Beginn wird eine *cell image matrix* gefüllt. Eine Kombination von Zellen, die gefeuert haben, wird dann mit diesem *Dictionary* verglichen. Diese Methode ist global, außer in Regionen, wo es einen Überlap der Tracks gibt.

*Template matching* wird oft als schnelle Trigger-Logik implementiert, so zum Beispiel beim CELLO-Detektor am DESY [BEH81].

Für das Experiment E711 am Fermilab wurde ein template mat-

ching Algorithmus auf einem schnellen Vektor-Prozessor implementiert [GEO86].

Charakteristisch ist, daß die Koordinaten eines Objektes nur die Werte 0 oder 1 annehmen können, wie in dem Beispiel in Abbildung 4.1. Dort werden die vier Zahlen 0, 1, 2 und 3 gezeigt, die auf ein Gitter abgebildet wurden. Die Ziffern 0 und 1 zeigen die Grauwerte Weiß und Schwarz an.

In ähnlicher Weise ergibt die Messung eines Objektes einen Vektor  $x$ . Wird die Klasse durch  $m$  Prototyp-Templates  $y_m$  beschrieben, muß ein Vergleich durchgeführt werden, um das Template zu finden, das am besten paßt. Diese Prozedur wird *template matching* genannt und besteht einfach aus einer Vektormultiplikation

$$n_m = y_m^T \cdot x. \quad (4.5)$$

$m$  läuft dabei über alle Templates. Gruppiert man alle template Vektoren in eine Matrix  $Y$ , reduziert sich die ganze Prozedur zu einer einzigen Matrixmultiplikation:

$$n = Y^T x. \quad (4.6)$$

Die maximale Komponente von  $n$  zeigt die beste Übereinstimmung an.

### 4.6.3 Der Minimum-Spanning-Tree

Um die *Minimum-Spanning-Tree-Methode* zu verstehen, muß man einige Begriffe aus der *Graphentheorie* kennen. Ein *Graph* besteht aus *Knoten* und *Kanten*. Die Knoten können zum Beispiel Punkte im Ortsraum sein, die Kanten könnten durch die Verbindungslinien dargestellt werden. Ein *Baum* beziehungsweise *tree* ist ein Graph ohne Schleifen. Ein *spanning tree* (Abbildung 4.6) ist ein zusammenhängender Graph ohne Schleifen. Ein *minimum spanning tree* ist ein Baum in einem Graphen, für den die Summe der Kantengewichte minimal für alle zusammenhängenden Bäume in diesem Graphen ist.

Bäume für ein Ereignis wie in Abbildung 4.1 kann man konstruieren, indem man die mit dem Detektor gemessenen Koordinaten, die sogenannten *Hits*, mit den Knoten identifiziert und den Abstand zwischen Paaren von Hits mit den Kantengewichten. Vereinzelte Hits erscheinen für gewöhnlich als Knoten mit nur einem Nachbar und können daher leicht eliminiert werden. Tracks können stückweise gefunden und zu vollständigen Tracks kombiniert werden, indem man Ähnlichkeitskriterien verwendet, wie zum Beispiel gemeinsame Punkte, ähnliche Steigung oder Krümmung.

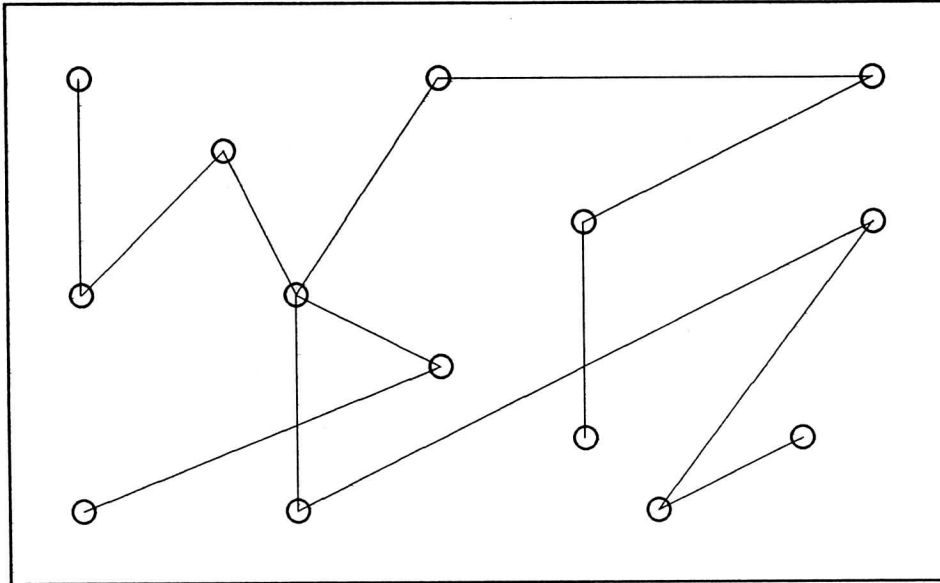


Abbildung 4.6: Ein spanning tree: Jeder Knoten ist mit mindestens einem anderen Knoten verbunden. Dies ist äquivalent zu der Tatsache, daß kein geschlossener Pfad (Loop) existiert.

Eine Variation dieser Methode wurde für den TASSO Detektor am DESY verwendet [CAS81]. Es konnte gezeigt werden, daß diese Methode effizienter als die Road-Methode ist. Das Basiselement in diesem Algorithmus ist nicht ein Punkt, sondern ein Punktepaar. Dieses Paar hat einen gewissen Abstand und eine bestimmte Richtung. Diese Paare werden zu einem Graphen kombiniert, falls sie gemeinsame Punkte und ähnliche Richtung haben. Die zweite Bedingung ist besonders effektiv, um Spiegelpunkte in einer Driftkammer (Links-Rechts-Ambiguität) zu verwerfen.

Auf diese Weise werden Track-Segmente mit einer minimalen Länge konstruiert. Diese erlauben es, Track-Parameter (Kreisradius und Kreis-mittelpunkt) zu bestimmen. Diese Track-Segmente werden dann zu vollen Tracks gruppiert, indem man Ähnlichkeitskriterien basierend auf den Segment-Parametern verwendet.

## Kapitel 5

# Einführung der Hough–Transformation

Die *Hough–Transformation* (HT) ist eine seit Jahrzehnten bekannte, sehr leistungsstarke Methode zur Parameterextraktion.

Sie wird bei vielen Bildverarbeitungsproblemen [DUD72, KIM75] wegen ihrer Vorteile angewendet:

- Die Anzahl der Operationen ist proportional zur Anzahl der Punkte.
- Sie ist robust gegen Untergrund.
- Sie ist sensitiv auch auf nur teilweise vorhandene Objekte.

In der Hochenergiephysik ist sie zum ersten Mal auf digitalisierte Streamerkammerbilder [PUE88] angewendet worden. Dort wurden lokale Methoden in Regionen hoher Track–Dichte als problematisch angesehen, und man wandte sich globalen Methoden zur Spurerkennung zu. Sie wurde auch schon auf Driftkammerdaten in der Hochenergiephysik [NOG86, PER86] angewendet. Anfänge, diese Methode auch auf Jet–Kammern [KLE87] anzuwenden findet man in [SCH88] und [SCH89].

Zunächst soll die HT an einem Beispiel erklärt werden: Wie kann man Geradenstücke in der Ebene mit der HT finden?

Man hat einen Bildraum mit einem Koordinatensystem  $XY$  (Abb. 5.1). Dann hat man einen quantisierten Parameterraum. Dabei stellt sich die Frage, wie man die Quantisierung macht. Sie hängt von der Genauigkeit ab, mit der man die Geradenparameter bestimmen will. Dabei gehen die Fehlern der Punkte im Bildraum ein. Der Parameterraum kann durch ein zweidimensionales Array dargestellt werden, einen *Akkumulator*.

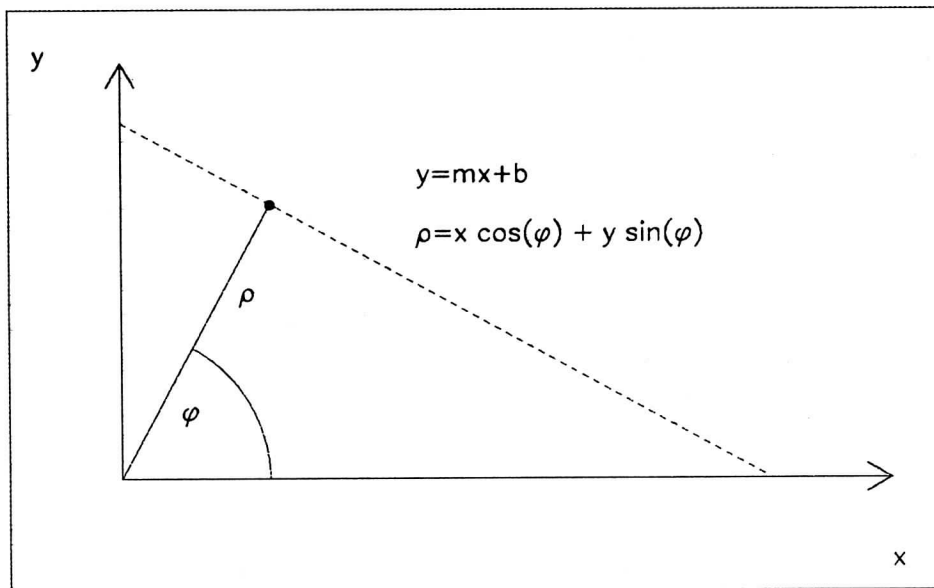


Abbildung 5.1: Die Hough-Transformation für ein Geradenstück

Im Bildraum hat man eine Punktmenge gewisser Dichte. Und man hat ein Modell: Die Punkte sollen auf einer Geraden liegen, sie sollen kollinear sein.

Um die Lagen der Geraden zu bestimmen, kann man auch verschiedene Koordinatensysteme verwenden. Oft sind kartesische Koordinaten von Vorteil, d.h. die Steigung  $m$  und der Ordinatenabschnitt  $c$ . Die Punkte  $(x, y)$ , die auf der Geraden liegen, erfüllen dann die Gleichung

$$y = mx + c. \quad (5.1)$$

Unter Umständen sind aber Polarkoordinaten zu bevorzugen, d.h. den kürzesten Abstand  $\rho$  der Geraden vom Koordinatenursprung und den Winkel  $\phi$  dieser Strecke gegen die  $x$ -Achse. Die Punkte  $(x, y)$  erfüllen dann die Gleichung

$$\rho = x \cos(\phi) + y \sin(\phi). \quad (5.2)$$

Die Abbildung des Bildraumes in den Parameterraum heißt HT. Eine solche Abbildung soll nun für das Beispiel der Geraden konstruiert werden. Man kann dazu folgendermaßen vorgehen:

Eine Möglichkeit wäre, die Geradengleichung 5.2 als Abbildungsgleichung zu benutzen. Man nimmt einen Punkt  $(x_1, y_1)$ , setzt ihn in die Gleichung ein und geht gemäß der gewählten Quantisierung alle Werte

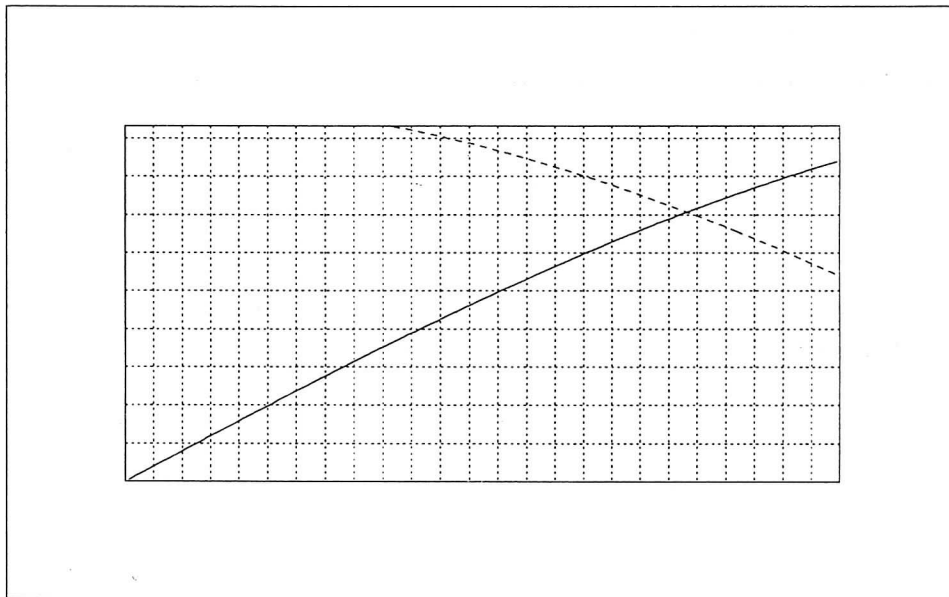


Abbildung 5.2: Parameterraum für ein Geradenstück

für  $\phi$  durch. Man berechnet den dazugehörigen Wert  $\rho$ . Dieser fällt in eine der Gitterzellen, den *Bins*. Der Inhalt dieser Bin wird um 1 erhöht. Ein *hit* im Bildraum erzeugt also im Parameterraum als Signal eine Kurve. Diese Implementation ist ein Beispiel für eine *1-to-many Methode*.

Bildet man einen zweiten Punkt auf die gleiche Weise ab, gibt es unter Umständen einen Schnittpunkt. Diese Bin hat einen höheren Eintrag als die Bins in der Umgebung. Auf diese Weise ergibt sich also ein kleiner Peak. Dieser Peak zeigt den Parametersatz für das Geradenstück an, auf dem die beiden Punkte liegen. Im Prinzip ist das die graphische Lösung eines Gleichungssystems aus zwei Gleichungen mit zwei Unbekannten.

Führt man diese Prozedur für alle Punkte aus, so zeigt die Zelle mit dem größten Eintrag, also dem höchsten Peak, die Stelle an, wo sich die meisten Kurven geschnitten haben. Auf einer Geraden mit diesen Parametern liegen also die meisten Punkte.

Die HT transformiert also das Problem, das Geradenstück zu finden, in das Problem, das signifikante Maximum im Akkumulator-Feld zu finden. Die Hough-Transformation erfordert also eine Methode diese Peaks zu finden.

Die 1-to-many Methode hat den Vorteil, linear zu sein. Der Inhalt des Akkumulators, das *Hough-Feld*, ergibt sich als Superposition der Kurvensignale. Jeder Bildpunkt wird in der gleichen Art und Weise behandelt.

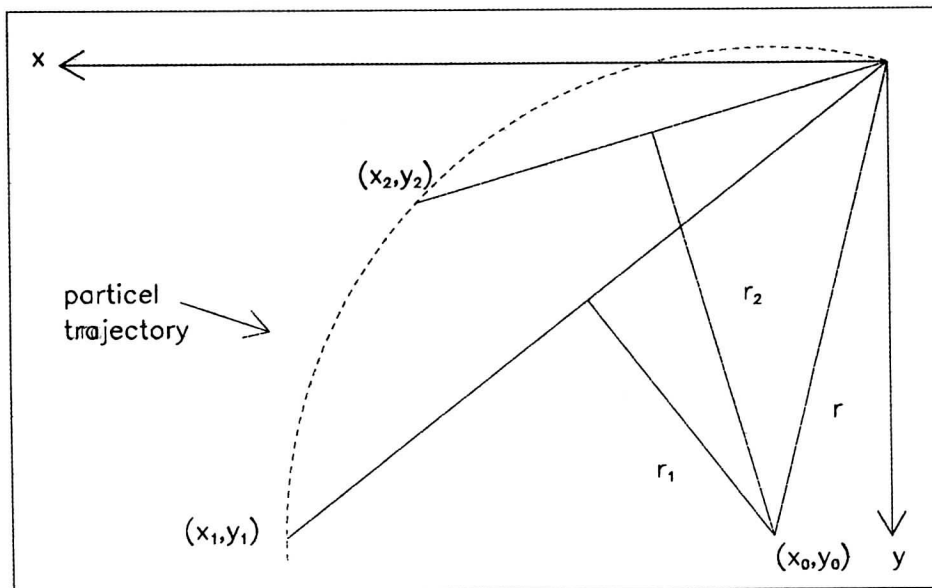


Abbildung 5.3: Die Hough-Transformation, um Kreise zu finden

Diesbezüglich ist diese Art der HT leichter zu verstehen und übersichtlicher. Der Rechenaufwand steht in direkter Beziehung zur Feinheit der Quantisierung des Parameterraumes. Das ist ein Nachteil bei großen Akkumulatorfeldern. Nachteilig ist auch der hohe Background.

Dieses Verfahren kann man leicht verallgemeinern: Sei irgendeine parametrisierbare Kurve gegeben. Hier ein allgemeines zweidimensionales Beispiel mit Parametern  $a$  und  $b$ :

$$f(x, y, a, b) = 0. \quad (5.3)$$

Hat man zwei feste Punkte,  $p_1 = (x_1, y_1)$  und  $p_2 = (x_2, y_2)$ , und setzt diese ein, so erhält man zwei verschiedene implizit gegebene Kurven im Parameterraum.

$$f(x_1, y_1, a, b) = 0. \quad (5.4)$$

$$f(x_2, y_2, a, b) = 0. \quad (5.5)$$

$a$  hängt von  $b$  funktional ab. Liegen die Punkte  $p_1$  und  $p_2$  auf der Kurve  $f$ , so haben diese beiden Kurven einen Schnittpunkt.

Man kann nun jeden Punkt gleichartig behandeln, also unabhängig von anderen Punkten und Kurvensignale im Akkumulator zulassen (*1-to-many Methode*), oder diesen Schnittpunkt ausrechnen und nur diesen Schnittpunkt abspeichern (*many-to-1 Methode*).

Im Falle eines Kreises sieht das so aus:

$$f(x, y, a, b) = (x - a)^2 + (y - b)^2 - r^2 = 0. \quad (5.6)$$

Nimmt man an, der Kreis geht durch den Ursprung, dann vereinfacht sich die Gleichung etwas, denn dann gilt

$$a^2 + b^2 = r^2. \quad (5.7)$$

Also:

$$f(x, y, a, b) = x^2 - 2ax + y^2 - 2by. \quad (5.8)$$

Man setzt  $p_1$  und  $p_2$  ein und erhält so zwei Geraden im Parameterraum:

$$\begin{aligned} p_1 &: b = q_1 a + s_1 \\ p_2 &: b = q_2 a + s_2 \end{aligned} \quad (5.9)$$

Den Schnittpunkt dieser beiden Geraden ausrechnen heißt, dieses Gleichungssystem lösen. Man erhält so die Parameter des Kreises, auf dem diese beiden Punkte liegen:

$$a = \frac{s_1 - s_2}{q_2 - q_1} \quad (5.10)$$

$$b = q_1 a + s_1 \quad (5.11)$$

Geometrisch sieht das folgendermaßen aus (Abb. 5.3): Die Annahme, die unserem Ansatz zugrundeliegt ist, daß die Tracks in der XY-Projektion Kreise sind, die alle vom Target kommen. Das Target wird als primärer Wechselwirkungspunkt, als *Common Vertex*, angenommen. Dabei wird dieser Vertex gleichzeitig als Koordinatenursprung gewählt.

Die Idee ist nun, jeden Hit mit dem Vertex zu kombinieren. Jeder Hit wird mit dem Vertex durch eine Gerade verbunden. Auf dieser Geraden wird die Mittelsenkrechte errichtet. Auf diese Weise definiert jeder Hit eine Gerade.

Man legt dann ein Gitter auf die XY-Ebene. Das Gitter entspricht einem zweidimensionalen Feld, dem Akkumulator. Man kann nun wieder diese ganze Gerade in den Akkumulator eintragen. Das ist aber bei einem großen Akkumulator sehr aufwendig. In diesem Fall bietet es sich an, den Schnittpunkt mit einer anderen Geraden ausrechnen und nur diesen Schnittpunkt eintragen. Das wäre ein Beispiel für eine *many-to-1 Methode*.

Man zählt dann die Punkte in einer Gitterzelle, einer *bin*, und erhöht entsprechend den Eintrag im Akkumulator. Auf diese Weise entsteht das Histogramm.

Vorteil der many-to-1 Methode ist:

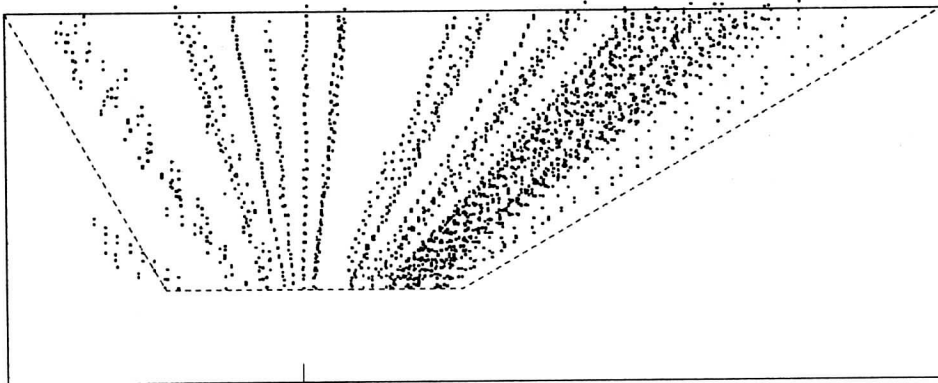


Abbildung 5.4: rz-Projektion des Ereignisses in der CDC von Abbildung 5.5. Die Auflösung des Detektors ist für die z-Koordinate um eine Größenordnung schlechter als für die x- und y-Koordinate.

- weniger Untergrund,
- der Aufwand ist unabhängig von der Akkumulatorgröße.

Nachteil ist, daß der Rechenaufwand quadratisch mit der Zahl der Punkte ansteigt.

Im idealen Fall sieht das so aus: Alle Hits, die zum gleichen Track gehören, erzeugen im Histogramm in der gleichen Bin einen Schnittpunkt. Auf diese Weise entsteht im Histogramm ein Peak. Falsche Kombinationen erscheinen nur als Untergrund im Akkumulator-Feld.

Das Problem, die Spuren in der CDC zu finden, wurde damit transformiert in das Problem, die Peaks im Histogramm zu finden.

Abbildung 5.5 zeigt ein typisches Punktemuster in der XY-Ebene (senkrecht zur Strahlachse) der CDC. Abbildung 5.4 zeigt ein typisches Punktemuster in der RZ-Ebene (parallel zur Strahlachse) der CDC.

Wendet man auf diese Punktemenge die Hough-Transformation an, so ergibt sich ein Histogramm für die Mittelpunktskoordinaten  $(x_c, y_c)$  der Kreisspuren wie es zum Beispiel in Abbildung 5.6 zu sehen ist. Die hohen Peaks korrespondieren mit den Tracks.

Als Parameter wurden die Mittelpunktskoordinaten eines Kreises durch den Koordinatenursprung gewählt. Außerdem wird aus technischen Gründen nur ein Viertel des Akkumulators gezeigt. Die rechte obere Ecke ist die Strahlrohrposition. Man sieht, daß dort der Untergrund der Hough-Transformation sehr stark ansteigt. Die strahlenförmigen Gebilde, die davon ausgehen, sind ebenfalls kombinatorischer Untergrund: Ein

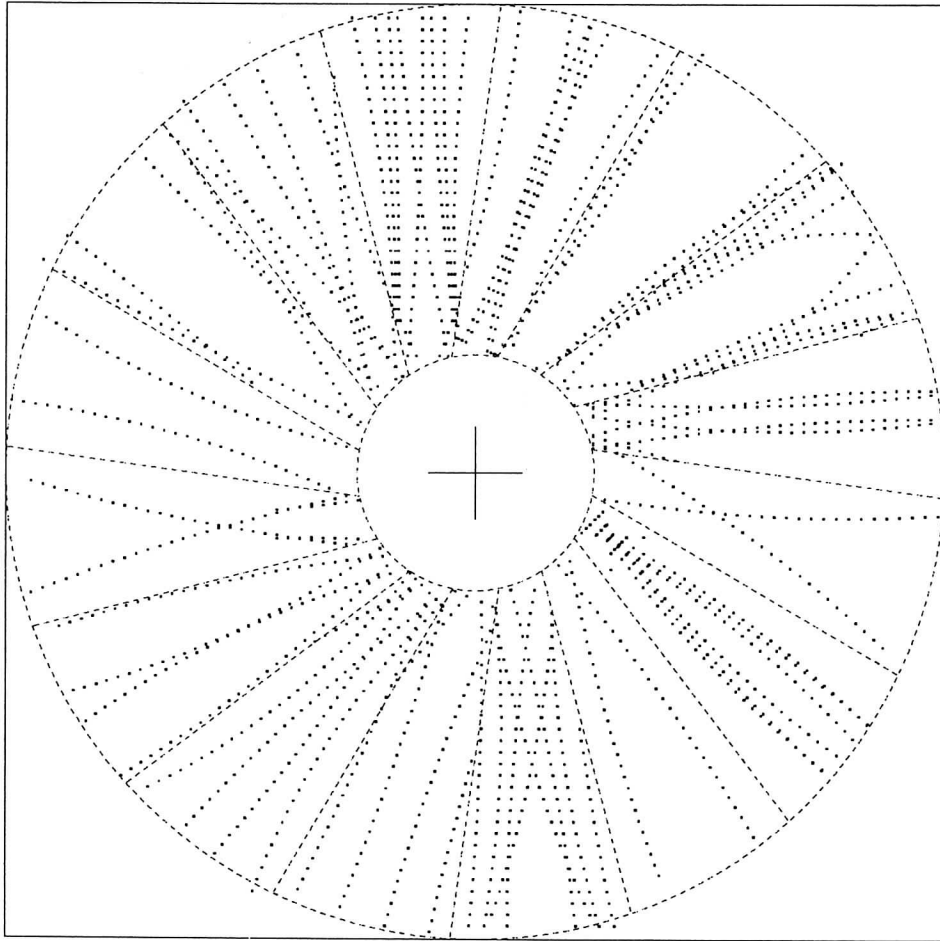


Abbildung 5.5: xy-Projektion eines Ereignisses in der CDC: Au+Au bei 1 GeV/u mit FREESCO simuliert (siehe Kapitel über das FOPI Simulationsprogramm).

falscher Hit kombiniert mit den Hits eines wahren Tracks ergibt diese korrelierten Gebilde im Akkumulator.

Ein Programm muß die richtigen Peaks, die mit den Tracks korrespondieren, durch Filtern im Histogramm extrahieren.

Die Fehler in den Positionen der Hits haben einen wesentlichen Einfluß auf die Gestalt dieser Peaks im Hough-Feld. Man kann davon ausgehen, daß die Fehler der Hits in x- und y-Richtung normalverteilt und voneinander unabhängig sind. Mit einer Fehlerfortpflanzungsrechnung für die many-to-1 Methode kann gezeigt werden [FRO88], daß die Peaks im

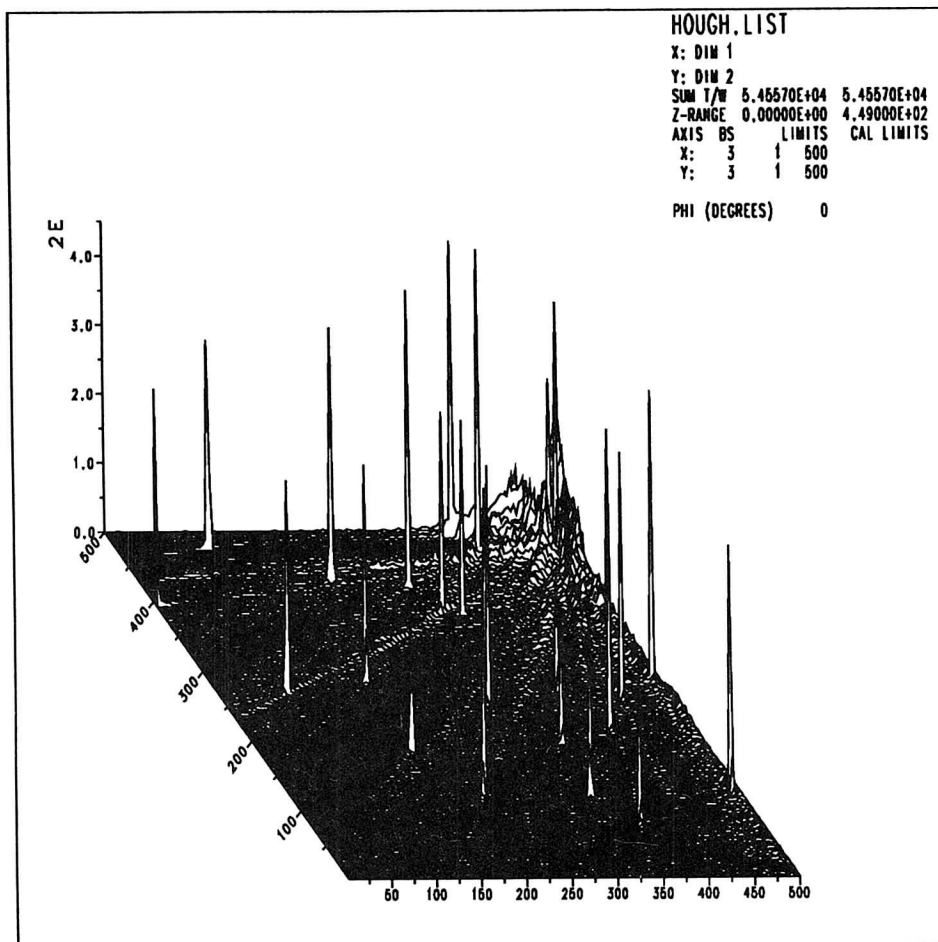


Abbildung 5.6: Akkumulator für eine Verteilung der Hits in der CDC wie es zum Beispiel in Abbildung 5.5 gezeigt wird. Die hohen Peaks im Akkumulator entsprechen den Tracks in dieser Punktmenge.

Hough-Feld dann ebenfalls gaußförmig sind, allerdings mit elliptischen Niveaulinien. Abbildung 5.7 zeigt die Gestalt der Peaks im einzelnen.

Transformiert man das Zentrum der Gaußverteilung auf den Nullpunkt, so hat ein Peak im Hough-Feld also die allgemeine Form

$$f(x, y) = \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \exp\{-h(x, y)\}. \quad (5.12)$$

$(x, y)$  ist dabei eine Position im Hough-Feld,  $\sigma_x$  und  $\sigma_y$  sind die Varianzen der zweidimensionalen Gaußglocke,  $\rho$  ist der Korrelationskoeffizient.

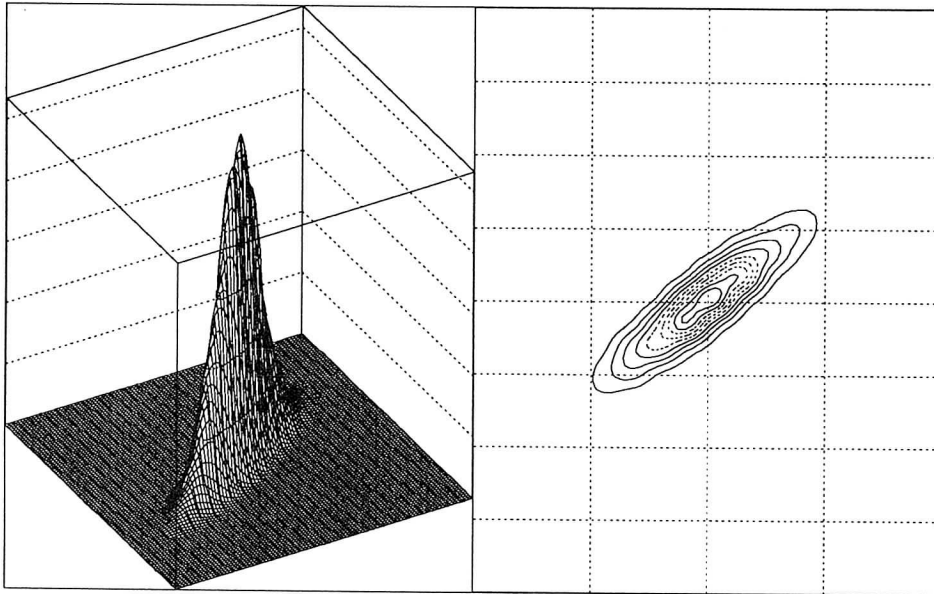


Abbildung 5.7: Verteilung der Schnittpunktefehler. Die Niveaulinien sind Ellipsen.

