Untersuchungen zum Trigger von ϕ-Mesonen mit neuronalen Netzen bei HERA

Diplomarbeit an der Fakultät für Physik der
Ludwigs-Maximilians-Universität

vorgelegt von

Markus Eberl

angefertigt am

Max-Planck-Institut für Physik

– Werner-Heisenberg-Institut –

München

März 1999

Erstgutachter: Prof. Dr. Christian Kiesling

Zweitgutachter: Prof. Dr. Dorothee Schaile

Inhalt

1 Die Physik bei HERA

1.1 Die Farben der Quarks

Die Theorie der starken Wechselwirkung, die Quantenchromodynamik (QCD), wurde in Analogie zur ausgesprochen erfolgreichen Quantenelektrodynamik (QED) als renormierbare Eichtheorie konstruiert. Sie beschreibt die Wechselwirkung zwischen stark wechselwirkenden Teilchen durch den Austausch von Eichbosonen, den Gluonen. Bevor es soweit kommen konnte, waren allerdings einige Schritte nötig. Den ersten großen Schritt taten Gell-Mann [3] mit dem Quarkmodell.

1.1.1 Das Quarkmodell

Inspiriert vom klassischen Isospinformalismus und fasziniert von Symmetrien in der Physik, schuf Gell-Mann ein Modell, welches – analog der SU(2) Symmetrie des Isospins bei Neutronen und Protonen – die $SU(3)_{Flavour}$ Symmetrie im Quark-Sektor einführte. Grundlage dieser Symmetrie war die Tatsache, daß die drei damals bekannten Quarks *u*, *d* und *s* nur verschiedene Geschmacksrichtungen (*Flavours*) ein und desselben Zustandes seien. Diese Symmetrie ist allerdings – wie auch die Isospinsymmetrie bei Neutron und Proton – nicht perfekt, sondern durch die unterschiedlichen Massen der Quarks gebrochen. Tabelle 1.1 zeigt einen Überblick über die Eigenschaften der Quarks.

Tabelle 1.1: Jedes Quark trägt den Spin 1/2 und die Baryonzahl 1/3. Der SU(2)-Isospinformalismus für u- (I_z=-1/2) und d-Quarks (I_z=1/2) wurde mit Entdeckung des s-Quarks (S=-1) zur SU(3)-Flavour-Symmetrie erweitert.[1]

	Flavour bzw.Isospin	*Masse (*[49]*)*	*Ladung*
u	I_z=-1/2	2-8 MeV	+2/3
d	I_z=1/2	5-15 MeV	-1/3
s	S=-1	100-300 MeV	-1/3

[1] Hier gilt die Konvention der Particel Data Group, daß *Flavour* und *Ladung* das gleiche Vorzeichen tragen.

Innerhalb dieses Modells lassen sich nun die Vielzahl der stark gebundenen Zustände zu sogenannten Multipletts ordnen. Die Baryonen – gebundene Zustände aus jeweils drei Quarks – lassen sich nach der Gruppenidentität $3 \otimes 3 \otimes 3 = 10 \oplus 8 \oplus 8 \oplus 1$ in ein Decuplett, zwei Oktetts und ein Singlett einordnen. Mesonen – gebundene Zustände aus einem Quark und einem Antiquark – bilden nach $3 \otimes \overline{3} = 8 \oplus 1$ ein Oktett und ein Singlett. Trägt man nun jedes Meson in ein Diagramm ein, dessen horizontale Achse die dritte Komponente des Isospins (I_z) und dessen vertikale Achse die Seltsamkeit (S) ist, so entsteht eine Darstellung der möglichen Quark-Antiquark Kombinationen.

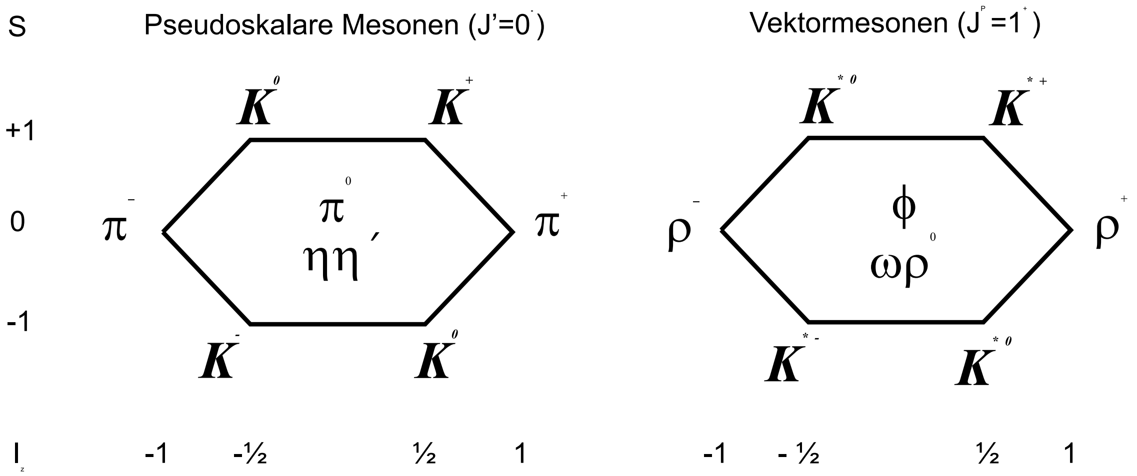

Bild 1.1: Multipletts der pseudoskalaren Mesonen (links) und Vektormesonen (rechts). Der I=0 Grundzustand des Oktetts mischt dabei mit dem entsprechenden I=0 Zustandes des Singletts zu η und η' (pseudoskalare Mesonen) bzw. ϕ und ω (Vektormesonen).

Neben dem rein ästhetischen Wert, den die Ordnung des Teilchenzoos mit sich bringt, hat das Modell auch noch einen weiteren Nutzen. Findet sich ein noch unbesetzter Platz in einem Multiplett so liegt die Vermutung nahe daß dieser durch ein noch unbekanntes Teilchen zu besetzen ist. Das Modell erlaubt also Vorhersagen über das Spektrum der Elementarteilchen. Auch Verzweigungsverhältnisse beim Zerfall der Mesonen lassen sich aus Symmetrieüberlegungen ableiten.

1.1.2 Farbe: die Ladung der starken Wechselwirkung

In der Quantenelektrodynamik sind die Kopplungen zwischen Fermionen und Eichbosonen proportional zur Elementarladung. Die analoge Einführung einer einzelnen Ladung, die proportional zur Kopplungsstärke in der QCD ist, führt jedoch zu experimentellen Widersprüchen. Fermi entdeckte 1951 in der $\pi^+ p$-Streuung eine zweifach positiv geladene Resonanz, das Δ^{++} mit einem Spin J=3/2. Sowohl Spin, als auch Ladung dieses neuen Teilchens lassen sich im Rahmen von $SU(3)_{Flavour}$ durch einen Quarkinhalt von $\left|\Delta^{++}\right\rangle = \left|uuu\right\rangle$ erklären.

Da jedes u-Quark Spin 1/2 und Ladung 2/3 beiträgt, wäre die Δ^{++} Resonanz einfach der baryonische Grundzustand dieser drei Quarks. Eine solche Wellenfunktion, aus drei

Fermionen, wäre jedoch symmetrisch, was die Fermistatistik verbietet. Die Lösung dieses Dilemmas liegt in der Einführung neuer Quantenzahlen, der Farbladungen. Jedes u-Quark bekommt einen eigene Farbladung, und wird so von den anderen beiden unterscheidbar. Die Fermistatistik ist gerettet.

Die Einführung verschiedener Ladungen löst zwar das Δ^{++} Rätsel, macht die zugehörige Theorie aber auch ein wenig komplizierter. Reichte in der QED als Eichgruppe noch die abelsche $U(1)$ der lokalen phasentransformationen $\Psi \rightarrow e^{i\alpha(x)}\Psi$, so benötigt man im Falle der QCD bereits die nichtabelsche $SU(3)_{Color}$.

Um die Invarianz der Lagrangedichte unter der zugehörigen lokalen Eichtransformation zu erhalten, ist es nötig, in die Lagrangedichte einen zusätzlichen Term einzuführen. Dieser weitere Term beschreibt die Selbstwechselwirkung der QCD-Eichbosonen – der Gluonen – untereinander. Im Gegensatz zu den Photonen der QED, in der es diese Selbstwechselwirkung nicht gibt, tragen die Eichbosonen der QCD selbst Farbladungen. Die Eichbosonen der starken Wechselwirkung nennt man Gluonen, da sie wie Kleber (Glue) die Hadronen zusammenhalten.

Im Gegensatz zu gewöhnlichem Kleber, führt bei den Gluonen deren Selbstwechselwirkung jedoch dazu, daß die Anziehung zwischen den einzelnen Teilchen zunimmt je größer ihr Abstand wird. Hier ist wieder der, im Vergleich zur QED, nichtabelsche Charakter der QCD zu sehen, der bei der Renormierung der Theorie zu fundamentalen Unterschieden führt. Im Gegensatz zu Quantenelektrodynamik, bei der die Kopplungskonstante α mit steigenden Impulsüberträgen ebenfalls zunimmt, verhält es sich bei der Kopplungskonstante α_s der starken Wechselwirkung genau umgekehrt. Sie ist groß (confinement) bei niedrigen Impulsüberträgen, und wird erst bei großem Impuls klein. Störungstheoretische Ansätze zur Berechnung von Wirkungsquerschnitten greifen also erst im Bereich hoher Impulsüberträge (typisch >5 GeV^2). Eine der Methoden um mehr über die starke Wechselwirkung zu lernen, ist die tiefinelasische Elektron-Proton-Streuung.

1.2 Vektormesonen bei HERA

Mesonen, also gebundene Zustände aus Quarks und Antiquarks, bieten neben tiefinelastischer Elektron-Proton-Streuung, die Möglichkeit, die starke Wechselwirkung zu studieren. Insbesondere die hadronische Natur des Photons, kann bei der Produktion von Vektormesonen, die wie das Photon Spin 1 haben, untersucht werden. Gerade die Tatsache, daß Vektormesonen in ihren Quantenzahlen mit dem Photon übereinstimmen, ermöglicht ein Mischen dieser beiden Zustände, und macht das Studium dieser Mesonen besonders interessant.

Diese Vektormesonen entstehen durch reelle oder virtuelle Photonen, die wiederum bei der Positron-Proton-Streuung anfallen. Dieser Produktionsmechanismus wird besonders dann interessant, wenn die Mesonen schwerer Quarks wie das ϕ oder J/Ψ produziert werden [2][15]. In diesem Massenbereich findet der Übergang von nichtperturbativer QCD in den perturbativen Bereich statt. Es eröffnet sich so die Möglichkeit, zur Beschreibung experimenteller Daten, nicht nur phänomänologische Modelle zu verwenden, wie die Regge-Theorie ([4][5]) oder das Vektormesonen-Dominanzmodell ([6][7]), sondern auch störungstheoretische Ansätze der QCD ([8]).

1.2.1 Eigenschaften leichter Vektormesonen

Im folgenden sollen die Eigenschaften der Vektormesonen ρ^0, ω und ϕ näher betrachtet werden. Alle drei Mesonen bestehen im Rahmen des Quarkmodells aus Überlagerungen von $u\bar{u}$, $d\bar{d}$ und $s\bar{s}$.

$$
\begin{aligned}
&|\omega\rangle \approx |u\bar{u}\rangle + |d\bar{d}\rangle \\
&|\rho^0\rangle \approx |u\bar{u}\rangle - |d\bar{d}\rangle \\
&|J/\Psi\rangle \approx |c\bar{c}\rangle \\
&|\phi\rangle \approx |s\bar{s}\rangle
\end{aligned}
$$

Dabei sind sowohl eventuelle Beimischungen von $s\bar{s}$ und $u\bar{d}$ zu den ρ^0 und ω Mesonen als auch $u\bar{u}$ und $d\bar{d}$ Beimischungen zum ϕ Meson gering [40]. Dieser Quarkinhalt zeigt sich auch in der Zerfallscharakteristik der Vektormesonen. Das ρ^0 Meson zerfällt nahezu ausschließlich in Pionen, die ebenfalls aus u und d Quarks und deren ladungskonjugierten Partnern aufgebaut sind. Gleiches gilt auch für das ω Meson, das ebenfalls hauptsächlich in Pionen zerfällt. Der Zerfall des ϕ Meson wird im folgenden noch genauer besprochen. Tabelle 1.2 faßt noch einmal die wichtigsten Quantenzahlen und den Quarkinhalt der Vektormesonen zusammen [40].

Tabelle 1.2: Übersicht der Vektormesonen

	J^{PC}	*Masse [MeV]*	*Quarkinhalt*	*Isospin I*
ρ	1^{--}	770	$u\bar{d}$, $u\bar{u}$, $d\bar{d}$	I=1
ω	1^{--}	782	$u\bar{u}$, $d\bar{d}$, $s\bar{s}$	I=0
ϕ	1^{--}	1019	$s\bar{s}$	I=0
J/Ψ	1^{--}	3097	$c\bar{c}$	I=0

Das ϕ-Meson eignet sich aufgrund seiner Masse von 1,02 GeV gut zur Untersuchung der starken Wechselwirkung im Übergangsbereich vom nichtperturbativen zum perturbativen Bereich der QCD. Das ϕ-Meson ist eine relative schmale Resonanz mit einer Breite von G=4,43 MeV.

Aufgrund seines Aufbaus aus seltsamen Quarks zerfällt das ϕ-Meson hauptsächlich in Kaonen. In Tabelle 1.3: sind die dominanten Zerfallskanäle des ϕ-Mesons aufgelistet:

Tabelle 1.3: Die dominanten Zerfallskanäle des ϕ-Mesons.

Zerfall	*Anteil*
$K^+ K^-$	49%
$K_L^0 K_S^0$	34 %
$\rho\pi + \pi^+\pi^-\pi^0$	16%

Die geladenen Kaonen zerfallen wiederum hauptsächlich in Myonen und ein Myon-Neutrino (64%), oder in ein geladenes und ein neutrales Pion (21%). Das neutrale Pion zerfällt anschließende in zwei Photonen. Das kurzlebige, neutrale Kaon zerfällt entweder in zwei geladene oder zwei neutrale Pionen. Die Zerfallsprodukte des langlebigen Kaons sind entweder

> drei neutrale Pionen (21%), zwei geladene und eine ungeladenes Pion (13%), ein geladenes Pion, ein Myon und ein Myon-Neutrino (27%)

oder

> ein geladenes Pion, ein Elektron (Positron) und das entsprechende Neutrino (39%).

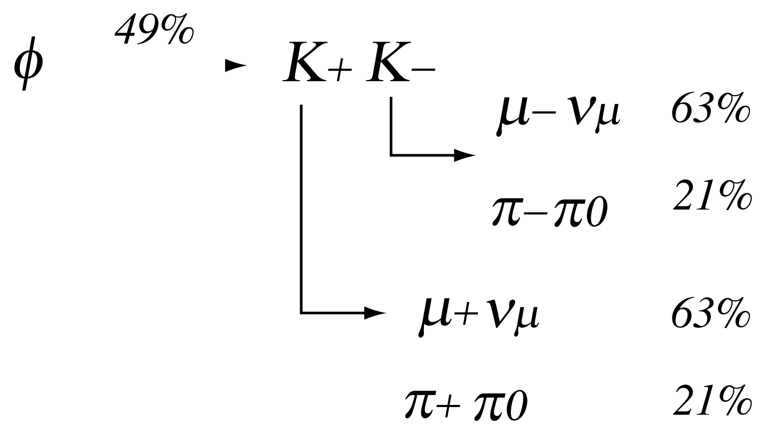

Bild 1.2: Der Hauptzerfallskanal des ϕ Mesons.

1.2.2 Kinematik der Elektron-Proton-Streuung

Vektormesonen werden bei HERA durch den Prozeß $e^+ p \rightarrow e^+ X$ erzeugt. Im folgenden soll k den Viererimpuls der einlaufenden Elektronen bezeichnen, k' den der auslaufenden und P denjenigen der einlaufenden Protonen.

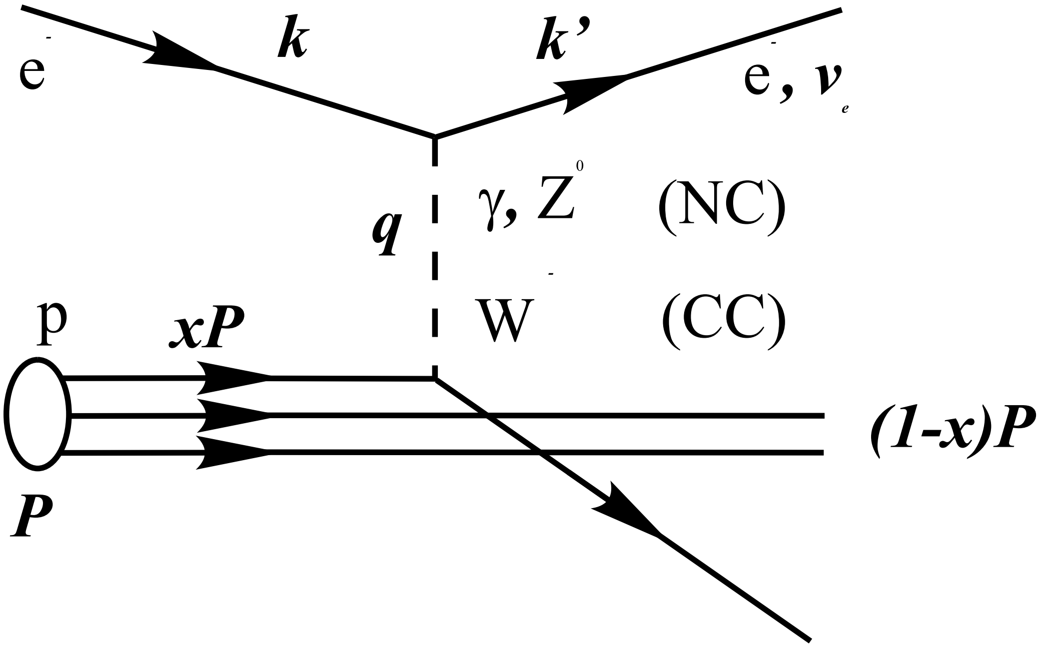

Bild 1.3: Kinematik der Positron-Proton-Streuung. Das Elektron streut am Proton und tauscht dabei entweder ein geladenes W-Boson (Charged Current CC), ein ungeladenes Z^0 oder ein Photon aus (Neutral Current NC). Die Viererimpulse der beteiligten Teilchen werden hier mit k, k', P bezeichnet.

Die Mandelstamm-Variable s, die dem Quadrat der Schwerpunktsenergie $E^2_{C.M}$ entspricht, läßt sich dann schreiben als:

$$E^2_{C.M.} = s = (\mathbf{k} + \mathbf{P})^2 = m_e^2 + m_P^2 + 4E_eE_P \tag{1.1}$$

Hier bezeichnen m_e die Elektronenmasse, und m_P die Masse des Protons. Für eine Elektron-Strahlenergie von E_e=27.5 GeV und eine Proton-Strahlenergie von E_P=820 GeV ergibt sich für die im Schwerpunkt zur Verfügung stehende Energie:

$$E_{C.M.} = \sqrt{s} \approx 300 \ \ GeV \tag{1.2}$$

Der Viererimpuls des Photons $\mathbf{q}$ ergibt sich aus der Differenz der Positronimpulses vor und nach der Streuung

$$\mathbf{q} = \mathbf{k} - \mathbf{k}' \tag{1.3}$$

Das negative Viererimpulsquadrat des Photons wird auch als Photonvirtualität bezeichnet, da das Quadrat eines Viererimpulses die Ruhemasse des entsprechenden Teilchen ergibt, und das Photon im Grenzfall $Q^2 \approx 0 \ \ GeV$ folglich quasi reell wird.

$$Q^2 = -\mathbf{q}^2 = Q^2_{\min} - 4p_e p_e' \sin(\theta_2 / 2) \tag{1.4}$$

Mit der minimalen, kinematisch erlaubten Q^2:

$$Q^2_{\min} = 2(E_e E_e' - p_e p_e' - m_e^2) \tag{1.5}$$

Hierbei bezeichnen E_e, p_e und E_e', p_e' die Energien und Impulse des Positrons vor bzw. nach der Streuung, und θ_e den Streuwinkel des Positrons gegenüber der ursprünglichen Flugrichtung. Weitere, zur Beschreibung der Kinematik, hilfreiche Größen sind die beiden Bjorken-Scaling Parameter x und y:

$$x = \frac{Q^2}{2\ \mathbf{P}\cdot\mathbf{q}}\ ,\ y = \frac{\mathbf{P}\cdot\mathbf{q}}{\mathbf{P}\cdot\mathbf{k}} \approx 1 - \frac{E'_e}{E_e}\cos^2(\theta_e/2) \tag{1.6}$$

Mit deren Hilfe läßt sich der Impulsübertrag im Grenzfall verschwindender Massen schreiben:

$$Q^2 \approx sxy \tag{1.7}$$

Neben diesen kinematischen Variablen gibt es noch zwei weitere lorentz-invariante Größen, die zur Beschreibung der Streuprozesse hilfreich sind. Zum einen ist dies der Betrag der Photon-Energie im Ruhesystems des Protons

$$\upsilon = \frac{\mathbf{P}\cdot\mathbf{q}}{m_P} = \frac{Q^2}{2xm_P} \tag{1.8}$$

und das Quadrat der Schwerpunktsenergie des Photon-Proton-Systems

$$W^2 = (\mathbf{q}+\mathbf{P})^2 = -Q^2 + 2\nu m_P + m_P^2 \approx Q^2\frac{1-x}{x} \tag{1.9}$$

Je nach Größe des Impulsübertrages Q^2, unterscheidet man zwei verschiedene Ereignisklassen. Der Bereich $Q^2 \approx 0$, in dem das ausgetauschte Photon quasi reell ist, wird als *Photoproduktion* bezeichnet. Ist $Q^2 >> 0$ liegt *tiefinelastische Streuung*[2] vor. Da das gestreute Positron bei sehr geringen Impulsüberträgen vom Detektor nicht mehr gesehen wird, muß eine endliche Energieschwelle festgelegt werden ab der DIS beginnt und Photoproduktion endet. Diese Schwelle liegt bei Q^2 = 1 GeV, da genau ab einem Impulsübertrag dieser Größe das Positron vom rückwertigen Kalorimeter (SPACAL, s. Kapitel 2.2.3.2) gesehen wird.

1.2.3 Tiefinelastische Elektron-Proton-Streuung

Ein experimentelles Verfahren um die starke Wechselwirkung bei großem Q^2 zu studieren ist tiefinelastische Elektron-Proton-Streuung (DIS, Deep Inelastic Scattering). Dabei werden die bei der Streuung entstehenden Teilchen nachgewiesen, um das Streuereignis möglichst vollständig rekonstruieren zu können. Das dabei zur Beschreibung des Wirkungsquerschnittes verwendete Modell beruht auf der Beschreibung des Protons durch Strukturfunktionen. Die grundlegende Idee dieser Funktionen ist es, die Abweichung des Wirkungsquerschnittes vom einem punktförmig gedachten Proton zu beschreiben ([9][10][11]). Betrachtet man nur Ereignisse, bei denen ein ungeladenes Boson zwischen Proton und Elektron ausgetauscht wird (Neutral Current), so läßt sich der totale Wirkungsquerschnitt – in der Näherung des Ein-Photon-Austausches – schreiben als:

[2] Deep Inelastic Scattering (DIS)

$$\frac{d^2\sigma}{dxdQ^2} = \frac{2\pi\alpha^2}{xQ^4}\left[\left(1+(1-y)^2\right)F_2(x,Q^2) - y^2 F_L(x,Q^2)\right] \qquad (1.10)$$

Die Messung dieser Strukturfunktionen F_2 und F_L ist eine der Hauptaufgaben des H1 Experimentes. Neben der Bestimmung der Proton-Strukturfunktionen, wird bei H1 auch die Produktion von Vektormesonen näher untersucht.

1.2.4 Photoproduktion

Bei der Photoproduktion von Vektormesonen ist der Impulsübertrag vom Elektron auf das Proton sehr klein, so das in guter Näherung $Q^2 \approx 0\,GeV^2$ gilt. Dies impliziert, daß das gestreute Elektron seine Richtung kaum ändert, und so nicht mehr nachgewiesen werden kann. Es hat nur eine verschwindend kleine transversale Impulskomponente und verbleibt daher nahe der Strahlachse außerhalb des sensitiven Bereichs des H1-Detektors.

Bei H1 liegt die Nachweisgrenze im rückwertigen Kalorimeter bei etwa $Q^2 <\approx 1\,GeV$ [15]. Bei kleinen Impulsüberträgen kann das Photon auch als quasi-reell betrachtet werden. Für quasi-reelle Photonen läßt sich der ep-Wirkungsquerschnitt in eine Komponente für die Emission des Photons $f_{\gamma/e}(Q^2, y)$ und den Photon-Proton Wirkungsquerschnitt $\sigma_{\gamma p}(Q^2, y)$ faktorisieren.

$$\frac{d^2\sigma_{ep}(Q^2,y)}{dy\ dQ^2} = f_{\gamma/e}(Q^2, y)\ \ \sigma_{\gamma p}(Q^2, y) \qquad (1.11)$$

Der Faktor für die Photon-Emission läßt sich wiederum in einen Anteil für longitudinal polarisierte Elektronen und einen für transversal polarisierte aufspalten. Da reelle Photonen nur transversale Polarisations-Freiheitsgrade besitzen, kann der longitudinale Anteil in guter Näherung vernachlässigt werden ([15] S. 21-23).

$$f^T_{\gamma/e}(Q^2, y) = \frac{\alpha}{2\pi Q^2}\left(\left(1+(1-y)^2\right) - 2(1-y)\frac{Q^2_{mm}}{Q^2}\right) \qquad (1.12)$$

mit

$$Q^2_{mm}(y) = m_e^2\frac{y^2}{(1-y)} \qquad (1.13)$$

Neben dem bekannten Term der Weizäcker-Williams-Approximation $\left(1+(1-y)^2\right)$ [12], enthält diese Formel auch noch einen Term, der die kleine, aber nicht verschwindende, Elektronenmasse berücksichtigt.

Wie aus Gleichung 1.12 ersichtlich, ist der Wirkungsquerschnitt proportional zu $1/Q^2$. Das bedeutet, daß mit abnehmender Photonvirtualität die Anzahl der Ereignisse zunimmt. Gleichzeitig kann für solche Ereignisse das Elektron aber nicht nachgewiesen werden. Diese Tatsache kann gezielt dazu verwendet werden, auf Photoproduktions-

Ereignisse zu triggeren. Bei Photoproduktionsereignissen wird im Gegensatz zu tiefinelastischen Ereignissen gefordert, daß das Positron vom Detektor *nicht* gesehen wird.

1.2.5 Diffraktive Erzeugung

Etwa 40% aller Photoproduktionsereignisse sind sogenannte diffraktive Ereignisse. Von diffraktiver Erzeugung von Vektormesonen spricht man, wenn in einem bestimmten Polarwinkelbereich um das auslaufenden Proton keine Ereignisse im Detektor nachgewiesen werden. Dieses Nachweisdefizit wird durch den Austausch eines Teilchens, des Pomerons, erklärt. Dieses Teilchen ist farbneutral. Der Feynmangraph des dabei stattfindenden Prozesses ist in Bild 1.4 dargestellt.

Das vom Elektron abgestrahlte virtuelle Photon (γ^*) fluktuiert in ein virtuelles Vektormeson (V^*). Durch starke Wechselwirkung mit dem Proton, bzw. mit den darin dessen Konstituenten, den Quarks, wird dieses virtuelle Vektormeson reell. Die starke Wechselwirkung des Vektormesons mit dem Proton kann dabei die Struktur des Protons erhalten oder auflösen. In ersterem Fall spricht man von einem *nicht dissoziativen*, in letzterem von einem *dissoziativen* Prozeß. Das zwischen Vektormeson und Proton, bzw. Quarks, ausgetauschte farbneutrale Teilchen ist das bereits erwähnte Pomeron.

