# Untersuchungen zu Neuronalen Netzen als Vertextrigger im H1–Experiment bei HERA

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

vorgelegt von

*Steffen Udluft*

angefertigt am

Max–Planck–Institut für Physik
– Werner–Heisenberg–Institut –

Mai 1996

Allen, die mich bei dieser Arbeit unterstützt haben, möchte ich herzlich danken.

# Inhaltsverzeichnis

# 1. Einleitung

Unseren derzeitigen Stand des Wissens über die Struktur der Materie im subatomaren Bereich verdanken wir fast ausschließlich Experimenten mit hochenergetischen Teilchen.

Heinrich Hertz (1891) und Phillip Lenard (1903) beobachteten, daß Kathodenstrahlen von etwa 40 kV Beschleunigungsspannung Metallfolien von vielen tausend Atomlagen Dicke praktisch ungehindert durchdringen. Dies führte zu der Erkenntnis, daß Atome keine kompakten Gebilde sein können, sondern überwiegend leerer Raum sind. Wie leer dieser Raum ist, konnten der britische Atomphysiker Ernest Rutherford und seine Mitarbeiter Marsden und Geiger in den Jahren 1906–1916 zeigen. Durch Experimente, bei denen sie Alphateilchen mit Energien von etwa 6 MeV benutzten, um dünne Goldfolien zu durchstrahlen, entdeckten sie den Atomkern und 1920 konnte Rutherford das Proton als Bestandteil des Stickstoffatoms nachweisen [1]. Ebenfalls aus Experimenten mit Strahlen von Alphateilchen, die 1930 von den deutschen Physikern Walter Bothe und Herbert Becker und 1932 von dem französischen Physikerpaar Frederic und Irene Joliot–Curie[1] durchgeführt wurden, konnte der britische Physiker James Chadwick die Existenz des Neutrons ableiten.

Um die Struktur der Nukleonen (Proton, Neutron) zu untersuchen, werden Teilchenstrahlen höherer Energie benötigt, da eine Teilchensonde keine Strukturen auflösen kann, die kleiner sind als seine de Broglie–Wellenlänge $\lambda = h/p$. Dabei ist $h$ das Plancksche Wirkungsquant und $p$ der Teilchenimpuls. Eine weitere Beobachtung, die unabdingbar auf hohe Teilchenenergien angewiesen ist, ist die Erzeugung von Teilchen der Masse $m$, die nach der Einsteinschen Masse–Energie Beziehung erst bei Energien von mindestens $E = mc^2$ im Schwerpunktsystem der Stoßpartner möglich wird. So war lange Zeit die Beobachtung der sehr energiereichen kosmischen Strahlung[2] die einzige Möglichkeit neue Teilchen zu entdecken. Das Positron (C. D. Anderson, 1932), Myon (C. D. Anderson, 1935) und das Pi-Meson (C. F. Powell, 1947) wurden durch Experimente mit kosmischer Strahlung gefunden.

Neben hoher Energie der Strahlung, ist eine hohe Intensität die zweite Voraussetzung für systematische Erforschung der subatomaren Strukturen. Dies erklärt die herausragende Bedeutung der Teilchenbeschleuniger, die eine wesentlich höhere Intensität als die kosmische Strahlung bereitstellen können. Die ersten Anlagen von John D. Cockcroft und Ernest T. S. Walton, die in Rutherfords Labor arbeiteten, beschleunigten im Jahr 1928 Protonen auf nahezu 400 keV. Weitere Ecksteine der Entwicklung der Teilchenbeschleuniger waren der Van-de-Graaf–Generator (Robert J. Van de Graaf, 1929) mit 8 MeV, das Zyklotron (Ernest O. Lawrence, 1932) mit anfangs 1,25 MeV, später (1939) 20 MeV und das Synchrotron (Wladimir J. Weksler, Edwin M. McMillan,

[1] Irene war Tochter von Pierre und Marie Curie

[2] Dabei handelt es sich um extraterrestrische Teilchenstrahlen verschiedenster Art, deren Quelle bislang unbekannt ist. Das Energiespektrum reicht bis $10^9$ TeV, der Mittelwert liegt zwischen 1 und 10 GeV [2].