Bezeichnet man für Punkt  $i$  die Varianz in  $x$ -Richtung mit einem ungeraden Index ( $\sigma_{2i-1}$ ) und die Varianz in  $y$ -Richtung mit einem geraden Index ( $\sigma_{2i}$ ), so ist

$$\rho = \frac{\sum_{i=1}^n \sigma_{2i-1,2i}}{\sqrt{\sum_{i=1}^n \sigma_{2i-1}^2 \sum_{i=1}^n \sigma_{2i}^2}}. \quad (5.13)$$

Für  $n$  Punkte ergibt sich für den Exponent  $h(x,y)$  die Gestalt:

$$h(x,y) = \frac{1}{1-\rho^2} \left\{ \frac{x^2}{\sum_{i=1}^n \sigma_{2i-1}^2} - 2\rho \frac{xy}{\sqrt{\sum_{i=1}^n \sigma_{2i-1}^2 \sum_{i=1}^n \sigma_{2i}^2}} + \frac{y^2}{\sum_{i=1}^n \sigma_{2i}^2} \right\}. \quad (5.14)$$

Damit hat man die Gaußverteilung für  $n$  Hits einer Spur vorliegen. Die Höhe dieser Glockenkurve, die dem Normierungsfaktor der Verteilungsfunktion entspricht, ist ein Maß für die Güte der Schnittpunkte, d.h. falls die Hits große Fehler aufweisen, dann ist die Gaußverteilung flach. Setzt man  $h(x,y)=K$ , wobei  $K$  eine beliebige positive Konstante ist, so erhält man die Linien konstanter Wahrscheinlichkeit. Sie verlaufen

ellipsenförmig. Die Halbachsen der Ellipsen sind:

$$p_1^2 = \frac{\sigma_x^2 \sigma_y^2 (1 - \rho^2)}{\sigma_y^2 \cos^2 \alpha - \sigma_x \sigma_y \rho \sin 2\alpha + \sigma_x^2 \sin^2 \alpha}, \quad (5.15)$$

$$p_2^2 = \frac{\sigma_x^2 \sigma_y^2 (1 - \rho^2)}{\sigma_x^2 \cos^2 \alpha - \sigma_x \sigma_y \rho \sin 2\alpha + \sigma_y^2 \sin^2 \alpha}. \quad (5.16)$$

Dabei ist  $p_1$  die längere und  $p_2$  die kürzere Halbachse.  $\alpha$  ist der Winkel von  $p_1$  gegen die x-Achse:

$$\alpha = \frac{1}{2} \arctan\left(\frac{2\rho\sigma_x\sigma_y}{\sigma_x^2 - \sigma_y^2}\right). \quad (5.17)$$

# Kapitel 6

## Die adaptive Hough–Transformation

*Iteration ist menschlich, Rekursion ist göttlich.*

– ANON

### 6.1 Die Idee der adaptiven HT

Die Frage der Quantisierung des Parameterraums tritt bei Daten mit Fehlern in den Mittelpunkt. Die Quantisierung des Parameterraumes wählt man gemäß der Auflösung für die Parameter, die erreicht werden soll. Andererseits spielen die Fehler der Hits eine entscheidende Rolle. Der Einfluß dieser Fehler kann durch eine Fehlerfortpflanzungsrechnung bei der HT bestimmt werden.

Wenn man die Größe einer Gitterzelle aber nach diesen Kriterien wählt, führt das im allgemeinen zu verhältnismäßig großen Bins. Die Auflösung reicht dann unter Umständen nicht aus, um zwei Peaks, respektive Tracks, die entsprechend dicht beieinanderliegen, zu trennen.

Außerdem funktioniert die Zuordnung 1 Bin – 1 Track nicht mehr. Man muß also wieder zu einem *Backmapping* greifen, einer inversen Abbildung vom Parameterraum in die ursprüngliche Punktmenge. Dazu muß also die Auflösung wesentlich gesteigert werden. Die Idee war daher: Die Daten sollten das Binning selbst regeln. Dies führte zur *adaptiven Hough–Transformation* (AHT) [ILL87].

In der gewöhnlichen HT wird das Problem, die Peaks im Hough–Feld zu finden, durch *exhaustive search* in einem genügend fein quantisierten Parameterraum gelöst.

Ein großer Akkumulator ist aber aus mehreren Gründen nicht

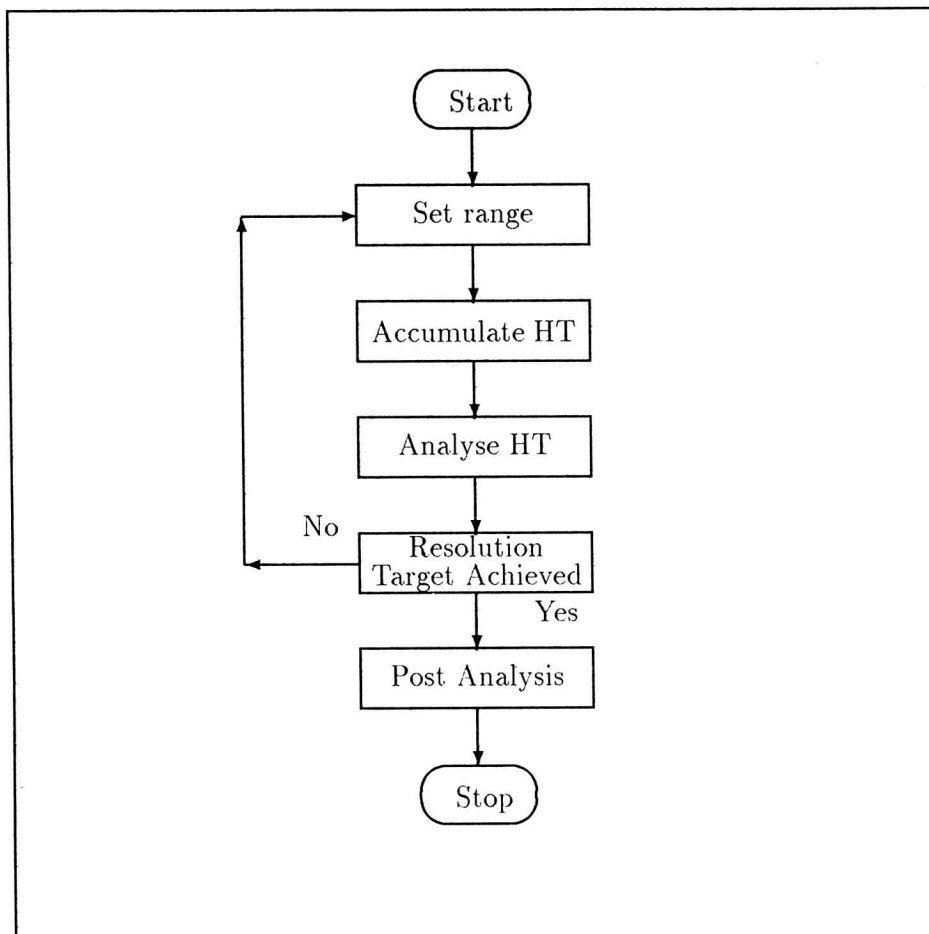


Abbildung 6.1: Flußdiagramm für die AHT

wünschenswert:

- Er nimmt einen großen Speicherplatz ein. Wenn  $k$  Parameter bestimmt werden müssen und der Parameterbereich in  $\alpha$  Intervalle zerfällt, dann hat man  $\alpha^k$  Elemente im Akkumulator.
- Viele Zellen müssen getestet werden.
- Man produziert hauptsächlich Untergrund.

Interessant sind aber nur die Zellen in der unmittelbaren Umgebung der Peaks. Die AHT ist eine Methode, die Peaks zu lokalisieren, oh-

ne den Untergrund im Detail auszurechnen. Sie wendet eine intelligente Histogramm- und Such-Strategie an. Diese Methode benutzt einen kleinen Akkumulator und zerlegt den Parameterbereich, der momentan interessiert, in diese wenigen Intervalle.

## 6.2 Flußdiagramm der AHT

Es soll nun die AHT für einen kleinen Parameterraum beschrieben werden (Abb. 6.1).

Zuerst wird die HT ausgeführt. Die Information im Akkumulator wird dann benutzt, um den Parameterbereich umzudefinieren, so daß interessante Bereiche mit größerer Auflösung untersucht werden können. Deshalb heißt die Methode auch adaptiv, weil sie sich auf die Information, die im Akkumulator enthalten ist, bezieht. Diese Iteration dauert solange, bis die Parameter mit der gewünschten Auflösung bestimmt sind.

Die bestimmten Parameter werden dann dazu benutzt, um mit einem Backmapping die Punkte zu bestimmen, die zu dem Objekt gehören. Diese Punkte werden aus der Datenbank, auf der das Programm arbeitet, herausgenommen, so daß der Zyklus für den nächsten Peak mit weniger Punkten arbeiten kann.

## 6.3 Das Binning

Eine der ersten Entscheidungen, die man bei der AHT treffen muß, ist die Größe des Akkumulators. Effizienz-Argumente würden zu einem Akkumulator führen, der so klein wie möglich ist:  $1 \times 2$  oder  $2 \times 2$ . Um aber den Parameterraum interpretieren zu können, braucht man mehr Struktur. Aus diesem Grunde wurde ein Akkumulator mit  $9 \times 9$  Zellen gewählt.

Um die Hough-Transformation in der *1-to-many Methode* auszuführen, ist es notwendig, die Parameterzellen zu identifizieren, die von der Kurve, die jedem Punkt zugeordnet ist, geschnitten werden. Im zweidimensionalen Fall ist das die Abbildung eines Punktes in eine Gerade im Parameterraum. Die Schnittpunkte dieser Linie mit dem Gitter können leicht ausgerechnet werden.

Für eine Parameterachse, die in  $\beta$  Intervalle unterteilt ist, gibt das  $\beta + 1$  Schnittpunkte (Abb. 6.2). Die Zellen auf jeder Seite des Schnittpunktes liegen auf der Linie, und es kann ein Eintrag in einem logischen Feld der gleichen Größe wie der Akkumulator gesetzt werden, um das

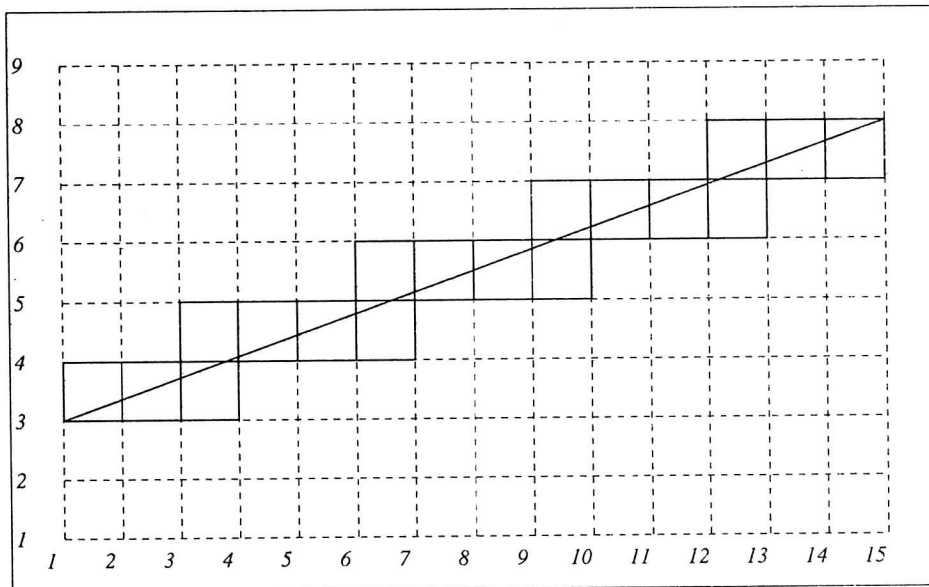


Abbildung 6.2: Binning für die AHT

anzuzeigen. Die Prozedur wird entlang der zweiten Parameterachse wiederholt. Auf diese Weise werden alle relevanten Zellen gekennzeichnet. Der Akkumulator wird dann inkrementiert, je nachdem, ob das logische Feld an dieser Stelle einen Eintrag hat. Die Akkumulation des logischen Feldes, das *Binning* also, erfordert die Berechnung von  $2 \cdot (\beta + 1)$  Gitterschnittpunkten für jeden Bildpunkt.

## 6.4 Das Clusterfinding mit dem Connected Components Algorithmus

Wenn der Parameterraum auf diese Weise populierte worden ist, muß er analysiert werden. Die signifikanten Maxima werden nach einem Verfahren bestimmt, das den Ideen von [LUM83] beziehungsweise [LSZ83] nachgebildet wurde:

Der *Connected Components Algorithmus* (CCA) ist eine Technik, binäre Bilder in *Regionen* aufzuteilen. Das resultierende Bild kann dann entsprechend den Wünschen des Benutzers weiter verarbeitet werden

Der CCA ist ein Operator, der ein binäres Eingabebild  $B$  in ein symbolisches Feld  $S$  transformiert. Das symbolische Bild besitzt dabei folgende Eigenschaften:

```

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
0 0 1 1 1 0 0 0 0 0 0 0 1 1 0 0 0 0 0
0 0 0 1 1 1 0 0 0 0 1 1 1 1 0 0 1 0 0
0 0 0 0 1 1 1 0 0 0 1 1 1 1 1 0 0 1 0 0
0 0 0 0 0 1 1 1 0 0 1 1 1 1 0 0 1 1 0 0
0 0 0 0 0 0 1 1 0 0 0 1 1 0 0 0 1 1 1 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1
0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 1 1 1 1
0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 1 0 0
0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0

```

Abbildung 6.3: Binäres Eingabebild für den Algorithmus, der die zusammenhängenden Regionen von 1-Werten identifizieren soll.

- 
- Alle Punkte in  $B$ , die den Wert 0 haben, bleiben 0 in  $S$ .
  - Jede *maximal zusammenhängende Teilmenge* von 1-Werten in  $B$ , ein sogenannter *Cluster*, wird mit einem *Label* versehen, einer bestimmten positiven ganzen Zahl.

Der CCA wird mit den Abbildungen 6.3 und 6.4 illustriert: Aus einem binären Eingabebild (Abbildung 6.3) wird ein symbolisches Ausgabebild (Abbildung 6.4) produziert.

Die Idee des CCA kann auf nichtbinäre Bilder verallgemeinert werden, falls es eine Eigenschaft gibt, aufgrund der jedem Bildpunkt eindeutig ein binärer Wert zugeordnet werden kann.

Sei beispielsweise ein Eingabebild, dessen Bildwerte aus der Menge  $\{1, 2, 3, 4, 5\}$  kommen, gegeben. Mit dem CCA ist man dann in der Lage, die maximal zusammenhängenden Menge von Bildpunkten, deren

```

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0
0 0 1 1 1 0 0 0 0 0 0 0 2 2 0 0 0 0 0
0 0 0 1 1 1 1 0 0 0 0 2 2 2 2 0 0 3 0 0
0 0 0 0 1 1 1 0 0 0 2 2 2 2 2 0 0 3 0 0
0 0 0 0 0 1 1 1 0 0 2 2 2 2 0 0 3 3 0 0
0 0 0 0 0 0 1 1 0 0 0 2 2 0 0 0 3 3 3 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 3 3 3
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 3 3 3
0 0 4 0 0 4 0 0 0 0 0 0 0 0 0 0 3 3 3 3
0 4 4 4 4 4 4 4 4 0 0 0 0 0 0 0 3 0 0
0 4 4 4 4 4 4 4 0 0 0 0 0 0 0 0 0 0 0
0 4 4 4 4 4 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 4 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 5 5 5 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 5 5 5 5 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 5 5 5 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 5 5 0 0 0 0 0 0

```

Abbildung 6.4: Dieses Bild erhält man als Ergebnis des *Connected Components Algorithmus*, wenn er auf das Eingabebild in Abbildung 6.3 angewendet wird.

Wert {4} ist, zu finden. Der CCA betrachtet das Eingabebild als binär, wobei der Wert {4} der 1 entspricht und die Werte {1, 2, 3, 5} der 0. Mit dem CCA erhält man also segmentierte Bilder. Auf jedem Teil kann dann separat weiter analysiert werden.

Das Ziel des CCA ist es, jeder maximal zusammenhängenden Gruppe von 1-Werten in einem binären Bild einen und genau einen Label zuzuordnen. Dies erreicht man dadurch, indem man jede Zeile in dem binären Bild von links nach rechts und von oben nach unten durchläuft. Die Einzelheiten werden an einem Ausschnitt aus einem binären Bild klar:

```

line r-1 x x B x x
line r   x C A x x

```

A, B, C sind die binären Werte, die untersucht werden, x steht für binäre Werte, die nicht betrachtet werden. Jeder Bildpunkt mit dem Wert 0 bleibt 0. Angenommen, Bildpunkt A in Zeile r ist 1. Man will dann diesem Bildpunkt einen Label zuordnen derart, daß seine Nähe zu B und C, die vorher mit einem Label versehen worden sind, berücksichtigt wird.

Es gibt mehrere mögliche Labels für A, je nachdem, welche Labels B und C schon haben. Haben B und C den Label 0, dann erhält A den nächsten freien Label. Hat C den Wert 0 und besitzt B schon einen Label, dann wird A der Label von B zugewiesen. Analog geht man vor, wenn B den Wert 0 hat und C schon ein Label zugeordnet wurde: A erhält den Label von C. Wenn beide, B und C, einen Label ungleich 0 besitzen, dann gibt es für A zwei Möglichkeiten:

- Besitzen B und C den gleichen Label, dann erhält A auch diesen Label.
- B und C weisen verschiedene Labels ungleich 0 auf. In diesem Fall verbindet Bildpunkt A zwei Cluster, die vorher als zwei verschiedene Regionen angesehen wurden.

Der zweite Fall wird in dem folgenden Beispiel illustriert:

#### Original Picture

line	pixel value
1	0 0 1 0 0 0 1
2	0 0 1 1 0 0 1
3	0 0 1 1 1 0 1
4	0 0 1 1 1 1 1

#### Processed Image

line	label value
1	0 0 1 0 0 0 2
2	0 0 1 1 0 0 2
3	0 0 1 1 1 0 2
4	0 0 1 1 1 1 A

Bildpunkt A verbindet die Regionen 1 und 2 und bekommt den kleineren der beiden Labels zugeordnet. Dazu wird noch die Äquivalenz von

Label 1 und 2 abgespeichert, so daß alle Bildpunkte mit dem Label 2 in diesem Bild eventuell zu Label 1 geändert werden können. Tabelle 6.1 zeigt noch einmal im Überblick die Labelzuordnung falls  $A = 1$ .

B	C	welcher Label
0	0	nächster Label
0	1	Label von C
1	0	Label von B
1	1	kleinerer Label

Tabelle 6.1: Labelzuordnung

Diese Methode ist sehr geradlinig mit Ausnahme des letzten Schrittes, wenn sich herausstellt, daß zwei Label äquivalent sind. Der Zeitaufwand der Prozedur hängt empfindlich von der Art und Weise ab, wie diese äquivalenten Label aufgelöst werden.

Die Standardmethode ist, zwei Vektoren einzuführen, wovon einer den Label der aktuellen Region enthält, der andere den zugehörigen kleinsten äquivalenten Label. Das Bild wird von oben nach unten durchlaufen, und alle äquivalenten Label werden in den beiden Vektoren gespeichert. Dann braucht man einen zweiten Durchgang, um jeden Label durch seinen kleinsten äquivalenten Label zu ersetzen.

Diese Standardmethode arbeitet zuverlässig, wird aber für große Bilder uneffektiv. Mit wachsender Bildgröße kann die Zahl verschiedener Regionen sehr groß werden. Für ein Bild der Größe  $2048 \times 2500$  Bildpunkte beträgt sie bereits 80000.

Da der Akkumulator in der hier diskutierten Anwendung klein ist und diese Prozedur nur ein Mal pro Iteration durchgeführt werden muß, ist der Aufwand gering und damit akzeptabel.

Der Algorithmus wird nun als eine präzise Sequenz von Operationen beschrieben. Um dies in konsistenter Weise ausführen zu können, wird die Notation von Tabelle 6.2 eingeführt.

Der Algorithmus sieht im einzelnen folgendermaßen aus:

Name	Erklärung
NLines	number of rows in the image
L	row index
NCOLUMN	number of columns
C	column index
I(L,P)	input value ( 0 or 1 ) at row L, column P
LABEL(L,P)	output value ( integer label ) at row L, column P
EQTABLE	equivalence table
A	the set of pixels on the current or previous line and adjacent to the pixel (L,P) that have input value 1 and are already labeled
M	the label initially assigned to pixel (N,P)
CREATE	a function that returns an empty equivalence table
ADJACENT	a function that determines if two pixels are adjacent
ISEMPTY	a function that determines if a set is empty
NEWLABEL	a function that returns the next chronological integer label that has not been used
ADD	a function that adds a pair to an equivalence table
RESOLVE	a procedure that takes in an equivalence table and constructs a partition of the labels in that table into their equivalence classes
EQUIVCLASS	a function that returns the equivalence class of a given label
MINLAB	a function that returns the minimum label in a given equivalence class, which is used to represent the class
LA	the set of labels of the current or previous line and adjacent to pixel (L,P) that are not 0

Tabelle 6.2: Variablendeklaration für den CCA

*Pass 1*

“create a single empty equivalence table“

EQTABLE := CREATE();

“go through each line L of the image“

**for** L:=1 to NLines **do**

**begin**

    “set all labels initially to 0“

**for** C:=1 to NColumn **do**

**begin**

```

LABEL(L,C) := 0;
“examine each pixel C on line L“
for C:=1 to NColumn do
  begin
    “disregard 0 labels“
    if I(L,P) ≠ 1 then NEXT endif;
    “find the set of adjacent labeled pixels“
    A := {(N, Q)|(I(N, Q) = 1)
    and (LABEL(N, Q) ≠ 0)
    and ((N = L) or (N = L - 1))
    and ADJACENT((N, Q), (L, P))};
    “choose the minimum label possible for the new pixel“
    if IEMPTY(A)
    then M := NEWLABEL()
    else M := min{LABEL(N, Q)|(N, Q) ∈ A} endif;
    LABEL(N,P):= M;
    “update equivalence table“
    for X ∈ {LABEL(N, Q)|(N, Q) ∈ A and X = M}
      call ADD((X,M)EQTABLE)
    end
  end
end

```

#### *Pass 2*

“Find equivalence classes. Each nonzero label is replaced with its smallest equivalent label.“

```
call RESOLVE(EQTABLE)
```

“relabel pixels based on equivalence classes“

```

for L:=1 to NLINES do
  for C:=1 to NColumn do
    begin
      if I(L,P) = 1
      then LABEL(L,P):=MINLAB(EQUIVCLASS(LABEL(L,P)))
      endif;
    end
  end
end

```

In den Abbildungen 6.5 und 6.6 wird eine typische Programmausgabe gezeigt.

0	0	0	0	0	39	19	8	0	0	0	0	0	0	0
0	0	0	0	0	451	34	14	0	0	0	0	0	0	0
0	0	0	0	0	28	49	20	0	0	0	0	0	0	0
0	0	0	0	0	3	84	44	5	0	0	0	0	0	0
0	0	0	0	1	0	110	57	4	0	0	0	0	0	0
0	0	0	0	0	3	93	143	11	0	0	0	0	0	0
0	0	0	0	0	1	27	713	27	0	0	0	0	0	0
0	0	0	0	0	0	4	590	40	0	0	0	0	0	0
0	0	0	0	0	0	0	785	223	0	0	0	0	0	0
0	0	0	0	0	0	0	398	603	0	0	0	0	0	0
0	0	0	0	0	0	0	0	65	0	0	0	0	0	0
0	0	0	0	0	0	0	5	990	2	0	0	0	0	0
0	0	0	0	0	0	0	1	723	14	0	0	0	0	0
0	0	0	0	0	0	0	0	450	116	0	0	0	0	0
0	0	0	0	0	0	0	0	158	223	1	0	0	0	0
0	0	0	0	0	0	0	0	15	700	12	0	0	0	0
0	0	0	0	0	0	0	0	6	216	23	0	0	0	0
0	0	0	0	0	0	0	0	4	218	17	5	0	0	0
0	0	0	0	0	0	0	0	2	124	17	11	0	0	0
0	0	0	0	0	0	0	0	12	66	9	9	1	0	0
0	0	0	0	0	0	0	0	35	23	488	12	5	0	0

Abbildung 6.5: Typisches Aussehen des Hough-Feldes

## 6.5 Der neue Parameterbereich

Nachdem die Peaks in ihrer Ausdehnung mit dem CCA bestimmt wurden, wird der *most significant peak* ausgewählt: Dazu wird die Dichte jedes Peaks bestimmt, d.h.

$$\frac{\text{Anzahl der Counts}}{\text{Anzahl der Zellen}} \quad (6.1)$$

Der Peak mit der größten Dichte wird ausgewählt. Die Ausdehnung dieses Peaks bestimmt den Parameterbereich für die nächste Iteration. Ein wichtiger Aspekt dabei ist, daß dies unabhängig in jeder Parameterdimension durchgeführt werden kann. Als Zentrum für den neuen Parameterbereich wird der Schwerpunkt des Peaks genommen. Wenn die Ausdehnung eines Parameterbereichs kleiner als 3 Zellen ist, wird er jedoch

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Abbildung 6.6: Herausgefilterter Peak von Abbildung 6.5

auf 3 Zellen gesetzt. Ansonsten hat man Probleme mit Peaks, die sich auf zwei Zellen aufteilen.

In zwei Fällen folgt die Berechnung der Parameterbereiche nicht dieser einfachen Strategie:

1. Immer wenn der Cluster kompakt in der Ausdehnung ist, aber an den Rand des Akkumulators stößt, wird angenommen, daß der Peak außerhalb des derzeitigen Parameterbereichs weitergeht. Die Parametergrenzen werden dann nur verschoben, die Größe einer Zelle bleibt unverändert.
2. Wenn ein Peak eine Linie über den ganzen Akkumulator ist und die Breite einer Zelle einnimmt, werden die Parameter nur in der dünnen Richtung geändert. Das führt zu einem gleichmäßiger popultierten und daher leichter zu interpretierenden Akkumulator.

## 6.6 Die Komplexität der AHT

Die Vorteile der AHT zeigt eine approximative Berechnung der Komplexität der AHT im Vergleich zur gewöhnlichen HT für zweidimensionale Parameterräume. Dabei wird die Annahme gemacht, daß die Auflösung bei jeder Iteration der AHT maximal erhöht wird.

Wenn man jeden Parameter mit  $\frac{1}{\alpha}$  ( $\alpha > 1$ ) seines vollen Bereichs auflösen will, benutzt man in der gewöhnlichen HT einen Akkumulator der Größe  $\alpha^2$ .

In der AHT hat man einen kleinen Akkumulator der Größe  $\beta^2$ ,  $\beta < \alpha$ . Der Speicherplatzvorteil ist also

$$R_s = \left(\frac{\alpha}{\beta}\right)^2. \quad (6.2)$$

Das kann ein sehr großer Quotient sein.

Der Zeitvorteil hängt vom Verhältnis  $\frac{1}{\gamma}$  ab, mit dem der Parameterbereich bei jeder Iteration verringert wird. Um die potentiellen Möglichkeiten der AHT zu zeigen, soll immer der gleiche Quotienten für beide Dimensionen und für alle Iterationen angenommen werden.

Die Anzahl der Iterationen  $n$ , bevor die AHT die gleiche Auflösung erreicht wie die gewöhnliche HT ist dann gegeben durch

$$\left(\frac{1}{\gamma}\right)^n = \frac{1}{\alpha}. \quad (6.3)$$

Also:

$$n = \frac{\log(\alpha)}{\log(\beta)}. \quad (6.4)$$

Wenn die Anzahl der Bildpunkte nur groß genug ist, wird der Hauptteil der Zeit für die Akkumulation verwendet. Bei der AHT hat man bei jedem Punkt  $2(\beta + 1)$  Gitterschnittpunkte zu berechnen.

Bei der gewöhnlichen HT würde man  $\alpha$  Werte für den abhängigen zweiten Parameter zu berechnen haben, wenn man die  $\alpha$  möglichen Werte für den unabhängigen ersten Parameter durchgeht.

Eine einzige Iteration der AHT ist also

$$\frac{\alpha}{2(\beta + 1)} \quad (6.5)$$

effizienter als die gewöhnliche HT.

Also ist die relative Effizienz, um das gleiche Objekt zu finden

$$R_c = O\left(\frac{\alpha}{2(\beta + 1)} \frac{\log(\gamma)}{\log(\alpha)}\right). \quad (6.6)$$

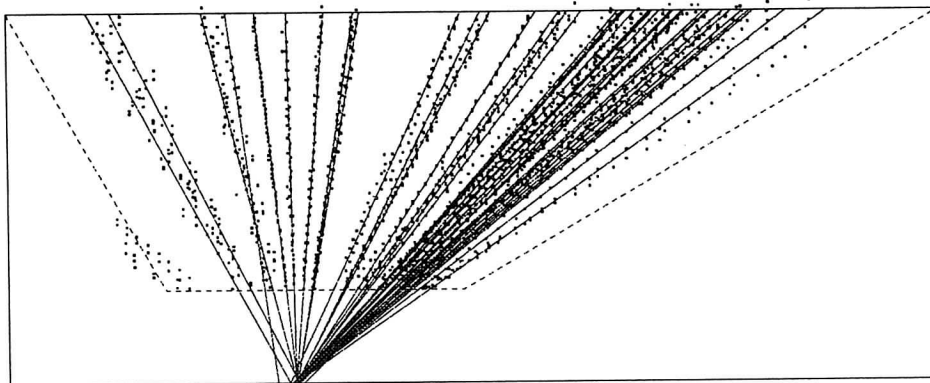


Abbildung 6.7: rz-Projektion eines Ereignisses in der CDC, die Auflösung des Detektors für die z-Koordinate ist um eine Größenordnung schlechter als in der xy-Projektion.

Ein Beispiel zur Komplexität zeigt die folgende einfache Rechnung:

$$\alpha = 1000$$

$$\beta = 9$$

$$\gamma = \frac{\beta}{3}$$

$$R_c = \frac{1000}{20} \frac{\log(3)}{\log(1000)} = 8.$$

Die AHT ist also potentiell um einen Faktor 8 schneller und benutzt einen Akkumulator, der etwa 10000 mal kleiner ist.

## 6.7 Die Strategie der Spurerkennung

### 6.7.1 Die Vorselektion der Daten

Zusätzlich zur xy-Projektion wird die rz-Projektion eines Ereignisses betrachtet. Dort sind die Tracks Sinuskurven, die durch Geraden, die vom Target kommen, approximiert werden können, falls der Transversalimpuls  $p_T$  größer als 200 MeV/c ist.

Da die z-Koordinate nur mit einer Auflösung bestimmt werden kann, die eine Größenordnung schlechter ist als die in der x- und y-Koordinate, erscheint dreidimensionales Tracking nicht sehr aussichtsreich. Aber die z-Information kann dazu benutzt werden, die Anzahl der Punkte, die in der xy-Ebene betrachtet werden soll, zu reduzieren.

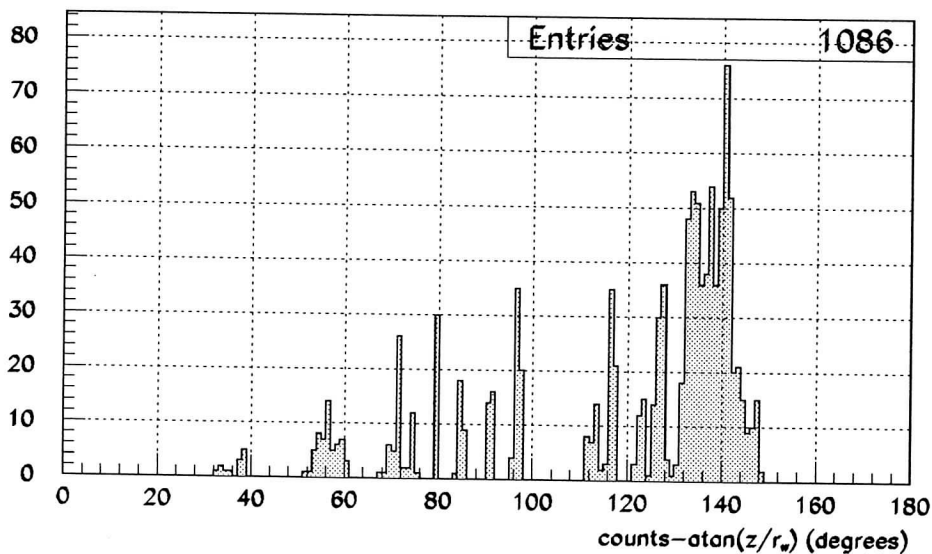


Abbildung 6.8:  $\theta$ -Histogramm für das Ereignis in Abbildung 6.7.  $180^\circ$  liegt in Strahlrichtung. Man sieht deutlich die höhere Trackdichte in Vorwärtsrichtung. Da maximal 30 Punkte auf einer Spur liegen, enthält ein Peak der Höhe 30 wahrscheinlich eine Spur.