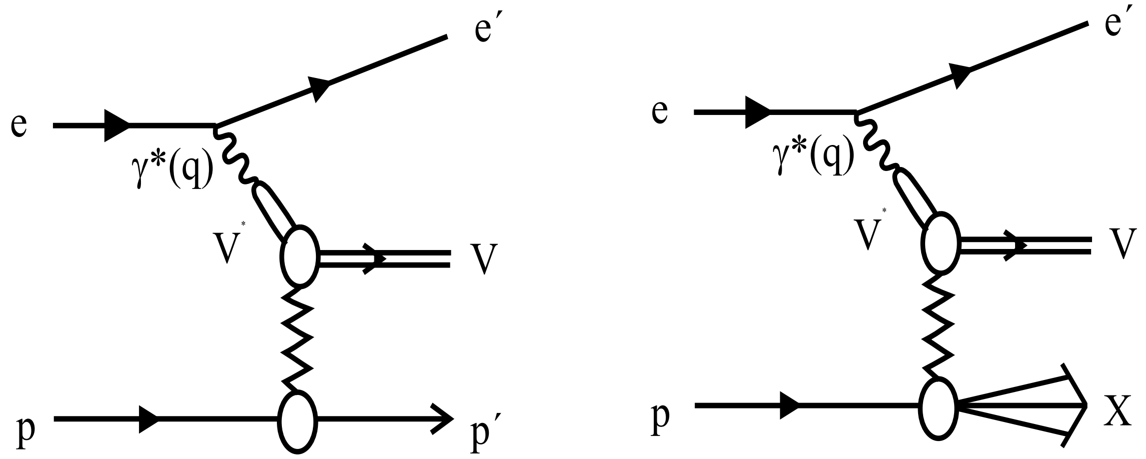

Bild 1.4: Diffraktive Erzeugung von Vektormesonen. Ein vom Elektron abgestrahltes virtuelles Photon (γ) fluktuiert in eine virtuelles Vektormeson (V*), welches anschließend, durch starke Wechselwirkung mit dem Proton, reell wird. Im Bild links ist der elastische Prozeß gezeigt, der die Struktur des Protons erhält, rechts der dissoziative Prozeß.*

Die Umwandlung des virtuellen Photons in ein virtuelles Meson kann z.B. durch das *Vektormesonen Dominanz Modell* (*VDM*) beschrieben werden ([6][7]). Dabei nimmt man an, daß der Zustandsvektor des virtuellen Photons neben der reinen, normierten QED-Komponenten auch einen hadronischen Vektormesonenanteil enthält. Dies ist möglich, da sowohl das Vektormeson als auch das Photon, wie bereits erwähnt, die selben Quantenzahlen (1^{--}) tragen. Durch dieses phänomenologische Modell kann die Produktion der Vektormesonen mit der Hadron-Hadron-Streuung verknüpft werden. Der Zustandsvektor des Photons läßt sich dann schreiben als:

$$|\gamma\rangle = N\left|\gamma_{QED}\right\rangle + \sum_{v} \frac{e}{\gamma v}|V\rangle$$

Hier bezeichnet $\left|\gamma_{QED}\right\rangle$ den reinen QED Zustand, N eine geeignete Normierungskonstante und $|V\rangle$ den Zustandsvektor eines Vektormesons. Die Kopplung γv kann aus der elektromagnetischen Zerfallsbreite des jeweiligen Vektormesons bestimmt werden.

1.2.6 Zerfallswinkelverteilung

Das ϕ-Meson ist ein Spin 1 Teilchen und hat daher eine Zerfallswinkelverteilung die von den Spins der Zerfallsprodukte und den beim Zerfall auftretenden relativen Drehimpulsen abhängt. Eine eventuell vorhandene Zerfalls-Anisotropie macht sich folglich in den, vom Detektor aufgezeichneten Daten, bemerkbar. Neben dem Zerfallswinkel lassen sich auch bei der Produktion des Vektormesons zwei Winkel definieren, die eventuell in der Triggerbedingungen verwendet werden könnten. Bild 1.5 zeigt die Definition der Produktions- und Zerfallswinkel.

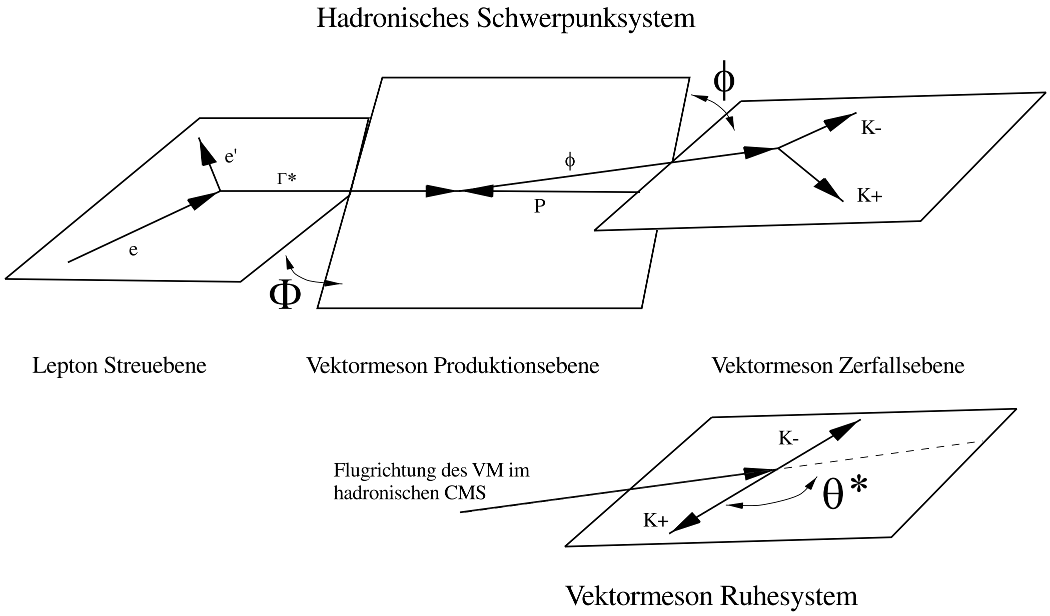

Bild 1.5: Definition der Zerfallswinkel im hadronischen Schwerpunktsystem und dem Helizitätssystem (Ruhesystem des erzeugten Vektormesons mit der Richtung des Vektormesons im hadronischen Schwerpunktsystem als Quantisierungsachse). (aus [2])

Der Winkel zwischen der Lepton-Streuebene und der Ebene, die durch Protonimpuls und Impuls des Vektormesons (Vektormeson-Produktionsebene) festgelegt wird, soll als Φ bezeichnet werden. Der Winkel ϕ bezeichnet dann die Drehung der Vektormeson-Produktionsebene gegen die Zerfallsebene des Vektormesons. Im Ruhesystems des Vektormesons kann noch der Zerfallswinkel θ^* definiert werden. Dieser wird zwischen der ursprünglichen Flugrichtung des Mesons und der Impulsrichtung der Zerfallsprodukte gemessen.

2 HERA und der H1 Detektor

2.1 HERA

Der Speicherring HERA (Hadron-Elektron-Ring-Anlage) ist seit 1992 am DESY (Deutschen Elektronen Synchrotron) in Betrieb. HERA wurde für die Kollision von Protonen und Elektronen konstruiert. Dabei werden Elektronen oder Positronen mit einer Energie von 27,6 GeV und Protonen mit einer Energie von 820 GeV aufeinander geschossen. So sind Experimente zur Untersuchung der Struktur des Protons in bisher unerreichten kinematischen Bereichen der tiefinelastischen Streuung möglich. Außerdem können Experimente zur Überprüfung des Standardmodells in diesen kinematischen Bereichen gemacht werden, sowie die Suche nach neuen Teilchen oder möglicher Substrukturen der Quarks und Elektronen. Ebenso sind Photoproduktions-Experimente mit einer Energie von 40 TeV im Proton-Ruhesystem, machbar. Die Energie entspricht dabei einem Photonenstrahl; bei der Kollision ruht das Target [13].

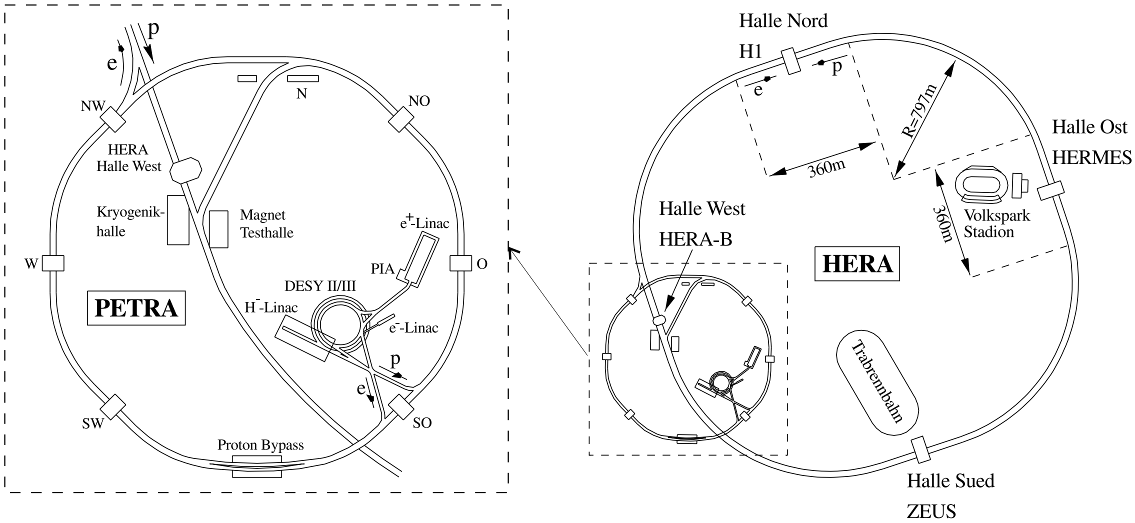

Bild 2.1: *Die gesamte HERA-Ring Anlage in der Übersicht (rechts). Im Detail (links) die Vorbeschleuniger LINAC, DESY II/III und PETRA II/III.*

Bei HERA werden vier Experimente betrieben: H1, ZEUS, HERA-B und HERMES.

- Das H1 Experiment und das Experiment ZEUS befassen sich mit der Vermessung der Struktur des Protons, dem Studium von fundamentalen Interaktionen zwischen Teilchen und der Suche nach Physik jenseits des Standardmodells von Elementarteilchen. Dazu werden Elektron- und Protonstrahlen aufeinander geschossen; deshalb auch die Bezeichnung: *Colliding Beam* Experimente.
- Im Gegensatz dazu wird bei HERA-B und HERMES ein festes Target verwendet (*Fixed Target* Experiment). HERA-B ist ein Experiment um die CP Verletzung im B-System zu studieren. Dabei werden Protonen verwendet. Das Ziel des Experimentes HERMES ist die Messung der Spin-Verteilung von Quarks in Protonen und Neutronen. Zum Einsatz kommt hierbei der Elektronen- oder Positronenstrahl des HERA-Rings.

2.1.1 Vorbeschleunigung für HERA

Die Protonen und Elektronen/Positronen können nicht direkt im HERA Speicherring auf ihre Energie gebracht werden. Sie werden erst durch mehrere Vorbeschleuniger auf die Injektionsenergie für den Speicherring gebracht [34]. Dabei beschleunigt man die beiden Teilchenarten unterschiedlich.

Bis Juli 1994 wurden Elektronen verwendet; danach – bis einschließlich 1997 – benutzte man Positronen. In der Datennahmeperiode 1998 wurden wieder Elektronen eingesetzt, um die ladungsbedingten Unterschiede in der $e^{+/-}$p-Reaktion zu studieren. Beim Einsatz von Elektronen gab es früher Probleme mit der Lebensdauer des Elektronenstrahls (~2 Stunden). Verwendet man anstatt der Elektronen, Positronen, kann die Lebensdauer des Strahls bis zu 20 Stunden betragen [14]. In jüngster Zeit ist es gelungen, das Vakuum in der Strahlröhre so weit zu verbessern, daß der Elektronenstrahl eine fast ebenso lange Lebensdauer hat, wie der Stahl aus Positronen[3].

2.1.1.1 Beschleunigung der Elektronen oder Positronen

Ein Linearbeschleuniger bringt die Positronen auf 500 MeV. Ein Speicherring sammelt sie dann, bevor sie in das Synchrotron DESY II injiziert werden. Dort werden sie auf 7 GeV beschleunigt und anschließend in Bündeln (bunches) von bis zu $0{,}4\times10^{11}$ Teilchen zu PETRA II geleitet. Dort können bis zu maximal 70 dieser Bündel gehalten werden. In PETRA II erhalten sie die Injektionsenergie von 14 GeV für den Elektronenring von HERA.

[3] Der Elektronenstrahl ionisiert die in der vakuumisierten Röhre verbliebenen Restatome. Die so positiv gewordenen Atomrümpfe werden vom negativen Elektronenstrahl angezogen und verursachen dann Kollisionen, welche die Strahllebensdauer verkürzen. Die positiven Positronen stoßen die ebenfalls positiven Atomrümpfe ab, wodurch es zu deutlich weniger Kollisionen kommt. Durch ein besseres Vakuum wird dieser Effekt abgeschwächt.

2.1.1.2 Beschleunigung der Protonen

Die Protonen werden als H^--Ionen in einem Linearbeschleuniger auf 50 MeV gebracht. Bevor sie in DESY III geleitet werden, entfernen Stripperfolien die beiden überschüssigen Elektronen.. DESY III beschleunigt die Protonen auf 7,5 GeV. Mit dieser Energie werden wieder jeweils bis zu 70 Bündel in PETRA II eingeleitet, wo sie vor ihrer Injektion in den HERA Protonring auf 40 GeV beschleunigt werden.

2.1.2 Der HERA Speicherring

Der HERA Speicherring besteht eigentlich aus zwei Ringen: einen für die Protonen und einen für die Elektronen. Sie befinden sich am DESY in einem unterirdischen Tunnel in 10-30m Tiefe. Die Speicherringe haben einen Umfang von 6336 m. In jedem Ring halten unterschiedliche Magnetfelder die Teilchen auf ihrer Bahn.

Für die Positronen wird ein Magnetfeld von 0,165 T benötigt; erzeugt von konventionellen Dipolmagneten. Supraleitende HF-Resonatoren, betrieben mit einer Frequenz von 500 MHz und einer Leistung von 13,2 MW, beschleunigen die Positronen und gleichen den Energieverlust hervorgerufen durch Synchrotronstrahlung aus. Die schwereren Protonen benötigen 4,68 T, welche durch supraleitende Magnete erzeugt werden.

Die HERA-Ringe können mit jeweils bis zu 210 Teilchenbündeln gefüllt werden. Diese treffen alle 96 ns an den vorgesehenen Wechselwirkungspunkten aufeinander (bunch crossing).

Die Luminosität L bestimmt die Leistungsfähigkeit eines Beschleunigers. Mit der Bündeldurchdringungsfrequenz f, dem Strahlquerschnitt A und der mittleren Anzahl der Eletronen n_e und Protonen n_p pro Bündel ergibt sich:

$$L = f \cdot \frac{n_e \cdot n_p}{A} \tag{2.14}$$

In Tabelle 2.4 sind die Beschleunigerdaten seit 1995 im Vergleich zu den Designwerten aufgelistet[33].

Tabelle 2.4: HERA Beschleinigerdaten seit 1995 im Vergleich zu den Designwerten.

		Design		*1995*		*1996*		*1997*		*1998*[1]	
		e^+	p	e^+	p	e^+	p	e^+	p	e^-	p
Strahlenergie	(GeV)	30,0	820	27,5	820	27,5	820	27,5	820	27,5	920
Schwerpunktsenergie	(GeV)	314		300		300		300		318	
Mittlerer Strahlstrom	(mA)	58	163	18	54	21	60	28	74	16	63
Teilchenpakete		210	210	174	174	175	175	175	175	175	175
Max. L	($cm^{-2}s^{-1}$)	$1{,}5\times10^{31}$		$0{,}59\times10^{31}$		$0{,}84\times10^{31}$		$1{,}01\times10^{31}$		$0{,}82\times10^{31}$	
Max. Spez. L	($cm^{-2}s^{-1}mA^{-2}$)	$4{,}0\times10^{29}$		$6{,}2\times10^{29}$		$7{,}6\times10^{29}$		$6{,}0\times10^{29}$		$16{,}3\times10^{29}$	
$\int L\,dt$ (HERA)	($pb^{-1}y^{-1}$)	100		10,1		13,3		32,4		8,1	
$\int L\,dt$ (von H1 genutzt)	($pb^{-1}y^{-1}$)	100		6,2		8,9		27,3		6,4	

[1] *Im Versuchsjahr 1998 wurden wieder – wie ursprünglich geplant – Elektronen verwendet.*

2.2 H1 Detektor

Ein internationales Konsortium aus 33 Instituten hat den H1 Detektor konzipiert und gebaut. Derzeit sind am Betrieb des Detektors 39 Institute aus 12 Ländern mit über 400 Physikern beteiligt. Der H1 Detektor ist ein Vielzweckdetektor für den Nachweis von geladenen und ungeladenen Teilchen. Aufgrund seiner hermetischen Bauweise kann er auch zum indirekten Nachweis von Neutrinos verwendet werden [37].

2.2.1 Übersicht

Der Detektor ist wegen der unterschiedlichen Impulse von Positronen und Protonen in Längsrichtung asymmetrisch aufgebaut. Der nominelle Wechselwirkungspunkt (*Vertex*) wird von den einzelnen Komponenten des H1-Detektors in mehreren Lagen umschlossen. Bild 2.2 zeigt eine schematische Ansicht des H1 Detektors mit seinen einzelnen Komponenten.

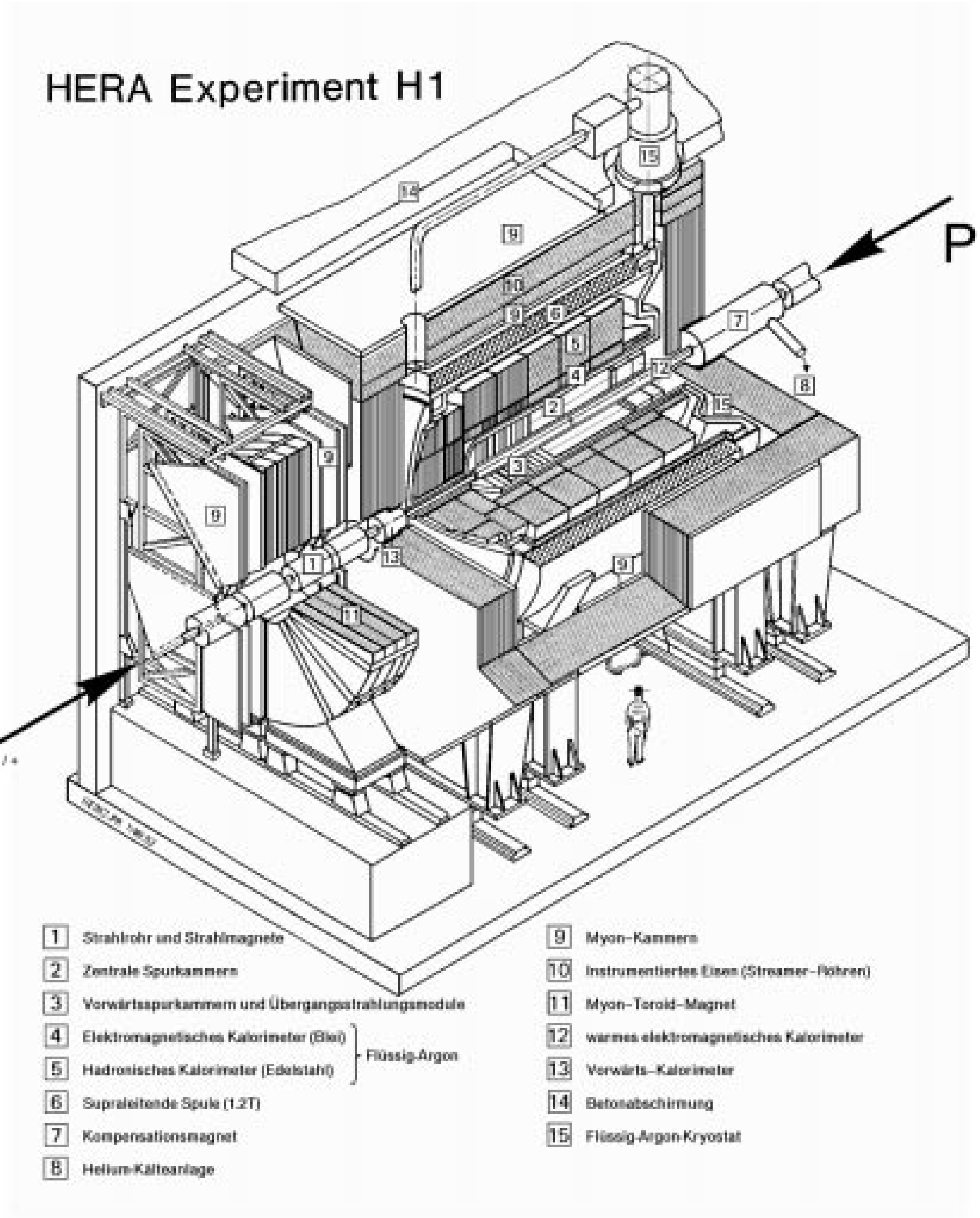

Bild 2.2: Schematische Ansicht des H1- Detektors.

Zylindrische Spurkammern (2) umgeben den Wechselwirkungspunkt. In Vorwärtsrichtung werden sie durch Vorwärtsspurkammern und Übergangsstrahldetektoren (3) ergänzt. Ein Flüssig-Argon-Kalorimeter (LAr-Kalorimeter) umschließt die Spurkammern. Es besteht aus einem inneren, elektromagnetischen (4) und einem äußeren, hadronischen Teil (5). Das Kalorimeter ist Hauptkomponente für den Nachweis von Elektronen, Photonen, Hadronen. Es hat eine sehr gute Energie und Ortsauflösung, eine hohe Genauigkeit in der absoluten Energieeichung und ermöglicht so eine präzise Vermessung der Kinematik von ep-Reaktionen. Im Rückwärtsbereich befindet sich ein Spaghetti-Blei-Szintillations-Kalorimeter (12). Von diesem wird bereits der elektromagnetische Teil eingesetzt; der Einsatz eines hadronischen Teils ist noch offen.

Eine supraleitende Spule (6), mit einem 1,15 T starken Magnetfeld, umgibt das LAr-Kalorimeter. So bewegen sich geladene Teilchen auf parallel zur z-Achse ausgerichteten Helixbahnen. Diese Bahnen werden in den Spurkammern rekonstruiert, und so die Impulse der Teilchen, die Ereignistopologie und die Position des Vertex bestimmt.

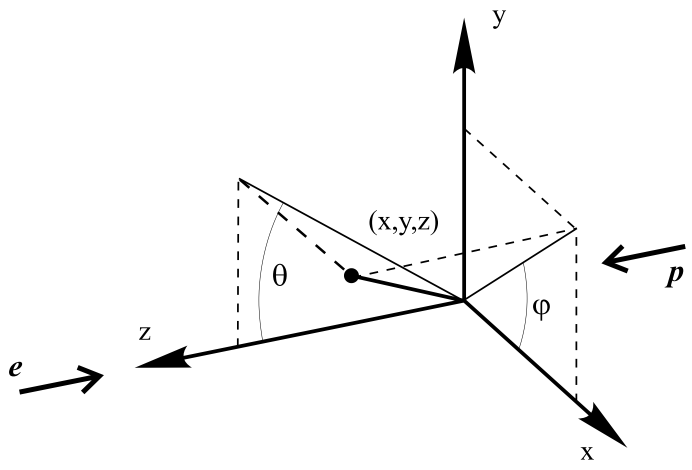

Bild 2.3: Das H1-Koordinatensystem

Bei H1 wird ein Koordinatensystem verwendet (Bild 2.3), das durch die von der Protonflugrichtung definierten z-Achse festgelegt ist. Die x-Achse zeigt in Richtung Zentrum des HERA-Rings. Der Ursprung des Koordinatensystems liegt im nominellen Wechselwirkungspunkt der *ep*-Reaktion. Es werden noch zwei Winkel definiert: der Azimutalwinkel wird gegen die positive x-Achse gemessen, der Polarwinkel gegen die positive z-Achse (Vorwärtsrichtung).

Bild 2.4 zeigt einen Längsschnitt durch den H1 Detektor. In den folgenden Abschnitten sollen die, für diese Arbeit wichtigen Komponenten des Detektors genauer beschrieben werden.

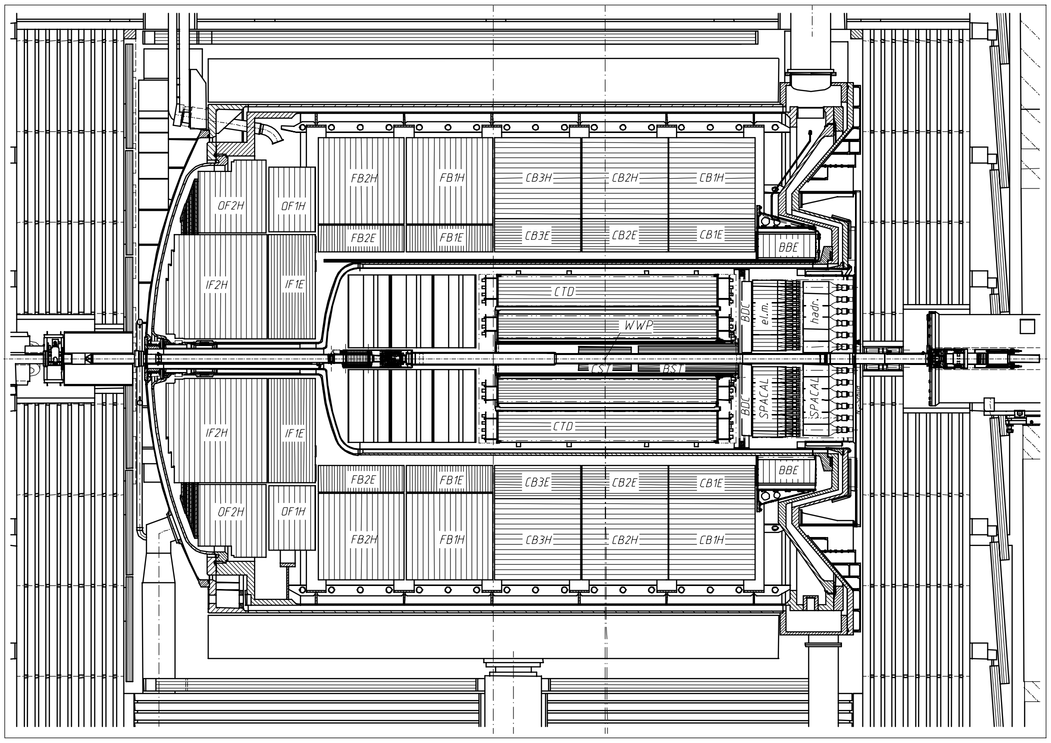

Bild 2.4: Längsschnitt durch den H1 Detektor.

2.2.2 Spurkammern

Es gibt ein zentrales Spurkammer-System und ein Vorwärts-Spurkammer-System. Zum zentralen Spurkammer-System gehören die zentralen Driftkammern, die z-Kammern und die zentralen Proportionalkammern. Zum Vorwärts-Spurkammer-System gehören die planaren und radialen Driftkammern sowie die Proportionalkammern. Der Winkelbereich geht von etwa 20° bis 160°.

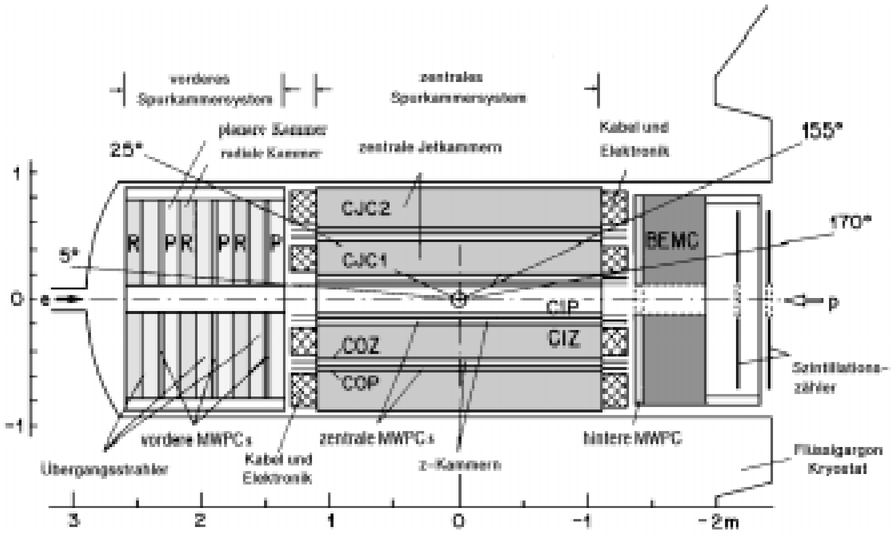

Bild 2.5: Seitenansicht des Spurkammersystems von H1.

2.2.2.1 Zentrale Proportionalkammern

Vieldraht-Proportionalkammern (MWPC) werden zur schnellen Erkennung von Spuren geladener Teilchen eingesetzt. Die inneren und äußeren Kammern *CIP* und *COP* (*Central Inner/Outer Proportionalchamber*) werden hauptsächlich als z-Vertex Trigger eingesetzt [43]. Sie bestehen aus einer Doppellage von Drähten und sind in z-Richtung 60-fach (CIP) und 18-fach (COP) segmentiert. Die beiden Kammern sind in der $r\varphi$-Ebene in 16 Sektoren unterteilt. Wird ein Signal in mindestens drei der vier Drahtlagensegmenten innerhalb eines Sektors registriert, wird dies zu einem Strahl (ray) zusammengefaßt. Das z-Vertex Histogramm enthält dann die Extrapolationen aller Rays auf die z-Achse. Das z-Vertex Histogramm ist in 16 Intervalle in einem Bereich von [-43,9;43,9] cm um den nominellen Wechselwirkungspunkt unterteilt [2].

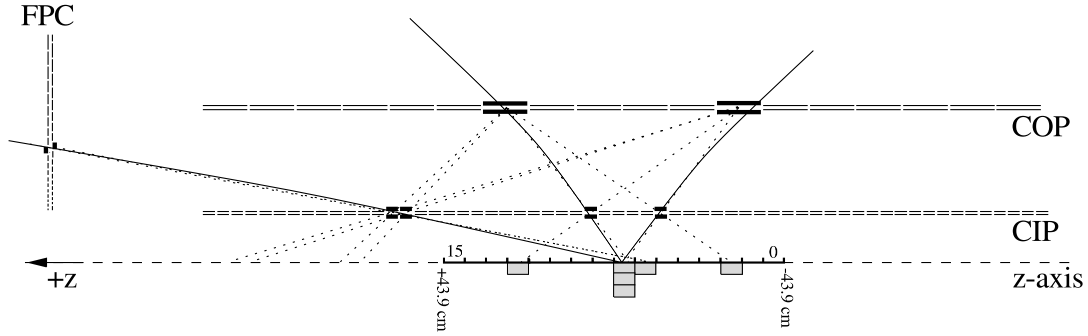

Bild 2.6: Der z-Vertex Histogramm Trigger. Die Zahl von extrapolierten Rays wird in dem z-Vertex Histogramm zusammengefaßt.