1945) mit 200–400 MeV. Im Jahre 1956 trug die Entwicklung immer leistungsfähigerer Teilchenbeschleuniger lang ersehnte Früchte – Robert Hofstadter konnte die Struktur leichter Atomkerne mit hochenergetischen Elektronen untersuchen und erste Erkenntnisse über die Struktur der Nukleonen gewinnen. Eine vorläufige Krönung nahm diese Entwicklung, als im Jahre 1966 am SLAC[3] ein Experiment durchgeführt wurde, das denen von Rutherford, Geiger und Marsden sehr ähnlich war. In der modernen Entsprechung wurden hochenergetische Elektronen von bis zu 16 GeV Energie auf ruhende Protonen geschossen. J. Bjorken und Richard P. Feynman konnten 1968 zeigen, daß die beobachtete Energie- und Winkelverteilung der gestreuten Elektronen durch elastische Streuung der Elektronen an drei punktförmigen Konstituenten des Protons erklärt werden kann. Diese Konstituenten, wurden mit den bereits 1964 von Murray Gell–Mann und George Zweig postulierten Quarks identifiziert [3].

Bei den bisher beschriebenen Experimenten mit ruhendem Target wächst die Energie im Schwerpunktsystem $E_{CMS}$ für relativistische Teilchenstrahlen nur noch mit der Wurzel der Strahlenergie $E$,

$$E_{CMS} \sim \sqrt{E}$$

da die Strahlenergie fast ganz in die gemeinsame Schwerpunktsbewegung übergeht. Eine volle Nutzung der Strahlenergie wurde erst durch die Verwendung von Speicherringen möglich, bei denen zwei gegenläufige Teilchenstrahlen zur Kollision gebracht werden.

Seit 1992 ist am DESY[4] in Hamburg der erste Speicherring, der Elektronen und Protonen kollidieren läßt, in Betrieb. Er wurde Hadron Elektron Ring Anlage (HERA) getauft und erreicht beim Zusammenstoß von 820 GeV Protonen und 26,7 GeV Elektronen eine Schwerpunktsenergie von 296 GeV, ein Wert, der einem Elektronenenstrahl, bei Kollisionen mit ruhenden Protonen, von 44 TeV entspricht.

Da ein zweiter Teilchenstrahl als Zielscheibe nicht so hohe Teilchendichte bieten kann wie ein festes Target, ist es notwendig, Strahlen höchster Intensität zu erzeugen, um eine ausreichende Stoßrate zu erhalten. Diese ist gegeben durch

$$\mathcal{R} = \sigma \mathcal{L},$$

wobei $\sigma$ der Wirkungsquerschnitt der betrachteten Reaktion und $\mathcal{L}$ die Luminosität ist. Sie errechnet sich bei zwei frontal kollidierenden Teilchenstrahlen zu

$$\mathcal{L} = \frac{f n N_1 N_2}{A},$$

wobei $f$ die Umlauffrequenz, $n$ die Anzahl der Teilchenpakete entlang des Rings, $A$ den Querschnitt der sich durchdringenden Strahlen und $N_1$ und $N_2$ die Zahl der Teilchen in einem Paket bezeichnet.

[3] Stanford Linear Accelerator Center; Stanford, Kalifornien (USA)

[4] Deutsches Elektronen–Synchrotron

Allerdings führt die gegenüber Synchrotrons stark erhöhte Intensität des Protonenstrahles, nicht nur zu einer hohen Luminosität, sondern auch zu einer hohen Rate an unerwünschten Untergrundereignissen, die in den Detektoren ep-Kollisionen vortäuschen. Es handelt sich dabei um Kollisionen der Strahlprotonen mit Kernen des restlichen Gases im Strahlrohr und mit Komponenten des Strahlführungssystemes. Da bei HERA die Rate dieser Untergrundereignisse die der ep-Kollisionen um etwa fünf Größenordnungen übersteigt, kommt dem, für die Auswahl und Filterung der beobachteten Ereignisse zuständigem Triggersystem, eine große Bedeutung zu.