Diese Vorselektion kann durch ein  $\theta$ -Histogramm erreicht werden.  $\theta$  ist dabei der Winkel, unter dem ein Hit in der  $rz$ -Ebene vom Koordinatenursprung aus gesehen wird, der sogenannte *Dipwinkel*. Für das Ereignis in Abbildung 6.7 erhält man das Histogramm in Abbildung 6.8. Der erste Schritt in der Spurerkennung besteht darin, mit Hilfe dieses  $\theta$ -Histogramms die Punktmenge in kleinere Pakete aufzuteilen.

Auf diese Weise wird also ein Ereignis hoher Multiplizität, zum Beispiel Au+Au bei 1 GeV/u, in eines niedriger Multiplizität, vergleichbar mit Ca+Ca bei 1 GeV/u, überführt. Diese Ereignisse niedriger Multiplizität werden dann in der  $xy$ -Ebene separat behandelt. Dabei wird ein Sektor nach dem anderen abgearbeitet. Diese Prozedur ist ähnlich der Vorgehensweise des Trackingprogramms für die DIOGENE-Driftkammer am SATURNE in Saclay [POI83].

Unter den so erhaltenen Punktmengen werden mit der AHT Trackkandidaten gesucht.

Das Schema sieht nun folgendermaßen aus (Abbildung 6.9):

- Eine ZEBRA bank [GOE92] mit den zu analysierenden Daten wird

eingelassen und ein  $\theta$ -Histogramm angelegt.

- Mit Hilfe des  $\theta$ -Histogramms wird eine Partition der Hits angelegt.
- Daraus wird die Eingangsdatenstruktur für die AHT aufgebaut.
- Die AHT wird ausgeführt.
- Die Hits, die zum gleichen Peak beitragen, werden mit einer inversen Abbildung (*Backmapping*) gesammelt. Mit einem Kreisfit werden die Trackparameter in der xy-Ebene für diese Punkte bestimmt. Zusätzlich wird ein robuster Geradenfit in der rz-Ebene auf diese Punkte angewendet. Schließlich wird ein Helixfit ausgeführt, um die endgültigen Trackparameter zu erhalten.
- Als letzter Schritt wird ein Vertexfit ausgeführt.
- Nachdem alle Cluster auf diese Weise untersucht worden sind, können präzisere (aber langsamere) Track- und Vertexfits, z. B. mit Berücksichtigung der Vielfachstreuung, durchgeführt werden.

Für die AHT ist in Abbildung 6.9 ein Flußdiagramm mit mehr Einzelheiten gezeigt.

## 6.7.2 Das Backmapping

Wurde ein Peak im Hough-Feld und damit ein Parametersatz für einen möglichen Track-Kandidaten identifiziert, so werden die Hits, die zu diesem Peak beigetragen haben, mit Hilfe folgender Algebra bestimmt:

Die CDC-Helix wird durch 5 Parameter bestimmt, die auch beim Track- und Vertex-Fitting benutzt werden:

$\omega = \frac{1}{R}$  : Krümmung ( mit Vorzeichen, positiv, falls im Uhrzeigersinn )

$\tan(\lambda)$  : Tangens des Dipwinkels

$\phi_0$  : Emissionswinkel, Winkel am Punkt größter Annäherung an den Koordinatenursprung in der xy-Ebene

$d_0$  : *Impact-Parameter*, Abstand zum Koordinatenursprung in der xy-Ebene, vorzeichenbehaftet, positiv, falls positiver Drehimpuls.

$z_0$  : z-Koordinate an der Stelle  $d_0$

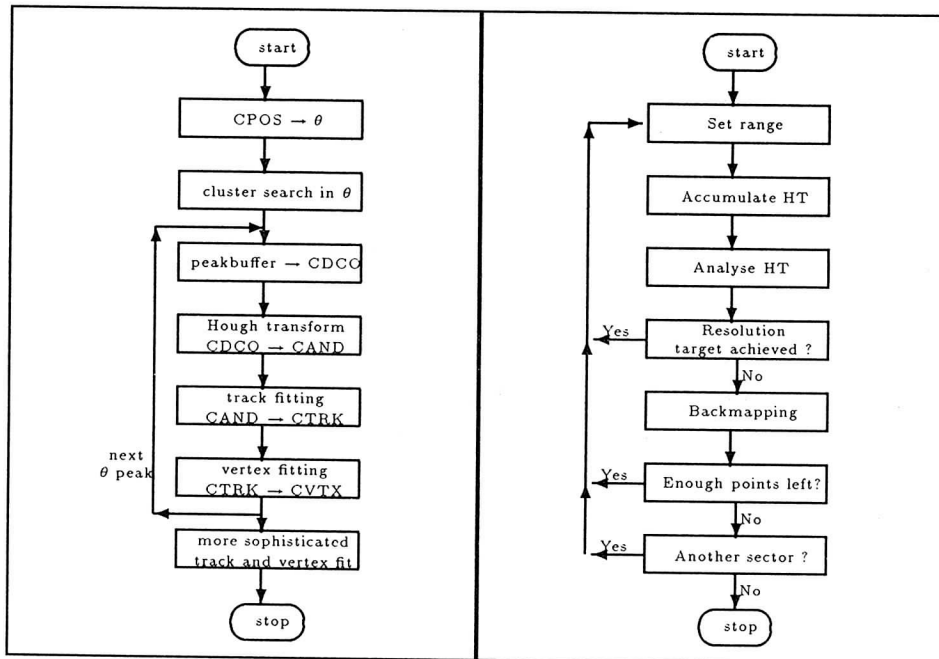


Abbildung 6.9: Flußdiagramm der Spurrekonstruktion

Die Mittelpunktskordinaten des Kreisbogens, den man erhält, indem man die helixförmige Teilchenspur entlang ihrer Symmetrieachse (= Strahlachse) auf die Ebene senkrecht zur Strahlachse (= xy-Ebene) projiziert, sind gegeben durch:

$$x_c = (d_0 - R) \cdot \sin(\phi_0),$$

$$y_c = (R - d_0) \cdot \cos(\phi_0).$$

Die Driftebene auf einem Draht kann folgendermaßen parametrisiert werden:

$x_w, y_w$  : Drahtkoordinaten,

$s$  : Driftstrecke,

$\alpha$  : Winkel der Driftebene gegen x-Achse (konstant für einen Sektor).

Ein Punkt auf der Driftebene ist gegeben durch:

$$x = x_w + s \cdot \sin(\alpha),$$

$$y = y_w + s \cdot \cos(\alpha).$$

Der Schnittpunkt der Helix mit der Driftebene kann (in der  $r\phi$ -Ebene) berechnet werden nach:

$$\begin{aligned} A &= (x_w - x_c)^2 + (y_w - y_c)^2, \\ B &= (y_w - y_c) \cdot \cos(\alpha) - (x_w - x_c) \cdot \sin(\alpha), \\ D &= B^2 - A + R^2, \\ s_{\pm} &= -B \pm \sqrt{D}, \end{aligned}$$

vorausgesetzt, daß  $D$  positiv ist. Die Lösung, die innerhalb der Sektor-  
grenzen liegt, wird akzeptiert (für richtige Tracks gibt es keine Zweideu-  
tigkeit).

# Kapitel 7

## Das FOPI Simulationsprogramm

Bevor ein Experiment mit der Datenaufnahme beginnen kann, müssen das optimale Detektorlayout bestimmt und die Rekonstruktionsprogramme geschrieben und getestet werden. Diese Arbeit umfasst normalerweise mehrere Jahre. Wenn dann experimentelle Daten endlich zur Verfügung stehen, können sie ohne große Zeitverzögerung analysiert werden.

Um dies zu gewährleisten, müssen Ereignisse mit den erwarteten Rohdaten so exakt wie möglich simuliert werden.

Daher wurden mit dem Simulationsprogramm GEANT [BRU87], das seit 1974 am CERN entwickelt wird, und verschiedenen Ereignisgeneratoren Spuren erzeugt, um die Spurrekonstruktion zu testen.

Die wesentlichen Schritte in der Simulation sind:

1. Generiere ein Ereignis.
2. (a) Speichere die Parameter der generierten Teilchen (Tracks).  
(b) Propagiere jedes Teilchen durch den Detektor und speichere Energieverlust, TOF, Impuls und Koordinaten entlang des Tracks.  
(c) Benutze die Koordinaten auf einer Spur, um die Schnittpunkte eines jeden Teilchens mit den verschiedenen Detektorkomponenten zu bestimmen.  
(d) Berechne von den Schnittpunkten, welche Drähte in der Driftkammer angesprochen haben.  
(e) Speichere die simulierten Daten mit dem gleichen Format wie die experimentellen Daten ab.

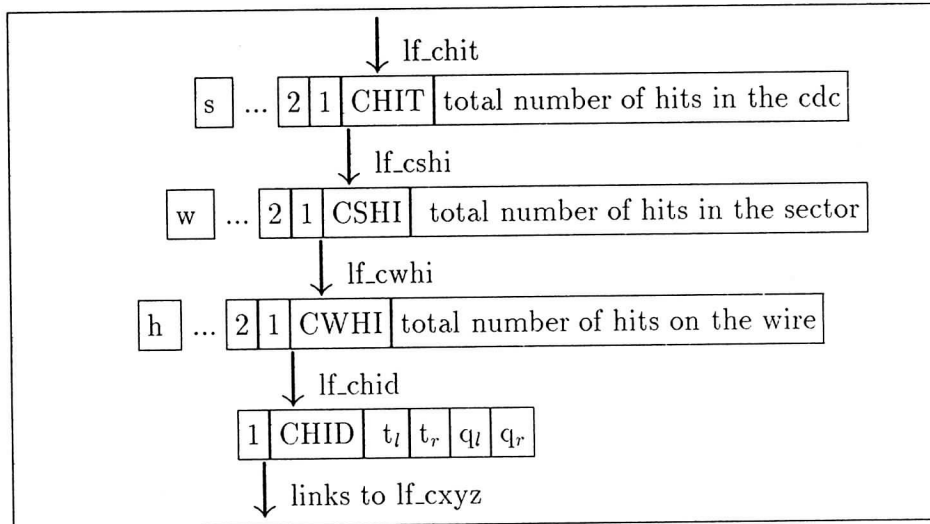


Abbildung 7.1: Struktur der Bank CHIT:  $t_1, t_2$  sind die Ankunftszeiten des Signals an beiden Drahtenden,  $q_1, q_2$  sind die Ladungen, die an beiden beiden Drahtenden ausgelesen werden.

3. Verarbeite das Ereignis mit dem Rekonstruktionsprogramm unter Berücksichtigung der Detektoreffizienz und der Ausschmierung aufgrund der Auflösung der verschiedenen Detektorkomponenten.

Die Simulation der Driftzeit wird gemäß einer linearen Driftzeit-Driftweg-Relation durchgeführt, und das endgültige Resultat wird in TDC-Einheiten angegeben. Für einen Track wird der kürzeste Abstand zu jedem Draht berechnet und in die entsprechende Driftzeit umgewandelt.

Die Information wird in ZEBRA banks [GOE92] gespeichert und auf Platte oder Tape geschrieben. Die Daten enthalten nicht nur die simulierten Rohdaten, sondern auch Informationen über die erzeugten Spuren. Diese werden benötigt, um das Rekonstruktionsprogramm aber auch das Simulationsprogramm selbst zu testen.

Der GEANT dient dazu, die von irgendeinem Ereignisgenerator mit einem bestimmten 4-Impuls erzeugten Teilchen durch das Detektorsystem zu propagieren und in regelmäßigen Abständen physikalische Informationen zu erzeugen und abzuspeichern. Dies geschieht unter Berücksichtigung der Materialeigenschaften und der damit verbundenen physikali-

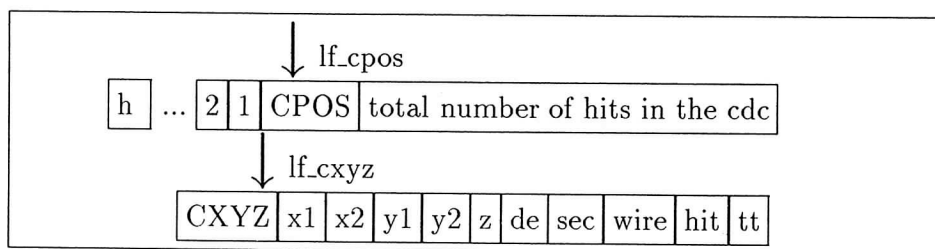


Abbildung 7.2: Struktur der Bank CPOS:  $x_1$ ,  $x_2$ ,  $y_1$ ,  $y_2$ , sind die kartesischen Koordinaten in der  $xy$ -Ebene eines Kammerhits, die im Koordinaten-Finding aus der Driftzeit berechnet werden,  $z$  ist die kartesische  $z$ -Koordinate wie sie über Ladungsteilung gewonnen wird.  $Q$  ist die Gesamtladung,  $tt$  die Driftzeit, gleichzeitig wird die Sektornummer und die Drahtnummer abgespeichert. Dazu kommt noch eine Zahl, die angibt, um den wievielten Hit auf dem Draht es sich handelt.

schen Prozesse, wie Energieverlust und Vielfachstreuung beispielsweise.

Diese Informationen entsprechen aber noch nicht den Daten, die man im Experiment vom Detektor erhält. Die Digitalisierung der Daten vom GEANT bleibt dem Benutzer überlassen, der für eine realistische Simulation möglichst vielen Eigenschaften des jeweils vorliegenden Detektors berücksichtigen muß. Die Routine GUDIGI, insbesondere die Routine CDCDIGI für die CDC, sorgt für die Digitalisierung der simulierten Daten, Erzeugung von Rohdaten wie im Experiment und Füllen der entsprechenden ZEBRA-Banks. Um die gleiche Analysesoftware für experimentelle und simulierte Daten anwenden zu können, wurde die gleiche Datenstruktur wie im Experiment verwendet. Hier wird das FORTRAN-Unterprogramm-Paket ZEBRA verwendet, das eine dynamische Datenverwaltung mit Hilfe von Zeigern und Listen oder Bäumen ermöglicht. Die in der Analyse zur Verfügung stehenden Daten haben damit die in Abbildung 7.1 dargestellte Struktur.

Für die Simulation von Ereignissen im  $4\pi$ -Detektor stehen verschiedene Ereignisgeneratoren zur Verfügung:

- MCRANCONE; eine bestimmte Anzahl von Testteilchen kann mit einem bestimmten Impuls in einen Konus emittiert werden.
- MCFRED: Goldhaber Fragmentation
- COSMIC: Cosmic Rays
- LANDAU FIREBALL

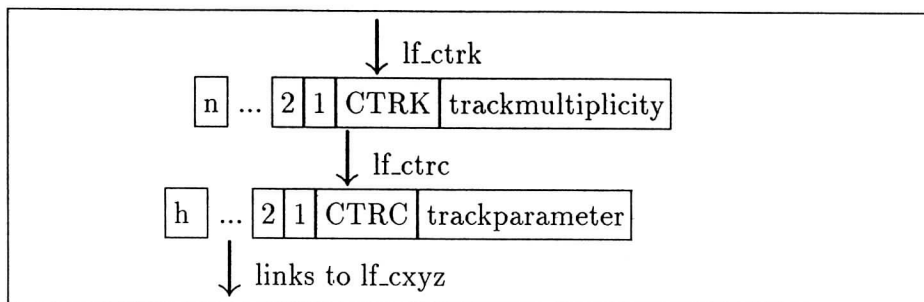


Abbildung 7.3: Struktur der Track-Bank CTRK: Einmal wird die Anzahl der Hits pro Track abgespeichert und ein Errorflag. Dann kommen die fünf Helixparameter und die dazugehörige Kovarianzmatrix, von der nur die fünfzehn oberen Elemente abgespeichert werden, da die Matrix symmetrisch ist.

- FREESCO

Für die Untersuchungen im Rahmen dieser Arbeit wurden nur MCRANCONE und FREESCO [FAI86] verwendet. Mit MCRANCONE wurden die numerischen Eigenschaften des Track- und Vertexfits einmal getestet und dann die Doppelspurauflösung der adaptiven Houghtransformation (AHT) bestimmt. Der FREESCO Ereignisgenerator kam hauptsächlich für die Untersuchung der Effizienz der AHT zum Einsatz.

FREESCO (FRAGMENTATION BY EXPLOSION/EVAPORATION SIMULATION CODE) ist ein 1986 von G. Fai und J. Randrup entwickeltes Programm, das den (statistischen) Zerfall von hochenergetischer Kernmaterie behandelt, wie sie bei Schwerionenstößen mit Projektilenergien oberhalb einiger zehn MeV/u auftritt [FAI86]. Das zugrundeliegende Modell stützt sich auf den verfügbaren mikrokanonischen Phasenraum und ist anwendbar bis zu einigen GeV/u Strahlenergie, wo die Erzeugung von anderen Teilchen als Pionen wichtig wird. Ziel ist unter anderem die Beschreibung der Häufigkeit, Energieverteilung und kollektiven Eigenschaften der emittierten Fragmente. In der experimentellen Schwerionenphysik wird das Programm beim Design von Detektoren sowie zur Bestimmung des kollektiven Flusses aus den experimentellen Daten benutzt.

Da die Simulation detaillierter dynamischer Prozesse viel zu aufwendig ist, wird ein Modell benutzt, das von einfachen geometrischen Überlegungen ausgeht. Die Nukleonen von Projektil und Target werden entsprechend dem "Fireball"-Modell in 3 unabhängige Gruppen (=Quellen)

eingeteilt:

1. Projektil-Zuschauer (“projectile spectators“)
2. Target-Zuschauer (“target spectators“)
3. “Participants“

Jede Quelle wird beschrieben durch:

- die Baryonenzahl
- die Ladung
- den 4er Impuls

Bei Quellen mit hoher Anregungsenergie wird das vollständige Zerplatzen der Quellen, bei Quellen mit niedriger Energie wird Abregung durch Abdampfen von Fragmenten simuliert. Als Grenze zwischen diesen beiden wird

$$E_{Grenze} = A \cdot \frac{\text{Bindungsenergie}}{\text{Nukleon}}$$

benutzt. Der Zerfall erfolgt in Pionen, Nukleonen, Deuteronen,  $\alpha$ -Teilchen und schwerere Fragmente, wobei die Kerne in einen entsprechend den tabellierten Daten definierten Endzustand übergehen. Konkurrierender, bzw. anschließender elektromagnetischer Zerfall wird nicht behandelt.

Der Zerfall von Quellen hoher Anregungsenergie wird statistisch unter Berücksichtigung des mikrokanonischen Ensembles beschrieben. Dabei wird nicht von einem Gleichgewichtszustand ausgegangen.

# Kapitel 8

## Test mit Simulationen

*Andererseits dürfen wir die Effizienz nicht einfach ignorieren.*

– JON BENTLEY

### 8.1 Effizienz und Zeitaufwand

Um die Performance eines Programms zu messen, braucht man ein quantitatives Maß. Dieses Maß wird im Bereich der Spurerkennung (Pattern Recognition) in der Physik als *Effizienz* bezeichnet. Es ist auf den ersten Blick überraschend, daß in vielen Veröffentlichungen auf diesem Gebiet die erhaltene Effizienz *nicht* diskutiert wird. Andererseits haben die Autoren aber sicher eine solche Größe, um ihre Methoden zu testen.

Ein Grund dafür ist gewiß, daß es keine universelle Definition einer Effizienz gibt und daher Befürchtungen vor ungerechtfertigten Vergleichen und Mißinterpretationen bestehen. Um dies zu illustrieren, seien  $m$  Ereignisse mit einer mittleren Anzahl von  $n$  Tracks pro Ereignis betrachtet. Es gibt nun verschiedenen Möglichkeiten, eine Effizienz zu definieren [GRO87]:

- Die mittlere Wahrscheinlichkeit, einen einzelnen Track zu finden, ist die Zahl der korrekt gefundenen Track dividiert durch die totale Anzahl der Tracks ( $n \cdot m$ ). Diese Zahl fällt normalerweise ziemlich langsam mit größer werdendem  $n$ , wenn also die Ereignisse immer konfuser werden.
- Die Zahl der Ereignisse, bei denen *alle* Tracks korrekt gefunden

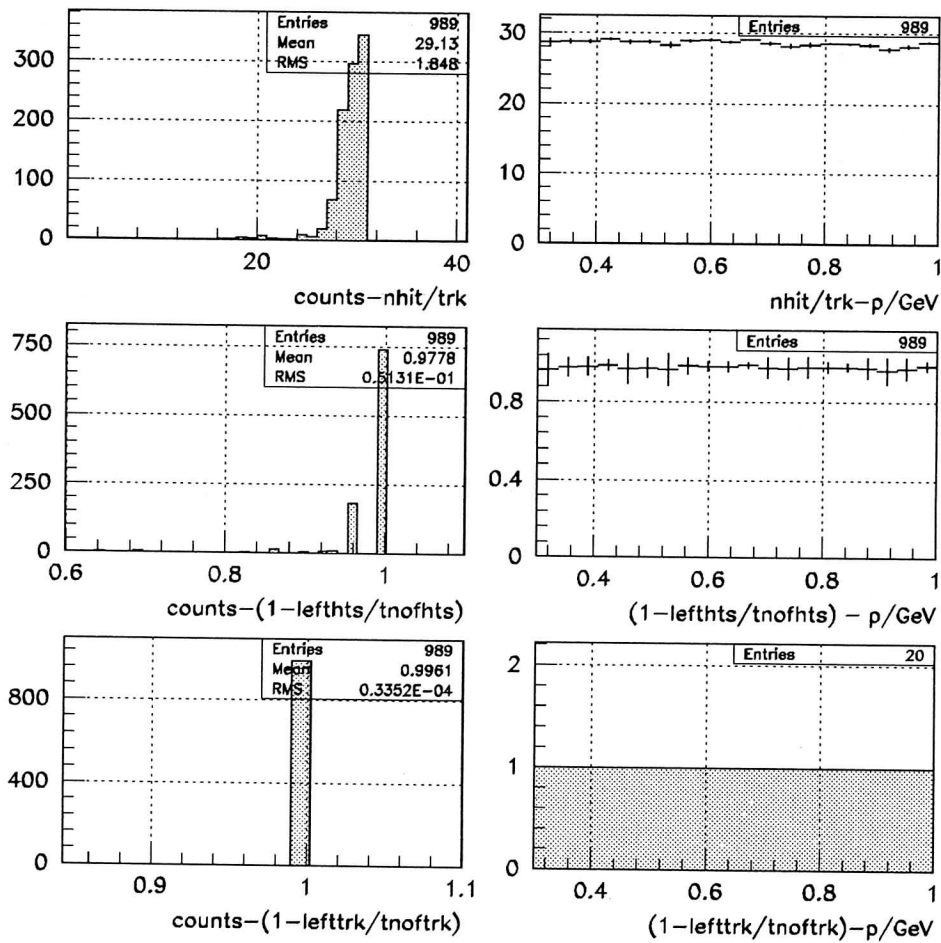


Abbildung 8.1: Effizienz der Spurrekonstruktion für eine Spur pro Ereignis: Die erste Zeile zeigt, aus wievielen Hits ein rekonstruierter Track besteht. Maximal sind 30 Hits möglich. Die Teilchenspuren werden über einen weiten Impulsbereich nahezu vollständig rekonstruiert. Zweite Zeile: Point-Removal-Effizienz und ihre Abhängigkeit vom Teilchenimpuls. Letzte Zeile: Track-Removal-Effizienz und ihre Abhängigkeit vom Teilchenimpuls.

wurden. Diese Größe stellt ein sehr empfindliches Maß dar und fällt normalerweise sehr schnell mit  $n$ .

- Der Anteil an Ereignissen, in dem wenigstens ein bestimmter Teil, beispielsweise 90%, der Tracks korrekt gefunden wurden.

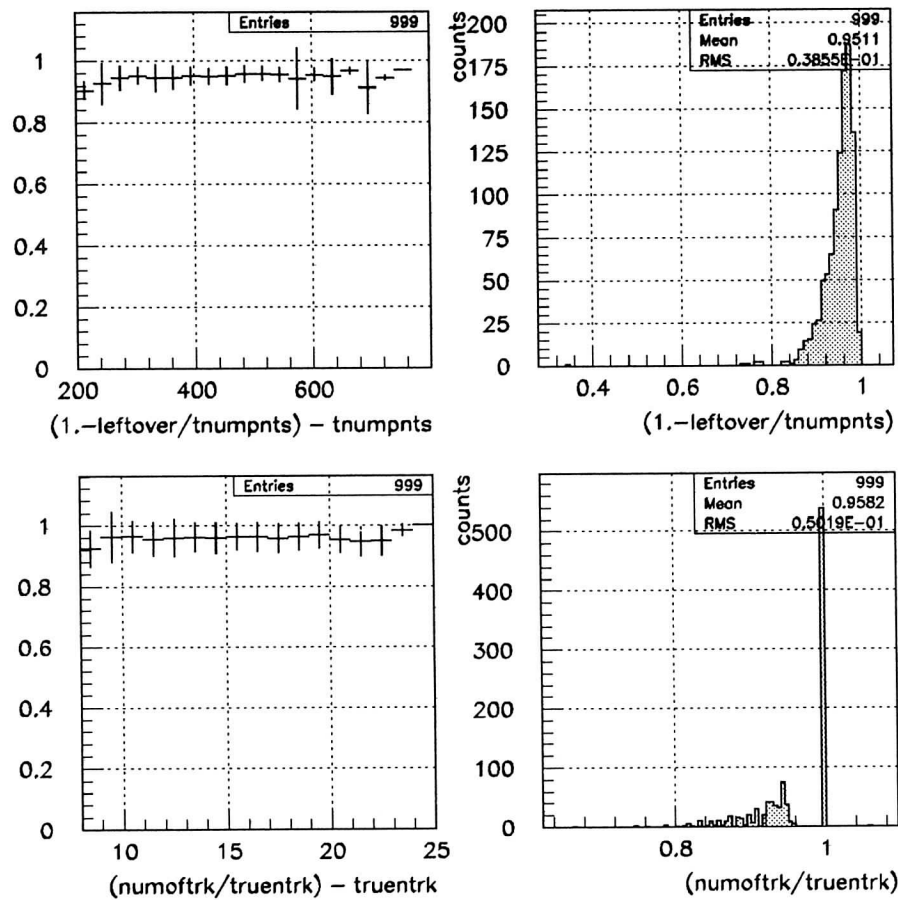


Abbildung 8.2: Effizienz der Spurrekonstruktion für etwa 1000 mit FREESCO simulierte Ca+Ca-Ereignisse mit 1 GeV/u: Erste Zeile: Point-Removal-Effizienz. Letzte Zeile: Track-Removal-Effizienz. Beide Effizienzen über 95%.

- Dann kann man jede der obigen Metriken nehmen, aber Tracks, die notorisch Schwierigkeiten bereiten, unberücksichtigt lassen, wie zum Beispiel Tracks mit einem Impuls, der kleiner als ein bestimmter Betrag ist, oder Tracks in einer besonders schwierigen Detektorregion.

Alle diese Vorschläge sind aber weit davon entfernt, ein zufriedenstellendes Maß für die Effizienz zu bieten. Um von irgendeinem Nutzen zu sein,

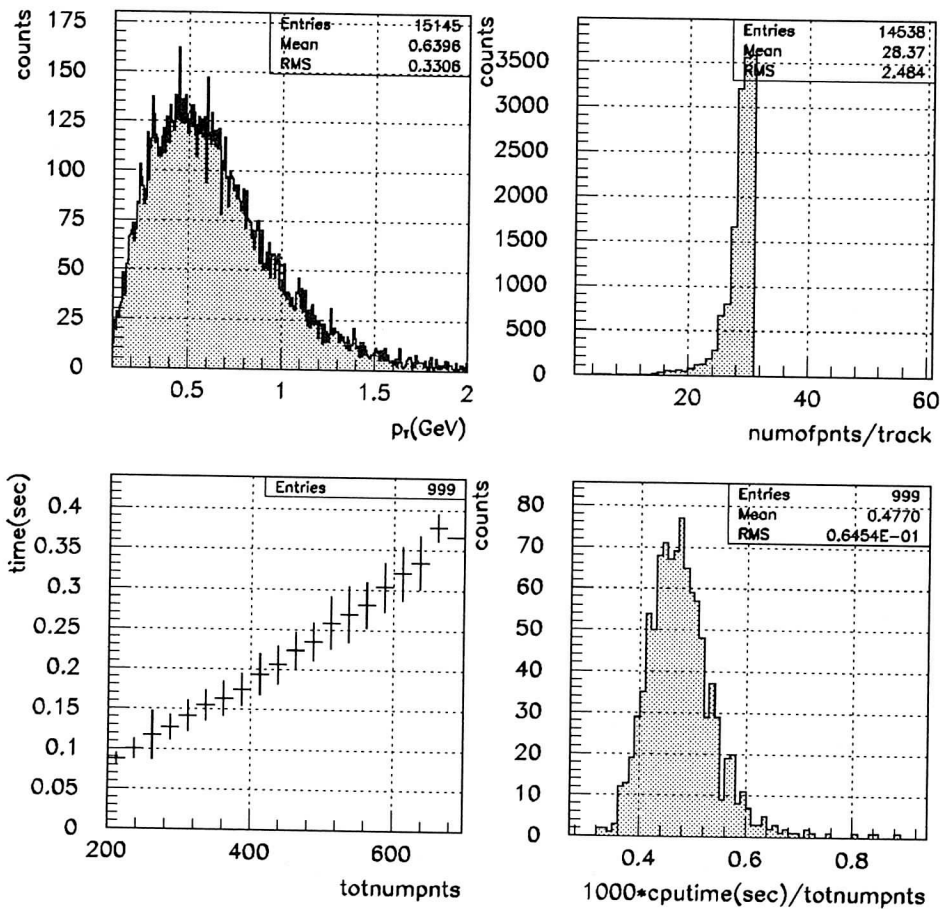


Abbildung 8.3: Erste Zeile: Transversale Impulsverteilung für Ca+Ca bei 1 GeV/u. Daneben: Vollständige Spurrekonstruktion. Letzte Zeile: Zeitaufwand mit der Anzahl der Punkte pro Ereignis.

müssen diese Definitionen auch die Zahl der nicht korrekt gefundenen Tracks enthalten, d.h. Tracks, die aus Punkten konstruiert wurden, die in Wirklichkeit zu anderen Tracks gehören, oder die Untergrund-Punkte enthalten, die keinem Track mit Sicherheit zugeordnet werden können: je höher die Zahl solcher Tracks ist, umso niedriger muß die Effizienz sein.

Wenn man all dies berücksichtigt, ist es nicht mehr so überraschend, daß solche Ergebnisse nicht gezeigt werden. Sie können nur unter Berücksichtigung der speziellen Bedingungen verstanden und normalerweise nicht mit anderen Experimenten verglichen werden. Auf der anderen Sei-

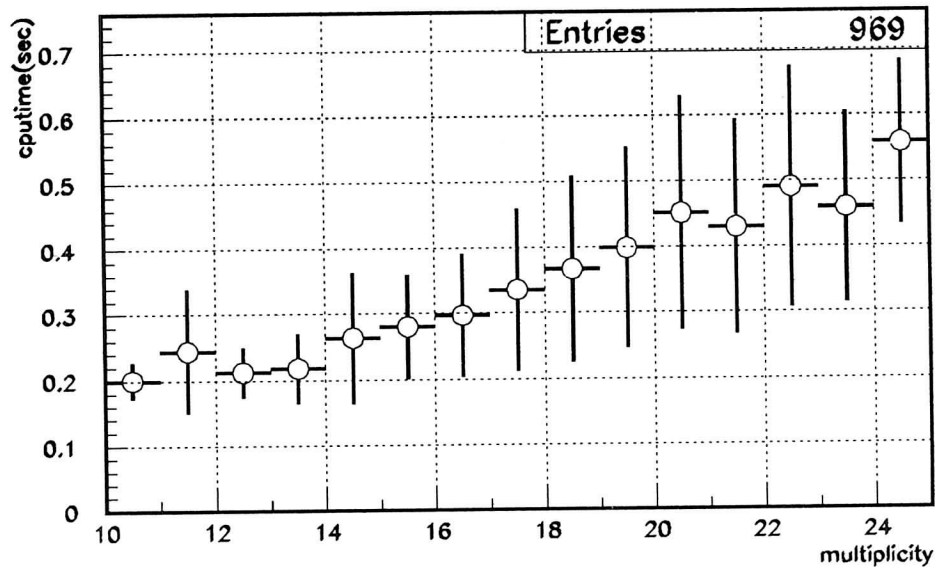


Abbildung 8.4: Zeitaufwand für ein Ca+Ca-Ereignis gegen die Anzahl der Spuren pro Ereignis aufgetragen.

te muß jeder, der ein Programm zur Rekonstruktion von Teilchenspuren schreibt und testet eine solche Größe für sich selbst definieren.