Die Signale der Propkammern stehen bereits nach 800ns zur Verfügung und werden für Triggerzwecke verwendet.

Tabelle 2.5: Die technischen Merkmale der zentralen Proportionalkammern.

		CIP	*COP*
Position der ersten Lage	(mm)	157	501
Position der zweiten Lage	(mm)	166	514
Aktive Länge Δz	(mm)	2190	2172
Anzahl der Felder in z		60	18
Anzahl der Felder in		8	416
Zeitauflösung	(ns)	21	

2.2.2.2 Z-Kammern

Die Aufgabe der inneren und äußeren z-Driftkammern *CIZ* und *COZ* (*Central Inner/Outer Z-Drift-Chamber*) ist es, die Lage einer Spur entlang der Strahlachse genau zu bestimmen[42]. Dazu verlaufen die Signaldrähte senkrecht zur Strahlachse, so daß die freigesetzten Elektronen parallel zur Strahlachse driften. Die Signaldrähte sind in vier Lagen, ringförmig um die Strahlachse angeordnet. Sie sind in 15 (CIZ) und 24 (COZ) Segmente aufgeteilt.

Tabelle 2.6: Die technischen Merkmale der z-Kammern.

		CIZ	*COZ*
Innerer Radius	(mm)	173,5	460
äußerer Radius	(mm)	200	485
Aktive Länge Δz	(mm)	1800	2160
Anzahl der Driftzellen		15	24
Anzahl der Signaldrähte pro Ring		4	4
Ortsauflösung $\sigma_{r\varphi}$	(mm)	28	58
Ortsauflösung σ_z	(mm)	0,26	0,20

2.2.2.3 Zentrale Driftkammer

Die zentralen Jetkammern (*CJC*, *Central Jet Chamber*) bestehen aus einer inneren (CJC1) und einer äußeren Kammer (CJC2) [41]. Sie enthalten eine Mischung aus Argon, Kohlendioxid und Methan im Verhältnis 89,5:9,5:1,0. Die Signaldrähte sind parallel zur Strahlachse gespannt.

Die innere Kammer ist in 30 Zellen, die äußere in 60 unterteilt. Dabei sind die einzelnen Zellen um 30 Grad geneigt, damit die Driftrichtung senkrecht zu einer vom Strahlrohrmittelpunkt kommenden Spur liegt. Die Ortsauflösung in der $r\varphi$-Ebene beträgt $\sigma_{r\varphi}$ 170 µm.

Tabelle 2.7: Die technischen Merkmale der zentralen Driftkammern.

		CJC1	*CJC2*
Innerer Radius	(mm)	203	530
äußerer Radius	(mm)	451	844
Aktive Länge Δz	(mm)	2200	
Anzahl der Driftzellen		30	60
Anzahl der Signaldrähte pro Zelle		24	32
Ortsauflösung $\sigma_{r\varphi}$	(mm)	0,17	
Ortsauflösung σ_z	(mm)	22	
Impulsauflösung σ_p/p^2	(GeV^{-1})	0,003	
dE/dx-Auflösung		$\sigma(dE)/dE$=6%	

2.2.3 Kalorimeter

Die Hauptkomponenten des H1-Detektors sind die Kalorimeter. Sie ermöglichen die Energiemessung und Teilchenidentifikation von Elektronen, Myonen, neutralen Teilchen oder hadronischen Schauern mit hohen Teilchendichten. Elektronen, Positronen oder Photonen geben im Kalorimeter Energie durch Bremsstrahlungsprozesse und e^+e^- Paarbildung ab. Dadurch entstehen weitere Teilchen, die den elektromagnetischen

Schauer bilden. Ein hadronischer Schauer entsteht durch inelastische Stöße von stark wechselwirkenden Teilchen mit Atomkernen. Die hadronischen Schauer haben in der Regel eine größere Ausdehnung als die elektromagnetischen.

2.2.3.1 LAr-Kalorimeter

Das *Liquid-Argon-Kalorimeter* (*LAr-Kalorimeter*) [35] ist das größte Kalorimeter des H1 Detektors; es deckt den Winkelbereich von $4° \leq \theta \leq 153°$ ab. Das LAr-Kalorimeter befindet sich in einer Kühlvorrichtung: dem Kryostaten. Es ist ein Schichtkalorimeter und besteht aus schichtweise angeordneten Absorberplatten aus Blei und Edelstahl. Zwischen diesen Platten befindet sich flüssiges Argon als aktives Ionisierungsmaterial. Das LAr-Kalorimeter ist außerdem unterteilt in einen inneren, elektromagnetischen Teil und einen äußeren, hadronischen Teil. Im elektromagnetische Teil wird Blei als Absorbermaterial verwendet; im hadronischen Edelstahl. Dadurch können elektromagnetische und hadronische Schauer durch ihre unterschiedliche Schauertiefe und Schauerausbreitung unterschieden werden.

Das LAr-Kalorimeter ist nicht kompensierend: bei gleicher Primärenergie ist das Signal eines elektromagnetisch wechselwirkenden Teilchens höher als das eines hadronischen. Die Kalibrierung der elektromagnetischen Skala sorgt dafür, daß die Energie der elektromagnetisch wechselwirkenden Teilchen, im Mittel richtig gemessen wird. Auf dieser Skala wird für hadronische Teilchen eine zu geringe Energie gemessen. Dieser Effekt wird durch eine Rekonstruktionstechnik (Gewichtung) korrigiert.

Die Energieauflösung beträgt für Elektronen im elektromagnetischen Teil $s(E)/E \sim 12\%/\sqrt{E(\text{GeV})}$ und für Pionen $s(E)/E \sim 50\%/\sqrt{E(\text{GeV})}$ im hadronischen Teil [35]. Sie ist dominiert durch statistische Fluktuaktionen an ionisierenden Elektronen in den aktiven Schichten.

2.2.3.2 SaCal

Das Spaghetti Calorimeter (SpaCal) ist ein Spaghetti-Blei-Szintillations-Kalorimeter [44]. Seinen Namen hat das SpaCal wegen seiner Konstruktion aus spaghettiförmigen, szintillierenden Lichtleitern, die in Bleiplatten eingelegt sind. Es wird zur Messung mit hoher Orts-, Energie- und Zeitauflösung von Teilchen im rückwärtigen Bereich gebraucht. Dazu deckt es einen Winkelbereich von $153° \leq \theta \leq 178°$ ab. Der Abstand vom Wechselwirkungspunkt beträgt 1,50m bis 2,45m; zuerst kommt der elektromagnetische Teil, dann der hadronische [45].

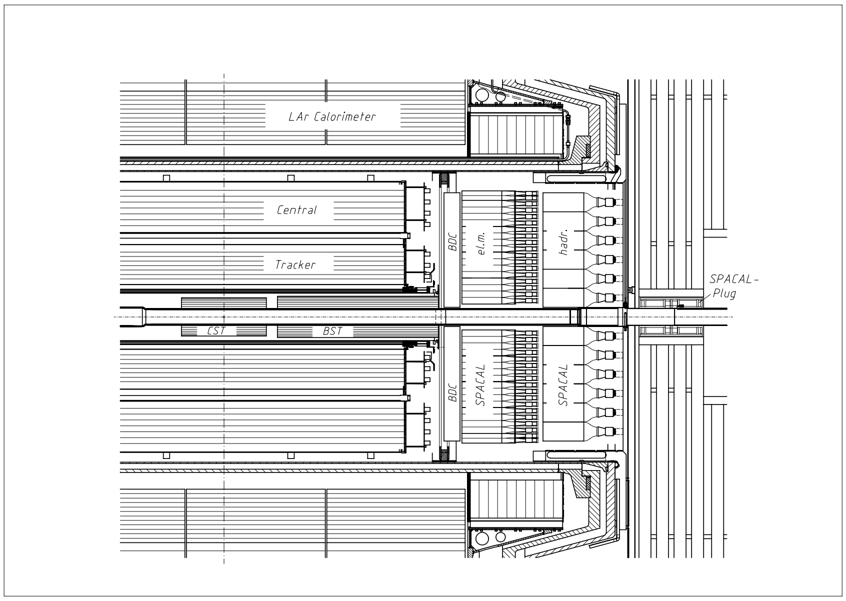

Bild 2.7: Längsschnitt durch den H1 Detektor. Zu sehen ist der rückwärtige Bereich, mit dem hadronischen und elektromagnetischen SpaCal.

Die kleinsten Einheiten des SpaCal sind die Submoduln. Im elektromagnetischen Teil besteht ein Submodul aus 52 Bleiplatten. Jede dieser Platten hat eine Größe von $81\times40{,}5\times250\ \mathrm{mm}^3$ und enthält 90 szintillierende Fasern, welche in zwei Bündeln zu den Photomultipliern geführt werden. Jeweils acht Submoduln faßt man zu einem Supermodul zusammen. Sechzig dieser Supermoduln bilden den elektromagnetischen Teil des SpaCal. Das Funktionsprinzip: Einfallende Teilchen erzeugen im Blei Energieschauer, welche die Fasern der Lichtleiter zur Szintillation anlegen. Das Licht wird in den Fasern durch Totalreflexion zu Photomultipliern geleitet.

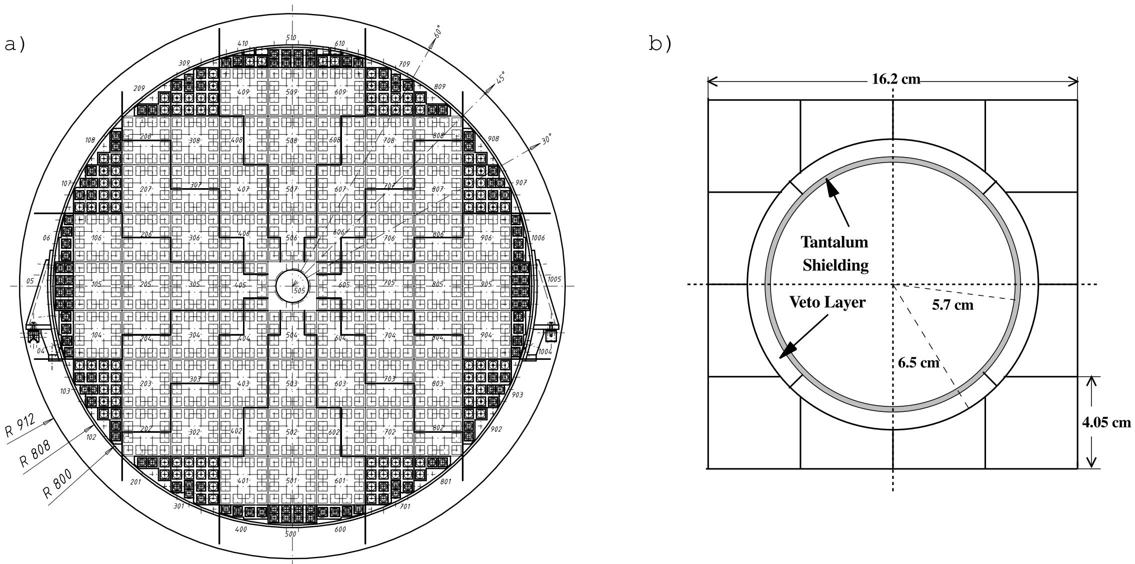

Bild 2.8: a) Querschnitt des elektromagnetischen SpaCals. b) Ausschnitt des Bereichs direkt um das Strahlrohr.

Der hadronische Teil des SpaCal ist Prinzip so aufgebaut wie der elektromagnetische; er unterscheidet sich nur durch die Zellanzahl und -größe (siehe Tabelle 2.8) und durch den doppelten Durchmesser der szintillierenden Fasern und damit dem Verhältnis von Blei zu Faser. In dieser wird auf den hadronischen Teil nicht nähre eingegangen.

Tabelle 2.8: SpaCal Eigenschaften

		elektromagnetisch	*hadronisch*
Anzahl der Zellen		1192	128
Zellgröße	(mm^2)	40,5×40,5	120×120
Verhältnis Blei/Faser		2:1	4:1
Energieauflösung	(%)	$\frac{(7{,}1 \pm 0{,}2)}{\sqrt{E}} \oplus (1{,}0 \pm 0{,}1)$	56,0±3,0
Ortsauflösung	(mm)	$\frac{(4{,}4 \pm 0{,}4)}{\sqrt{E/GeV}} + (1{,}0 \pm 0{,}2)$	
Winkelauflösung	(mrad)	≤ 2	
Zeitauflösung	(ns)	0,38±0,03	1

2.2.4 Zentrales Myon-System

Das zentrale Myon-System besteht aus Luranyl-Rohren (*Limited Streamer Tubes, LST*). Sie werden als Gaszähler betrieben und zu insgesamt 16 Lagen zusammengefaßt. Im Zentrum jedes Rohres befinden sich Signaldrähte; zusätzlich sind die Rohre in 5 Lagen mit senkrecht zu den Drähten verlaufenden Streifen ausgerüstet. Die anderen 11 Lagen werden über Felder ausgelesen.

Die Felder liefern analoge Signale, die zur Energiemessung von hadronischen Teilchen benutzt werden, welche nicht im LAr-Kalorimeter absorbiert wurden. Dabei beträgt die Energieauflösung $\sigma / E \cong 100\% / \sqrt{E}$.

Die Signaldrähte und die Steifen werden digital ausgelesen. Die so gewonnene Information wird verwendet um Treffermuster zu erstellen, welche zu Myonspuren kombiniert werden [16]. Zur Auswertung der digitalen Signale siehe auch Kapitel 5.3.1.4.

2.2.5 Weitere Komponenten

2.2.5.1 Luminositätssystem

Die Luminosität wird mit dem Bethe-Heitler Prozeß ($ep \rightarrow ep\gamma$) bestimmt [36], dessen Wirkungsquerschnitt hinreichend groß ist. Dabei muß eine Koinzidenz zwischen dem

Photon und dem abgelenkten Positron nachgewiesen werden. Hierfür benötigt man zwei Detektoren: den Elektronen-Tagger und den Photon-Detektor. Beide sind Cerenkov-Kalorimeter, die über Photomultiplier ausgelesen werden.

2.2.5.2 Proton-Tagger

Der Proton-Tagger dient dem Nachweis von Fragmenten des gestreuten Protons, die – unsichtbar für den Hauptdetektor – durch das Strahlrohr entweichen. Er befindet sich 24m in Vorwärtsrichtung vom Wechselwirkungspunkt entfernt. Der Proton-Tagger deckt den Winkelbereich von $0{,}06° < \theta < 0{,}25°$ ab, und besteht aus sieben Paaren von – in Koinzidenz betriebenen – Plastikszintillatoren.

2.2.5.3 Flugzeitsystem und Vetowand

Das *Flugzeitsystem* (*Time Of Flight, TOF*) befindet sich im hinteren Bereich des H1-Detektors und besteht aus mehreren Szintillatoren in Form eines Rings im Abstand von 3m vom Wechselwirkungspunkt. Das System identifiziert Untergrundereignisse mittels Flugzeitmessung, wobei die Zeitauflösung 2-4ns beträgt. Untergrundereignisse werden von echten ep-Wechselwirkungen durch ein Zeitfenster von 13ns Breite unterschieden.

Ebenfalls in Rückwärtsrichtung befinden sich die *Vetowände* aus doppelten Szintillatorebenen in einem Abstand von 6,5m und 8,1m vom Wechselwirkungspunkt. Die Signale der beiden Wände haben eine Zeitauflösung von 3ns bzw. 8ns.

2.2.6 Triggersystem

Die Aufgabe des Triggersystems ist die schnelle und effiziente Erkennung von Physik- und Untergrundereignissen. Der Trigger soll die, von den Detektoren, für jede Kollision (also alle 96ns) gelieferte Datenmenge, auf ein verarbeitbares Maß reduzieren, da es nicht möglich ist alle einlaufenden Daten zu verarbeiten; und Physik und Untergrund in so kurzer Zeit effizient zu trennen. Die Quellen für Untergrundereignisse sind:

Strahl-Gas-Kollisionen
: Das Vakuum im Strahlrohr ist nicht perfekt. Es enthält eine Restgasdichte von ungefähr einer Million Molekülen pro Kubikzentimeter. Die Protonen wechselwirken mit dem Restgas und erzeugen so mit einer Rate von einigen kHz Untergrundereignisse.

Strahl-Wand-Ereignisse
: Die Protonen wechselwirken auch mit der Strahlrohrwand und produzieren Ereignisse mit einer Rate von 50-100kHz.

Synchrotronstrahlung
: Die Positronen als beschleunigte, geladene Teilchen geben Energie in Form von Synchrotronstrahlung ab. Diese kann in verschiedenen Detektorkomponenten Signale erzeugen.

Kosmische Strahlung

Kosmische Höhenstrahlung in Form von Myonen trifft mit einer Rate von bis zu 2 kHz auf den Detektor und produziert so Signale.

Diese Untergrundereignisse zu verwerfen ist Aufgabe des Triggersystems. Zusätzlich sollen der Trigger Physikereignisse mit einer Rate von bis zu rund 10 Hz selektieren. Aufgrund der hohen Ereignisrate, arbeitet der Trigger mehrstufig.

2.2.6.1 Datennahme

Die Kollisionen von Protonbündel und Positronbündel erfolgen mit einer Frequenz von 10,4 MHz; also alle 96ns.

2.2.6.2 Erste Triggerstufe (L1)

Diese Triggerstufe hat eine festverdrahtete Logik in Form von elektronischen Schaltungen. So ist es hier möglich, innerhalb von 2,3 µs zu entscheiden, ob ein Ereignis akzeptiert wird oder nicht. Grundlage für die Entscheidung sind relativ einfache, meist topologische Größen wie z.B. Spurkandidaten in einem bestimmten Impulsbereich, Energieschwellen oder Vertexsignifikanz. Diese Triggerinformationen werden von den meisten Komponenten des H1-Detektors in Form von sogenannten Triggerelementen zur Verfügung gestellt. Mehrere dieser Triggerelemente können durch logische Operationen zu maximal 128 Triggerbedingungen – auch Subtrigger genannt – verknüpft werden.

Wenn mindestens eine Subtrigger-Bedingung erfüllt ist, ist dies das Signal zum Auslesen des Detektors. Die Zeitverzögerung bis dahin, sind die bereits erwähnten 2,3 µs. Während dieser Zeit werden die Triggerdaten in einer Pipeline zwischengespeichert. Somit arbeitet der L1 Trigger bis zu dem Moment totzeitfrei, in dem eine positive Triggerentscheidung gefällt wird.

Wenn die Subtrigger-Bedingung zu wenig streng ist, kann die Ereignisrate zu hoch werden. In diesen Fällen ist es möglich einen *Prescale* anzuwenden. Ein Prescale von n bedeutet, das jedes n-te Ereignis akzeptiert, was einer Ratenreduktion um den Faktor n entspricht.

2.2.6.3 Zweite Triggerstufe (L2)

Hier stehen die Informationen aus den L1 Subtriggern zur Verfügung. Sie müssen innerhalb von 20 µs weiterverarbeitet werden; dies entspricht der Forderung nach maximal 10% Totzeit.

In der zweiten Triggerstufe werden zwei verschiedene Hardware-Trigger eingesetzt. Das ist zum einen der *Neuronale Netzwerktrigger (L2NN)*; er wird in Kapitel 3 detailliert beschrieben. Zum anderen der *Topologische Trigger (L2TT)*. Dieser untersucht die Daten mit Hilfe von Matrixoperationen und logischen Verknüpfungen auf topologische Korrelationen.

2.2.6.4 Dritte Triggerstufe (L3)

Diese Triggerstufe ist mit einem Mikroprozessor aufgebaut. Dadurch können Algorithmen in üblichen Hochsprachen programmiert werden. L3 bekommt die selben Informationen wie L2, wobei die Verarbeitungszeit 800 µs beträgt. Der Trigger wurde bisher noch nicht eingesetzt.

2.2.6.5 Vierte Triggerstufe (L4)

Die vierte Triggerstufe besteht aus 36 Mikroprozessoren, die programmiert werden können. In L4 steht die gesamte Detektorinformation zur Verfügung, wodurch bereits eine vereinfachte Ereignisrekonstruktion möglich ist. Ein Ringpuffer sammelt die Daten der verschiedenen Detektorkomponenten aus bis zu 30 Ereignissen. Jedes Ereignis in dem Puffer wird von einem Prozessor für eine partielle Ereignisrekonstruktion ausgelesen. Mehrere Selektionsalgorithmen entscheiden, ob ein Ereignis permanent gespeichert oder verworfen wird.

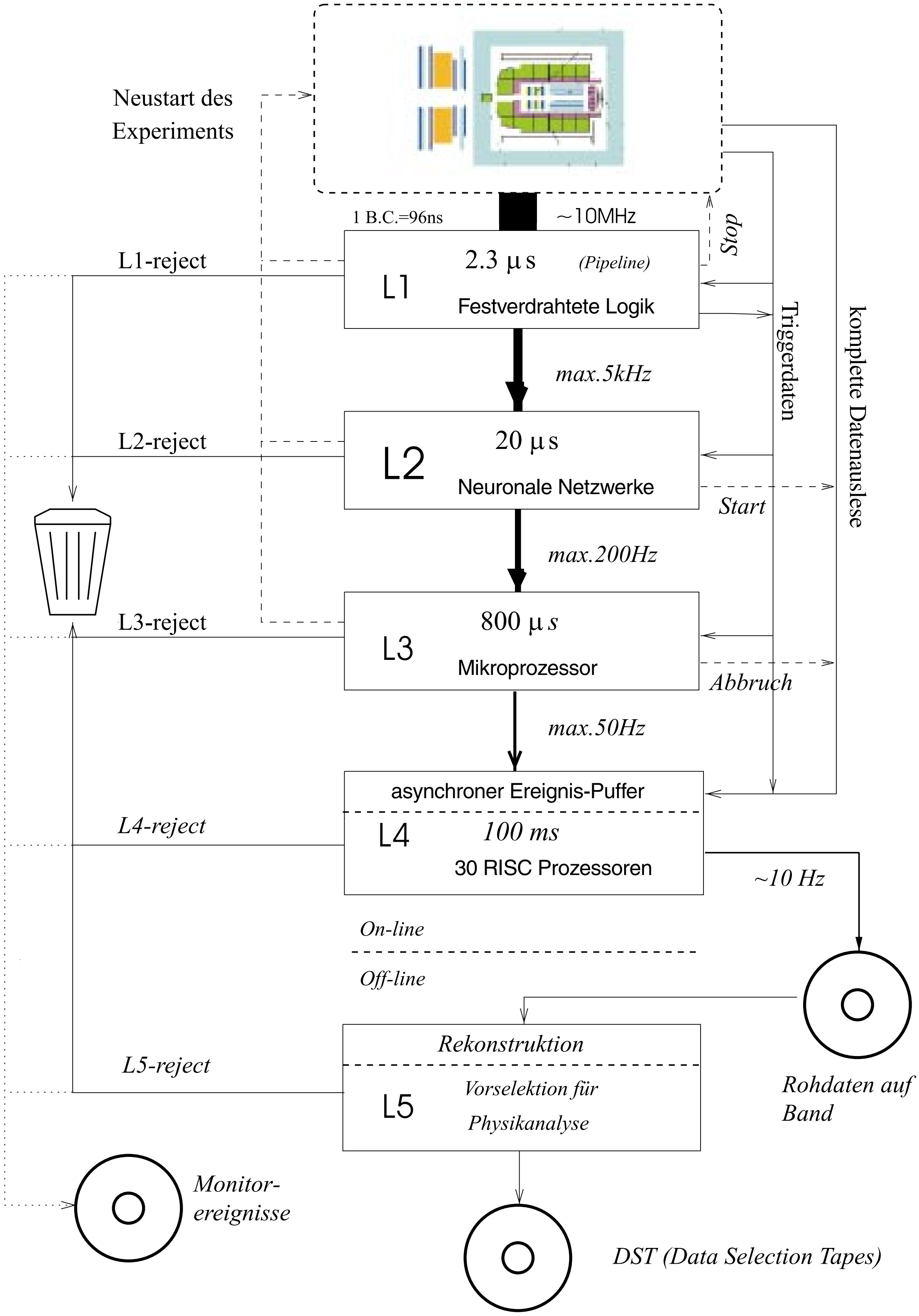

Bild 2.9: Datennahme- und Triggersystem bei H1. Signale sind gestrichelt; Datenflüße durchgezogen und gepunktet. Die angegeben Raten sind maximale Designwerte[15].

3 Der neuronale Netzwerktrigger bei L2

Der L2-Trigger ist ein neuartiger Trigger auf Basis neuronaler Netze. Mit dem L2-Trigger wurde zum ersten Mal ein Mustererkennungssystem mit neuronalen Netzen für einen allgemeinen Trigger in der Hochenergiephysik in Betrieb genommen. Der Trigger arbeitet seit 2 Jahren zuverlässig. Die erreichten Erhöhungen der Triggereffizienzen erlauben neuartige Physikanalysen [13].

3.1 Prinzipieller Aufbau

Der L1-Trigger ist für die primäre Ereignisauswahl zuständig. Er reduziert die gegebene Paketkollisionsfrequenz von rund 10 MHz in den kHz Bereich und darunter. Eine weitere Reduktion ist meist nur durch Prescaling erreichbar, was aber mit einer Absenkung der Luminosität erkauft werden muß. Der L2-Trigger soll nun diesen trade off vermeiden.

Auf L2 stehen die Triggerinformationen von allen Subdetektoren auf dem sogenannten L2-Datenbus zur Verfügung. Der Trigger hat einen Entscheidungzeitraum von 20 µs, womit die Totzeit kleiner als 10% ist, für verworfene Ereignisse. Diese Totzeit entsteht durch folgenden Vorfall: Ein L1-Keep-Signal hält das Experiment an, um die Datenauslese zu ermöglichen. Verwirft L2 das Ereignis, verhindert das den Beginn der Datenauslese und reduziert die durch das Auslesen bedingte Totzeit um ~100 ms. Dem Experiment geht also nur die Zeit bis zur L2 Entscheidung verloren.

Die Grundidee des L2 Triggers ist die Ausnutzung von Korrelationen zwischen Triggergrößen. Der L2-Trigger soll diese Muster erkennen, und so physikalisch relevante Ereignisse von Untergrundereignissen trennen. Für eine solche Mustererkennung sind neuronale Netze besonders geeignet.

Neuronale Netze arbeiten besser, wenn die ihnen gebotenen Eingangsgrößen speziell für das jeweilige Problem ausgewählt werden. Aus diesem Grund besteht der neuronale Algorithmus hardwaremäßig aus zwei Teilen: einer Vorverarbeitung der Eingangsgrößen und der Hardware für die eigentlichen neuronalen Berechnungen.

Die Vorverarbeitung übernehmen spezielle Boards, die *Data Distribution Boards (DDB)*, welche eigens entwickelt wurden. Die Triggergrößen können hier durch Transformation und Verknüpfung zu geeigneten Eingangsgrößen umgewandelt werden. Die so aufbereiteten Eingangsgrößen werden für die neuronalen Berechnungen in *CNAPS*[4] Parallelrechner geleitet. Der CNAPS kann den Netzoutput aus bis zu 64 Inputvariablen und 64 versteckten Knoten innerhalb von 9µs berechnen.

3.2 Aufgaben

Die Aufgabe des L2-Triggers besteht darin, die vom L1-Trigger vorgegebene Ereignisrate um mindestens eine Größenordnung zu reduzieren. Da L3 noch nicht genutzt wird, soll die Rate nach L2 in den Bereich von 30 Hz kommen.

- Untergrundnahe Ereignisse sollen bei geringer Statistik mit hohen Raten getriggert werden.
- Untergrund soll aufgrund topologischer Information verworfen werden.
- L1-Trigger sind teilweise empfindlich gegenüber Strahlbedingungen. Hier sollen die schwankenden L1 Raten stabilisiert werden.

Die Vorteile beim Einsatz neuronaler Netze zur Bewältigung dieser Aufgaben:

- Neuronale Netze sind flexibel: Es ist nicht nötig sich verschiedene Algorithmen für verschiedene Probleme auszudenken. Das Netz „denkt“ wodurch eine Hardware für verschiedene Lösungen eingesetzt werden kann.
- Netze können versteckte Korrelationen finden, von deren Existenz noch niemand weis, und so auch keinen speziellen Algorithmus dafür entwickeln könnte.
- Das Training geht schnell.
- Netze sind massiv parallelisierbar und können daher in Hardware sehr schnell ausgeführt werden.

Diese Vorteile müssen mit ein paar Einschränkungen erkauft werden:

- Für das Training wird eine hohe Statistik benötigen.
- Die versteckten Korrelationen gibt das Netz nicht preis. Das bedeutet: es ist nicht einfach herauszufinden, wieso das Netz mehr oder weniger gut trennt.

Das Konzept des L2NN-Triggers sieht vor, für bestimmte physikalische Reaktionen optimierte Netze parallel und unabhängig zu berechnen, die jeweils auf eine spezielle Auswahl von L1-Triggerdaten zurückgreifen. So werden einfache Netze möglich, und es wird eine gute Modularität erreicht.

