Untersuchungen zum Trigger von Charm-Ereignissen mit Hilfe Neuronaler Netze beim H1-Experiment

Diplomarbeit an der Fakultät für Physik
der
Ludwig-Maximilians-Universität München

vorgelegt von
Philip Scheideler

angefertigt am

Max-Planck-Institut für Physik

München
Juli 1998

Erstgutachter: Prof. Dr. Christian Kiesling

Zweitgutachterin: Prof. Dr. Dorothee Schaile

Inhaltsverzeichnis

Kapitel 1

Einleitung

Das Standardmodell der fundamentalen, elektroschwachen und starken Wechselwirkung bildet die Grundlage für unser heutiges Verständnis vom Aufbau der Materie. Es enthält zwölf Fermionen (sechs Quarks und sechs Leptonen) als Grundbausteine, die über den Austausch von Eichbosonen miteinander wechselwirken. Die elektroschwache Theorie beinhaltet drei massive (W^+, W^- und Z^0) und ein masseloses (γ) Eichboson, welche mit den Fermionen wechselwirken können, während die starke Wechselwirkung im Rahmen der Quantenchromodynamik (QCD) durch den Austausch von Gluonen zwischen den Quarks beschrieben wird. Das meiste Wissen der modernen Teilchenphysik stammt dabei aus Streuexperimenten, bei denen subatomare Teilchen mit hohen Energien kollidieren und die Reaktionsprodukte ihrer Wechselwirkung mit speziellen Nachweisgeräten (Detektoren) analysiert werden.
Bei HERA[1] in Hamburg werden Elektronen und Protonen in zwei gegenläufig betriebenen Speicherringen beschleunigt und zur Kollision gebracht. Dabei liegt die erreichbare Schwerpunktsenergie mit 314 GeV um mehr als eine Größenordnung über den bisher in Lepton-Nukleon-Streuexperimenten erreichten Energien. Physikalische Untersuchungen in einem bisher unerschlossenen, kinematischen Bereich werden so möglich, und das Standardmodell kann weiter getestet werden.
Hauptinteressensgebiete bei HERA sind neben der tief-unelastischen Elektron-Proton-Streuung bei großen Impulsüberträgen, die vor allem die Verteilung der Quarks im Proton untersuchen soll, auch die Streuung von fast-reellen Photonen bei sehr kleinen Impulsüberträgen. Die Erzeugung und der Nachweis von schweren Quarks (Charm und Beauty) bietet die Möglichkeit, die Dichteverteilung der Gluonen im Proton zu messen und die Modelle zu Erzeugung und Zerfall von schweren Quarks zu untersuchen.
Als Grundvoraussetzung für diese Untersuchungen müssen bei den HERA-Experimenten große Datenmengen bewältigt werden. Durch Wechselwirkungen des Protonstrahls mit Restgasatomen oder Strahlführungskomponenten ergibt sich eine hohe Rate von Untergrundereignissen $\mathcal{O}(100\ \mathrm{kHz})$, die die Rate der ep-Kollisionen um etwa 5 Größenordnungen übersteigt. Eine Senkung der hohen Untergrundraten, bei gleichzeitiger Herausfilterung der physikalisch interessanten Ereignisse ist daher notwendig.

[1] **H**adron-**E**lektron-**R**ing-**A**nlage

Zuständig für diese Vorauswahl ist ein Triggersystem, welches beim Kollisionsexperiment H1 in vier Stufen (Level) aufgeteilt ist. Für das Jahr 2001 ist bei H1 eine Verbesserung der Ausleseelektronik geplant, die auf der zweiten Triggerstufe Trefferpunkte der Teilchen in den Driftkammern bereitstellen soll. Da eine vollständige Rekonstruktion der Ereignisse aus diesen Informationen auf frühem Triggerniveau aus Zeitgründen nicht möglich ist, wird in der vorliegenden Arbeit untersucht, in wie weit mit der Spurinformation Anwendungen für die auf neuronalen Netzen basierende zweite Triggerstufe realisiert werden können. Neuronale Netze ermöglichen aufgrund ihrer inhärenten Parallelität eine sehr schnelle Entscheidung und werden bei H1 bereits erfolgreich als Trigger eingesetzt, um mit Daten die derzeit nach Level 1 verfügbar sind, z.B. J/ψ und ϕ-Ereignisse, herauszufiltern.

Ziel dieser Arbeit ist es, Ereignisse mit D*-Mesonen zu triggern, wobei die Grundidee darin besteht, aus den Spurpunkten auf invariante Massen zu triggern. Bestimmte Trigger auf der ersten Stufe haben bereits hohe Effizienz für diese Ereignisse, sind jedoch mit einer zu hohen Rate behaftet, d.h. es werden noch zu wenig Untergrundereignisse verworfen. Diese hohe Rate soll mit Hilfe der neuronalen Netze, bei Beibehaltung der Effizienz, gesenkt werden, wobei der Untergrund bis zur nächsten Triggerstufe um ca. 2 weitere Größenordnungen reduziert werden muß.
Es soll jedoch zunächst geprüft werden, ob man mit neuronalen Netzen überhaupt auf invariante Massen triggern kann. Anhand von Monte-Carlo Simulationen wird am Beispiel des D^{*+} bzw. D^0 gezeigt, daß dies prinzipiell möglich ist. Die Güte des Ergebnisses ist dabei jedoch von vielen Faktoren, insbesondere der Datenvorverarbeitung, abhängig. Weiterhin wird gezeigt, daß sich neuronale Netze auch für diese Datenvorverarbeitung einsetzen lassen, um in Zukunft zusammen mit anderen Algorithmen z.B. Probleme wie das Online-Tracking zu lösen.

Die Arbeit beginnt mit einer kurzen Einführung in die Theorie der Charm-Produktion (Kap. 2) und einer Beschreibung des H1-Detektors (Kap. 3). Kapitel 4 gibt eine kurze Einführung in die Theorie der Neuronalen Netze und beschreibt die technische Realisierung des neuronalen Triggers in der zweiten Triggerstufe im H1 Experiment. Die Beschreibung der zum Netzwerktraining generierten Daten erfolgt in Kapitel 5, die Ergebnisse des Trainings werden in Kapitel 6 vorgestellt. Kapitel 7 beschäftigt sich mit einem weiteren neuronalen Algorithmus, der sich durch seine Fähigkeit, ähnliche Daten in Klassen einzuteilen (clustering) zur Datenvorverarbeitung einsetzen läßt.

Kapitel 2

Produktion und Zerfall von Charm-Quarks

2.1 Produktion von Charm-Ereignissen

Im Standardmodell wird die elektroschwache Wechselwirkung von Quarks und Leptonen durch den Austausch von $W^{\pm}$-und Z^0 Bosonen bzw. γ vermittelt. Die Quark-Flavour-Eigenzustände der Quarks mit Ladung $+\frac{1}{3}e$ sind jedoch nicht die Massen-Eigenzustände, sondern Mischungen dieser:

$$\begin{pmatrix} u \\ d \end{pmatrix} \rightarrow \begin{pmatrix} u \\ d' \end{pmatrix}$$

mit dem Eigenzustand $\mathrm{d}' = \mathrm{d} \cdot \cos\theta_\mathrm{c} + \mathrm{s} \cdot \sin\theta_\mathrm{c}$. Der Mischungswinkel θ_c wurde von Cabibbo eingeführt und heißt deshalb Cabibbo-Winkel. Dieser Zustand besitzt keine eindeutige Strangeness, so daß Übergänge mit $|\Delta \mathrm{S}| = 1$ möglich sind. Noch vor der Entdeckung des Charm-Quarks schlugen Glashow, Iliopoulos und Maiani (GIM[1]) die Existenz eines vierten Quarks mit Ladung $+\frac{2}{3}e$ vor, das in einer weiteren Dublett-Struktur (c,s$'$) mit $\mathrm{s}' = -\mathrm{d} \cdot \sin\theta_\mathrm{c} + \mathrm{s} \cdot \cos\theta_\mathrm{c}$ vorkommen sollte. In der Cabibbo-Theorie auftretende Probleme, wie Übergänge mit $|\Delta \mathrm{S}| = 1$ in den neutralen Strömen (FCNC[1]), die möglich waren - jedoch nicht beobachtet wurden - konnten so gelöst werden.

Für 3 Generationen mit 6 Flavours (u, d, s, c, b, t) ergeben sich die Mischzustände über eine Mischungsmatrix, die Cabibbo-Kobayashi-Maskawa-Matrix (CKM-Matrix [2,3]):

$$\begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{ts} \end{pmatrix} \cdot \begin{pmatrix} d \\ s \\ b \end{pmatrix}$$

[1] **F**lavour **C**hanging **N**eutral **C**urrents

Die Elemente V_{ij} sind proportional zur Kopplungskonstante der Quarks i und j an die W-Bosonen. Während sich bei Reaktionen mit neutralem Strom der Quarkflavour nicht ändert, ermöglichen CC-Ereignisse durch ihre flavourändernde Wirkung die Produktion von Charm-Quarks aus Down- oder Strange-Quarks. So können bei HERA, wo Elektronen mit Protonen kollidieren (siehe Kap. 3), Charm-Quarks entstehen.

$$\sigma(e^- p) \rightarrow \nu_e \bar{c} X = |V_{cd}|^2 \cdot \sigma_{\bar{d}} + |V_{cs}|^2 \cdot \sigma_{\bar{s}} \tag{2.1}$$

Die entsprechenden Übergangselemente der CKM-Matrix sind mit Vcq bezeichnet, σ_q steht für die Wirkungsquerschnitte an leichten Quarks q. Aufgrund der hohen Masse des W-Bosons ist der Wirkungsquerschnitt für die CC-Reaktionen bei den bei HERA erreichbaren Schwerpunktsenergien sehr klein.
Es gibt eine Vielzahl von Prozessen, die zur Erzeugung von Charm-Quarks führen. Im Gegensatz zur Produktion leichter Quarks u, d, s kann die Charm-Produktion wegen der hohen Charm-Masse mit perturbativer QCD behandelt werden.

2.1.1 Die Boson-Gluon-Fusion

Bei HERA werden Charm-Quarks hauptsächlich durch den QCD Prozeß der Boson-Gluon-Fusion (BGF) erzeugt. Das vom Elektron abgestrahlte Eichboson fusioniert mit einem Gluon des Protons. Das ausgetauschte Eichboson ist ein γ oder ein Z^0 bei Ereignissen mit neutralem Strom

$$\gamma/Z^0 + \mathrm{g} \rightarrow \mathrm{c} + \bar{c} \qquad \text{(NC-Prozesse)}$$

und ein W^- bei Ereignissen mit geladenem Strom.

$$W^- + \mathrm{g} \rightarrow \mathrm{c} + \bar{Q} \qquad \text{(CC-Prozesse)}$$

Die ep-Streuung bei HERA wird durch den Austausch von Photonen bei niedrigem Q^2 dominiert. Der Beitrag der Eichbosonen Z^0 und W^- kann wegen ihrer hohen Massen vernachlässigt werden. Charm-Quarks werden bei HERA also hauptsächlich durch Photoproduktion mit quasi-reellen Photonen bei $Q^2 \approx 0$ erzeugt. Die BGF reduziert sich in diesem Fall auf die Photon-Gluon-Fusion (PGF). Der Feynman-Graph der PGF ist in Abb. 2.1 gezeigt.
Wenn das Photon, wie in diesem Fall wie ein punktförmiges Eichboson (direkte Komponmente) wechselwirkt, spricht man von direkten Prozessen.
Das Photon kann aber auch in einen hadronischen Zustand fluktuieren und dann als ein Teilchen mit hadronischer Struktur (aufgelöste Komponente) wechselwirken. Wie beim Proton im Partonmodell wird die Wechselwirkung dann durch Strukturfunktionen beschrieben. Man spricht auch von aufgelösten Prozessen. Bei diesen kann Charm in führender Ordnung entweder über Gluon-Gluon-Fusion

$$\mathrm{g} + \mathrm{g} \rightarrow \mathrm{c} + \bar{c}$$

oder Quark-Antiquark-Annihilation

$$\mathrm{q} + \bar{q} \rightarrow \mathrm{c} + \bar{c}$$

erzeugt werden. (siehe Abb. 2.2)

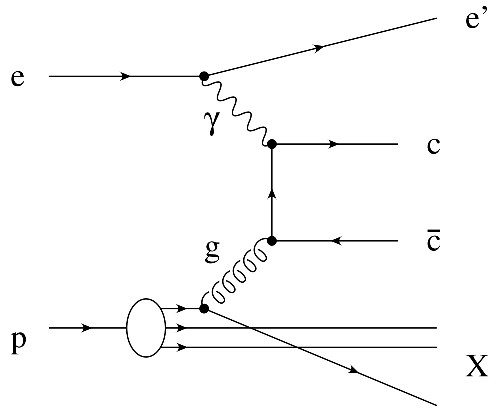
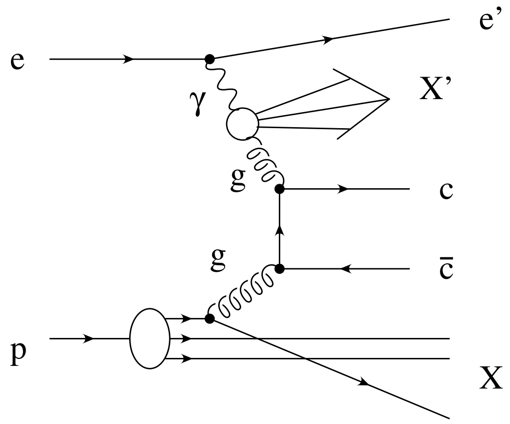


Abbildung 2.1: *Die Photon-Gluon-Fusion*



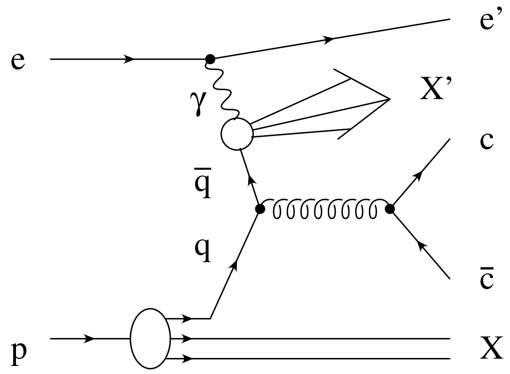


Abbildung 2.2: *a)Die Gluon-Gluon-Fusion* *b) Die Quark-Antiquark Annihilation*

Der quarkinduzierte Anteil 2.2 b) am Charm-Wirkungsquerschnitt ist vernachlässigbar klein [4]. Mit 90% Beitrag liefert die Gluon-Gluon-Fusion, die von der Gluondichte im Photon abhängig ist, den wichtigsten Beitrag. Das Verhältnis zwischen der direkten und der aufgelösten Komponente ist von der Schwerpunktsenergie des γ-p Systems und von der Charm-Quark Masse abhängig. Der Beitrag der aufgelösten Komponente variiert daher von weniger als 5% bei früheren Fixed-Target-Experimenten [5] bis zu $\leq 30\%$ bei den hohen Energien von HERA [6].
Bei der Charm-Produktion dominiert im Gegensatz zur Photoproduktion von leichten Quarks die direkte Komponente. Prozesse höherer Ordnung, die zum Charm-Wirkungsquerschnitt beitragen, werden z.B. in [7, 8] beschrieben.

Kinematik
Zur Vermessung der PGF-Ereignisse benötigt man zusätzlich zu den zwei in der tief-inelastischen Streuung verwendeten kinematischen Variablen, x und y, noch drei weitere. In Photoproduktionsereignissen wird ein Photon mit Energieanteil y vom einlaufenden Elektron abgestrahlt. Das Gluon fliegt in positiver z-Richtung und trägt den Bruchteil x_g des Protonimpulses.

$$P_g = x_g \cdot P_p \tag{2.2}$$

$$x_g = \frac{Q^2 + \hat{s}}{sy} \approx \frac{\hat{s}}{sy} \qquad mit \qquad Q^2 \approx 0 \tag{2.3}$$

Die Schwerpunktsenergie des erzeugten c$\bar{\text{c}}$ Paares beträgt $\sqrt{\hat{s}} \approx \sqrt{4x_g y E_e E_p}$, bei Vernachlässigung der Massen.
Der totale Wirkungsquerschnitt zur Produktion von Charm-Quarks durch PGF γp $\to$ c$\bar{\text{c}}$X wird durch

$$\sigma_{\gamma p \to c\bar{c}X}(y) = \int g(x_g, \mu_f^2)\, \hat{\sigma}_{\gamma g}(\hat{s})\, dx_g \tag{2.4}$$

beschrieben, kann also in den Photon-Gluon-Wirkungsquerschnitt $\hat{\sigma}_{\gamma g \to c\bar{c}}(\hat{s})$ und in die Gluondichte g(x_g,μ_f^2) faktorisiert werden. Die Gluondichteverteilung ist ein Maß für die Wahrscheinlichkeit, ein Gluon mit Impulsanteil x_g im Proton vorzufinden. μ_f ist dabei die Faktorisierungsskala an der die Gluondichte des Protons ausgewertet wird. Der Elektron-Proton-Wirkungsquerschnitt ergibt sich dann in der Weizäcker-Williams-Approximation [9–11] zu:

$$\sigma_{ep \to c\bar{c}X}(y) = \int f_{\gamma/e}(y)\, \sigma_{\gamma p \to c\bar{c}X}(y)\, dy \tag{2.5}$$

wobei

$$f_{\gamma/e}(y) = \frac{\alpha}{2\pi}\left(\frac{1 + (1-y)^2}{y} \cdot \ln \frac{Q^2_{max}}{Q^2_{min}} - 2\frac{1-y}{y}\right) \tag{2.6}$$

der Photonfluß ist und

$$Q^2_{max} = E_e^2(1-y)(5\text{mrad})^2 \tag{2.7}$$

$$Q^2_{min} = \frac{(y \cdot m_e)^2}{1-y} \tag{2.8}$$

Gluondichte
Die Untersuchung von Charm-Quarks ermöglicht es, die Gluondichte zu bestimmen. Dies ist sowohl indirekt über die Messung des totalen Wirkungsquerschnitts möglich (siehe Gleichung 2.4), als auch über die explizite Rekonstruktion von x_g und Bestimmung von g(x_g,μ_f^2). Die Untersuchung von schweren Quarks bietet dabei folgende Vorteile:

- Die Bestimmung der Gluondichte ist direkt möglich. Dies ist bei ihrer Bestimmung aus der Skalenverletzung von F_2(x,Q^2) in der tief-inelastischen Streuung nicht möglich.
- Die BGF ist der einzige Prozeß, der zur direkten Charm-Produktion beiträgt (es gibt keine andere wesentliche Untergrundsquelle, wie z.B. die QCD-Compton-Streuung bei der direkten Messung der Gluondichte in der tief-inelastischen Streuung [8])
- Im Gegensatz zur Photoproduktion von leichten Quarks dominiert die direkte Komponente und liefert den Hauptbeitrag zum Wirkungsquerschnitt.

2.2 Zerfälle von Teilchen mit Charm

Produzierte Charm-Quarks lassen sich durch Rekonstruktion eines Hadrons mit Charm nachweisen, wobei je nach Quark-Konfiguration zwischen Mesonen (c$\bar{\text{q}}$), Baryonen (cqq) und den Charmonium-Zuständen (c$\bar{\text{c}}$) unterschieden wird. Die Wahrscheinlichkeiten der Fragmentierung der Charm-Quarks in die verschiedenen möglichen Hadronen ist nicht genau bekannt. Aus Messungen an e^-e^+ Speicherringen und vor allem aus Studien zur Produktion von λ_c Baryonen [12] folgt jedoch, daß der überwiegende Teil (90%) in Mesonen fragmentiert. Im folgenden wird lediglich auf Zerfälle der Mesonen, insbesondere des D^*, eingegangen.
Vektormesonen mit Charm zerfallen über die starke und elektromagnetische Wechselwirkung in die leichteren pseudo-skalaren D-Mesonen. Die D-Mesonen zerfallen schwach und haben deshalb eine relativ große Lebensdauer, in der Größenordnung von einer Picosekunde. Sie zerfallen in leichtere Zustände, wobei das Charm-Quark in ein leichteres s oder d Quark übergeht. Die angeregten Zustände der D-Mesonen zerfallen stark oder elektromagnetisch in ihre Grundzustände. In Tabellen 2.1 und 2.2 sind die Grundzustände und die ersten angeregten Zustände der D-Mesonen aufgeführt.
Es gibt viele verschiedene Zerfallskanäle mit einem oder mehreren Pionen und Kaonen im Endzustand. Die Verzweigungsverhältnisse sind alle ziemlich klein. Hier

Meson	Quarkinhalt	Masse [MeV/c^2]	Breite [MeV/c^2]	Zerfallskanäle
$D^{*\pm}$	$c\bar{d}$, $\bar{c}d$	2010.0 $\pm$ 0.5	$<$ 0.131	$D^0\pi^\pm$, $D^\pm\pi^0$, $D^\pm\gamma$
D^{*0}, $\bar{D}^{*0}$	$c\bar{u}$, $\bar{c}u$	2006.7 $\pm$ 0.5	$<$ 2.1	$D^0\pi^0$, $D^0\gamma$
$D_S^{*\pm}$	$c\bar{s}$, $\bar{c}s$	2110.0 $\pm$ 1.9	$<$ 4.5	$D_S^\pm\gamma$

Tabelle 2.1: *Die ersten angeregten Zustände der D-Mesonen (D^* Resonanzen, $J^P = 1^-$)*

Meson	Quarkinhalt	Masse [MeV/c^2]	Lebenszeit [ps]
$D^\pm$	$c\bar{d}$, $\bar{c}d$	1869.4 $\pm$ 0.4	1.057 $\pm$ 0.015
D^0, $\bar{D}^0$	$c\bar{u}$, $\bar{c}u$	1864.6 $\pm$ 0.5	0.415 $\pm$ 0.004
$D_S^\pm$	$c\bar{s}$, $\bar{c}s$	1968.5 $\pm$ 0.7	0.467 $\pm$ 0.017

Tabelle 2.2: *Grundzustände der D-Mesonen ($J^P = 0^-$)*

soll nur der Zerfallskanal $D^{*+} \rightarrow D^0\pi_s^+ \rightarrow (K^-\pi^+)\pi_s^+$ genauer betrachtet werden. Die Verzweigungsverhältnisse [13] betragen dabei:

$$B(D^{*+} \rightarrow D^0\pi_s^+) = (68.1 \pm 1.3)\% \tag{2.9}$$

und

$$B(D^0 \rightarrow K^-\pi^+) = (4.01 \pm 0.14)\% \tag{2.10}$$

Die Massendifferenz zwischen D^{*+} und D^0 Meson beträgt

$$\Delta \text{ m} = m_{D^{*+}} \text{ - } m_{D^0} = (145.42 \pm 0.05) \text{ Mev}/c^2$$

Dieser Wert ist etwas größer als eine Pionmasse, so daß der starke Zerfall $D^{*+} \rightarrow D^0\pi_s^+$ gerade noch möglich ist.
Die in dem Zerfall zur Verfügung stehende Energie ist mit

$$\text{Q} = [m_{D^{*+}} \text{ - } (m_{D^0} + m_{\pi_s^+})]c^2 = 5.8 \text{ MeV}$$

sehr gering und stellt einen großen Vorteil bei der Rekonstruktion des Zerfalls dar[14]. Im Ruhesystem des D^{*+} ergibt sich für die Impulse von D^0 und π_s^+ ein Wert von

$$p^* = 39 \text{ MeV/c}.$$

Der Öffnungswinkel zwischen Pion- bzw. D-Flugrichtung und der D^{*+}-Flugrichtung ist im Laborsystem daher sehr klein ($< 10^0$ für $p_t(D^{*+}) > 2.5$ GeV/c). Das Pion π_s^+ spiegelt deshalb neben der Ladung auch die Richtung des D^{*+}-Mesons wider. Der Pionimpuls im Laborsystem beträgt näherungsweise

$$p_{\vec{\pi}_s^+} \approx \frac{\sqrt{m_{\pi_s^+}^2 c^4 + p^{*2}c^2}}{m_{D^{*+}}c^2} \cdot p_{\vec{D}^{*+}} \approx \frac{1}{14} p_{\vec{D}^{*+}},$$

weswegen es auch als langsames (slow) Pion π_s bezeichnet wird.
Bei HERA sind die Transversalimpulse der produzierten D^*-Mesonen relativ niedrig, weshalb viele langsame Pionen extrem kleine Transversalimpulse nahe der Nachweisgrenze besitzen.
Im Standard-Modell wird der Zerfall der Charm-Quarks in die leichten s und d Quarks durch den schwachen, geladenen Strom beschrieben. Aus dem Charm-Quark ensteht dabei ein $W^{\pm}$-Boson und ein leichtes Quark. Die Zerfallswahrscheinlichkeiten sind proportional zu den CKM-Matrixelementen

$$
\begin{array}{ll}
c \rightarrow sW^+ \propto |V_{cs}|^2 \approx \cos^2\theta_c \approx (0.98)^2 & (Cabibbo - erlaubt) \\
c \rightarrow dW^+ \propto |V_{cd}|^2 \approx \sin^2\theta_c \approx (0.23)^2 & (Cabibbo - unterdrückt)
\end{array}
$$

Das Eichboson zerfällt dann entweder hadronisch in ein q$\bar{\text{q}}$-System oder leptonisch in ein $l\bar{\nu}_l$-System.
Die Wahrscheinlichkeiten für den W$^+$-Zerfall sind ebenfalls proportional zu den CKM-Matrixelementen

$$
\begin{array}{ll}
W^+ \rightarrow u\bar{\mathrm{d}} \propto |V_{ud}|^2 \approx \cos^2\theta_c \approx (0.98)^2 & (Cabibbo - erlaubt) \\
W^+ \rightarrow u\bar{\mathrm{s}} \propto |V_{ud}|^2 \approx \sin^2\theta_c \approx (0.23)^2 & (Cabibbo - unterdrückt)
\end{array}
$$

Welche Zerfälle bevorzugt und welche unterdrückt werden, ergibt sich unter Berücksichtigung des Cabibbo-Winkel θ_c. Übergänge von c $\rightarrow$ d oder W$^+$ $\rightarrow$ u$\bar{\text{s}}$ heißen einfach Cabibbo-unterdrückt. Zerfälle, die beide Übergänge enthalten, heißen doppelt Cabibbo-unterdrückt.

Kapitel 3

Das H1 Experiment

3.1 Der Speicherring HERA

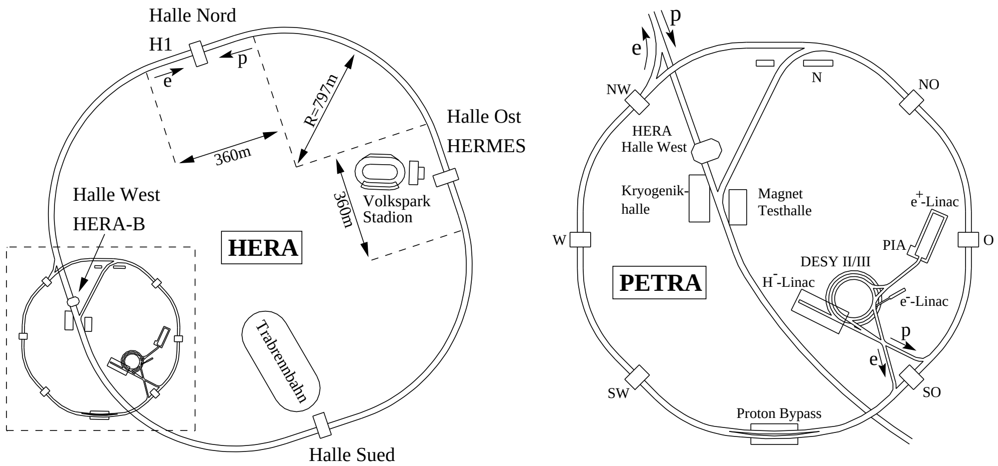


Abbildung 3.1: *Der HERA Speicherring und seine Vorbeschleuniger.*

In der **H**adron-**E**lektron-**R**ing-**A**nlage HERA werden Elektronen mit einer Energie von 27.5 GeV und Protonen mit einer Energie von 820 GeV bei einer Schwerpunkts-energie E_{CMS} von 300 GeV zur Kollision gebracht. Insgesamt nutzen 4 Experimente (H1, ZEUS, HERA-B, HERMES) HERA, wobei mit den Detektoren H1 [15] und ZEUS [16] ep - Wechselwirkungen untersucht werden.
In HERA gibt es für Elektronen und Protonen zwei getrennte Speicherringe, die sich in einem ca. 10-25 m tiefen ringförmigen Tunnel von 6.3 km Umgang befinden. Von 1994 bis 1997 verwendete man Positronen anstelle von Elektronen, da sich mit ihnen ein besseres Vakuum im Ring und damit längere Strahllebensdauern erreichen ließen. Für die kommende Run-Periode werden jedoch wieder Elektronen benutzt.
Bevor die beiden Teilchensorten separat in den HERA-Ring injiziert werden, durchlaufen sie ein System von Vorbeschleunigern[1]. Positronenbündel (ein Bündel enthält

[1] Endenergien der Vorbeschleuniger jeweils in Klammern

bis zu $0.4 \cdot 10^{11}$ Positronen) aus einem 500 MeV Linearbeschleuniger beschleunigt man nacheinander in den kleineren Speicherringen DESY II (7 GeV) und PETRA II (14 GeV) auf ihre Einschußenergie.
Eine Stripperfolie sorgt dafür, daß H^--Ionen, die zuvor mit einem Linearbeschleuniger auf 50 MeV beschleunigt wurden, die Elektronen verloren gehen, so daß Protonenbündel zuerst den Speicherring DESY III (7.5 GeV) und dann PETRA II(40 GeV) durchlaufen, bevor sie in HERA eingespeist werden.
Da in PETRA II nur jeweils maximal 70 Bündel von Positronen bzw. Protonen gehalten werden und man HERA mit 210 Bündeln füllen will, muß dieser Vorgang dreimal wiederholt werden.
In HERA werden die Positronen weiter auf 27.5 GeV, die Protonen auf ihre Endenergie von 820 GeV beschleunigt. Begrenzt werden diese Endenergien einerseits durch die Beschleunigungsspannung der supraleitenden Hohlraumresonatoren, die den Energieverlust der Positronen durch Synchrotronstrahlung ausgleichen. Andererseits ist die Endenergie der Protonen durch die Feldstärke der supraleitenden Magneten (4.68 T) begrenzt, welche benötigt wird, um die Protonen auf ihrer Bahn im Speicherring zu halten.
Die Teilchenbündel durchdringen sich alle 96 ns an den vorhergesehenen Wechselwirkungspunkten (1 B.C. = bunch crossing). Wegen des totalen Wirkungsquerschnitts der ep - Streuung findet im Mittel nur alle 10^5 Kreuzungen eine im Detektor beobachtbare Streuung statt, was einer Ereignisrate von 100 Hz entspricht.
In Tab. 3.1 sind einige für H1 wichtige technische Parameter von HERA für die letzten beiden Datennahmeperioden aufgeführt. Wie man sieht, liegt die maximale Luminosität bereits nahe am Designwert und auch die über ein Jahr integrierte Luminosität ist 1997 [17] nur noch um den Faktor 3 davon entfernt.
Für das Jahr 2000 ist sogar noch eine Erhöhung der Luminosität geplant, die den Designwert um den Faktor 5 übersteigen soll [18]. Diesen Wert will man einerseits durch höhere Protonströme, andererseits durch ein besser Fokussierungssystem in der Wechselwirkungszone erreichen.