Zuletzt sollte betont werden, daß der *Zeitaufwand* einer Methode ebenso ein wichtiger Aspekt ist, der teilweise dazu zwingt, Kompromisse einzugehen zwischen einer guten aber langsamen und einer schnellen, dafür aber weniger effizienten Methode. In den meisten Fällen ist aber die Effizienz die wichtigere Größe, und man wird normalerweise versuchen eine gute Methode zu beschleunigen, indem man den Algorithmus versucht, zu optimieren, anstatt eine weniger effiziente Methode zu benutzen, oder, wie Weinberg (1972) gesagt hat:

*Jedes Programm, das funktioniert, ist besser als eines, das nicht funktioniert.*

Als globale Größen für den Vergleich verschiedener Tracking Algorithmen könnten folgende benutzt werden [ALA81]:

$m_k$  : Anzahl der gefundenen Objekte

$m_T$  : Anzahl der richtig gefundenen Objekte

$m_F$  : Anzahl der falsch gefundenen Objekte

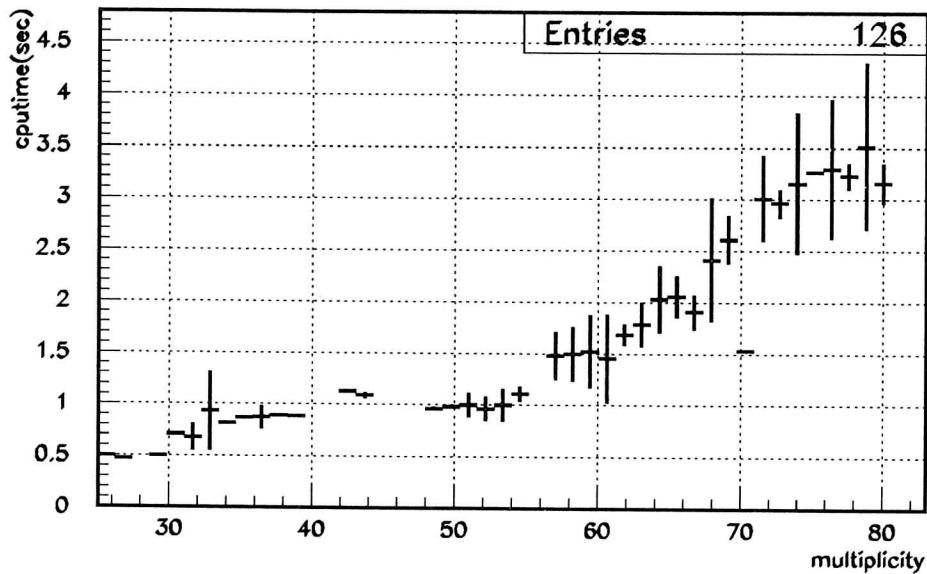


Abbildung 8.5: Zeitaufwand für ein Au+Au-Ereignis gegen die Anzahl der Spuren pro Ereignis aufgetragen.

$M_k$  : wahre Anzahl der Objekte

Für die Robustheit

$$\begin{aligned}
 P &= \text{Precision} &= \frac{m_T}{m_k} \\
 R &= \text{Recall} &= \frac{m_k}{M_k} \\
 G &= \text{Goodness} &= \frac{M_k - m_F}{M_k} \\
 E &= \text{Komplexität (Zeitaufwand)}
 \end{aligned} \tag{8.1}$$

Die ersten drei Größen müssen für eine gute Performance möglichst groß sein, der Zeitaufwand dagegen möglichst gering.

Um zwischen verschiedenen Algorithmen diskriminieren zu können, ist es besser eine Größe zu haben:

$$D = \frac{\alpha P + \beta R + \gamma G}{\delta E + 1} \tag{8.2}$$

Dabei sind  $\alpha, \beta, \gamma, \delta$  relative Gewichte.

Von den endgültigen Track-Kandidaten werden die Track-Indizes der Hits extrahiert. Dann werden die Effizienzen für die realen Tracks bestimmt.

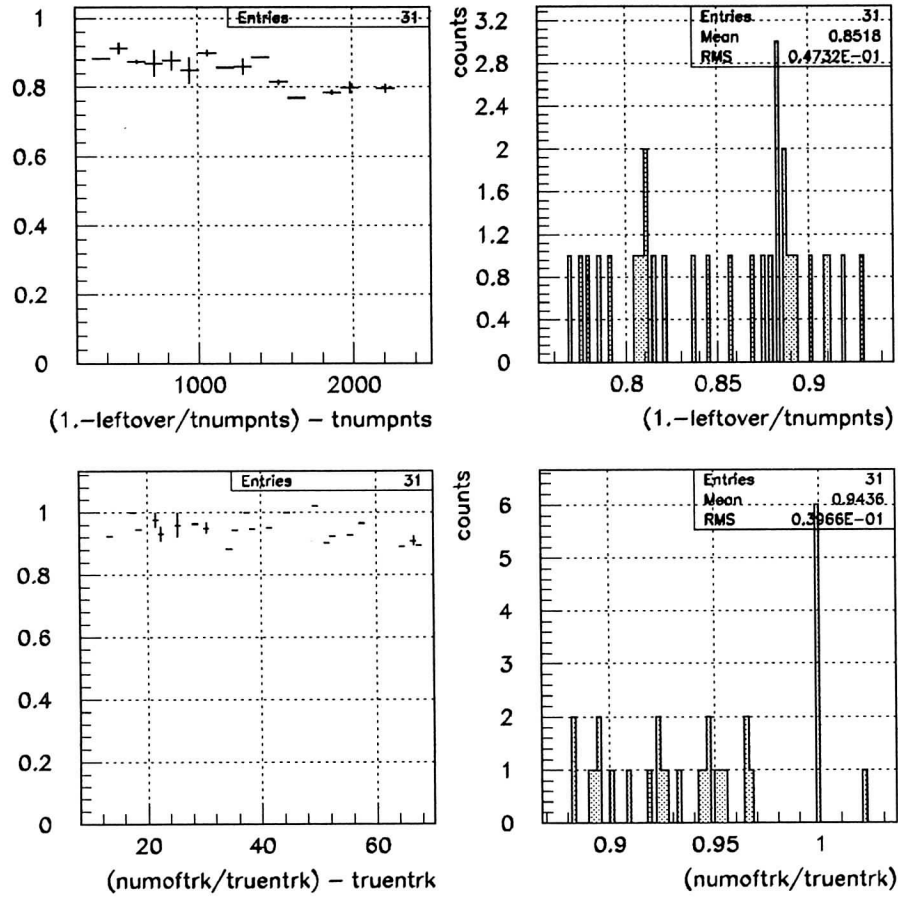


Abbildung 8.6: Effizienz der Spurrekonstruktion für etwa 30 mit FREES-CO simulierte Au+Au-Ereignisse mit 1 GeV/u: Erste Zeile: Point-Removal-Effizienz. Letzte Zeile: Track-Removal-Effizienz.

In dieser Arbeit wurde folgendermaßen vorgegangen: Ein Track wird als guter Track bezeichnet, wenn weniger als 20% der Hits falsch sind im Vergleich mit dem simulierten Track. Ein Hit wird als falsch bezeichnet, falls er nicht zum simulierten Track gehört oder fehlt. Daher wird die *Track-Effizienz*  $\epsilon_{trk}$  folgendermaßen definiert:

$$\epsilon_{trk} = 1 - \frac{\text{total number of (bad tracks + missing tracks)}}{\text{total number of simulated tracks used}}. \quad (8.3)$$

Ein simulierter Track wird nur dann für die Analyse benutzt, falls er

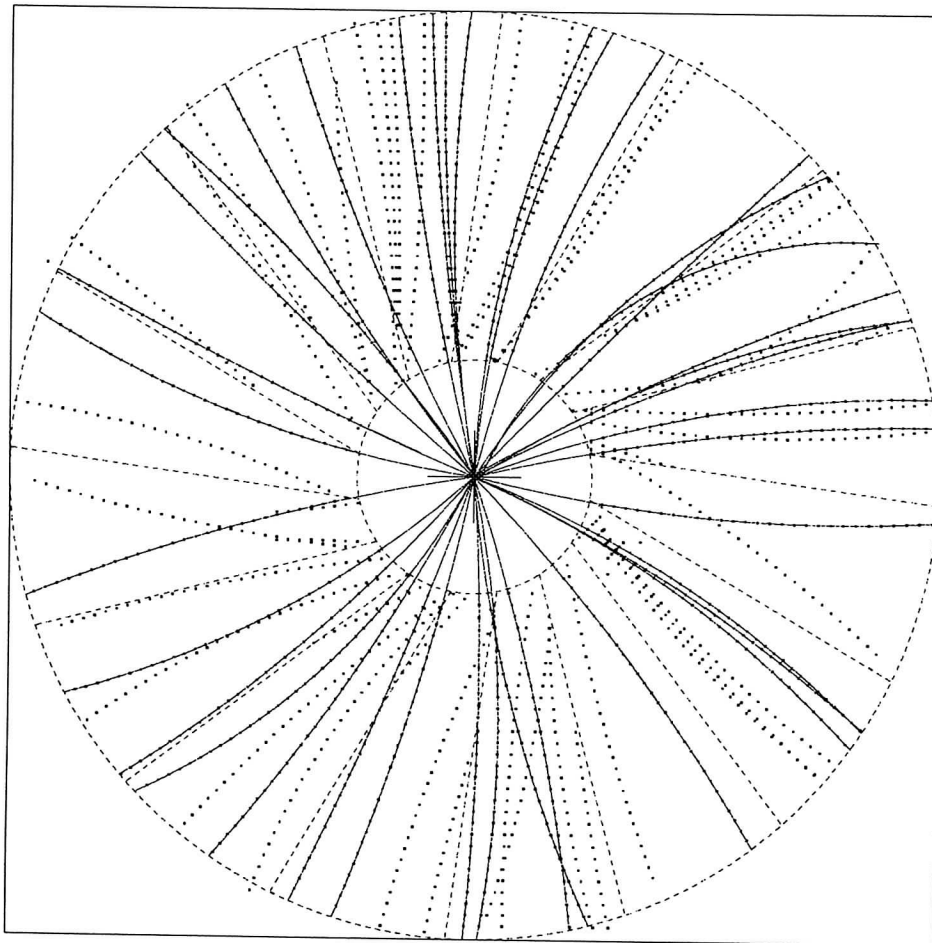


Abbildung 8.7: xy-Projektion eines Ereignisses in der CDC: Au+Au bei 1 GeV/u mit FREESCO simuliert (siehe Kapitel über das FOPI Simulationsprogramm). Alle 37 Spuren wurden mit der adaptiven Hough-Transformation rekonstruiert

mehr als vier Punkte hat. Zwei *Point-Finding-Effizienzen* können definiert werden:

$$\epsilon_{pnt}(1) = 1 - \frac{\text{total number of (wrong points on all good tracks + all points on all bad and missing tracks)}}{\text{total number of points on all simulated tracks}} \quad (8.4)$$

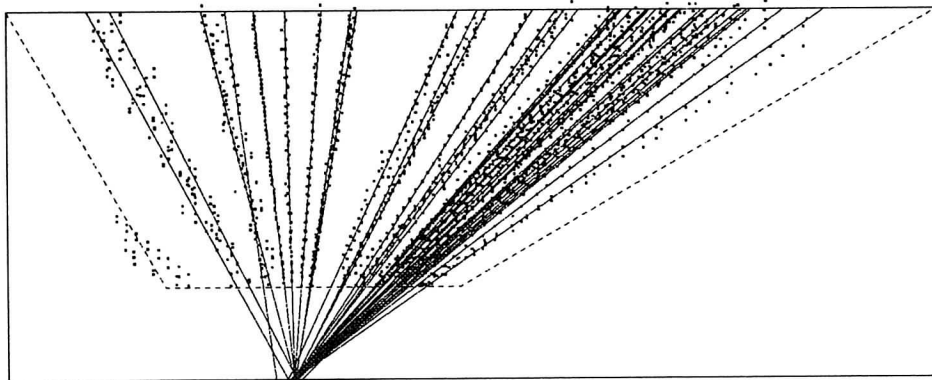


Abbildung 8.8: rz-Projektion des Ereignisses in der CDC von Abbildung 8.7. Die Auflösung des Detektors ist für die z-Koordinate eine Größenordnung schlechter als für die x- und y-Koordinate.

$$\epsilon_{pnt}(2) = 1 - \frac{\text{total number of wrong points on all good tracks}}{\text{total number of points on all good tracks}} \quad (8.5)$$

Offensichtlich steht  $\epsilon_{pnt}(1)$  mit  $\epsilon_{trk}$  in Beziehung und ist immer kleiner.

Abbildung 8.1 zeigt die Effizienz, einen Track pro Ereignis zu finden: Die erste Reihe zeigt die Verteilung der Anzahl der Hits pro Track, die zugordnet werden konnten. Die Simulation beinhaltete 30 Drähte pro Sektor, 30 Hits können also maximal einem Track zugeordnet werden. Die Abbildung zeigt: Wenn ein Track gefunden wurde, so wird er im allgemeinen nahezu vollständig gefunden. Dies ist über einen weiten Impulsbereich, 0.3 bis 1 GeV, erfüllt, die Abhängigkeit vom Teilchenimpuls ist in diesem Bereich flach.

In der zweiten Reihe wird die *Point-Removal-Efficiency*  $\epsilon_{pnt}(1)$  und deren Abhängigkeit vom Impuls gezeigt, in der dritten Reihe die *Track-Removal-Efficiency*  $\epsilon_{trk}$ . Im Mittel wurden 98% aller Hits pro Ereignis einem Track zugeordnet. Auch diese Größe ist in dem betrachteten Impulsbereich flach. Die Track-Removal-Effizienz beträgt 100% über den ganzen Impulsbereich von 300 - 1000 MeV. Es gibt also keine Teilchen Spuren, die aufgrund ihrer geometrischer Lage im Detektor oder aufgrund ihres Impulses (im Bereich von 300 - 1000 MeV) prinzipiell nicht gefunden werden. Ineffizienzen der adaptiven Hough-Transformation rühren also nur sehr unwahrscheinlich von daher.

Das Programm wurde erfolgreich mit 1000 Ca+Ca und 100 Au+Au simulierten Ereignissen auf einer DEC 5000/125 getestet. Die Testdaten wurden mit dem Ereignisgenerator FREESCO bei 1 GeV/u erzeugt. Ineffizienzen des Detektors und Untergrundhits wurden dabei berücksichtigt. Die Abbildung 8.4 zeigt den Zeitaufwand für die Ca-Ereignisse gegen die Trackmultiplizität des Ereignisses aufgetragen, Abbildung 8.5 den Zeitaufwand für die Au-Ereignisse gegen die Trackmultiplizität aufgetragen. Der Zeitaufwand liegt in der Größenordnung von einigen Sekunden bei den Au+Au Ereignissen, bleibt aber bei Ca stets unter einer halben Sekunde.

Abbildung 8.2 schließlich zeigt die hohe Effizienz der adaptiven Hough-Transformation für diese realistischeren Ereignisse. Die mittlere Zahl der *leftovers*, also der Hits, die nach einer Analyse noch übriggeblieben sind und nicht zugeordnet werden konnten, beträgt nur 5%, dementsprechend liegt die *Track-Removal-Effizienz* bei über 95%.

Abbildung 8.3 gibt dazu die transversale Impulsverteilung der Ereignisse an. Daneben ist die mittlere Zahl der Hits pro rekonstruierter Spur abgebildet: Das Ergebnis der Untersuchungen zur Einzelspureffizienz konnte hier wiederholt werden. Die rekonstruierten Spuren sind im Vergleich zur maximal möglichen Zahl von Hits pro Spur (30) verhältnismäßig vollständig. Trackkandidaten mit weniger als 80% der maximal möglichen Zahl von Hits pro Spur wurden nicht als gute Kandidaten betrachtet. Bei den Fit-Prozeduren ist aber zwecks Verbesserung des  $\chi^2$  eine Wegnahme von Punkten möglich: daher der Ausläufer des Peaks unter 20.

Die letzte Zeile zeigt das Skalenverhalten der adaptiven Houghtransformation (AHT) in der Zeit mit der Anzahl der Punkte pro Ereignis. Diese Größe ermöglicht auch einen direkteren Vergleich mit dem Experiment, da die totale Anzahl der Punkte pro Ereignis ein leicht zugänglicher Wert ist. In dieser Abbildung zeigt sich besser das nahezu lineare Verhalten der adaptiven Hough-Transformation. Geht man von einer mittleren Zahl von 30 Hits pro Track aus, kann man aus der Steigung, deren Verteilung die Abbildung rechts unten zeigt, eine mittlere Zeit von 15 ms pro Track bestimmen. Dabei ist Track- und Vertex-Fitting schon eingeschlossen.

Bei Au+Au Ereignissen mit 1 GeV/u sind die Ergebnisse analog, aufgrund der beschränkten Speicherkapazität jedoch nur mit schlechter Statistik (Abbildung 8.6). Abbildung 8.7 und Abbildung 8.8 zeigen ein mit FREESCO simuliertes Au+Au-Ereignis mit 1 GeV/u. Mit der AHT ist man in der Lage, Driftkammerereignisse hoher Teilchenmultiplizität

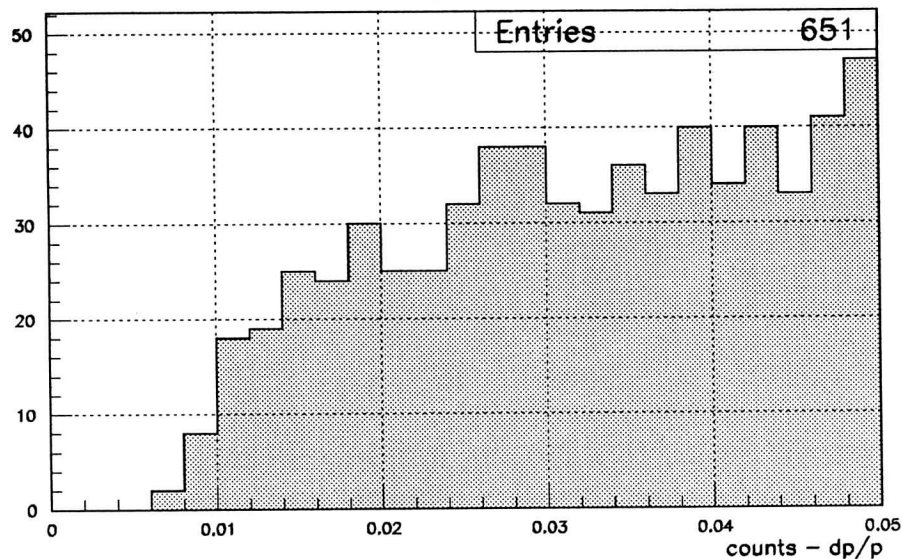


Abbildung 8.9: Doppelspurauflösung der adaptiven Houghtransformation (AHT): etwa bei 1%, wenn der Detektor eine Doppelspurauflösung von 5 mm aufweist.

zu analysieren.

## 8.2 Die Doppelspurauflösung

Nicht nur beim Bau einer Driftkammer ist die sogenannte Doppelspurauflösung, der Abstand zweier Teilchen im Phasenraum, die gerade noch aufgelöst werden, von grundlegender Bedeutung, auch die verschiedenen Methoden zur Spurrekonstruktion unterscheiden sich bei der Frage, wann zwei dicht beieinanderliegende Tracks noch als zwei Tracks rekonstruiert werden.

Um die Doppelspurauflösung der adaptiven Houghtransformation (AHT) zu testen, wurde folgendermaßen vorgegangen:

Ein Teilchen, beispielsweise ein Proton, wurde am primären Vertex mit einem bestimmten Impuls erzeugt und durch das CDC-Volumen propagiert. Dazu wurde ein zweites Teilchen der gleichen Art erzeugt mit dem Abstand Null im Ortsraum, also dem gleichen Erzeugungsort und der gleichen Emissionsrichtung, und einem Impuls, der auf einer Kugel mit Radius  $\Delta p$  im Impulsraum um den Impuls des ersten Teilchen liegt.

Bei der Simulation der Spur dieses Teilchens in der CDC wurde die Doppelspurauflösung der Driftkammer von 5 mm berücksichtigt.

Abhängig vom vorgegebenen  $\Delta p$  und von der Doppelspurauflösung des Detektors gibt es einen kritischen Impuls, bei dem die Anzahl der rekonstruierten Tracks von zwei auf eins sinkt.

Auf diese Weise erhält man die Doppelspurauflösung der AHT, die bei 1% gefunden wurde (Abbildung 8.9).

Die bisher untersuchten Größen betrachten detailliert die Zuverlässigkeit der Spurerkennung. Um die Zuverlässigkeit der physikalischen Resultate zu untersuchen, muß die Qualität der Spuranpassung (Fit) an die gemessenen Punkte untersucht werden.

# Kapitel 9

## Track–Fitting

*No passion in the world is equal to the  
passion to alter someone else's draft.*

– H. G. WELLS

### 9.1 Einführung

Track- und Vertex-Fitting sind ein wichtiger Bestandteil der Datenanalyse in einem Hochenergieexperiment. Der Track-Fit extrahiert und komprimiert die Information, die in den Meßwerten von Tracking-Detektoren enthalten ist, durch die Anwendung von statistischen Methoden. Das Resultat des Track-Fits besteht aus einem oder mehreren Vektoren mit Track-Parametern, dem Ausgangsort, der Richtung und der Krümmung der Spur in Bezug auf eine Referenzfläche. Im Vertex-Fit werden die Track-Parameter als virtuelle Meßwerte betrachtet. Sie werden benutzt, um die Position des Wechselwirkungspunktes und die Impulsvektoren der Teilchen am Wechselwirkungspunkt zu bestimmen.

Track-Fit und Vertex-Fit sind die letzten Schritte in einer langen Analyseketten, die im allgemeinen folgende Aufgaben umfaßt:

- on-line Selektion der Daten (multi-level triggering),
- Kalibration der Rohdaten,
- Pattern-Recognition in allen Detektor-Komponenten und globale Track-Suche.

Modell	Fehler	
	normalverteilt	nicht normalverteilt
Linear	optimal unter allen linearen Schätzern	äquivalent zur MLM mit einer genügenden Statistik und deshalb ein Schätzer mit minimaler Varianz selbst für eine endliche Anzahl von Meßwerten
Nichtlinear	keine allgemeinen Eigenschaften für eine endliche Anzahl von Meßwerten	Äquivalent zur MLM und deshalb asymptotisch ein Schätzer mit minimaler Varianz

Tabelle 9.1: Eigenschaften der Methode der kleinsten Quadrate (LSM) im Vergleich zur Maximum Likelihood Methode (MLM)

Ein effizienter und zuverlässiger Algorithmus zur Spurerkennung ist von besonderer Wichtigkeit für die Gesamteffizienz der Spurrekonstruktion, da ein Track, der nicht gefunden wurde, auch nicht angepaßt werden kann und daher unwiderruflich verloren ist. Der Track-Fit hat außerdem zu entscheiden, ob ein Track-Kandidat, wie er von der Track-Suche geliefert wird, einem physikalischen Track entspricht. Entsprechend muß der Vertexfit entscheiden, ob eine Spur wirklich vom primären Vertex kommt, oder ob es wahrscheinlicher ist, daß die Spur von einem sekundären Prozess oder Zerfall herrührt. Diese Entscheidungen werden mit Hilfe statistischer Testvariablen gefällt.

In dieser Arbeit wurde hauptsächlich die Methode der kleinsten Quadrate verwendet.

## 9.2 Konzepte

Die Track-Rekonstruktion soll zwei Aufgaben erfüllen:

- Einmal sollen die Parameter, die den Track geometrisch beschreiben, bestimmt werden, das heißt den Emissionsort, die Richtung und die Krümmung, die optimal im statistischen Sinn ist; das ist der sogenannte Track-Fit.

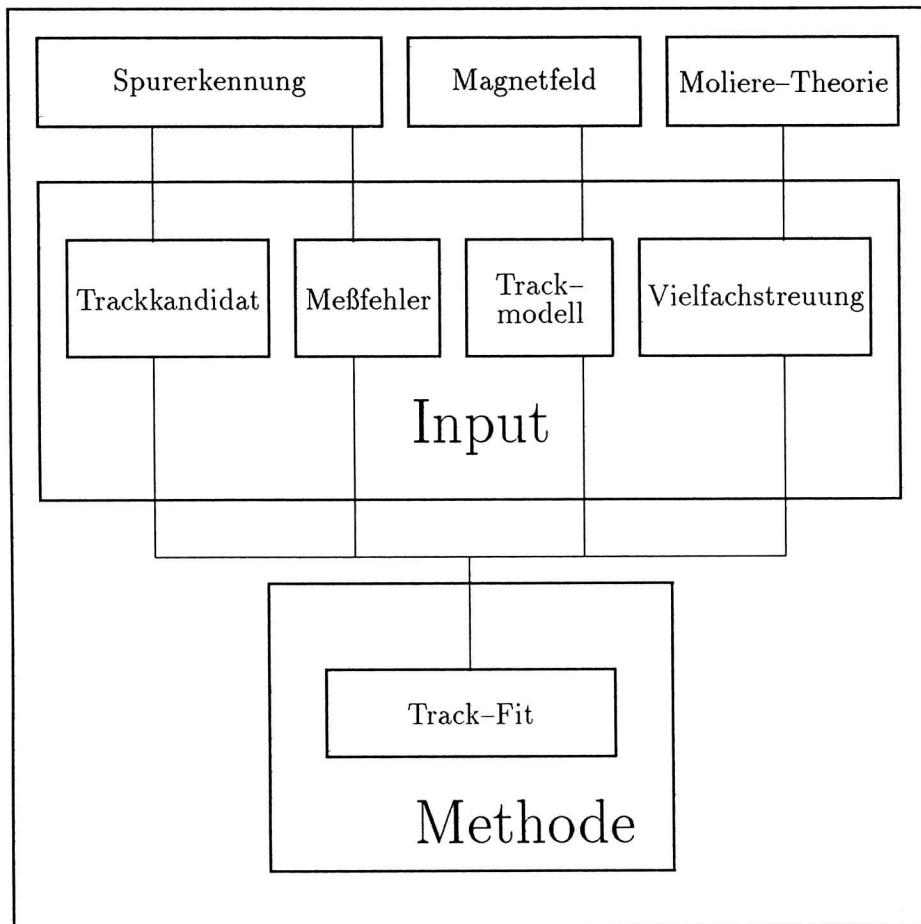


Abbildung 9.1: Input und Methode beim Track-Fit [FRU90]

- Zweitens müssen die Ergebnisse der vorangegangenen Schritte in der Analyse bestätigt oder verworfen und eventuelle Mehrdeutigkeiten aufgelöst werden. Dies erfordert statistische Testvariablen, um die Qualität der Track-Kandidaten zu bestimmen.

Zusätzlich zu den Parametern muß die Kovarianzmatrix bestimmt werden, damit die Werte im nächsten Schritt der Analyse, dem Vertex-Fit, mit der geeigneten Gewichtung eingehen können.

In jeder Diskussion über Track-Fitting muß zwischen dem *Input* und der *Methode* unterschieden werden. Der Input für einen Track-Fit ist gegeben durch:

- Ein durch die Track-Suche gefundener Track-Kandidat  $\mathbf{m}$ .
- Das Track-Modell.
- Die Meßfehler.
- Ein statistisches Modell für die Zufallsprozesse, die die Trajektorie stören, wenn sich das Teilchen durch Materie bewegt.

Im allgemeinen wird angenommen, daß der Input nur schwach von den Track-Parametern abhängt. Falls der Track-Fit doch zu einer großen Änderung der Track-Parameter führt, so muß man unter Umständen einen Schritt zurückgehen und einige Größen des Inputs neu berechnen, so zum Beispiel die Fehler der gemessenen Koordinaten aufgrund der Vielfachstreuung. Möglicherweise ist der Track-Fit nicht erfolgreich und der Track-Kandidat muß wieder aufgelöst werden. Die Koordinaten müssen dann an die Track-Suche zurückgegeben werden.

Die Methode, auf der anderen Seite, ist die Art und Weise, die Parameter, deren Kovarianzmatrix und geeignete Teststatistik unter Berücksichtigung der Größen für den Input zu berechnen. In dieser Arbeit wurde die Methode der kleinsten Quadrate zur Bestimmung der Parameter verwendet. Können die Track-Parameter so gewählt werden, daß das Track-Modell linear in der Nähe der Meßpunkte  $\mathbf{m}$  approximiert werden kann und hängt die Kovarianzmatrix der Fehler nur schwach von den Trackparametern ab, so hat die Methode der kleinsten Quadrate wichtige optimale Eigenschaften:

- Der Schätzwert hat keinen *Bias*,
- und hat minimale Varianz unter allen linearen Schätzern.
- Wenn die Fehler einer Gaußverteilung folgen, sind die nach der Methode der kleinsten Quadrate bestimmten Parameter die optimalen, kein nichtlinearer Schätzer kann bessere bestimmen.
- Außerdem ist die Methode einfach, leicht zu programmieren und weit verbreitet.

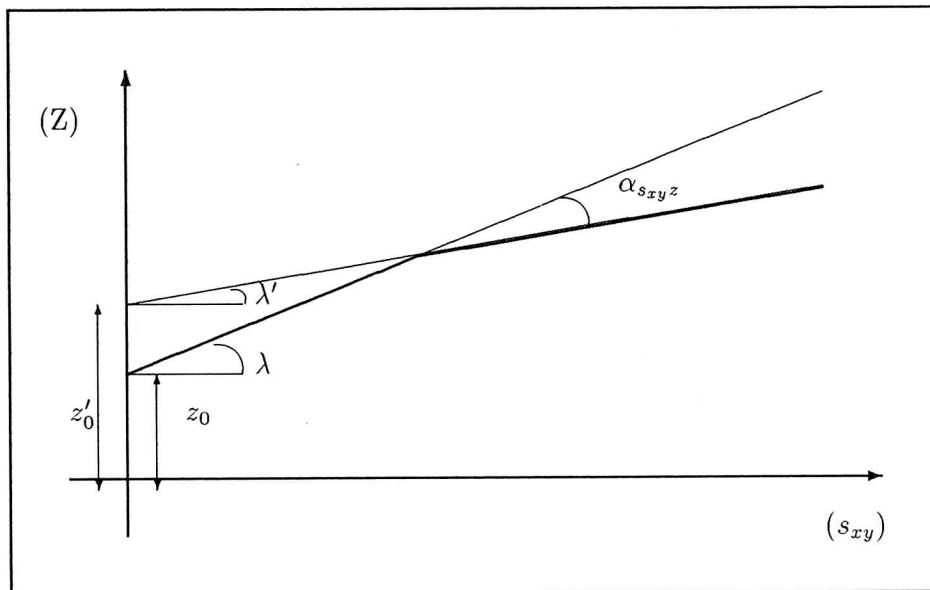


Abbildung 9.2: Effekt der Vielfachstreuung auf den Track

### 9.3 Das Track-Modell

Die Trajektorie eines geladenen Teilchens in einem statischen magnetischen Feld wird durch folgende Bewegungsgleichung gegeben:

$$\frac{d^2 \mathbf{x}}{ds^2} = (kq/p) \cdot \frac{d\mathbf{x}}{ds} \otimes \mathbf{B}(\mathbf{x}(s)) \quad (9.1)$$

wobei:

- s... Pfadlänge
- k... Proportionalitätskonstante
- q... Teilchenladung (mit Vorzeichen)
- p... Betrag des Teilchenimpulses
- $\mathbf{B}(\mathbf{x})$ ... statisches Magnetfeld

Ein Track wird im Vakuum dargestellt durch 5 Anfangswerte  $\mathbf{p}$  auf einer Referenzfläche, den sogenannten Trackparametern: zwei Raumkoordinaten, zwei Richtungswinkeln und der Krümmung.