Bei H1, einem der beiden großen Kollisions–Experimente bei HERA, ist das Triggersystem in vier Stufen aufgeteilt, um die Untergrundrate schrittweise zu senken. In der vorliegenden Arbeit wird eine neue Anwendung der auf künstlichen neuronalen Netzen (KNN) basierenden zweiten Triggerstufe untersucht. Anhand von Monte–Carlo Simulationen wird gezeigt, daß sich bei Ereignissen mit Spuren im neuen Silizium–Minivertex–Detektor BST die gewünschte Klassifikation anhand des Reaktionsortes (Vertex) erreichen läßt. Es wird gezeigt, daß sich die Rate einer bisher schwer zu fassenden Klasse von Untergrundereignissen dadurch auf ein Hundertstel reduzieren und eine hohe Effizienz für den Nachweis von ep-Kollisionen aufrechterhalten läßt. Insbesondere wird eine hohe Stabilität der KNN gegenüber elektronischem Rauschen des BST nachgewiesen.

# 2. Das H1 Experiment bei HERA

## 2.1 Der ep-Speicherring HERA

In HERA werden Elektronen und Protonen mit der höchsten je erreichten Schwerpunktsenergie $E_{CMS}$ von 296 GeV zur Kollision gebracht.

Die Anlage besteht aus zwei übereinander liegenden, unabhängigen Speicherringen, die sich in einem 10–25 m tiefen, ringförmigen Tunnel von 6,4 km Umfang befinden. Der begrenzende Faktor für die erreichbaren Strahlenergien ist für den Protonenstrahl die Feldstärke der supraleitende Ablenkmagnete von 4,7 T, bei den Elektronen die Beschleunigungsspannung der Hohlraumresonatoren von 145 MV pro Umlauf[1]. Durch den Einbau zusätzlicher supraleitender Hohlraumresonatoren soll diese Beschleunigungsspannung auf 241 MV pro Umlauf erhöht werden, wodurch der Elektronenstrahl seine Designenergie von 30 GeV erreichen und die Schwerpunktsenergie 314 GeV betragen wird [4].