3.2 Der H1-Detektor

Der schematische Aufbau des H1-Detektors ist in Abb. 3.2 dargestellt. Eine ausführlichere Beschreibung des H1-Detektors und seiner Komponenten findet man in [19]. Hier werden nur Komponenten beschrieben, die für diese Arbeit relevant sind. In Abb. 3.2 ist auch das in H1 verwendete Koordinatensystem eingezeichnet.
Der Ursprung des Koordinatensystems liegt dabei im nominellen Wechselwirkungspunkt und die Protonstrahlrichtung wird als Vorwärtsrichtung definiert. Der Detektor besitzt mehr Komponenten in Vorwärtsrichtung, da sich das hadronische Schwerpunktssystem wegen der unterschiedlichen Strahlenergien in der Regel in Richtung des auslaufenden Protons bewegt. Der nominelle Wechselwirkungspunkt ist von zylindrischen Spurkammern umgeben, welche vom LAr-Kalorimeter umschlossen werden. Das Kalorimeter besteht dabei aus einem inneren, elektromagnetischen und dem äußeren, hadronischen Teil und ist von einer supraleitenden Spule umgeben,

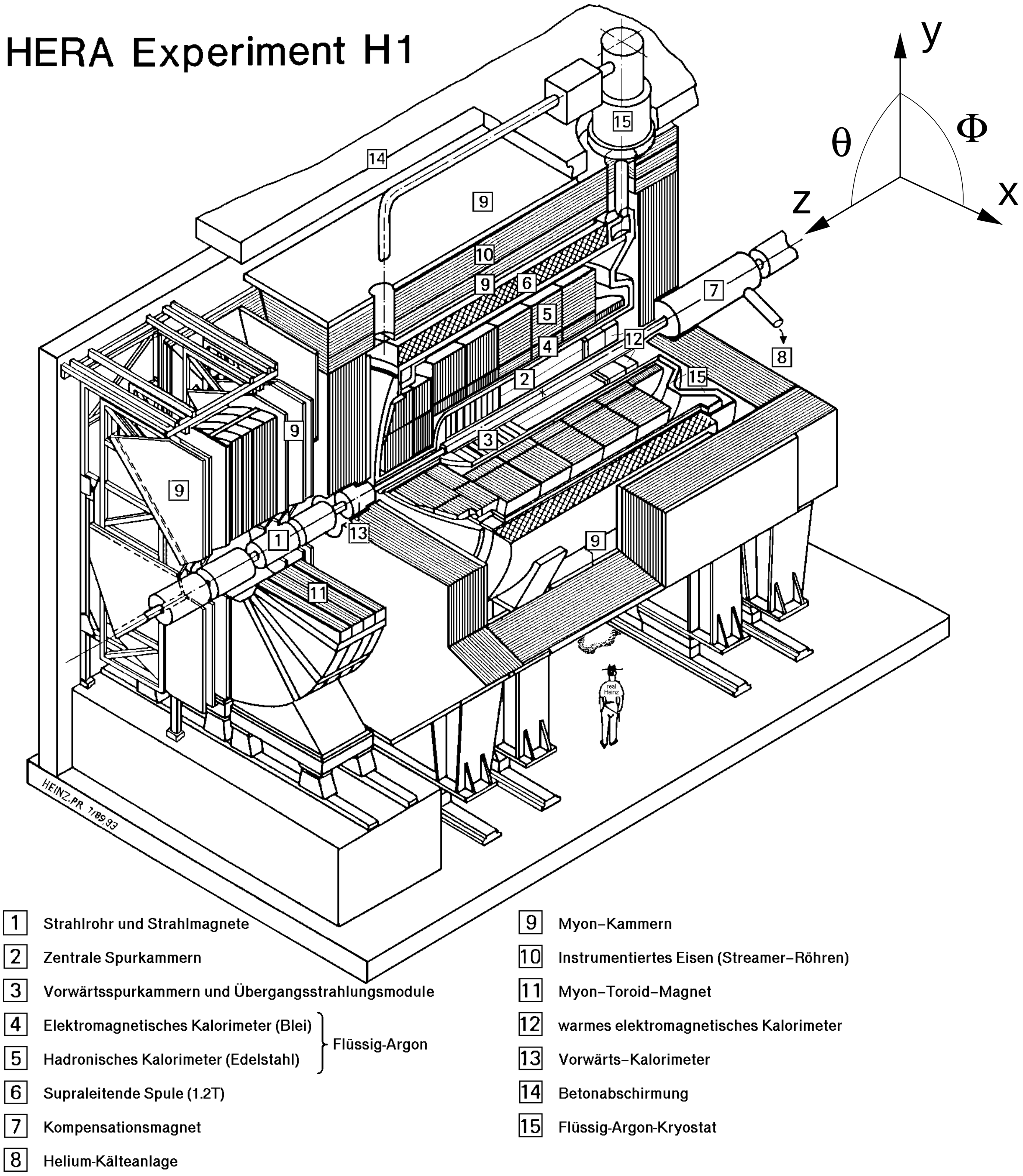


Abbildung 3.2: *Schematische Ansicht des H1-Detektors.*

HERA-Parameter		Design e^+	Design p	1996 e^+	1996 p	1997 e^+	1997 p
Strahlenergie	(GeV)	30.0	820	27.5	820	27.5	820
Schwerpunktsenergie	(GeV)	314		300		300	
mittl. Strahlstrom	(mA)	58	163	21	60	28	74
Teilchenpakete		210	210	175	175	175	175
Max. $\mathcal{L}$	$(\mathrm{cm}^{-2}\mathrm{s}^{-1})$	1.50×10^{31}		0.84×10^{31}		1.01×10^{31}	
Max. Spez. $\mathcal{L}$	$(\mathrm{cm}^{-2}\mathrm{s}^{-1}\mathrm{mA}^{-2})$	4.0×10^{29}		6.2×10^{29}		6.0×10^{29}	
$\int \mathcal{L}$ dt(HERA)	$(\mathrm{pb}^{-1}\mathrm{y}^{-1})$	100		13.3		32.4	
$\int \mathcal{L}$ dt(H1 genutzt)	$(\mathrm{pb}^{-1}\mathrm{y}^{-1})$	100		8.9		27.3	

Tabelle 3.1: *Einige Parameter des Speicherrings HERA für die Datennahmeperioden 1996 und 1997 im Vergleich zu den Designwerten.*

die parallel zum Strahlrohr ein homogenes Magnetfeld von 1.15 T erzeugt. Geladene Teilchen bewegen sich dadurch auf Helixbahnen, die sich mit Hilfe der Spurkammern rekonstruieren lassen. So kann man die Impulse der erzeugten Teilchen, sowie die Ereignistopologie und die Position des Vertex messen. Die dazugehörigen Energien der erzeugten Teilchen lassen sich in den Kalorimetern messen.

3.2.1 Das Spurkammersystem

Abb. 3.3 zeigt die schematische Seitenansicht der H1 Spurkammern. Ergänzt wurde das Spurkammersystem 1996 durch zwei Silizium-Vertex-Detektoren. Der zentrale Silizium-Vertex-Detektor (CST) dient der Verbesserung der Vertexauflösung($\Delta z \approx 60\,\mu m$) in Zusammenarbeit mit den zentralen Spurkammern, während der rückwärtige Silizium-Vertex-Detektoren (BST) eine Vertexbestimmung bis zu Streuwinkeln von 175^0 erlaubt.

Zentrale Driftkammern
Im Bereich des Polarwinkels θ von etwa 20^0 bis 160^0 (bezogen auf den nominellen Wechselwirkungspunkt) erfolgt die Spurrekonstruktion mit Hilfe von 4 Driftkammern, von denen zwei der Rekonstruktion in der rϕ-Ebene (Central Jet Chambers; CJC1, CJC2) und zwei der Rekonstruktion der z-Koordinate (Central Inner/Outer Z-Chambers; CIZ/COZ) dienen. Einen radialen Querschnitt durch die zentralen Spurkammern kann man in Abb. 3.4 sehen.
Die jeweils mit einem Gasgemisch aus Ar und Ethan (im Verhältnis 50/50) gefüllten Jetkammern werden durch Ebenen von Kathodendrähten, die parallel zur Strahlachse gespannt sind, in Zellen unterteilt. In der Mitte jeder Zelle befindet sich eine Ebene

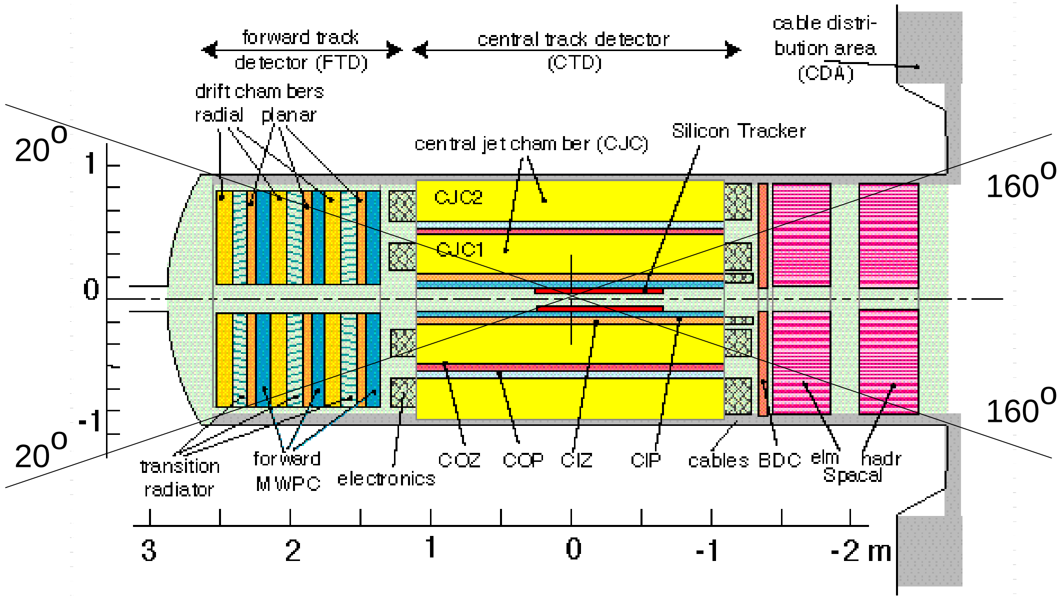


Abbildung 3.3: *Schematische Seitenansicht der Spurkammern in H1. Nicht eingezeichnet ist der rückwärtige Siliziumspurdetektor BST*

von Anodendrähten (Signaldrähte). Die innere Driftkammer CJC1 besteht aus 30 Zellen mit jeweils 24 Signaldrähten, die äußere Driftkammer CJC2 hat 60 Zellen mit jeweils 32 Signaldrähten. Zwischen den Anoden- und den Kathodendrahtebenen entsteht ein nahezu homogenes elektrisches Feld, in welchem die von einem durchfliegenden ionisierenden Teilchen hervorgerufene Elektronenwolke auf die Anodendrähte zudriftet. Einerseits wird sie dabei durch das elektrische Feld beschleunigt, andererseits durch elastische Stöße mit Gasatomen abgebremst, so daß sich im Gleichgewicht eine konstante Driftgeschwindigkeit v_D einstellt. So läßt sich aus der Driftzeit der Ort der Primärionisation bestimmen, wobei es jedoch zu Doppeldeutigkeiten den sogenannten Spiegeltreffern kommt, da man anhand der Driftzeit nicht unterscheiden kann, auf welcher Seite des Signaldrahtes der Treffer wirklich liegt.

Durch inhomogene elektrische Felder und hohe Feldstärken in unmittelbarer Nähe der Anodendrähte können die driftenden Elektronen genügend Energie gewinnen um ein Gasatom zu ionisieren.Da die dabei freiwerdenden Elektronen weitere Gasatome ionisieren können, kommt es zu einer Elektronenlawine. Erst wenn die Elektronenwolke den Gasverstärkungsbereich ca. 1cm um den Anodendraht erreicht hat entsteht ein meßbares Signal, welches verstärkt und mit 104 MHz von einem FADC digitalisiert wird. So kann der Signalzeitpunkt mit einer Auflösung von ca. 4 ns bestimmt werden, was bei einer Driftgeschwindigkeit von ca. 50 μm/ns einer Ortsauflösung von etwa 200 μm in der r-ϕ-Ebene entspricht.

Alle Anodendrähte werden an beiden Enden ausgelesen und ermöglichen so über

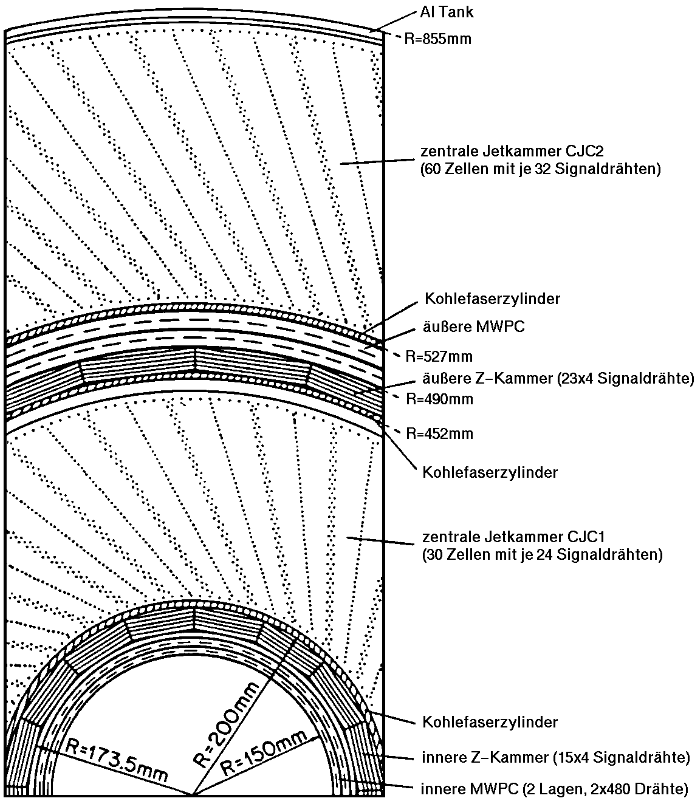


Abbildung 3.4: *Radialansicht der zentralen Spurkammern*

Ladungsteilung eine Messung der z-Koordinate der Primärionisation. Die Auflösung in z ist mit ca. 2.5 cm wesentlich schlechter als die in der r-ϕ-Ebene.
Die einzelnen Zellen der Driftkammer sind nicht genau radial ausgerichtet, sondern um 30^0 verdreht. Dadurch ist die Driftrichtung in Anwesenheit des magnetischen Feldes nahezu senkrecht zur Flugrichtung hochenergetischer, steifer Spuren. So erhält man einerseits die optimale Spurauflösung, andererseits kreuzen die Spuren dadurch meistens mehrere Zellen, wodurch die Spurerkennung erleichtert wird, da scheinbare Spuren, die aus Spiegeltreffern bestehen, in den Nachbarzellen nicht fortgeführt werden.
An der Innen- und Außenseite der CJC1 schließen sich die dünnen Driftkammern CIZ und COZ an, deren Drähte ringförmig um die Strahlachse in vier Lagen aufgespannt sind, mit einer Unterteilung in 15 (CIZ) bzw. 24 (COZ) Segmente. Die Auflösung in z liegt bei 260 μm für die CIZ bzw. 200 μm für die COZ.

Zentrale Proportionalkammern
Die Haupaufgabe der inneren und äußeren Proportionalkammern (CIP/COP; Central Inner/Outer Proportional Chamber) ist es, für die Triggerstufe 1 Raumpunkte von Teilchenspuren zu liefern, um dort durch eine schnelle Vertexerkennung bereits Untergrundereignisse, Rauschen und Synchrotronstrahlung zu verwerfen. Die Kammern bestehen jeweils aus einer Doppellage von Drähten und sind in z-Richtung 60-fach (CIP) bzw. 18-fach (COP) segmentiert und in der rϕ Ebene jeweils 16-fach unterteilt. Wenn vier bzw. drei Drahtlagensegmente innerhalb eines Sektors ein Signal aufweisen, wird daraus ein sogenannter Strahl gebildet, der in der rz - Ebene eine Gerade ist. Die Verlängerung dieser Strahlen auf die z-Achse wird in das sogenannte z-Vertex Histogramm eingetragen, welches auf der zweiten Triggerstufe genutzt wird.

Rückwärtige Driftkammer
Die rückwärtige Driftkammer (Backward Drift Chamber,BDC) ist 1.42 m vom nominellen Wechselwirkungspunkt aus in Rückwärtsrichtung installiert. Sie besteht aus vier Doppeldrahtlagen, die in rϕ Ebene jeweils in 8 Sektoren unterteilt sind und deckt den Polarwinkelbereich von $153^0 < \theta < 177^0$ ab. Mit der BDC wird vor allem die Winkelrekonstruktion verbessert.

Vorwärtsspurkammern
In dem Bereich von $5^0 < \theta < 25^0$ dient das vordere Spurkammersystem zur Spurrekonstruktion. Es besteht aus drei Supermodulen, die sich jeweils aus drei gegeneinander um 60^0 gedrehten planaren Driftkammern, einer Proportionalkammer , einer Übergangsstrahlungskammer und einer radialen Driftkammer zusammensetzen.

3.3 Das Datennahme- und Triggersystem von H1

Aufgabe des Triggersystems ist es, Untergrundereignisse von physikalisch interessanten ep - Wechselwirkungen zu trennen. Bei HERA übersteigt die Rate der Untergrundereignisse die der ep - Wechselswirkungen dabei um mehrere Größenordnungen. Vier Quellen des Untergrundes werden im folgenden beschrieben.

Strahl-Gas-Ereignisse
Wechselwirkungen der Strahlprotonen mit dem im Strahlrohr verbliebenen Restgas werden als Strahl-Gas-Reaktionen bezeichnet und sind eine wesentliche Untergrundquelle. Aus gemessenen Wirkungsquerschnitten für pp-Kollisionen kann man eine Abschätzung der Rate von etwa 1 kHz/m [20] für einen Protonenstrom von 100 mA machen. Strahl-Gas-Ereignisse können vor allem durch eine Verschiebung des z-Vertex von ep-Wechselwirkungen unterschieden werden.

Strahl-Wand-Ereignisse
Andere von Protonen induzierte Untergrundereignisse sind Strahl-Wand-Wechselwirkungen, bei denen Haloprotonen auch Strahlrohr und Detektorkomponenten treffen. Die Rate dieser Reaktionen beträgt 40-250 kHz. [21]

Synchrotronstrahlung
Geladene Teilchen strahlen bei Beschleunigung Energie ab, insbesondere wenn sie auf eine Kreisbahn gezwungen werden. Der Energieverlust ist dabei proportional m^{-4}. Die so ausgesandte Synchrotronstrahlung ist daher im Fall der Protonen irrelevant und kann bei den Elektronen durch Kollimatoren größtenteils unterdrückt werden.

Kosmische Strahlung
Kosmische Höhenstrahlung besteht beim Erreichen der Erde u.a. aus Myonen, die den Detektor mit einer Rate von etwa 1 kHz treffen. Durch Fordern eines Ereignisvertex aus der Wechselwirkungszone läßt sich diese Rate jedoch leicht reduzieren und liefert daher keinen nennenswerten Beitrag zum Untergrund.

Vom H1 Detektor sind bei einem Ereignis 270 000 Signalkanäle auszulesen, wofür eine Zeit von 1 ms benötigt wird, die für den Detektor als Totzeit in der Datennahme verloren geht. Um unnötige Totzeiten möglichst zu verhindern, müssen die oben beschriebenen Untergrundreaktionen von den ep-Wechselwirkungen getrennt werden. Beim H1-Experiment verwendet man zur Trennung ein vierstufiges Triggersystem, wobei die einzelnen Stufen mit L1 bis L4 (L steht für Level) bezeichnet werden. Das Triggersystem soll im folgenden kurz vorgestellt werden.

Die erste Triggerstufe (L1)
Um zu einer Triggerentscheidung zu gelangen, werden die Signale der einzelnen Subdetektoren benötigt. Jedoch liegen diese Daten, begrenzt durch Drift-, Sammel- und Übertragungsgeschwindigkeit, erst nach 2.3 μs vor. Deshalb werden die Signale der einzelnen Subdetektoren in sogenannten Pipelines gehalten, bis das zentrale Triggersystem nach 24 B.C. eine globale L1 Entscheidung fällt. Die Triggersignale der insgesamt neun Subdetektoren werden Triggerelemente genannt, wobei bis zu 192 solcher Signale vorliegen. Diese werden durch logische Verknüpfungen in der zentralen Triggerlogik (CTL, **C**entral **T**rigger **L**ogic) zu 128 Subtriggern kombiniert. Ist bereits eine dieser 128 Subtriggerbedingungen erfüllt, wird ein L1Keep Signal ausgelöst und das Ereignis von den nachfolgenden Triggerstufen geprüft. Auf L1 wird so die Eingangsrate bereits auf unter 5 kHz reduziert. Falls einer der Subtrigger diese Rate übertrifft, wird er zusätzlich mit einem prescale, d.h. nur jedes n-te Ereignis wird akzeptiert, versehen. Durch die Zwischenspeicherung der Ereignisse in den Pipelines wird durch L1 erst Totzeit erzeugt, wenn ein L1Keep Signal erzeugt wird.

Die zweite Triggerstufe (L2)
Grundlage für die Entscheidung der zweiten Triggerstufe bilden im Wesentlichen die Triggerinformationen der einzelnen Subdetektoren, die L1-Entscheidungen sowie statistische Angaben wie beispielsweise das z-Vertex-Histogramm. Zur Verarbeitung dieser Daten werden zwei verschiedene Systeme, der L2NN (L2 **N**euronaler **N**etzwerktrigger)[22] und der L2TT (L2 **T**opologischer **T**rigger)[23]

verwendet. Die maximale Entscheidungszeit auf Level2 liegt bei 20 μs. Wird ein Ereignis angenommen (L2Keep), beginnt die totale Auslese der Detektordaten. Bei einer negativen Entscheidung (L2Reject) wird die Auswertung abgebrochen und die L1-Pipeline wieder in Gang gesetzt. Der neuronale Netzwerktrigger, der für diese Arbeit von besonderem Interesse ist, wird im nächsten Kapitel ausführlicher beschrieben.

Die dritte Triggerstufe (L3)
Die dritte Triggerstufe soll bei einem L2keep-Signal gestartet werden und als ein auf Mikroprozessoren basierter Softwaretrigger agieren. Als Entscheidungszeit sind dabei 800 μs vorgesehen, jedoch wird bisher diese Triggerstufe nicht eingesetzt, so daß die Ratenreduktion auf maximal 50 Hz bereits von der zweiten Triggerstufe geleistet werden muß.

Die vierte Triggerstufe (L4)
Ist ein Ereignis von allen vorhergehenden Triggerstufen akzeptiert worden, gelangt es zur vierten Stufe, die aus einer Prozessorfarm mit 36 Mikroprozessoren besteht. Die Daten werden in einem Ringpuffer mit einer Kapazität von 30 Ereignissen zwischengespeichert. Mit der vollen Detektorinformation wird dann asynchron eine vereinfachte Ereignisrekonstruktion durchgeführt und innerhalb von 100 ms eine Triggerentscheidung getroffen. Die Ausgangsrate von Level 4 ist, bedingt durch die maximale Datentransferrate von etwa 1.2 MByte/s, auf 10 Hz begrenzt, da ein komprimiertes Ereignis ungefähr eine Größe von (50 -100)kByte besitzt.

Die Ereignisrekonstruktion
Die Ereignisse, die das vierstufige Triggersystem passiert haben, werden als Rohdaten auf Magnetband geschrieben. Auf einer weiteren fünften Stufe, die in Analogie L5 genannt wird, wird dann offline innerhalb weniger Tage eine vollständige Ereignisrekonstruktion durchgeführt, wobei weitere Untergrundereignisse erkannt und verworfen werden. Die physikalisch relevanten Ereignisse werden dabei vordefinierten Physikklassen zugeordnet und auf **D**ata **S**ummary **T**apes (DST) gespeichert, wo sie zur weiteren Physikanalyse zur Verfügung stehen.

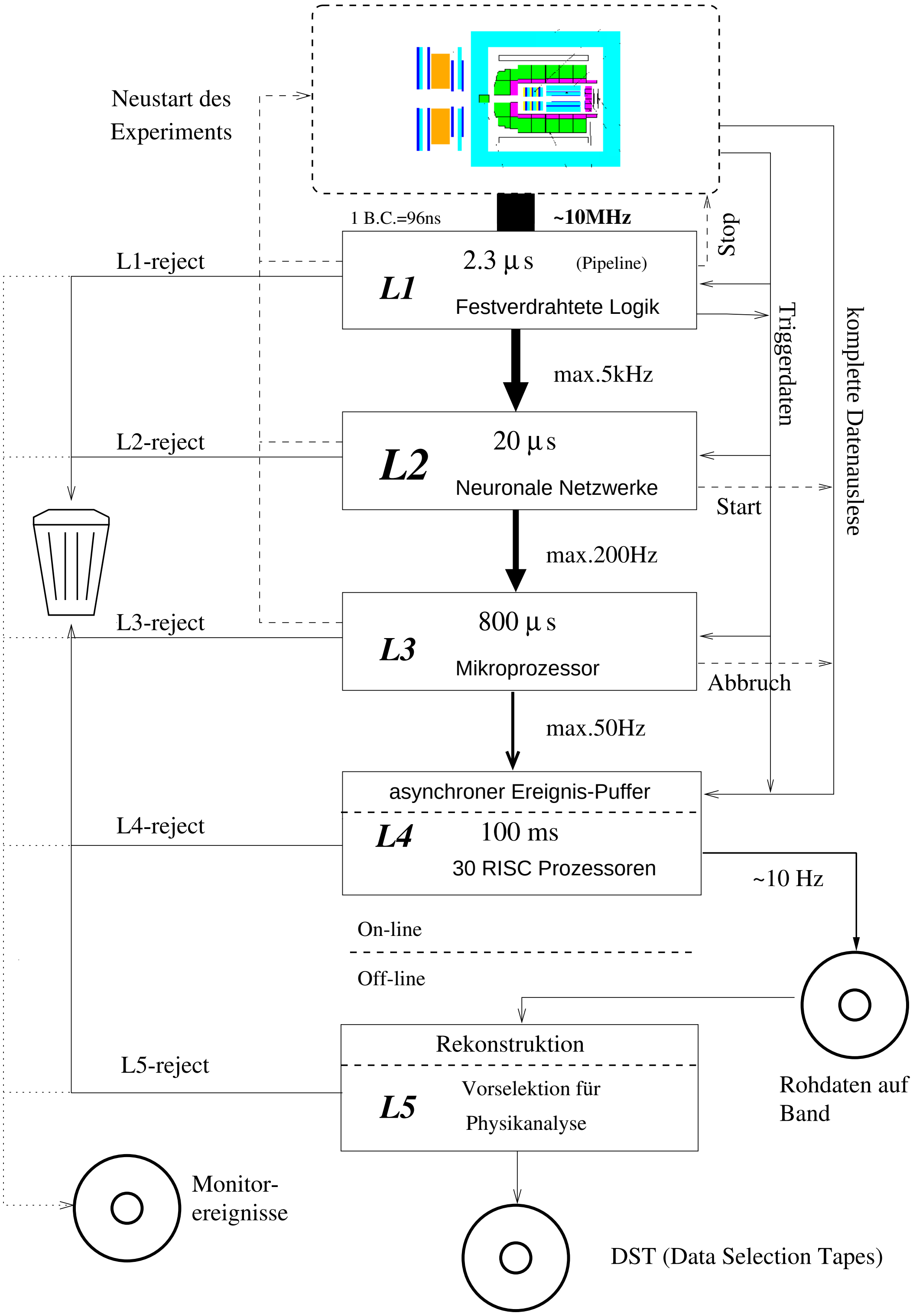


Abbildung 3.5: *Das Trigger- und Datennahmesystem des H1-Experiments. Signale sind gestrichelt, Datenflüsse durchgezogen und gepunktet dargestellt. Die angegebenen Raten sind maximale Designwerte.*

Kapitel 4

Neuronale Netze und der L2 Trigger

4.1 Biologische Grundlagen

Das menschliche Gehirn besteht aus ungefähr 10^{11} Neuronen oder Nervenzellen [24]. In Abb. 4.1 erkennt man deutlich die drei funktionellen Einheiten einer Nervenzelle: den Zellkörper (Neuron), die Dendriten und das Axon.