Wenn Meßfehler und die Wechselwirkung des Teilchens mit Materie vernachlässigt werden können, ist der Vektor aller Meßwerte  $\mathbf{m}$  eine Funktion des Anfangswertes  $\mathbf{p}$ :

$$\mathbf{m} = \mathbf{f}(\mathbf{p}) \quad (9.2)$$

Die Funktion  $\mathbf{f}$  heißt Trackmodell. Man erhält es, wenn man die Bewegungsgleichungen löst. Das Trackmodell wird eindeutig bestimmt durch das Magnetfeld, die Anordnung der Trackingdetektoren und die Wahl der Trackparameter. Für die Methode der kleinsten Quadrate braucht man auch die Ableitungen des Trackmodells, das heißt die partiellen Ableitungen der Funktion  $\mathbf{f}$  bezüglich  $\mathbf{p}$ .

Die letzte Gleichung kann auch unter einem anderen Gesichtspunkt betrachtet werden. Wird  $\mathbf{m}$  als Vektor im  $n$ -dimensionalen Raum aufgefaßt, so definiert die Funktion  $\mathbf{f}$  implizit eine 5-dimensionale Mannigfaltigkeit, die sogenannte *constraint surface*. Die *constraint surface* besteht aus allen Punkten, die einer exakten Lösung der Bewegungsgleichung entsprechen.

Gibt es kein Magnetfeld, so ist die Trajektorie eine Gerade und das Trackmodell ist eine lineare Funktion. In diesem trivialen Fall gibt es nur 4 Anfangswerte, der Impuls kann nicht bestimmt werden. Die *constraint surface* ist eine 4-dimensionale Hyperfläche.

Bei einem homogenen Magnetfeld ( $\mathbf{B}=\text{const}$ ) ist die Trajektorie eine Helix mit einer Achse parallel zu  $\mathbf{B}$ . Der Schnittpunkt einer Helix mit ebenen oder zylindrischen Detektoren kann analytisch bestimmt werden und damit auch die Ableitungen nach den Anfangswerten.

Bei einem inhomogenen Feld müssen die Bewegungsgleichungen numerisch gelöst werden. Das ist das sogenannte Trackfollowing.

## 9.4 Globale Methode der kleinsten Quadrate

Die Abweichung der Meßwerte vom exakten Track-Modell beruht auf zwei Effekten: dem Meßfehler des Detektors und der Vielfachstreuung. Diese Beobachtung kann in die Sprache der Statistik übersetzt werden, indem man sagt, der Vektor aller Meßwerte  $\mathbf{m}$  ist ein Beispiel für einen Zufallsvektor  $\mathbf{c}$  der Gestalt:

$$\begin{aligned} \mathbf{c} &= \mathbf{f}(\mathbf{p}_{true}) + \epsilon \\ \epsilon &= \gamma + \delta \end{aligned}$$

$$\begin{aligned}
cov(\epsilon) &= \mathbf{V} \\
cov(\gamma) &= \mathbf{V}_{ms} \\
cov(\delta) &= \mathbf{V}_{det} \\
\mathbf{V} &= \mathbf{V}_{ms} + \mathbf{V}_{det}
\end{aligned}$$

Die Kovarianzmatrix  $\mathbf{V}_{det}$  ist eine blockdiagonale Matrix mit einem Block pro Detektor. Die Kovarianzmatrix für die Vielfachstreuung,  $\mathbf{V}_{ms}$ , ist im allgemeinen nichtdiagonal. Wegen der Vielfachstreuung und den Meßfehlern liegt der Vektor  $\mathbf{m}$  in einer gewissen Distanz von der constraint surface. Einen Track-Fit kann man auffassen als eine Projektion des Vektors  $\mathbf{m}$  auf diese constraint surface:

$$\mathbf{m} \mapsto \tilde{\mathbf{c}} = \mathbf{f}(\tilde{\mathbf{p}}). \quad (9.3)$$

Der Wert  $\tilde{\mathbf{p}}$  ist ein Näherungswert für die Trackparameter. Die Projektion wird so gewählt, daß der resultierende Näherungswert die kleinste Varianz hat. Dies wird durch die Methode der kleinsten Quadrate erfüllt. Die Trackparameter  $\mathbf{p}$  werden bestimmt, indem man folgende Funktion minimiert:

$$Q(\mathbf{p}) = [\mathbf{m} - \mathbf{f}(\tilde{\mathbf{p}})]^T \mathbf{G} [\mathbf{m} - \mathbf{f}(\tilde{\mathbf{p}})] \quad (9.4)$$

wobei  $\mathbf{m}$  der Vektor aller Meßwerte ist,  $\mathbf{f}$  ist das Trackmodell und  $\mathbf{G}$  ist eine positiv definite Matrix mit Gewichten. Der Wert mit der kleinsten Varianz ergibt sich mit

$$\mathbf{G}^{-1} = cov(\epsilon) = \mathbf{V}_{det} + \mathbf{V}_{ms} = \mathbf{V}. \quad (9.5)$$

Die Ableitung von  $Q$  bezüglich  $\mathbf{p}$  wird gegeben durch

$$\frac{\partial Q}{\partial \mathbf{p}} = -2[\mathbf{m} - \mathbf{f}(\tilde{\mathbf{p}})]^T \mathbf{G} \mathbf{A} \quad (9.6)$$

$$\mathbf{A} = \frac{\partial \mathbf{f}}{\partial \mathbf{p}}. \quad (9.7)$$

Um nun die Nullstelle von  $\frac{\partial Q}{\partial \mathbf{p}}$  zu finden, wendet man die Newtonmethode an und erhält:

$$\begin{aligned}
\tilde{\mathbf{p}}_i &= \tilde{\mathbf{p}}_{i-1} + (\mathbf{A}^T \mathbf{G} \mathbf{A} - \mathbf{H} \mathbf{G} [\mathbf{m} - \mathbf{f}(\tilde{\mathbf{p}}_{i-1})])^{-1} \mathbf{A}^T \mathbf{G} [\mathbf{m} - \mathbf{f}(\tilde{\mathbf{p}}_{i-1})] \\
\mathbf{H} &= \frac{\partial^2 \mathbf{f}}{\partial \mathbf{p}^2}.
\end{aligned} \quad (9.8)$$

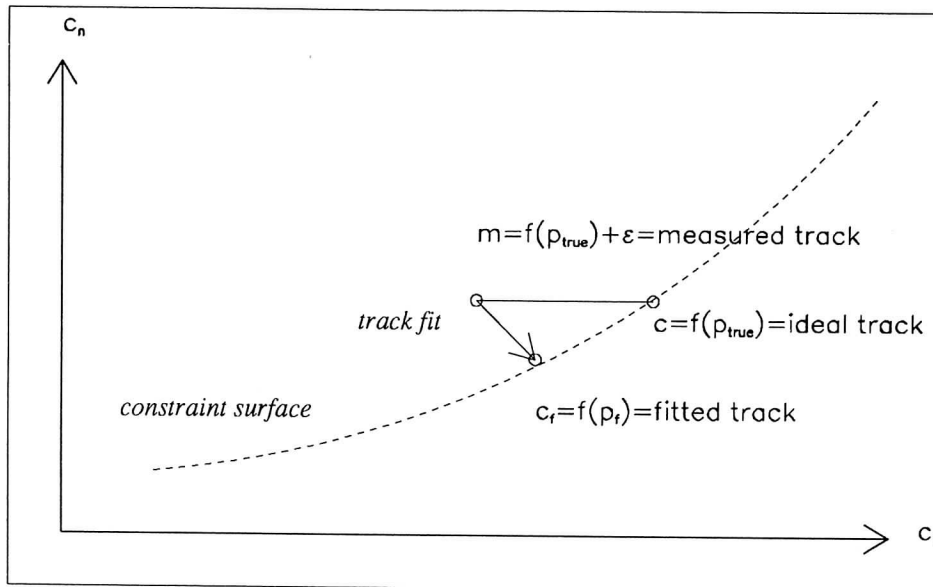


Abbildung 9.3: Der Track-Fit als Projektion auf die constraint surface

Die Matrix der Ableitungen  $\mathbf{A}$  und der Tensor der zweiten Ableitungen werden an der Stelle  $\tilde{\mathbf{p}}_{i-1}$  bestimmt. Diese Prozedur entspricht dem Fitzen einer Parabel an  $Q(\mathbf{p})$  an der Stelle  $\tilde{\mathbf{p}}_{i-1}$  und dem Berechnen des Parabelvertex.

Ist das Trackmodell nicht explizit bekannt, kann die Berechnung der zweiten Ableitungen sehr zeitaufwendig sein. In diesem Fall vernachlässigt man diesen Term. Die resultierende Formel ist dann die gleiche, die man erhält mit einer Linearisierung der Funktion  $\mathbf{f}$ :

$$\tilde{\mathbf{p}}_i = \tilde{\mathbf{p}}_{i-1} + (\mathbf{A}^T \mathbf{G} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{G} [\mathbf{m} - \mathbf{f}(\tilde{\mathbf{p}}_{i-1})]. \quad (9.9)$$

In der Praxis sind selten mehr als 3 Iterationen nötig, um dem Minimum von  $Q$  hinreichend nahe zu kommen. Die Kovarianzmatrix von  $\tilde{\mathbf{p}}$  wird dann gegeben durch:

$$\text{cov}(\tilde{\mathbf{p}}) = (\mathbf{A}^T \mathbf{G} \mathbf{A})^{-1}. \quad (9.10)$$

Der Wert von  $Q$  am Minimum, das sogenannte Chi-Quadrat des Fits, ist eine wichtige Testgröße. Im linearen Fall mit normalverteilten Fehlern ist  $Q$  exakt  $\chi^2$ -verteilt mit  $n - m$  Freiheitsgraden, wobei  $n$  die Zahl der linear unabhängigen Meßwerte ist und  $m$  die Zahl der bestimmten Parameter. Seine Wichtigkeit als Teststatistik besteht darin, daß man entscheiden

n	prob( $ p_i - \tilde{p}_i $ )(%)	Confidence level (%)
1	68.3	31.7
2	95.4	4.6
3	99.7	0.3

Tabelle 9.2: Konfidenzintervalle für normalverteilte Größen

kann, ob die Hypothese, der Vektor der Meßwerte  $\mathbf{m}$  entspricht einem Track, akzeptiert werden kann oder verworfen werden muß. Ist die Vielfachstreuung verstanden und sind die Detektorfehler optimal eingestellt worden, dann sind die Chi-Quadrat-Werte eines Fits in der Tat in guter Näherung  $\chi^2$ -verteilt. Da fast jede Art von Untergrund die Tendenz hat, zu grösseren Chi-Quadrat-Werten zu führen, ist ein *Cut* auf das Chi-Quadrat ein einfacher Weg, Fehler der Spurerkennung zu finden. Die Verluste durch einen solchen *Cut* kann man nach folgender Formel berechnen:

$$\alpha = \int_{\chi_c^2}^{\infty} g_k(x) dx = 1 - G_k(\chi_c^2), \quad (9.11)$$

wobei  $\chi_c^2$  der *Cut* ist,  $G_k(x)$  die kumulative  $\chi^2$ -Verteilung mit  $k$  Freiheitsgraden, und  $\alpha$  ist der Anteil an guten Tracks, den man durch den *Cut* verliert. Im linearen Fall mit normalverteilten Fehlern hat das Chi-Quadrat des Fits die schöne Eigenschaft, stochastisch unabhängig von den Parametern zu sein, so daß der  $\chi^2$ -*Cut* keinen Bias verursacht in der Menge der akzeptierten Tracks, die dann repräsentativ sind für die Gesamtmenge.

Die Komponenten des Vektors der Meßwerte  $\mathbf{m}$  können individuell getestet werden mit Hilfe der folgenden Teststatistik:

$$r_i = [m_i - f_i(\mathbf{p}_i)]/\sigma_i,$$

$$\sigma_i = \sqrt{[\mathbf{V} - \mathbf{A}(\mathbf{A}^T \mathbf{G} \mathbf{A})^{-1} \mathbf{A}^T]_{ii}}.$$

Diese Größen, die sogenannten *normalisierten Residuen* oder *Pulls*, sollten normalverteilt sein mit Mittelwert 0 und Varianz 1.

Parameter	Definition
$\omega$	vorzeichenbehaftete Krümmung, $\omega = \frac{1}{R \cdot sst}$
$\tan(\lambda)$	$= \frac{dz}{ds_{xy}} =$ Tangens des Dip-Winkels in der $s_{xy}z$ -Ebene
$\phi_0$	Emissionswinkel in der $xy$ -Ebene am Punkt größter Annäherung an den Ursprung in der $xy$ -Ebene
$d_0$	Impact-Parameter in der $xy$ -Ebene mit Vorzeichen, $d_0$ ist positiv, falls die $z$ -Achse von der Helix eingekreist wird
$z_0$	$z$ -Koordinate an der Stelle $x^2 + y^2 = d_0^2$

Tabelle 9.3: Parametrisierung der Teilchenspur

## 9.5 Track-Fit für die SIS $4\pi$ CDC

### 9.5.1 Parametrisierung der Helix

Der CDC Tracking-Detektor kann maximal 60 dreidimensionale Punkte für jede Spur eines elektrisch geladenen Teilchens liefern. Da das Magnetfeld innerhalb der CDC in guter Näherung homogen ist, sind die Spuren geladener Teilchen in guter Approximation Helix-Kurven. Im folgenden wird die Routine beschrieben, die einen simultanen Fit in den fünf Helix-Parametern ausführt. Der Code wurde adaptiert von Strategien, die beim ALEPH Detektor am CERN [NOR88] entwickelt wurden.

Die gewählte Parametrisierung nutzt die radiale Struktur der CDC aus. Ein Punkt  $(x, y, z)$  auf einer Helixkurve wird durch die polaren Koordinaten  $(r, \phi, z)$  beschrieben.  $\phi$  und  $z$  werden als Funktion von  $r$ , dem radialen Abstand in der  $xy$ -Ebene,  $r = \sqrt{x^2 + y^2}$ , ausgedrückt:  $\phi$  ist der azimutale Winkel, unter dem ein Punkt auf der Helixkurve liegt, und seine Abhängigkeit von  $r$  und den Helixparametern lautet:

$$\phi(r, \mathbf{h}) = \phi_0 + (ssr - sst) \frac{\pi}{2} + ssr \cdot \arcsin \left\{ \frac{\omega \cdot (r^2 - d_0^2)}{2r \cdot (d_0 \cdot \omega - sst)} + \frac{d_0}{r} \right\} \quad (9.12)$$

Die  $z$ -Koordinate

$$z(r, \mathbf{h}) = z_0 + \tan(\lambda) \cdot s_{xy}(r) \quad (9.13)$$

ist linear in der Größe

$$s_{xy}(r) = \frac{2}{\omega} \cdot \arcsin \left\{ \frac{\omega}{2} \sqrt{\frac{r^2 - d_0^2}{1 - s_{sr} \cdot d_0 \cdot \omega}} \right\}, \quad (9.14)$$

dem Bogenelement auf dem Kreis, der durch die Projektion der Helix auf die  $xy$ -Ebene entlang der  $z$ -Achse entsteht. Das Bogenelement wird vom Punkt größter Annäherung des Kreisbogens an den Koordinatenursprung aus gemessen. Dabei sind die Helixparameter  $\mathbf{h}$  (siehe Abbildung 9.4 und 9.5), gegeben durch

$$\mathbf{h} = (\omega, \tan(\lambda), \phi_0, d_0, z_0), \quad (9.15)$$

folgendermaßen definiert (siehe auch Tabelle 9.3):

- $s_{st}$  = 1, falls die Spur gegen Strahlrichtung im Uhrzeigersinn gekrümmt ist, sonst -1.
- $s_{sr}$  =  $s_{st}$ , falls der Punkt zur ersten Hälfte der Helix gehört, sonst  $-s_{st}$ .
- $\omega$  = vorzeichenbehaftete Krümmung,  $\omega = \frac{1}{R \cdot s_{st}}$ .
- $\tan(\lambda) = \frac{dz}{ds_{xy}}$  = Tangens des Dip-Winkels in der  $s_{xy}z$ -Ebene.
- $\phi_0$  = Emissionswinkel in der  $xy$ -Ebene am Punkt größter Annäherung an den Ursprung.
- $d_0$  = Impact-Parameter in der  $xy$ -Ebene mit Vorzeichen.
- $z_0$  =  $z$ -Koordinate an der Stelle  $x^2 + y^2 = d_0^2$ .

Falls Helixparameter im Code auftauchen sind sie gemäß der obigen Reihenfolge gespeichert. Die Vorzeichenkonvention ist:

- $\omega$  positiv  $\rightarrow$  die Spur ist vom Ursprung aus gesehen im Uhrzeigersinn gekrümmt
- $d_0$  positiv  $\rightarrow$  die  $z$ -Achse wird von der Helix umkreist.

Man beachte, daß die Vorzeichenkonvention für  $\omega$  *umgekehrt* zu der ist, die üblicherweise *mathematisch positiv* genannt wird. Die Fehler dieser Parameter werden aus den Fehlern für die Koordinaten bestimmt, die einzige Ausnahme sind Fehler aufgrund der Vielfachstreuung im Kammergas.

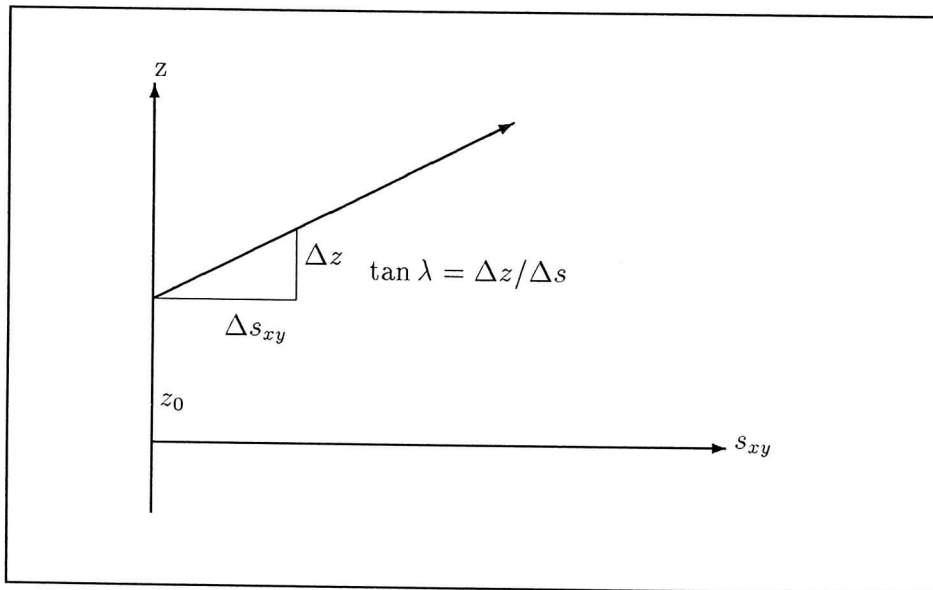


Abbildung 9.4: Wahl der Helixparameter in der  $s_{xy}z$ -Ebene.

### 9.5.2 Die Track-Fit-Strategie

Die Aufgabe des Track-Fit-Programms ist die Bestimmung der exakten Helix-Parameter, die Beseitigung schlecht gemessener Punkte und die Erkennung von *Klein-Winkel-Kinks*, aufgrund größerer Energieverluste infolge der Landau-Fluktuationen.

Für die Bestimmung der Helixparameter wird die Vielfachstreuung berücksichtigt, indem die Fehler an den Koordinaten entsprechend der Bogenlänge im CDC-Volumen vergrößert werden. Die Details des Trackfits werden in einem eigenen Abschnitt beschrieben. Hier soll nur die Logik im Zusammenhang mit dem Trackfit erläutert werden.

Da ein Trackfit eine ziemlich zeitaufwendige Aufgabe ist, wurde das Programm nach dem Motto geschrieben: *only try fancy tools if simple one fail*. Genauer gesagt, jeder der folgenden Teile wird nur dann aktiviert, falls der vorhergehende keinen akzeptablen Fit ergeben hat:

- Fit für den Track-Kandidaten ohne Modifikationen.
- Ausschluß von maximal 2 Punkten, entsprechend ihrem Beitrag zum  $\chi^2$  des gesamten Fits.
- Suche nach Klein-Winkel-Kinks.

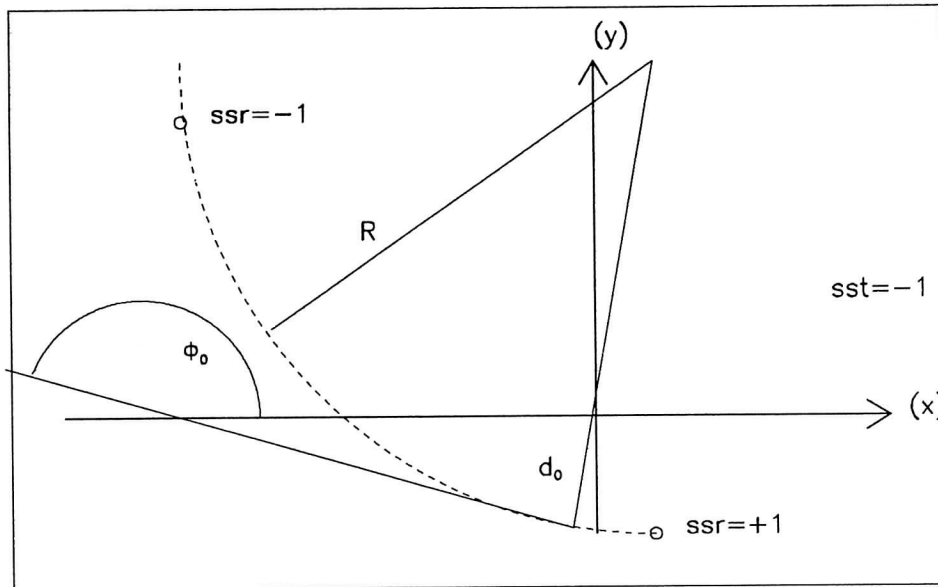


Abbildung 9.5: Wahl der Helixparameter in der  $xy$ -Ebene.

- Versuch, die  $\chi^2$ -Wahrscheinlichkeit durch Ausschluß von höchstens 2 Punkten mit einer kombinatorische Suche nach schlechten Punkten zu maximieren.

Falls keine der obigen Methoden zu einem akzeptablen Fit führt, wird der Track-Kandidat ohne Modifikation gespeichert.

Ausschluß eines einzelnen Punktes entsprechend seines Beitrags zum  $\chi^2$  des gesamten Fits heißt, daß der Punkt mit

$$\frac{(r\phi_{measured} - r\phi_{fitted})^2}{\sigma_{r\phi,measured}^2} + \frac{(z_{measured} - z_{fitted})^2}{\sigma_{z,measured}^2} = maximum \quad (9.16)$$

aus der Liste der Punkte, die angepaßt werden sollen, gelöscht wird. Diese Methode ist aber nicht sehr zuverlässig, da sie davon ausgeht, daß ein schlecht gemessener Punkt nur einen kleinen Effekt auf das Resultat des Fits hat.

Der endgültige Algorithmus zur Bestimmung von Kinks ist immer noch in der Diskussion. Zur Zeit wird eine Kink-Suche gemäß dem Vorschlag von Georg Stimpfl [STM91] benutzt. Dabei betrachtet man folgenden Ausdruck:

$$\Delta\chi^2 = \chi_{max}^2(r = \text{"worst kink point"}) - \chi_{min}^2(r = \text{"best kink point"})$$

Ist  $\Delta\chi^2$  größer als ein bestimmter Cut, so wird der Track-Kandidat an der Stelle in zwei Teile zerlegt, bei der das totale  $\chi^2$  am kleinsten war. Falls nötig werden bei beiden Tracks, die man auf diese Weise erhält, Punkte nach dem obigen Verfahren gelöscht.

Wird kein signifikanter Kink entdeckt, wird eine kombinatorische Suche nach schlechten Punkten ausgeführt, indem ein Punkt nach dem anderen ausgeschlossen und ein Helix-Fit ausgeführt wird.

Der Track-Fit besteht nun darin, die Werte  $\mathbf{h} = (\omega, \tan(\lambda), \phi_0, d_0, z_0)$  der Helix zu bestimmen. In der derzeitigen Prozedur können zwei aufeinanderfolgende Teile unterschieden werden:

1. Schnelle Bestimmung der Helix-Parameter. Einmal wird ein Kreisfit auf den in die  $xy$ -Ebene projizierten Track angewendet und dann ein Geradenfit in der  $s_{xy}z$ -Ebene. Auf dieser Ebene werden Effekte der Vielfachstreuung noch nicht berücksichtigt.
2. Eine Newton-Iteration in allen fünf Helix-Parametern, um das  $\chi^2$  zu verbessern. An dieser Stelle werden die Fehler in der CDC einer speziellen Behandlung unterzogen, um die Effekte der Vielfachstreuung im Kammergas zu berücksichtigen.

### 9.5.3 Ein schneller Kreisfit

Der hier benutzte Kreisfit wurde von N. I. Chernov und G. A. Ososkov [CHE84] entwickelt. Das Problem ist, den *besten* Kreis zu finden, der  $n$  gegebene Punkte  $(x_i, y_i)$  in der  $xy$ -Ebene beschreiben kann.

Eine Lösung besteht darin, den mittleren quadratischen Abstand der gegebenen Punkte zu einem Kreis, gegeben durch  $(x-a)^2 + (y-b)^2 = R^2$ , zu minimieren, das heißt, das folgende Funktional zu minimieren:

$$L(a, b, R) = \sum_{i=1}^n \rho_i^2, \quad (9.17)$$

$$\rho_i = \sqrt{(x_i - a)^2 + (y_i - b)^2} - R. \quad (9.18)$$

Wegen der nichtlinearen Abhängigkeit von  $\rho_i$  bezüglich der Kreisparameter führt diese Methode zu umfangreichen Rechnungen. Dieses Problem wird vermieden, indem man statt  $L(a,b,R)$  das folgende Funktional minimiert:

$$M(a, b, R) = \sum_{i=1}^n ((x_i - a)^2 + (y_i - b)^2 - R^2)^2 \quad (9.19)$$

$$= 4R^2 \sum_{i=1}^n \rho_i^2 \left(1 + \frac{\rho_i}{2R}\right)^2. \quad (9.20)$$

In diesem Fall führt die Minimierung zu einem linearen Gleichungssystem. Das macht die Methode sehr schnell. Sie hat aber im Vergleich zur ersten Methode einen wesentlichen Nachteil: Ist  $\rho_i \ll R$  dann gilt die folgende Approximation:

$$M(a, b, R) \approx 4R^2 \sum_{i=1}^n \rho_i^2 = 4R^2 L(a, b, R). \quad (9.21)$$

Dies zeigt, daß es für diese Methode "vorteilhafter" ist  $R$  anstatt  $L(a, b, R)$  zu minimieren.

Die Lösung von N. I. Chernov und G. A. Ososkov beruht nun darauf, daß  $M(a, b, R) \approx 4R^2 L(a, b, R)$ . Sie schlagen daher vor, folgendes Funktional zu minimieren:

$$K(a, b, R) = M(a, b, R)R^{-2} \quad (9.22)$$

$$= \sum_{i=1}^n \left( \frac{x_i^2 + y_i^2}{R} - 2\frac{a}{R}x_i - 2\frac{b}{R}y_i + \frac{x_i^2 + y_i^2 - R^2}{R} \right)^2. \quad (9.23)$$

Setzt man die Ableitungen von  $K(a, b, R)$  bezüglich  $a$ ,  $b$  und  $R$  Null, so stellt man fest, daß das Problem reduziert wurde einmal auf die Lösung eines linearen Gleichungssystems in  $a$  und  $b$  und dann die Bestimmung der Wurzeln eines Polynoms vierter Ordnung, um  $R$  zu erhalten. Ihnen gelang es, das Problem eines Kreisfits auf die Bestimmung der Wurzel eines Polynoms vierter Ordnung zu reduzieren. Die Schönheit ihrer Methode besteht darin, daß die Koeffizienten des Polynoms nur einmal ausgerechnet werden müssen. Diese Methode wurde von den Autoren in einem Programm *Circle* implementiert. Das ursprüngliche Programm *Circle* wurde an drei Punkten modifiziert:

1. Es wurden die Meßfehler berücksichtigt, indem folgende Ersetzung vorgenommen wurde:

$$\frac{1}{N_{points}} \sum_{i=1}^N x_i^k y_i^k \rightarrow \frac{\sum_{i=1}^N \frac{x_i^k y_i^k}{\sigma_{r\phi, i}^2}}{\sum_{i=1}^N \frac{1}{\sigma_{r\phi, i}^2}} \quad (9.24)$$

2. Die Newton-Iteration für den Kreisfit wurde stabilisiert. Im ursprünglichen Programm wurde die Iteration terminiert, falls die zu

minimierende Variable  $|Y(X)|$  nach einer Iteration wieder anwuchs. Diese Bedingung wurde ersetzt durch die folgende:

$$|Y^{(n+1)}(X^{(n+1)})| > |Y^{(n)}(X^{(n)})| \quad (9.25)$$

$$\rightarrow |Y^{(n+1)}\left(\frac{X^{(n)} + X^{(n+1)}}{2}\right)| > |Y^{(n)}(X^{(n)})| \quad (9.26)$$

Dabei ist  $(n)$  die Anzahl der Iterationen. Diese kleine Modifikation erhöhte die Robustheit des Fits, dessen Effizienz ohnehin im Bereich von 99.9% war.

3. Ein expliziter Schutz gegen den Grenzwert einer Geraden wurde eingebaut:

$$|\omega| \rightarrow |\omega| + 10^{-10} \quad (9.27)$$

#### 9.5.4 Der Helix-Fit (CALL CFTHHEL)

Der Helix-Fit besteht aus drei aufeinanderfolgenden Teilen:

- ein Kreis-Fit in der  $xy$ -Ebene nach der Methode von Chernov und Ososkov
- ein Geraden-Fit in der  $s_{xy}z$ -Ebene
- eine simultane Newton-Iteration in allen Helix-Parametern

Die Newton-Iteration ist optional.

Die Resultate des Kreisfits werden dann benutzt, um die Größen  $s_{xy}(r)$  zu bestimmen, die im Geradenfit benötigt werden. Von den angepaßten Werten  $a$ ,  $b$  und  $R$  erhält man leicht  $\omega$ ,  $\phi_0$  und  $d_0$ . Damit erhält man die Werte  $s_{xy,i}$ :

$$s_{xy,i} = \frac{2}{\omega} \arcsin \left\{ \frac{\omega}{2} \sqrt{\frac{r_i^2 - d_0^2}{1 - \text{ssr} \cdot d_0 \cdot \omega}} \right\}, \quad (9.28)$$

$$r_i = \sqrt{x_i^2 + y_i^2}. \quad (9.29)$$

Diese Werte erlauben es dann, einen Geradenfit in der  $s_{xy}z$ -Ebene auszuführen und die Parameter  $z_0$  und  $\tan(\lambda)$  zu bestimmen gemäß:

$$z = z_0 + \tan(\lambda)s_{xy}. \quad (9.30)$$

Der Helix-Fit besteht nun in einer Minimierung des Abstandes in der  $\phi z$ -Ebene zwischen den gemessenen Punkten und der Helixkurve. Der  $\chi^2$ -Ansatz hat damit folgende Gestalt:

$$\chi^2 = \sum_{i=1}^{N_{points}} \frac{(r\phi_i^{fit} - r\phi_i^{measured})^2}{\sigma_{\phi,i}^2} + \sum_{i=1}^{N_{points}} \frac{(z_i^{fit} - z_i^{measured})^2}{\sigma_{z,i}^2}. \quad (9.31)$$

Um die Parameterwerte zu finden, die das  $\chi^2$  minimieren, wird eine Newton-Iteration in allen 5 Helix-Parametern gemäß der Standardformel ausgeführt:

$$\mathbf{h}_{\alpha}^{(n+1)} = \mathbf{h}_{\alpha}^{(n)} + (\mathbf{E}_{\alpha\beta})^{-1} \mathbf{G}_{\beta}. \quad (9.32)$$

Dabei ist  $\mathbf{h}^{(n)}$  der Vektor, der die Helixparameter nach n Iterationen enthält:

$$\mathbf{h} = (\omega, \tan(\lambda), \phi_0, d_0, z_0). \quad (9.33)$$

$\mathbf{E}$  ist die inverse Kovarianzmatrix, die Fehlermatrix, und kann folgendermaßen berechnet werden:

$$E_{\alpha\beta} = \frac{1}{2} \frac{\partial^2 \chi^2}{\partial h_{\alpha} \partial h_{\beta}}. \quad (9.34)$$

$\mathbf{G}$  ist der Gradientenvektor und gegeben durch:

$$G_{\beta} = -\frac{1}{2} \frac{\partial \chi^2}{\partial h_{\beta}}. \quad (9.35)$$

Die analytischen Ausdrücke von  $E_{\alpha\beta}$  und  $G_{\beta}$  sind in Appendix A angegeben. Da die Fits in den Projektionen so genau sind, kann die Anzahl der fünfdimensionalen Iterationen auf eins beschränkt werden.