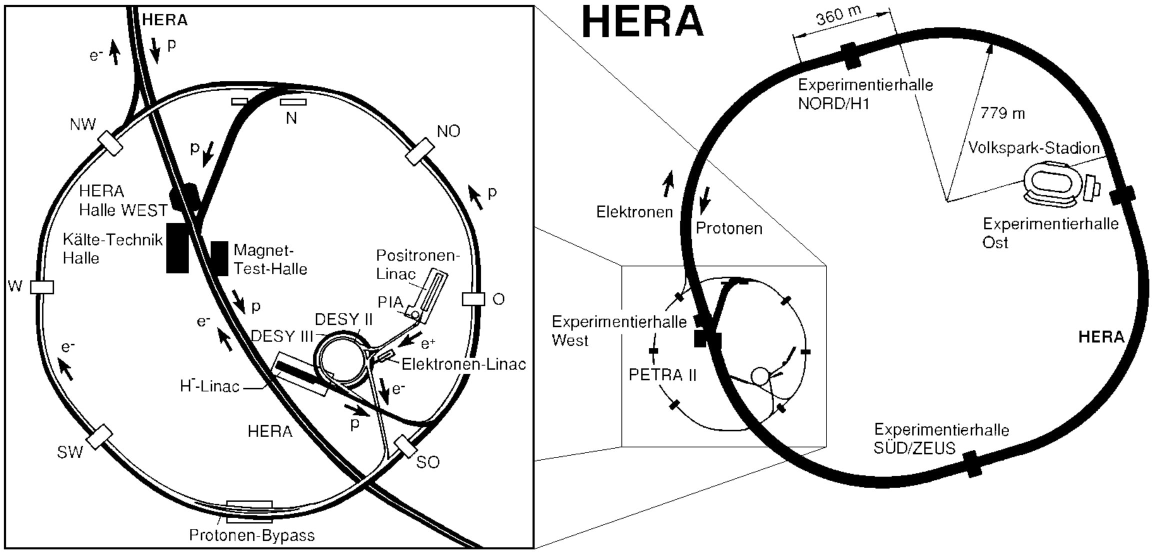



**Abbildung 2.1:** Der Elektron-Proton Speicherring HERA bei DESY in Hamburg mit Vorbeschleunigern

[1] Pro Umlauf strahlt ein 26,7 GeV Elektron eine Energie von 145 MeV in Form von Synchrotronstrahlung ab.

Um die angestrebte Luminosität von $1,5 \cdot 10^{31}\mathrm{cm}^{-2}\mathrm{s}^{-1}$ zu erreichen, sind die Strahlen in 220 Abschnitte („buckets") unterteilt, die mit insgesamt $2,1 \cdot 10^{13}$ Protonen bzw. $0,8 \cdot 10^{13}$ Elektronen gefüllt werden. Durch die Limitierung des PETRA Vorbeschleunigers auf 70 Bündel, können maximal 210 dieser Abschnitte mit Teilchen gefüllt werden. Bei einem Umfang von 6336 m bedeutet dies, daß sich die Teilchenbündel an den vorgesehenen Wechselwirkungspunkten alle 96 ns durchdringen [5]. Aufgrund des totalen Wirkungsquerschnittes der ep-Streuung wird im Mittel nur alle $10^5$ Kreuzungen eine im Detektor beobachtbare Streuung erwartet, was einer Ereignisrate von 100 Hz entspricht. Bisher wurde jedoch nur ein Zehntel der geplanten Luminosität von 100 $pb^{-1}$ pro Jahr erzielt.

| Name | Institut | Umfang [km] | Art der Teilchen | Strahlenergien [GeV] | in Betrieb seit |
|---|---|---|---|---|---|
| SPEAR | SLAC, Stanford | 0,23 | $e^-e^+$ | 4 + 4 | 1972 |
| DORIS | DESY, Hamburg | 0,29 | $e^-e^+$ | 6 + 6 | 1973 |
| CESR | Cornell, NY | 0,77 | $e^-e^+$ | 8 + 8 | 1979 |
| PETRA | DESY, Hamburg | 2,30 | $e^-e^+$ | 23 + 23 | 1978 |
| PEP | SLAC, Stanford | 2,20 | $e^-e^+$ | 18 + 18 | 1980 |
| Tristan | KEK, Tsukuba | 3,02 | $e^-e^+$ | 30 + 30 | 1987 |
| SLC | SLAC, Stanford | 1,45 | $e^-e^+$ | 50 + 50 | 1989 |
| LEP | CERN, Genf | 26,66 | $e^-e^+$ | 50 + 50 | 1989 |
| | | | | 95 + 95 | 1994 |
| $Sp\bar{p}S$ | CERN, Genf | 6,91 | $p\bar{p}$ | 310 + 310 | 1981 |
| Tevatron I | Fermilab, Chicago | 6,28 | $p\bar{p}$ | 1000 + 1000 | 1987 |
| HERA | DESY, Hamburg | 6,34 | $e^{\pm}p$ | 27 + 820 | 1992 |

**Tabelle 2.1:** Umfang und Energien der größten Speicherringe [6][7]

## 2.2 Physik bei HERA

Elektron–Proton Streuungen, wie sie bei HERA stattfinden, werden im Rahmen des Standardmodells der Teilchenphysik beschrieben. Nach dem Standardmodell sind die Grundbausteine der Materie punktförmige Fermionen, die als Leptonen oder Quarks klassifiziert werden, wobei nur letztere an der starken Wechselwirkung teilhaben. Neben der starken Wechselwirkung, die durch die Theorie der Quantenchromodynamik (QCD) beschrieben wird, sind drei weitere fundamentale Wechselwirkungen bekannt, die elektromagnetische Wechselwirkung, die schwache Wechselwirkung und die Gravitation, deren Einfluß in der Teilchenphysik zu vernachlässigen ist.