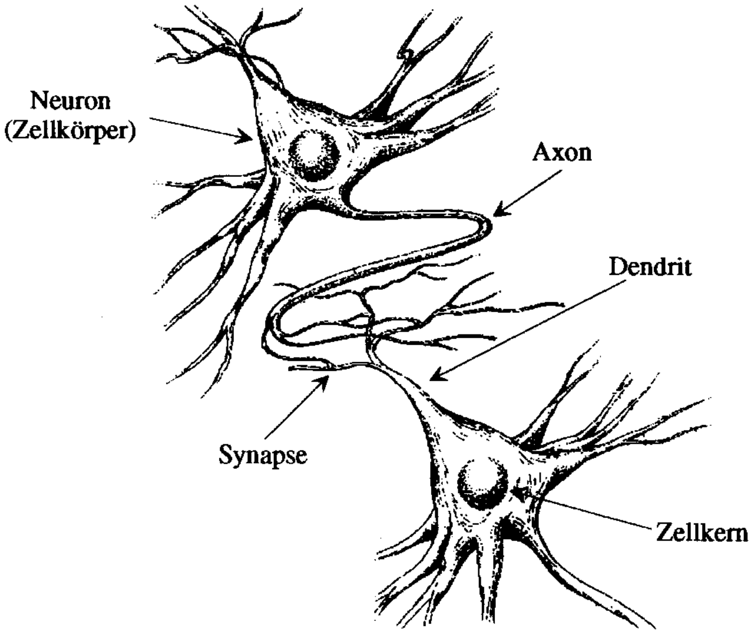


Abbildung 4.1: *schematische Darstellung zweier Neuronen*

Der Zellkörper enthält den Zellkern und andere lebenswichtige Komponenten der Zelle. Fein verzweigte Ausläufer des Zellkörpers, die Dendriten, stellen Kontakte zu

benachbarten Neuronen her und empfangen Signale von diesen. Das Axon ist ein kabelähnlicher Ausläufer des Zellkörpers, der das Ausgabeelement einer Nervenzelle bildet und am Ende ebenfalls fein verzweigt ist. Es stellt über sogenannte synaptische Verbindungen einige tausend Kontakte mit anderen Neuronen her. Je nach Länge des Axons können Signale auch in entfernte Gehirnregionen übertragen werden.
Bei der Signalübertragung wird durch die Synapsen im allgemeinen auf chemischen Wege eine einlaufende Erregungswelle auf die Dendriten der kontaktierten Nervenzellen übertragen. Dabei werden Botenstoffe, sogenannte Neurotransmitter freigesetzt, die das elektrische Potential in der Empfänger Zelle erhöhen oder senken, je nachdem ob es sich um inhibitorische (hemmende) oder exzitatorische (anregende) synaptische Verbindungen handelt. Überschreitet das zeitlich und räumlich aufsummierte Potential einen bestimmten Schwellenwert, wird das Neuron erregt und feuert mit einem bestimmten Aktionspotential. Dieser Puls wird dann über die Synapsen an andere Zellen übertragen. Detaillierte Beschreibungen der Neurologie des Gehirns finden sich u.a. in [25–27].
Bleibt die Frage zu klären, wie das Gehirn Informationen speichert. Da die Neuronen lediglich Signale übertragen, kann die Information offensichtlich nur in den synaptischen Verbindungen, bzw. der Topologie des Gehirns kodiert sein. Lernen ist also ein Vorgang, bei dem neue synaptische Verbindungen entstehen, bzw. synaptische Verbindungsstärken modifiziert werden. Die enorme Leistungsfähigkeit des Gehirns liegt dabei nicht in der Komplexität der einzelnen Nervenzelle, sondern in ihrer großen Anzahl.

4.2 Das mathematische Modellneuron

Die Beobachtung des Schwellwertverhaltens biologischer Neuronen motivierte zu einer mathematischen Modellierung der Neuronen und ihrer Verbindungen, um so das menschliche Gehirn zu simulieren. Ein einfaches Neuronen-Modell wurde bereits 1943 von McCulloch und W.Pitts vorgeschlagen [28].
Eingabewerte x_i werden an das j-te Neuron geschickt, wo sie gewichtet aufsummiert werden und von der Summe anschließend ein Schwellenwert t_j abgezogen wird. Die Gewichtung ω_{ij} entspricht dabei der synaptischen Verbindungsstärke. Auf das Ergebnis dieser Berechnung wird noch eine Stufenfunktion g angewendet:

$$y_j = g\left(\sum_{i=1}^{N} \omega_{ij} x_i - t_j\right). \tag{4.1}$$

mit:

$$g(x) = \theta(x) := \begin{cases} 0 & \text{für} \quad x \leq 0 \\ 1 & \text{für} \quad x > 0. \end{cases} \tag{4.2}$$

Das Neuron feuert, wenn seine Eingabe den Schwellenwert t_j überschreitet.
Während McCulloch und Pitts binäre Neuronen verwendeten, ist es biologisch plausibler, für die sogenannte Aktivierungsfunktion g eine sigmoidale Funktion (siehe

Abb. 4.2 zu verwenden, deren Sättigungsverhalten für $x \to \pm\infty$ der neurologischen Signalfunktion sehr nahe kommt.

$$g(x) = \frac{1}{1 + \exp(-\frac{x}{T})} \qquad \text{mit} \qquad T \neq 0 \;, \tag{4.3}$$

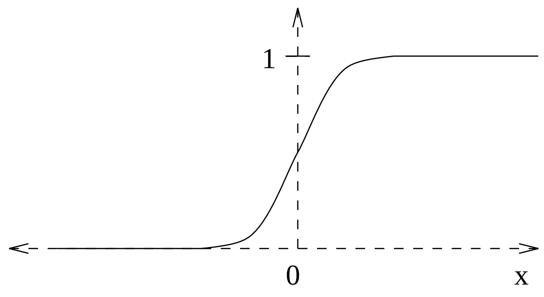


Abbildung 4.2: *Die Fermi-Funktion als sigmoidale Aktivierungsfunktion.*

Durch die Wahl von T wird die Steilheit der Kurve am Nullpunkt definiert, wobei T als eine Art Temperatur interpretiert werden kann.

Zur Veranschaulichung sieht man in Abb. 4.3 die Skizze eines mathematischen Modellneurons.

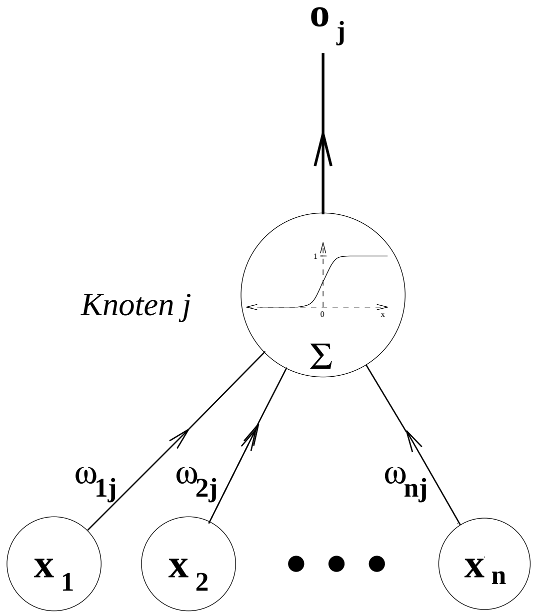


Abbildung 4.3: *Das mathematische Modell eines Neurons. Die Eingabewerte* $\mathrm{x_i}$ *werden mit* ω_{ij} *individuell gewichtet und im Knoten* j *aufsummiert. Über die Aktivierungsfunktion wird aus der Summe das Ausgabesignal* $\mathrm{o_j}$ *gebildet.*

In ihrer Einfachheit und ihrem Aufbau sind die Modellneuronen vergleichbar mit den Nervenzellen im Gehirn, wenngleich sie diese natürlich nicht exakt simulieren.

Ein künstliches Neuron gibt z.B. im Gegensatz zu einem echten, das eine Folge von Impulsen ausgibt, nur eine Zahl aus die seiner Aktivität entspricht.
Typische Leistungsmerkmale des menschlichen Gehirns, wie Mustererkennung, Assoziation, Abstraktion, Parallelität, Fehlertoleranz und Lernfähigkeit versucht man durch künstliche neuronale Netze (KNN), bestehend aus untereinander verbundenen Modellneuronen, die auch Knoten genannt werden, nachzubilden. Die künstlichen Netze können allerdings bislang ein typisches Verknüpfungsmuster von Dendriten und Axonen weder in der Anzahl noch in der Komplexität der Verschaltungen nachbilden.

4.3 Charakterisierung von neuronalen Netzen

Jedes neuronale Netz wird durch die Merkmale

- Netzwerkarchitektur
- Lernalgorithmus

charakterisiert. Auch der Algorithmus, nach dem ein neuronales Netz bei gegebenen Gewichten aus dem Eingabevektor den Ausgabevektor berechnet und insbesondere die dabei verwendete Aktivierungsfunktion, kann variieren.

4.3.1 Netzwerkarchitektur

Die verwendete Netzwerkarchitektur, also die Art und Weise nach der die einzelnen Neuronen miteinander vernetzt sind, hängt von der jeweiligen Problemstellung ab. Einige Beispiele verschiedener Netzwerkarchitekturen sind:

- Feed-Forward-Netze, bestehend aus mehreren Schichten von Neuronen, wobei der Informationsfluß zwischen den Schichten nur in eine Richtung stattfindet und keine Verbindungen innerhalb einer Schicht existieren. Sie werden z.B. zur Parameterschätzung verwendet.
- Rückgekoppelte Netzwerke (z.B. Hopfield-Netze [29]), bei denen die Ausgabe des Netzes auch wieder als Eingabe verwendet wird, der Informationsfluß also in beiden Richtungen stattfindet. Sie eignen sich z.B. zur Analyse von verrauschten Mustern, da iterativ das Rauschen des Anfangseingabemusters reduziert werden kann.
- Neuronale Netze, bei denen die Neuronen jeweils mit ihren Nachbarzellen verbunden sind, also laterale Verschaltungen existieren (z.B. Kohonens Self-Organizing-Map [30] oder iterative Netzwerke wie z.B. Growing Cell Structures [31] oder Growing Neural Gas [32]). Solche Netzwerke werden z.B. zur Datenvorverarbeitung und Datenreduktion verwendet (siehe Kap. 7).

Weitere Informationen über neuronale Netzwerke findet man z.B. in [33, 34]

4.3.2 Der Lernalgorithmus

Neuronale Netze werden für vielfältige Aufgaben wie Mustererkennung, Approximation, Klassifikation, Parameterschätzung oder Prozesskontrolle genutzt. Um diese verschieden Aufgabestellungen zu bewältigen, müssen die Gewichte der Neuronen entsprechend eingestellt werden. Diesen iterativen Prozeß nennt man Lernen oder auch Training. Obwohl es vielfältige Algorithmen gibt, um diese Gewichtsänderungen vorzunehmen, unterscheidet man prinzipiell überwachtes und unüberwachtes Lernen:

Überwachtes Lernen
Bei überwachtem Lernen, das auch Lernen mit Lehrer genannt wird, werden dem Netzwerk Eingabevektoren und die geforderten Ausgabevektoren (Zielvektoren) präsentiert. Das Netzwerk soll also durch den Traininsprozeß die Funktion $\mathbf{R^n} \rightarrow \mathbf{R^m}$ approximieren. Überwachtes Lernen wird verwendet, um KNN's darauf zu trainieren, Zusammenhänge zwischen komplizierten , mehrdimensionalen Eingabedaten zu erkennen und daraus Paramter zu schätzen.

Unüberwachtes Lernen
Bei unüberwachtem Lernen werden dem Netz nur Eingabevektoren und keine Zielvektoren präsentiert. Die Hauptaufgabe von neuronalen Netzen, die durch unüberwachtes Lernen trainiert werden, besteht in der Datenreduktion. Dies kann durch Reduzierung der Dimension der Eingabedaten oder durch Reduzierung der Eingabedaten erreicht werden. Im zweiten Fall teilt das Netz die Eingabedaten in verschiedene Klassen, auch Cluster genannt ein, so daß ähnliche Eingabedaten durch einen einzigen Repräsentanten dieser Klasse beschrieben werden. In diesem Fall sind die Gewichtsvektoren die Referenzvektoren , also die Prototypen eines Clusters.
Unüberwachtes Lernen wird zur Datenvorverarbeitung eingesetzt, um komplexe Merkmale in Klassen einzuteilen, während mit überwachtem Lernen trainierte Netze dann aus diesen reduzierten Daten weitere Merkmale der extrahierten Klassen berechnen können.

4.4 Der neuronale Netzwerktrigger auf Level 2

Wie bereits in Kapitel 2 beschrieben, läuft beim H1 Experiment auf der zweiten Triggerstufe ein neuronaler Netzwerktrigger. Mit diesem wird im Prinzip eine Mustererkennung im abstrakten Raum der Triggervariablen durchgeführt, wobei Physikereignisse und Untergrund voneinander getrennt werden sollen. Ein neuronales Netz wird dabei jeweils für eine bestimmte Klasse von Physikereignissen trainiert. KNN's eignen sich besonders für diese Aufgabe, da mit Lernalgorithmen Korrelationen in bekannten Datensätzen gefunden werden können, die sich dann auf unbekannte Daten verallgemeinern lassen. Außerdem ist aufgrund der inhärenten Parallelität von KNN's mit geeigneter Hardware (siehe 4.4.3) eine sehr hohe Rechenleistung und damit kurze Zeit für die Triggerentscheidung erreichbar. Auf L2 kommen im wesentlichen Feed-Forward-Netze zum Einsatz. Diese sollen deshalb im folgenden etwas genauer erläutert werden.

4.4.1 Feed-Forward-Netze

Bei Feed-Forward-Netzwerken werden die Neuronen in mehreren Lagen angeordnet, wobei jedes Neuron einer Schicht mit jedem Neuron einer darunterliegenden Schicht verbunden ist.

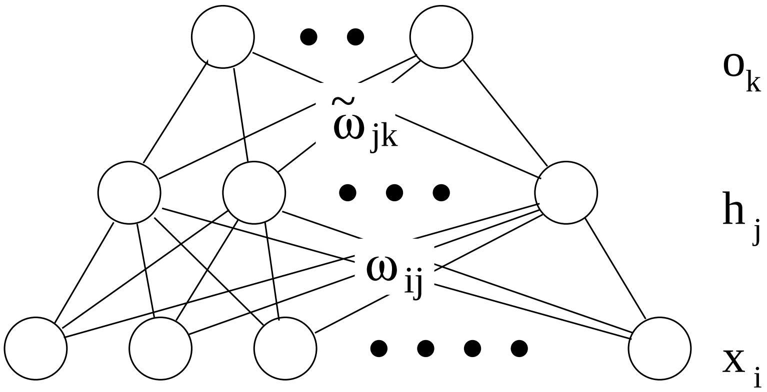


Abbildung 4.4: *Ein dreilagiges Feedforward-Netzwerk. $\vec{x}$ ist der Eingabevektor, $\vec{o}$ der Ausgabevektor und $\vec{h}$ der Ausgabevektor der versteckten Schicht.*

Abb. 4.4 zeigt ein dreilagiges Feedforward Netz, bestehend aus einer Eingabeschicht, einer versteckten Schicht (hidden layer) und einer Ausgabeschicht (output). Während der Eingabevektor $\vec{x}$ in der Eingabeschicht lediglich gespeichert wird, werden in der versteckten Schicht die Berechnungen ausgeführt. Die so erhaltenen Ausgabewerte werden an die Neuronen der tiefer liegenden Schicht, in diesem Fall bereits der Ausgabeschicht, übertragen. Der Informationsfluß findet nur in Vorwärtsrichtung statt, es gibt keine Rückkopplungen der tiefer liegenden Schichten zu höher liegenden, daher auch der Name Feed-Forward-Netz.
Im Prinzip können Feed-Forward-Netze beliebig viele versteckte Lagen besitzen. In der Praxis werden jedoch meist Netze mit einer bzw. zwei versteckten Lagen verwendet. Aus theoretischer Sicht ist selbst eine Beschränkung auf dreilagige Feed-Forward-Netze problemlos, da mit Hilfe eines allgemeinen Theorems von Kolmogorov [35] gezeigt werden kann, daß damit unter wenig einschränkenden Randbedingungen jede kontinuierliche m-dimensionale Funktion mit n Eingabe-, $2n+1$ versteckten und m Ausgabeknoten durch ein dreilagiges Netz approximiert werden kann. [36–38].
Bei geeigneter Wahl der Netzgröße und der einzelnen Gewichte sollten sich mit Feed-Forward-Netzwerken im Prinzip beliebige Aufgaben der Mustererkennung lösen lassen. In der Praxis ist der Wert dieser Aussage jedoch durch die vorher nicht konstruierbare Aktivierungsfunktion eingeschränkt, außerdem läßt sich die Netzarchitektur nicht losgelöst vom Lernalgorithmus betrachten.
Da der Lernalgorithmus eine entscheidende Rolle beim Ermitteln der geeigneten Gewichte spielt, wird im folgenden ein weitverbreiteter Algorithmus, der auch zum Training der FFN's für den L2 verwendet wird, genauer beschrieben.

4.4.2 Der Backpropagation-Algorithmus

Beim Training von Feed-Forward-Netzen handelt es sich um überwachtes Lernen, das heißt man präsentiert dem Netz ein Musterpaar ($\vec{x}$,$\vec{z}$), wobei $\vec{x}$ der Eingabevektor und $\vec{z}$ der geforderte Ausgabevektor ist. Der geforderte Ausgabevektor wird Zielvektor genannt. Um zu erreichen, daß der vom Netz berechnete Ausgabevektor nicht mehr oder nur gering vom Zielvektor abweicht, müssen die Gewichte des Netzwerkes verändert werden. Beim 1974 von Werbos [39] entwickelten Backpropagation-Lernalgorithmus BPA wird die Summe der quadratischen Abstände zwischen Ausgabevektoren $\vec{o_\mu}$ und Zielvektoren $\vec{z_\mu}$ berechnet und durch ein Gradientenabstiegsverfahren minimiert. Dieser Algorithmus fand erst breitere Anwendung nach seiner Wiederentdeckung 1986 von McClelland und Rumelhart [40]. Beim BPA werden folgende Schritte durchlaufen:

1. Präsentation eines Eingabemusters $\vec{x}$

2. Berechnung des Ausgabewerts der versteckten Schicht nach der Formel :

$$h_j = g\left(\sum_{i=1}^{n} \omega_{ij} x_i - t_j\right) . \tag{4.4}$$

   wobei g die Aktivierungsfunktion ist (siehe Glg. 4.3) und n die Dimension des Eingabevektors.

3. Weiterleitung des Ergebnisses der versteckten Schicht an die Ausgabeschicht.

4. Berechnung des Ausgabewertes der Ausgabeschicht nach der Formel

$$o_k = g\left(\sum_{j=1}^{m} \tilde{\omega}_{jk} h_j - t_k\right) . \tag{4.5}$$

   wobei als Eingabevektor der Ausgabevektor der versteckten Schicht verwendet wird und m die Anzahl der versteckten Knoten ist.

5. Vergleich des Ausgabevektors und des Zielvektors und Berechnung eines Ausgabefehlers

$$E(\omega, \tilde{\omega}) = \frac{1}{2} \sum_{\mu=1}^{N} |\vec{z}(\vec{x}_\mu) - \vec{o}(\vec{x}_\mu, \omega, \tilde{\omega})|^2, \tag{4.6}$$

   für jeden Knoten.

6. Änderung der Gewichte zwischen Ausgabe- und versteckter Schicht in Richtung des steilsten Abstiegs auf der Hyperfläche der Fehlerfunktion E (Gradientenabstiegsverfahren)

$$\Delta\tilde{\omega}_{jk} = -\epsilon \frac{\partial E}{\partial \tilde{\omega}_{jk}} + \alpha \Delta\tilde{\omega}_{jk}^{alt} = -\epsilon \delta_k h_j + \alpha \Delta\tilde{\omega}_{jk}^{alt} \tag{4.7}$$

$$\text{mit } \delta_k = (o_k - z_k) \cdot g'(\widetilde{\vec{\omega}_k} \cdot \vec{h}). \tag{4.8}$$

7. Schichtweise Rückverfolgung (Backpropagation) des Fehlergradienten auf den Berchnungen in der Ausgabeschicht aufbauend, so daß sich die Gewichtsänderungen zwischen Eingabe- und versteckter Schicht wie folgt berechnen

$$\Delta\omega_{ij} = -\epsilon\frac{\partial E}{\partial\omega_{ij}} + \alpha\Delta\omega_{ij}^{alt} = -\epsilon\delta_j x_i + \alpha\Delta\omega_{ij}^{alt} \tag{4.9}$$

$$\text{mit } \delta_j = \sum_{k=1}^{m} \tilde{\omega}_{jk}\delta_k g'(\vec{\omega}_j \cdot \vec{x}). \tag{4.10}$$

8. Präsentation des nächsten Eingabemusters

Bei Netzen mit mehr als drei Lagen werden die zusätzlichen versteckten Lagen analog behandelt. Der Parameter ϵ wird auch Lernparameter genannt und ist frei wählbar. Er beeinflußt die Lerngeschwindigkeit sowie die erreichbare Approximationsgenauigkeit eines Trainings und ist von der Problemstellung abhängig.
Der zweite freie Parameter $\alpha \in [0;1]$, Momentumparameter genannt, soll durch ein schwächer gewichtetes Hinzuaddieren der vorangegangenen Gewichtsänderung helfen, flache Regionen und lokale Minima in E schneller zu überwinden.

4.4.3 Hardware

Berechnet werden diese Netzwerke von einem CNAPS[1]/VME Parallelrechner, der in SIMD[2]-Technologie gebaut ist. Eine zentrale Befehlseinheit, genannt CSC Sequencer, kontrolliert die I/O-Datenbusse des Feldes von parallelen Prozessorknoten (PN[3]) und versorgt diese mit Instruktionen aus dem Programmspeicher. Jeder PN verfügt über einen lokalen Speicher von 4 KByte und stellt einen kompletten Rechner in Festkomma-Arithmetik dar. Eine CNAPS/VME Karte kann bis zu 8 Chips, mit bis zu jeweils 64 PNs besitzen. In L2 werden acht Karten mit je 64 PNs, eine mit 128 und eine mit 256 PNs eingesetzt. Die Taktfrequenz des CNAPS-Rechners beträgt 20 MHz, er wird bei H1 jedoch leicht übertaktet, mit der HeraClock-Rate von 20.8 MHz, betrieben.
Die Daten, die auf L2 zur Verfügung stehen und als Eingabegrößen für die Netzwerke dienen, werden zuvor mit einer speziellen Bus-Interface und Verarbeitungskarte (DDB)[4] aufbereitet und in Form von 8 bit Worten auf den CNAPS weitergeleitet.

Die DDB realisiert dabei Funktionen wie:

- Auswahl von high und low Byte aus einem 16 bit Wort
- Summenbildung von high und low Byte

[1] Connected Network of Adaptive ProcessorS
[2] Single Instruction, Multiple Data
[3] Processor Node
[4] Data Distribution Board

- Bestimmung der Position eines bestimmten Bits
- Anwendung einer Wertetabelle (Look Up Table)

Die Weiterleitung der Daten zum CNAPS erfolgt unmittelbar nach der Selektion, ungefähr 6-7 μs nach dem L1-Triggersignal. Insgesamt stehen dem CNAPS dann etwa 10 μs oder 210 Taktzyklen zur Verfügung um eine Triggerentscheidung zu treffen, da die L2-Gesamtentscheidungszeit 20 μs nicht überschreiten darf. Dies mag auf den ersten Blick sehr wenig erscheinen, der Aufbau eines Prozessorknotens des CNAPS ermöglicht es jedoch, daß Multiplikation, Addition sowie I/O und Speicheroperationen innerhalb eines Taktzyklusses durchgeführt werden können.
Ein Vektor-Matrix-Multiplikationsschritt bei der Berechnung eines Feed-Forward-Netzes nimmt also nur einen Taktzyklus in Anspruch. Durch die Parallelisierung ist der Rechenaufwand proportional zur Dimension des Eingabevektors und nicht zur Dimension der Matrix. Für ein 32x32x16x1 Netz werden also nur 81 Taktzyklen benötigt. Das macht 4.05 μs bei 20 MHz. Zusätzlich benötigen unvermeidliche Steuerbefehle ca. weitere 50 Zyklen, also ungefähr weitere 2.5 μs.
Eine schematische Übersicht der Hardware des L2NN ist in Abb.4.5 dargestellt.

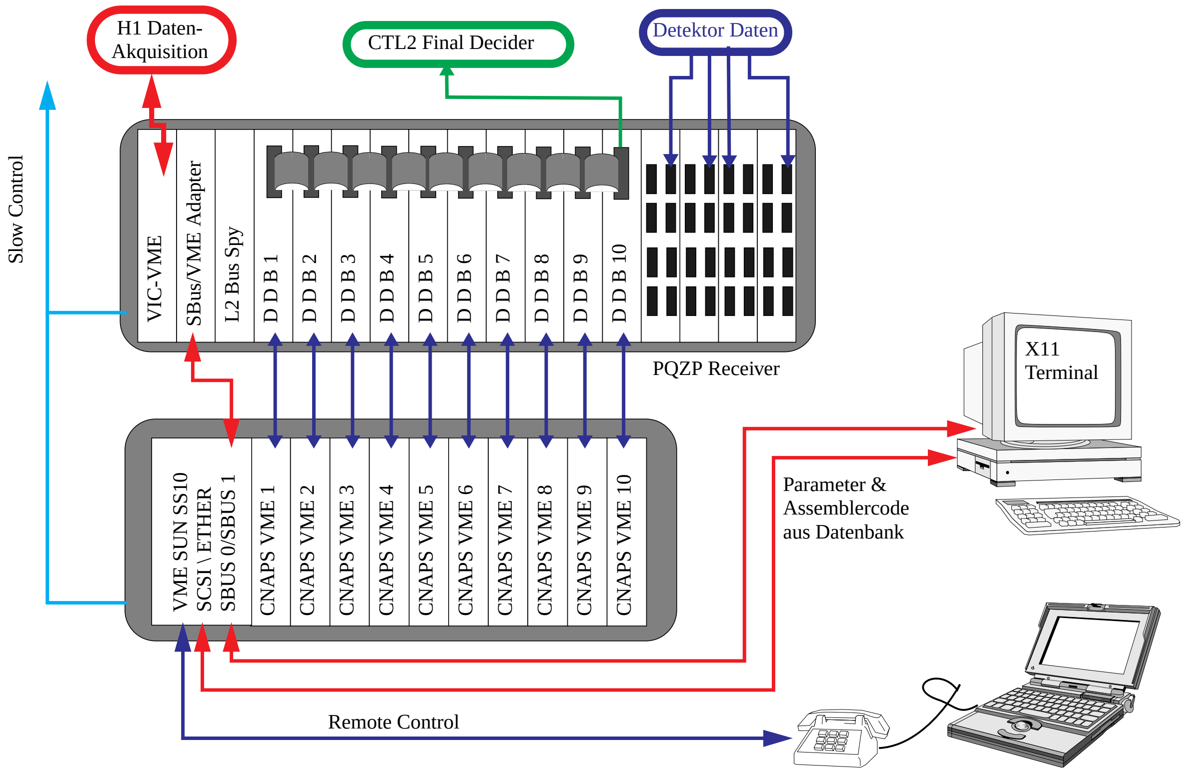


Abbildung 4.5: *Die Hardware des neuronalen L2 Triggers.*

Kapitel 5

Triggern von invarianten Massen aus Spurpunkten

In dieser Arbeit soll untersucht werden, ob es möglich ist, aus der Information der Spurpunkte der Driftkammern einen L2 Trigger zu verwirklichen, der auf invariante Massen triggern kann.

5.1 Motivation

Bei H1 vergehen 2.3 μs bis ein Ereignis von L1 bewertet worden ist. Daten der Kalorimeter und Spurkammersysteme können in diesem Zeitraum nur begrenzt ausgelesen werden. Die Daten, die für das Training eines neuronalen Netzes, das auf L2 laufen soll, zur Verfügung stehen, sind dadurch limitiert.
Mit Triggern, die auf Spurinformationen basieren, wird versucht die besonderen topologischen und kinematischen Merkmale bestimmter Ereignisse zu erkennen und zu triggern.

Auf Triggerebene ist mit den Spurkammern bisher jedoch nur eine rudimentäre Spuranalyse möglich. Im Moment basieren die bei H1 laufenden Spurtrigger auf z-Vertex Projektionen der Treffer in den Proportionalkammern (z-Vertex Trigger) oder auf r-ϕ-Projektionen der zentralen Driftkammern (DCRϕ - Trigger). Diese Trigger sind jedoch nicht selektiv genug und erreichen nicht die geforderte Datenreduktion. Ein Grund dafür ist, daß bis jetzt noch nicht die volle Spurinformation genutzt werden kann und so nicht der Grad an Redundanz erreicht wird, der nötig wäre um bessere Ergebnisse zu erzielen. Es existiert jedoch ein Verbesserungsvorschlag für die Datenausleseelektronik, die durch ein totzeitfreies Auslesesystem der Driftkammern einem L2 Trigger die gesamte Spurinformation der Driftkammern zur Verfügung stellen würde [41].

Es soll untersucht werden, wie diese gesamte Spurinformation für einen L2 Trigger genutzt werden kann.