---

[4] CNAPS: **C**onnected **N**etwork of **A**daptive **P**rocessors

3.3 Künstliche neuronale Netze

Die Beschreibung von künstlichen neuronalen Netzen (KNN) beginnt in vielen Schriften mit dem Vergleich zur Biologie und dem Wunsch des Menschen sein eigenes Denkvermögen auf Rechenmaschinen zu übertragen. Während die Analogie zur Biologie gerade auch aus der Geschichte der KNN naheliegt, so ist es doch noch ein sehr weiter Weg bis die künstliche Intelligenz der menschlichen ebenbürtig ist.

Ein mehr pragmatischer Ansatz sieht die KNN als relativ einfache mathematische Modelle – oder deren Implementierung – mit der Möglichkeit, Zusammenhänge oder Muster in hochdimensionalen Räumen zu erkennen. Genau diese Fähigkeit wird in der Teilchenphysik und damit bei H1 benötigt.

3.3.1 Das Modell

Motiviert durch die Biologie, entwarfen die beiden Amerikaner Warren McCulloch und Walter Pitts im Jahre 1943 [21] ein vereinfachtes Modell realer Vorgänge in neuronalen Strukturen. Ein Ergebnis dieser Untersuchungen war das Grundmodell eines Neurons, bei dem das Neuron als eine Art Addierer mit Schwellenwert betrachtet wird.

Neuronen haben Verbindungen, mit denen sie Aktivierungen mit jeweils einer bestimmten Stärke (*Gewicht*) von anderen Neuronen aufnehmen können. Daraus wird am Ausgang eines Neurons durch Summieren eine *Aktivität* erzeugt; unter der Bedingung, daß die Summe einen vorher festgelegten *Schwellenwert* überschreitet. Beim McCulloch/Pitts Modellneuron wurden noch alle Gewichte als gleich angenommen. Mit diesem vereinfachten Ansatz ist es jedoch bereits möglich, jeden finiten logischen Ausdruck zu berechnen.

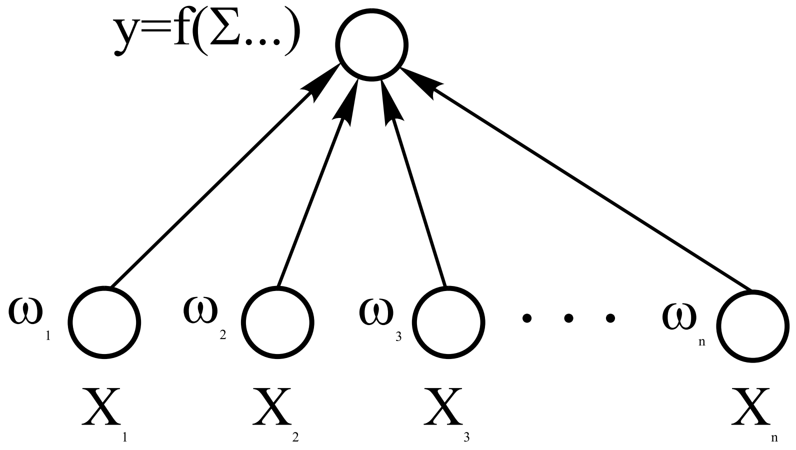

Bild 3.1: Das mathematische Neuron

So läßt sich ein formales Neuron folgendermaßen definieren [19]:

> Ein formales Neuron v ist ein Tupel (x,ω,z,y) aus Eingabevariablen x, Gewichten w, einer Aktivitätsfunktion z und einer Ausgabefunktion y(z).

Ein Neuron besitzt also ein oder mehrere Eingänge, die über eine Aktivitätsfunktion mit den Gewichten bewertet werden und durch die Ausgabefunktion den Ausgabewert erzeugen. Die Ausgabefunktion berechnet sich durch Multiplikation der Eingangswerte mit den Gewichten, und anschließender Subtraktion des Schwellenwertes. Die Aktivitätsfunktion läßt formal als Skalarprodukt schreiben:

$$z(\omega, x) = \sum_j \omega_j x_j - t_j = \vec{\omega}^T \vec{x} - \vec{T} \tag{3.15}$$

Als Ausgabefunktionen werden eindimensionale beschränkte Funktionen verwendet. Zum Einsatz kommen hierbei die Stufenfunktion, Rampenfunktion, sigmoide Funktionen oder Gauß-glockenförmige Funktionen. Bei L2NN wird eine Sigmoidfunktion folgender Form verwendet.

$$f(z) = \frac{1}{1+e^{-z}} \tag{3.16}$$

Diese Fermifunktion ist differenzierbar und hat einen dynamischen Übergangsbereich mit asymptotischen Sättigungsverhalten. Sie ist so einfacher zu handhaben als die ähnliche Stufenfunktion. In Kapitel 3.3.3 wird gezeigt, wieso die Funktion differenzierbar sein muß.

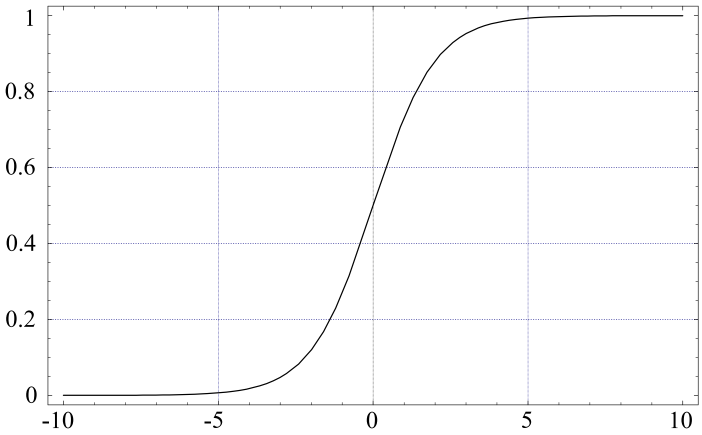

Bild 3.2: Die Fermifunktion

Die gesamte Ausgabefunktion für das mathematische Modellneuron hat somit diese Gestalt [20]:

$$y_i = \mathrm{f_i}\left(\sum_j \omega_{ij} x_j - t_i\right) \tag{3.17}$$

Mit diesen Modellneuronen kann man ein neuronales Netz definieren [19]:

> Ein neuronales Netz ist ein gerichteter Graph G := (K,E) aus der Knotenmenge K={v} aller formalen Neuronen und der Kantenmenge E aller Verbindungen zwischen den formalen Neuronen.

Wie genau diese Verbindungen (die Kanten des Graphen) aussehen, legt die Definition nicht fest. Hier gibt es grundsätzlich zwei Möglichkeiten: Die Verbindungen bilden

Zyklen oder nicht. Das Netz wird vorwärtsgerichtet genannt, wenn der Graph keine Zyklen enthält, und rekursiv, wenn er Zyklen enthält.

Im L2NN werden nur vorwärtsgerichtete (feedforward) Netze verwendet. Ein wichtiger Grund hierfür ist die maximale Zeit, welche für die Verarbeitung zur Verfügung steht. Bei zyklischen Netzen ist der Berechnungsprozeß nicht mehr durch die Vernetzung allein eindeutig festgelegt. So kann die Ausgabe eines Knotens auch als Eingabe im selbst zugeführt werden. Diese rekursive Berechnung bedarf einer Abbruchbedingung. Das Ergebnis nach einer bestimmten Zeit ist vorher nicht genau bekannt.

Sind in einem Netz keine Rückkopplungsschleifen vorhanden, fließt also die Information zwischen den Knoten in eine festgelegte Richtung, dann ist der Berechnungsprozeß eindeutig. Eine bestimmte Eingabe führt zu einer eindeutigen Ausgabe [18].

3.3.2 Feedforward Netze

Feedforward Netze werden auch Mehrschicht-Perzeptron oder Vorwärtsvermittlungsnetze genannt. Ein neuronales Netz wird genau dann als *feedforward* bezeichnet, wenn der gerichtete Graph *zyklenfrei* ist [19]. Dies bezieht sich normalerweise ausschließlich auf die Signalflußrichtung in der Funktionsphase. In der Lernphase hingegen ist diese Richtung nicht festgelegt.

In Feedforward Netzen lassen sich die Neuronen in Schichten (*Lagen*) einteilen. Dabei gibt es immer eine Eingabeschicht, eine Ausgabeschicht und ein oder mehrere versteckte Schichten. Bei L2NN wird ein dreilagiges Netz mit einer versteckten Schicht und einem Ausgabeneuron benutzt.

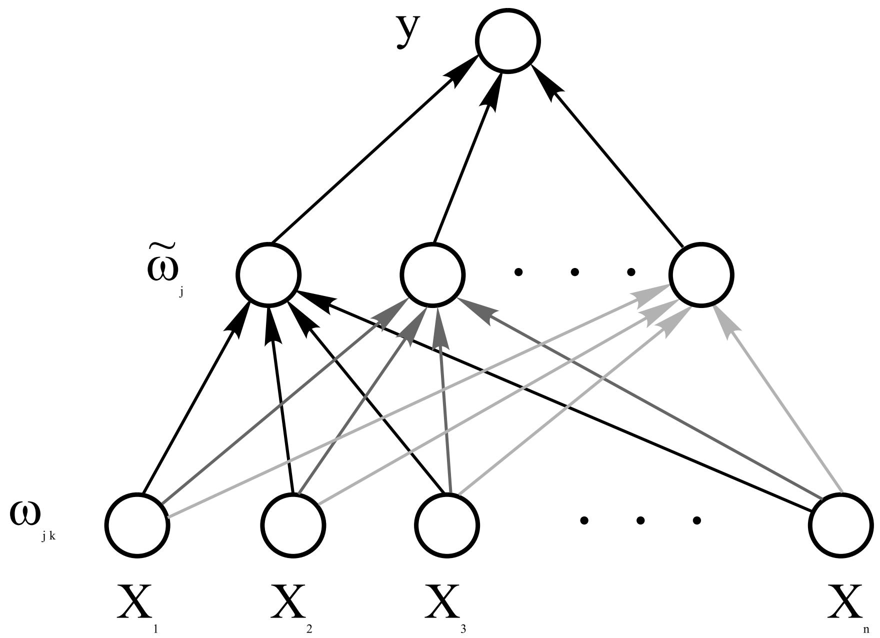

Bild 3.3: Datenfluß in einem Feedforward Netz mit einer versteckten Schicht und einem Ausgabeneuron.

Die Ausgabefunktion für ein dreilagiges Feedforward Netz hat somit folgende Gestalt:

$$y = \mathrm{f}\left(\sum_{j=1}^{n} \tilde{\omega}_j \mathrm{f}\left(\sum_{k=1}^{m} \omega_{jk} x_k - t_j\right) - \tilde{t}\right) \tag{3.18}$$

y Ausgabewert

f Aktivitäts- oder Transferfunktion

n Anzahl Knoten der versteckten Schicht

m Anzahl Eingabeknoten (Eingangsvariablen)

Enthält ein Feedforward Netz keine versteckte Schicht (Perzeptron), lassen sich damit nur linear separierbare Probleme lösen [23]. Beliebige stetige Funktionen $\mathrm{R}^n \rightarrow \mathrm{R}^m$ können realisiert werden, wenn eine versteckte Schicht vorhanden ist [26][27][28][29]. Dies kann anhand eines allgemeinen Theorems von Kolmogorov [25] gezeigt werden.

Feedforward Netze werden bei L2NN eingesetzt um Repräsentanten von verschiedenen Wahrscheinlichkeitsverteilungen zu klassifizieren. Da sich diese Wahrscheinlichkeitsverteilungen (Verteilung von Physik und Untergrund) überlappen, läßt sich die Wahrscheinlichkeit einer Fehlklassifikation nicht beliebig reduzieren [24].

Sei der Ereignisraum durch n Klassen K_i paarweise disjunkt zerlegt, dann ist die bedingte Wahrscheinlichkeit $P(K_i|x)$ – bei Beobachtung von x die liegt Klasse K_i vor – gegeben (Bayessche Regel) [24]:

$$P(K_i \mid x) = \frac{P(K_i)P(x \mid K_i)}{\sum_{i=1}^{n} P(K_i)P(x \mid K_i)} \tag{3.19}$$

Eine minimale Wahrscheinlichkeit für eine Fehlklassifikation läßt sich dadurch erreichen, indem man x der Klasse K_i mit der größten bedingten Wahrscheinlichkeit zuweist [24]. Im allgemeinen sind die Wahrscheinlichkeitsverteilungen jedoch nicht bekannt und so läßt sich Gleichung (3.19) nicht direkt zur Klassifizierung benutzen.

Mit Feedforward Netzen ist es möglich eine optimale Approximation für Gleichung (3.19) zu finden. Voraussetzung dafür sind geeignete Gewichte, die durch eine als „überwachtes Lernen“ bezeichnete Methode gefunden werden. Die Kenntnis über Wahrscheinlichkeitsverteilungen kann sich dabei auf einen Satz von Beispielpaaren beschränken. Dieser Satz ist die noch öfter erwähnte so wichtige Statistik an Trainingswerten für das Netz. Je größer die Statistik ist, desto besser ist die Güte der erreichbare Approximation.

3.3.3 Lernen mit dem Backpropagation-Algorithmus

Bisher ist klar geworden, daß die Gewichte entscheidende Bedeutung für die Ausgabe eines neuronalen Netzes haben. Diese Gewichte können nicht direkt berechnet werden, wie der vorige Abschnitt gezeigt hat. Vielmehr muß das Netz diese Gewichte „lernen“. Dies geschieht durch Training, womit ein Optimierungsprozeß bezeichnet wird, der die

Gewichte an die gestellte Aufgabe anpaßt. Hierzu werden Paare von Eingabe- und gewünschten Ausgabevektoren benötigt – die sogenannten Trainingsdaten. Die *gewünschten* Ausgabevektoren werden auch als Targetvektoren bezeichnet.

Der Backpropagation-Algorithmus (BPA) von Rumelhardt und McClelland [30][31] minimiert den globalen quadratischen Fehler der Ausgabevektoren y in bezug auf die Targetvektoren t mit der Gradientenmethode der numerischen Mathematik:

$$E(\omega,\tilde{\omega}) = \sum_{\mu} \left\| \vec{y}^{\mu} - \vec{t}^{\mu} \right\| \tag{3.20}$$

Hierbei ist eine differenzierbare Transferfunktion nötig; das ist die in Kapitel 3.3.1 eingeführte Fermifunktion.

Der BPA geht nun folgendermaßen vor: Beginnend mit der Ausgabeschicht „werden die Gewichte der einzelnen Schichten proportional zu Gradienten des globalen Fehlers abgeändert“ [24]. Zu jedem Trainingsmuster x berechnet das Netz die Aktivitäten h der versteckten Schicht und den Ausgabewert y. So ergibt sich die Änderung der Gewichte zu [24]:

$$\Delta\tilde{\omega}_{ij} = -\eta \frac{\partial E}{\partial \tilde{\omega}_{ij}} = \eta \sum_{\mu} \tilde{\delta}_i^{\mu} h_j^{\mu} \tag{3.21}$$

mit

$$\tilde{\delta}_i^{\mu} = (y_i^{\mu} - t_i^{\mu}) \cdot g'(\vec{\tilde{\omega}}_i \vec{h}) \tag{3.22}$$

Die Änderung der Gewichte in der versteckten Schicht werden in analoger Weise berechnet. Der Fehler kann zur versteckten Schicht mit den Berechnungen der Ausgabeschicht zurückverfolgt werden; daher der Name Backpropagation-Algorithmus.

$$\Delta\omega_{jk} = -\varsigma \frac{\partial E}{\partial \omega_{jk}} = -\varsigma \sum_{\mu} \delta_j^{\mu} x_k \tag{3.23}$$

mit

$$\delta_j^{\mu} = \sum_{i=1}^{m} \tilde{\omega}_{ij} \tilde{\delta}_i^{\mu} g'(\vec{\omega}_j \vec{x}^{\mu}) \tag{3.24}$$

In der bei L2NN verwendeten Abwandlung des BPA mit der Bezeichnung Momentum-BPA werden die obigen Gleichungen für die Änderung der Gewichte jeweils um einen Term ergänzt:

$$\Delta\tilde{\omega}_{ij} = -\eta \frac{\partial E}{\partial \tilde{\omega}_{ij}} + \underbrace{\alpha \Delta\tilde{\omega}_{ij}^{alt}}_{Momentumterm} \quad \text{und} \quad \Delta\omega_{jk} = -\eta \frac{\partial E}{\partial \omega_{jk}} + \underbrace{\alpha \Delta\omega_{jk}^{alt}}_{Momentumterm} \tag{3.25}$$

α	Momentumfaktor, bestimmt den Einfluß der vergangen Lernschritte auf den aktuellen
$\Delta\omega_{jk}^{alt}$, $\Delta\tilde{\omega}_{ij}^{alt}$	Gewichtsänderungen der letzten Iteration

Mit dem Momentum-BPA können Plateaus in der Fehlerlandschaft schneller überwunden, Oszillationen gedämpft und ein stabileres Lernverhalten erzielt werden. Der Momentumfaktor muß für das jeweilige Problem empirisch ermittelt werden.

Der Lernparameter η legt fest, wie stark sich die Gewichte bei jedem Lernschritt verändern sollen. Er wird auch als Lernrate bezeichnet und muß der Problemstellung empirisch angepaßt werden.

Ein Training startet mit der Initialisierung der Gewichte. Der naheliegende Ansatz, alle Gewichte auf den selben Wert zu setzen funktioniert nicht. Bei L2NN werden die Gewichte pseudozufällig und reproduzierbar initialisiert:

$$\omega_{jk} \in [-\varepsilon, \varepsilon] \ \text{ mit } \ \mathring{a} = \frac{3}{\sqrt{m}} \tag{3.26}$$

und

$$\tilde{\omega}_{ij} \in [-\varepsilon, \varepsilon] \ \text{ mit } \ \mathring{a} = \frac{3}{\sqrt{n}} \tag{3.27}$$

Diese Wahl basiert auf Untersuchungen von Wessels und Barnard [32], die gezeigt haben, daß so ungünstige lokale Minima vermieden werden. Sind die Startgewichte, Trainingsmuster und die Trainingsparameter gegeben, kann ein einmal durchgeführtes Training jederzeit reproduziert werden.

3.3.4 Bewertungskriterien von KNN

Neuronale Netze werden bei L2NN eingesetzt, um die beiden Klassen, Physikereignisse und Untergrundereignisse zu trennen. Bei der Entwicklung eines neuen Netzes für ein spezielles physikalisches Problem, müssen verschiedene Netze mit unterschiedlichen Eingangsgrößen ausprobiert werden. Um ihre Qualität beurteilen zu können, benötigt man ein einfaches Hilfsmittel, das quasi auf einen Blick zeigt, welches das bessere Netz ist. Dieses Hilfsmittel ist die Effizienzkurve.

Die Effizienzkurve stellt die Physikeffizienz ε_P gegen die Untergrundeffizienz ε_U dar. Die Effizienzen sind die Anteile der korrekt klassifizierten Ereignisse. Man erzeugt die Effizienzkurve in zwei Schritten. Zunächst wird die Verteilung der Ausgabewerte des Netzes ermittelt. Das ergibt zwei Kurven, die bereits einen ersten Eindruck davon vermitteln, wie gut die beiden Klassen getrennt werden. Beim zweiten Schritt wird zu jedem Ausgabewert des Netzes die Physikeffizienz gegen die Untergrundeffizienz aufgetragen.

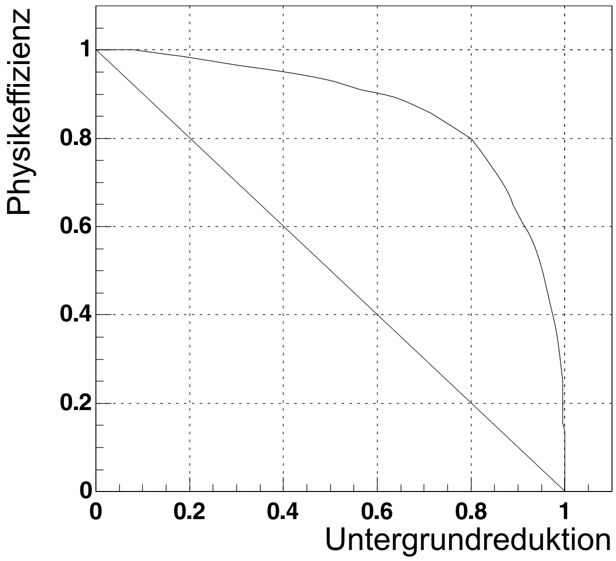

Bild 3.4: Die Effizienzkurve gibt die Physikeffizienz für jede Untergrundreduktion an. Die 45° Linie ist der sogenannte Prescale. Der Prescale ist die Kurve für die Physikeffizienz, die sich ergibt, wenn man nur jedes n-te Ereignis akzeptiert.

Es läßt sich gut erkennen, daß ein Netz um so besser ist, je weiter sich die Kurve der rechten oberen Ecke nähert. Daraus läßt sich ein Bewertungskriterium ableiten: die *Physikeffizienz bei geforderter Reduktion der Untergrundrate*. Die Reduktion der Untergrundrate R ist:

$$R = \frac{1}{1-\varepsilon_U} \tag{3.28}$$

Zusammen mit der Effizienzkurve kann man einem geforderten Reduktionsfaktor eine Physikeffizienz zuordnen. Somit lassen sich verschiedene Netze, aufgrund der erreichten Physikeffizienz bei festgelegtem Reduktionsfaktor, vergleichen.

Ebenfalls aus der Effizienzkurve gewinnt man ein weiteres Kriterium: die *beste mittlere Effizienz (BME)*. Das „ist der maximale Mittelwert aus Physik- und Untergrundeffizienz, wie er sich bei geeigneter Wahl des Schnittes ergibt“[24].

$$BME = \max\left(\frac{\varepsilon_P + \varepsilon_U}{2}\right) \tag{3.29}$$

Damit stehen mehrere Bewertungskriterien für neuronale Netze zur Verfügung, die je nach Anforderung eingesetzt werden können. Dabei sollten diese Kriterien nicht auf Netzausgaben zu den Trainingsdaten angewendet werden, sondern auf Testmustern, die nicht schon im Training eingesetzt wurden.

Um schon nach dem Training einen ersten Eindruck von dessen Erfolg zu bekommen, bietet sich der mittlere quadratische Fehler als Kriterium an. Mit der Zahl der Trainingsmuster M ist das:

$$MSE_{Train} = \frac{1}{M} \sum_{\mu=1}^{M} \left\| \vec{y}^{\mu} - \vec{t}^{\mu} \right\|^2 \qquad (3.30)$$

Trägt man diesen MSE gegen die Anzahl trainierter Epochen auf, erhält man eine Fehlerkurve, mit Hilfe derer man einfach einen Satz von Netzen direkt vergleichen kann. Dies wird besonders dafür eingesetzt, die optimale Lernrate und Momentumfaktor herauszufinden (siehe Kapitel 3.5.3).

3.4 Hardware

Der L2-Trigger besteht aus 12 sogenannter Triggerboxen. Damit lassen bis zu 12 neuronale Netze als Trigger für unterschiedliche physikalische Problemstellungen einsetzen. Eine Triggerbox enthält als Hauptkomponenten ein DDB und ein CNAPS-Board. Diese beiden Teile sind in Crates[5] untergebracht; einem DDB-Crate mit dem L2-Bus auf der Rückseite und dem CNAPS Crate, einem D32/A32 VME Crate.

Der 128 Bit breite L2-Bus befördert die Daten vom L1-Trigger zum L2-Trigger. Dort werden sie von einem PQZP[6] Receiver aufgenommen und an eine DDB weitergeleitet. Diese selektiert die Daten und bereitet sie für die weiteren Berechnungen auf. Eine DDB besteht aus einer Mutterplatine und bis zu vier Tochterplatinen. Die Mutterplatine erledigt die Konfigurations- und Kommunikationsaufgaben zum VME-Bus und zum CNAPS; die Tochterplatinen die Datenselektion und –aufbereitung zweier L2-Subbusse.

Die DDB spaltet ein Datenwort in sein hohes und niederes Byte auf. Diese beiden Bytes auf die für die Errechnung einer Eingabegröße des Parallelrechners notwendige Struktur abgebildet. Solche Funktionen sind: Bitpositionsbestimmung, Bitzähler, Überlaufbehandlung, und ein Bitzähler für ein spezifisches Datenformat des SpaCal-Kalorimeters. Das Datenwort kann ebenso auch transparent weitergereicht werden. Eine Addierer-Multiplexer-Schaltung erzeugt die Eingabegrößen für den Parallelrechner. Die Werte können addiert, logisch verknüpft oder einfach im Output-RAM abgelegt werden. Ungefähr 6-7 μs nach einem L1-Triggersignal – unmittelbar nach der Selektion – erfolgt die Weiterleitung zum CNAPS.

Auf den CNAPS-Boards befinden sich 64 Prozessoren; auch Processor Node (PN) genannt. Ein PN repräsentiert ein Neuron. Dazu kann es Berechnungen in Festkommaarithmetik durchführen, und verfügt über 4 kByte lokalen Speicher für Gewichtstabellen und Look-Up-Tables. Ein gemeinsamer Bus versorgt alle Prozessoren mit identischen Instruktionen. Sie werden jedoch mit unterschiedlichen Daten initialisiert. Ein Multiplikations-Addier-Schritt wird in einem Taktzyklus ausgeführt. Die Rechenzeit steigt so-

[5] Crate: Gehäuse mit Einschubfächern, Steckerleisten und Stromversorgung

[6] PQZP: **P**arallel **Q**uickbus **Z**ero-suppression **P**rocessor

mit linear mit der Zahl der Knoten, und nicht quadratisch. Ein Feedforward-Schritt eines 64×64×1 Netzes kann in 166 Taktzyklen oder 8 µs berechnet werden.

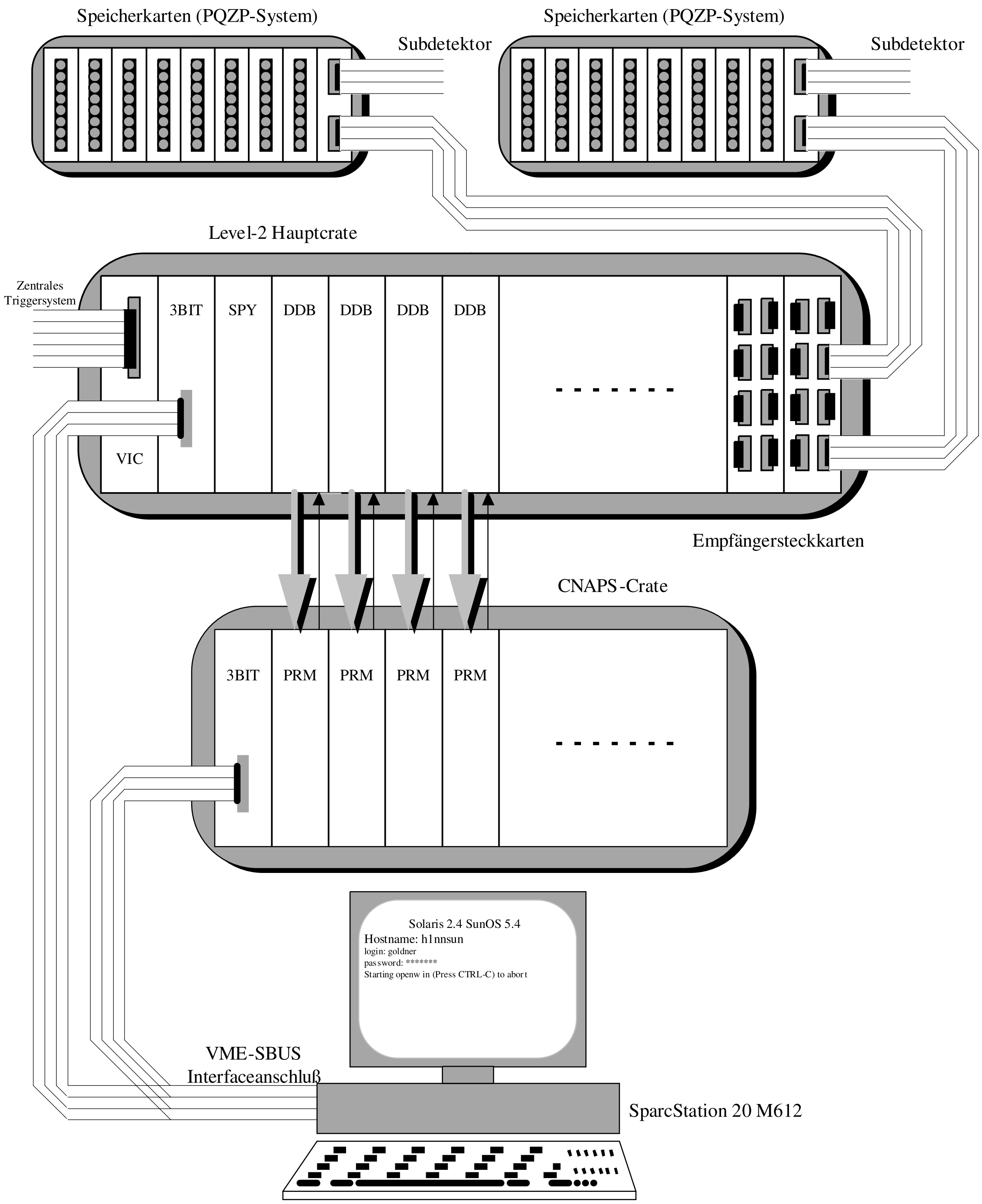

Bild 3.5: Prinzipieller Aufbau der Hardware des L2-Triggers

3.5 Netztraining

Bevor ein neuronales Netz in der Hardware zum Einsatz kommt, muß es erst für den jeweiligen Einsatzzweck entwickelt und trainiert werden. Wie dies prinzipiell algorithmisch geschieht wurde in Abschnitt 3.3 gezeigt. In der Praxis wird dazu eine Folge von Programmen – die Trainingskette – benutzt [17].