S. L. Glashow, A. Salam und S. Weinberg konnten um 1970 die elektromagnetische und die schwache Wechselwirkung durch eine einheitliche Theorie erklären [8][9][10]. Diese elektroschwache Theorie bildet zusammen mit der Quantenchromodynamik das Fundament des Standardmodells, das alle fundamentalen Wechselwirkungen durch den Austausch von Eichbosonen beschreibt. In der elektroschwachen Wechselwirkung sind dies $\gamma, Z^0, W^+, W^-$, in der starken Wechselwirkung die Gluonen $g$.

In diesem Modell wird die ep-Streuung in erster Näherung (Bornsche Näherung) durch den Austausch eines der Bosonen $\gamma, Z^0, W^-$ zwischen Elektron und einem Quark innerhalb des Protons beschreiben[2]. Da beim Austausch eines $W^\pm$ Ladung vom Lepton auf das Quark transferiert wird, werden beide Stoßpartner umgewandelt. In diesem Falle spricht man von „geladenem Strom“ (charged current, CC), das Elektron wandelt sich dabei in ein Neutrino um, das im Detektor nicht direkt nachgewiesen werden kann. Trägt das Austauschteilchen keine Ladung, spricht man von „neutralem Strom“ (neutral current, NC).

### 2.2.1 Kinematische Variablen

Bei bekannter Schwerpunktsenergie ist ein elastischer Stoß vollständig durch eine Variable bestimmt. Im Falle hochrelativistischer Projektile wird üblicherweise das lorentzinvariante, negative Quadrat des Viererimpulsübertrages gewählt.

$$Q^2 = -q^2 = -(P_l - P_l')^2 \tag{2.1}$$

$$s = (P_l + P_P)^2 \approx 4E_l E_P \tag{2.2}$$

| | |
|---|---|
| $P_l, E_l$ | Viererimpuls bzw. Energie des einlaufenden Leptons |
| $P_l'$ | Viererimpuls des auslaufenden Leptons |
| $P_P, E_P$ | Viererimpuls bzw. Energie des einlaufenden Protons |
| $s$ | Quadrat der Schwerpunktsenergie $E_{CMS}$ |

Die Näherung in (2.2) ist gültig bei Vernachlässigung der Ruhemassen von Elektron und Proton.

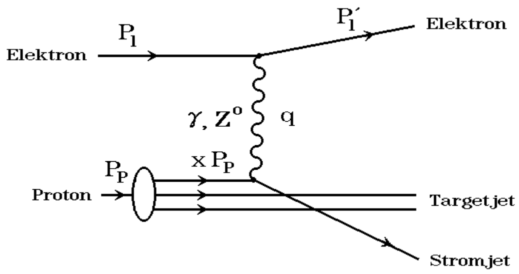
[2] Bei Positronen statt Elektronen als Stoßpartner des Protons können $W^+$ statt $W^-$ ausgetauscht werden.

Nach J. D. Bjorken und R. P. Feynman wird die inelastische Elektron–Proton Streuung als elastische Streuung des Elektrons an einem Parton innerhalb des Protons verstanden (Partonmodell) [11][12]. Die Partonen werden heute mit den Quarks identifiziert. In diesem Modell wird die Kinematik des inelastischen Falles, durch eine der dimensionslosen Lorentzskalare $x$ oder $y$ festgelegt

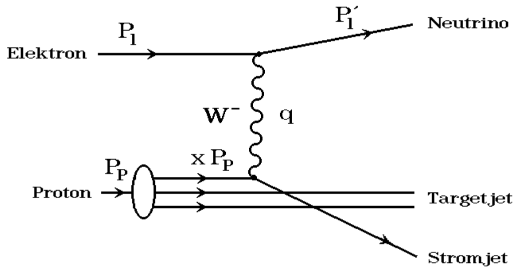
$$x = \frac{Q^2}{2P_P \cdot q} \tag{2.3}$$

$$y = \frac{q \cdot P_P}{P_l \cdot P_P}. \tag{2.4}$$

Im einfachen Partonmodell wird $x$ als Impulsanteil des einlaufenden Partons am Gesamtimpuls des Protons interpretiert, während $y$ den relativen Energieübertrag des einlaufenden Leptons im Ruhesystem des Protons festlegt und damit ein Maß für die Inelastizität des Prozesses darstellt. Bei inelastischer Streuung verläßt das gestoßene Quark das Proton und materialisiert sich zu einem engen Hadronenbündel (Stromjet), das im Detektor nachgewiesen wird.