5.2 Datengenerierung

Zunächst soll geprüft werden, ob ein neuronales Netz lernen kann, invariante Massen aus Detektortrefferpunkten zu rekonstruieren. Es soll aus Spurkammertreffern von zwei, bzw. drei Spuren die invariante Masse des Mutterteilchens rekonstruieren. In dieser Arbeit wird versucht, so D^*-Mesonen zu triggern. Um die Aufgabe für das neuronale Netz zu vereinfachen, wird nur der Zerfallskanal $D^{*+} \rightarrow \mathrm{D}^0\ \pi_s^+$; $\mathrm{D}^0 \rightarrow \mathrm{K}^-\ \pi^+$ betrachtet. Bei diesem Zerfall kann man das Netz darauf trainieren, aus den Spuren von K^- und π^+ die invariante Masse des D^0 zu rekonstruieren und so indirekt D^{*+}-Ereignisse triggern. Als zweite Möglichkeit präsentiert man drei Spuren (π_s^+, π^+, K^-) und trainiert auf die invariante Masse des D^{*+}-Mesons (siehe Abschnitt 6.3.6).
Normalerweise werden neuronale Netze, die als Trigger verwendet werden, durch ein sogenanntes Zweiklassentraining trainiert, wobei dem Netz Physikereignisse und Untergrundereignisse präsentiert werden. Die Netze erhalten in diesem Fall den Zielwert 0 für ein Untergrundereignis und 1 für ein Physikereignis. In dieser Arbeit wird eine andere Methode beim Training verwendet. Da man auf invariante Massen triggern will, läßt man ein fiktives Teilchen zerfallen. Zunächst wird der Zerfall in $K^-\ \pi^+$ betrachtet. In den Untersuchungen variiert die Ruhemasse dieses fiktiven Teilchens von 1 GeV/c^2 bis 3 GeV/c^2. Außerdem wird der Anfangsimpuls des fiktiven Teilchen gleichverteilt zwischen 2 und 5 Gev/c gewürfelt. Als Zielwert wird dem Netz die invariante Masse des fiktiven Teilchens mitgegeben. Das neuronale Netz soll lernen, diese Masse zu schätzen.
Die Kinematik für den Zerfall des fiktiven Teilchens wird mit einem im Rahmen dieser Arbeit erstellten Zerfallsgenerator generiert. Aus den kinematischen Daten werden dann die Trefferpunkte im Detektor berechnet. Es folgt eine kurze Beschreibung der verwendeten Programme:

5.2.1 Beschreibung der Simulationsprogramme

Der Zerfallsgenerator
Entscheidend für die Kinematik des Zerfalls sind die Masse und der Impuls des fiktiven Teilchens, die in den oben beschriebenen Wertebereichen gewürfelt werden. Die Winkel θ und ϕ des fiktiven Teilchens im Laborsystem werden gleichverteilt zwischen 0^0 und 180^0 bzw. 0^0 und 360^0 gewählt.
Für den Zerfall des fiktiven Teilchens in $\mathrm{D}^0\ \pi_s^+$; $\mathrm{D}^0 \rightarrow \mathrm{K}^-\ \pi^+$ werden zuerst die Impulse der Einzelzerfälle im Ruhesystem des jeweiligen zerfallenden Teilchen berechnet. Dabei wird in den Ruhesystemen ϕ zwischen 0^0 und 360^0 und θ zwischen -1 und 1 gleichverteilt angenommen. Die erhaltenen Impulse werden dann durch Lorentz Transformation ins Laborsystem umgewandelt. Für den zunächst betrachteten Zerfall des fiktiven Teilchens in $\mathrm{K}^-\ \pi^+$ wird einfach der Zerfall in $\mathrm{D}^0\ \pi^+$ weggelassen.

Aus den kinematischen Daten müssen anschließend die Spurkammertreffer simuliert werden. Da es sich um eine Grundlagenstudie handelt, und zuerst der grundlegende Einfluß von Parametern wie r-ϕ-Auflösung, z-Auflösung, Anzahl der Detektorlagen und Hebelarm untersucht werden soll, wurde zur Simulation nicht bereits existierende Software verwendet, sondern eine einfache Detektorsimulation für diese Arbeit entwickelt, die im folgenden kurz beschrieben wird:

Die Detektorsimulation

Das Programm liest zuerst die Teilcheneigenschaften wie 4-er Impuls, Entstehungsort (z.B. bei sekundären Vertices), Masse und Ladung ein. Im Magnetfeld, das den Detektor durchdringt, bewegen sich die Teilchen auf Helixbahnen. In der Projektion auf die r-ϕ-Ebene erhält man als Teilchenspuren Kreise. Für das gewählte Magnetfeld werden zunächst die Radien dieser Teilchenkreise, sowie die Mittelpunkte dieser Kreise in der r-ϕ-Ebene berechnet (dieser ist abhängig von Ladung, Impuls und Entstehungsort). Die Anzahl der Detektorlagen, sowie deren Position, können im Programm beliebig gewählt werden. Es werden zylinderförmige Detektorlagen angenommen. Die Detektortrefferpunkte in der r-ϕ-Ebene erhält man dann durch Schneiden zweier Kreise. Der Detektorkreis hat dabei seinen Mittelpunkt bei 0 und als Radius den jeweiligen Abstand der Detektorlage von der Strahlachse. Der Teilchenkreis hat den vom Magnetfeld und den Teilcheneigenschaften abhängigen Mittelpunkt und Radius. Aus den Schnittpunkten der Kreise erhält man die x-y-Koordinaten des Detektortrefferpunktes. Aus der Flugzeit des Teilchens bis zu diesem Durchstoßpunkt und der z-Komponente des Teilchenimpulses wird dann die dazugehörige z-Koordinate des Trefferpunktes berechnet. Auf diese exakt berechneten Treffer werden für x und y gaußsche Fehler mit der Standardabweichung $\sigma_{r\phi}$ und für z mit σ_z addiert, um die Meßungenauigkeit der Driftkammern zu simulieren.

Für die Simulation der Driftkammern CJC1 und CJC2 beim H1-Experiment wurde das Magnetfeld B = 1.15 T, sowie $\sigma_{r\phi} = 200$ μm und $\sigma_z = 2.5$ cm gewählt. Die Driftkammern besitzen 56 Signaldrähte, von denen der äußerste 85 cm radial vom Strahlrohr entfernt ist (siehe Abschnitt 3.2.1), wodurch die Anzahl und der Ort der Detektorlagen für die Untersuchungen im nächsten Kapitel begrenzt wird.

5.2.2 Erzeugte Daten

In Abb. 5.1 sind einige der kinematischen Größen und Winkel für 20000 generierte Ereignisse dargestellt. Je größer die Masse des fiktiven Teilchens ist, desto mehr Impuls erhalten das π und das K beim Zerfall. Dies erklärt die Impulswerte beider Teilchen, die größer als 5 GeV sind. Bei den θ-Winkel Verteilungen sieht man die mit Monte Carlo generierte Gleichverteilung von $\cos(\theta)$.

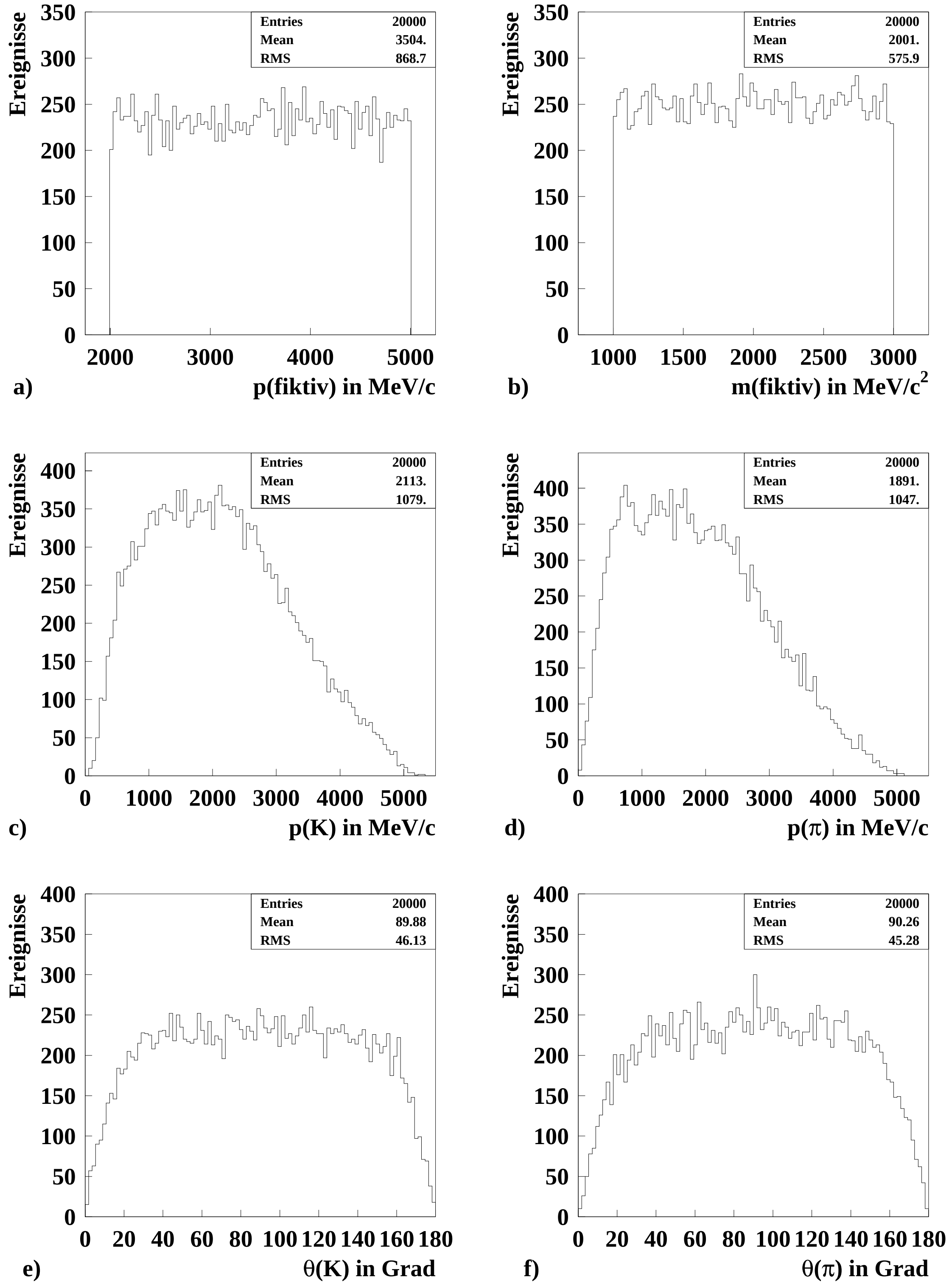


Abbildung 5.1: *Die kinematischen Größen a) Impuls des fiktiven Teilchens b) Masse des fiktiven Teilchens c) Impulsverteilung des Kaons d) Impulsverteilung des Pions c) Verteilung des Winkels θ des Kaons d) Verteilung des Winkels θ des Pions*

Kapitel 6

Ergebnisse des Netztrainings

In diesem Kapitel wird zuerst das Training der Netze durch den in 4.4.2 vorgestellten Backpropagation-Algorithmus (BPA) untersucht. Die mittels BPA erzielten Ergebnisse hängen von der Netzwerkarchitektur und den gewählten Trainingsparametern ab. Das Training wurde auf dem CNAPS-Parallelcomputer durchgeführt, der mit seiner hohen Rechenleistung eine umfangreiche Parameterstudie ermöglichte. Ziel dieser Studien ist es, einen Parametersatz zu finden, der für verschiedene Problemstellungen möglichst stabil zu guten Ergebnissen führt, da der Aufwand alle Parameter für jede Trainingsaufgabe neu zu optimieren zu zeitaufwendig ist.

6.1 Trainings- und Testdaten für den CNAPS

Mit den in Kapitel 5 beschriebenen Programmen werden die Koordinaten der Detektortrefferpunkte für den Zerfall eines fiktiven Teilchens in K^- und π^+ mit der gewählten Auflösung generiert. Diese Daten liegen zunächst in double-Format vor und können so nicht für das Training auf dem CNAPS verwendet werden, da dieser nur in folgenden zwei Softwaremodi betrieben werden kann:

- Der 8/16-Bit Modus (low precision)
  Es werden 8 Bit Daten und 16 Bit Gewichte verwendet. Die Gewichtsänderungen während des Trainings $\omega_{ij}, \tilde{\omega}_{jk}$ werden mit 16 Bit Genauigkeit berechnet.

- Der 16/32-Bit Modus (high precision)
  Es werden 16 Bit Daten und 32 Bit Gewichte und Gewichtsänderungen verwendet. Die Variation einer Eingabegröße ist also nicht so stark eingeengt wie beim 8/16-Bit Modus. Ganz generell eignet sich der 16/32-Bit Modus besser für das Training, da durch die Fähigkeit kleinere Gewichtsänderungen durchführen zu können , bessere Trainingsresultate erzielt werden.

Um ein Training im 16/32-Bit Modus durchzuführen, müssen die erhaltenen Daten also zunächst in 16-Bit Format umgewandelt werden. Aus den x und y-Koordinaten

werden zuerst Winkel ϕ berechnet, die so den Trefferpunkt in der r-ϕ-Ebene kodieren. Dies hat den Vorteil, daß nur zwei Eingabeknoten pro Trefferpunkt (Winkel ϕ + z-Koordinate) benötigt werden und damit durch einen einfachen Vorverarbeitungsschritt Rechenzeit gespart werden kann. Der erhaltene Winkel ϕ wird dann so umgewandelt, daß das Intervall $[0^0,360^0]$ auf [-32767,+32767] abgebildet wird. In z-Richtung haben die Driftkammern eine Ausdehnung von -1.1 m bis 1.1 m vom nominellen Wechselwirkungspunkt aus gesehen. Für die z-Koordinate bildet man also [-1.1,1.1] auf [-32767,+32767] ab. Die Masse des fiktiven Teilchens wird so transformiert, daß 1 GeV/c^2 -32767 und 3 GeV/c^2 +32767 entspricht. Sie wird an jedes Trainingsmuster als Zielwert angehängt.
In Abb. 6.1 sieht man den D^{*+}-Zerfall skizziert. Die Trefferpunkte auf vier Detektorlagen in der r-ϕ-Projektion sind für das π^+, welches aus dem D^0 Zerfall entsteht, eingezeichnet. Für die Untersuchung sind zunächst nur die Spuren des K^-

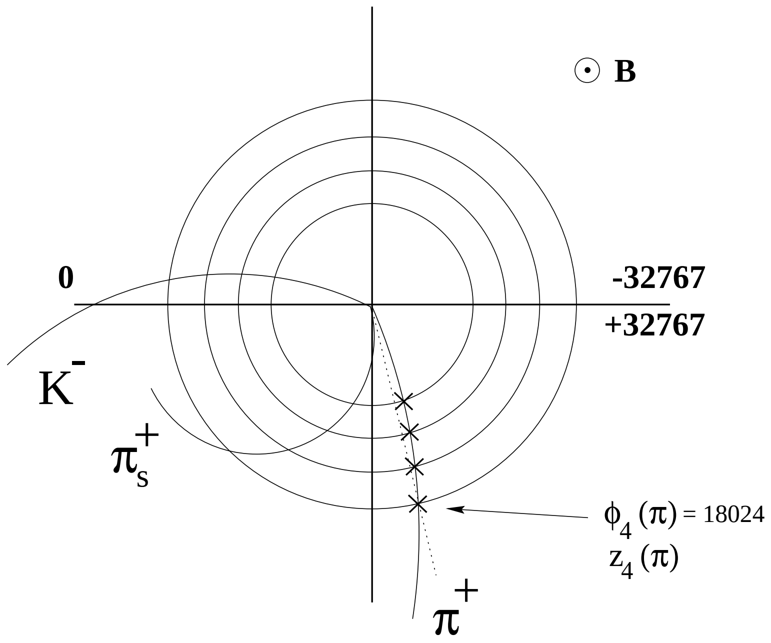


Abbildung 6.1: *Teilchenspuren der Zerfallsprodukte eines D^{*+}-Zerfalls in r-ϕ-Projektion. Das K^- und das π^+ entstehen am sekundären Vertex, der jedoch nicht eingezeichnet ist.*

und des π^+ von Interesse. Als Bedingung für die generierten Daten wurde gefordert, daß jede Spur jede Detektorlage schneidet. Dadurch werden Ereignisse, bei denen eines der Teilchen einen kleineren Teilchenradius als 42.5 cm[1] und damit zu geringen Transversalimpuls besitzt, nicht zum Training verwendet. Auch der θ - Bereich und damit der Longitudinalimpuls wird durch diese Forderung begrenzt, da die Teilchen bis zum Durchstoßen der Detektorlage bei 85 cm nicht weiter als $\pm$ 1.1 m in z geflogen sein dürfen. So werden nur Ereignisse zum Training angeboten für die ungefähr $40^0 < \theta < 140^0$ gilt. In Abb. 6.2 sieht man diese Einschränkung der Trainingsdaten am Beispiel des Pions.

[1] Dies entspricht der Hälfte des äußersten Radius der CJC2 (r = 85 cm)

Für ein Training auf dem CNAPS muß immer ein Trainingsdatensatz und ein

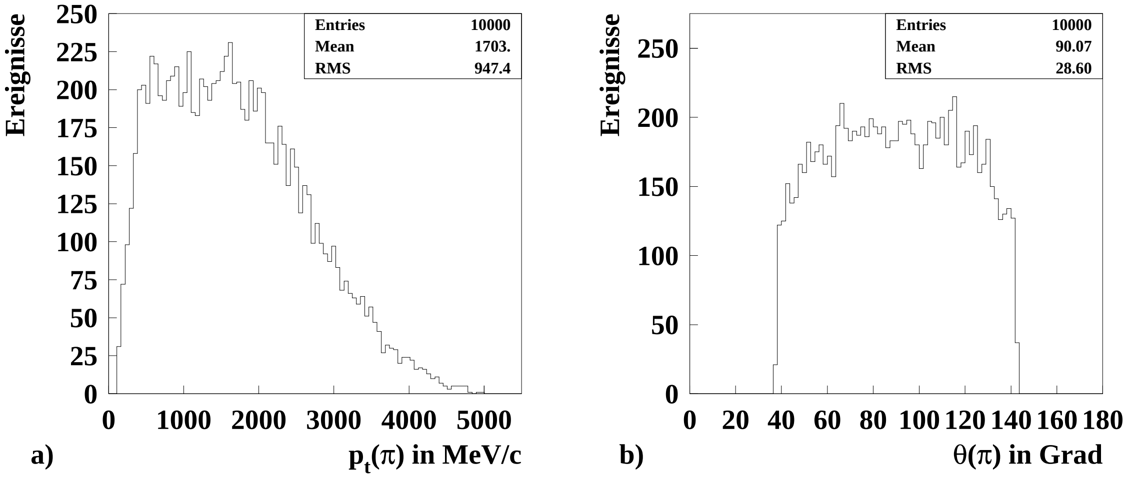


Abbildung 6.2: *a) Transversalimpuls des Pions in den Trainingsdaten b) Verteilung des θ-Winkels des Pions in den Trainingsdaten.*

Testdatensatz erstellt werden. Zur Erstellung dieser Daten wird in den Simulationsprogrammen jeweils eine andere Initialisierung des Monte-Carlo Generators gewählt. Während des Trainings werden dem Netz pro Trainingsschritt, der auch Epoche genannt wird, jeweils alle Trainingsdaten präsentiert. Danach wird der mittlere quadratische Fehler MSE[2] (siehe Glg. 6.1) sowohl für Trainings- als auch Testdaten berechnet, wobei die Ziel- und Ausgabewerte vorher auf das Intervall [0,1] abgebildet werden. Der Verlauf des MSE für die Trainings- bzw. Testdaten mit zunehmender Epochenzahl wird Lernkurve genannt.

$$MSE = \frac{1}{N} \sum_{\mu=1}^{N} ||o_\mu - z_\mu||^2 \tag{6.1}$$

mit: o : *Ausgabewert des Netzes,*
z : *Zielwert,*
N : *Anzahl der Muster*

Für die Trainingsdaten wird die Lernkurve im Normalfall ständig fallen, da das Netz mit zunehmender Epochenzahl dazu tendiert, die Trainingsdaten auswendig zu lernen. Dagegen dient der MSE auf dem Testdatensatz dazu, eine Aussage über die Qualität des trainierten Netzes zu treffen. Ab einer bestimmten Epoche sinkt dieser nicht mehr bzw. beginnt sogar zu steigen. Die Gewichte des Netzes sind dann zu gut an die Trainingsdaten angepaßt. Die Generalisierungsfähigkeit nimmt ab, das heißt der mittlere Fehler des Testdatensatzes nimmt wieder zu. Dieser Effekt wird auch als „Overtraining“ bezeichnet. Ein Training läuft deshalb immer zuerst die volle Epochenanzahl, danach wird nocheinmal bis zur Epoche, an der der MSE

[2] mean squared error

auf den Testdaten am niedrigsten war, trainiert. Die Gewichte bei dieser Epoche werden als die besten Gewichte „best.wts“ abgespeichert. Zur Verdeutlichung siehe Abb. 6.3 a). Für eine spätere Bewertung eines anderen Datensatzes müssen dann lediglich diese Gewichte des austrainierten Netzes in den CNAPS geladen werden.

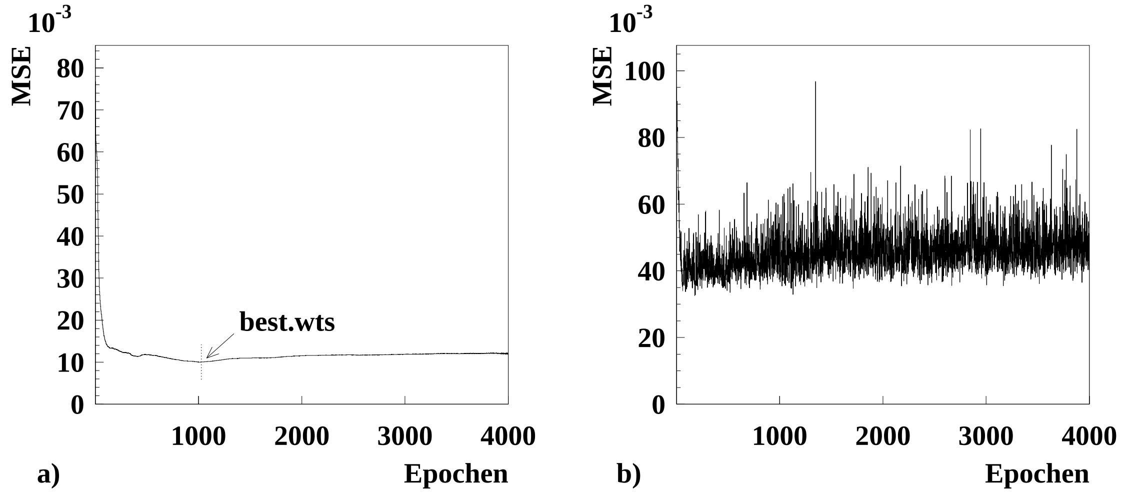


Abbildung 6.3: *Beispiele für Lernkurven a) Verdeutlichung der besten Gewichte b) Lernkurve für ein fehlgeschlagenes Training mit zu großem Lernparameter*

Der niedrigste Wert, den der MSE auf den Testdaten erreicht, aber auch der Verlauf der Lernkurve gibt Aufschluß darüber, ob ein Training erfolgreich verlaufen ist. In Abb. 6.3 b) sieht man z.B. die Lernkurve für ein Training bei dem der Lernparameter zu groß gewählt wurde. Der MSE bleibt nicht nur auf einem relativ hohen Wert hängen, das Ergebnis ist auch noch sehr instabil. Die Lernkurve pendelt so stark hin und her, weil mit dem großen Lernparameter kein Minimum der Hyperfläche der Fehlerfunktion erreicht werden kann. Bildlich gesprochen, springt der Algorithmus zwischen den Wänden einer Schlucht hin und her, ohne zum Boden zu gelangen.

6.2 Parameterstudien

Um die Parameter des Netzwerkes zu bestimmen wurde für die Berechnung der Eingabedaten als Detektorauflösung $\sigma_{r\phi} = 200\ \mu$m und $\sigma_z = 2.5$ cm verwendet, was der Driftkammerauflösung entspricht. Die Studien wurden mit acht Detektorlagen in der CJC1 und CJC2 durchgeführt, das heißt es wurden pro Teilchen acht Trefferpunkte präsentiert, also acht Winkel ϕ und acht z-Koordinaten. Insgesamt besitzt das Netz also 32 Eingabeknoten. Für die Anfangsstudien werden dem Netz die Trefferpunkte der einzelnen Teilchen immer in der gleichen Reihenfolge präsentiert, das heißt die ersten 16 Eingabeneuronen erhalten stets Eingaben des π und die zweiten 16 immer die Trefferpunkte des K. In Abschnitt 6.3.5 werden andere Eingabemöglichkeiten untersucht, die keine derartige Datenvorverarbeitung voraussetzen. Mehrere Stichproben mit unterschiedlichen Lernparametern ergaben, daß die Ergebnisse der Parameter-

studien bei diesem Problem auch für unterschiedliche Datenvorverarbeitung gültig bleiben.
Für sämtliche Untersuchungen wurden jeweils 10000 Trainingsdaten und 2000 Testdaten verwendet. Die Zahl der Testdaten wurde geringer gewählt um Rechenzeit zu sparen, außerdem ist sie vollkommen ausreichend um das Netztraining zu bewerten. Für die Durchführung der Studien wurde das Programm bpexp2 verwendet, das eine bequeme Steuerung des Trainings über „steering-cards“ ermöglicht.

6.2.1 Vergleich von drei- und vierlagigen Netzen

Das Training auf dem CNAPS ermöglicht, es Netze mit einer oder mit zwei versteckten Lagen zu trainieren. Die Parameterstudie wurde daher sowohl für dreilagige- als auch vierlagige Netze durchgeführt.

Dreilagige Netze
Bei den Parameterstudien dreilagiger Netze wurde sowohl der Lernparameter über einen weiten Bereich variiert, als auch die Knotenzahl der versteckten Lage. Die Knotenanzahl wurde dabei beginnend von acht Knoten jeweils verdoppelt, der Lernparameter beginnend von 0.00031 jeweils vervierfacht. In Abb. 6.4 ist der mittlere quadratische Fehler auf den Testdaten für die einzelnen Parameterkombinationen aufgetragen.

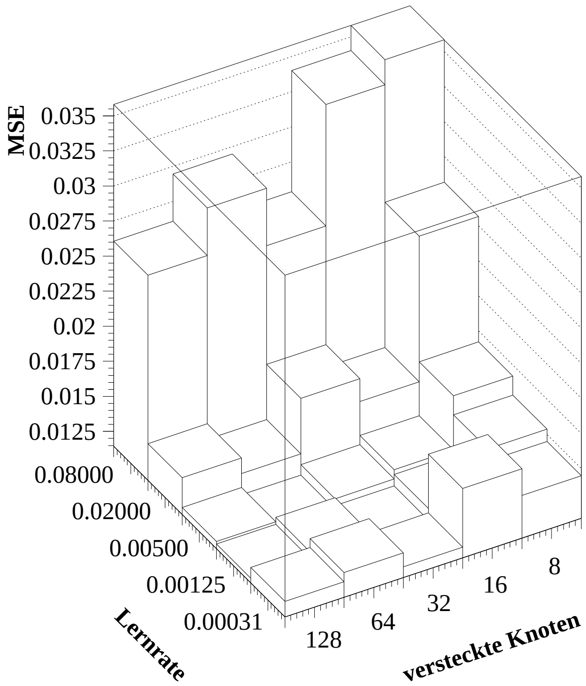


Abbildung 6.4: *Parameterstudie der Knotenanzahl und des Lernparameters für ein dreilagiges Netzwerk.*

Trainiert wurde jeweils bis zu einer Maximalzahl von 4000 Epochen, das heißt dem Netz werden sämtliche Trainingsdaten 4000 mal präsentiert. Man sieht, daß die Netze ab einer Größe von 16 versteckten Knoten in der versteckten Lage sehr stabile Ergebnisse über einen weiten Bereich des Lernparameters liefern. Lediglich der Lernparameter 0.08 ist zu groß gewählt. Das globale Minimum der Fehlerfunktion kann mit diesem Lernparameter nicht gefunden werden (siehe auch Abb. 6.3 b)).

Vierlagige Netze

Die Parameterstudie für vierlagige Netze wurde mit fünf verschiedenen Kombinationen von versteckten Knoten in der ersten und zweiten versteckten Lage durchgeführt, wobei immer das Verhältnis 2:1 gewählt wurde. Natürlich wären beliebige Kombinationen möglich, eine derartige Untersuchung verbietet sich aber aus Zeitgründen. Außerdem soll ausschließlich die Größenordnung des besten und stabilsten Netzes ermittelt werden. Als maximale Epochenzahl wurde wieder 4000 gewählt.

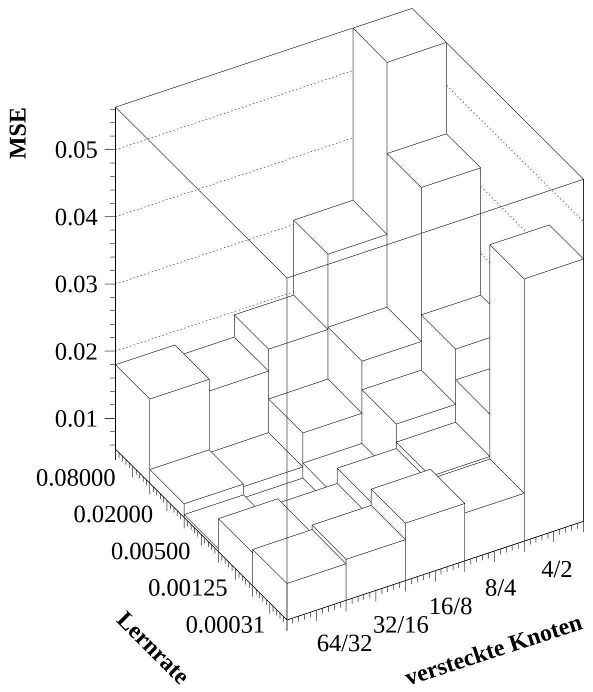


Abbildung 6.5: *Parameterstudie der Knotenanzahl und des Lernparameters für ein vierlagiges Netzwerk.*

Wie man in Abb. 6.5 sieht, sind die besten Ergebnisse der vierlagigen Netze (MSE $\approx$ 0.005) mehr als doppelt so gut wie die der dreilagigen Netze (MSE $\approx$ 0.0100). Ab der Knotenkombination 16/8 für die versteckten Lagen liefern die Netze wieder ungefähr dieselben Ergebnisse, diese sind allerdings nicht mehr ganz so stabil gegenüber einer Variation des Lernparameters. Als bester Lernparameter stellte sich in dieser Studie $\epsilon = 0.005$ heraus.