Für die Zwecke einer *bad  $\chi^2$  diagnostics* hat die Fitroutine noch zwei *Entry-Punkte*, über die man Information über die Konsistenz der angepaßten Koordinaten mit den Fitparametern erhalten kann. Die Routine CFTHED liefert die Differenzen in  $r\phi$  und  $z$  zwischen den Datenpunkten und der angepaßten Kurve. Der Entry CFTEC ergibt den individuellen Beitrag eines Punktes zum  $\chi^2$  des gesamten Fits.

### 9.5.5 Helixfit mit Vielfachstreuung (CALL CMS-HEL)

Diese Routine ist eine Weiterentwicklung des obigen Helixfit-Routine. Sie besteht aus:

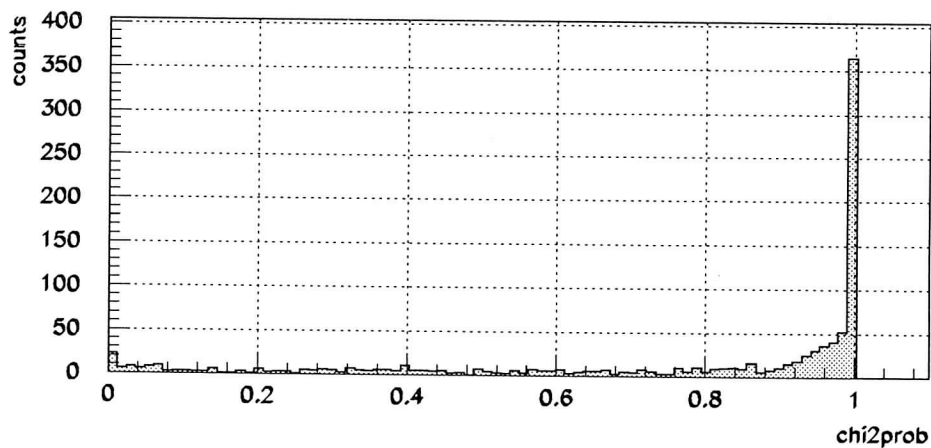
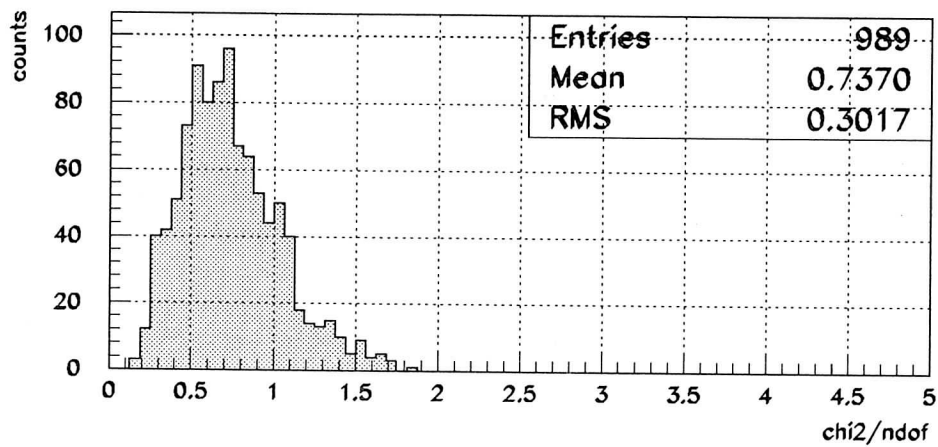


Abbildung 9.6:  $\chi^2$ -Verteilung und  $\chi^2$ -Wahrscheinlichkeit des Trackfits: Die  $\chi^2$ -Verteilung zeigt den richtigen Verlauf, das Maximum liegt etwas unter 1. Dementsprechend weist die Wahrscheinlichkeitsverteilung für  $\chi^2$ , die flach sein soll, einen Peak bei 1 auf. Ansonsten liegt aber Gleichverteilung vor.

1. einem vorläufigen Kreisfit in der  $xy$ -Ebene
2. einem vorläufigen Geradenfit in der  $s_{xy}z$ -Ebene
3. eine Berechnung der Fehler aufgrund der Vielfachstreuung im Kammergas
4. einem zweiten Kreisfit

5. einem zweiten Geradenfit

6. eine simultane Newton-Iteration in allen Helixparametern

Der letzte Schritt ist wieder optional. Alle Berechnungen außer für die Vielfachstreuung werden ganz genauso wie in CFTHL ausgeführt.

Die Fehler für die Vielfachstreuung wurden von der Standardformel

$$\frac{\sigma}{l_{track}} \simeq \frac{1}{\sqrt{3}} \frac{0.0141}{p/Z[GeV]} \sqrt{\frac{l_{track}}{l_{rad}}} \quad (9.36)$$

erhalten [PAR90], wobei  $l_{track}/l_{rad}$  die Tracklänge ist, angefangen bei der ersten Koordinate, gemessen in Strahlungslängen. Die Fehler von der Vielfachstreuung werden dann quadratisch zu den Meßfehlern der Koordinaten addiert.

### 9.5.6 Resultate des Trackfits

Um einen Eindruck von der Leistungsfähigkeit des Trackfits zu bekommen, wurden tausend Protonen-Ereignisse mit Impulsen zwischen 300 und 1000 MeV erzeugt. Dazu wurde GEANT benutzt mit einem eigenen Ereignisgenerator, der Protonen gleichverteilt im Impuls und in der Richtung innerhalb der CDC-Akzeptanz produziert.

Abbildung 9.6 zeigt oben die reduzierte  $\chi^2$ -Verteilung der Track-Fits. Man erwartet einen Peak bei 1, wenn die Anzahl der Freiheitsgrade  $ndof$  und die Fehler der Hits richtig sind. Bei den Simulationen, mit denen der Trackfit getestet werden sollte, wurde ein mittlerer Fehler von  $300\mu m$  für die x- und y-Koordinate und ein mittlerer Fehler von 0.5 cm für die z-Koordinate angenommen. Die Anzahl der Freiheitsgrade für den Helixfit wie er im Rahmen dieser Arbeit entwickelt wurde ist  $2 \cdot \text{Anzahl der Hits} - 5$ . Das in Abbildung 9.6 oben gezeigte Histogramm für die vom Helix-Fit erhaltenen Werte von  $\chi^2$  weist den Verlauf einer  $\chi^2$ -Verteilung auf. Die Verteilung der reduzierten  $\chi^2$ -Werte sollte bei 1 ein Maximum aufweisen. Die  $\chi^2$ -Werte des Helixfits weisen eine systematische Abweichung zu kleineren Werten auf. Die kumulative  $\chi^2$ -Verteilung ( $\chi^2$ -Wahrscheinlichkeit) sollte gleichverteilt sein: Die Verteilung der  $\chi^2$ -Wahrscheinlichkeit, zu sehen in Abbildung 9.6 unten, ist hinreichend flach mit einem Peak bei 1, der anzeigt, daß die Werte von  $\chi^2$  systematisch etwas zu klein sind.

Die Größe  $\chi^2$  ist allerdings nur eine globale Größe zur Beurteilung der Qualität eines Fits. Wenn aber über die Form des Funktionals, mit

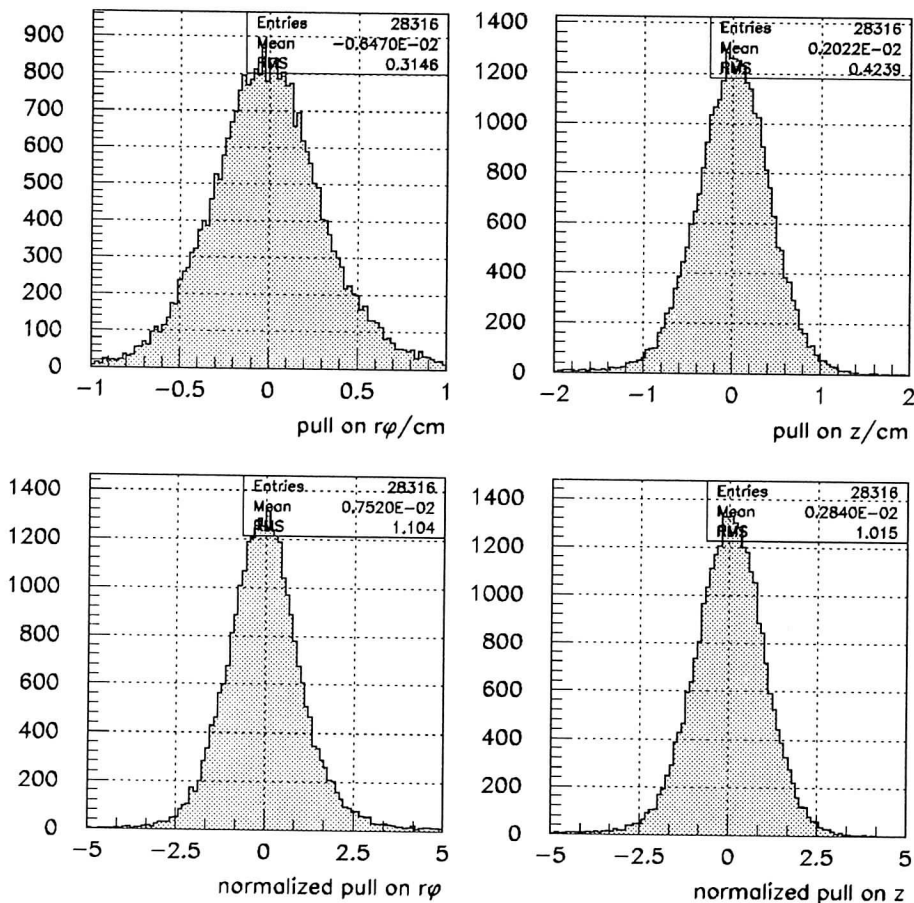


Abbildung 9.7: Pulls und normalisierte Pulls der Trackhits

dem der  $\chi^2$ -Ansatz aufgestellt wird, nicht ohne weiteres Klarheit besteht, beziehungsweise aus Gründen der Effizienz in dieser Frage Kompromisse eingegangen werden müssen, ist eine *Pullanalyse* nützlich, um Vertrauen in die erhaltenen Parameter zu gewinnen. Einen besseren Eindruck von den Fehlern der Hits erhält man durch die Häufigkeitsverteilung der Pulls, also der Abstände der Hits von der angepaßten Spur. Abbildung 9.7 zeigt in der ersten Reihe die Pulls. Sie sind normalverteilt mit Zentrum 0. Die Breite der Verteilung sagt etwas über den mittleren Fehler der Hits aus. Die angenommenen mittleren Fehler von  $300 \mu\text{m}$  in der  $x$ - und  $y$ -Koordinate und  $0.5 \text{ cm}$  in der  $z$ -Koordinate können reproduziert werden. Die Abweichungen sind darauf zurückzuführen, daß im Helixfit die Fehler

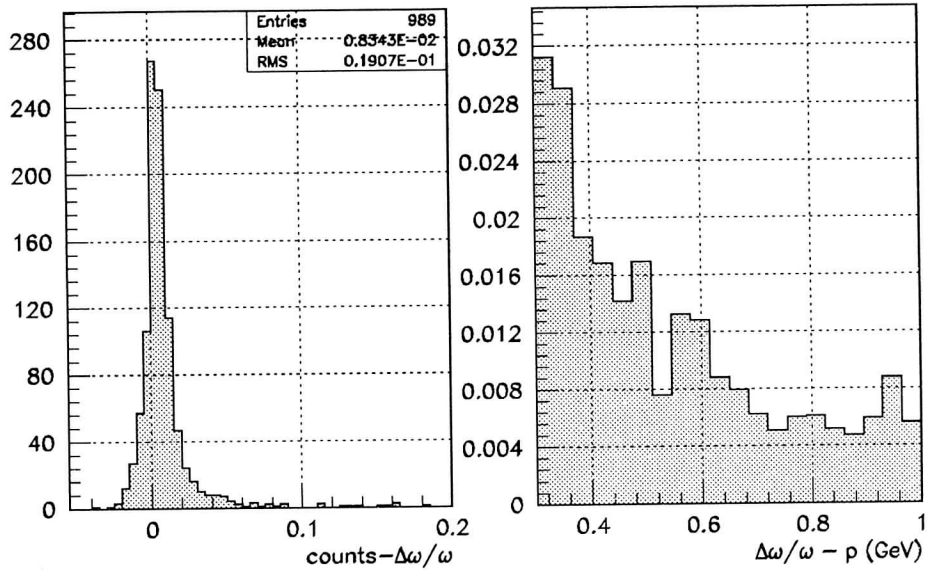


Abbildung 9.8: Auflösungen des Trackfits für die Krümmung des Tracks (links) und dessen Abhängigkeit vom Teilchenimpuls  $p$  (rechts).

aufgrund der Vielfachstreuung berücksichtigt wurden. Nimmt man die Breite dieser Verteilungen und normiert die Pulls damit, so erhält man die beiden unteren Verteilungen in Abbildung 9.7. Die normierten Pulls sollten auch normalverteilt sein mit Zentrum 0 und Breite 1. Die Pulls zeigen also ein statistisch korrektes Verhalten. Daher ist der Helixfit in der Praxis einsetzbar.

Vertrauen in die durch den Helixfit bestimmten Parameter gewinnt man durch Simulationen, die es ermöglichen, die berechneten Trackparameter mit den generierten zu vergleichen. Abbildung 9.9 zeigt in der ersten Spalte die Pulls der Trackparameter  $\mathbf{P} = (\omega \propto \frac{1}{p_T}, \tan(\lambda), \phi_0, d_0, z_0)$ , das heißt die Größe

$$\mathbf{P}^{\text{fitted}} - \mathbf{P}^{\text{generated}}$$

ist dort histogrammiert. Diese Größen sollten normalverteilt sein mit Zentrum 0. Die Plots zeigen auch dieses Verhalten. Wichtig ist, daß kein Bias in den bestimmten Parametern festgestellt werden kann außer bei der Krümmung  $\omega$ , also dem transversalen Impuls, für den normalerweise etwas kleinere Werte als die generierten bestimmt werden. Dies liegt am Energieverlust.

Abbildung 9.9 zeigt in der zweiten Spalte die Auflösung der gefit-

teten Helixparameter ( $\omega \propto \frac{1}{p_T}, \tan(\lambda), \phi_0, d_0, z_0$ ) in Abhängigkeit vom Teilchenimpuls. Die besten Resultate werden für Teilchenimpulse größer als 500 MeV/c erreicht. Das liegt daran, daß für Teilchen mit kleineren Impulsen Vielfachstreuung eine größere Rolle spielt. Damit ist das Anwachsen von  $\Delta\omega$  für kleiner werdende Teilchenimpulse zu erklären. Als Folge der überschätzten Krümmung wird auch der Emissionswinkel leicht überschätzt, während die anderen Größen nur eine schwache Abhängigkeit vom Betrag des Teilchenimpulses  $\mathbf{p}$  zeigen.

Außerdem ist es wichtig zu sehen, inwieweit den Fehlern aus dem Trackfit zu trauen ist. Um dies zu untersuchen, wurden die normierten Pulls berechnet:

$$\frac{(\mathbf{P}^{fitted} - \mathbf{P}^{generated})}{\sigma \mathbf{P}}$$

Dabei ist  $\sigma \mathbf{P}$  die Wurzel aus dem korrespondierendem Diagonalelement in der Fehlermatrix, die man vom Helixfit erhält. Die erwartete Verteilung für die Pulls ist eine Gaußverteilung mit Zentrum 0 und einer Breite von 1. In Abbildung 9.10 sind die normierten Pulls für die fünf Helixparameter, wie sie im Trackfit benutzt werden, in der ersten Spalte zu sehen. Die normierten Pulls der Trackparameter zeigen das geforderte Verhalten. Für die Spurkrümmung  $\omega$  kommt dagegen der Bias aufgrund des Energieverlustes noch deutlicher zum Vorschein. Die Fehler selbst sind wieder  $\chi^2$ -verteilt wie in Abbildung 9.10, zweite Spalte, zu sehen ist. Ihre Werte zeigen, daß der Helixfit zufriedenstellend arbeitet.

Die Abbildung 9.8 zeigt, daß die Krümmung einer Spur mit einer Genauigkeit von etwa 2% rekonstruiert wird. Der Bias in der Bestimmung der Spurkrümmung  $\omega$  tritt auch hier als Ausläufer des Peaks nach positiven Werten auf. Gleichzeitig wird aber deutlich, daß dieser Bias für die Frage der Auflösung vernachlässigt werden kann. Diese Größe zeigt auch eine starke Abhängigkeit vom Betrag des Teilchenimpulses  $\mathbf{p}$ , was auf das Verhalten von  $\Delta\omega$  zurückgeführt werden kann.

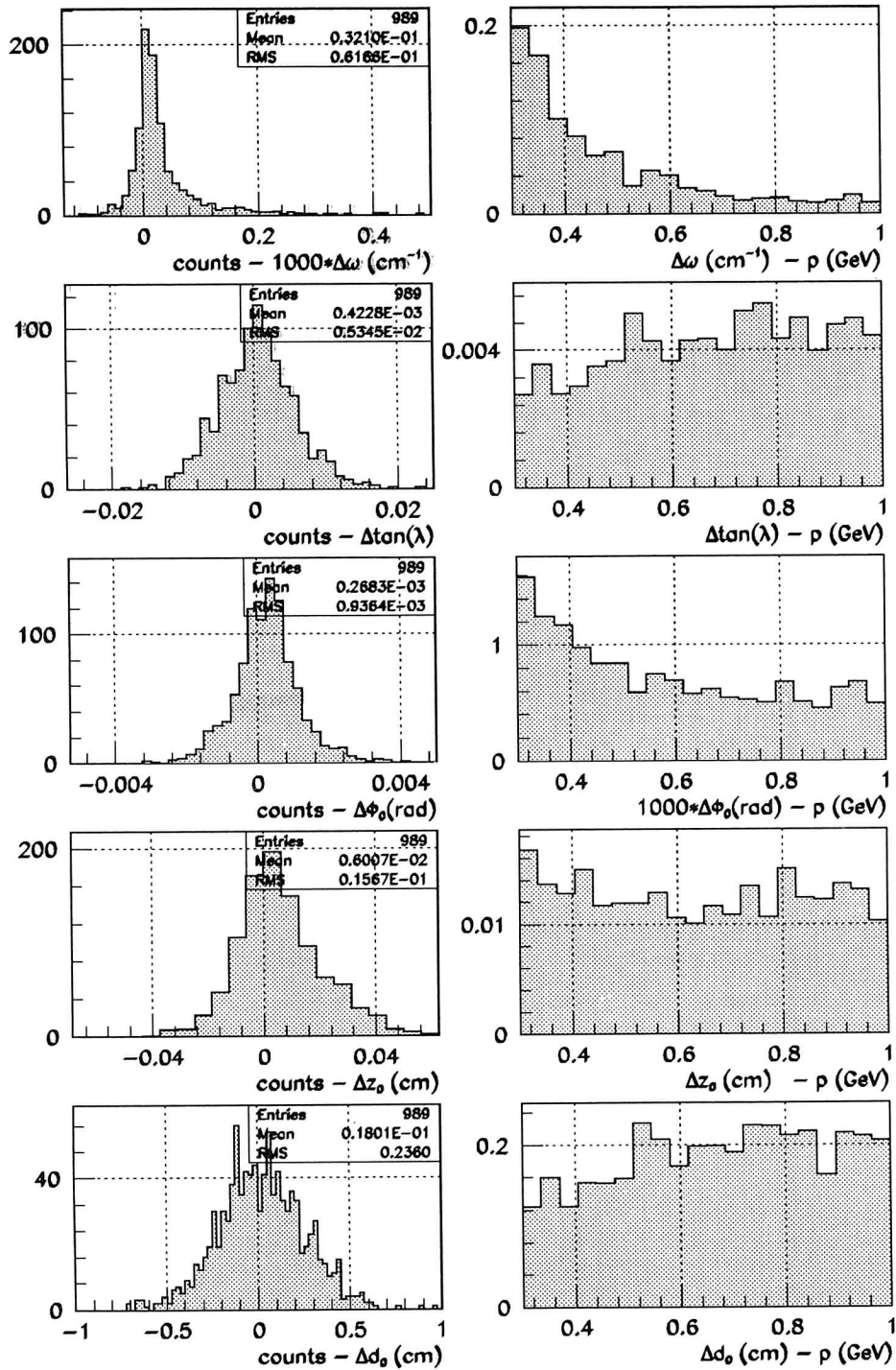


Abbildung 9.9: Nichtnormalisierte Pulls der Trackparameter (links) und deren Verhalten mit dem Teilchenimpuls  $p$ .

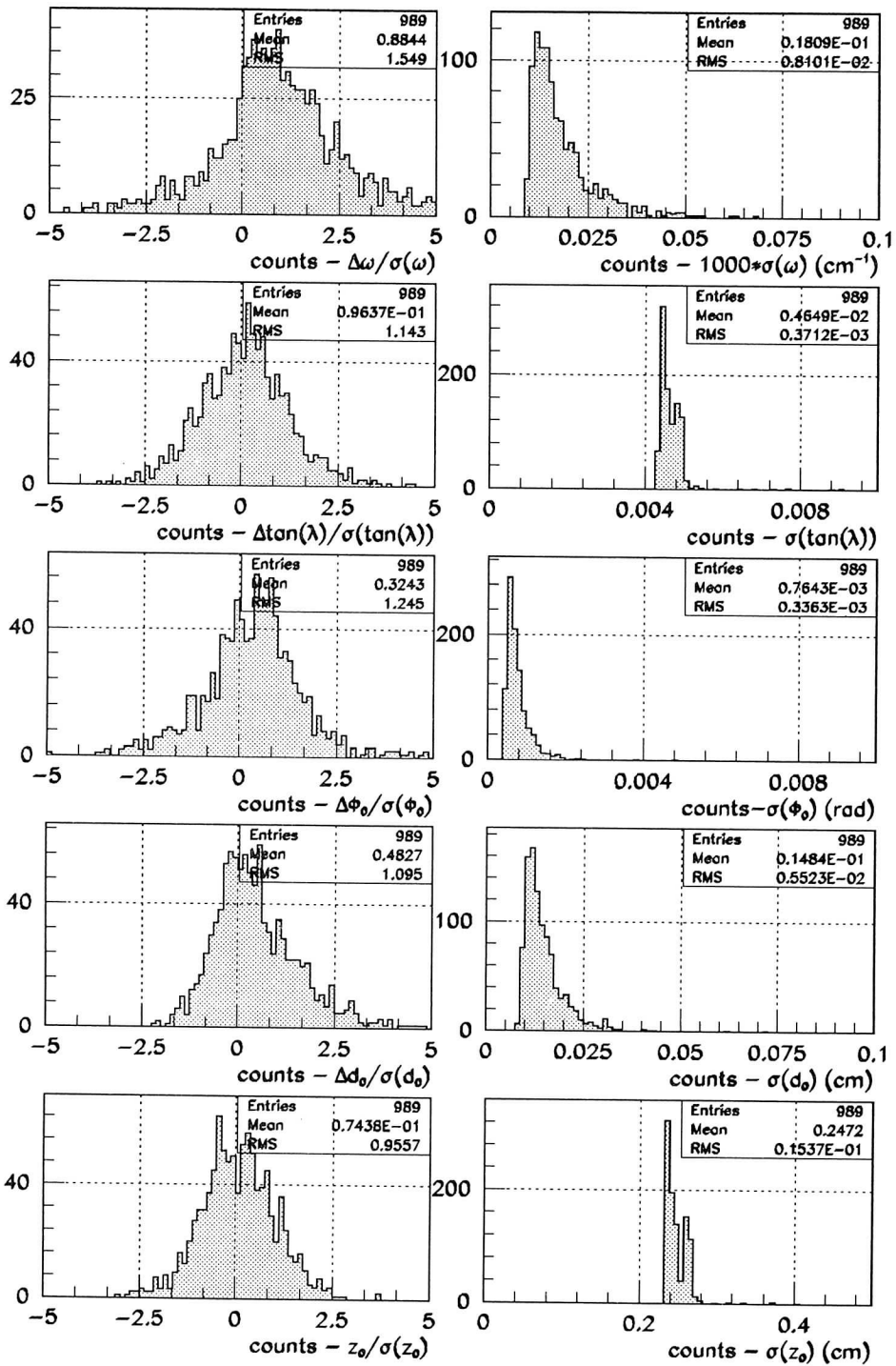


Abbildung 9.10: Normalisierte Pulls der Trackparameter (linke Spalte) und die Verteilung der Fehler der Trackparameter (rechte Spalte).

# Kapitel 10

## Vertex-Fitting

*Hier sind Drachen.*

– ALTE LANDKARTE

### 10.1 Konzepte

Die Aufgabe des Vertex-Fits ist genau wie die des Track-Fits zweifach: Einmal kann die Präzision der Track-Parameter verbessert werden durch die zusätzliche Zwangsbedingung, daß alle Tracks von einem *Common Vertex* kommen müssen; zweitens ist es möglich, indem man nach Tracks sucht, die offensichtlich nicht vom Common Vertex kommen, sekundäre Zerfälle zu finden.

Wegen der hohen Multiplizität in einer relativistischen Schwerionenkollision sollte die Rechenzeit für das Vertexfitting nicht schneller als linear mit der Anzahl der Tracks anwachsen. Im Vertexfit werden die vom Trackfit bestimmten Trackparameter als virtuelle Meßwerte betrachtet. Die Trackparameter verschiedener Tracks können als stochastisch unabhängig angesehen werden. Die Parameter, die bestimmt werden müssen, sind der Vertexort und die Impulsvektoren aller Tracks am Ort des Common Vertex (Abbildung 10.1).

### 10.2 Vertex-Fit für die SIS $4\pi$ CDC

Der Code stützt sich auf Strategien, die am TASSO [SAX85] entwickelt wurden. Ziel ist es, den Punkt und die dazugehörige Fehlermatrix zu finden, der den Abstand zu einer gegebenen Menge von *tracks*, von

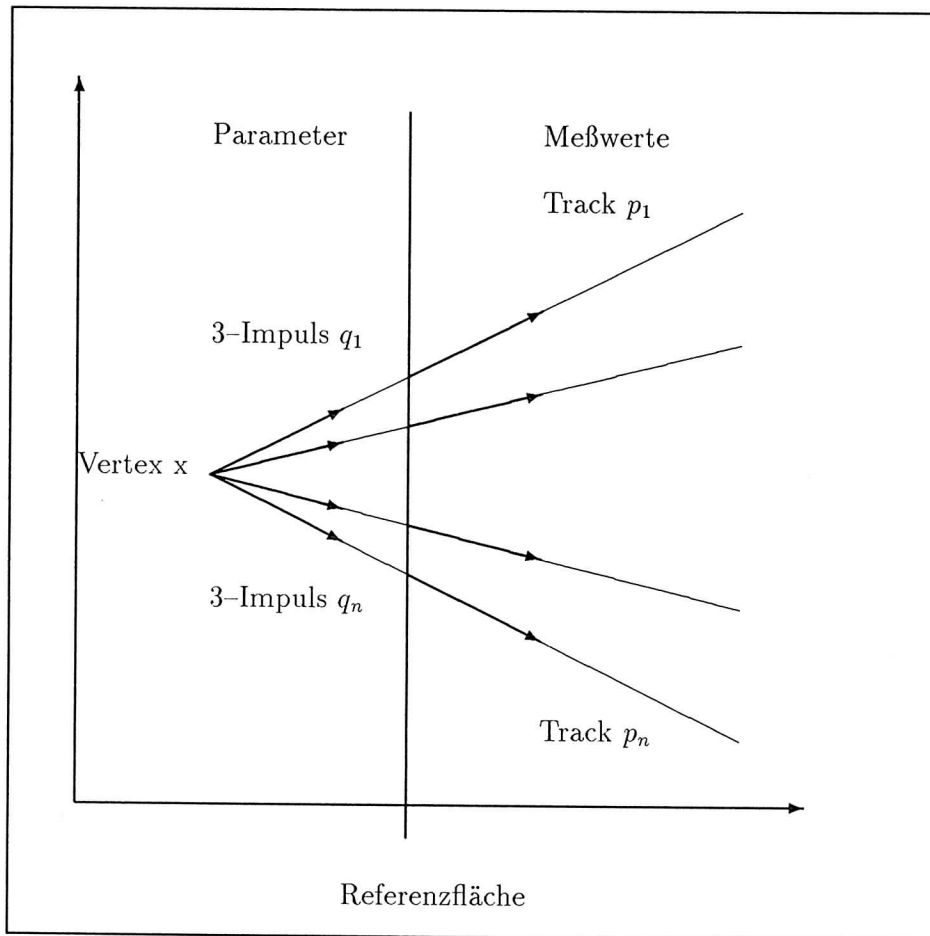


Abbildung 10.1: Meßwerte und Parameter beim Vertex-Fit

denen angenommen wird, daß sie von einem *common vertex* kommen, minimiert.

Ein wichtiger Aspekt des Vertex-Fits ist, daß sein  $\chi^2$  dazu benutzt werden kann zu überprüfen, ob die *tracks* von einem *common vertex* kommen, indem man es mit  $\sum \chi^2$  der einzelnen Tracks vergleicht. Da keine Beschränkung in der Zahl der Tracks besteht, kann diese Methode auch dazu benutzt werden, den *Primary Vertex* zu fitten.