(a) neutraler Strom




(b) geladener Strom

**Abbildung 2.2:** Feynman–Graphen der Bornstreuung

Da die Lorentzskalare $Q^2, s, x$ und $y$ gemäß $Q^2 = sxy$ voneinander abhängen und die Schwerpunktsenergie $\sqrt{s} = 296$ GeV bekannt ist, läßt sich die Ereigniskinematik durch zwei der drei Größen $Q^2, x$ und $y$ festlegen, beispielsweise durch $Q^2$ und $y$. Im Experiment geschieht dies durch Messung des Polarwinkels $\theta_l'$ und der Energie $E_l'$ des gestreuten Elektrons,

$$Q^2 = 4E_l E_l' \cos^2(\frac{\theta_l}{2}) \tag{2.5}$$

$$y = 1 - \frac{E_l'}{E_l} \sin^2(\frac{\theta_l'}{2}). \tag{2.6}$$

Zu beachten ist, daß der Polarwinkel im H1–Koordinatensystem bezüglich der Protonenrichtung gemessen wird, der Streuwinkel des Elektrons somit $180^0 - \theta_l'$ beträgt.

Alternativ lassen sich $Q^2$ und $y$ durch Messung der Impulskomponenten $P_\perp^i$, $P_{||}^i$ und der Energie $E^i$ aller nachgewiesenen Hadronen $i$ bestimmen:

$$Q^2 = \frac{1}{1-y}(\sum_i P_\perp^i)^2 \tag{2.7}$$

$$y = \frac{1}{2E_l}\sum_i (E^i - P_{||}^i). \tag{2.8}$$

Bei CC-Ereignissen bleibt das auslaufende Neutrino unbeobachtbar, so daß sich die kinematischen Variablen ausschließlich nach den Gleichungen (2.7) und (2.8) bestimmen lassen (Jacquet Blondel Methode) [13].

### 2.2.2 Ereignisklassen

Die ep-Kollisionen bei HERA werden in verschiedene Ereignisklassen eingeteilt, je nach Art der involvierten Wechselwirkungen und Ereigniskinematik. An dieser Stelle kann keine vollständige Liste der einzelnen Ereignisklassen gegeben werden, die Diskussion beschränkt sich auf drei Klassen, die in späteren Abschnitten noch Erwähnung finden.

#### Photoproduktion

Photoproduktionsereignisse besitzen den größten Wirkungsquerschnitt aller Ereignisklassen und dominieren daher die ep-Streuung bei HERA. Es handelt sich dabei um Reaktionen, die durch den Austausch quasi–reeller ($Q^2 \approx 0$) Photonen beschrieben werden, das Elektron wird dabei nur wenig abgelenkt, und läuft unter einem Polarwinkel $\theta$ nahe $180^0$ aus. Man unterscheidet den direkten Prozeß, in dem das Photon an ein Quark koppelt und den aufgelösten Prozeß, bei dem das Photon in ein Quark–Antiquarkpaar fluktuiert und mittels Gluonaustausch mit einem Quark des Protons wechselwirkt.