6.2.2 Einfluß weiterer Parameter

Die BPA-Modifikation, die auf dem CNAPS implementiert ist, läßt eine Einstellung weiterer Parameter zu.

Startgewichte
Bevor ein Training gestartet werden kann, müssen die Gewichte initialisiert werden. Dabei können nicht alle Gewichte auf identische Werte gesetzt werden, da der BPA diese Symmetrie unter den Gewichten nicht brechen kann. Beim zum Training verwendeten Programm bpexp2 werden die Gewichte ω_{ij} der einzelnen Lagen deshalb pseudozufällig aber reproduzierbar nach Glg. 6.2 generiert, wobei n die Anzahl der Knoten in der jeweiligen Schicht ist.

$$\omega_{ij} \in [-\epsilon, \epsilon] \qquad mit \qquad \epsilon = \frac{3}{\sqrt{n}} \tag{6.2}$$

Untersuchungen von L.F.A. Wessels und E. Barnard zeigten, daß diese Wahl der Gewichte vorteilhaft ist, um ungünstige lokale Minima zu vermeiden [42].
Es wurde trotzdem untersucht ob die Trainingsergebnisse beim betrachteten Problem von einer unterschiedlichen Anfangsinitialisierung der Gewichte abhängig sind. Diese legt nämlich für das Gradientenabstiegsverfahren einen anderen Startpunkt auf der Hyperfläche der Fehlerfunktion fest. Unterschiedliche Startpunkte können dann zu verschiedenen lokalen Minima führen, falls das globale Minimum der Fehlerfunktion nicht immer gefunden wird. Für die Untersuchung wurde das Training für das 32/32/16/1 Netzwerk mit fünf verschiedenen Anfangsinitialisierungen der Gewichte und drei Lernparametern durchgeführt.
In Abb. 6.6 sieht man, daß eine Variation der Startgewichte Einfluß auf die Trainingsergebnisse hat. Besonders bei dem kleinen Lernparameter 0.00125 ist der MSE beim besten Training (Initialisierung 2) mit 0.00494 mehr als doppelt so gut als beim schlechtesten Training (Initialisierung 4), das den MSE = 0.01037 liefert. Bei den größeren Lernparametern ist diese Variation nicht so drastisch, aber auch hier tritt ein bemerkbarer Effekt auf. Die wesentlich größeren Schwankungen beim Lernparameter 0.00125 lassen sich dadurch erklären, das sich ungünstige lokale Minima mit einem sehr kleinen Lernparameter schlecht bzw. nicht überwinden lassen. Anschaulich gesehen, bleibt der BPA beim Gradientenabstieg dann in ungünstigen „Tälern“ stecken, die er mit dem kleinen Lernparameter nicht wieder verlassen kann.

Lernintervall
Dieser Parameter bestimmt wann eine Gewichtsänderung durchgeführt wird. Bei einer Gewichtsänderung nach Präsentation aller Trainingsdaten spricht man von „batch learning“. Eine Gewichtsänderung nach jedem einzelnen Muster, was einem Lernintervall von 1 entspricht, wird als „online learning“ bezeichnet. Zwischen diesen Extremwerten ist jeder Wert einstellbar. In früheren Arbeiten [43, 44] wurde gezeigt, daß „online learning“ die besten Ergebnisse liefert. Das Lernintervall wurde deshalb für alle Studien auf 1 festgesetzt.

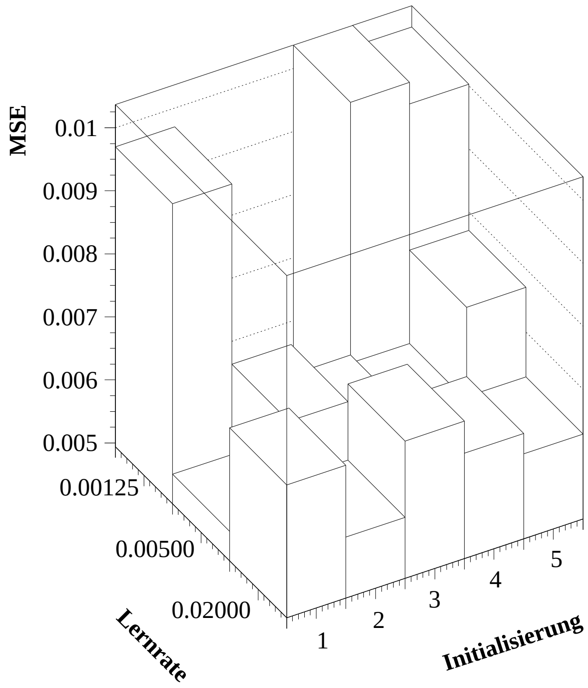


Abbildung 6.6: *Einfluß der unterschiedlichen Initialisierung der Startgewichte.*

Momentumfaktor

Wie in Glg. 4.7 bzw. Glg. 4.9 bereits gezeigt, können bei den Trainingsschritten auch die Gewichtsänderungen $\Delta\tilde{\omega}_{jk}^{alt}$ und $\Delta\omega_{ij}^{alt}$ der letzten Iteration, gewichtet mit dem Momentumfaktor α berücksichtigt werden. So soll ein stabileres Lernverhalten erreicht werden und Plateaus in der Fehlerlandschaft schneller überwunden werden. In vorher durchgeführten Studien [44] wurde jedoch keine Verbesserung durch einen von null verschiedenen Momentumfaktor festgestellt. Da ein Training auf dem CNAPS bei Verwendung eines Momentumfaktors etwa die doppelte Rechenzeit benötigt, wurde auf eine weitere Parameterstudie des Momentumfaktors aus Zeitgründen verzichtet und für alle Trainings $\alpha = 0$ gesetzt.

Hidden Learning Scale

Normalerweise wird beim BPA für Gewichtsänderungen in der Ausgabeschicht und den versteckten Schichten derselbe Lernparameter ϵ verwendet. Bei der verwendeten Modifikation des BPA ist es möglich, die Lernrate der versteckten Schicht um einen Faktor β größer zu wählen als die Lernrate in der Ausgabeschicht. Der Faktor β wird als Hidden Learning Scale bezeichnet und soll das Trainingsverhalten von Netzen mit nur einem Ausgabeknoten verbessern. Eine Wahl dieses Parameters größer als 1 bietet sich daher in dem untersuchten Fall an. In dieser Arbeit wurde für alle Studien, der von Adaptive vorgeschlagene default-Wert von $\beta = 4.0$ benutzt [45] der in einer weiteren Studie für gut befunden wurde [46].

6.2.3 Ergebnis der Parameterstudien

Aus der Parameterstudie werden folgende Schlüsse für die weiteren Untersuchungen gezogen.

- Bei dem untersuchten Problem und der gewählten Epochenzahl erweisen sich vierlagige Netze als weitaus leistungsfähiger als dreilagige Netze. Man könnte vermuten, daß sich mit steigender Epochenzahl die Ergebnisse wieder angleichen. Jedoch ergab auch ein Training des dreilagigen Netzes mit 32 versteckten Knoten ($\epsilon = 0.005$) bis zu einem Maximum von 20000 Epochen mit einem MSE von 0.01126 nur ein unwesentlich besseres Ergebnis als das mit 4000 Epochen (MSE = 0.01220). Für die weiteren Untersuchungen werden deshalb nur noch vierlagige Netze verwendet.

- Aus der Lernparameterstudie ergibt sich, daß das Ergebnis über den weiten Bereich von $\epsilon = 0.00125$ bis $\epsilon = 0.02$ ziemlich stabil ist. Beim kleinsten Parameter stellte sich jedoch eine größere Empfindlichkeit gegenüber der Variation der Startgewichte heraus. Für alle weiteren Trainings wird der Lernparameter $\epsilon = 0.005$ verwendet.

- Weiterhin werden die Studien ohne Momentumfaktor $\alpha = 0$ und mit dem Lernintervall 1 durchgeführt. Der Hidden Learning Scale wird auf $\beta = 4.0$ gesetzt.

6.3 Variation der Trainingsdaten

Im folgenden Abschnitt soll untersucht werden, welche Auswirkung eine Variation der Parameter der Simulationsprogramme, wie z.B. Detektorauflösung, Detektorlagen und verwendete Driftkammern oder Massen- und Impulsbereich des fiktiven Teilchens auf die Trainingsergebnisse hat. Weiterhin soll gezeigt werden, wie sich eine unterschiedliche Präsentation der Eingabedaten, die weniger Datenvorverarbeitung voraussetzt, auswirkt und welches Ergebnis ein Training mit drei Teilchenspuren erzielt.
Als Bewertungsgrundlage für die unterschiedlich trainierten Netze wird nun der RMS-Fehler[3] (siehe Gleichung 6.3) auf den Testdaten berechnet.

$$RMS = \sqrt{\frac{\sum_{i=1}^{N}(\bar{x} - x_i)^2}{N-1}} \tag{6.3}$$

$$\begin{array}{lcl} \textit{mit: } \bar{x} & : & \frac{1}{N}\sum_{\mu=1}^{N}(o_\mu - z_\mu), \\ x_i & : & (o_i - z_i) \end{array}$$

[3] root mean squared error

Die Zielwerte z und die Ausgabewerte o transformiert man dazu vom 16-Bit Wertebereich wieder auf den double-Wert der Masse, wobei die Ausgabewerte mit den Gewichten des austrainierten Netzes best.wts berechnet werden. Der RMS Fehler wird in MeV/c^2 angegeben und zeigt direkt, wie gut die Massenschätzung des neuronalen Netzes ist. In Abb. 6.7 sieht man für 2000 Testdaten die Netzausgabe gegen den Zielwert aufgetragen. Man sieht die Abweichungen vom optimalen Fall - bei dem sich eine Gerade ergeben würde -, da dann jedes Eingabemuster auf den geforderten Zielwert abgebildet wird. Die Tendenz, daß kleine Massen zu groß und große Massen zu klein geschätzt werden, die man in der Abbildung an den leichten Knicken im oberen und unteren Wertebereich erkennt, ist dabei eine Folge der zum Berechnen des Ausgabewertes verwendeten Aktivierungsfunktion (siehe auch Abb. 4.3).

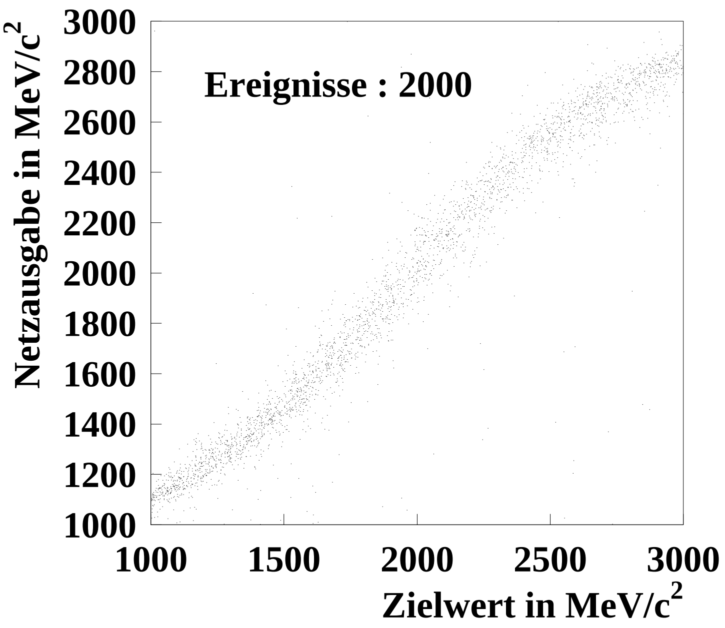


Abbildung 6.7: *Netzausgabe aufgetragen gegen den geforderten Zielwert der invarianten Masse.*

Da bereits die Variation der Startgewichte einen leichten Einfluß auf das Endergebnis eines Trainings hat, wurden für die folgenden Untersuchungen jeweils drei Trainings mit verschiedenen Trainings- und Testdatensätzen durchgeführt. So erhält man eine erste Abschätzung für den Fehler auf den Ergebnissen des Netztrainings, der in den folgenden Abbildungen durch die Fehlerbalken dargestellt wird. Eine größere Anzahl an durchgeführten Trainings wäre aus statistischen Gründen wünschenswert gewesen, war aber aus Zeitgründen unmöglich.

6.3.1 Anzahl der Detektorlagen

Zur Rekonstruktion von invarianten Massen werden die Teilchenimpulse sowie die Öffnungswinkel zwischen den Teilchenspuren benötigt. In Gleichung 6.4 sieht man wie die invariante Masse für einen Zwei-Teilchen-Zerfall berechnet wird.

$$m_{inv} = \sqrt{m_1^2 + m_2^2 + 2E_1E_2 + p_1p_2 \cos\alpha} \tag{6.4}$$

mit: m_1, m_2 : *Massen der Zerfallsteilchen,*
E_1, E_2 : *Energien der Zerfallsteilchen,*
p_1, p_2 : *Impulse der Zerfallsteilchen,*
α : *Winkel zwischen den beiden Teilchen,*

Für beide Teilchen müssen Massenhypothesen angenommen werden, da aus der Spurrekonstruktion keine Teilchenidentifikation möglich ist.
Um eine Teilchenspur zu rekonstruieren, werden zumindest drei Trefferpunkte im Detektor benötigt. In der Rekonstruktion des Transversalimpulses bzw. der Krümmung nimmt die Genauigkeit dabei mit der Anzahl der verwendeten Trefferpunkte zu, da die Meßfehler der einzelnen Punkte weniger stark ins Gewicht fallen.
Eine Abschätzung für den Fehler der Spurkrümmung nach Glückstern [47] ist:

$$\sigma_\kappa = \frac{\epsilon}{L^2}\sqrt{\frac{720}{N+4}} \tag{6.5}$$

mit: ϵ : *Genauigkeit der Einzelmessung in der Ebene, senkrecht zum Magnetfeld,*
L : *Spurlänge in dieser Ebene,*
N : *Anzahl der Punkte, an denen diese Spur gemessen wurde,*

Gleichung 6.5 gilt dabei für Spuren, die an N äquidistanten Punkten gemessen wurden, was bei der CJC in der Regel nicht der Fall ist und ist streng genommen nur für großes N exakt. Die Fehler werden dennoch näherungsweise beschrieben. Es wurde untersucht, ob die Anzahl der Detektorlagen und damit die Anzahl der Punkte, an denen eine Spur gemessen wird, auch bei neuronalen Netzen eine Auswirkung auf die Genauigkeit der Schätzung der invarianten Masse hat. Die Studie wurde unter Verwendung beider Driftkammern (CJC1 + CJC2) durchgeführt. Trainiert wurden Netze mit vier, acht und sechzehn Trefferpunkten als Eingabe. In Tab. 6.3.1 sieht man, in welchem Abstand die Detektorlagen von der Strahlachse für diese drei Fälle gewählt wurden. In Abb. 6.8 sieht man die Ergebnisse des Netztrainings für die verschiedene Anzahl von Detektorlagen. Wie man sieht verbessert redundante Information beim Problem der Massenschätzung mit neuronalen Netzen nicht die Genauigkeit des Ergebnisses.
Lediglich der Fehler, der durch die drei Trainings entsteht, nimmt mit steigender Anzahl der Detektorlagen ab. Das Netz mit mehr Eingabeknoten erreicht also bei diesen Trainings im Mittel etwas stabilere Endergebnisse, benötigt jedoch mehr Zeit zum Training. Die weiteren Untersuchungen wurden alle mit acht Detektorlagen durchgeführt.

Detektorradius	1	2	3	4	5	6	7	8
4 Lagen	20	45	53	85				
8 Lagen	20	28	36	45	53	63	74	85
16 Lagen	20	23	27	30	34	37	41	45

Detektorradius	9	10	11	12	13	14	15	16
16 Lagen	53	58	63	67	72	76	81	85

Tabelle 6.1: *Die für die Parameterstudien verwendeten Detektorlagen. Die Radien sind so gewählt, daß die CJC1 und die CJC2 für die jeweilige Anzahl von Lagen voll genutzt werden. Der jeweilige Detektorradius ist in cm angegeben.*

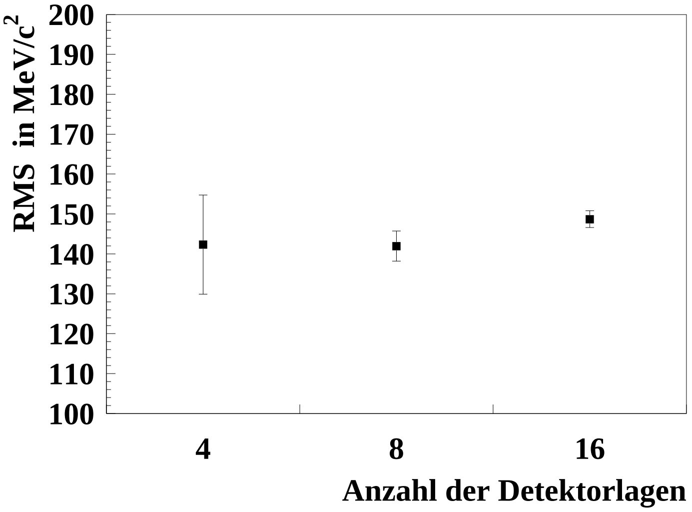


Abbildung 6.8: *Die Trainingsergebnisse für Netze mit vier, acht und sechzehn Trefferpunkten als Eingabe.*

6.3.2 Verwendete Driftkammern

Wie in Glg. 6.5 zu sehen ist, verringert sich der Fehler der Impulsmessung, wenn die Länge der projezierten Spur L zunimmt. Detektoren, die eine größere Ausdehnung und damit einen größeren Hebelarm besitzen, ermöglichen also bei längeren Spuren eine bessere Impulsschätzung, was sich auch in der Rekonstruktion invarianter Massen bemerkbar macht.

Auch beim Netztraining erwartet man einen Einfluß des verwendeten Hebelarms, da die Eingabemuster durch einen längeren Hebelarm unterschiedlicher werden und die Krümmungen der Teilchen in den Eingabedaten deutlicher zum Vorschein treten. Verwendet wurden Datensätze, die mit jeweils acht Trefferpunkten nur die CJC1 oder die CJC2 ausnützen und ein Datensatz der Trefferpunkte beider Driftkammern verwendet.

Da für die Eingabedaten gefordert wird, daß die Teilchen auf jeder Detektorlage einen Trefferpunkt besitzten, unterscheiden sich die Datensätze für die CJC1 und die CJC2 in der verwendeteten Θ-Region. Bei der Forderung der vollen Trefferzahl und der z-Ausdehnung der Driftkammern von -1.1 m bis +1.1 m vom Wechselwirkunspunkt aus gesehen, wird mit der CJC1 ca. der Bereich $20^0 < \theta < 160^0$ abgedeckt, während bei der CJC2 und dem Datensatz, der die CJC1 und CJC2 verwendet, nur noch ungefähr der Bereich $40^0 < \theta < 140^0$ abgedeckt wird. So kann man im Prinzip nur die Ergebnisse für die CJC2 alleine und für beide Driftkammern gemeinsam vergleichen. In Abb. 6.9 sieht man die Ergebnisse. Der längere Hebel bei Verwendung beider Driftkammern liefert die besten Resultate.

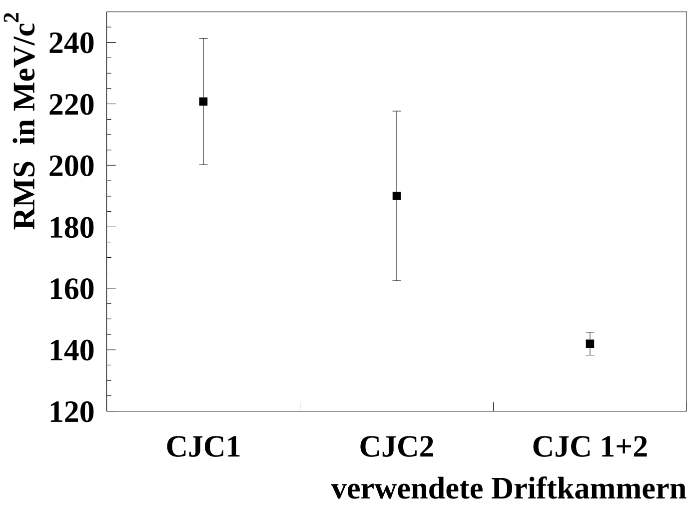


Abbildung 6.9: *Auswirkungen des Hebelarms auf die Trainingsergebnisse.*

Für die weiteren Untersuchungen werden beide Driftkammern verwendet.

6.3.3 Detektorauflösung

Die Genauigkeit der Einzelmessung ϵ ist der dritte Parameter in Glg. 6.5, der Einfluß auf den Krümmungsfehler hat. Abhängig von der Detektorauflösung erwartet man unterschiedlich gute Ergebnisse. Um invariante Massen zu rekonstruieren, benötigt man die Gesamtimpulse der Teilchen, also auch die Information über den Longitudinalimpuls. Dessen Fehler wird durch die Ungenauigkeit der Messung der z-Koordinate bestimmt. Bei den Driftkammern ist die Auflösung in r-ϕ-Ebene mit $\sigma_{r\phi} = 200\ \mu$m wesentlich besser als in z ($\sigma_z = 2.5$ cm). Es wurde untersucht inwieweit eine Variation dieser Auflösungen sich auf das erzielte Trainingsergebnis auswirkt. In Abb. 6.10 a) sieht man, daß sich die z-Auflösung wesentlich geringer auf das Endergebnis auswirkt als die Variation der r-ϕ-Auflösung 6.10 b).

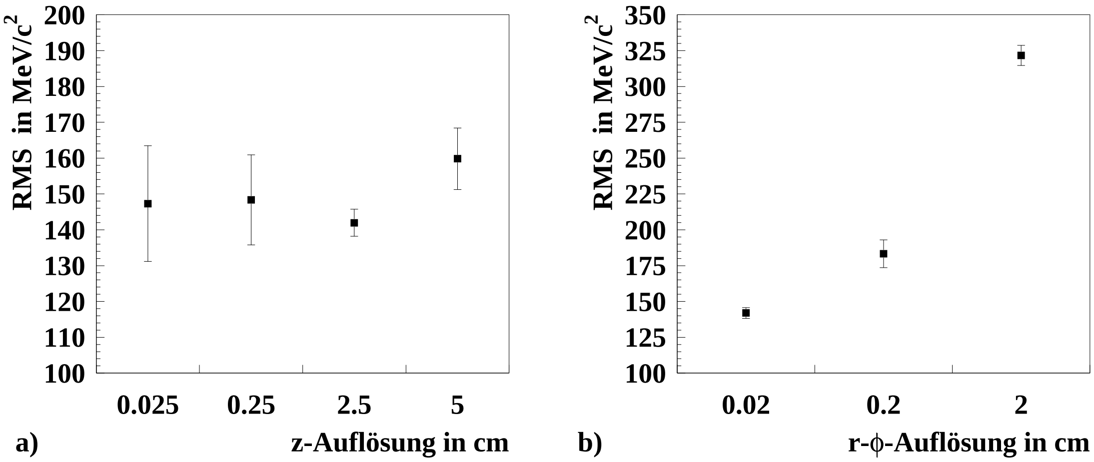


Abbildung 6.10: *Einfluß der Detektorauflösung a) Variation der z-Auflösung b) Variation der r-ϕ-Auflösung*

Dies läßt sich jedoch durch die verwendeten Trainingsdaten begründen. Betrachtet man nämlich die Verteilung des Longitudinalimpulses der generierten Daten für beide Teilchen in Abb. 6.11, sieht man, daß die meisten Teilchen wenig Longitudinalimpuls besitzen.

Der Grund dafür ist die Einschränkung des θ-Bereichs auf ca. 40^0 bis 140^0, der sich durch die Forderung der vollen Trefferzahl ergibt. Auch an den z-Verteilungen des Pions und des Kaons auf der äußersten Detektorlage (Abb.6.12) macht sich dies bemerkbar. Bei $\pm$ 1.1 m werden die Verteilungen abgeschnitten, da die Teilchen sonst nicht alle Treffer besitzen würden.

Die meisten Ereignisse besitzen also mehr Transversal- als Longitudinalimpuls. Es ist daher einsichtig, daß das neuronale Netz empfindlicher auf eine Variation der r-ϕ Auflösung reagiert als auf die der z-Auflösung.

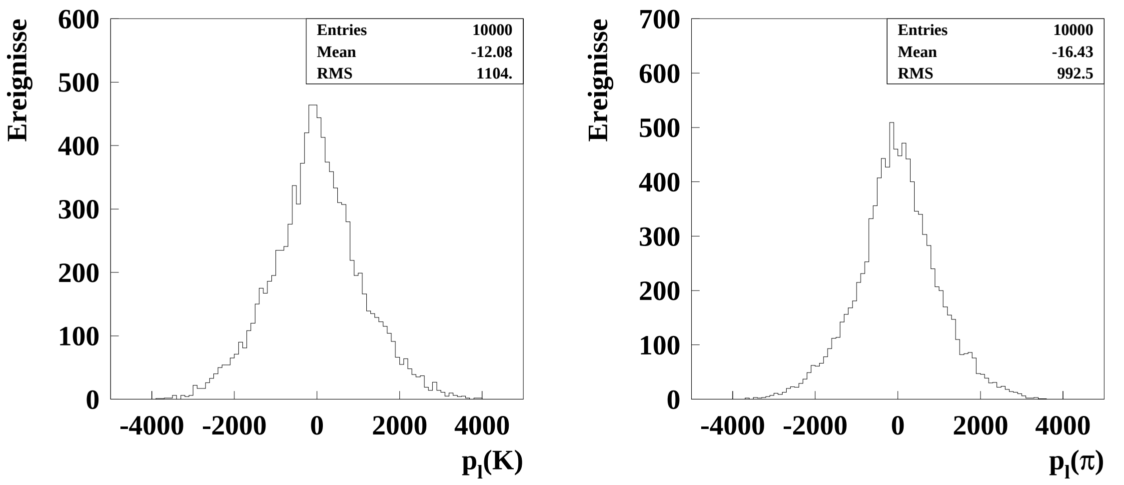


Abbildung 6.11: *Verteilung des Longitudinalimpulses des Kaons und des Pions.*

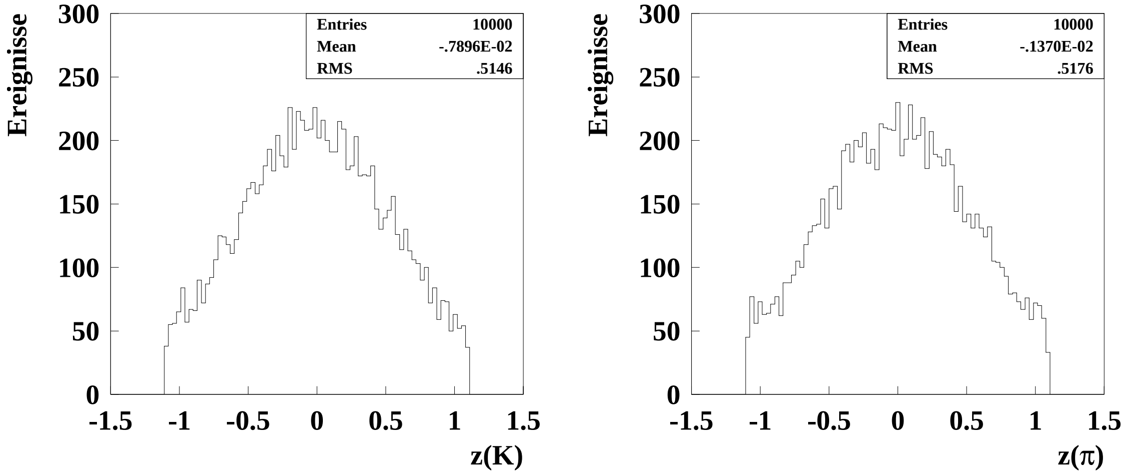


Abbildung 6.12: *Verteilung der z-Koordinaten auf der äußersten Detektorlage für das Kaon und das Pion.*

6.3.4 Trainierter Wertebereich

In den bisherigen Untersuchungen wurde die Masse des fiktiven Teilchen zwischen 1 und 3 GeV/c^2 gewählt. Die Größe dieses Bereichs ist dabei im Prinzip willkürlich festgelegt worden. Das wirkliche D^0 liegt mit seiner Masse von 1865 MeV/c^2 ungefähr in der Mitte des untersuchten Massenbereiches. Eine Vergrößerung bzw. Verkleinerung des trainierten Massenbereich bei gleichbleibendem Impulsbereich sollte direkte Auswirkungen auf die Ergebnisse haben, da sich die Anzahl der verschiedenen Muster analog verändert. In Abb. 6.13 a) sieht man die erwarteten Ergebnisse.