Der Fit besteht in einer Newton-Iteration. Die Track-Parameter werden benutzt, um die Abstände  $d(t)_i$  (Abbildung 10.3) und  $d(z)_i$  (Abbildung 10.2) jedes Tracks  $i$  vom *Common Vertex* ( $A, B, C$ ) in der  $(x, y)$ -

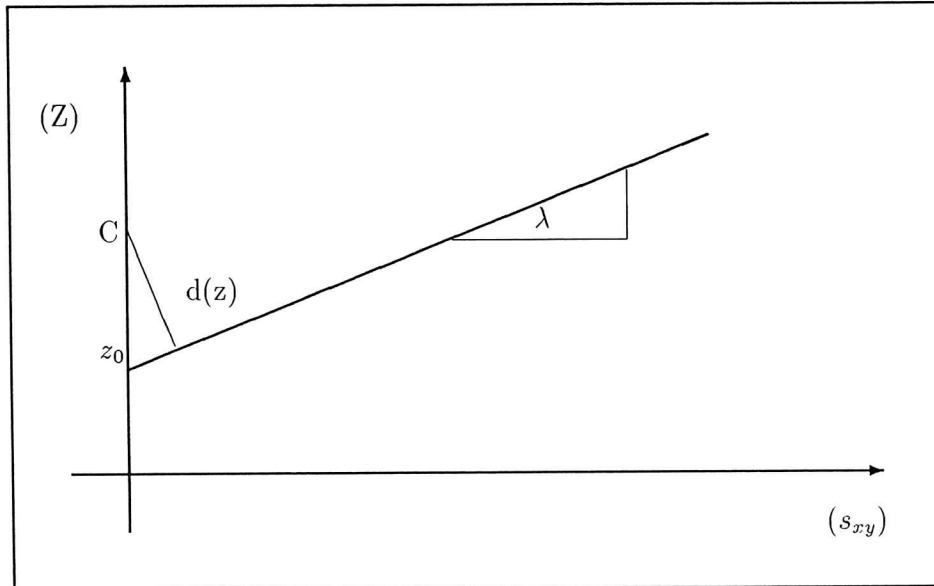


Abbildung 10.2: Wahl der Parameter für den Vertex-Fit in der  $s_{xy}z$ -Ebene

und  $z$ -Ebene zu minimieren. Dabei sind:

$$d(t)_i = s - \rho_i \quad (10.1)$$

$$s = \sqrt{(A - x_{ci})^2 + (B - y_{ci})^2} \quad (10.2)$$

und

$$d(z)_i = C - z_{0i} - \rho_i \lambda_i \psi \quad (10.3)$$

$$\psi = 2 \arcsin \left\{ \frac{\sqrt{(x_i - x_{0i})^2 + (y_i - y_{0i})^2}}{2\rho_i} \right\} \quad (10.4)$$

Die Parameter sind:

- $\rho_i$  = Krümmungsradius des projizierten Tracks  $i$
- $x_{ci}$  =  $x$ -Koordinate vom Mittelpunkt des projizierten Tracks
- $y_{ci}$  =  $y$ -Koordinate vom Mittelpunkt des projizierten Tracks

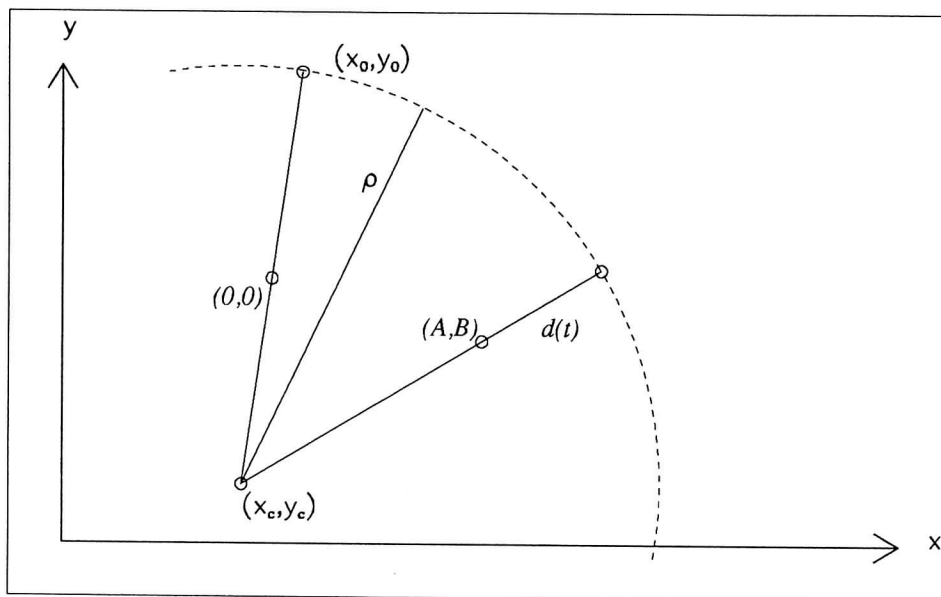


Abbildung 10.3: Wahl der Parameter für den Vertex-Fit in der  $xy$ -Ebene

- $x_i$  = x-Koordinate des Punktes auf dem Track, der am nächsten zu  $(A,B)$  liegt =  $x_{ci} + \rho_i \cdot (A - x_{ci})/s$
- $y_i$  = y-Koordinate des Punktes auf dem Track, der am nächsten zu  $(A,B)$  liegt =  $y_{ci} + \rho_i \cdot (B - y_{ci})/s$
- $z_{0i}$  = z-Koordinate des Punktes auf dem Track der am nächsten zum Ursprung in der  $(x,y)$ -Ebene liegt
- $x_{0i}$  = x-Koordinate des Punktes auf dem Track der am nächsten zum Ursprung in der  $(x,y)$ -Ebene liegt
- $y_{0i}$  = y-Koordinate des Punktes auf dem Track der am nächsten zum Ursprung in der  $(x,y)$ -Ebene liegt
- $\lambda_i$  = Tangens des Helix Dip-Winkels

Der Fit besteht darin, die Werte

$$\mathbf{v} = \begin{pmatrix} A \\ B \\ C \end{pmatrix} \quad (10.5)$$

zu finden, die ein  $\chi^2$  minimieren, das durch die folgende Summe über die  $N$  Tracks gebildet wird:

$$\chi^2 = \sum_{i=1}^N \left\{ \frac{d^2(t)_i}{\epsilon^2(t)_i} + \frac{d^2(z)_i}{\epsilon^2(z)_i} \right\} \quad (10.6)$$

Für Tracks, die aus der Nähe des geometrischen Zentrums kommen, werden die Fehler in transversaler Richtung und in  $z$ -Richtung gut durch die Fehler in  $d_0$  und  $z_0$ , wie sie durch den Track-Fit erhalten werden, approximiert. Stehen gefittete Fehler nicht zur Verfügung, können sie nach Glückstern [GLU63] abgeschätzt werden.

Die übliche Newton-Iteration wird dann angewendet: In jedem Iterationsschritt wird eine Korrektur zum Vertex bestimmt,

$$\mathbf{v}^{k+1} = \mathbf{v}^k - \mathbf{G}^{-1} \mathbf{g}, \quad (10.7)$$

bis die Änderung in  $\chi^2$  unter einen vorgegebenen Wert fällt.

Die Ableitungen von  $\chi^2$ ,

$$g_\alpha = \frac{\partial \chi^2}{\partial v_\alpha}, \quad G_{\alpha\beta} = \frac{\partial^2 \chi^2}{\partial v_\alpha \partial v_\beta}, \quad (10.8)$$

wurden explizit berechnet (Appendix B). Dabei wurde für  $d(z)$  die Approximation  $\psi = \sin(\psi)$  verwendet.

Wurde Konvergenz erreicht, erhält man die Kovarianzmatrix einfach durch  $2 \mathbf{G}^{-1}$ .

### 10.3 Resultate des Vertex-Fits

Um den Vertex-Fit zu testen, wurden mit GEANT einfache Ereignisse erzeugt. Mit einem eigenen Ereignisgenerator wurde eine bestimmte feste Anzahl von Protonen mit gleichverteiltem azimuthalen und polaren Emissionswinkeln und flachem Impulsspektrum in den Akzeptanzbereich der zentralen Driftkammer emittiert. Bei den Teilchenspuren wurde ein mittlerer Fehler  $\sigma_{xy}$  von  $300 \mu\text{m}$  und ein mittlerer Fehler  $\sigma_z$  von  $0.5 \text{ cm}$  der Hits berücksichtigt. Die von der adaptiven Hough-Transformation gefundenen und dann angepaßten Spuren waren die Eingabe für den darauffolgenden, oben beschriebenen, Vertex-Fit. Der Fit war der bereits beschriebene Helix-Fit unter Berücksichtigung der Fehler infolge der Vielfachstreuung.

Der Vertex wurde dann rekonstruiert unter Berücksichtigung der Fehler der Track-Parameter, wie sie vom Track-Fit bestimmt werden. Abbildung 10.4 zeigt das Resultat, wobei die normierten Pulls die Differenz

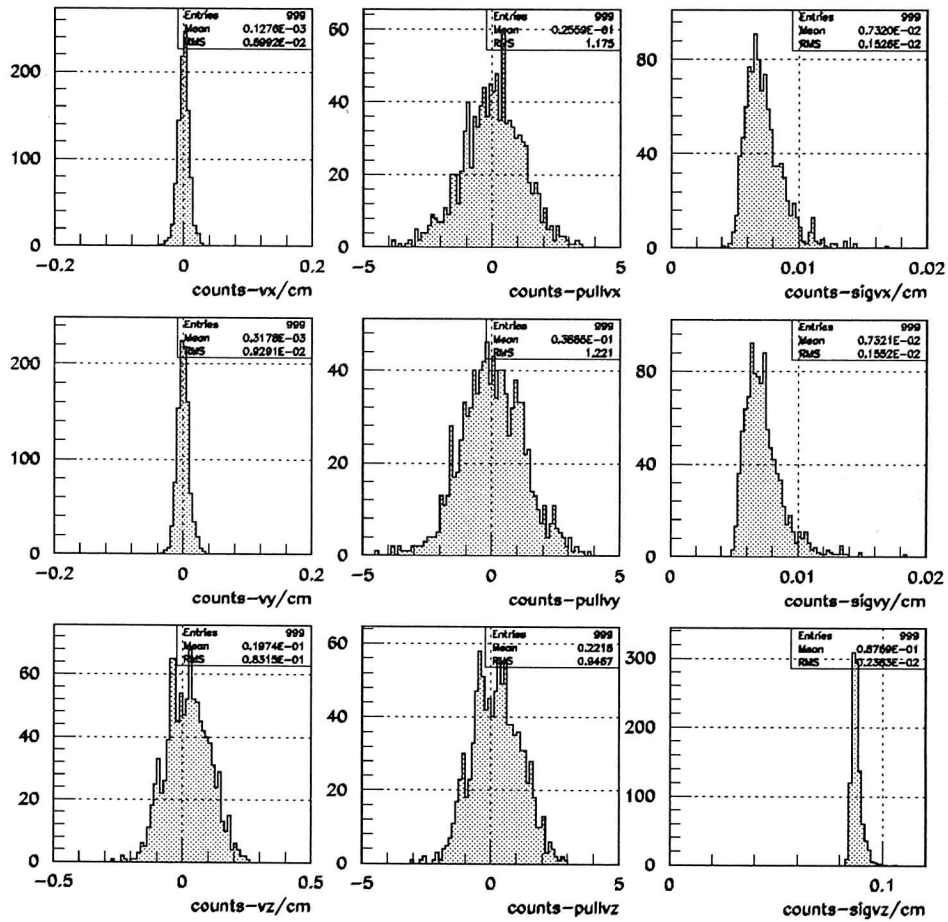


Abbildung 10.4: Die nichtnormierten Pullwerte der rekonstruierten Vertizes sind in der ersten Spalte zu sehen, in der zweiten die normierten. Da sie gaußverteilt sind, ist der Vertexfit zuverlässig. Die letzten Spalte zeigt die  $\chi^2$ -Verteilung der Vertexfehler.

zwischen der nominellen primären Vertexposition und der rekonstruierten sind, dividiert durch den gefitteten Fehler:

$$pull(x) = \frac{x^{true} - x^{fitted}}{\sigma_x}$$

Die Verteilung der nicht normierten Pulls ist in Spalte 1 von Abbildung 10.4 zu sehen und wird hinreichend gut durch eine Gaußverteilung mit Zentrum 0.0 beschrieben. Da der nominelle Vertex stets der Nullpunkt

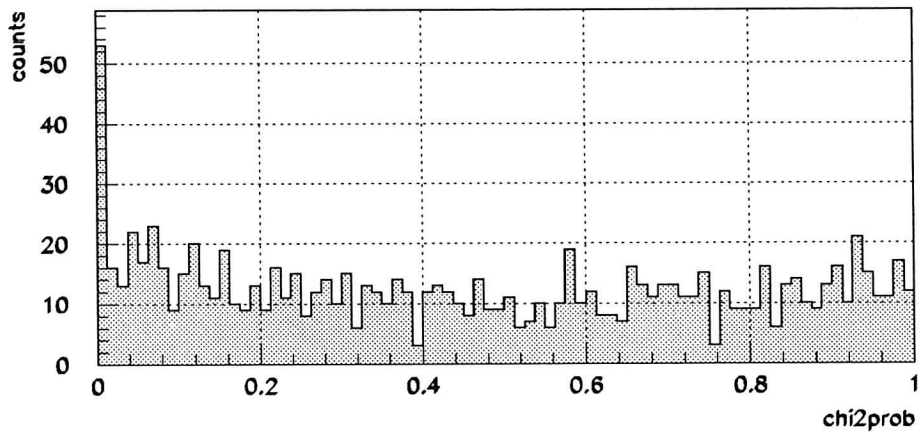
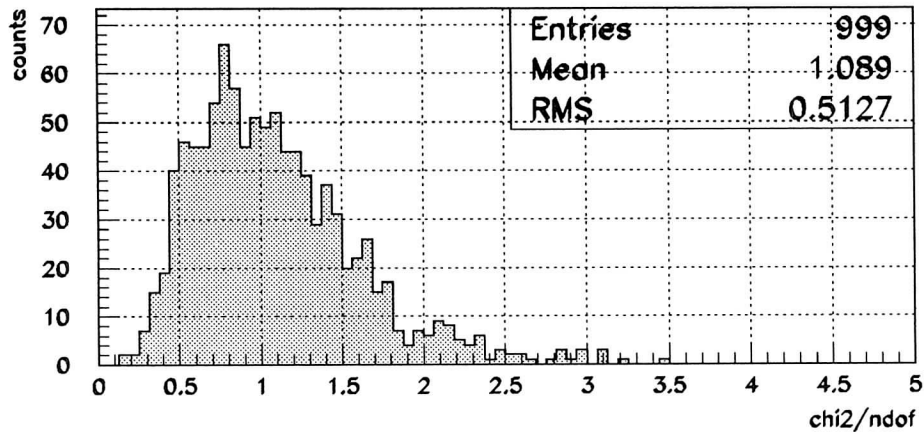


Abbildung 10.5:  $\chi^2$ -Verteilung und -Wahrscheinlichkeit für den Vertex-Fit

war, kann man aus den Verteilungen sehen, daß der primäre Vertex in der  $xy$ -Ebene auf  $\mu\text{m}$  genau rekonstruiert werden konnte. Die Breite liegt in der gleichen Größenordnung. Die  $z$ -Koordinate des primären Vertexes konnte dagegen nur zwei Größenordnungen schlechter bestimmt werden. Die Ursache dafür liegt darin, daß die Fehler in den  $z$ -Koordinaten der Hits eine Größenordnung größer sind.

Abbildung 10.4 zeigt auch die normierten Pulls. Sie sind in der zweiten Spalte zu sehen, und sie lassen sich durch eine Gaußverteilung mit Breite 1.0 und Zentrum 0.0 beschreiben. Die Breite für die normierten Pulls in  $v_x$  und  $v_y$  sind etwas größer als 1, was darauf hindeutet, daß

die vom Vertexfit bestimmten Fehler in diesen beiden Variablen etwas unterschätzt werden, der umgekehrte Fall liegt für  $v_z$  vor. In Spalte 3 von Abbildung 10.4 ist die Verteilung der Vertex-Fehler gezeigt, die eine  $\chi^2$ -Verteilung aufweisen müssen. Dies ist auch der Fall. Die Fehler sind also zuverlässig. Die Fehler für  $v_x$  und  $v_y$  liegen in der Größenordnung von  $10\ \mu\text{m}$ , für  $v_z$  in der Größenordnung von  $1\ \text{mm}$ .

In Abbildung 10.5 sieht man die Verteilung der reduzierten  $\chi^2$ -Werte des Vertex-Fits, die hinreichend gut bei 1.0 ein Maximum hat. Die  $\chi^2$ -Werte zeigen also das geforderte Verhalten, und man kann dem Vertex-Fit vertrauen. Die Verteilung der  $\chi^2$ -Wahrscheinlichkeit in Abbildung 10.5 zeigt einen Peak bei 0.0 aufgrund von Ausreißern im Fit, ist aber sonst hinreichend flach. Es gibt also keinen signifikanten Bias durch die Vertex-Fit-Prozedur.

Eine detailliertere Untersuchung ist hier sicher noch nötig, vorzugsweise mit experimentellen Daten und einer größeren Variation der Topologien.

# Kapitel 11

## Test mit experimentellen Daten

*Diese Typen sind nicht abstrakt, sie sind so real wie int und float.*

– DOUG McILLROY

### 11.1 Die Strahlzeit

Anfang April 1992 fand eine erste Teststrahlzeit für die zentrale Driftkammer statt. Bis zu diesem Zeitpunkt waren drei Sektoren, die Sektoren 11, 12 und 13, mit 56 Drähten ausgerüstet. Das Magnetfeld hatte eine Stärke von 0.5 T. Ein Nickel-Strahl mit 500 MeV/u wurde auf ein Kupfertarget geschossen. Abbildung 11.1 zeigt ein Ereignis aus dieser Strahlzeit.

Diese Strahlzeit bot die erste Gelegenheit, Track-Finding mit der adaptiven Hough-Transformation an experimentellen Daten zu testen.

Es konnten, wie im folgenden zu sehen sein wird, gute Resultate erzielt werden, obwohl das Kammerverhalten und die Strahlqualität weit davon entfernt waren, optimal zu sein. Insbesondere war die grundlegende Annahme bei der adaptiven Hough-Transformation, so wie sie in dieser Arbeit ausgeführt wird, nämlich daß der primäre Vertex sehr präzise bekannt ist, nicht erfüllt, wie an den Resultaten des Fits für den primären Vertex zu sehen ist. Aufgrund der niedrigen Teilchen-Multiplizität in diesem Experiment, konnte dem aber dennoch erfolgreich begegnet werden. Die Trackmultiplizität in diesen drei Sektoren war maximal 10.

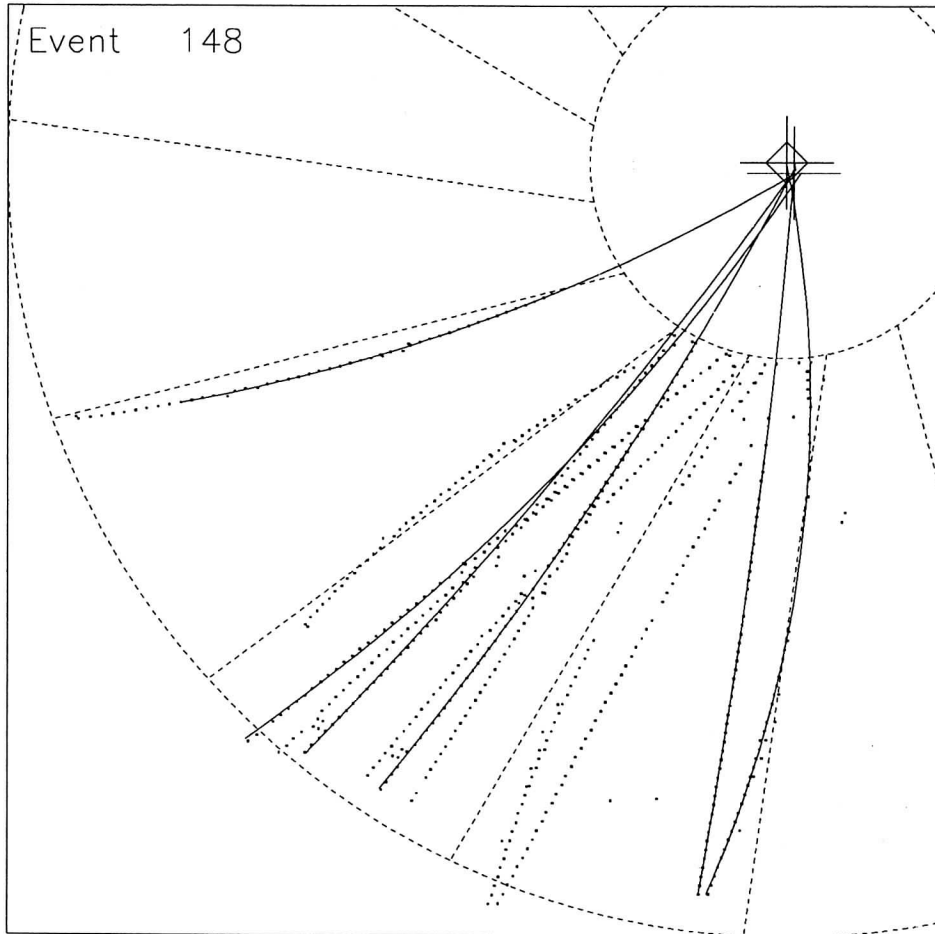


Abbildung 11.1: Ein typisches Ereignis von der Strahlzeit im April 1992, Nickel-Strahl von 500 MeV/u auf ein Kupfertarget. Alle Spuren wurden mit der Hough-Transformation rekonstruiert. Die Abbildung zeigt die gefitteten Spuren. Das Quadrat in der oberen rechten Ecke stellt dimensionsgerecht das Target dar, das Kreuz symmetrisch dazu, den Koordinatenursprung. Ein zusätzliches Kreuz zeigt den rekonstruierten Common Vertex an.

## 11.2 Das Verhalten der Driftkammer

Da die zentrale Driftkammer zum ersten Mal unter Strahlbedingungen betrieben wurde, entsprach ihre Effizienz noch nicht den gewünschten Anforderungen, so daß oft keine vollständigen Spuren gesehen werden

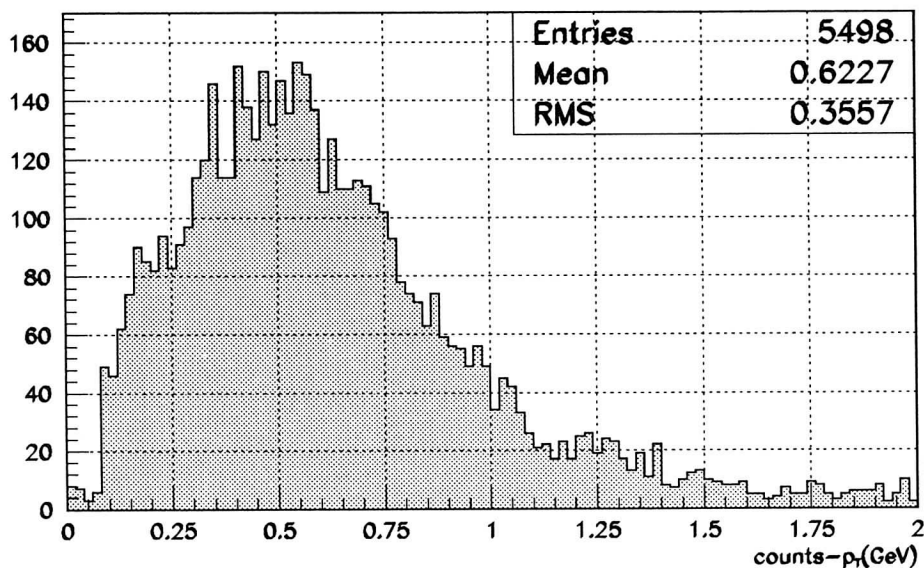


Abbildung 11.2:  $p_T$ -Verteilung der Strahlzeit von Anfang April 1992.

konnten. Die Elektronik war noch nicht hinreichend gut gegen äußere Störimpulse abgeschirmt, so daß teilweise die gesamte Ausleseelektronik zum Schwingen kam.

Nachschwingungen der Driftkammerpulse verursachten zudem Untergrund, der mit den echten Teilchenspuren korreliert war; aufgrunddessen waren sie zum Teil von einer oder mehreren *Geisterspuren* begleitet, die aber ohne eine genaue Kenntnis des primären Vertex nicht zu diskriminieren sind.

Schließlich, als weitere Schwierigkeit, schwankten die Kalibrationskonstanten Driftgeschwindigkeit und Timeoffset von Ereignis zu Ereignis.

Schwierigkeiten bereitete aber auch das Triggersystem. Getriggert wurde mit Hilfe eines Barrel-Moduls (6 Szintillator- und 6 Čerenkov-Detektoren), das unter dem mittleren der drei Sektoren (Sektor 12) angebracht war. In einigen Ereignissen konnte man Eventmixing feststellen, was zu weiteren Ineffizienzen in der Spurrekonstruktion führte.

Dennoch war die Strahlzeit, wie die folgenden Ergebnisse zeigen, aufschlußreich. Sie hat nicht nur wichtige Ergebnisse über das Kammerverhalten erbracht, sondern auch gezeigt, daß die Trackrekonstruktion mit der adaptiven Hough-Transformation (AHT) auch unter realen Bedingungen ihre Aufgabe erfüllen kann.

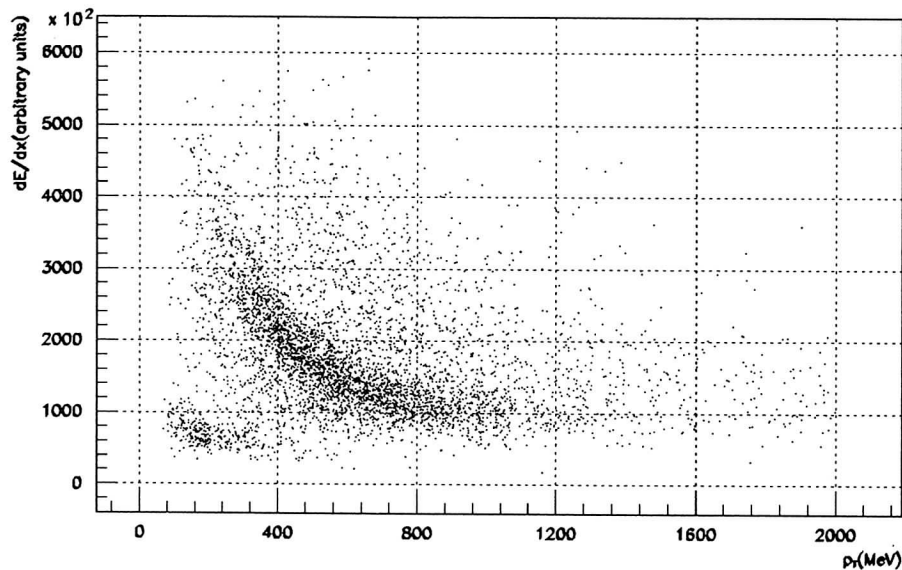


Abbildung 11.3:  $dE/dx$ -Plot aus der Strahlzeit von Anfang April 1992: Da die  $z$ -Koordinate unbrauchbar war, konnte nur eine  $p_T$ -Bestimmung durchgeführt werden. Trotz der nur vorläufigen Kalibrierung von Driftgeschwindigkeit, Timeoffsets und Lorentzwinkel, insbesondere aber der Eichung der Ladungen, können Pionen (links unten) und Protonen (darüber) deutlich aufgelöst werden. Selbst Deuteronen (weiter rechts oben) sind noch sichtbar,  $\alpha$ -Teilchen, die noch weiter rechts oben liegen müßten, konnten aber offenbar nicht registriert werden.

### 11.3 Kalibrierung der Driftkammer

Erschwerend für die Trackrekonstruktion mit der AHT kam noch hinzu, daß die Kalibrierung der zentralen Driftkammer erst am Anfang stand:

Driftgeschwindigkeit, globale Timeoffsets und Lorentzwinkel wurden *optisch kalibriert*. Die individuellen Timeoffsets der Signaldrähte wurden nicht berücksichtigt. Eine Pulsereichung für die Ladung wurde auch erst in den Wochen nach der Strahlzeit durchgeführt.

Mit Hilfe eines Eventdisplays für die  $xy$ -Ebene wurden die Kalibrationskonstanten so lange interaktiv verändert, bis sich ein konsistentes Bild ergab.

Werden falsche Kalibrationskonstanten verwendet, kann man das folgendermaßen am Eventdisplay erkennen: Teilchenspuren, die von einem Sektor in den benachbarten laufen, weisen einen Versatz an der Sektor-

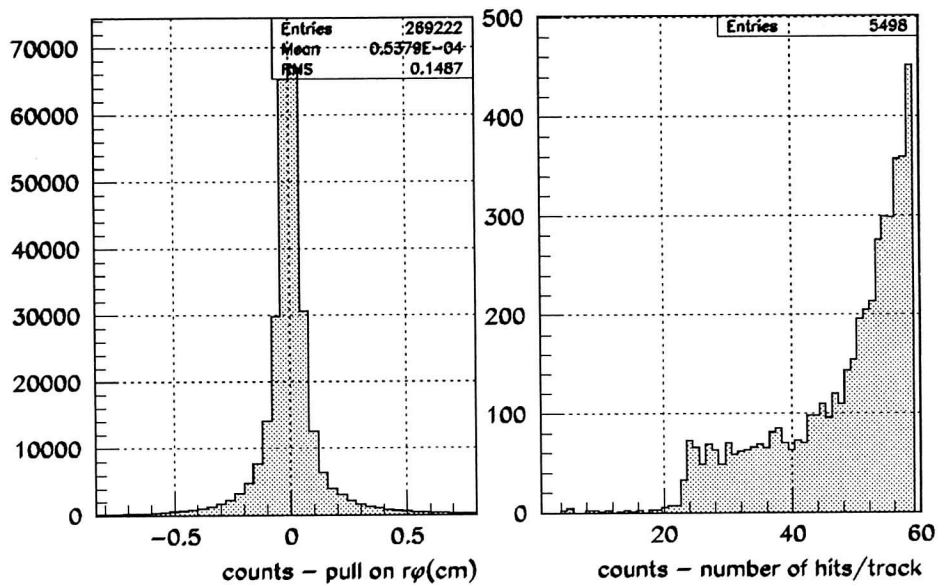


Abbildung 11.4: Die Pulls geben Aufschluß über die Ortsauflösung der zentralen Driftkammer. Die Verteilung ist in guter Näherung normal mit Zentrum 0. Die Ortsauflösung war weit davon entfernt optimal zu sein, etwa  $1500 \mu\text{m}$ . Rechts: Histogramm der Anzahl der Hits pro Track. Der Peak bei 60 zeigt, daß mit der HT die Tracks weitgehend vollständig rekonstruiert werden im Gegensatz zu lokalen Methoden, die dazu neigen, Tracks zu zerstückeln.

grenze auf. Ebenso gibt es einen Versatz für Spuren, die eine Auslesedrahtebene innerhalb eines Sektors kreuzen. Mit einem falschen Lorentzwinkel sieht es so aus, als ob die wahren Spuren nicht aus der Targetregion kommen, sie sind von den Spiegelspuren weniger deutlich zu unterscheiden.

Für die Kalibration der Driftgeschwindigkeit benötigt man Teilchenspuren, die von einem Sektor in einen anderen laufen. Die beiden Stücke der Spur treffen an der Sektorgrenze für die richtige Driftgeschwindigkeit aufeinander.

Um die globalen Timeoffsets zu kalibrieren werden Teilchenspuren verwendet, die eine Auslesedrahtebene in einem Sektor schneiden. Mit dem richtigen globalen Timeoffset stoßen die beiden Teilstücke der Spur an der Auslesedrahtebene zusammen.

Weniger sensitiv ist das Bild auf den Lorentzwinkel. Bei der Einstellung des Lorentzwinkels hilft, daß die wahren Teilchenspuren aus dem Targetbereich kommen. Aufgrund der geneigten Sektoren kommt

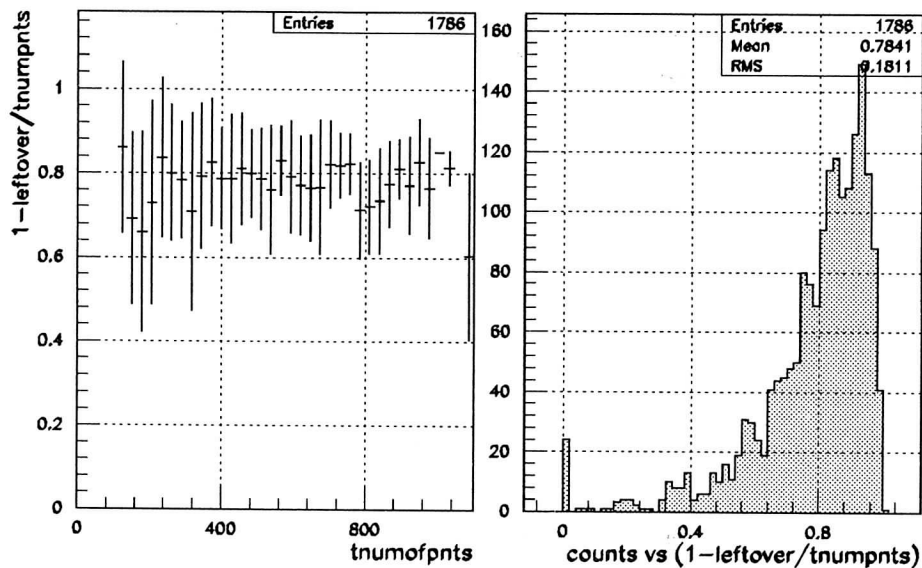


Abbildung 11.5: Point-Removal-Effizienz im Experiment:  $1 - (\text{Anzahl der left overs})/(\text{Totale Anzahl der Punkte})$  pro Ereignis aufgetragen gegen die totale Anzahl der Punkte pro Ereignis. Die Effizienz liegt hier auf dem ersten Blick bei 80%. Berücksichtigt man den Untergrund von etwa 10% und die Tatsache, daß nicht ereignisweise kalibriert wurde, die Kalibrationskonstanten (Driftgeschwindigkeit, Lorentzwinkel, Timeoffsets) aber von Ereignis zu Ereignis unter Umständen stark schwankten, kann man von einer Effizienz ausgehen, die deutlich über 90% liegt

die Spiegelspur nicht aus dieser Region. Dies ist ein weiterer Grund, das Target möglichst klein zu halten und den primären Vertex gut zu kennen.