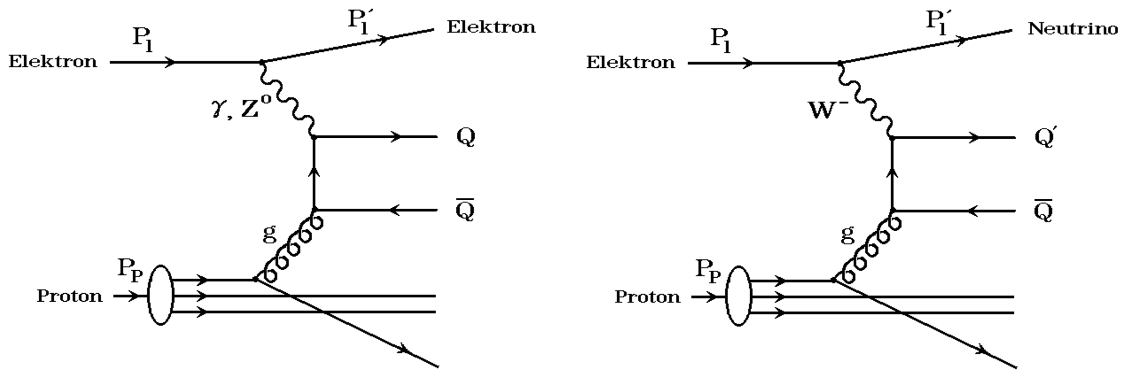
#### Erzeugung schwerer Quarks

Durch Boson–Gluon Fusion, wie in Abbildung 2.3 dargestellt, können in ep–Kollisionen schwere Quarks ($c$ charm, $b$ bottom) erzeugt werden, bei den von HERA erreichten Energien dominiert $\gamma$–Gluon Fusion.

Mit den inklusiven Wirkungsquerschnitten von

$$\begin{aligned} \sigma(ep \to cX) &\approx 500\ nb \\ \sigma(ep \to bX) &\approx 5\ nb \end{aligned} \tag{2.9}$$

können bei der geplanten Luminosität von 100 $pb^{-1}$ pro Jahr etwa $10^8$ $c, \bar{c}$ und $10^6$ $b, \bar{b}$ Quarks erzeugt werden [14].




(a) neutraler Strom (b) geladener Strom

**Abbildung 2.3:** Feynman–Graphen der Boson–Gluon Fusion

### Physik bei kleinem $x$

Bei kleinen Werten der Bjorkenschen Skalenvariable $x$ (auch $x_{Bjorken}$ genannt) stößt das Partonmodell der quasifreien Partonen innerhalb des Protons an seine Grenzen, die geforderte Unabhängigkeit der Strukturfunktionen von $Q^2$ (Skaleninvarianz), die sich aus der Annahme punktförmiger Partonen ergibt, wird verletzt. Dies läßt sich durch die Anwesenheit von Gluonen innerhalb des Protons erklären. Aus der beobachteten Abweichung von der Skaleninvarianz kann auf die Gluonenverteilung geschlossen werden. Störungstheoretische QCD Berechnungen sagen einen steilen Anstieg der Gluonendichte für $x \to 0$ voraus [15][16], was zu unphysikalisch hohen Wirkungsquerschnitten bei sehr kleinem $x$ führt. Man erwartet, daß Abschirmungseffekte und Rekombination von Gluonenpaaren zu einer Sättigung führen werden.

Ein Ziel von HERA ist, den Übergangsbereich zwischen störungstheoretischer QCD und nicht störungstheoretischer QCD bei kleinem $x$ zu erforschen. Dabei kann erstmals der Bereich bis $x \sim 5 \cdot 10^{-5}$ untersucht werden, zwei Größenordnungen niedriger als bisherige Festtarget–Experimente.

## 2.3 Der H1–Detektor

Die in den übereinander liegenden Ringen in entgegengesetzter Richtung umlaufenden Pakete können bei HERA an insgesamt vier Wechselwirkungszonen zur Kollision gebracht werden. Bisher werden zwei dieser Bereiche für Kollisions–Experimente genutzt, H1 im nördlichen und ZEUS im südlichen Teil.

Um ein möglichst vollständiges Bild der beobachteten ep-Kollisionen zu erhalten, besteht der H1–Detektor aus verschiedensten Detektorkomponenten, die den nominellen Wechselwirkungspunkt (nWWP) umgeben. Entsprechend den sehr unterschiedlichen Strahlenergien ist der Detektor in Richtung des Protonenstrahls aufwendiger instrumentiert.