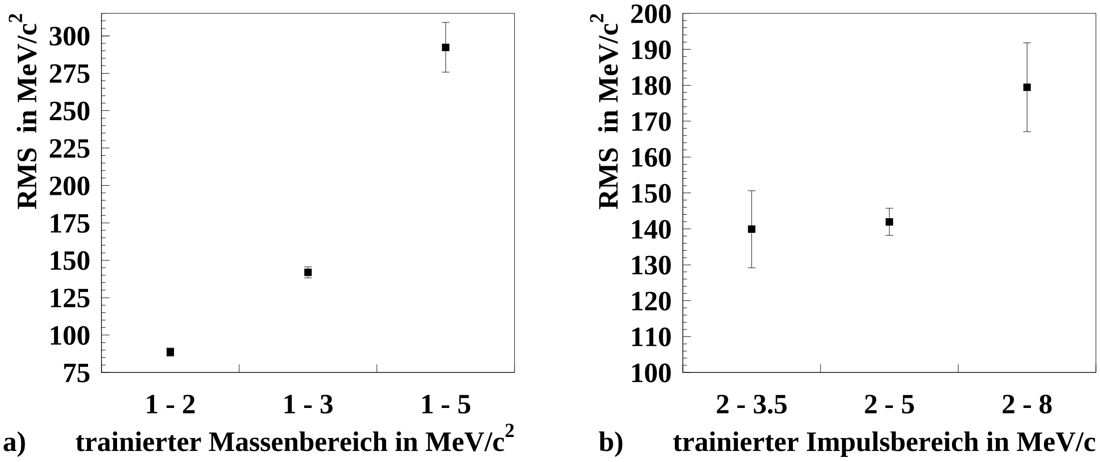


Abbildung 6.13: *Ergebnisse für verschiedene Wertebereiche a) Variation des Massenbereichs b) Variation des Impulsbereichs*

Das Training für den Massebereich 1 bis 2 GeV/c^2 liefert mit ca. 90 MeV/c^2 Massenauflösung das beste Ergebnis. Für den Massenbereich 1 bis 3 GeV/c^2 werden die Ergebnisse ca. 50 % schlechter. Das Training für den Massebereich 1 bis 5 GeV/c^2 ist ca. um den Faktor 3 schlechter als das für den Massebereich 1 bis 2 GeV/c^2. Analog kann man den Impulsbereich des fiktiven Teilchens bei einem festen Massenbereich variieren (Abb. 6.13 b). Auch hier ergibt sich eine Verschlechterung mit größer werdendem Wertebereich, wenngleich die Verdopplung des Impulsbereichs auf 2 - 8 GeV/c nur ca. 30 % schlechtere Ergebnisse liefert als bei 2 - 5 GeV/c.

6.3.5 Datenvorverarbeitung

Die Genauigkeit der Massenschätzung, die mit neuronalen Netzen erreicht werden kann, ist in entscheidendem Maße von dem Grad an Datenvorverarbeitung der Eingabe abhängig. Für die bisherigen Untersuchungen wurde ein maximaler Grad an Vorverarbeitung vorausgesetzt, indem eine Teilchenidentifikation gefordert wurde. Dadurch erhalten die gleichen Eingabeneuronen stets Eingaben des gleichen Teilchens K^- oder π^+. Da es das Ziel ist, die neuronalen Netze als Trigger anzuwenden, ist diese Annahme natürlich unrealistisch, weil in der kurzen Zeit bis zur Triggerentscheidung eine derartige Datenvorverarbeitung bisher nicht möglich ist.

Eine weitere Möglichkeit ist es, keine Teilchenidentifikation vorauszusetzen, sondern lediglich einen Trackingalgorithmus, mit dem Trefferpunkte, die zu einer Spur gehören, identifiziert werden. Zu welchem Teilchen diese Spur gehört ist in diesem Fall irrelavant, wenngleich die Zuordnung in dem untersuchten einfachen Fall mit zwei Teilchen unterschiedlicher Ladung natürlich trivial wäre. Dem Netz werden die gefundenen Spuren - sortiert nach dem Winkel des Trefferpunktes auf der ersten Detektorlage - präsentiert. Bei N Teilchen gibt es daher N! verschiedene Eingabemöglichkeiten.
Wenn man die Trefferpunkte der Teilchen auf jeder Detektorlage nach Winkeln sortiert und so dem Netz präsentiert, ist abgesehen von einem Algorithmus der aus den gesamten Treffern das Rauschen entfernt, keine weitere Datenvorverarbeitung nötig. Diese Eingabe wäre mit dem geringsten Aufwand an Vorverarbeitung verbunden, ist aber gleichzeitig für das Netz die schwierigste Aufgabe. Bei Teilchenspuren , die sich kreuzen, kann sich an einer beliebigen Stelle, die abhängig vom Schnittpunkt der Spuren ist, die Reihenfolge der Teilchentrefferpunkte ändern.

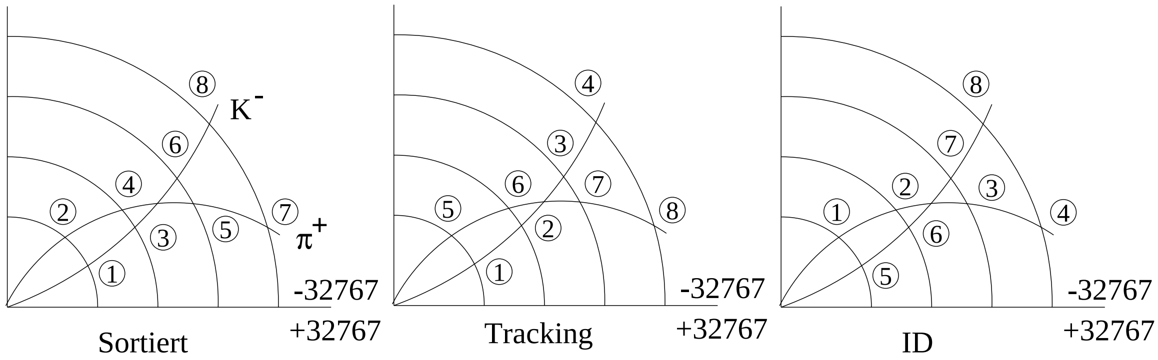


Abbildung 6.14: *unterschiedliche Möglichkeiten der Eingabe. Die Zahlen in den Kreisen geben an, welches Neuronenpaar welchen Trefferpunkt als Eingabe erhält.*

Zur Verdeutlichung sieht man in Abb. 6.14 die unterschiedlichen Netzeingaben für die drei untersuchten Eingabemöglichkeiten, bei vier verwendeten Detektorlagen. Die Zahlen in den Kreisen geben an, welches Neuronenpaar welchen Trefferpunkt (ϕ + z) als Eingabe erhält. Im Fall der Eingabe mit ID erhalten die ersten vier Neuronenpaare immer die Trefferpunkte des π^+, Neuronenpaare 5 - 8 die des K^-. Beim Tracking dagegen erhalten die Neuronenpaare 1 - 4 im gezeigten Fall die Treffer des K^-. Es wäre aber auch der Fall möglich, daß sie das π^+ erhalten, wenn dieses bei einem Ereignis den kleineren ϕ-Winkel auf der ersten Detektorlage hätte. Bei der sortierten Eingabe erhalten die Neuronenpaare, solange sich die Spuren nicht schneiden, immer abwechselnd den Treffer eines Teilchens. Zwischen der zweiten und dritten Detektorlage schneiden sich die Spuren, wodurch Neuronenpaar 4 und 5 Trefferpunkte des π^+ erhalten. Es findet also ein Bruch in der Eingabe statt, der dem Netz die Aufgabe weiter erschwert.
Die Trainingsergebnisse sind in Abb. 6.14 dargestellt. Man sieht, daß die Ergebnisse sowohl für die sortierte Eingabe als auch für Eingabe mit Tracking, wie erwartet schlechter werden, wobei die Ergebnisse für zwei Teilchen bei den beiden Methoden gleich sind.

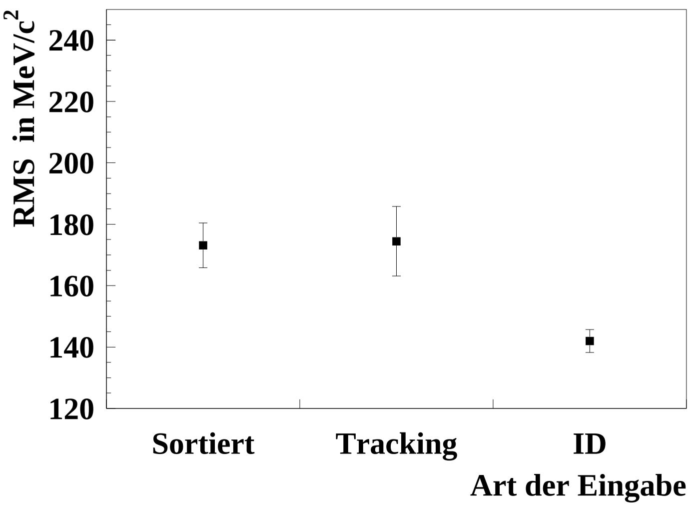


Abbildung 6.15: *Ergebnisse des Trainings bei unterschiedlicher Datenvorverarbeitung.*

6.3.6 Drei Teilchenspuren

Im folgenden soll noch untersucht werden, wie das Netz die Masse für den Zerfall eines fiktiven Teilchens in D^0 π^+ und dem anschließenden Zerfall des D^0 in K^- und π^+ schätzen kann. Untersucht wird dabei nur der Einfluß der unterschiedlichen Eingabe. Die Masse des fiktiven Teilchens wurde in dieser Untersuchung zwischen 2004 und 4004 MeV/c^2 gewählt. 2004 MeV/c^2 ist die Summe der Ruhemassen von D^0 und π und somit die untere Grenze für die Masse des fiktiven Teilchens. Mit zunehmender Teilchenzahl würde man erwarten, daß die Ergebnisse für die sortierte Eingabe am schlechtesten werden, da mehrere Schnittpunkte zwischen den Spuren auftreten können und die Eingabe noch variabler wird. Diesen Trend sieht man in Abb. 6.16 a) deutlich. Insgesamt verschlechtern sich die Ergebnisse deutlich gegenüber dem Zwei-Teilchen-Zerfall.

Anzahl der Trainingsdaten

Da die Zeit, die für ein Netztraining benötigt wird linear mit der Anzahl der Trainingsdaten ansteigt, wurde diese für die bisherigen Studien auf 10000 begrenzt. Mit dem verwendeten Trainingsprogramm bpexp2 sind jedoch bis zu 65535 verschiedene Trainingsdaten möglich. Da das Problem der Massenschätzung aus den Trefferpunkten eine schwierige Aufgabe für die Netze ist und z.B. die Rotationssymmetrie des Problems vom Netz gelernt werden muß, erwartet man in dem untersuchten Fall eine Verbesserung des Ergebnisses für eine größere Anzahl an Trainingsbeispielen. In Abb. 6.16 b) wird dies ersichtlich. Die Ergebnisse werden für alle Eingabearten besser. Der Trend daß die sortierte Eingabe schlechtere Ergebnisse liefert als die Eingabe mit Tracking bzw. ID, bleibt erhalten.

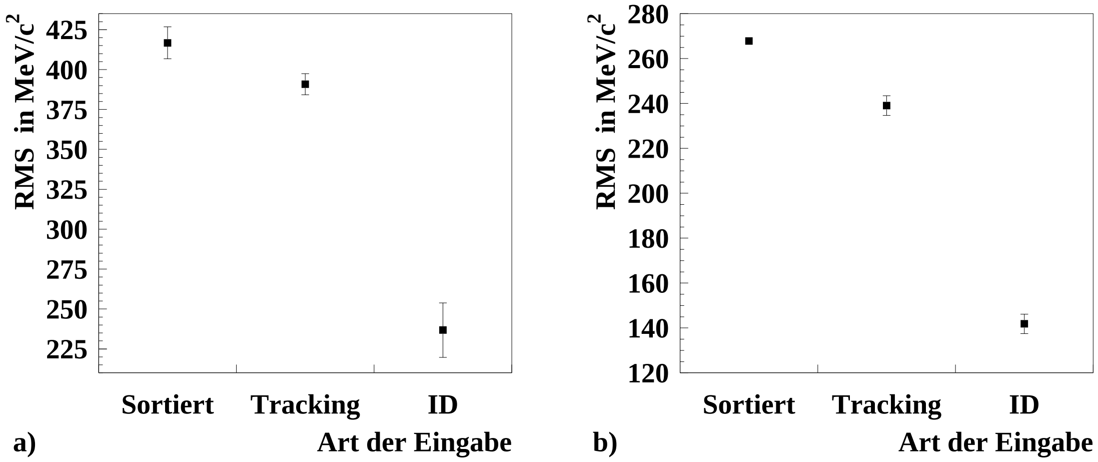


Abbildung 6.16: *Ergebnisse der Massenschätzung für den Zerfall eines fiktiven Teilchens in $D^0\ \pi_s^+$; $D^0 \rightarrow K^-\ \pi^+$. a) Ergebnisse für 10000 Trainingsdaten b) Ergebnisse für 65535 Trainingsdaten*

6.4 Analyse der Trainingsergebnisse

Im folgenden Abschnitt soll analysiert werden, welche Art von Ereignissen besonders gut bzw. besonders schlecht gelernt werden kann. Die Untersuchung wurde dabei für ein Netz durchgeführt, daß auf den Zwei-Teilchen-Zerfall mit sortierter Eingabe trainiert wurde. In Abb. 6.17 sieht man die vier Möglichkeiten, die sich bei zwei Spuren in der r-ϕ-Projektion topologisch ergeben können.

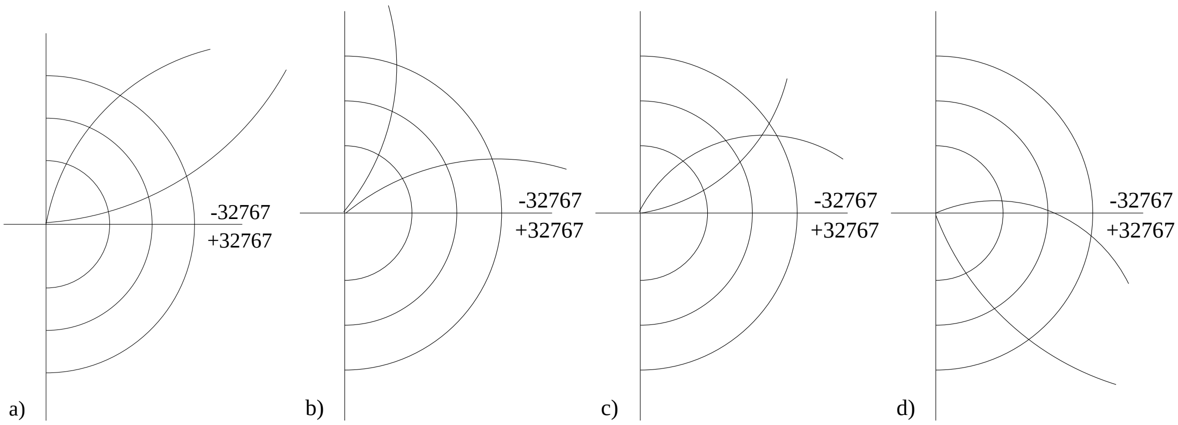


Abbildung 6.17: *Die verschiedenen Eingabemuster für zwei Spuren a) zwei zusammengehende Spuren b) zwei auseinandergehende Spuren c) zwei sich schneidende Spuren d) eine Spur beim Übergang von 0^0 nach 360^0*

Die Spuren können zusammengehen ohne sich im Detektor zu schneiden (Bild a), auseinandergehen (Bild b) oder sich im Detektor schneiden (Bild c). Bild d) zeigt einen Spezialfall, der sich durch die Abbildung des Wertebereichs von 0 bis 2π auf den Wertebereich des CNAPS [-32767,32767] ergibt. Falls bei einer Spur der Übergang von

0^0 nach 360^0 oder umgekehrt stattfindet, erhält man in den CNAPS-Eingabedaten einen Sprung von -32767 nach +32767 bzw. von +32767 nach -32767.
Für die Analyse wurden 20000 Testdaten mit einem Netz, welches die durchs Training gefundenen Gewichte best.wts verwendet, bewertet. In Abb. 6.18 sieht man die Ergebnisse des Netzes, unterteilt in die vier verschiedenen Klassen.
Man sieht, daß zusammengehende Spuren etwas besser gelernt werden können (Bild a) als auseinandergehende Spuren (Bild b), wenn man sich schneidende Spuren (Bild c) nicht mitberücksichtigt. In Bild c) sieht man den erwarteten Effekt, daß sich in r-ϕ-Projektion schneidende Spuren, wegen der wechselnden Eingabe (siehe Abschnitt 6.3.5) vom Netz wesentlich schlechter gelernt werden. Ereignisse, bei denen bei einer Spur der Übergang von 0^0 nach 360^0 oder umgekehrt, stattfindet, sind für das Netz am schwierigsten zu lernen (Bild d). Insgesamt gibt es 37 Ereignisse, bei denen sich die Spuren sowohl kreuzen als auch der Übergang stattfindet. Diese Ereignisse erscheinen sowohl in Bild c) als auch in Bild d). Der RMS für diese besonderen Ereignisse beträgt 612.1 MeV/c^2. Bild e) zeigt das Ergebnis für alle 20000 Testereignisse zusammen.
Ein Grund dafür, daß die Ereignisse aus Abb. 6.17 c) und d) schlechter gelernt werden, ist die geringere Anzahl solcher Ereignisse im Satz der Trainingsmuster. Man könnte die Ergebnisse verbessern wenn man einen Datensatz, der zum Training verwendet wird, zusätzlich mit schwer zu erlernenden Ereignissen anreichern würde. Jedoch würde sich kein besseres Ergebnis erzielen lassen, als das in Bild f), welches das Ergebnis für alle Events, außer denen aus Abb. 6.17 c) und d) zeigt.

Vergleich zwischen sortierter Eingabe und Eingabe mit ID
Um den Einfluß der unterschiedlichen Datenvorverarbeitung deutlich zu machen, sind in Abb. 6.19 die zu Abb. 6.18 c) und d) analogen Ergebnisse für ein Netz, das die Eingabe mit Teilchen ID erhält, dargestellt. Es wurde derselbe Testdatensatz verwendet, jedoch erhalten die ersten 16 Eingabeneuronen nun immer die Trefferpunkte des π^+. Dies macht sich bei den kreuzenden Spuren (Abb. 6.19 a) bemerkbar, die um den Faktor 2 besser geschätzt werden, als bei sortierter Eingabe (Abb. 6.18 c). Betrachtet man die Ereignisse mit dem Übergang, so stellt man fest, daß diese unabhängig von der Datenvorverarbeitung schlecht zu lernen sind. Der große Sprung im Eingabemuster ($\pm 32767 \rightarrow \mp 32767$) verursacht dabei die schlechten Ergebnisse.

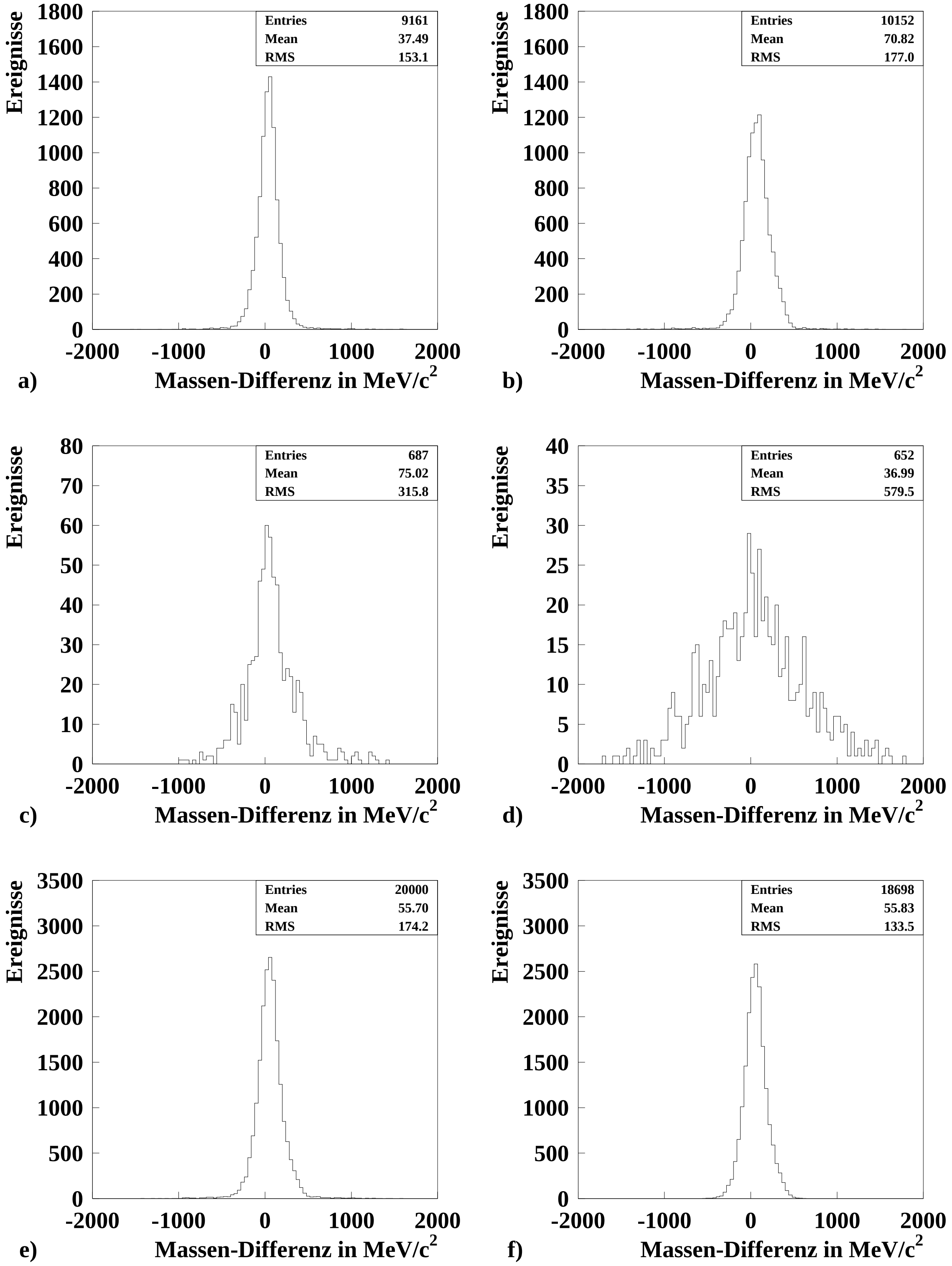


Abbildung 6.18: *Die Ergebnisse der einzelnen Klassen a) ΔM für zwei zusammengehende Spuren b) ΔM für zwei auseinandergehende Spuren c) ΔM für zwei kreuzende Spuren d) ΔM falls eine oder zwei Spuren den Übergang von 0^0 nach 360^0 bzw. umgekehrt, besitzen e) ΔM für alle Ereignisse f) ΔM für alle Ereignisse außer denen aus Bild c) und d)*

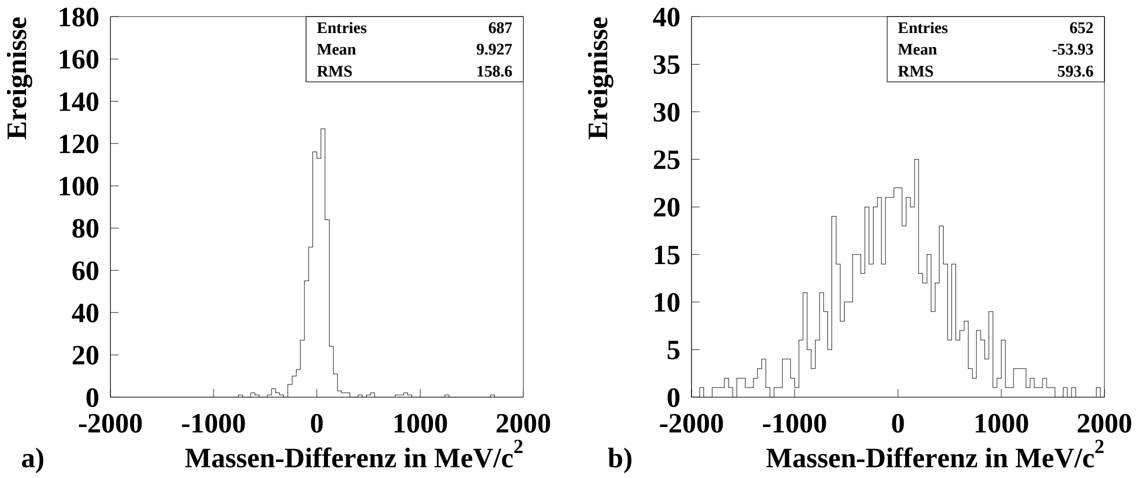


Abbildung 6.19: *Die Ergebnisse bei Eingabe mit ID a) ΔM für zwei kreuzende Spuren b) ΔM für zwei Spuren mit Übergang von 0^0 nach 360^0 bzw. umgekehrt.*

6.5 Vergleich mit der Rekonstruktion

Mit Glg. 6.4 in Abschnitt 6.3.1 läßt sich die invariante Masse eines Zwei-Teilchen-Zerfalls berechnen. Dazu benötigt man im Prinzip die P-θ-ϕ Tripel jedes Teilchens. Um die Qualität der Ergebnisse des Netztrainings zu beurteilen, soll eine Abschätzung gemacht werden, welche Massenauflösung man in der Rekonstruktion aus Driftkammerpunkten ungefähr erreichen kann.
Betrachtet werden richtige D^0 Zerfälle, die mit dem Zerfallsgenerator, diesmal mit fester Masse von 1865.0 MeV/c^2 generiert wurden. Man erhält zunächst die exakten Daten für P,θ und ϕ des Kaons und des Pions. Die Fehler dieser Daten wurden wie folgt simuliert:

Fehler bei der Impulsmessung
Gleichung 6.5 aus Abschnitt 6.3.1 gibt den Krümmungsfehler für Spuren an. Die Genauigkeit, mit der die Krümmung gemessen wird, hängt nicht von der Krümmung selbst ab. Für den Transversalimpuls

$$p_t = 0.003 \frac{GeV}{T \cdot cm} \cdot \frac{B}{\kappa} \tag{6.6}$$

gilt daher :

$$\sigma_{p_t} = \frac{1}{0.003 \frac{GeV}{T \cdot cm} B} p_t^2 \sigma_\kappa \tag{6.7}$$

Die Impulsauflösung verschlechtert sich mit wachsendem Transversalimpuls, da die Sagitta der Spur kleiner wird. Vor allem die Spurlänge hat großen Einfluß auf den Fehler.

Während sich für lange Spuren (L_{max} = 60cm ,N_{max} = 56) eine nominelle Auflösung von

$$\sigma_{p_t}/p_t = 0.5\%\ GeV^{-1} \cdot p_t$$

ergibt, verschlechtert sich dieser Wert für sehr kurze Spuren (L = 10 cm) auf ca. 50% $\mathrm{GeV}^{-1} \cdot p_t$.
Beim Netztraining wurde stets die volle Trefferzahl in beiden Driftkammern und damit L = 60 cm, N = 8 gefordert. Mit diesen Werten erreicht man 1.2% $GeV^{-1} \cdot p_t$. Für einen Vergleich mit den Netzen wurde deshalb dieser Fehler für den Transversalimpuls angenommen.

Fehler bei der Bestimmung von θ
Aus dem exakten θ-Winkel wurde die z-Koordinate auf der äußersten Detektorlage (r=85cm) berechnet. Auf diese wird ein gaußscher Fehler mit σ_z = 2.5 cm addiert und dann wieder der θ-Winkel berechnet.

Fehler bei der Bestimmung von ϕ
Mit dem ϕ-Winkel werden die x- und y-Koordinaten auf der äußersten Detektorlage berechnet. Fehler mit $\sigma_{r\phi}$ = 200 μm werden addiert und ϕ daraus zurückgerechnet.

In Abb. 6.20 ist die Massenauflösung für 10000 Events simuliert worden. Dabei sieht man in Bild a) den Fehler, der alleine durch die Impulsmessung entsteht, in Bild b) ist der Fehler, der durch ungenaue θ-Auflösung ensteht, dargestellt. Bild c) zeigt die Begrenzung der Genauigkeit durch die ϕ-Auflösung. Wie man sieht, sind die Fehler, die durch die ungenaue z-Auflösung und die Impulsmessung entstehen, ungefähr gleich groß, während der Fehler durch die begrenzte ϕ-Auflösung vernachlässigt werden kann. In Bild d) sieht man die in der Rekonstruktion erreichbare Massenauflösung unter Berücksichtigung aller Fehler. Die Massenauflösung in der Rekonstruktion des D^0 unter genannten Vorraussetzungen beträgt also ca. 35 MeV/c^2.