Doch bevor damit begonnen werden kann, mit den Programmen der Kette Netze zu trainieren, müssen Daten, d.h. Ereignisse (Events) vorhanden sein. Dabei gibt es zwei mögliche Quellen:

- Mit Monte-Carlo generierte Ereignisse
- Reale Daten vom Trigger aus vergangenen Datennahmeperioden.

Monte-Carlo Daten erzeugt man mit einem Generator, der jeweils für das gewünschte physikalische Problem Ereignisse simuliert. Das Ergebnis ist eine Datei mit einer bestimmten Anzahl Vierervektoren zu den Events. Diese werden nun einer Detektorsimulation zugeführt, um festzustellen, wie der Detektor auf die Ereignisse reagieren würde. Anschließend werden die Teilchen rekonstruiert. Das Ergebnis dieser Simulation-Rekonstruktion[7] ist eine DST[8] Datei.

Die Realdaten stehen sofort im DST Format zur Verfügung. Ab jetzt werden Monte-Carlo und Realdaten gleich behandelt.

Für eine erste Auswertung kann man die DST Dateien im Event Display betrachten. Als nächsten Schritt sollen aus allen vorhandenen Ereignissen ganz bestimmte ausgewählt werden. Hierzu erzeugt man aus der DST-Datei eine hBook-Datei[9]. Somit steht ein Ntuple zur Verfügung auf das man die gewünschten Cuts (Selektionen) anwenden kann. Dies wird mit Hilfe der hBook-Datei[10] gemacht, in dem durch Anwendung von Makros jedes einzelne Ereignis auf die Selektionsbedingungen hin geprüft wird. Sind die Bedingungen erfüllt, wird die jeweilige Runnunmber und Eventnumber in eine Datei geschrieben.

Damit sind zunächst alle Vorbereitung für ein Netztraining mit der Trainingskette gemacht. Alle Programme der Kette werden prinzipiell gleich bedient. Alle Parameter zur Steuerung der Programme sind in sogenannten Steering Dateien gespeichert. Diese werden beim Aufruf übergeben.

```
Programm < Steering-File > Log-File 2> Error-File
```

[7] Die Detektorsimulation und Rekonstruktion der Teilchen erledigt das Programm SimRec (Verzeichnis: /.h1/h1/h1sim/bin)

[8] DST: **D**ata **S**ummary **T**ape

[9] Die hBook Datei erzeugt das Programm phiphan.

[10] Einige H1-Mitglieder verwenden seit Neustem das Programm root für diese Auswertungen. Bevor die hBook Dateien mit root bearbeitet werden können, müssen sie mit dem Programm h2root in das root eigene Format umgewandelt werden.

Die Logging- und Fehlermeldungen werden in eigene Dateien umgeleitet, um den Verlauf der Programmabarbeitung nachträglich besser überprüfen zu können.

Eine detaillierte Beschreibung aller Programme mit Beispielen von Steering Dateien findet sich in [17].

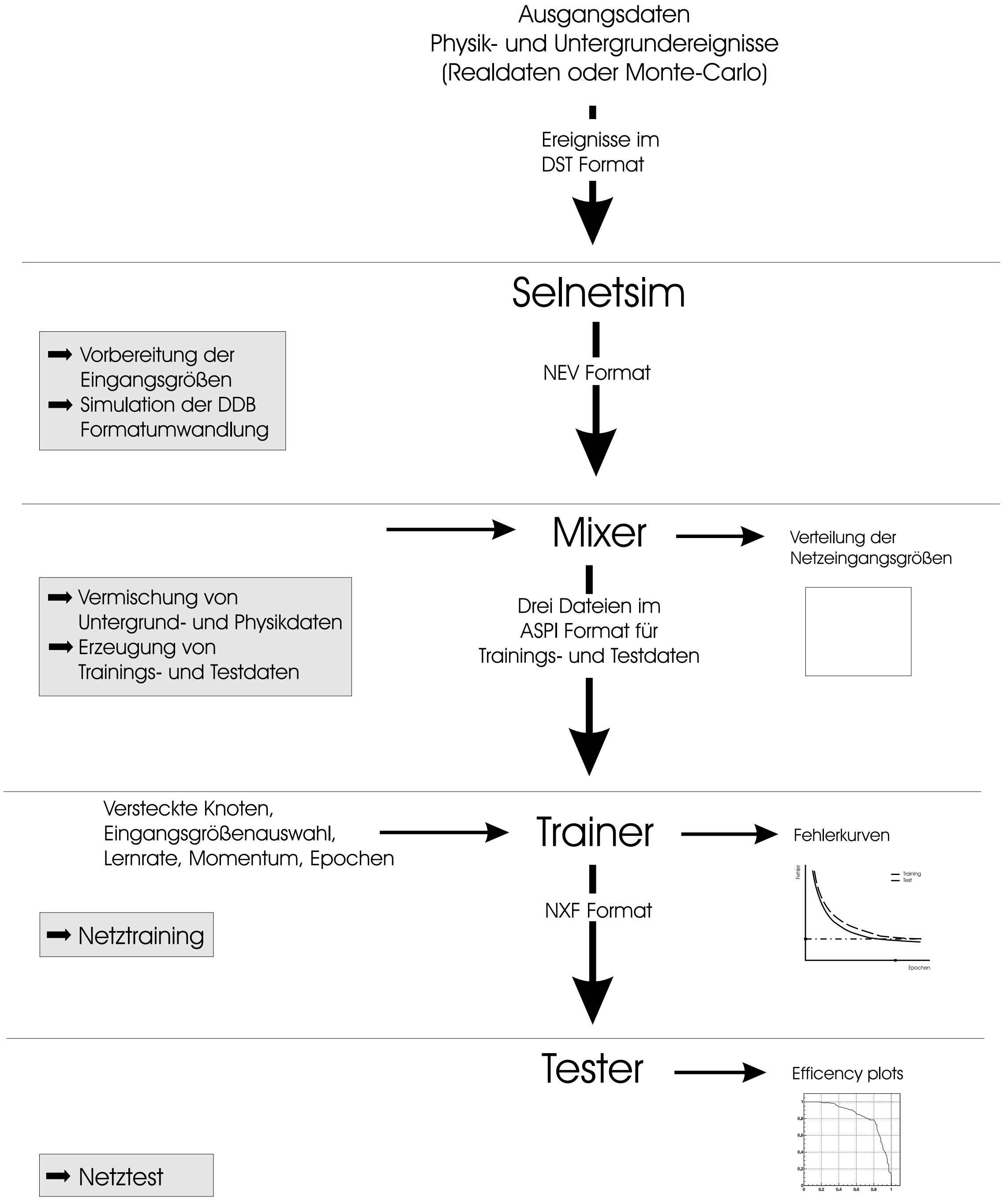

Bild 3.6: Die Trainingskette für neuronale Netze besteht aus vier Programmen. Sie haben jeweils spezialisierte Aufgaben und erzeugen verschiedene Ausgabedateien zur Weiterverarbeitung und Auswertung. Selnetsim konvertiert und stellt Variablen bereit. Mixer vermischt Untergrund und Physik. Der Trainer erledigt das eigentliche Netzwerktraining, dessen Qualität im Tester überprüft wird.

3.5.1 Selnetsim

Das Programm Selnetsim hat die Aufgabe, die Eingangsgrößen für das neuronale Netz zu berechnen. Durch Anpassen des Programms kann auch die Funktion der DDB oder DDB2 simuliert werden, was ein wesentlicher Teil dieser Arbeit ist (siehe dazu Kapitel **Fehler! Verweisquelle konnte nicht gefunden werden.**).

Selnetsim benötigt die DST-Datei (oder auch Dateien) mit den Events. Zusätzlich muß eine Liste mit Runnumber und Eventnumer der zu verwendenden Ereignisse vorliegen. Diese Liste muß Selnetsim mittels der Steering-Datei übergeben werden. Dies geschieht entweder durch Einfügen der Liste in die Datei, oder durch Aneinanderhängen der beiden Dateien:

```
cat steer-file runevent-file | selnetsim < Steering-File
> Log-File 2> Error-File
```

Das Ergebnis sind NEV[11] Dateien in gezippter Form. Diese Dateien enthalten die Formatbeschreibung und alle Daten des Eventvektors[12].

3.5.2 Mixer

Der Mixer generiert Dateien zum Trainieren und Testen von neuronalen Netzen. Es werden Dateien mit Trainings- und Testdaten benötigt, wobei Untergrund- und Physikereignisse genau definiert vermischt werden sollen. Der Mixer erzeugt diese Mischdateien für Training und Test. Der Mixer erlaubt noch die Auswahl bestimmter Netzeingangsgrößen für die Verwendung im Netz. So kann man sich durch Selnetsim Dateien mit praktisch allen vorhandenen Variablen erzeugen lassen und dann erst im Mixer bestimmen, welche verwendet werden sollen. Das spart Zeit, da die Ausführung von Selnetsim in der Regel deutlich länger dauert als die des Mixers.

Zusätzlich generiert der Mixer eine hBook-Datei mit den Verteilungen der ausgewählten Variablen. Damit lassen sich bereits erste Einschätzung über die Bedeutung der einzelnen Variablen (Eingangsgrößen) für das Training machen. Hier ist es sinnvoll ein einfaches Makro bereit zu haben, welches quasi auf Knopfdruck die einzelnen Plots übersichtlich auf einer Seite darstellt.

Der Mixer erzeugt als Output neben den hBook-Dateien typischerweise drei ASPI[13]-Dateien: eine Datei mit den gemischten Trainingsdaten, eine Datei mit Physikdaten und eine mit Untergrunddaten zum testen. Die Datei mit den Trainingsdaten wird vom Trainer verwendet, die beiden anderen vom Tester.

[11] NEV: **N**ew **E**vent **V**ector

[12] Eventvektor: Datenstruktur zum Austausch der Daten von Ereignissen zwischen verschiedenen Programmen und Programmteilen.

[13] ASPI: Format des Programms ASPIRIN.

3.5.3 Trainer

Die Aufgabe des Trainers ist es nun, das neuronale Netz zu trainieren, also einen Satz optimaler Parameter (Gewichte) für das Netz zu finden. Wie in Kapitel 3.3 beschrieben wird ein neuronales Netz durch seine Parameter – die Gewichte – bestimmt. Dafür wird der beschriebene Momentum-Backpropagation-Algorithmus verwendet. Pro Aufruf kann der Trainer mehrere Netze berechnen, die sich dann vergleichen lassen.

Der Trainer erhält als Input die vom Mixer generierten Trainingsdaten, bestehend aus Physikdaten und Untergrunddaten. In der Steering-Datei müssen einige Netzwerkparameter von Hand gesetzt werden, da der Trainer nicht in der Lage ist, diese zu berechnen. Zunächst muß entschieden werden, welche Eingangsgrößen das Netz verarbeiten soll und wie viele versteckte Knoten benutzt werden sollen.

Es gibt keine festen Regeln für den Wert dieser Größen; dieser muß immer erst durch mehrere Trainings- und Testzyklen herausgefunden werden. Es gibt jedoch Daumenregeln, die sich im Lauf der Zeit bewährt haben:

- Die Auswahl der Eingangsgrößen hängt vom Problem ab. Es sollen Größen verwendet werden, die eine unterschiedliche Verteilung für Physik und Untergrund aufweisen.
- Die Zahl der versteckten Knoten soll nicht größer als die Zahl der Eingangsgrößen sein, wenn die Eingangsgröße als Zahl und nicht als Bitmaske betrachtet werden soll.

Dabei ist aber zu beachten, daß es eben eine Stärke von neuronalen Netzwerken ist, Zusammenhänge in hochdimensionalen Parameterräumen zu finden, die von einem menschlichen Betrachter auf den ersten Blick nicht zu erkennen sind. So ist es möglich, daß eine Eingangsgröße wichtig für die Qualität des Netzes ist, obwohl die Verteilung von Physik und Untergrund nahezu identisch aussehen.

Um möglichst gute Ergebnisse zu erzielen ist es also unumgänglich verschiedene Parametersätze auszuprobieren und deren Qualität miteinander zu vergleichen. Neben den Gewichten der trainierten Netze liefert der Trainer auch eine hBook-Datei mit den Fehlerkurven der verschiedenen Netze, die berechnet werden sollten. Diese Fehlerkurven bieten Vergleichsmöglichkeiten um die optimale Lernrate und Momentumfaktor zu bestimmen. Damit wird auch derjenige Lernschritt (Epoche) mit dem niedrigsten Fehler ermittelt.

Bild 3.7 zeigt die Fehlerfunktionen eines optimalen Netztrainings. Dabei ist zu beachten, daß die Testkurve den niedrigsten Wert von allen verglichenen Netzen erreicht. Die Kurven sollten im wesentlichen „glatt“ sein, keine lokalen Minima aufweisen und zum Ende hin nicht wieder ansteigen. Das Netz könnte noch verbessert werden, indem man die Anzahl an versteckten Knoten reduziert oder den Lernparameter erniedrigt, unter der Bedingung, daß sich dabei der Fehler nicht erhöht.

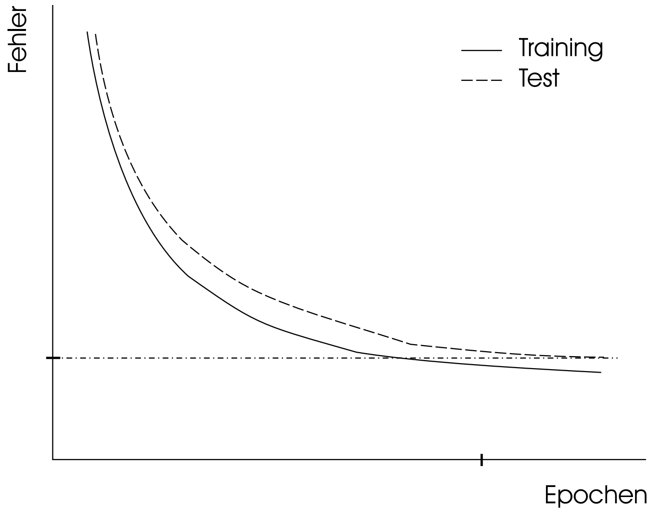

Bild 3.7: Fehlerfunktionen eines optimalen Netztrainings.

In Bild 3.8 ist die Fehlerkurve eines guten Trainings dargestellt. Jedoch steigt die Testkurve gegen Ende hin an. Das bedeutet, der niedrigste Fehler wurde schon zu einer früheren Epoche erreicht. Das Netz wurde „übertrainiert“. Hier kann eine Verringerung der versteckten Knoten oder der Lernrate zu einem besseren Ergebnis führen.

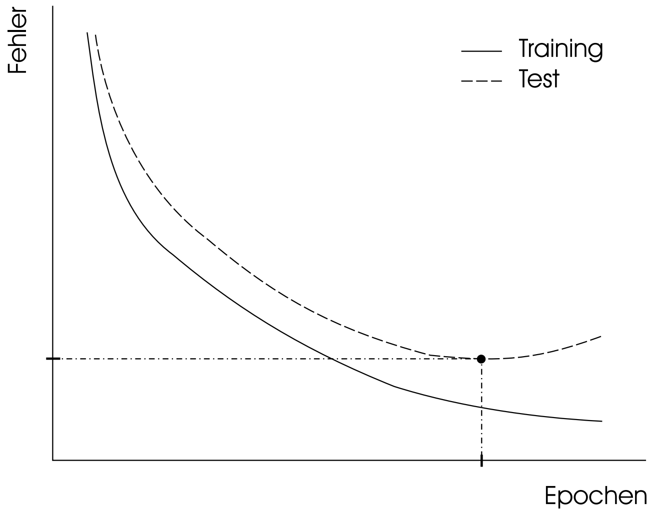

Bild 3.8: Fehlerkurve eines guten Trainings, welches aber noch durch Reduzierung der Anzahl an versteckten Knoten oder Verkleinerung der Lernrate verbessert werden könnte.

Bild 3.9 zeigt die Fehlerkurve für ein Training mit zu hoher Lernrate. Das Training ist unbrauchbar; es hilft nur ein niedrigere Lernrate.

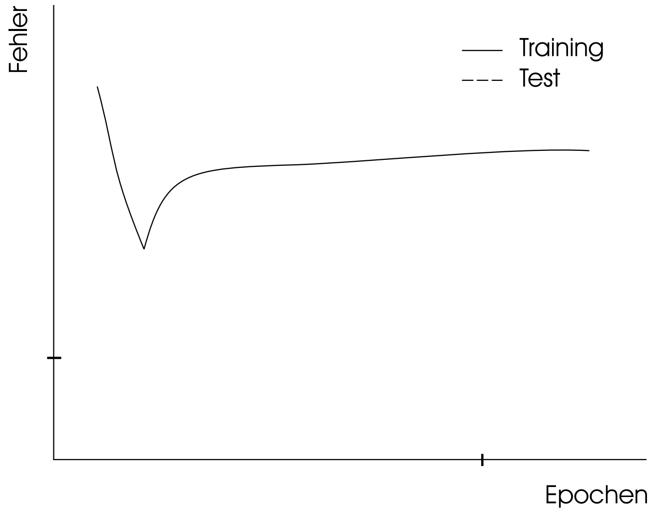

Bild 3.9: Fehlerkurve eines Training mit zu hoher Lernrate.

Die Fehlerkurve in Bild 3.10 ist das Ergebnis eines Trainings mit zu kleiner Lernrate und zu geringer Anzahl an Epochen. Die Kurven sind bei der letzten Epoche noch zu steil und lassen so vermuten, daß der niedrigste Fehler noch nicht erreicht ist.

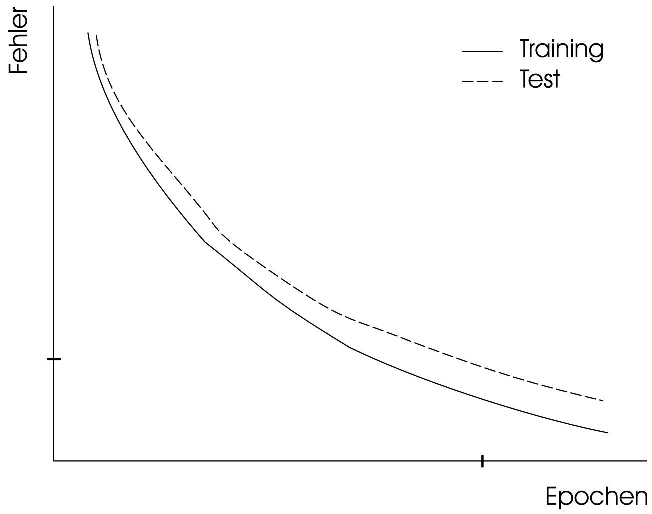

Bild 3.10: Fehlerkurve eines Trainings mit zu kleiner Lernrate und zu geringer Anzahl an Epochen

Die sehr bald wieder ansteigende Fehlerkurve der Testdaten in Bild 3.11 deutet darauf hin, daß die Lernrate zu hoch gewählt wurde und die Anzahl an Epochen zu hoch war.

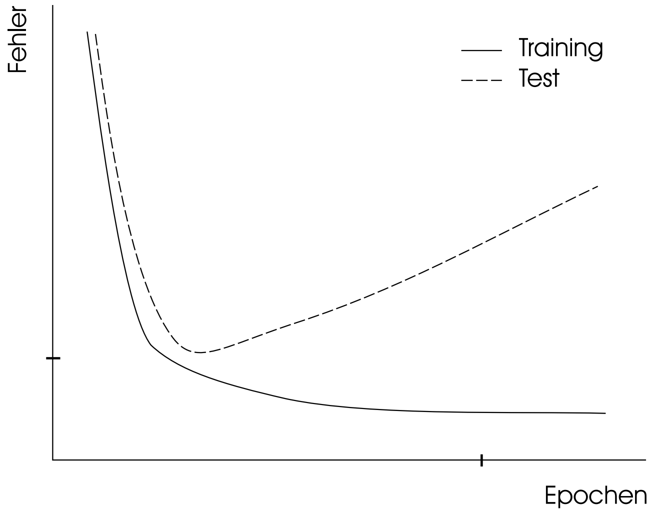

Bild 3.11: Fehlerkurve eines Training mit zu großer Lernrate und zu großer Anzahl an Epochen.

Die Entwicklung eines neuen Netzes beginnt üblicherweise mit einer ersten Auswahl an Eingangsgrößen. Mit diesen werden dann mit dem Trainer einige Netze mit jeweils unterschiedlichen Lernparametern, Momentumfaktor und eventuell Epochen trainiert. Erfüllt nach diesem ersten Versuch bereits eines der Netze in etwa die obigen Kriterien für ein optimales Netz kann mit Tester weitergearbeitet werden. Ansonsten muß die Prozedur mit – nach obigen Erfahrungsregeln – veränderten Parametern neu durchlaufen werden.

3.5.4 Tester

Mit dem Tester ist es möglich, die Qualität der vom Trainer berechneten Netze zu beurteilen. Dazu verwendet er die vom Mixer erzeugten Dateien mit den Daten für Physik und Untergrund. Diese Testdaten werden in das Netz eingegeben. Da Physik und Untergrund in verschiedenen Dateien gespeichert sind, kann der Tester feststellen, ob das Netz jeweils Physik und Untergrund richtig erkannt hat bzw. getrennt hat.

Der Tester erzeugt eine hBook-Datei, aus der sich mit einem Makro die Effizienzkurve erstellen läßt. Zur besseren Übersicht läßt man sich meist einen Satz von Plots erstellen. Diese Plots zeigen die Verteilungen der Ausgabewerte des Netzes für Physik- und Untergrundereignisse, die Untergrund- und Physikeffizienzen und die Effizienzkurve (siehe Kapitel 4). Wie sie zustande kommen wurde bereits in Kapitel 3.3.4 erläutert.

Besonders wichtig ist, daß diese Kurven aus Netzausgaben von Testmustern gewonnen werden. Es soll ja überprüft werden, wie gut das Netz auf Muster reagiert, die es noch nicht „gesehen“ hat.

4 Trigger von ϕ-Mesonen mit neuronalen Netzen

Das Triggerkonzept für ϕ–Mesonen umfaßt eine Bedingung für die erste Triggerstufe L1 und für den neuronalen Trigger L2. Dies geschieht, um auch die Photoproduktion von ϕ–Mesonen analysieren zu können. ϕ–Mesonen aus Elektroproduktion sind deutlich leichter zu triggern, da hierbei das gestreute Elektron im SPACAL Energie deponiert; bei der Photoproduktion entkommt das Elektron im Strahlrohr.

Bei der Elektroproduktion kann ein brauchbares Triggersignal durch das Elektron ausgelöst werden. Es ist für eine ausreichend große Ausbeute an ϕ–Mesonen aus Elektroproduktion kein Trigger auf L2 Ebene mehr nötig, und wird auch derzeit nicht eingesetzt. Bei ϕ–Mesonen aus Photoproduktion kann ein Triggersignal nur von den beiden geladenen Spuren (Kaonen) ausgelöst werden. Hinzu kommt die Schwierigkeit, daß die Zerfallskaonen einen relativ geringen Impuls (127 MeV/c im Ruhesystem des ϕ–Mesons) aufgrund ihrer, im Vergleich zum ϕ, relativ hohen Masse haben.

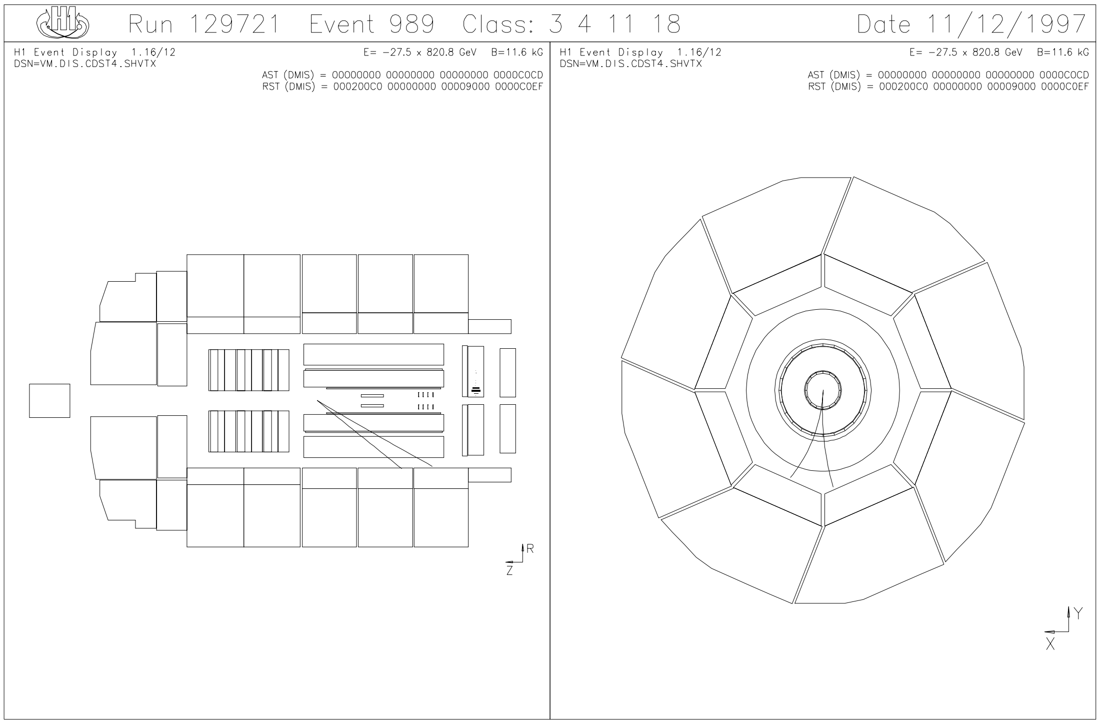

Bild 4.12: Ein typisches ϕ Ereignis im Eventdisplay. Gut erkennbar sind die Spuren der beiden Kaonen; aufgrund ihrer unterschiedlichen Ladung, sind die Spuren auch unterschiedlich gekrümmt.

Um die besonders schwierig zu triggernden ϕ–Mesonen aus Photoproduktion analysieren zu können, werden beide Triggerstufen, L1 und L2, benötigt. Dazu wurde zunächst eine effiziente L1-Bedingung gefunden.

4.1 Die L1-Bedingung für ϕ-Mesonen aus Photoproduktion

In [2] wurde eine L1-Bedingung erarbeitet, die unter der Bezeichnung Subtrigger 68 (S68) zum Ende der Datennahmeperiode 1996 aktiviert wurde. Die S68 Bedingung wird seit dem, zum Trigger von ϕ–Mesonen auf L1 Ebene verwendet:

```
(DCRPh_Ta&&zVtx_Cls&&!FwdMu_Fwd_Vtx&&LAr_IF==0&&Topo_Mult==2)&&"BGVETO"
```

Die einzelnen Triggerelemente haben folgende Bedeutung:

DCRPh_Ta

Die in der CJC gemessenen Spurelemente werden auf die rφ-Ebene projiziert. Die Projektionen werden mit vordefinierten Masken verglichen und so Triggersignale für unterschiedlichen Impuls und Ladung erzeugt. Das Triggerelement DCRPh_Ta verlangt, daß mindestens eine solche Maske angesprochen hat.

zVtx_Cls

In Kapitel 2.2.2.1 wurde gezeigt, wie aus den Spursignalen der MWPC ein z-Vertex Histogramm entsteht. Die Triggerbedingung zVtx_Cls verlangt, daß alle Einträge in dem z-Vertex Histogramm in vier benachbarten Bins liegen. Für Ereignisse mit zwei Spuren, wie dem ϕ–Meson, ist das sehr wahrscheinlich.

FwdMu_Fwd_Vtx

Wird in der ersten Lage des Vorwärtsmyonspektrometers ein Signalpaar aufgezeichnet, wird FwdMu_Fwd_Vtx gesetzt. Dieses Triggerelement wird als Veto verwendet, da für diffraktive Ereignisse keine Aktivität im Vorwärtsbereich erwartet wird.

LAr_IF==0

Diese Triggerbedingung verlangt, daß die Energie im vorderen Bereich (inner forword, IF) des LAr-Kalorimeters weniger als 2 GeV beträgt.

Topo_Mult==2

Diese Triggerbedingung verlangt, daß in genau 2 φ-Segmenten der $\varphi\theta$-Mappe ein oder mehrere Einträge vorhanden sind. Diese Mappe enthält Signale aus der MWPC und der Vorwärtsproportionalkammer und ist 16-fach in φ unterteilt und 14-fach in θ. Der zVtx-Trigger bildet aus der Mappe das Triggerelement Topo_Mult, welches aus vier Bit besteht, und die über θ integrierte Multiplizität der Mappe enthält.

BGVETO

Die Triggerbedingung repräsentiert verschieden Vetobedingungen, die für alle L1-Subtrigger gefordert werden. Diese liefert das Flugzeitsystem, oder Triggerkomponenten mit hoher Zeitauflösung. Auf diese Standardbedingungen soll hier nicht näher eingegangen werden.

Die mittlere Ausgangsrate des S68 liegt bei 5-15 Hz ([2]). Der zulässige Wert liegt bei etwa 1 Hz. Deshalb muß auf der Triggerstufe L2 die Rate weiter reduziert werden.