## 11.4 Ergebnisse der Strahlzeit

Für die Analyse der so gewonnenen Ereignisse konnte allerdings nur die zweidimensionale Version des Trackingprogramms verwendet (Rekonstruktion der Kreisspuren in der  $xy$ -Ebene) und daher nur eine  $p_T$ -Bestimmung (Abbildung 11.2) erzielt werden.

Der Ursache dafür war der hohe Untergrund und die fehlende Ladungseichung. Dies führte zu einer unbrauchbaren  $z$ -Koordinate, die über Ladungsteilung bestimmt wird. Im Nachhinein stellte sich heraus, daß externe Signale in die Driftkammerelektronik eingekoppelt hatten.

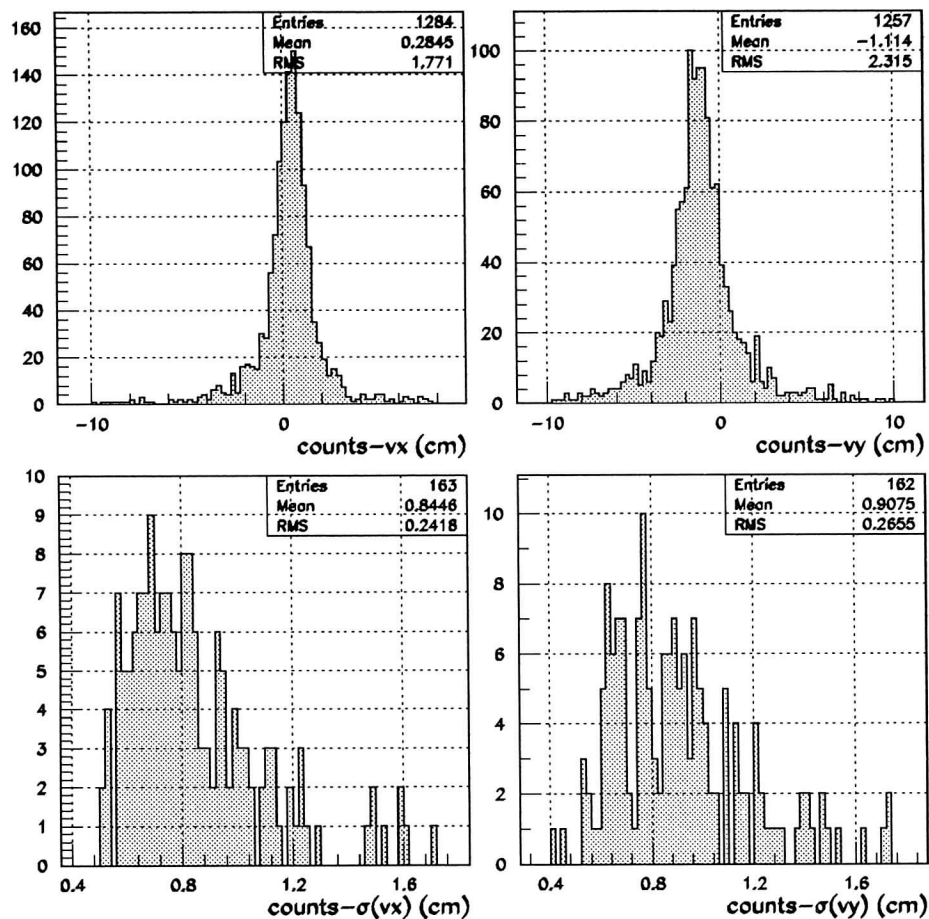


Abbildung 11.6: Ergebnis des Vertex-Fits für 3000 Ereignisse aus der Strahlzeit von April 1992. Der nominelle Vertex liegt bei (0,0). Da das Testexperiment nur parasitär war, war nur schlecht auf das Target fokussiert. Das Resultat des Vertex-Fits zeigt, daß die x-Koordinate des Primary Vertex um 2.8 mm vom nominellen Vertex abwich und die y-Koordinate sogar um mehr als 1cm unter dem nominellen Wert lag.

Die Quellen dafür sind aber inzwischen erkannt, die Beseitigung dieser Schwierigkeiten ist zum Teil schon gelungen, zum Teil noch in der Bearbeitung. Eine kleine Abwandlung der verwendeten Vorverstärker, ein völlig neues Design der Motherboards und eine wesentlich bessere Abschirmung der Netzgeräte führten zu erheblichen Verbesserungen.

Aus diesem Grunde kann der in Abbildung 11.3 gezeigte dE/dx-Plot

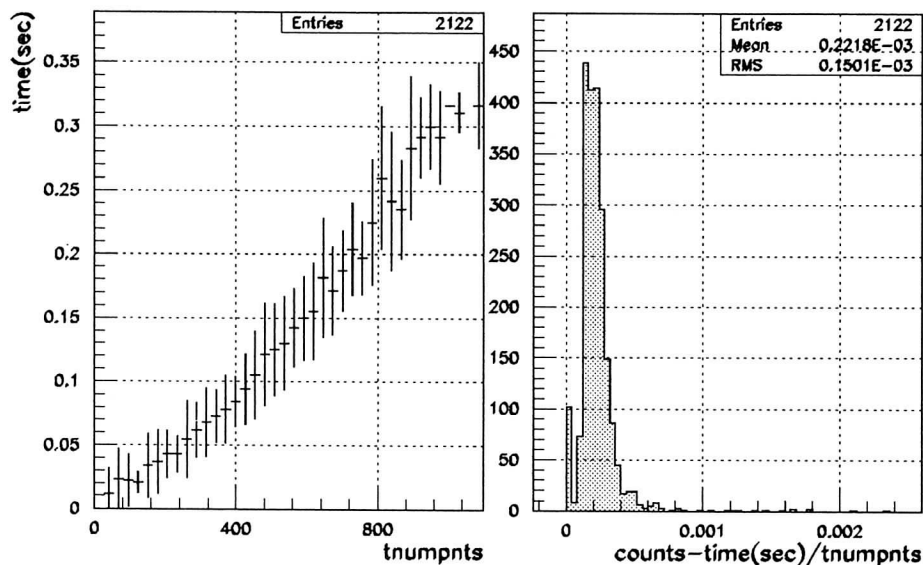


Abbildung 11.7: CPU-time in Sekunden auf einer DEC 5000/125 gegen die totale Anzahl von Punkten pro Ereignis aufgetragen für 3000 Ereignisse aus der Strahlzeit von April 1992.

auch nur einen ersten Eindruck von den Möglichkeiten der CDC geben. Eine automatische *Particle Identification* existierte bis zu diesem Zeitpunkt noch nicht. In der Abbildung ist  $dE/dx$  gegen den transversalen Impuls  $p_T$  aufgetragen, Korrekturen aufgrund des polaren Winkel der Spur in der Driftkammer sind also nicht enthalten.

Trotzdem kann man Pionen ( $\pi^\pm$ , die Anhäufung links unten) und Protonen (die Anhäufung über den Pionen) und sogar Deuteronen noch weiter darüber erkennen. Dabei wurden nur Tracks mit mindestens 30 rekonstruierten Punkten berücksichtigt. Für  $dE/dx$  wurde ein sogenanntes *truncated mean* bestimmt. Aufgrund der Landau-Fluktuationen gibt es mit kleiner Wahrscheinlichkeit einen großen Energieverlust. Dieser verfälscht das  $dE/dx$  und verursacht für die Teilchenidentifizierung Probleme. Daher sortiert man die auf einer Spur gemessenen Ladungen nach ihrer Größe und benutzt zur Bestimmung des Energieverlustes nicht alle Punkte, sondern nur 60% beispielsweise. Die Qualität des  $dE/dx$ -Plots verbessert sich auch tatsächlich, wenn statt 100% nur die 60% kleinsten Energieverluste auf einer Spur verwendet werden. Ein truncated mean von 60% wurde für den  $dE/dx$ -Plot in Abbildung 11.3 verwendet.

Trägt man die Pulls der gefitteten Tracks auf, so erhält man einen

Eindruck von der Ortsauflösung des Detektors. Aufgrund des starken Untergrundes ist diese noch weit vom angestrebten Wert entfernt. In dieser Strahlzeit wurde etwa  $1500 \mu\text{m}$  erreicht, das siebenfache des gewünschten Wertes von  $200\text{-}300 \mu\text{m}$ . Ein weiterer Grund dafür ist aber sicherlich auch die nur vorläufige Kalibrierung gewesen. Eine verbesserte Kalibrierung mit Berücksichtigung der individuellen Timeoffsets der Auslesedrähte und unter Zuhilfenahme einer wesentlich genaueren Position des primären Vertexes eröffnet noch einige Perspektiven. Mit der oben genannten Verbesserung der Elektronik wurde bereits eine erhebliche Reduktion des Untergrundes um weit mehr als 50% und damit eine Erhöhung der Ortsauflösung der CDC erreicht. Ein Laserkalibrationssystem für die zentrale Driftkammer befindet sich im Aufbau. Damit konnten bereits zuverlässige  $z$ -Koordinaten bestimmt werden. Angesichts dieser Fortschritte kann man davon ausgehen, daß die typischen Werte für die Ortsauflösung einer Driftkammer dieses Typs von  $300 \mu\text{m}$  in der  $xy$ -Ebene und von  $2 \text{ cm}$  in  $z$  (1% der Drahtlänge) erreicht werden.

Die beeinträchtigte Ortsauflösung der zentralen Driftkammer in dieser ersten Strahlzeit übt auch einen Einfluß auf das Track-Finding aus. In diesem Licht muß eine Point-Removal-Effizienz von 80% als Erfolg gewertet werden (Abbildung 11.5). Geht man von dem Erfahrungswert dieser Strahlzeit, daß 10% der Hits Untergrund sind, aus, so liegt die Effizienz der adaptiven Hough-Transformation deutlich über 90%.

Dies ist um so erstaunlicher, als die vorausgesetzten Verhältnisse für die Position des primären Vertexes nicht erfüllt waren. Der primäre Vertex lag etwa  $1 \text{ cm}$  unter dem Ursprung des Koordinatensystems. Außerdem war die Streuung beträchtlich. Das ist in Abbildung 11.6 zu sehen.

Besonders an diesem Punkt ergeben sich neben der Steigerung der Kammerperformance noch weitere Möglichkeiten, die Tracking-Performance im Experiment zu steigern. Die Ionenoptik ist eine sehr präzise Technik, außerdem kann man das Target von den zuletzt benutzten Dimensionen ( $3 \text{ cm} \times 3 \text{ cm}$ ) drastisch verkleinern und versuchen den Strahlfleck wesentlich zu verbessern. Auf diese Weise sollte es ohne größeren Aufwand möglich sein, den primären Vertex auf  $1 \text{ mm}$  genau zu kennen.

Betrachtet man dazu noch den Zeitaufwand für AHT, Kreisfit und Vertexfit in der  $xy$ -Ebene (Abbildung 11.7), kommt man zu dem Schluß, daß man schon zu diesem Zeitpunkt in der Lage ist, die riesige Menge an Driftkammerdaten, die man in Zukunft zu erwarten hat, zu analysieren. Der durchschnittliche Zeitaufwand für eine Spur beträgt  $20 \text{ ms}$ .

# Kapitel 12

## Zusammenfassung

*Perfektion wird erst am Punkt des Zusammenbruchs erreicht.*

– C. N. PARKINSON

Nach Jahren der Entwicklung hat sich das Feld der Spurerkennung konsolidiert. Detektoren, Algorithmen und Computer haben ihre Performance in einer Art und Weise verbessert, daß die Spurerkennung für geplante Experimente keine problematische Rolle mehr spielt.

Konsolidierung bedeutet jedoch nicht Stagnation. Vektorprozessoren [HAU87, GEO86, GEO87] und möglicherweise Parallelprozessoren versprechen eine erhebliche Beschleunigung der Spurerkennung. Dies zieht unter Umständen einen radikalen Wechsel der verwendeten Methoden und Algorithmen nach sich.

Dies deutet sich jetzt schon an mit einer wachsenden Anzahl von Vorschlägen, neuronale Netzwerke [DEN88] auf Pattern Recognition Probleme in der Hochenergiephysik anzuwenden.

In diese Entwicklung ordnet sich die in dieser Arbeit verwendete adaptive Hough-Transformation (AHT) ein. Sie ist intrinsisch parallel, und das Hough-Feld kann als Energiefläche aufgefaßt werden, in dem ein neuronales Netzwerk globale Minima, die den Teilchenspuren entsprechen, bestimmen kann.

Die Hough-Transformation (HT) gehört zu den modernsten Verfahren auf dem Gebiet der Spurerkennung. Sie kann prinzipiell auf jede parametrisierbare Kurve angewendet werden. Die meisten Aktivitäten gab es in den letzten Jahren im Zusammenhang mit der Hough-Transformation und neuronalen Netzwerken, mit denen die HT in engem Zusammenhang steht.

Zukünftige Schwerionenexperimente an der GSI stehen vor dem Problem, Driftkammerereignisse hoher Teilchenmultiplizität zu analysieren.

Das bekanntermaßen chronische Versagen lokaler Methoden bei hohen Multiplizitäten aufgrund entweder mangelnder Doppelspurauflösung oder überproportionalem Anstieg der Rechenzeit mit der Anzahl der Punkte pro Ereignis, und die große Menge zu erwartender Driftkammerdaten, haben zu dem Entschluß geführt, sich den neuesten Entwicklungen auf dem Gebiet der Spurerkennung anzuschließen und auf diesem Feld aktiv zu werden.

Die in dieser Arbeit behandelte AHT ist eine Methode, die Peaks im Hough-Feld zu lokalisieren, ohne den Untergrund im Detail auszurechnen. Sie wendet eine intelligente Histogramm- und Suchstrategie an. Sie benutzt einen kleinen Akkumulator und zerlegt den Parameterbereich, der momentan von Interesse ist, in wenige Intervalle, in welche die HT die Hits abbildet. Die Information im Akkumulator wird dann benutzt, um den Parameterbereich umzudefinieren, so daß interessante Bereiche mit größerer Auflösung untersucht werden können. Die Iteration dauert so lange, bis die Parameter mit der gewünschten Auflösung bestimmt sind. Im Mittel werden 4-7 Iterationen benötigt, um die Mittelpunktskoordinaten einer Kreisspur bis auf 1 mm genau zu erhalten.

Es wurde gezeigt, daß die AHT die Tracking-Möglichkeiten zu sehr hohen Track-Dichten erweitert. Der Zeitaufwand liegt für 100 Spuren mit Track- und Vertex-Fitting im Bereich von 4-5 Sekunden. Damit zeigt sich diese Methode im Vergleich zu lokalen Verfahren in diesem Bereich der Trackmultiplizitäten als überlegen, da sie nicht mit kombinatorischen Schwierigkeiten konfrontiert wird.

Dabei bleibt die Track- und Point-Removal-Effizienz bei über 95%, und die Doppelspurauflösung bei 1%. Die überwiegende Mehrzahl der Teilchenspuren wird außerdem nahezu vollständig rekonstruiert, während lokale Methoden dazu neigen, Spuren nur stückweise zu erkennen, was zum Versagen dieser Ansätze bei hohen Teilchenmultiplizitäten führt. Die Ursachen für die noch vorhandenen Ineffizienzen sind weiter Gegenstand der Untersuchung.

Die in dieser Arbeit vorgestellten Implementierung setzt die genaue Kenntnis des Reaktionspunktes voraus. Die Anzahl der Parameter für die Hough-Transformation konnte auf diese Weise von drei auf zwei reduziert werden. Dies hatte eine enorme Steigerung der Tracking-Performance zur Folge.

Das anfänglich nur zweidimensionale Tracking-Programm wurde durch Einbeziehung des Binnings im Dipwinkel  $\theta$  und der Hinzunahme der z-Koordinate nach dem Pattern Recognition in der  $r\phi$ -Ebene zu einem voll dreidimensionalen Rekonstruktionsprogramm erweitert.

Die adaptive Hough-Transformation liefert nicht nur einen schnellen Eindruck von der Trackmultiplizität eines Ereignisses, sondern liefert auch bereits approximative Trackparameter. Diese müssen noch durch ein geeignetes Track- und Vertex-Fit-Programm verbessert werden. Daher wurden umfangreiche Hilfsprogramme nicht nur für einen Kreisfit sondern auch für einen vollen Helix-Fit entwickelt. Dessen statistische Eigenschaften wurden im Rahmen dieser Arbeit ebenfalls eingehend untersucht. Die rekonstruierten Parameter erscheinen als zuverlässig. Der Einfluß des Teilchenimpulses auf die Qualität des Trackfits erscheint im Vergleich zur Ortsauflösung des Detektors als akzeptabel. Eine analoge statistische Absicherung wurde für den Vertexfit gezeigt. Da die Fitprobleme nichtlinear sind, wurde die Newtonmethode zur Parameterbestimmung verwendet. In der Regel wird das Abbruchkriterium nach 3-4 Iteration erreicht.

Die Fitprogramme existieren als zwei- und dreidimensionale Version (ohne und mit Einbeziehung der z-Koordinate). In der Analyse der experimentellen Daten kamen nur die zweidimensionalen Versionen zum Einsatz. Dies ist jedoch nur als Anfangsstadium zu betrachten, weshalb in dieser Arbeit der Schwerpunkt auf der vollen dreidimensionalen Rekonstruktion lag.

Die Anwendung der adaptiven Hough-Transformation auf die ersten experimentellen Daten hat nicht nur Aufschluß über die Auflösung des Detektors geliefert, sondern auch eindrucksvoll die Leistungsfähigkeit der adaptiven Hough-Transformation trotz der nur vorläufig kalibrierten Daten, der großen Abweichung des realen vom nominalen primären Vertex, der Ineffizienzen der Driftkammer, der Probleme mit dem Timing und des hohen Untergrundes gezeigt. Die höchste erreichte Trackmultiplizität in der drei Sektoren des ersten Experiments war 10. Dabei blieb der Zeitaufwand der Spurrekonstruktion stets unter einer halben Sekunde. Die Point-Removal-Effizienz betrug 80%. Unter Berücksichtigung des Untergrunds kann man daraus auf eine Effizienz für die Rekonstruktion der Tracks von weit über 90% schließen. Dies bestätigt die generelle Funktionstüchtigkeit der in dieser Arbeit entwickelten Methode, der adaptiven Hough-Transformation.

*I thought that my voyage had come to its end at the last limit of my power – that the path before me was closed, that provisions were exhausted and the time come to take shelter in a silent obscurity.*

*But I find that thy will knows no end in me. And when old words die out on the tongue, new melodies break forth from the heart; and where the old tracks are lost, new country is revealed with its wonder.*

– RABINDRA NATH TAGORE, GITANJALI

# Anhang A

## Formeln für den Track-Fit

Die Fehlermatrix  $\mathbf{E}$  ist folgendermaßen definiert:

$$\begin{aligned}
 E_{\alpha\beta} &= \frac{1}{2} \frac{\partial^2 \chi^2}{\partial h_\alpha \partial h_\beta} & (A.1) \\
 &= \sum_{i=1}^{n_{\text{points}}} \frac{1}{\sigma_{\phi,i}^2} \frac{\partial \phi_i}{\partial h_\alpha} \frac{\partial \phi_i}{\partial h_\beta} + \sum_{i=1}^{n_{\text{points}}} \frac{1}{\sigma_{z,i}^2} \frac{\partial z_i}{\partial h_\alpha} \frac{\partial z_i}{\partial h_\beta} \\
 &\quad + \sum_{i=1}^{n_{\text{points}}} \frac{(\phi_i^{\text{fit}} - \phi_i^{\text{measured}})}{\sigma_{\phi,i}^2} \frac{\partial^2 \phi_i}{\partial h_\alpha \partial h_\beta} \\
 &\quad + \sum_{i=1}^{n_{\text{points}}} \frac{(z_i^{\text{fit}} - z_i^{\text{measured}})}{\sigma_{z,i}^2} \frac{\partial^2 z_i}{\partial h_\alpha \partial h_\beta}
 \end{aligned}$$

Vernachlässigt man Terme, die Ableitungen zweiter Ordnung enthalten, so erhält man:

$$E_{\alpha\beta} \simeq \sum_{i=1}^{n_{\text{points}}} \frac{1}{\sigma_{\phi,i}^2} \frac{\partial \phi_i}{\partial h_\alpha} \frac{\partial \phi_i}{\partial h_\beta} + \sum_{i=1}^{n_{\text{points}}} \frac{1}{\sigma_{z,i}^2} \frac{\partial z_i}{\partial h_\alpha} \frac{\partial z_i}{\partial h_\beta} \quad (A.2)$$

Der Gradientenvektor  $\mathbf{G}$  ist definiert durch:

$$\begin{aligned}
 G_\beta &= -\frac{1}{2} \frac{\partial \chi^2}{\partial h_\beta} & (A.3) \\
 &= -\sum_{i=1}^{N_{\text{points}}} \frac{\phi_i^{\text{fit}} - \phi_i^{\text{measured}}}{\sigma_{\phi,i}^2} \frac{\partial \phi_i}{\partial h_\beta} - \sum_{i=1}^{N_{\text{points}}} \frac{z_i^{\text{fit}} - z_i^{\text{measured}}}{\sigma_{z,i}^2} \frac{\partial z_i}{\partial h_\beta}
 \end{aligned}$$

Die Ableitungen von  $z$  sind:

- $\frac{\partial z}{\partial \kappa} = \frac{z_0 - z}{\kappa} + \frac{\tan(\lambda)}{\kappa^2} \frac{X}{\sqrt{1 - X^2}} \left( 2 + \frac{sst \, d_0 \, \kappa}{1 - sst \, d_0 \, \kappa} \right)$

- $\frac{\partial z}{\partial \tan(\lambda)} = \frac{z_0 - z}{\tan(\lambda)}$
- $\frac{\partial z}{\partial \phi_0} = 0$
- $\frac{\partial z}{\partial d_0} = 2 \frac{\tan(\lambda)}{\kappa} \frac{X}{\sqrt{1-X^2}} \left( \frac{sst \kappa}{2(1-sst d_0 \kappa)} - \frac{d_0}{r^2 - d_0^2} \right)$
- $\frac{\partial z}{\partial z_0} = 1$

Dabei ist  $X = \frac{\kappa}{2} \sqrt{\frac{r^2 - d_0^2}{1 - sst d_0 \kappa}}$ .

Die Ableitungen von  $\phi$  sind:

- $\frac{\partial \phi}{\partial \kappa} = \frac{sst \, sst \, (d_0^2 - r^2)}{2r(d_0 \kappa - sst)^2 \sqrt{1-Y^2}}$
- $\frac{\partial \phi}{\partial \tan(\lambda)} = 0$
- $\frac{\partial \phi}{\partial \phi_0} = 1$
- $\frac{\partial \phi}{\partial d_0} = \frac{sst}{2r\sqrt{1-Y^2}} \left( 1 + \frac{1 - \kappa^2 r^2}{(d_0 \kappa - sst)^2} \right)$
- $\frac{\partial \phi}{\partial z_0} = 0$

Dabei ist  $Y = \frac{\kappa(r^2 - d_0^2)}{2r(d_0 \kappa - sst)} - \frac{d_0}{r}$ .

# Anhang B

## Formeln für den Vertex-Fit

Der Ansatz für das  $\chi^2$  beim Vertex-Fit lautet:

$$\chi^2 = \sum_{i=1}^N \{w_i^2 T^2 + w_i^2 L^2\} \quad (\text{B.1})$$

Dabei ist :

$$\begin{aligned} T &= s - \rho \\ L &= C - z_0 - \lambda \rho \psi \\ s &= \sqrt{(A - x_c)^2 + (B - y_c)^2} \\ \psi &= 2 \arcsin\left(\frac{Q}{2\rho}\right) \\ Q &= \sqrt{(x - x_0)^2 + (y - y_0)^2} \end{aligned}$$

Wobei:

$$\begin{aligned} x &= x_c + \rho \frac{A - x_c}{s} \\ y &= y_c + \rho \frac{B - y_c}{s} \end{aligned}$$

Für die Berechnung der Ableitungen wurde die Approximation

$$L \approx C - z_0 - \lambda Q$$

verwendet.

Die Ableitungen von  $\chi^2$  sind:

Einmal die ersten Ableitungen:

- $\chi_A^2 = 2 \sum \{w_i^2 TT_A + w_i^2 LL_A\}$
- $\chi_B^2 = 2 \sum \{w_i^2 TT_B + w_i^2 LL_B\}$
- $\chi_C^2 = 2 \sum \{w_i^2 TT_C + w_i^2 LL_C\}$

Dann die Ableitungen zweiter Ordnung:

$$\begin{aligned} \chi_{AA}^2 &= 2 \sum \{w_i^2 (TT_{AA} + T_A^2) + w_i^2 (LL_{AA} + L_A^2)\} \\ \chi_{AB}^2 &= 2 \sum \{w_i^2 (TT_{AB} + T_A T_B) + w_i^2 (LL_{AB} + L_A L_B)\} \\ \chi_{AC}^2 &= 2 \sum \{w_i^2 (TT_{AC} + T_A T_C) + w_i^2 (LL_{AC} + L_A L_C)\} \\ \chi_{BA}^2 &= 2 \sum \{w_i^2 (TT_{BA} + T_A T_B) + w_i^2 (LL_{BA} + L_A L_B)\} \\ \chi_{BB}^2 &= 2 \sum \{w_i^2 (TT_{BB} + T_B^2) + w_i^2 (LL_{BB} + L_B^2)\} \\ \chi_{BC}^2 &= 2 \sum \{w_i^2 (TT_{BC} + T_B T_C) + w_i^2 (LL_{BC} + L_B L_C)\} \\ \chi_{CA}^2 &= 2 \sum \{w_i^2 (TT_{CA} + T_A T_C) + w_i^2 (LL_{CA} + L_A L_C)\} \\ \chi_{CB}^2 &= 2 \sum \{w_i^2 (TT_{CB} + T_B T_C) + w_i^2 (LL_{CB} + L_B L_C)\} \\ \chi_{CC}^2 &= 2 \sum \{w_i^2 (TT_{CC} + T_C^2) + w_i^2 (LL_{CC} + L_C^2)\} \end{aligned}$$

Die einzelnen Ableitungen berechnen sich dabei folgendermaßen:

$$T = \dots \quad \dots = \dots$$

- [DUD72] R. O. Duda, P. E. Hart: Use of the Hough Transformation to Detect Lines and Curves in Pictures, Comm. ACM, Vol. 15, No. 1, Jan. 1972, p. 11
- [DRU80] U. Drumm et al.: Nucl. Instr. and Meth. 178(1980)333
- [FAI86] G. Fai and J. Randrup: Comp. Phys. Comm. 42(1986)385-397
- [FRO88] K. Froherz: Diplomarbeit, FH Darmstadt, 1988
- [FRU90] R. Frühwirth: Track and Vertex Fitting, 1989 Cern School of Computing, CERN 90-06, p. 217
- [GEO86] C. H. Georgiopoulos et al.: A Non-Numerical Method for Track Finding in Experimental High Energy Physics Using Vector Computers, NIM A249 (1986) 451-454.
- [GEO87] C. H. Georgiopoulos et al.: A Vectorized Track Finding and Fitting Algorithm in Experimental High Energy Physics Using a Cyber-205, FSU-SCRI-87-08.

- [HAU87] J. Hauser: Vectorization of Track Finding and Fitting Algorithms for HEP, *Comp. Phys. Comm.* 45 (1987) 121–125.
- [HOU62] P. V. Hough: Method and means for recognizing complex patterns. US Patent 3069654 (1962)
- [ILL87] J. Illingworth, J. Kittler: The Adaptive Hough Transform, *IEEE Trans. Pattern Anal. Machine Int.*, Vol. PAMI-9, No. 5, Sept. 1987, p.690
- [KIM75] C. Kimme, D. Ballard, J. Sklansky, Finding Circles by an Array of Accumulators, *Comm. ACM*, Vol. 18, No. 2, Feb. 1975, p. 120
- [KLE87] K. Kleinknecht: *Detektoren für Teilchenstrahlung*, Teubner Stuttgart 1987
- [LUM83] R. Lumia: A New Three-Dimensional Connected Components Algorithm, *Comp. Vision, Graphics, Image Processing* 23, 207-217(1983)
- [LSZ83] R. Lumia, L. Shapiro, O. Zuniga: A New Connected Components Algorithm for Virtual Memory Computers, *Comp. Vision, Graphics, Image Processing* 22, 287-300(1983)
- [NAG84] S. Nagamiya: Search for New Forms of Nuclear Matter with High Energy Heavy Ion Beams, Invited Talk presented at the 1984 INS-IPCR International Symposium on Heavy Ion Physics (Heavy Ion '84), Part I, Tokyo, August 24–25, 1984
- [NOG86] Y. Noguchi, A. Ono: Global Pattern Recognition In Layered Track Chambers, *NIM A253* (1986) 27–37.
- [NOR88] J. Norton, P. Knobloch:  $\aleph$  88-46/Note 88-6
- [OHL91] Mattias Ohlson, Carsten Peterson, Alan L. Yuille: Track Finding with Deformable Templates – The Elastic Arms Approach, *LU TP* 91–27
- [OLS80] Olsson J., Steffen P., Goddard M. C., Pearce G. F., Nozaki T.: *NIM* 176(1980)403-407
- [PAR90] Particle Data Group: *Particle Properties Data Booklet*, April 1990
- [PER86] J. Perl, A. Schwarz, A. Seidel, A. J. Weinstein: Track Finding With The Mark II/SLC Drift Chambers, *NIM A252* (1986) 616–620.
- [PIM85] G. N. Pimiä: University of Helsinki preprint HU–P–D45

- [POI83] J. Poitou: Pattern Recognition Program For The Pictorial Drift Chamber Diogene, NIM 217 (1983) 373
- [PUE88] F. Pühlhofer et al.: Track Recognition in Digitized Streamer Chamber Pictures, NIM A263 (1988) 360–367.
- [RIT85] H. G. Ritter et al.: Nucl. Phys. A447(1985)3c
- [SAX85] D. H. Saxon: Three-Dimensional Track and Vertex Fitting in Chambers with Stereo Wires, NIM A234(1985)258
- [SCH89] D. Schall: The 'Hough Transform' Method for Pattern Recognition in the SIS  $4\pi$  Central Drift Chamber, priv. comm.
- [SCH88] D. Schall, K. Winkelmann, K. Köhler: A Global Method for Pattern Recognition in the in the SIS  $4\pi$  Central Drift Chamber, GSI Jahresbericht 1988.
- [STM91] Georg Stimpfl-Abele: Recognition of decays of charged tracks with neural network techniques, Computer Phys. Comm. 67(1991)183
- [STO85] R. Stock: The Cern Heavy Ion Program, Nucl. Phys. A447 (1985) 371c–386c
- [STO86] R. Stock: Particle Production in High Energy Nucleus Nucleus Collisions, Phys. Rep. 135(1986)259
- [TRZ88] Martin Trzaska: Versuch eines Pattern-Recognition-Programms für die zentrale Driftkammer des  $4\pi$ -Detektor am SIS, priv. comm.

# Danksagung

Am Schluß dieser Arbeit möchte ich meinen Eltern danken, die mir mein Studium ermöglicht haben.

Danken möchte ich Herrn Henning, der mir diese Arbeit an der GSI bei Darmstadt ermöglicht hat.

Vielen Dank Herrn D. Schall. Meine Bekanntschaft mit ihm war für mich sehr folgenreich, durch ihn bin ich zum Tracking-Geschäft gekommen und an der GSI hängengeblieben.

Ohne die Initiative von Herrn Gobbi wäre diese Arbeit nie zustande gekommen. Wegen seinem Interesse hatte ich den Entschluß gefaßt, meine Diplomarbeit auf diesem schwierigen Gebiet zu versuchen.

Die Mitglieder der Arbeitsgruppe von Herrn D. Pelte aus Heidelberg, die CDC-Gruppe der  $4\pi$ -Kollaboration, N. Herrmann, M. Trzaska, M. Gnirs, T. Wienold und U. Sodan, G. Goebels und E. Häfele, bildeten mein soziales Umfeld.

Nicht nur ihnen, sondern auch der ganzen Gruppe KP1 an der GSI, sei an dieser Stelle herzlich gedankt.