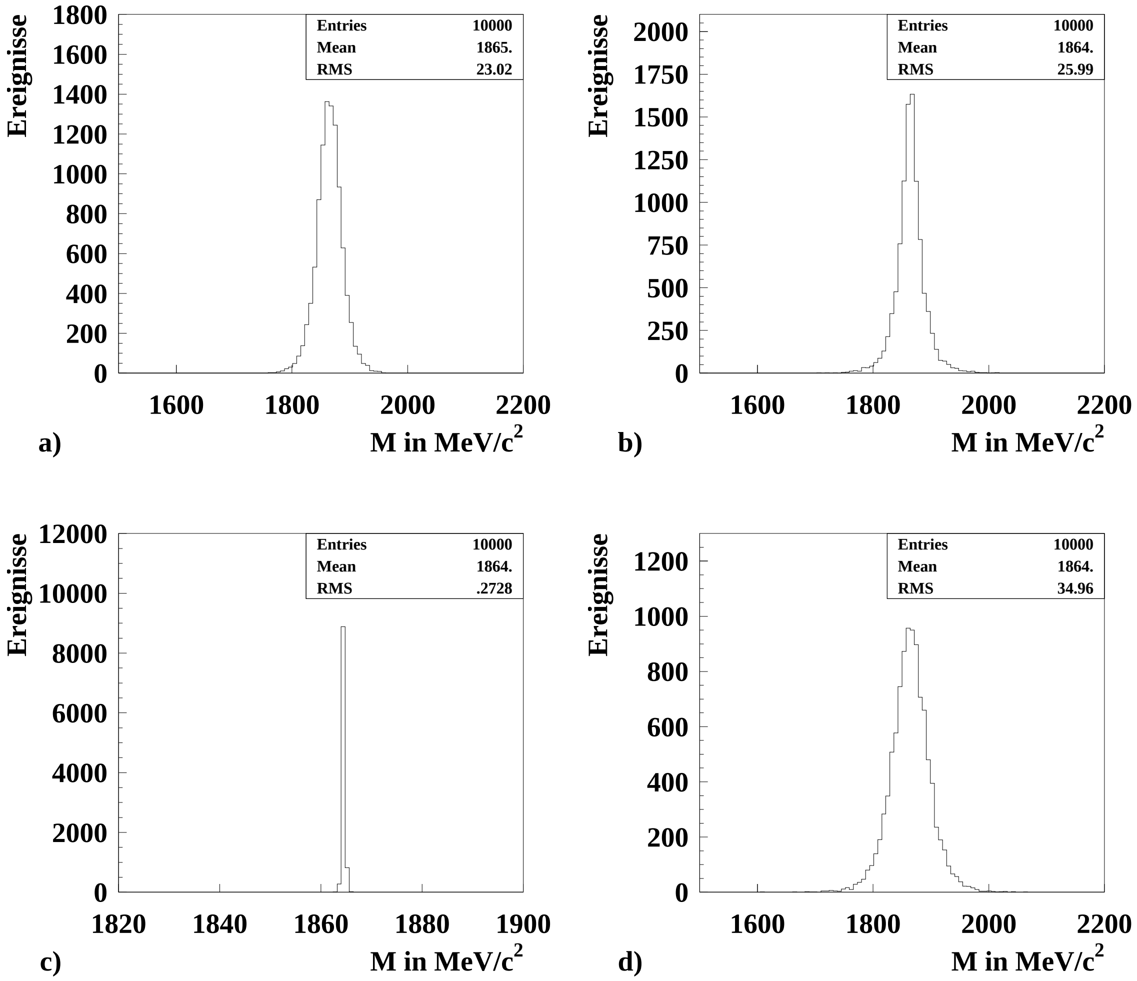


Abbildung 6.20: *Abschätzung für die Fehler bei der Rekonstruktion invarianter Massen. Aufgetragen sind jeweils die rekonstruierten Massen unter Berücksichtigung der jeweiligen Fehler a) Fehler durch Impulsmessung b) Fehler durch Ungenauigkeit von θ c) Fehler durch Ungenauigkeit von ϕ d) Rekonstruierte Masse, unter Berücksichtung der Fehler in P, θ und ϕ*

6.6 Diskussion der Ergebnisse

Zusammenfassend kann festgestellt werden, daß eine Schätzung von invarianten Massen mit neuronalen Netzen für einfache Probleme möglich ist. Die Ergebnisse sind wie in der Rekonstruktion von Eigenschaften des Detektors abhängig, wenngleich nicht jede Variation einer Detektorgröße Einfluß hatte (z.B. Anzahl der Detektorlagen). Verglichen mit der Rekonstruktion sind die Ergebnisse der Netze schlechter. Eine Massenauflösung von ca. 100 MeV/c^2 wäre für einen neuronalen Trigger ein gutes Ergebnis, diese Genauigkeit wird jedoch nur bei stark vereinfachten Problemen erreicht. Erschwert man die Aufgabe, indem man geringere Datenvorverarbeitung fordert (sortierte Eingabe) oder Massen aus drei Teilchenspuren rekonstruiert, so werden die Ergebnisse mit 270 MeV/c^2 bei drei Teilchenspuren mit sortierter Eingabe und maximaler Anzahl an Trainingsdaten deutlich schlechter. Bereits bei drei Teilchenspuren werden die Grenzen des Konzepts aufgezeigt. Da diese Ergebnisse nicht mehr im Rahmen der geforderten Genauigkeit von ca. 100 MeV/c^2 liegen, wurden realistische Probleme wie unvollständige Spuren (Trefferpunkte nur auf einer bestimmten Anzahl von Detektorlagen), Rauschpunkte im Detektor oder Spuren von Teilchen, die nicht zum betrachteten Zerfall gehören, nicht untersucht. Auch wurde in der Untersuchung nur der Zerfall des D^{*+} betrachtet, wobei ein Trigger natürlich auch den ladungskonjugierten Fall erkennen müßte. Die Ergebnisse würden mit steigendem Schwierigkeitsgrad natürlich schlechter werden, so daß man zu dem Schluß kommen kann, daß der Einsatz eines neuronalen Netzes, welches ohne massive Datenvorverarbeitung aus Spurpunkten invariante Massen rekonstruieren soll, verworfen werden muß. Das reale Problem erweist sich als so schwer, daß es mit der betrachteten, einfachen Netzwerkarchitektur nicht in einem Schritt von neuronalen Netzen gelöst werden kann. Eine weitere Datenvorverarbeitung der Detektortrefferpunkte ist daher notwendig. Ein Konzept, wie eine intelligente Vorverarbeitung von Spurpunkten mit Hilfe von neuronalen Netzen durchgeführt werden kann, soll daher im nächsten Kapitel vorgestellt werden.

Kapitel 7

Datenvorverarbeitung mit neuronalen Netzen

Bereits in Abschnitt 4.3.2 wurde erwähnt, daß sich neuronale Netze, die durch unüberwachtes Lernen trainiert werden, zur Datenvorverarbeitung einsetzen lassen. Es soll ein Konzept vorgestellt werden, das sowohl mit parallelen Softwarealgorithmen, als auch neuronalen Netzen aus Trefferpunkten in der Driftkammer Spuren rekonstruieren soll. Das Ziel wäre es, in Zukunft mit einer derartigen Kombination Online-Tracking zu ermöglichen.

7.1 Clustern von P, θ, ϕ Spurelementen

Eine vollständige Spurrekonstruktion nur mit neuronalen Netzen ist z.B. mit der Elastic-Arms Methode [48, 49] oder dem „Growing Neural Gas“-Algorithmus [50] möglich, ist jedoch mit derzeitiger Hardware zu rechenintensiv um online eingesetzt werden zu können. Auch die herkömmliche Methode, an Detektortrefferpunkte Spuren zu fitten, dauert zu lange um auf früher Triggerebene Spuren zu rekonstruieren. Denkbar wäre jedoch ein Software/Hardware-Algorithmus, der aus den Trefferpunkten jeweils Kombinationen von drei Treffern betrachtet und eine Helix durchlegt. Da die Driftkammern insgesamt 56 Signaldrähte besitzen, könnte man z.B. Trefferpunkte bestimmter Lagen verwenden, um so die Kombinationsmöglichkeiten geringer zu halten. Dazu teilt man die Driftkammer in Sektoren ein, die radial in weitere Bereiche unterteilt werden. In diesen Bereichen werden dann jeweils wenige Spuren berechnet. Kombinationen, bei denen der Aufpunkt der Helix von der Wechselwirkungszone entfernt ist, könnten sofort verworfen und so eine Filterung von Rauschpunkten durchgeführt werden. Eine Spur wird durch P, θ und ϕ parametrisiert, wobei die erhaltenen Tripel mit unterschiedlichen Fehlern behaftet sind. Für wirkliche Teilchen sollten sich im dreidimensionalen Raum (P-θ-ϕ) Anhäufungen (Cluster) ergeben. Kombinationen, die von keiner echten Spur stammen, würden nur vereinzelt auftreten, da die meisten bereits durch die Forderung, daß eine Teilchenspur aus der Wechselwirkungszone kommen muß, eliminiert werden. Interessant sind die Clusterzentren, da sie ungefähr die Impulse und Winkel der echten Teilchen darstellen.

Eine Möglichkeit diese Clusterzentren zu finden, ist es, diese mit herkömmlichen sequentiellen Clusteralgorithmen zu berechnen. Da es das Ziel, ist Online-Tracking zu machen, werden parallele Algorithmen aus Zeitgründen besser geeignet sein. Hier soll gezeigt werden, daß sich neuronale Netze mit ihrer inhärenten Parallelität für diese Aufgabe eignen. Verwendet werden dazu Netze , die durch unüberwachtes Lernen die Clusterzentren finden. Das Training mit den gefundenen Tripeln als Eingabedaten darf dabei nur wenige Epochen dauern, da es beim laufenden Experiment online durchgeführt werden würde. Für erste Tests wird in dieser Arbeit ein „Growing Neural Gas“-Netzwerk verwendet, weshalb der Lernalgorithmus im folgenden genauer erläutert wird.

7.2 Der „Growing Neural Gas“-Algorithmus

Der von B. Fritzke [32, 51] entwickelte „Growing Neural Gas“-Algorithmus (GNG) ist ein Netzwerkmodell, das in der Lage ist, durch unüberwachtes Lernen die Topologie von Eingabedaten nachzubilden.
Die im folgenden betrachteten Netzwerke bestehen aus:

- Einer Anzahl von Zellen (Knoten), die jeweils durch einen Referenzvektor $\omega \in R^n$ beschrieben werden. Die Referenzvektoren spiegeln die Position der Zellen im Eingaberaum wieder.

- Einer Anzahl von Verbindungen (Kanten), die jeweils Paare von Knoten verbinden. Ihnen wird jedoch nicht wie bei anderen Netzwerkarchitekturen ein Gewicht zugeordnet. Sie dienen lediglich zur Definition einer Nachbarschaftsbeziehung und zur Modellierung der topologischen Struktur des Netzwerkes.

Präsentiert werden n-dimensionale Eingabedaten, die einer (unbekannten) Wahrscheinlichkeitsverteilung $P(\chi)$ genügen.
Die Topologie des GNG Netzwerkes wird schrittweise durch kompetitives Hebbsches Lernen gebildet, einer Methode die von Martinetz [52] vorgeschlagen wurde. Die Grundidee des kompetitiven Hebbschen Lernens besteht darin, bei jedem Anpassungsschritt Kanten zwischen den zwei Zellen, die am nächsten am Eingabevektor liegen, zu bilden.
Der GNG Algorithmus wird im Prinzip durch einen Alterungsprozeß der Kanten vorangetrieben und sieht folgendermaßen aus:

1. Beginne mit einer Kante, die zwei Zellen a und b, an zwei zufälligen Positionen ω_a und ω_b verbindet und deren Alter zu Beginn Null ist.

2. Präsentiere ein Eingabemuster χ.

3. Berechne den euklidischen Abstand zwischen χ und allen Zellen.

4. Finde die am nächsten gelegene Zelle ("winner") und die zweitnächste Zelle.

5. Falls diese Zellen durch eine Kante verbunden sind, setze ihr Alter auf Null; Falls sie nicht verbunden sind, kreiere eine Verbindung zwischen ihnen.

6. Erhöhe das Alter aller Kanten, die von der Gewinnerzelle ausgehen.

7. Entferne Kanten mit einem Alter $> a_{max}$. Falls dadurch unverbundene Zellen entstehen, entferne diese ebenfalls.

8. Summiere die quadratischen Abstände zum Eingabevektor χ in einer lokalen Zählervariable n auf.

9. Falls diese Variable eine bestimmte Schwelle überschreitet, teile die Kante zum entferntesten, topologischen Nachbarn auf und füge eine neue Zelle zwischen den beiden ein.

10. Verschiebe die Gewinnerzelle und ihre topologischen Nachbarn zum Eingabevektor χ hin.

GNG Netzwerke haben den Vorteil, daß sie keine Bedingungen an die Dimension der Eingabedaten stellen. Die Topologie von GNG Netzwerken spiegelt die Topologie der Verteilung der Eingabedaten wieder und kann in verschiedenen Bereichen des Eingaberaums unterschiedlich dimensional sein. In Abb. 7.2 sieht man dies veranschaulicht.
Im betrachteten Problem besteht der Eingaberaum aus mehreren Anhäufungen dreidimensionaler Datenpunkte (Tripel). Durch den Lernprozeß teilt das Netz die Eingabedaten in verschiedene Klassen, auch Cluster genannt, ein. Je länger ein Training dauert, umso besser paßt sich das Netz an die Eingaben an.

In Abb. 7.1 sieht man ein Training für zweidimensionale Eingabedaten. Es wurden zufällig Datenpunkte aus den umrandeten Bereichen, die die Cluster begrenzen, präsentiert. Es zeigt sich, daß bei einem austrainierten Netz mehrere Neuronen für einen Cluster zuständig sind. Der Trick beim Clustern von p-θ-ϕ Tripeln besteht dar-

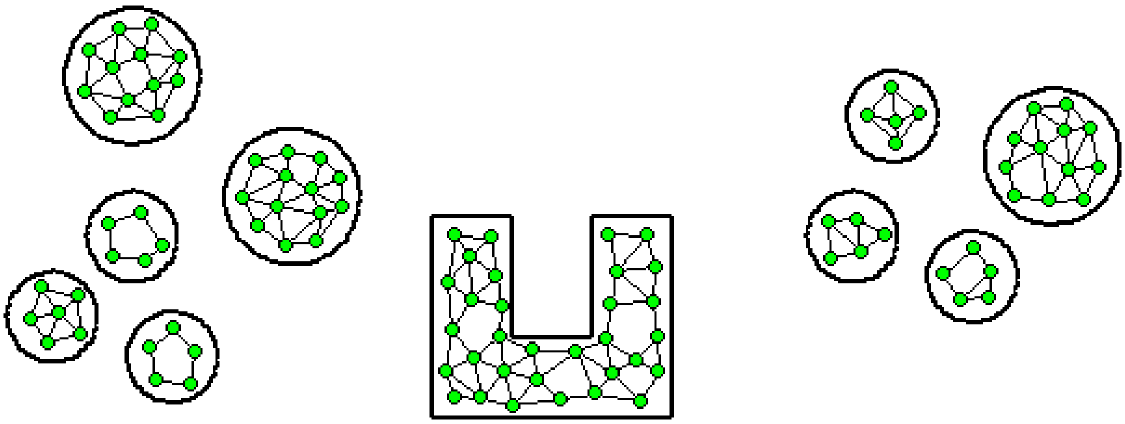


Abbildung 7.1: *Cluster, gefunden mit einem „Growing Neural Gas“-Algorithmus.*

in, gerade so kurz zu trainieren, daß die Cluster bereits gefunden sind aber möglichst noch nicht durch mehere Neuronen repräsentiert werden. Ähnliche Eingabedaten werden dann durch einen einzigen Repräsentanten dieser Klasse beschrieben, der den Impuls und die Winkel eines Teilchens widerspiegelt.

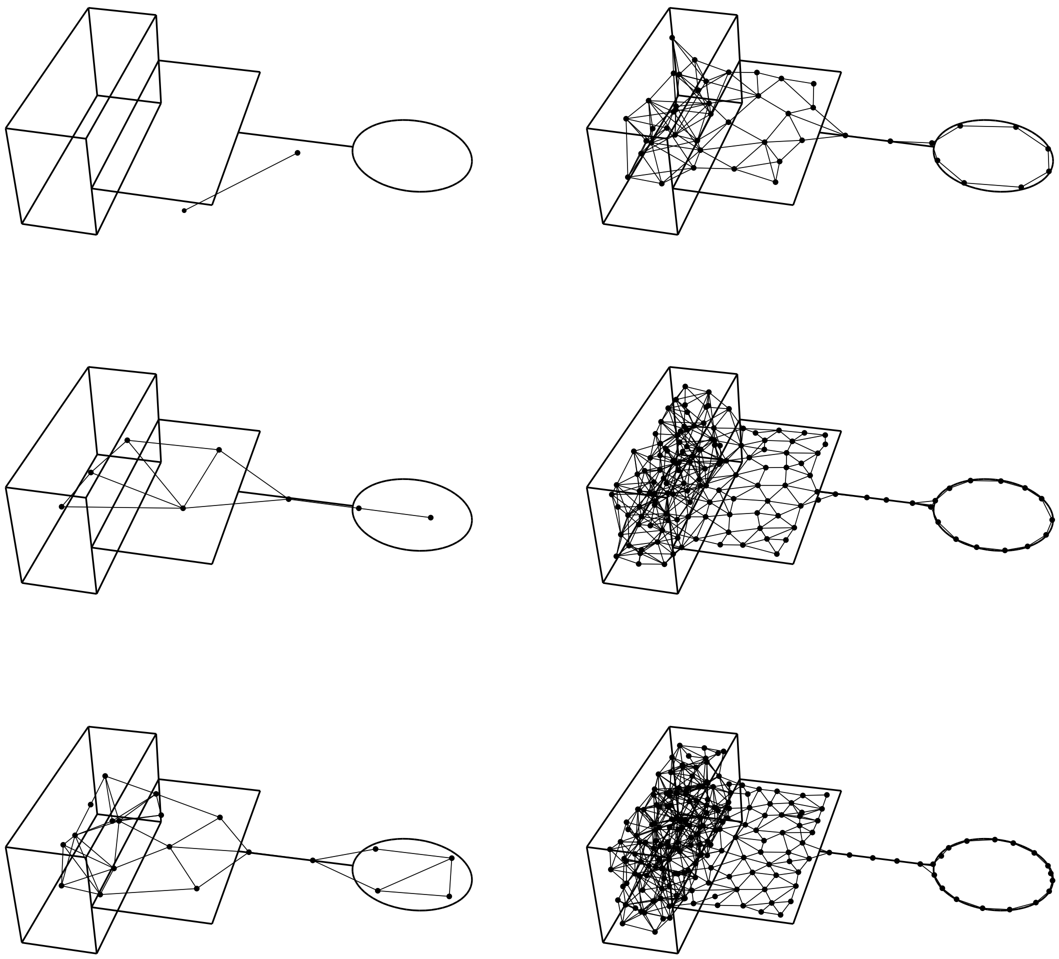


Abbildung 7.2: *Das Growing Neural Gas Netzwerk paßt sich an eine Signalverteilung unterschiedlicher Dimensionalität an. Gezeigt ist das Netz zu Beginn, bestehend aus zwei zufällig gesetzten Zellen und die Netzwerke nach 600, 1800, 5000, 15000 und 20000 präsentierten Datenmustern. Das Beispiel stammt aus [32].*

7.3 Trainingsdaten

Für die Erzeugung der Trainingsdaten geht man von einem P-θ-ϕ Tripel aus, das einem wirklichen Teilchen entspricht. Für diese Ausgangswerte werden, wie in Abschnitt 6.5 mit gaußschen Fehlern behaftete Tripel generiert, die den einzelnen Spurelementen, gebildet aus drei Spurpunkten, entsprechen. Dies wird für jede Spur durchgeführt. Zum Training wurde das Programm GNG von L. Janauschek verwendet, welches den vorgestellten Algorithmus implementiert. Es ermöglicht die Einstellung zahlreicher Parameter, wobei in dieser Untersuchung jedoch nur einige verwendet wurden. Eine genauere Beschreibung des Programms und seiner Möglichkeiten findet man in [53]. Für das Training von Relevanz erwiesen sich folgende Parameter:

- ageThreshold: gibt an, wann eine Zelle wieder entfernt werden soll.
- maxNodeCount: gibt die maximale Anzahl von Neuronen an
- counterThreshold: bestimmt, wann eine Zelle geteilt wird.
- learnParamWinner: Lernparameter des Gewinnerknotens
- learnParamNeighbours: Lernparameter der Nachbarn des Gewinnerknotens
- lowerCounterThreshold: Mit diesem Parameter können nach dem Training Neuronen entfernt werden, die kaum trainiert wurden.

In dieser Arbeit wurden aus Zeitgründen nur erste Tests durchgeführt, so daß keine genauere Parameterstudie vorliegt. Die verwendeten Werte sind aber ein guter Anhaltspunkt für weitere Studien.

Beim Test erwiesen sich folgende Parameter als vorteilhaft:

Eine Einstellung des ageThresholds auf ca. 10 erwies sich als vorteilhaft, da dann kaum trainierte Zellen schon während des Trainings wieder entfernt werden. Der Wert ist allerdings abhängig von der im Mittel pro Spur präsentierten Tripel. Auch der counterThreshold ist abhängig von der Anzahl an Spurtripeln pro Spur. Ein Wert um 20 erwies sich bei den Testbeispielen als vorteilhaft. Als maximale Anzahl von Neuronen (maxNodeCount) wurde immer eine deutlich größere Anzahl als tatsächliche Spuren gewählt. In den untersuchten Beispielen wurde 12 gewählt. Bei der Einstellung der Lernparameter erwies es sich als günstig, einen möglichst großen Lernparameter für den Gewinnerknoten und einen möglichst kleinen für die Nachbarn zu wählen. Für den learnParamWinner wurde 0.3, für learnParamNeighbours 0.001 gewählt. Diese Wahl legt nahe, in eventuellen weiteren Arbeiten auch andere unüberwachte Algorithmen, bei denen keine Nachbarschaftsbeziehung zwischen den Knoten besteht, zu testen. Als lowerCounterThreshold erwies sich ein Wert von ca. 10 als günstig.

7.4 Funktionalität des Algorithmus

Sechs Cluster

Zunächst soll gezeigt werden, daß sich der Algorithmus zum Auffinden von Clusterzentren eignet. In Abb. 7.4 sieht man ein Probetraining für sechs Spuren abgebildet. Die kleinen Punkte sind dabei jeweils gefundene Spurelemente, wobei pro Spur jeweils 30 Tripel mit den in Abschnitt 6.5 verwendeten Fehlern für P, θ und ϕ präsentiert wurden. Die Vierecke stellen die Neuronen dar, die sich während des Trainings immer besser in die Clusterzentren bewegen. Bereits nach drei Epochen sind in diesem Beispiel sämtliche Cluster gefunden worden. In der Mitte sieht man ein Neuron, das nur am Rande eines Clusterzentrums liegt. Es ist allerdings so selten trainiert worden, daß es durch Anwendung des lowerCounterThreshold nach Beendigung des Trainings entfernt wird.

Zwei Cluster

Für einen ersten Test, ob der Algorithmus die Auflösung verbessert, wurden für einen D^0-Zerfall jeweils 30 Spurelemente für das K^- und das π^+, mit den Fehlern aus Abschnitt 6.5 generiert. In Wirklichkeit werden die einzelnen Fehler größer sein, da ein Tripel jeweils nur aus drei Spurpunkten rekonstruiert wird. Es soll lediglich die Funktionalität des Algorithmus bewiesen werden.
Es wurden 2000 D^0-Ereignisse jeweils drei Epochen lang trainiert. Dabei wurden bei 1815 Ereignissen beide Cluster gefunden. Die nach dem Training erhaltenen Positionen der Neuronen sind die Clusterzentren und damit gemittelte Impulse und Winkel. Rekonstruiert man aus diesen Werten die invariante Masse des D^0, so erwartet man also einen besseren Wert als den Wert von ca. 35 MeV/c^2 aus Abschnitt 6.5, da die Fehler der Größen durch die Mittelung noch kleiner werden. In Abb. 7.3 sieht man das Ergebnis für die 1815 erkannten Ereignisse. Der RMS beträgt nur noch

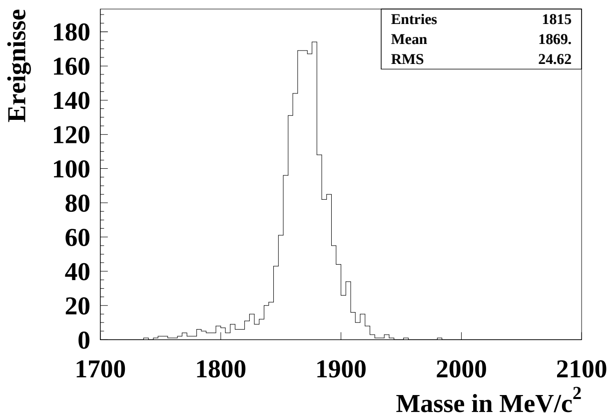


Abbildung 7.3: *Rekonstruierte Masse des D^0 aus 1815 erkannten Ereignissen.*

24.62 MeV/c^2, was die Funktionalität des Algorithmus beweist.

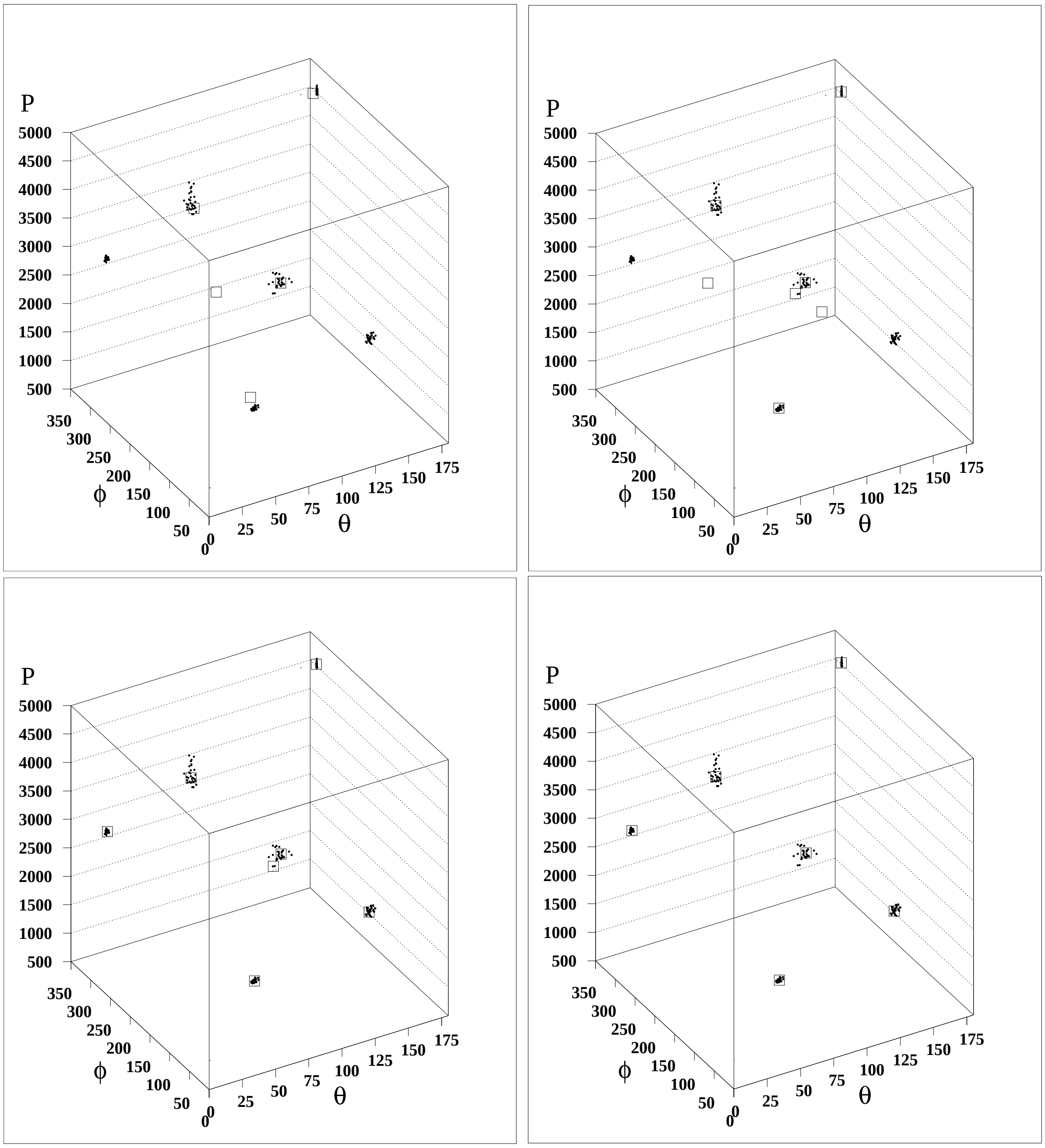


Abbildung 7.4: *Training eines Netzes mit sechs Clustern. Von links oben nach links unten sieht man das Ergebnis nach einer, zwei bzw. drei Epochen. Das Bild rechts unten zeigt die Neuronen nach Entfernung, des wenig trainierten Knotens in der Mitte durch Anwendung des lowerCounterThresholds.*

7.5 Rekonstruktion von D*-Mesonen

Nach den ersten erfolgreichen Tests, soll nun ein realistischeres Problem betrachtet werden. Bei der Realisierung des Konzepts werden die Fehler für die Impulse nämlich wesentlich größer sein, als im vorigen Abschnitt angenommen.
Betrachtet man, wie vorgestellt, nur jeweils drei Punkte aus einem bestimmten Bereich der Spurkammer, so ist der Fehler für die Spurkrümmung in Glg. 6.5 quadratisch abhängig von der Größe dieses Bereichs, da er die projezierte Spurlänge L bestimmt. Teilt man die Driftkammer in Sektoren auf, von denen jeder z.B. in sieben Bereiche (jeweils acht Signaldrähte) aufgeteilt wird, so ist die projezierte Spurlänge eines Spurelements aus einem Bereich mit L $\approx$ 9 cm sehr kurz. Da die Spur nur noch an drei Punkten gemessen wird, ist der Faktor $\sqrt{\frac{720}{N+4}}$ in Glg. 6.5 nicht mehr korrekt. Nach [47] wird stattdessen der Faktor 9 verwendet. Insgesamt erhält man mit diesen Werten eine nominelle Auflösung von $\sigma_{p_t}/p_t \approx 60\%\ GeV^{-1} \cdot p_t$. Selbst wenn man pro Bereich zwei Tripel berechnet, so daß man pro Spur insgesamt 14 Tripel erhält, kann der Fehler nicht besser als $\sigma_{p_t}/p_t/\sqrt{N} \approx 16\%\ GeV^{-1} \cdot p_t$ für N = 14 werden. Diese Auflösung ist jedoch zu schlecht, wenn man eine angestrebte Massenauflösung von ca. 50 MeV/c^2 aus den gefundenen Spuren erhalten will.
Man hat also zwei gegenläufige Effekte. Mit zunehmender Größe der gewählten Bereiche der Spurkammern wird einerseits die Anzahl der Kombinationsmöglichkeiten von drei Punkten zu einer Spur größer und es treten mehr Probleme mit Spuren auf, die über mehrere Sektoren gehen, andererseits wird die erreichbare Impulsauflösung besser. Um eine Massenauflösung von ca. 50 MeV/c^2 zu erhalten, benötigt man eine Impulsauflösung von ca. $\sigma_{p_t}/p_t \approx 2.5\%\ GeV^{-1} \cdot p_t$ wenn man die Fehler für θ und ϕ aus Abschnitt 6.5 annimmt. Dies entspricht einer projezierten Spurlänge von ca. 44 cm. Da man mehrere Spurelemente mißt, können etwas größere Fehler für die Tripel erlaubt werden, da der Fehler mit der Anzahl der Messungen abnimmt.