4.2 Bisherige Ergebnisse

In [2] gelang es erstmals, bei H1 ein Signal für ϕ–Mesonen aus Photoproduktion aufzuzeichnen. Dies wurde möglich durch die Kombination des im letzten Abschnitts vorgestellten Subtriggers 68, und dem Einsatz des, im folgenden genauer beschriebenen, neuronalen Netzes für die Triggerstufe L2.

Im vorigen Abschnitt wurde bereits darauf hingewiesen, daß ϕ–Mesonen aus Photoproduktion schwer zu triggern sind, und so gab es auch in den H1-Daten von 1995[14] auch keine solchen Ereignisse, die für ein Netzwerktraining zur Verfügung stehen könnten. Möchte man aber gerade diese bestimmten ϕ–Mesonen bekommen, steht man vor einem Dilemma: Zum Einen benötigt man für das Netztraining Beispieldaten, zum Anderen sind diese Daten aber nicht vorhanden.

Einen Ausweg bietet der Ansatz, als Trainingsdaten die reichlich vorhandenen ϕ–Mesonen aus Elektroproduktion zu verwenden. Dies kann man dann tun, wenn man das gestreute Elektron im SPACAL nicht als Triggerbedingung benutzt, d.h. wenn man weder auf L1 noch auf L2 Informationen des SPACAL einsetzt. So sollten diese Ereignisse auf Triggerniveau nicht von denen aus Photoproduktion zu unterscheiden sein[15]. Jedoch wird der Öffnungswinkel der beiden Kaonen eine andere Verteilung aufweisen; dies gilt es gerade bei der Verwendung von topologischen Eingangsgrößen für ein Netz zu beachten.

Als weitere Quelle für Trainingsdaten bietet sich die Monte-Carlo-Simulation an. Wenn alle verwendeten Triggergrößen die Simulation genau beschreiben, wäre das eine sinnvolle Alternative. Jedoch kann die Qualität der Simulation aufgrund des fehlenden Signals in den Daten nicht überprüft werden.

Aus den genannten Gründen wurden für das in [2] entwickelte, und auch für alle anderen in dieser Arbeit vorgestellten Netze, ϕ–Mesonen aus Elektroproduktion als Trainingsdaten verwendet.

[14] Das in [2] entwickelte Netz basiert auf den Daten aus dem Jahr 1995.

[15] Die Richtigkeit dieser Arbeitshypothese ist noch nicht ausreichend bewiesen. Es finden hierzu weitere Untersuchungen statt.

Tabelle 4.9 zeigt die verwendeten Eingangsgrößen für das Netz aus [2]. Die Trainingsdaten für den Untergrund sollen in ihrer statistischen Zusammensetzung genau denjenigen Ereignissen entsprechen, die das Netz verarbeiten soll. Dazu muß der Subtrigger 68 aktiv sein und auf Triggerstufe L4 dürfen keine weiteren Schnitte angebracht werden (*L4 transparent*).

Tabelle 4.9: Eingabegrößen für das neuronale Netz zum Trigger von ϕ-Mesonen aus Photoproduktion [2].

Name	*Bedeutung*	*Wert*
cpvsum	Summe aller Einträge im zVtx Histogramm	0-15
cpvmax	Wert des Maximums im zVtx-Histogramm	0-255
cpvpos	Position des Maximums im zVtx-Histogramm	0-255
trhineg	# negative Spuren - hoher Impuls	0-45
trhipos	# positive Spuren - hoher Impuls	0-45
trloneg	# negative Spuren - niedriger Impuls	0-45
trlopos	# positive Spuren - niedriger Impuls	0-45

Das Ergebnis des Netztrainings mit Lernrate 0.001, Momentum 0.95, 5 versteckten Knoten und 100 Lernschritten ist in Bild 4.13 dargestellt.

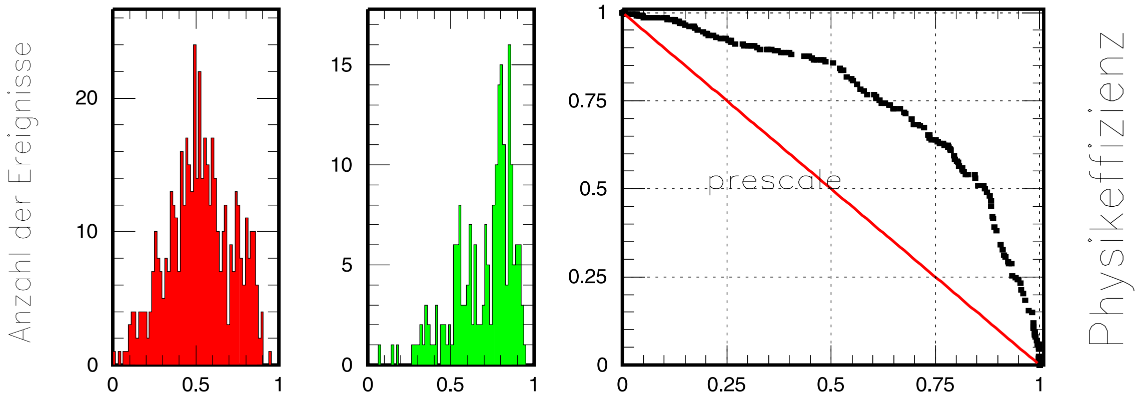

Bild 4.13: Ergebnis des Netzwerktrainings für elastische ϕ-Photoproduktion für den S68. Links die Ausgabeverteilung für Untergrund- und Physikdatensatz; rechts die Effizienzkurve.

Aus der Effizienzkurve läßt sich direkt die zu erwartende Ratenreduktion ablesen: bei einer Physikeffizienz von 55% ergibt sich eine Ratenreduktion um den Faktor 5; das entspricht 80% Untergrundunterdrückung.

Auf Basis dieser Erkenntnisse, wurden weitere Untersuchungen gemacht, um effizientere Netze zu entwickeln, und die Rate noch weiter zu reduzieren.

4.3 Untersuchungen zu Verbesserungsmöglichkeiten

Um Ergebnisse der Untersuchungen besser mit dem jetzt auf L2 laufenden Netz aus dem vorigen Abschnitt vergleichen zu können, wurde in [52] dieses Netz mit den Physik- und Untergrunddaten ausgewertet, die auch in allen anderen, in dieser Arbeit folgenden Untersuchungen verwendeten wurden.

Die Physik- und Untergrunddaten stammen aus der Datennahmeperiode 1997. Auf L1 war der Subtrigger 68 aktiv. Die Untergrunddaten stammen aus einem L2L4-transparent Run[16]. Zeigt die Verteilungen der Netzeingangsgrößen.

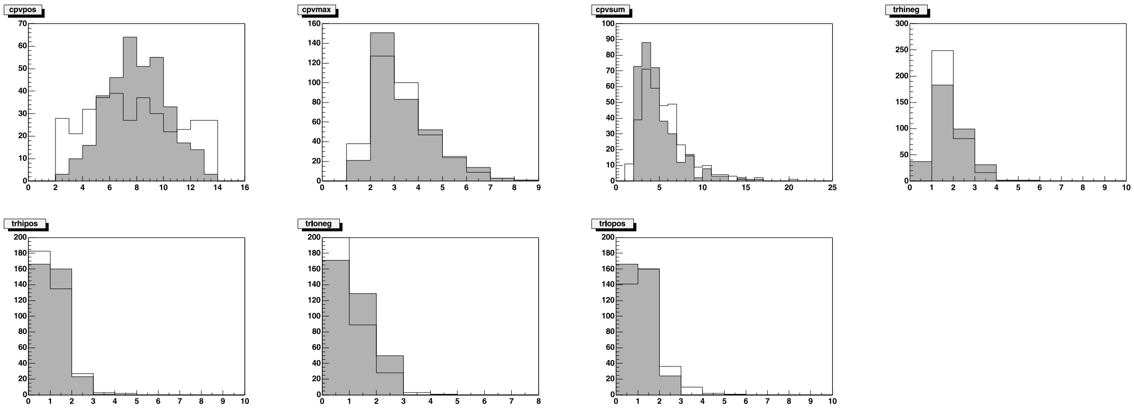

Bild 4.14: Die Netzeingangsgrößen für Physik und Untergrund aus den Daten von 1997. (Physik grau, Untergrund offen)

Die Ausgabeverteilung und die Effizienzkurve des Netzes für die angegebenen Physik- und Untergrunddaten sind in Bild 4.15 dargestellt.

Auf L1 wird ein dynamischer Prescale von 4-5 angewendet. Das Netz auf L2 wird mit 80% Untergrundreduktion betrieben. Wie aus Bild 4.15 zu ersehen ist, ergibt das eine Physikeffizienz für das Netz von 65%, und eine gesamte Physikeffizienz von L1 und L2 von durchschnittlich etwa 11%.

Dieses Ergebnis bietet Raum für Untersuchungen zur Effizienzsteigerung.

---

[16] L2L4-transparent bedeutet, daß weder auf L2 noch auf L4 Daten ausselektiert wurden. Es wurden nur die L1- Bedingungen angewendet. Diese Daten entsprechen also genau dem, was ein Netz auf L2 auch „sehen“ würde

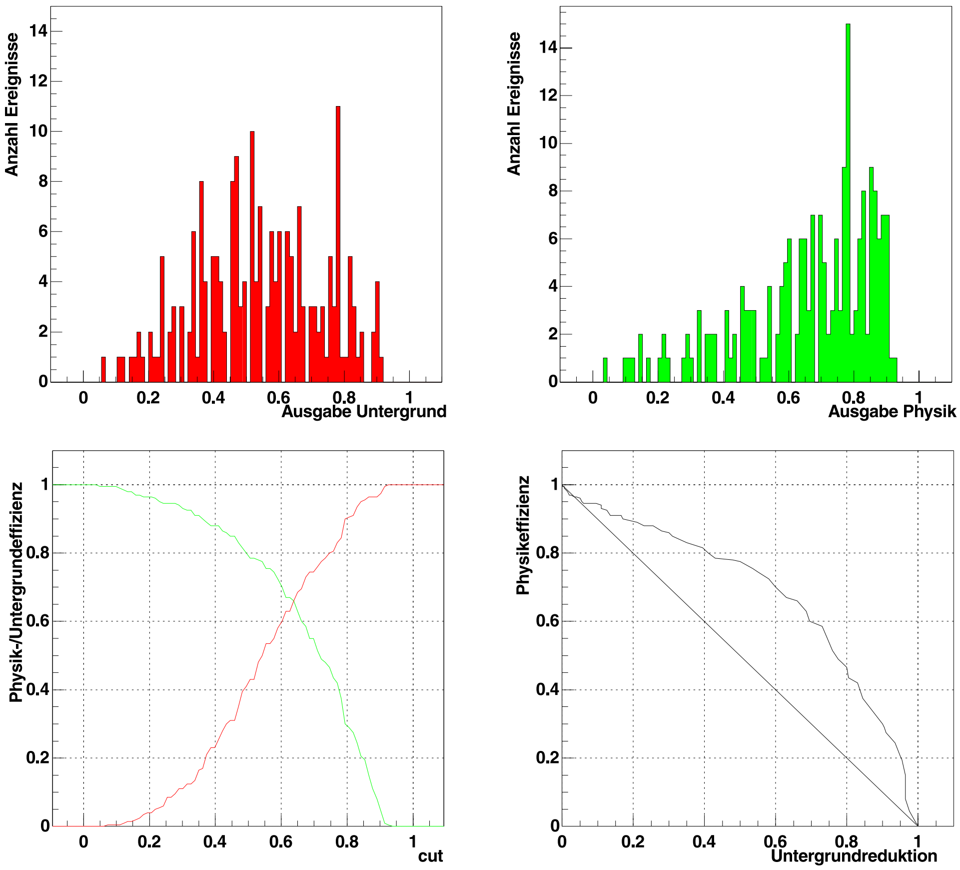

Bild 4.15: Die Ausgabeverteilung und die Effizienzkurve des auf L2 aktiven Netzes aus [2] mit den Daten von 1997.

Ein naheliegender Ansatz, um Netze effizienter zu machen ist, sie mit mehr Eingangsgrößen arbeiten zu lassen. Dies kann man tun, wenn ausreichend Statistik für ein Training mit vielen Eingangsgrößen vorhanden ist (siehe dazu Kapitel 3.5.3). Einen Anhaltspunkt, welche Größen als Eingangsvariablen für das Netz in Frage kommen, liefern die Verteilungen dieser Größen. Letztendlich kann aber nur durch intensives ausprobieren verschiedener Größen herausgefunden werden, welche Kombination an Eingangsvariablen die besten Ergebnisse liefert.

In [52] wurden verschiedene Kombinationen getestet. Als Ergebnis wurden die in Tabelle 4.10 aufgelisteten Größen für ein Netzwerktraining ausgewählt.

Tabelle 4.10: Bedeutung der neuen Eingangsgrößen

Name	*Bedeutung*	*Wert*
larife	LAr-Kalorimeter inner forward	0-255
larfbe	LAr-Kalorimeter forward barrel	0-255
larcbe	LAr-Kalorimeter central barrel	0-255
ironbb	Eisen backward barrel (Muon)	0-255

Für das Netztraining wurde mit verschiedenen Cuts versucht, aus den Daten ein möglichst reines ϕ-Signal zu bekommen, also einen möglichst hohen Anteil an ϕ-Mesonen in den Trainingsdaten. Tabelle 4.11 enthält die angewendeten Bedingungen und ihre Bedeutung.

Tabelle 4.11: Bedeutung der einzelnen Bedingungen des Cuts.

Bedingung	*Bedeutung*
(nphi==1)	Es wurde ein ϕ gefunden.
(ntrack2b==2)	Es wurden genau zwei Spuren im Zentralbereich des Detektors gefunden, wobei die eventuelle Spur des gestreuten Elektrons nicht mitgezählt wird.
(trig[68]==1)	Subtrigger 68 ist aktiv
(phipt>0.1)	Der Transversalimpuls des ϕ soll größer 100 MeV sein.
(encls[0]>12)	Im SPACAL (em) wurde ein Cluster mit mehr als 12 GeV gefunden.
((ecalo-eb[0]-eb[1]-eforw1)<0.5)	Die Gesamtenergie, ohne den Forwärtsbereich und ohne die extrapolierten Spuren, im LAr-Kalorimeter ist kleiner als 0,5.
(eforw<1.0)	Die Energie im Forwärtsbereich ist kleiner 1,0. Cut gegen protondissoziative Ereignisse.
(yjb*55>45)	Dieser Cut unterdrückt radiativen Untergrund[17].
(abs(phimass-1.020)<0.015))	Die erlaubte ϕ-Masse wird auf einen engen Bereich eingeschränkt.

Durch die Anwendung der Bedingung auf die Physik- und Untergrunddatensätze wurden Ereignisse selektiert, aus denen die Eingangsvariablen generiert wurden, deren Verteilung Bild 4.16 zeigt.

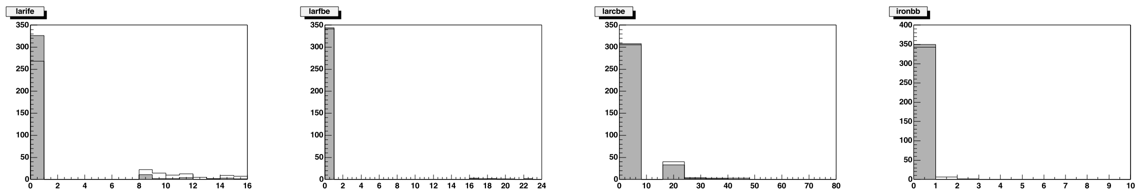

Bild 4.16: Verteilungen der neuen Netzeingangsgrößen. (Physik grau, Untergrund offen)

[17] Radiativer Untergrund entsteht, wenn das Elektron ein Photon abstrahlt, wobei diese Reaktion mit der eigentlich zu untersuchenden Reaktion nichts zu tun hat.

Diese Größen wurden zusätzlich zu den im vorigen Abschnitt vorgestellten Größen als Eingangsvariablen verwendet, und damit das Netz trainiert, dessen Ausgabeverteilung und Effizienzkurve in Bild 4.17 zu sehen ist.

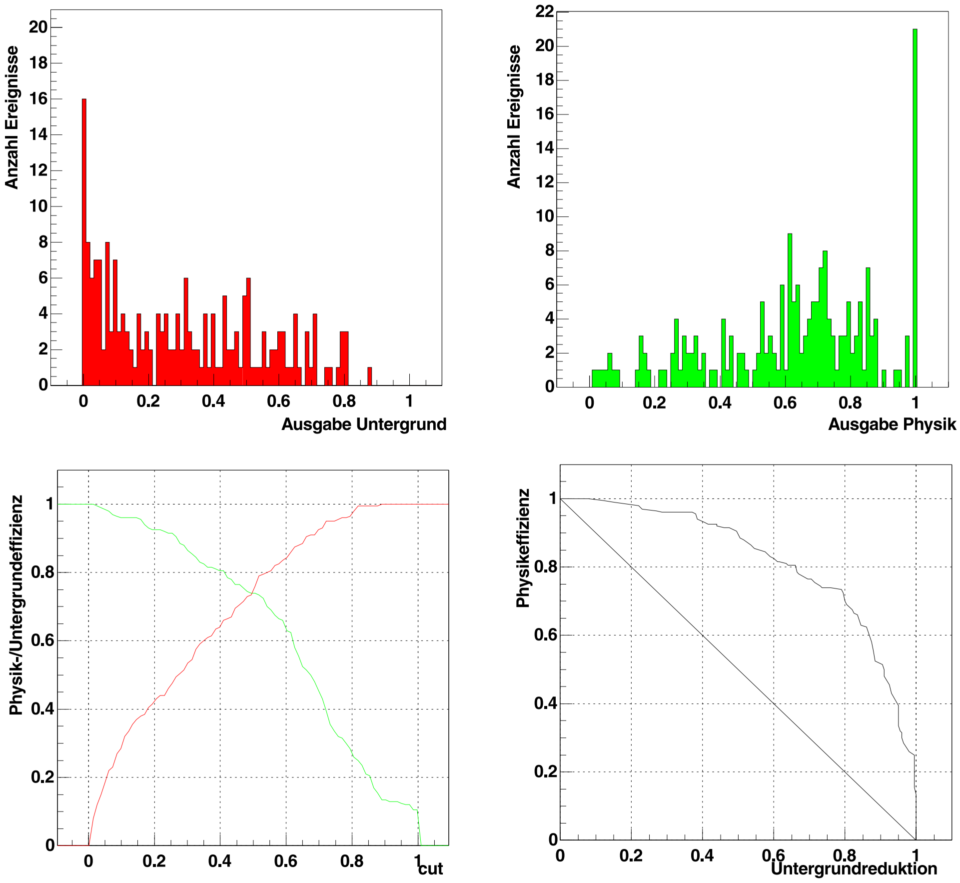

Bild 4.17: Die Effizienz des Netzes konnte durch die Verwendung der neuen Variablen deutlich gesteigert werden (aus [52]).

Würde man dieses Netz einsetzen, könnte die Untergrundreduktion auf 85% erhöht werden. Die Auswertung der Effizienzkurve in Bild 4.17 ergibt daraus eine Physikeffizienz von 65%; der Prescale auf L1 könnte auf 3 reduziert werden. Die gesamte Physikeffizienz nach L1 und L2 würde dann etwa 21% betragen, was einer Verbesserung im Vergleich zum jetzigen Netz um den Faktor 2 gleich kommt.

Mit diesem Netz dürfte die maximale Effizienz auf Basis der zur Verfügung stehenden Eingangsgrößen erreicht sein. Eine weitere Effizienzsteigerung wird durch Eingangsgrößen möglich sein, die zu diesem Zeitpunkt noch nicht auf L2 vorhanden sind. Solche Größen müssen also erst noch berechnet werden. Dies ist die Grundlage für die folgende Untersuchung.

4.4 Verbesserung der Netze durch Vorverarbeitung

Eine weiterer Ansatz, um Netze mit höherer Effizienz zu erhalten, ist es, Eingangsvariablen zu verwenden, die topologische Information beinhalten. Solche Größen liegen derzeit noch nicht vor. Für Untersuchungen müssen diese in Simulationen berechnet werden[18].

Naheliegende topologische Größen sind die Winkel φ und θ der Subdetektorzellen, in denen Einschläge registriert wurden. Da ein Teilchen oder Teilchenbündel meist mehrere, beieinander liegende Subdetektorzellen auslöst, ist es sinnvoll, diese zu einem einzigen Einschlag zusammenzufassen. Dieses Verfahren wird als Clustering bezeichnet und ist ein grundlegendes Konzept der in dieser Arbeit entwickelten DDB2. Es soll hier nicht näher auf die Details des Clusterings eingegangen werden; dies wird ausführlich in Kapitel 5.3.1 getan.

Bereits in [22] wurden die Winkel der Cluster für das LAr-Kalorimeter und die MWPC berechnet. Für den Trigger von ϕ–Mesonen bietet es sich an, die Daten in den Driftkammern ebenfalls zu clustern und die Winkel der einzelnen Cluster zu verwenden (siehe Kapitel 5.3.1.3). Es hat sich gezeigt, daß die Winkel aus dem LAr-Kalorimeter keine Verbesserung brachten. Dies ist nicht überraschend, da die Kaonen nur sehr wenig Energie im Kalorimeter deponieren und sich dadurch keine gut verwertbaren Winkelverteilungen ergeben. bigdoa0

Tabelle 4.12: Bedeutung der topologischen Eingangsgrößen

Name	*Bedeutung*	*Wert*
bigdpa0	φ Winkel zwischen zwei Cluster in den MWPC	0-255
bigdta0	θ Winkel zwischen zwei Cluster in den MWPC	0-255
cjcdiv	Abstand zweier Sektoren in den Driftkammern (CJC)	0-255

In den Proportionalkammern und in den Driftkammern, werden die Kaonen sehr gut detektiert, wodurch sich die Winkelverteilungen, die aus diesen Subdetektoren gewonnen werden, auch gut auswerten lassen. Bild 4.16 zeigt die Verteilungen der neu berechneten Eingangsgrößen.

[18] Das bedeutet, die Funktion der DDB wird simuliert. Wie dies genau geschieht, wird ausführlich in Kapitel 1 erörtert.

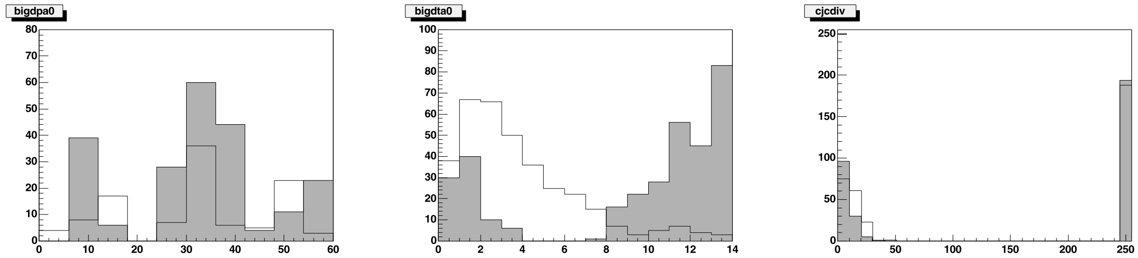

Bild 4.18: Verteilungen der neuen Netzeingangsgrößen. (Physik grau, Untergrund offen)

Für das Training wurden die gleichen Physik- und Untergrunddaten verwendet, wie im vorigen Abschnitt. Die verwendeten Cuts waren ebenfalls identisch. In Tabelle 4.13 sind die verwendeten Trainingsparameter aufgelistet.

Tabelle 4.13: Trainingsparameter für das neuronale Netz mit den neuen Eingangsgrößen.

Parameter	*Wert*
Lernrate	0,005
Momentum	0,0
Zahl der versteckten Knoten	7
Zahl der Lernschritte (Epochen)	1600

Mit diesem Netz erhält man bei einer Untergrundreduktion von 90%, eine Physikeffizienz von 85%. Man könnte den L1 Prescale auf 2 senken, und würde so eine gesamte Physikeffizienz nach L1 und L2 von etwa 42% erhalten. Im Vergleich zum derzeit aktiven Netz, ergibt sich eine Verbesserung um den Faktor 4.

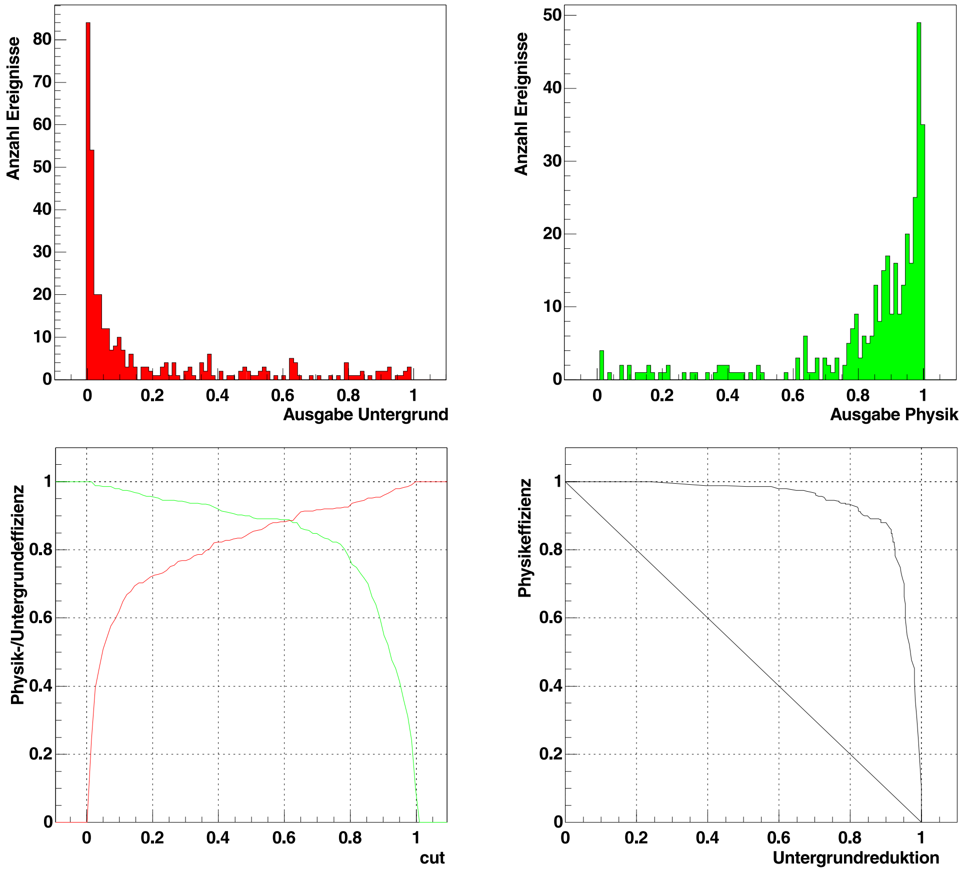

Bild 4.19: Die Verwendung neuer, topologischer Eingangsgrößen führt zu einer weiteren Verbesserung der Effizienz.

Tabelle 4.14 gibt einen Gesamtüberblick über alle drei, im vorigen und diesem Abschnitt, entwickelten Netze.

Tabelle 4.14: Vergleich der drei vorgestellten Netze

	Netz 1[1]	*Netz 2*[2]	*Netz 3*[3]
Untergrundreduktion	80%	85%	90%
Physikeffizienz	45%	65%	85%
L1 Prescale	4	3	2
Physikeffizienz nach L1 und L2	11%	21%	42%

[1] *Derzeit auf L2 aktives Netz*
[2] *In [52] entwickeltes Netz*
[3] *Netz mit zusätzlichen topologischen Größen (Winkel)*

Aufgrund dieser Ergebnisse, wird im folgenden ein neues Konzept zur Datenvorverarbeitung für das neuronale Netz bei L2 entwickelt und umgesetzt.

5 Entwicklung der DDB2

Das Ziel der DDB2 ist es, dem L2NN Trigger durch „intelligente“ Vorverarbeitung neue Eingangsgrößen zur Verfügung zu stellen. Diese Eingangsgrößen sollen sich passend für das jeweilige physikalische Problem aus einem Pool von bereits erzeugten Größen auswählen lassen; eine problemspezifische Weiterverarbeitung (*Postprocessing*) ist dabei immer noch möglich. So können neue Netze für spezielle Probleme schnell und effizient getestet und eingesetzt werden. Die DDB2 soll ab dem Jahr 2000 die vorhandene DDB ergänzen.

5.1 Arbeitsschritte

Bis zum endgültigen Einsatz der DDB2 bei H1 müssen mehrere Schritte abgearbeitet werden. Einige davon sind Teil dieser Arbeit:

Planung

Aufgrund der Erfahrungen mit der bereits vorhandenen DDB und den neuen Anforderungen in verschiedenen Physikanalysen entwickelt die L2 Gruppe von H1 am MPI ein Grundkonzept und grundlegende Algorithmen.

Softwaresimulation

Das Konzept und die Algorithmen werden mittels eine Anpassung der H1 Netzsimulationssoftware simuliert. Dazu müssen die geforderten Datenstrukturen und Abläufe implementiert werden. Erste Untersuchungen und Ausführungen hierzu finden sich in [22]. Die Softwaresimulation ist wesentlicher Bestandteil dieser Arbeit.

Test

Mit den Ausgaben der simulierten DDB2 sollen neuronale Netze trainiert werden. Durch Vergleich mit vorhergehenden Untersuchungen kann die Verbesserung der Effizienz dokumentiert werden. In dieser Arbeit werden ϕ-Mesonen als Testkanal verwendet.