Um die Auswirkungen einer schlechteren Impulsauflösung zu studieren, wurde wieder der Zerfall $D^{*+} \rightarrow \mathrm{D}^0\ \pi_s^+$; $\mathrm{D}^0 \rightarrow \mathrm{K}^-\ \pi^+$ betrachtet. Als Trainingsdaten wurden pro Ereignis jeweils 15 Spurtripel für die Zerfallsteilchen generiert. Bei der Erzeugung wurde immer von den exakten P, θ und ϕ-Werten der Teilchen ausgegangen. Es wurden drei Datensätze mit je 2000 Ereignissen für die unterschiedliche Impulsauflösung $\sigma_{p_t}/p_t = 2.5\%$, 5% und 10% $GeV^{-1} \cdot p_t$ erstellt. Dies entspricht einer projezierten Spurlänge von ca. 44 cm, 31 cm bzw. 22 cm für die aus drei Punkten gebildeten Spuren. Die Fehler für θ und ϕ wurden wie in Abschnitt 6.5 simuliert. Es wurde nur drei Epochen auf den Daten trainiert. Der lowerCounterThreshold wurde gleich 7.0 gewählt.

In Abb. 7.5 sieht man in der rechten Spalte die aus den in drei Epochen gefundenen Clustern rekonstruierten Massen. Ein Cluster wurde dabei als „gefunden“ klassifiziert, wenn der Impuls vom exakten Ausgangswert nicht mehr als 500 MeV/c differiert und die Winkel jeweils nicht mehr als 10^0 vom Ausgangswert abweichen. Um die invariante Masse des D*-Mesons zu berechnen, mußten die gefundenen Clusterzentren, den richtigen Teilchen durch Vergleich mit den fehlerfreien Aus-

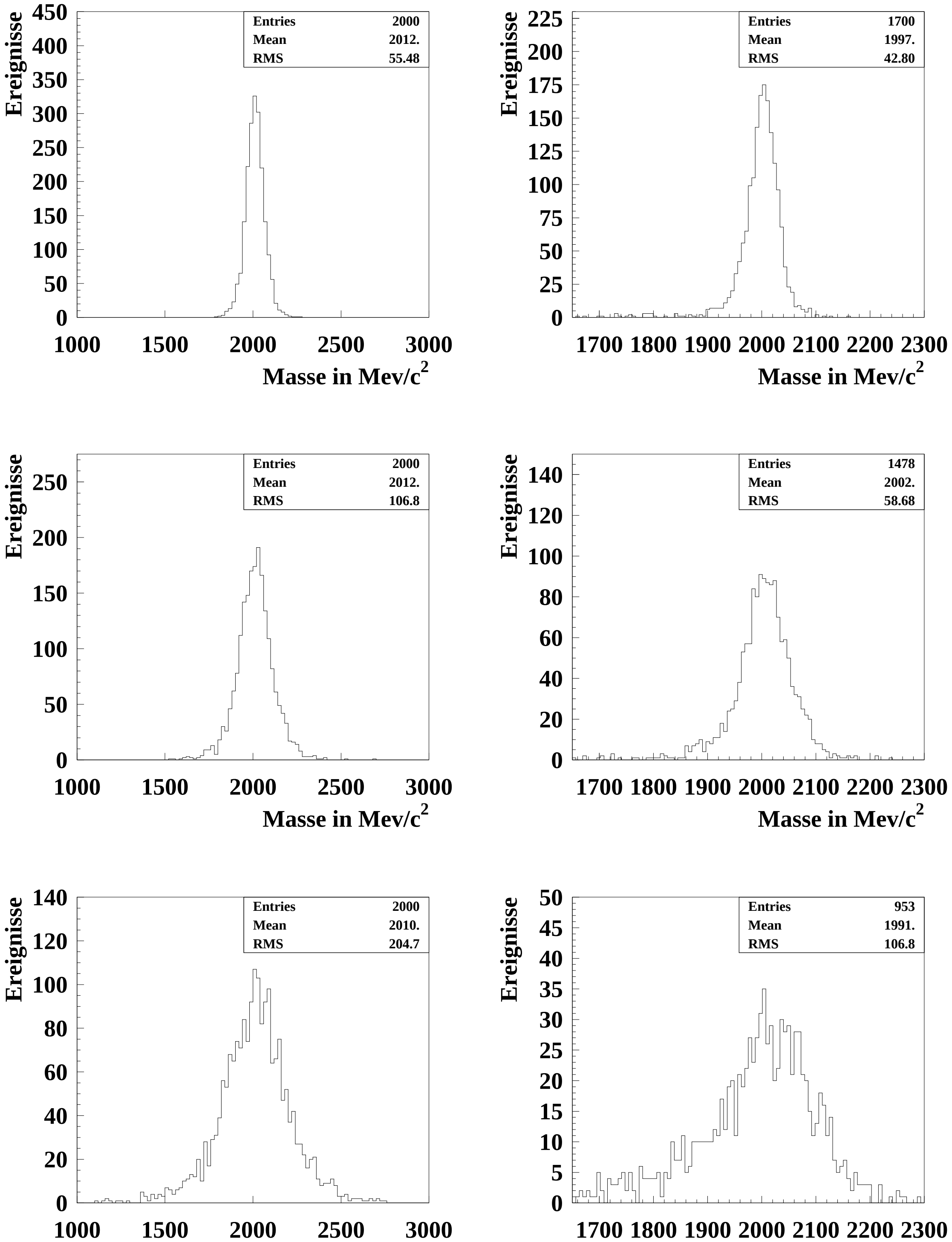


Abbildung 7.5: *Rechte Spalte: Die Ergebnisse für drei Trainings mit jeweils 15 Spurelementen pro Teilchen und $\sigma_{p_t}/p_t \approx 2.5\%$ (oben), 5% (Mitte) und 10%(unten) $GeV^{-1} \cdot p_t$ für die Spurelemente. Linke Spalte: Ergebnisse in der Rekonstruktion für 2000 D^* Zerfälle, bei denen die Spuren der Pionen und Kaonen mit dengleichen Fehlern behaftet sind.*

gangsdaten zugeordnet werden.
In Abb. 7.5 nehmen die Fehler der Spurelemente von oben nach unten zu. Wie man sieht, nimmt nicht nur der RMS Fehler der invarianten Masse mit schlechter werdender Impulsauflösung zu. Auch die in drei Epochen gefundenen Spuren nehmen ab. In der linken Spalte sieht man zum Vergleich eine Monte-Carlo-Simulation der Massenauflösung in der Rekonstruktion für 2000 D^*-Zerfälle, bei denen die Spuren der Pionen und Kaonen mit denselben Fehlern behaftet sind, wie die einzelnen Spurelemente, die zum Training verwendet wurden. Bei den Ereignissen, die bereits nach drei Epochen gefunden wurden, ergibt sich also eine Verbesserung der Massenauflösung.

Probleme
Das Hauptproblem der schlechten Impulsauflösung - verursacht durch die kleinen Spurelemente - besteht darin, daß die entstehenden Cluster so weit aufgefächert sind, daß der GNG Algoritmus mehrere Neuronen in ein Cluster legt, da er versucht die Neuronen möglichst gut an die Eingabedaten anzupassen. In Abb. 7.6 sieht man einige typische auftretende Probleme bei großen Fehlern der einzelnen Spurelemente. Die Cluster sind so weit ausgedehnt, daß mehrere Neuronen ein Cluster repräsentieren (Bilder oben). Der Impuls ist so groß, daß ein Cluster in P extrem verschmiert ist[1] (Bild unten links) oder die Clusterzentren sind nach drei Epochen noch nicht gefunden (Bild unten rechts).

7.6 Ergebnisse und Ausblick

Erste Untersuchungen mit dem „Growing Neural Gas“-Algorithmus zum Auffinden von Clustern sehen vielversprechend aus. Abhängig von den angenommenen Fehlern für die einzelnen Spurelemente werden die Clusterzentren bereits nach drei Epochen gefunden. Die Anzahl der richtig gefundenen Spuren (Clusterzentren) nimmt jedoch mit steigendem Spurfehler ab, außerdem werden mehrere Neuronen in einen Cluster gelegt, so daß man nicht wirklich einen einzigen Repräsentanten des Clusters, der das echte Teilchen repräsentiert, erhält.
Da die Anzahl der gefundenen Neuronen vom Parameter lowerCounterThreshold abhängig ist, wäre vorallem eine genauere Untersuchung einer Variation dieses Parameters in Abhängigkeit der im Mittel pro Spur präsentierten Tripel interessant. Im Training stellte es sich als vorteilhaft heraus den Lernparameter der Nachbarn des Gewinnerknotens sehr klein zu wählen. Es sollten daher auch andere selbstorganisierende Algorithmen untersucht werden, die ohne Nachbarschaftsbeziehung lernen. Da das anvisierte Problem des Online-Trackings sehr zeitkritisch ist, könnte dies auch Zeitvorteile bringen.
Der GNG Algorithmus startet mit zwei zufällig gewählten Neuronen, die im ungünstigen Fall weit von einem Cluster entfernt sind. Da die Dimension des betrachteten Problems fest ist, wäre es denkbar den Algorithmus mit einer großen Anzahl von Neuronen, die über den Eingaberaum (P,θ,ϕ) gleichverteilt sind, zu

[1] σ_{p_t} ist quadratisch von p_t abhängig

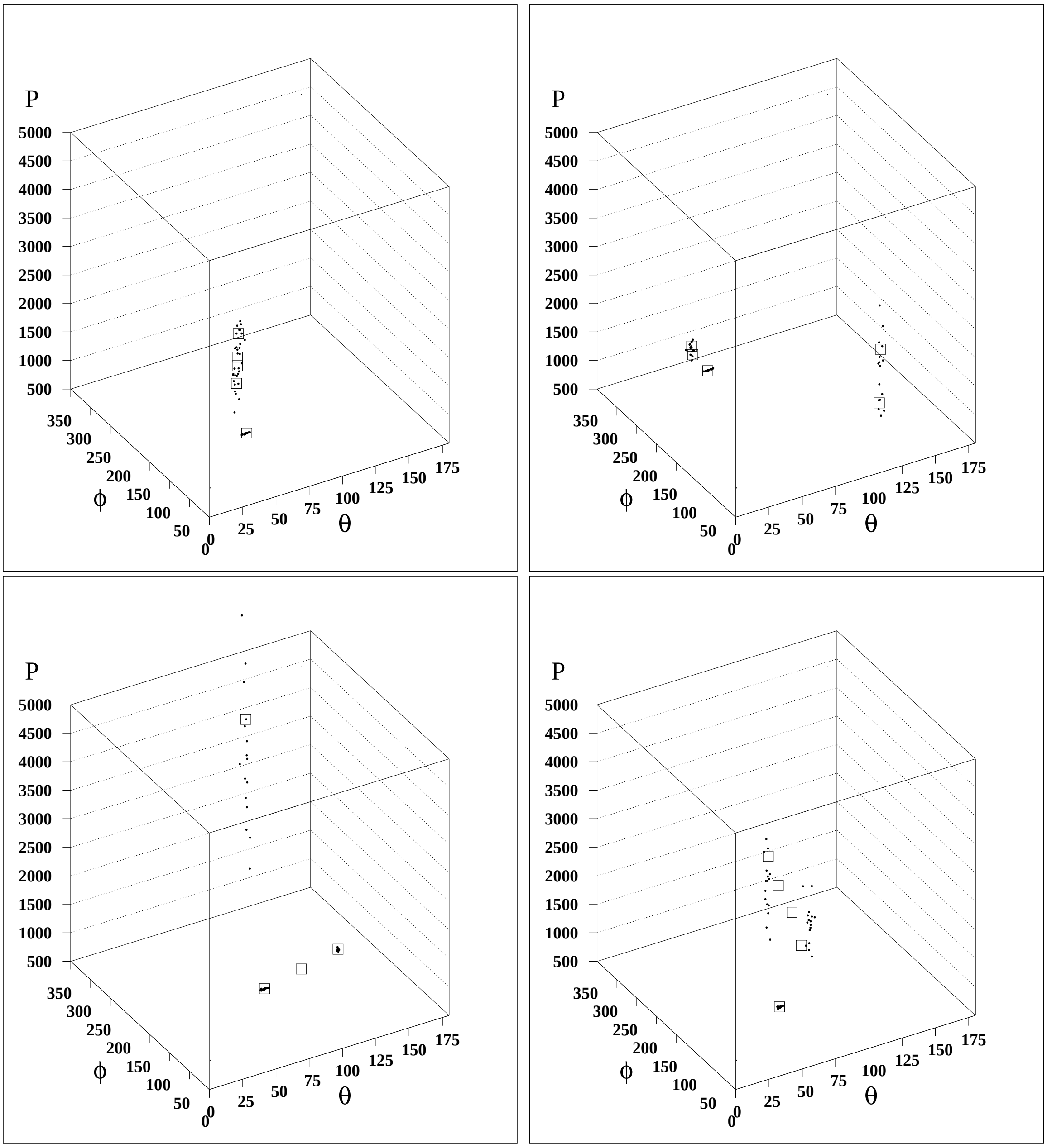


Abbildung 7.6: *Probleme, die beim Finden der Cluster bei großen Fehlern fü die einzelnen Spurelemente auftreten können. Oben links: Zwei Cluster, die nahe beieinander liegen und durch vier Neuronen repräsentiert werden. Oben rechts: Nach dem Training sind fünf Neuronen für die drei Spuren gefunden. Unten links: Ein sehr weit ausgedehnter Cluster, der durch den hohen Impuls und den daraus großen resultierenden Fehler der einzelnen Spurelemente zustande kommt. Unten rechts: Zwei weiter ausgedehnte Cluster. Die Clusterzentren sind nach drei Epochen noch nicht gefunden worden.*

starten. Irgendein Neuron ist dann bereits sehr nahe bei einem Cluster gelegen. Da alle Neuronen parallel ihre Abstände zu den präsentierten Eingabedaten berechnen, verliert man durch die große Anzahl der Startneuronen keine Rechenzeit. Die kaum trainierten Neuronen, die sich nicht in der Nähe eines Clusters befinden, werden bereits nach der ersten Epoche aussterben, während die übrigen, wegen des besseren Startpunkts die Clusterzentren schneller auffinden. Der Algorithmus wäre sozusagen ein „Shrinking Neural Gas“, bei dem am Ende nur noch die Neuronen in den Clusterzentren übrig bleiben.

In dieser Arbeit konnte noch kein Augenmerk auf den Rechenzeitverbrauch gelegt werden, da das verwendete Programm GNG nur auf einer sequentiell arbeitenden Workstation getestet wurde und nicht zeitoptimiert war. Für zukünftige Arbeiten ist diese Untersuchung allerdings notwendig, um zu erfahren ob der Einsatz eines neuronalen Netzes zum Online-Tracking auch den zeitkritischen Bedingungen auf Triggerebene standhält. Auch ist bisher nicht klar, wie der Algorithmus zur Bestimmung der Spurtripel aussehen soll. Es kann sein, daß sich dieses Problem als das schwieriger zu lösende erweist, da nicht feststeht wie groß die einzelnen Bereiche beim Bilden der Spurtripel gewählt werden sollen. Probleme mit kurzen Spuren und Spuren, die über mehrere Sektoren gehen, müssen dabei berücksichtigt werden.

Kapitel 8

Zusammenfassung

Für das Jahr 2001 ist beim H1-Experiment eine Verbesserung der Ausleseelektronik vorgesehen, die Spurpunkte aus den Driftkammern auf der zweiten Triggerstufe zur Verfügung stellt.

Ein Hauptpunkt der Arbeit war es, ein neuronales Netz darauf zu trainieren, mit Spurpunkten als Eingabe D^*-Mesonen zu triggern. Betrachtet wurde der Zerfallskanal $D^{*+} \rightarrow D^0\ \pi_s^+$; $D^0 \rightarrow K^-\ \pi^+$. Für das Training wurde der Zerfall eines fiktiven Teilchens mit variabler Ruhemasse in $K^-\ \pi^+$ betrachtet, wobei die Netze auf die invariante Masse des fiktiven Teilchens trainiert wurden. Die Daten für das Training wurden mit selbsterstellten Monte-Carlo-Simulationen erzeugt. Als neuronale Netzwerke wurden Feed-Forward-Netze mit drei bzw. vier Lagen ausgewählt. Das Training der Netzwerke wurde mit dem Backpropagation-Algorithmus auf dem CNAPS-Parallelcomputer durchgeführt. Es wurden sowohl die Anzahl der versteckten Schichten und Knoten als auch der Lernparameter variiert. Die besten Ergebnisse in der Parameterstudie wurden mit einem vierlagigen Netz mit der Knotenkombination 32/16 in den versteckten Schichten erzielt. Vierlagige Netze erwiesen sich in den Untersuchungen mit mehr als doppelt so guten Ergebnissen wesentlich leistungsfähiger als dreilagige Netze.
Weiterhin wurden Auswirkungen der Variation der Eingabegrößen untersucht. Getestet wurde sowohl die Auswirkung von Detektoreigenschaften, wie auch der Nutzen von Datenvorverarbeitung. Dabei zeigte sich, daß auch das neuronale Netz von grundlegenden Spureigenschaften, wie Hebelarm und Auflösung abhängig ist, und daß Datenvorverarbeitung, d.h. die Art und Weise wie dem Netz die Eingaben präsentiert werden, entscheidenden Einfluß auf die Qualität des Ergebnisses hat. Mit maximaler Datenvorverarbeitung, d.h. einer Präsentation der Spurpunkte mit Teilchen ID erreichte das Netzwerk für den trainierten Massenbereich von 1 GeV/c^2 bis 3 GeV/c^2 und den Impulsbereich 2 GeV/c bis 5 GeV/c bei 10000 Trainingsdaten eine Massenauflösung von ca. 140 MeV/c^2, verglichen mit ca. 35 MeV/c^2 in der Rekonstruktion eines D^0 unter denselben Bedingungen für Auflösung, Hebelarm und Trefferzahl. Es wurde auch der Zerfall eines fiktiven Teilchens in $D^0\ \pi_s^+$; $D^0 \rightarrow K^-\ \pi^+$ untersucht, wobei die invariante Masse nun aus drei Spuren rekonstruiert werden sollte. Bei vergleichbaren Wertebereichen für Masse und

Impuls des fiktiven Teilchens ergab sich bei gleicher Anzahl der Trainingsmuster mit ca. 230 MeV/c^2 ein schlechteres Ergebnis, das sich ohne Datenvorverarbeitung, d.h. bei radienweise, sortierter Eingabe der Trefferpunkte auf ca. 420 MeV/c^2 verschlechterte. Bei Erhöhung der Anzahl der Trainingsmuster auf den Maximalwert 65535, wurde im besten Fall (Eingabe mit ID) wieder ungefähr eine Auflösung von 140 MeV/c^2 erreicht. Es zeichnete sich jedoch ab, daß sich die Ergebnisse mit steigendem Schwierigkeitsgrad und realistischeren Daten (z.B. Störspuren, Rauschpunkte etc.) weiter verschlechtern würden und sich die Netze mit der verwendeten, einfachen Architektur ohne Datenvorverarbeitung somit nicht als Trigger einsetzen lassen können. Das Training auf dem CNAPS war jedoch durch eine maximale Zahl von vier Lagen und die begrenzte Rechengenauigkeit[1] limitiert. Eine Untersuchung des Problems mit komplizierteren Netzwerkarchitekturen und besserer Rechengenauigkeit, könnte eventuell bessere Ergebnisse liefern.

Ein Hauptergebnis der Arbeit ist die Tatsache, daß Datenvorverarbeitung bei neuronalen Netzen eine entscheidende Rolle spielt. Ein weiterer Bestandteil der Arbeit war daher der Test, ob sich Datenvorverarbeitung auch mit neuronalen Algorithmen realisieren läßt. Geprüft wurde, ob sich aus P-θ-ϕ-Spurelementen, die man zuvor mit einem Softwarealgorithmus aus jeweils drei Spurpunkten bilden könnte, die richtigen Spuren mit einem selbstorganisierenden Netzwerk finden lassen. Bei richtigen Spuren ergeben sich im dreidimensionalen Raum Anhäufungen (Cluster) der Spurelemente, die der getestete „Growing Neural Gas“-Algorithmus in einem Online-Training innerhalb weniger Epochen finden kann. Bisher ist allerdings noch nicht klar, wie der Algorithmus zum Bilden der Spurelemente aus den Trefferpunkten in der Driftkammer aussehen soll. Betrachtet man jedoch das Teilproblem des Clusterfindens mit neuronalen Netzen so sehen die ersten Ergebnisse vielversprechend aus. Weitere, genauere Untersuchungen bzgl. Spurmultiplizität, Trainingsdauer und Auflösung der zum Training verwendeten Spurelemente, sowie Tests anderer neuronaler Algorithmen sind zu empfehlen.

[1] Der CNAPS wird in Festkomma-Arithmetik betrieben, Eingabedaten können maximal 16-Bit Genauigkeit haben

Literaturverzeichnis

[1] Iliopoulos J. Glashow S. L. und Maiani L. *Phys. Rev.*, **D2**, p. 1285, 1970.

[2] Cabibbo N. *Phys. Rev.*, **10**, p. 531, 1963.

[3] Kobayashi M. und Maskawa T. *Prog. Theor. Phys.*, **49**, p. 652, 1973.

[4] Wins M. Produktionsmechanismen schwerer Quarks am H1-Experiment bei HERA. Diplomarbeit, RWTH Aachen, 1995.

[5] Ellis R. K. und Nason P. *Nucl. Phys.*, **B312**, p. 551, 1989.

[6] Nason P. Frixione S., Mangano M. L. und Ridolfi G. *Phys. Lett.*, **B348**, p. 633, 1995.

[7] Erdmann W. Untersuchung der Photoproduktion von D^*-Mesonen bei HERA. Dissertation, Eidgenösische Technische Hochschule Zürich, 1996.

[8] Kathage Ulf. Photoproduktion von D^{*+}-Mesonen bei HERA. Dissertation, Ecole Polytechnique, Paris, 1997.

[9] C.F. von Weizsäcker. Ausstrahlung bei Stößen sehr schneller Elektronen. *Z. Phys.*, **88**, p. 612, 1934.

[10] E.J. Williams. Nature of the high energy particles of penetrating radiation and status of ionisation and radiation formulae. *Phys. Rev.*, **45**, p. 729, 1934.

[11] Stefano Frixione, Michelangelo L. Mangano, Paolo Nason, und Giovanni Ridolfi. Improving the Weizsäcker-Williams approximation in electron - proton collisions. *Phys. Lett.*, **B319**, pp. 339–345, 1993. hep-ph/9310350.

[12] P. Avery et al. CLEO Collaboration. *Phys. Rev.*, **D43**, p. 3599, 1991.

[13] Particle Data Group. *Phys. Rev.*, **D50**, p. 1173, 1994.

[14] G.J. Feldmann et al. *Physical Review Letters*, **38**, p. 1313, 1977.

[15] H1 Kollaboration, 1997. `http://www-h1.desy.de/`.

[16] ZEUS Kollaboration, 1997. `http://www-zeus.desy.de/`.

[17] H1 Kollaboration, Luminositätstabellen von HERA aus den Jahren 92 bis 97, 1998. `http://www-h1.desy.de/ levonian/luminosity/summary_tables`.

[18] H1 Kollaboration, . ep Physics beyond 1999. DESY Report DESY H1-97, DESY, 1997.

[19] H1 Kollaboration, I. Abt et al. The H1 detector at HERA. *Hamburg DESY - Int.Rep. H1-96-01 158 p., Update of Nucl. Instr. and Meth.A. 386, pp310*, 1997. `http://www-h1.desy.de/h1/www/h1det/detpaper/contents.html`.

[20] D. Handschuh. Studien zum Protoninduzierten Untergrund am H1 Detektor und Entwicklung von Filteralgorithmen zu dessen Unterdrückung. Dissertation, Universität Hamburg, 1993.

[21] Arnd Meyer. Nachweis von Myonen unter kleinem Polarwinkel im H1-Detektor bei HERA. Diplomarbeit, Universität Hamburg, II. Institut für Experimentalphysik, Februar 1995.

[22] J. Möck et al., 1996. `http:/wwwh1.mppmu.mpg.de/projects/neuro`.

[23] J.C. Bizot et al. Status of Simulation for a topological level 2 trigger. H1–Note H1-02/92-212, DESY, Notkestr. 85, 22607 Hamburg, 1992.

[24] Krogh A. Hertz J. und Palmer R.G. Introduction to the Theory of Neural Computation. Addison Wesley Publishing Company, Redwood City, CA., 1991.

[25] Spektrum der Wissenschaft. Gehirn und Nervensysteme. Spektrum der Wissenschaft Verlag, Heidelberg, 10. Aufl., 1992.

[26] J. W. Cohen. Funktionelle Annatomie des Nervensystems. Schattauer, Stuttgart/New York, 4. Aufl., 1985.

[27] J. C. Eccles. The Understanding of the Brain. McGraw-Hill, New York, 1977.

[28] W. S. McCulloch und W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Math. Bio.*, **5**, pp. 115–133, 1943.

[29] Hopfield J.J. Neural networks and physical systems with emergent collective computational abilities. *Procs. Nat. Acad. Sc.*, **79**, p. 2554, 1982.

[30] Kohonen T. Self-organization and Associative Memory. Springer-Verlag, Berlin, 3. Aufl., 1989.

[31] Fritzke B. Growing Cell Structures - a self-organizing network for supervised and unsupervised learning. *Neural Networks 1994*, **7**, pp. 1441–1460, 1994.

[32] Fritzke B. A Growing Neural Gas Network learns Topologies. *Advances in Neural Information Processing Systems*, **7**, 1994.

[33] Martinetz T. Ritter H. und Schulten K. Neural Networks. Addison Wesley, 3. Aufl., 1992.

[34] R. Rojas. Theorie der neuronalen Netze. Springer-Verlag, 1993.

[35] A. N. Kolmogorov. On the Representation of Continous Functions of several Variables by Superposition of One Variable and Addition. *Dokl.Akad.Nauk. USSR*, **115**, pp. 953–956, 1957.

[36] R. Hecht-Nielsen. Kolmogorov's Mapping Neural Network Existence Theorem. In *First Annual Int. Conf. on Neural Networks*, volume III. IEEE, 1987.

[37] Hecht-Nielsen. Neurocomputing. Addison-Wesley, 1991.

[38] R.J.P. de Figueiredo. Implications and Applications of Kolmogorov's Superposition Theorem. *Math. Anal*, **38**, p. 1227, 1980.

[39] P. J. Werbos. Beyond Regression. Appl. math. , Harvard University, November 1974.

[40] D. E. Rumelhart und J. L. McClelland. Parallel Distributed Processing : Foundations, volume 1. MIT Press Cambridge, 1986. Im wesentlichen p318-362.

[41] E.Elsen A.Campbell und W.Haynes. Track based triggering for H1 using the Central Drift Chamber. H1 report, H1, 1997.

[42] L. F. A. Wessels und E. Barnard. Avoiding False Local Minima by Proper Initialsiation of Connections. *IEEE Transactions on Neural Networks*, **3(6)**, pp. 899–905, 1992.

[43] J. Möck. Einsatz neuronaler Netze als intelligente Trigger im H1–Experiment. Diplomarbeit, Technische Universität München, März 1994. `http://wwwh1.mppmu.mpg.de/personal/jum/dip/dip.html` .

[44] Udluft S. Untersuchungen zu neuronalen Netzen als Vertextrigger beim H1-Experiment. Diplomarbeit, LMU München, 1996.

[45] Adaptive Solutions Inc., Beaverton, OR 97006. CNAPS Data Book, 2.0 Version, März 1995. Siehe auch `http://www.asi.com/asi/manuals.html`.

[46] Martin Schied. Persönliche Mitteilung, 1998.

[47] Glückstern R.L. Uncertainties in track momentum and direction due to multiple scattering and measurement errors. *Nuclear Instruments and Methods*, **24**, pp. 381–389, 1963.

[48] Ohlsson M. und Peterson C. Track Finding with Deformable Templates - The Elastic Arms Approach. *Computer Physics Communications*, **71**, pp. 77–98, 1992.

[49] Ohlsson M. Extensions and Exploration of the Elastic Arms Algorithm. *Computer Physics Communications*, **77**, pp. 19–32, 1993.

[50] M. Kunze J. Steffens. Growing Cell Structure and Neural Gas. *Proceedings of the 4th AIHEP Workshop, Pisa, World Scientific*, 1995.

[51] Fritzke B. Growing Self Organizing Networks - Why. *ESANN'96 European Symposium on Artificial Neural Networks*, , pp. 61–72, 1996.

[52] Martinetz T. M. Competetive Hebbian learning rule forms perfectly topology preserving maps. *ICANN'93 International Conference on Artificial Neural Networks*, , pp. 427–434, 1993.

[53] Janauschek L. Untersuchungen zur Untergrundreduktion mit einem neuronalen Trigger. Diplomarbeit, LMU München, 1998.

Danksagung

Diese Arbeit wurde am Max-Planck-Institut für Physik in München angefertigt. An dieser Stelle möchte ich mich bei allen Mitgliedern der H1-Gruppe für die Hilfsbereitschaft und die angenehme Arbeitsatmosphäre bedanken.

Prof. Dr. Christian Kiesling danke ich für viele hilfreiche Ratschläge und wertvolle Diskussionen. Dr. Armin Wegner danke ich vorallem für die Tips bzgl. der Monte-Carlo-Simulation der Massenrekonstruktion. Dank auch an Burkard Reisert, Markus Enzensberger und besonders Steffen Udluft für die sehr gute Betreuung und die vielen anregenden Diskussionen über neuronale Netzwerke und Probleme physikalischer Natur. Martin Schied sei herzlich für die gemeinsame Programmierarbeit an den Programmen zur Simulation des Zerfalls und des Detektors gedankt. Ludger Janauschek danke ich für die Bereitstellung und Erläuterung seines Programms GNG.
Allen Anderen, die zum Gelingen dieser Arbeit beitrugen, danke ich für ihre Unterstützung und ihre Anregungen.

Der Max-Planck Gesellschaft, insbesondere Prof. Dr. Gerd Buschhorn danke ich für die Bereitstellung der erforderlichen Arbeitsmittel.

Erklärung

Hiermit versichere ich, die vorliegende Arbeit unter Angabe aller wesentlichen Quellen und Hilfsmittel selbständig verfaßt zu haben.

München, Juli 1998

(Philip Scheideler)