Realisierung in Hardware

Erfüllt das Konzept und die Umsetzung die gestellten Anforderungen, können die bisher nur simulierten Algorithmen in Hardware umgesetzt werden. Dies ist nicht mehr Bestandteil dieser Arbeit.

5.2 Die Grundidee

Die Grundidee für die DDB2 kann folgendermaßen skizziert werden:

> Durch eine „intelligente" Vorverarbeitung sollen aus den Triggerdaten Netzeingangsgrößen erzeugt werden, welche – im optimalen Fall – wirklichen physikalischen Objekten[19] entsprechen. Dazu faßt man passende Daten aus den Subdetektoren zu Clustern zusammen (Clustering). Wahrscheinlich zusammengehörige Cluster verschiedener Subdetektoren verbindet man zu Objekten (Matching).

Vergangene Untersuchungen (siehe Kapitel 4.2) bestätigen die plausible Annahme, daß eine geeignete Auswahl der Eingangsgrößen eines neuronalen Netzwerkes dessen Effizienz deutlich verbessern kann. Die DDB (DDB2) versorgt bei L2 das Netz mit diesen Eingangsgrößen. Bisher wurden die Größen nur durch einfache Rechenoperationen gebildet. Grundsätzlich sind hier jedoch aufwendigere Operationen möglich.

Die bisher verwendeten Eingangsvariablen enthalten nur grobe topologische Information. Es liegt jedoch die Annahme nahe, daß ein neuronales Netz gerade diese Information besonders effizient nutzen könnte, besteht seine Stärke doch in der Mustererkennung. Da physikalische Ereignisse bestimmten topologischen Mustern folgen, sollte eine Auswahl bestimmter, für das jeweilige Problem ausgesuchter Variablen die gewünschte Effizienzsteigerung bringen.

Die erste Aufgabe muß nun darin bestehen, einen Pool von Variablen zur Verfügung zu stellen, der zwei Dinge ermöglicht:

1. Die im Pool gegebenen Variablen sind bereits zur Lösung vieler Probleme gut geeignet.
2. Diese Variablen können durch ein Postprocessing einfach zu weiteren Größen verarbeitet, die das gestellte Problem eventuell effizienter lösen.

Bei der Annäherung an die Frage, wie dieser Pool an Variablen aussehen sollte, geht man von zwei Voraussetzungen aus: die Art und Weise, wie die Triggerinformation der DDB zur Verfügung stehen und wie ein Mensch „von Hand" diese Information aufbereiten würde.

[19] Teilchen oder Teilchenbündel

Die Triggerinformation enthält Subdetektorgrößen pro Ereignis in Form von topologischen Mappen. Das können ein- oder zweidimensionale Mappen sein, sie können nur einzelne Bits oder auch ganzzahlige Größen enthalten. Programmiertechnisch sind es immer Arrays aus Bitfeldern oder Integerwerten. Hier sind die Einschläge durch gesetzte Bits, oder durch die im Subdetektor deponierte Energie, markiert.

Stellt man einen Menschen vor die Aufgabe, aus diesen Mappen die gewünschten Teilchen zu finden, würde er sicher die pro Ereignis vorhandene Information zunächst reduzieren. Er würde geschlossene Bereiche in den Mappen zu einem Wert zusammenfassen, und die anderen beteiligten Punkte löschen. Diese Vorgehensweise bezeichnet man als *Clustering*. Sie wurde bereits in [22] erfolgreich implementiert und getestet. Es werden also ein oder mehrere Werte einer Subdetektormappe zu einem Wert – dem Cluster – zusammengefaßt[20].

Diese so gewonnenen Informationen werden pro Mappe – also pro Subdetektor – generiert. Damit stehen für jedes Ereignis, für alle behandelten Subdetektoren und für alle gefundenen Cluster folgende Informationen zur Verfügung:

- Ort des Clusters in den Winkeln Theta und Phi
- jeweils aggregierte Einschlagsgröße[21]
- zusätzliche Informationen über den Aufbau des Clusters wie: Gesamtzahl beteiligter Zellen, Einschlagsgröße im Zentrum und im Ring

Als nächstes würde ein Mensch versuchen, einen Zusammenhang zwischen diesen Clustern herzustellen. Er würde sicher Cluster aus verschiedenen Subdetektoren, die in der selben Region liegen, miteinander in Verbindung bringen. Dieses Vorgehen bezeichnet man als *Matching*.

Beim Matching wird also versucht, ein Teilchen oder ein Teilchenbündel quer durch alle Subdetektoren zu verfolgen, und die so zusammengehörigen Cluster in einer neuen Datenstruktur, einem Objekt, zusammenzufassen.

Die aus den Clustern gebildeten Objekte können nun als Eingangsgrößen für ein neuronales Netz benutzt werden. Die spezielle Zusammensetzung eines Objekts, bzw. die Untermenge der davon ausgewählten Größen ist noch nicht festgelegt. Dies genügt der Forderung nach einem flexiblen Einsatz der DDB2 für verschiedenste physikalische Kanäle.

---

[20] Anstatt *Cluster* wird in diesem Zusammenhang auch manchmal der Begriff *Jet* verwendet; anstatt *Clustering* der Begriff *Jetfinding*. Die Verwendung des Begriffs Jet kann in diesem Zusammenhang leicht zu Unklarheiten und Verwechslungen führen. Daher werden im folgenden die Begriffe Cluster und Clustering benutzt.

[21] Die Einschlagsgröße kann die Energie sein, wie beim Kalorimeter oder einfach nur gesetzte Bits wie bei den Driftkammern. Meist wird, der Einfachheit halber, nur der Begriff Energie benutzt, auch wenn es für den jeweiligen Detektor eigentlich keine Energie gibt.

Die letztendliche Zusammensetzung des Objektes (die Eingangsvariablen für das Netz) kann für die jeweilige Problemstellung angepaßt werden. Hierzu können die Objekte nach jeder Größe sortiert werden. Dieses *Ordering* bietet die Möglichkeit, Objekte mit bestimmten Kriterien auszuwählen. So ist es möglich, beispielsweise nur die Cluster mit der größten Energie im LAr als Variablen für das Netz zu verwenden. Darüber hinaus ist es sehr wahrscheinlich, daß ein Netztraining mit sortierten Größen zu einem besseren Ergebnis, also besseren Netzen, führt[22].

5.3 Die verschiedenen Verarbeitungsstufen der DDB2

Im vorigen Abschnitt wurde der prinzipielle Ablauf der Datenverarbeitung in der DDB2 skizziert. Nun soll detailliert beschrieben werden, wie diese Schritte insgesamt und für jeden Detektor abgearbeitet werden[48].

Tabelle 5.15: Die Verarbeitungsschritte der DDB2

	Bezeichnung	*Sinn*
1	Clustering	Finden und zusammenfassen der Cluster in den einzelnen Subdetektoren
2	Matching	Verbinden von zusammengehörigen Clustern zu Objekten
3	Ordering	Sortieren der Objekte nach verschiedenen Kriterien
4	Net Inputs	Auswahl der passenden Größen aus den Objekten
5	Postprocessing	Eventuell Berechnung neuer Größen aus den Objekten

Auch wenn es auf den ersten Blick so scheint, als wäre dieser Ablauf fest vorgegeben, so ist es doch dem Anwender überlassen, einzelne Schritte auszulassen oder zu adaptieren. So können die nach dem Clustering vorhandenen Informationen bereits als Eingabegrößen benutzt werden ohne Matching und Ordering anzuwenden.

5.3.1 Clustering

Clustering ist ein Schritt, der in dem Konzept der DDB2 immer durchgeführt wird. Wie schon erklärt, werden hier durch Zusammenfassen (clustern) von Bereichen in einer Mappe neue Größen, die Cluster, erzeugt. Zu jedem Cluster werden folgende Größen errechnet und gespeichert:

- Nummer des Triggerelements im Zentrum des Clusters
- Theta-Index des Triggerelements im Zentrum des Clusters

[22] Untersuchungen von Steffen Udluft haben gezeigt, daß durch die sortierte Darreichung von Netzeingangsgrößen, das Training – bei gleichem Ergebnis – schneller beendet werden konnte. Es ist jedoch kein allgemein gültiger Beweis für diese Hypothese bekannt.

- Phi-Index des Triggerelements im Zentrum des Clusters
- Totale Einschlagsgröße
- Einschlagsgröße im Zentrum des Clusters
- Summierte Einschlagsgrößen im Ring des Clusters
- Anzahl Einschläge, die den Cluster bilden

Wie die Größen bei den einzelnen Subdetektoren genau berechnet werden, wird in den folgenden Abschnitten gezeigt. Wie diese Daten weiterverarbeitet werden, wird in Kapitel 5.3.2 dargestellt.

Der Algorithmus für das Clustering muß auf zwei verschiedene Datenformate der Subdetektormappen anwendbar sein: auf Bitmappen und auf Integermappen. In Bitmappen sind die einzelnen Einträge Nullen oder Einsen; bei Integermappen können die Einträge Werte zwischen 0 und 255 annehmen.

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

a)

0	0	0	0	0
0	4	0	2	0
0	5	8	0	0
0	0	3	0	0
0	0	0	0	0

b)

Bild 5.20: Beispiel für eine Bitmappe a) und für eine Integermappe b)

Wie aus Bild 5.20 zu ersehen ist, kann man in Integermappen den größten Wert innerhalb eines Clusters durch einfachen Vergleich finden; in Bitmappen ist dies nicht möglich. Um in beiden Fällen den selben Algorithmus verwenden zu können, summiert man in Bitmappen alle direkten Nachbarn (nearest neighbor) auf, und speichert das Ergebnis in einer temporären Mappe ab (*Preclustering*). Auf diese kann dann das eigentliche *Clusterfinding* angewendet werden.

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

Preclustering $\longrightarrow$

0	0	0	0	0
0	4	4	0	0
0	5	7	0	0
0	0	4	3	0
0	0	0	0	0

Bild 5.21: Der Ablauf beim Preclustering: Für jeden von Null verschiedenen Eintrag in der Bitmappe links werden seine direkten Nachbarn summiert und in eine temporäre Mappe (rechts) eingetragen. Auf diese kann dann das Clusterfinding angewendet werden.

Beim Clusterfinding wird jeder Eintrag mit seinen direkten Nachbarn vergleichen. Ist er größer als jeder Nachbar, wird er als Clusterzentrum erkannt, und die oben aufgeführten Daten zu diesem neu gefundenen Cluster gespeichert. Dabei werden einmal verwendete Zellen auf Null gesetzt, um doppeltes Zählen zu vermeiden. Eine detaillierter Beschreibung findet sich in [22].

Damit die gewonnenen Größen bereits nach dem Clustering einfach als Netzeingangsvariablen verwendet werden können, werden die Jets in jedem Detektor schon zu diesem Zeitpunkt nach der Energie sortiert. Dies geschieht hier aus programmiertechnischen Gründen, zur einfacheren Verarbeitung der Datenstrukturen, und hat nichts mit dem Verarbeitungsschritt *Ordering* zu tun. So ist ein weiterer Baustein für einen flexiblen Einsatz der DDB2 geschaffen und die Möglichkeit eröffnet, Netze zu vergleichen, die Variablen nach dem Clustering oder nach dem Matching verwenden.

5.3.1.1 LAr-Kalorimeter

Das LAr-Kalorimeter ist der größte und wichtigste Detektor bei H1. Er besteht aus einem elektromagnetischen und einem hadronischen Teil. Diese sind in jeweils 252 indizierte Bigtower[23] eingeteilt. Für jeden Teil gibt es eine eigene Mappe mit diesen 252 Einträgen. In den Mappen ist für jede Zelle jeweils ein Energieindex als Integerwert angegeben.

Das Clustering für beide Teile wurde bereits in [22] gemacht. Dort ist auch eine detaillierte Beschreibung des entwickelten Algorithmus mit Fallbeispielen zu finden. Zu dieser Arbeit wurden noch Veränderungen vorgenommen.

- Die Summation und der Vergleich werden nur innerhalb eines Subbusses durchgeführt. Dadurch läßt sich der Algorithmus leichter in Hardware umsetzen, ohne

[23] Bigtower: Mehrere Zellen im LAr-Kalorimeter, die, vom Vertex aus gesehen, in die selbe Richtung zeigen, werden zu Bigtowern zusammengefaßt.

das Qualitätseinbußen zu befürchten sind. So gibt es keine Connected Regions[24] mehr, und auf Hardware-Ebene kann parallel in den 8 FADC[25] Crates gearbeitet werden.

- Die neuen Größen, Zentrumsenergie, Ringenergie und Anzahl betroffener Zellen werden nun berechnet und gespeichert.

Bei der Implementierung wird zur Bestimmung des direkten Nachbarn eine vorher berechnete, und als Konstante gespeicherte Lookup Table benutzt. Eine solche Tabelle existiert auch zur Berechnung der Winkel Theta und Phi aus der jeweiligen Zellennummer.

Tabelle 5.16: Die Größen des LAr-Kalorimeters im Eventvektor nach dem Clustering

Größe	*Beschreibung*
BTENEJ	Totale Energie des Clusters (em)
BTECEJ	Energie im Zentrum des Clusters (em)
BTEREJ	Energie im Ring des Clusters (em)
BTNHEJ	Anzahl am Cluster beteiligter Hits (em)
BTTHEJ	Theta Winkel des Clusters (em)
BTPHEJ	Phi Winkel des Clusters (em)
SEEDEJ	Nummer des Zentrums-Clusters (em)
BTENHJ	Totale Energie des Clusters (hd)
BTECHJ	Energie im Zentrum des Clusters (hd)
BTERHJ	Energie im Ring des Clusters (hd)
BTNHHJ	Anzahl am Cluster beteiligter Hits (hd)
BTTHHJ	Theta Winkel des Clusters (hd)
BTPHHJ	Phi Winkel des Clusters (hd)
SEEDHJ	Nummer des Zentrums-Clusters (hd)

5.3.1.2 Proprotionalkammern (MWPC)

Auch das Clustering für die Proportionalkammern wurde schon in [22] implementiert. Es wird eine 16x16 Bit-Mappe geclustert. Die Information ob einzelne Zellen getroffen wurden, sind als Bits in Wörtern kodiert und müssen erst in eine besser verarbeitbare Form gebracht werden.

---

[24] Connected Region: Cluster, die sich aus Zellen von zwei oder mehr verschiedenen Subbussen zusammensetzen

[25] FADC: **F**lash **A**nalog **D**igital **C**onverter

Ist dies geschehen, steht wieder eine Mappe zur Verfügung, auf die wieder Summation (Preclustering) und Clusterfinding angewendet werden.

Tabelle 5.17: Die Größen der Proportionalkammern im Eventvektor nach dem Clustering

Größe	*Bezeichnung*
BIGENA	Anzahl am Cluster beteiligter Zellen
BIGTHA	Theta Winkel des Clusters
BIGPHA	Phi Winkel des Clusters
BIGSEA	Nummer der Zelle im Zentrum des Clusters

5.3.1.3 Driftkammern

Die Daten zu den Driftkammern werden in einer Mappe von der Dimension 45 geliefert. Dies entspricht den 45 Sektoren der Driftkammern im Winkel Phi. Es ist keine Information über den Winkel Theta vorhanden. Die Einträge in der Mappe enthalten kodiert Daten über Energie und Ladung einer Spur in einem bestimmten Sektor.

Die kodierten Daten müssen für eine einfachere Verarbeitung zunächst in eine übersichtlichere Form gebracht werden. So entstehen vier eindimensionale Mappen mit jeweils 45 Einträgen. Es gibt jeweils eine Mappe für:

- hohe Energie, positive Ladung
- hohe Energie, negative Ladung
- niedrige Energie, positive Ladung
- niedrige Energie, negative Ladung

Das Clustering ist für alle vier Mappen identisch und funktioniert nach dem bereits bekannten Muster für Bitmappen: zuerst wird für jeden Eintrag die Summe aller Nachbarn gebildet; im zweiten Schritt wird für jeden Eintrag verglichen, ob er der Größte in seiner Umgebung ist. Da die Mappen nur eindimensional sind, wird keine Lookup Table benötigt.

Vor dem Clustering

```
0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
```

Nach dem Clustering

```
0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0 0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
```

Bild 5.22: Beispiele für bereits dekodierte CJC Mappen vor und nach dem Clustering.

Für die weitere Verwendung – eventuell schon als Eingangsvariable für das Netz – werden zu jeder Mappe zwei Größen gespeichert: der Sektor in dem der Jet gefunden wurde, und die Anzahl betroffener Sektoren.

Tabelle 5.18: Die neuen Größen der CJC Driftkammern im Eventvektor nach dem Clustering.

Größe	*Beschreibung*
CJCHPS	Cluster Nummer (hipos)
CJCHP	Clustereinträge (hipos)
CJCLPS	Cluster Nummer (lopos)
CJCLP	Clustereinträge (lopos)
CJCHNS	Cluster Nummer (hineg)
CJCHN	Clustereinträge (hineg)
CJCLNS	Cluster Nummer (loneg)
CJCLN	Clustereinträge (loneg)

5.3.1.4 Muonkammern

Die Daten der Muonkammern werden nicht geclustert; die Körnung ist hier so groß, daß ein Clustering nicht mehr sinnvoll ist. Die Daten der Muonkammern sind in Bitfeldern kodiert und müssen für die weitere Verarbeitung in eine andere Form gebracht werden.

Die Bits sind in zwei Bitfeldern zu je 32 Bit kodiert. Dabei gilt folgende Einteilung:

Tabelle 5.19: Die Bedeutung der einzelnen Bits in den Bitfeldern des digitalen Muonsystems

Bitfeld	*Bit*	*Größe*
1	0-15	Backward-Endcap
	16-31	Backward-Barrel
2	32-47	Forward-Barrel
	48-63	Forward-Endcap

Die topologische Zuordnung der Bits zu den einzelnen Kammern ist in Bild 5.23 (nach [39]):

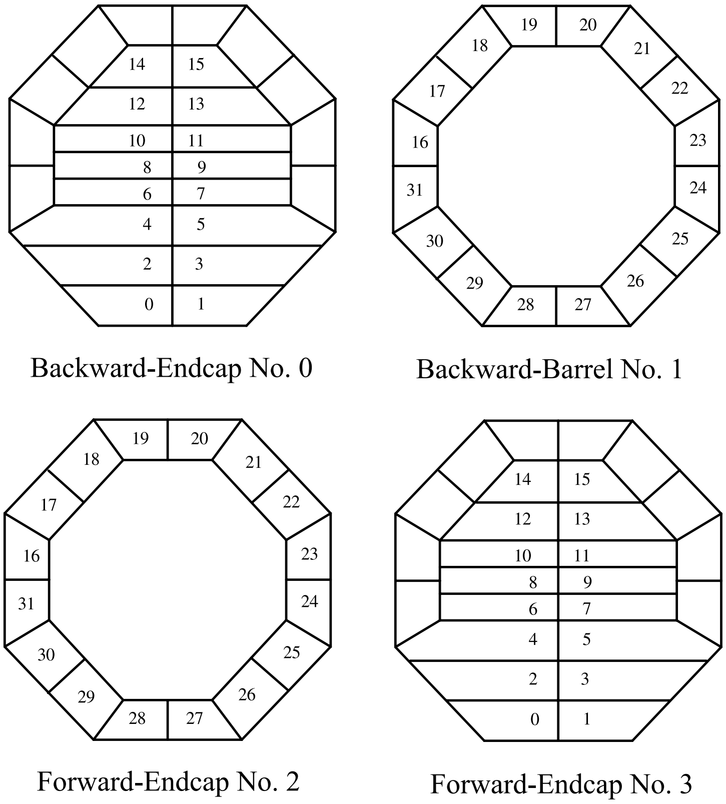

Bild 5.23: Topologische Zuordnung der Bits des digitalen Muonsystems. Die Blickrichtung ist die Richtung der positiven z-Achse (Protonrichtung).

Die Daten in der Muon-Mappe liegen nach der Verarbeitung in gepackter Form vor. Jeder Eintrag entspricht hier dem Index des Bits im Muonsystem.

Tabelle 5.20: Die Mappe des Muonsystems im Eventvektor

Größe	*Beschreibung*
MUMAP	Index der gesetzten Bits in der Muon Mappe

5.3.1.5 SPACAL

Das SPACAL sollte ursprünglich aus einem elektromagnetischen und einen hadronischen Teil bestehen. Derzeit wird jedoch nur der elektromagnetische Teil eingesetzt; die Zukunft des hadronischen Teil wird noch diskutiert und ist daher noch offen. Aufgrund dieser Ungewißheit wird in dieser Arbeit auch nur der elektromagnetische Teil behandelt.

Die benötigten SPACAL-Größen, lagen noch nicht im Eventvektor vor; konnten also nicht sofort für eine Weiterverarbeitung verwendet werden. Sie mußten erst aus den SIET-Bänken[26] ausgelesen werden. Grundsätzlich liefert das SPACAL eine zweidimensionale 20x20 Bitmappe. Diese ist allerdings in 25 Wörtern zu je 16 Bit kodiert. Somit

---

[26] SIET: **S**PACAL **I**nclusive **E**lectron **T**rigger

ergeben sich 400 Einträge. Als ersten Zwischenschritt werden die Wörter dekodiert und aus dem Ergebnis eine einfache 20x20 Integermappe aufgebaut. Diese wird im Eventvektor in der Variable SPAMAP gespeichert[27].

Die Dekodierung der Wörter, d.h. der Extraktion der Bitinformation, sieht jedes Wort als kleine 4x4 Mappe eines Bereichs der gesamten SPACAL Mappe. Bild 5.24 zeigt den Zusammenhang anhand eines Beispiels. Die Information, welches Wort aus der SIET-Bank, welchen Bereich der SPACAL-Mappe darstellt, ist in einer Lookup Table gespeichert[28]. Die Berechnung der Spalte und Zeile in der SPACAL Mappe erfolgt nun durch einfache Divisionen und Modulooperationen.

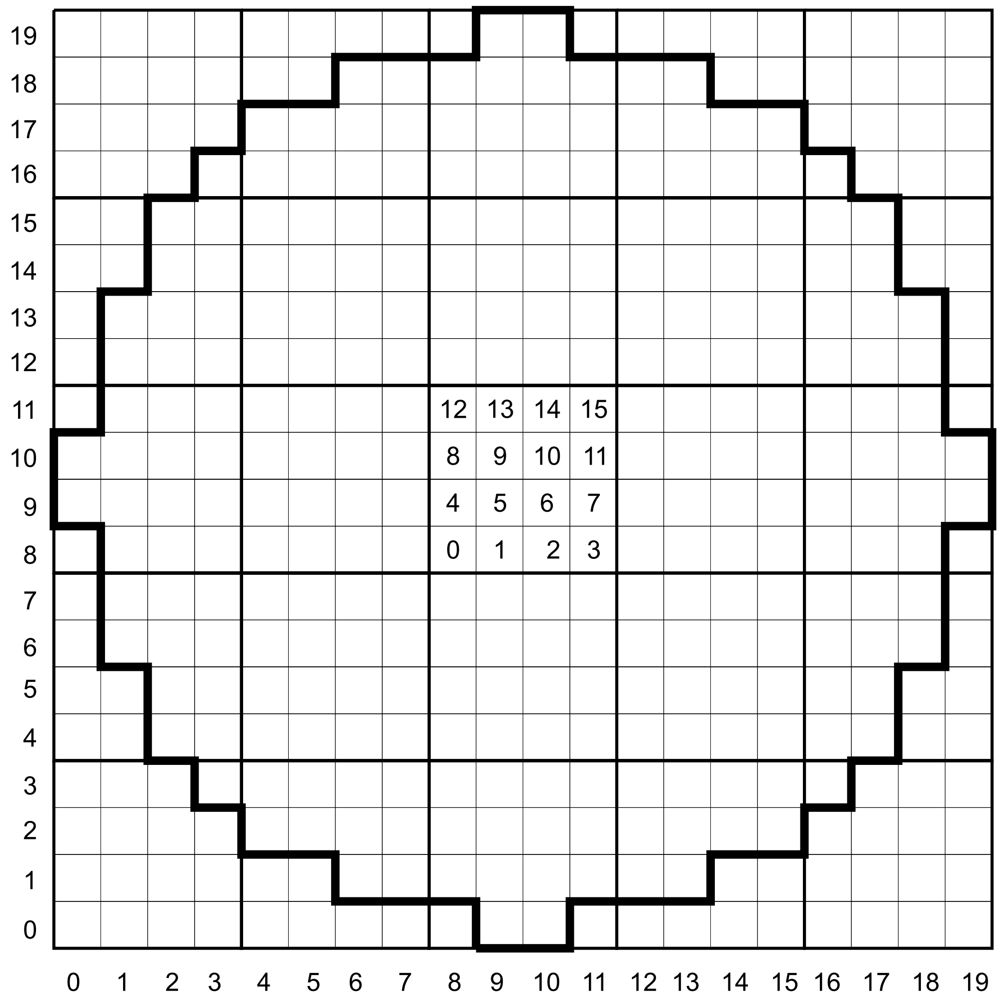

Bild 5.24: Die erzeugte SPACAL Mappe mit ihren 400 Einträgen. Diese sind in 25 Wörtern zu je 16 Bit kodiert. Die Lage der Bits für ein Wort ist eingezeichnet.

[27] Dieser Vorgang muß für jedes Ausgangsformat (POT, DST) getrennt gemacht werden. Es wurden daher die entsprechenden Stellen in den Funktionen *nsi_unpack_SPACAL* (nsi_unpack.c) und *nsi_unpack_TL23_SPACAL* (nsi_unpack_TL23.c) gemacht.

[28] Das ist die Konstante *pqzp_map* in der Datei *nsi_unpack.h*.

Die SPACAL Mappe wird dann mit dem bewährten Verfahren geclustert. Zuerst werden alle direkten Nachbarn eines Eintrags summiert und das Ergebnis in eine temporäre Mappe geschrieben. Daran anschließend wird die Größe jeder dieser summierten Einträge mit seinen direkten Nachbarn verglichen. Ist er der Größte in seiner Umgebung wird er als Jet mit seiner Zellennummer, Anzahl Hits, totale Energie, Energie im Zentrum und Energie im Ring gespeichert.

Tabelle 5.21: Neue Größen im Eventvektor für das SPACAL.

Größe	*Beschreibung*
SPAMAP	Gesamte SpaCal Mappe (em), 400 Einträge
SPENEJ	Totale Energie des Clusters (em)
SPECEJ	Energie im Zentrum des Clusters (em)
SPEREJ	Energie im Ring des Clusters (em)
SPNHEJ	Anzahl am Cluster beteiligter Hits (em)
SPSEEJ	Nummer des Zentrums-Clusters (em)

So sind im Eventvektor neue Größen für die Jets im SPACAL entstanden, die bereits als Eingangsgrößen für ein Netz benutzt werden können. Die Jets sind nach ihrer totalen Energie sortiert.

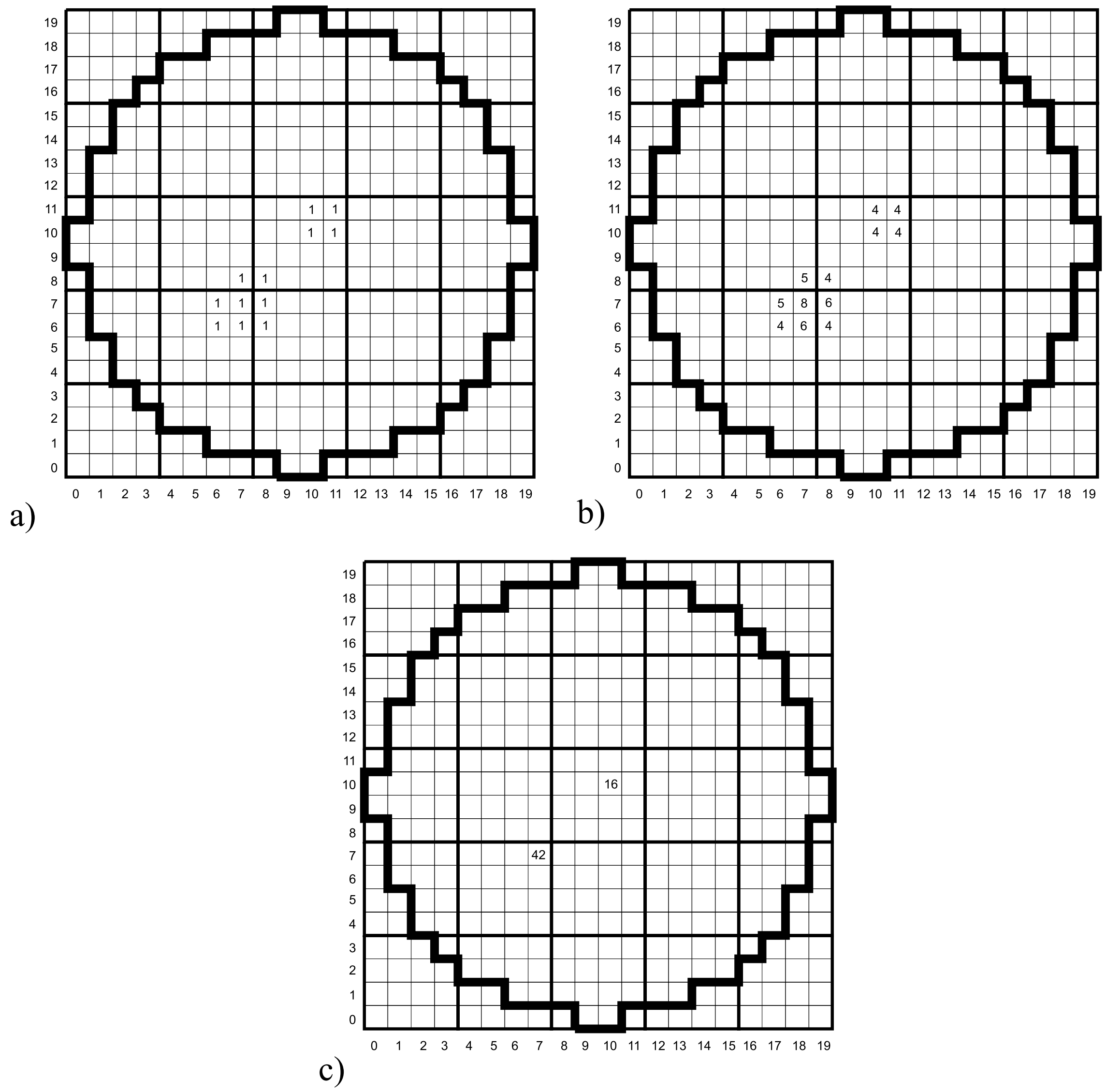

Bild 5.25: SPACAL Mappe eines J/Ψ CC[29] Ereignisses (RUNNUM: 232137; EVTNUM: 156313). Gut zu erkennen sind die beiden Spuren. In (a) die Rohmappe; in (b) die summierten Werte; in (c) der gefundenen Cluster .

5.3.2 Matching

Das Ziel des Matching ist es, die durch das Clustering gewonnen Informationen über die Cluster in den einzelnen Detektoren zu einer neuen Struktur zu verbinden. Diese Struktur wird *Objekt* genannt. Ein Objekt wird aus zusammengehörigen Clustern quer durch alle Subdetektoren gebildet. Die Zusammengehörigkeit ergibt sich aus der topologischen Nähe.

Alle Detektoren – außer dem SPACAL – sind zylinderförmig und liegen in Schichten übereinander. Auch wenn dies die Realität stark vereinfacht, so dient dieses Bild als

[29] CC: Cluster-Cluster; beide Elektronen im SPACAL sichtbar.

Grundlage für das Matching. Wie bereits beim Clustering beschrieben, liegen die Informationen der Detektoren als Mappen vor. Diese Mappen lassen sich übereinander legen, wodurch sich Nachbarschaftsbeziehungen zwischen den Zellen verschiedener Detektoren aufstellen lassen. Dabei muß auf die unterschiedlichen Ausdehnungen der Detektoren im Winkel Theta (also der Höhe der Zylinder) geachtet werden.

Das Matching läuft in fünf Schritten (Passes) ab. Dies ist nötig, da alle Cluster aus allen Subdetektoren zu Objekten verarbeitet werden sollen.

Begonnen wird mit dem LAr-Kalorimeter (em). Über eine Matching-Nachbarschaftstabelle, die wieder als konstante Lookup-Table gespeichert ist, und die benachbarten Zellen der anderen Subdetektoren enthält, werden topologisch passende Cluster zu Objekten zusammengefaßt. Cluster, die einmal zu einem Objekt verarbeitet wurden, werden nicht noch einmal verwendet, um eine Doppelzählung zu vermeiden. Ist dieser Vorgang (Pass 1) abgeschlossen, geschieht das gleiche jeweils für jeden anderen Subdetektor. Bild 5.26 zeigt den Ablauf des 5 Pass Matching schematisch.

Ein Ausnahme bildet hier das SPACAL, dieses Kalorimeter vollständig im Rückwärtsbereich liegt, und sich so keine vernünftigen Nachbarschaften oder Matching-Kriterien definieren lassen. Der einheitlichen Behandlung wegen, werden die SPACAL Daten in die Objekte kopiert. Sie sind aber nicht als Bestandteile des durch Matching entstandenen Vektors zu verstehen. Vielmehr sind sie Teil einer erweiterten Betrachtungsweise, welche die Objekte als n-Tuple aller Netzeingangsvariablen versteht.

Die Daten der passenden Jets werden programmtechnisch im Matching-Vektor gespeichert, der die Objekte repräsentiert. Beim Matching werden die bereits vorhandenen Daten der Cluster nicht mehr verändert, sondern lediglich neu organisiert.

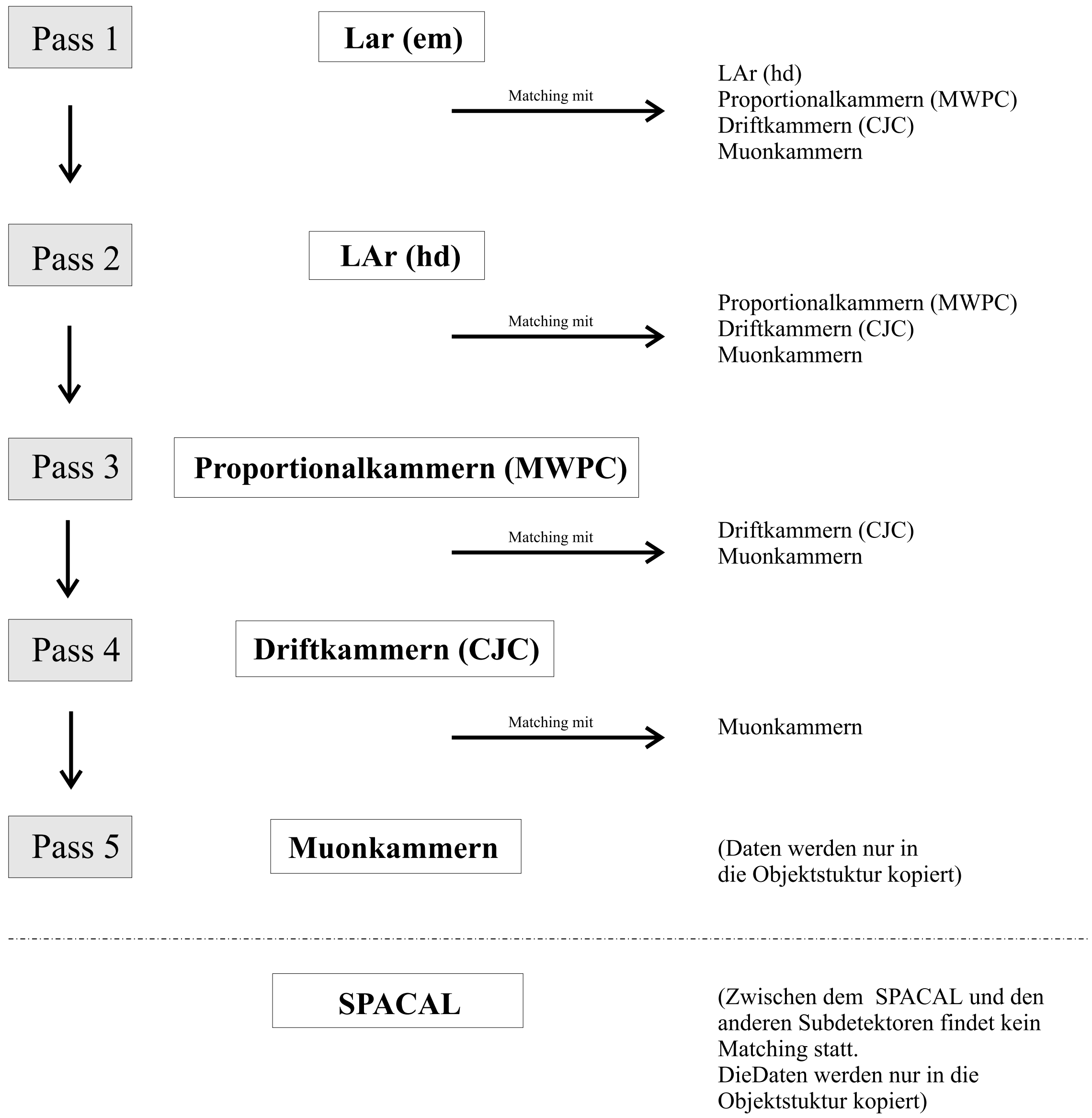

Bild 5.26: Das 5 Pass Matching: Ausgehend vom LAr-Kalorimeter wird versucht, Cluster in benachbarten Region der anderen Subtrigger zu finden. Dies wird ebenso für jeden anderen Subtrigger solange durchgeführt, bis alle Cluster auch Objekten zugeordnet sind. Eine Ausnahme bildet das SPACAL. Dessen Cluster werden nicht mit den Clustern der anderen Subdetektoren zu Objekten verbunden.

Tabelle 5.22: Die 28 einzelnen Größen des Matching-Vektors. Dieser Vektor wird programmtechnisch als Teil des Eventvektors gespeichert[30].

Größe	*Beschreibung*
VTHETA	Theta Winkel
VPHI	Phi Winkel
VLESE	LAr: Nummer des Zentrums-Clusters (em)
VLEET	LAr: Totale Energie des Clusters (em)
VLEEC	LAr: Energie im Zentrum des Clusters (em)
VLEER	LAr: Energie im Ring des Clusters (em)
VLENH	LAr: Anzahl am Cluster beteiligter Hits (em)
VLHSE	LAr: Nummer des Zentrums-Clusters (hd)
VLHET	LAr: Totale Energie des Clusters (hd)
VLHEC	LAr: Energie im Zentrum des Clusters (hd)
VLHER	LAr: Energie im Ring des Clusters (hd)
VLHNH	LAr: Anzahl am Cluster beteiligter Hits (hd)
VPSEED	MWPC: Nummer der Zelle im Zentrum des Clusters
VPET	MWPC: Anzahl am Cluster beteiligter Zellen
VCHPSE	CJC: Cluster Nummer (hipos)
VCHPET	CJC: Clustereinträge (hipos)
VCLPSE	CJC: Cluster Nummer (lopos)
VCLPET	CJC: Clustereinträge (lopos)
VCHNSE	CJC: Cluster Nummer (hineg)
VCHNET	CJC: Clustereinträge (hineg)
VCLNSE	CJC: Cluster Nummer (loneg)
VCLNET	CJC: Clustereinträge (loneg)
VMSEED	Muon: Index der gesetzten Bits in der Muon Mappe
VSESE	SpaCal: Nummer des Zentrums-Clusters (em)
VSEET	SpaCal: Totale Energie des Clusters (em)
VSEEC	SpaCal: Energie im Zentrum des Clusters (em)
VSEER	SpaCal: Energie im Ring des Clusters (em)
VSENH	SpaCal: Anzahl am Cluster beteiligter Hits (em)

[30] Die Größen in der Tabelle sind daher auch in der Datei *nsi_evv.desc* definiert. Sie befindet sich im Verzeichnis *nsi/misc*.

Die maximale Anzahl an Vektoren pro Ereignis ist momentan auf 16 festgelegt. Dies ist für die meisten Anwendungen mehr als ausreichend und dürfte daher selten überschritten werden. Selbstverständlich läßt sich diese Obergrenze einfach anpassen. Soll die Grenze nach oben verschoben werden, gilt zu beachten, daß nur maximal 64 Eingangsgrößen für das Netz möglich sind, und die Zahl an Vektoren multipliziert sich mit der Zahl der Größen pro Vektor und ergibt so die Anzahl an verwendeten Eingangsgrößen. Und wie bekannt, muß bei dieser Anzahl immer auf die Größe der vorhandenen Statistik geachtet werden.

5.3.3 Ordering

Das Ordering oder Sortieren der Vektoren nach bestimmten Größen ist wichtig für die spätere Auswahl der Eingabegrößen. So können bevorzugt Jets mit den größten oder kleinsten Winkeln ausgesucht werden, um z.B. gezielt den Vorwärts- oder Rückwärtsbereich zu selektieren. Ebenso lassen sich die Jets mit der größten Energie im LAr-Kalorimeter oder im SPACAL auswählen.

Bei der Implementierung wurde der Sortieralgorithmus „direktes Austauschen“ verwendet. Dabei werden nur einfache Operationen ohne Rekursion verwendet, wodurch er sich gut für eine Umsetzung in Hardware eignen dürfte.

Für die geringe Zahl an Elementen (momentan maximal 16 Vektoren) hat dieser Algorithmus sicher ausreichend Performance. Diese kann noch dadurch gesteigert werden, wenn die maximale Zahl an Elementen von derzeit 16 auf das jeweils benötigte Maß heruntergesetzt wird. In vielen Fällen wird es ausreichen, sich auf zwei bis drei Vektoren zu beschränken.

5.3.4 Net Inputs

Die DDB2 liefert die Eingangsgrößen für das neuronale Netz. Nachdem alle Operationen auf den Größen ausgeführt wurden und ein Pool an möglichen Eingangsvariablen vorliegt, gilt es nun sich für eine Untermenge dieser Variablen zu entscheiden, da die maximale Anzahl an Variablen 64 ist.

Die Entscheidung beeinflussen folgende Kriterien:

- Das physikalische Problem

  Die Relevanz der Variablen für das jeweilige physikalische Problem bestimmt deren Einsatz.

- Die für das Netztraining zur Verfügung stehende Statistik

  Je mehr Variablen verwendet werden sollen, desto höher muß die Trainingsstatistik sein.

Die endgültige Auswahl an Variablen wird nur durch viele Versuche mit unterschiedlichen Netzen optimiert. Dabei ist noch zu bedenken, daß mehrere Sätze von Eingangsgrößen parallel dem Netz zugeführt werden können. So lassen sich 4 verschiedene 16-Vektoren oder 8 verschiedene 8-Vektoren jeweils parallel verarbeiten.

Die Auswahl der Netzeingangsgrößen ist aber nicht auf die neu erzeugten Größen des Vektors beschränkt. Es können auch weiterhin die bisher zum Einsatz gekommen Variablen (z.B. die Vertexgrößen) verwendet werden.

5.3.5 Post Processing

Die DDB2 soll es ermöglichen, auf Basis des erzeugten Vektors an möglichen Eingabegröße, noch weitere Größen zu berechnen. Solche Größen könnten beispielsweise die Winkel zwischen den Vektoren sein, oder auch einfach nur aggregierte Größen, wie die Anzahl an Vektoren pro Ereignis.

Die Voraussetzungen für das Post Processing sind mit dieser Arbeit geschaffen. Die beschriebenen Datenstrukturen lassen sich relativ einfach anpassen, wobei der Programmcode als Grundlage für Erweiterungen dienen kann.

6 Ergebnisse

Im Idealfall sollen die Objekte, die durch die DDB2 gefundenen werden, physikalischen Teilchen entsprechen. Ein erster Ansatz um die Qualität der Objekte zu überprüfen, ist, die Differenz von verschiedenen Objektgrößen mir rekonstruierten Größen zu ermitteln. Es soll also herausgefunden werden, in wieweit die gefundenen Objekte mit den rekonstruierten Physikereignissen in Verbindung gebracht werden können.

Im Fall der ϕ-Mesonen, ergibt sich ein gutes Kriterium, wenn man die Winkel θ und φ der beiden Kaonen betrachtet. Man kann die Differenz der rekonstruierten Spuren zu den in den Objekten gespeicherten Winkeln ermitteln. Die Winkel der Objekte, sind die Winkel des Zentrums des jeweilige Detektors.

Für die Auswertung wurden die Daten aus 1997 verwendet, die bereits in Kapitel 4.3 ausführlich beschrieben wurden.

Durch die Art und Weise des Clustering und Matching, bleibt bei zwei nahe aneinander liegenden Spuren meist nur ein Objekt übrig. Daher konnte für eine sinnvolle Auswertung jeweils nur ein gefundenes Objekt pro Event mit den rekonstruierten Kaonen verglichen werden. Da die Spur der geladenen Kaonen in der φ-Ebene aufgrund des Magnetfelds im Detektor gekrümmt ist, darf für die Auswertung nicht der Winkel am Vertex verwendet werden, der bei der Rekonstruktion berechnet wird, sondern der Winkel, der sich aus der Verbindungslinie zwischen Vertex und Auftreffpunkt im Subdetektor ergibt.

Wenn die gefundenen Objekte, und somit das Konzept der DDB2, ausreichend gut sind, sollte die Differenz der Winkel nur klein sein. Im Idealfall, würde sich ein einzelner Peak bei Null ergeben. Dies kann aber nicht erreicht werden, da die Winkel der Objekte immer in Schritten erhöht werden, die der Körnung des jeweiligen Subdetektors entsprechen. Also sollte die Kurve, im Idealfall bei Null einen hohen Wert haben, und mit steigender Winkeldifferenz abfallen.

Bild 6.1 zeigt die Anzahl an Ereignissen – in diesem Fall gleich der Anzahl der Objekte – gegen die Winkeldifferenz in θ zu dem Kaon mit dem kleineren θ (links) und dem größeren θ (rechts)[31].

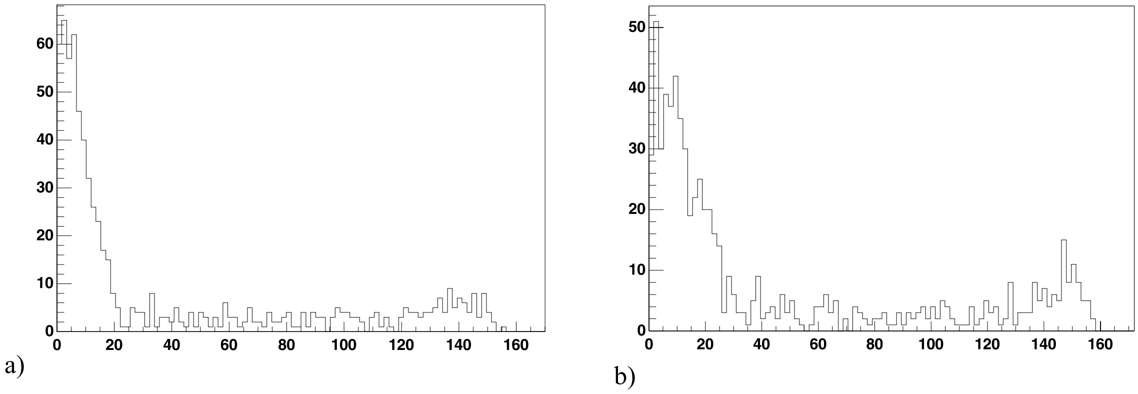

Bild 6.1: Winkeldifferenz in θ der Objekte zu den rekonstruierten Kaonen a) Kaon mit kleinerem θ b)Kaon mit größerem θ.

In Bild 6.1 ist zu erkennen, daß die Differenz der Objektwinkel zu den Winkeln der Kaonen für das Kaon mit größerem θ etwas größer ist. Eine Erklärung hierfür ist die Reihenfolge, in der der Clusteralgorithmus nach Clustern sucht. Dies geschieht in aufsteigenden θ Winkeln, so daß bevorzugt Cluster – und damit auch Objekte – mit kleinerem θ gefunden werden.

In Bild 6.2 ist die Winkeldifferenz für den Winkel φ zu sehen. Hier ergibt sich aufgrund des beschriebenen Effektes keine so große Abweichung für die beiden Kaonen.

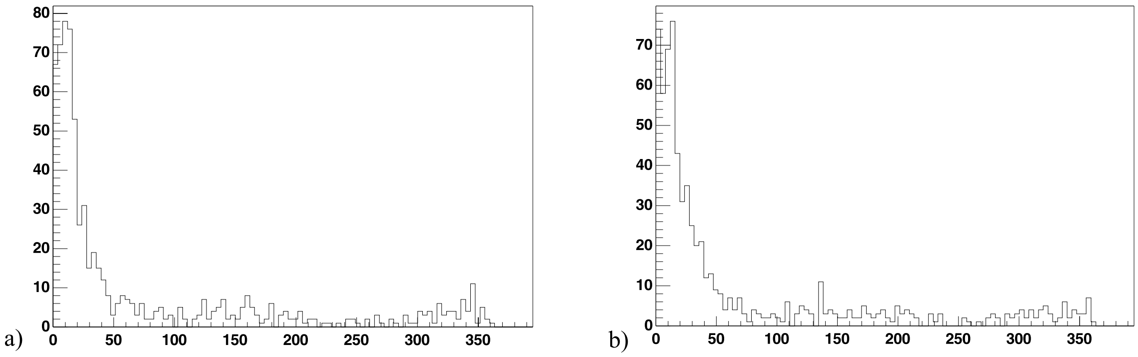

Bild 6.2: Winkeldifferenz in φ der Objekte zu den rekonstruierten Kaonen a) Kaon mit kleinerem θ b)Kaon mit größerem θ.

Die Bilder zeigen das erhoffte Ergebnis und bestätigen somit das Konzept der DDB2.

[31] Bei der Rekonstruktion werden die Daten zu den Kaonen in Arrays mit der Dimension 2 gespeichert. Der erste Eintrag enthält das Kaon mit dem kleineren Winkel θ.

Quellen

[1] Gaede, Frank: *Exklusive Produktion von ϕ-Mesonen in ep-Streuung am H1-Experiment bei HERA*. Dissertation, Christian-Albrechts-Universität Kiel, 1997.

[2] Gell-Mann, M.: *A Schematic Model of Baryons and Mesons*. *Phys. Lett.* {8} (1964) 214.

[3] Gell-Mann, M.: *The Eightfold Way: A Theory of Strong Interaction Symmetry*. *Caltech Report CTSL-20* (1961).

[4] T. Regge: *Introduction to complex orbital momenta. Nuovo Cim.*, 14, p. 951,1959.

[5] P.D.B. Collins: *An introduction to Regge theory and high energy physics.* Cambridge, 1977.

[6] J. J. Sakurai: *Theory of strong interactions.* Annal. d. Phys., **11**, pp. 1-48, 1960.

[7] J. J. Sakurai: *Vector meson dominance and high-energy electron proton inelasticscattering.* Phys. Rev. Lett., **22**, pp. 981-984, 1969.

[8] Leonid Frankfurt, Werner Koepf, und Mark Strikman: *Hard diffractive electroproduction of vector mesons in QCD*. Phys. Rev., **D54**, pp. 3194-3215, 1996.hep-ph/9509311.

[9] H1 Collaboration, I. Abt, et al: *Measurement of the proton structure functionF$_2$(x; Q^2) in the low -x region at HERA.* Nucl. Phys. B, **407**, pp. 515-535, 1993.

[10] H1 Collaboration, I. Abt, et al: *A measurement of the proton structure functionF$_2$(x; Q^2).* Nucl. Phys. B, **439**, pp. 471-502, 1995.

[11] H1, S. Aid et al: *A Measurement and QCD analysis of the proton structurefunction F$_2$(x; Q^2) at HERA.* Nucl. Phys., **B470**, pp. 3-40, 1996. hep-ex/9603004.

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

[13] Max-Plank-Institut für Physik, Werner-Heisenberg-Institut: *Jahresbericht 1997*, 1997.

[14] Kelly, D.R.C. et al.: The Electron Beam Lifetime Problem in HERA. *In Reports at the 1995 Particle Acceleretor Conference PAC95.* DESY, 1995.

[15] Möck, Jürgen: *Untersuchung diffraktiver J/Ψ -Ereignisse im H1-Experiment bei HERA und Entwicklung neuronaler Triggeralgorithmen*, Dissertation, Technische Universität München, 1997.

[16] Rüter, Karsten: *Untersuchungen des schwachen geladenen Stroms in tiefinelastischen Positron-Proton Kollisionen mit dem H1-Detektor am HERA-Speicherring*, Dissertation, Ludwig-Maximilians-Universität München, 1996.

[17] J. Dichtl, M. Eberl, L. Janauschek: *Training and Testing of a Neural Network for the L2 Neural Network Trigger*, January 27, 1999, Max-Planck Institut für Physik, München, Germany

[18] Rojas, Raúl: *Theorie der neuronalen Netze: eine systematische Einführung*, Heidelberg, Springer, 1993

[19] Brause, Rüdiger: *Neuronale Netze: eine Einführung in die Neuroinformatik*, Stuttgart, Teubner, 1991

[20] Braun, Heinrich: *Neuronale Netze: Optimierung durch Lernen und Evolution*, Heidelberg, Springer, 1997

[21] W. S. McCulloch, W. H. Pitts: A Logical Calculus of the Ideas Imminent in Neural Nets; Bulletin of Mathematical Biophysics (1943) Vol. 5

[22] Concetti, Enrico: *Untersuchungen zu neuronalen Triggern für J/Ψ-Erzeugung in Elektron-Proton-Stößen bei HERA*, Diplomarbeit, Ludwig-Maximilians-Universität München, 1998

[23] M. Minsky, S. Papert: *Perceptrons: An Introduction to Computional Geometry*, MIT Press, Cambridge, (1969)

[24] Udluft, Steffen: *Untersuchungen zu Neuronalen Netzen als Vertextrigger im H1-Experiment bei HERA*, Diplomarbeit, Ludwig-Maximilians-Universität München, 1996.

[25] A.K. Kolmogorov: *On the Representation of Continuous Functions of severeal Variables by Superposition of One Variable and Addition*, Dokl. Akad. Naud. 114, 1957

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

[27] R. Hecht-Nielsen: *Kolmogorov's Mapping Neural Network Existence Theorem*, IEEE First Annual Int. Conf. On Neural Networks, BandII, 1987

[28] K. Hornik, M. stinchcomb, H. White: *Multilayer Feedforward Networks are Universal Approximators*, Neural Networks 2, 1991

[29] G. Cybenko: *Approximation by superpositions of sigmoidal functions*, Math. Of Control, Signal and systems 2, 1989

[30] D.E. Rumelhardt, J.L. McClelland: *Extrapolations in the Microstructure of Cognition*, in Parallel Distributed Processing Bd. 1und 2, Cambridge Ma., 1986

[31] D.E. Rumelhardt, G.E. Hinton, R.J. Williams: *Learning representations by Backpropagating Errors*, Nature 323, S. 533, 1986

[32] L.F.A. Wessels, E. Barnard: *Avoiding False Local Minima by Proper Initialization of Connections*, IEEE Transactions on Neural Networks 3 (6), 1992

[33] H1 Kollaboration, Luminositätstabellen, `www-h1.desy/~f11lev/luminosity/-summary_tables`.

[34] B.H. Wiik. *HERA Status*. In W. Buchmüller and G. Ingelmann, Proceedings of the Workshop: Physics at HERA, volume 1, pagea 1-22. DESYS, October 1991

[35] H1 Calorimeter Group, B. Andrieu et al: *Results from Pion Calibration Runs for the H1 Liquid Argon Calorimeter and Comparisons with Simulations*. Nucl. Instrum. Meth., **A 336**, pages 499-509, 1993.

[36] H. Bethe, W. Heitler: *On the Stopping of fast Particles and on the Creation of Positive electrons*. Proc. Roy. Soc., **A 146**, page 83, 1934.

[37] H1 Collaboration, I. Abt et al.: *The H1 Detector at HERA*. DESY Report DESY-93-103, DESY, 1993.

[38] Udluft, Steffen: *The L2NN Training Chain*. Vortrag beim L2NN Workshop, DESY, Hamburg, 27.01.1999.

[39] Goldner, Dirk: *Online-Datenaufbereitung für die Hardwarerealisierung eines künstlichen neuronalen Netzwerkes als Level-2 Trigger im H1-Experiment*. Dissertation, Universität Dortmund, 1996.

[40] Zobel, Tobias: *Neuronale Netzwerke als Trigger für die J/Ψ-Produktion in ep-Kollisionen am Beispiel des Zerfalls $J/\Psi \to \mu^+\mu^-$*. Diplomarbeit, Ludwig-Maximilians-Universität München, 1996.

[41] J. Bürger et al.: *The Central Inner z Drift Chamber of the H1 Experiment*. Nucl. Instrum. Meth., **A 279**, page 217, 1989.

[42] S. Egli et al.: *The Central Inner z Drift Chamber of the H1 Experiment*. Nucl. Instrum. Meth., **A 283**, page 487, 1989.

[43] K. Müller et al.: *Construction and Prformance of a Thin Cylindrical Multiwire Proportional Chamber with Cathode Pad Readout for the H1 Experiment*. Nucl. Instrum. Meth., **A 312**, page 457, 1992.

[44] H1 SPACAL Group, T. Nicholls et al.: *Performance of an Electromagnetic Lead/Scintillating Fiber Calorimeter for the H1 Detector*. Nucl. Instrum. Meth.,

A 374, pages 149-156, 1996.

[45] H1 SPACAL Group, T.D. Appuhn et al.: *H1 Backward Upgrade with a SPACAL Calorimater: the Hadronic Section*. DESY Report DESY-96-013, DESY, 1996.

[46] Westner, Dominik: *Ein neuronaler Netzwerktrigger für die Produktion von J/Ψ-Teilchen in ep-Streuung (J/Ψ → e+e-).* Diplomarbeit, Ludwig-Maximilians-Universität München, 1996.

[47] C. Kiesling, E. Concetti: *Upgrade of the Neuronal Network Trigger: Version II of the Data Distribution Board*. Interner Bericht der MPI-Gruppe, November 1997.

[48] C. Kiesling: *Design Specifications for the DDB22 (Revision 3).* Internes Papier zum MPI Meeting. 03.12.1998.

[49] Review of Particel Physics, Particle Data Group, 1998.

[50] I.J.R. Aitchison, A.J.G. Hey: *Gauge Theories in Particle Physics*. Adam Hilger Ltd. 1982.

[51] Alan D. Martin, Francis Halzen: *Quarks and Leptons: An Introductory Course to Modern Particel Physics.* New York, 1984.

[52] Ellgaß, Daniela: *Neuronale Algorithmen zum Trigger photoproduzierter ϕ-Mesonen in ep-Kollisionen bei HERA*. Zulassungsarbeit. Ludwig-Maximilians-Universität München, 1999 in Vorbereitung.