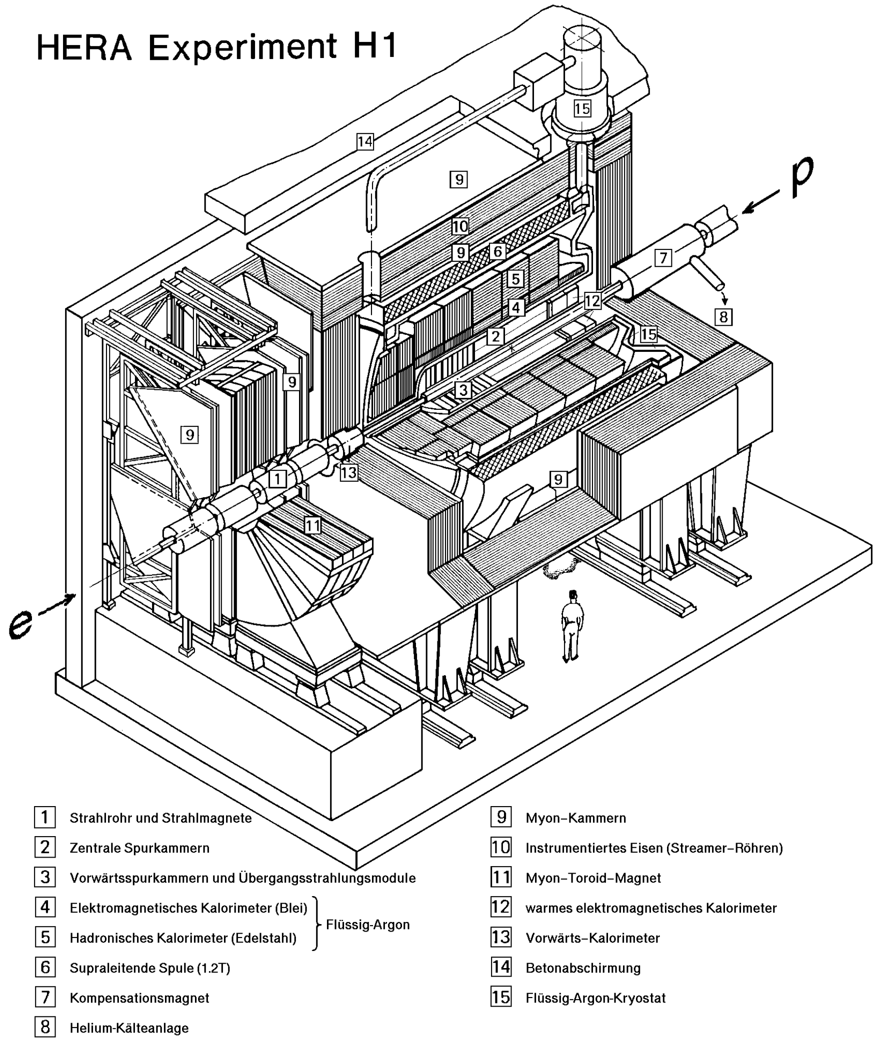



**Abbildung 2.4:** Der H1–Detektor

Das bei H1 übliche Koordinatensystem hat seinen Ursprung am nWWP, die $x$-Achse weist zum HERA–Mittelpunkt, die $y$-Achse weist vertikal nach oben und der Protonenstrahl zeigt in Richtung der positiven $z$-Achse, auch „Vorwärtsrichtung“ genannt.

### 2.3.1 Die Spurkammern

Im inneren Bereich des Detektors befindet sich axial um die Strahlröhre die zentrale Spurkammer CJC (central jet chamber), die aus einer inneren (CJC1) und einer äußeren (CJC2) Einheit besteht. Es handelt sich dabei um Driftkammern, deren Signal- und Felddrähte parallel zur Strahlröhre angeordnet sind, wobei eine Ortsauflösung von etwa 170 $\mu$m in der $xy$-Ebene ($\equiv r\phi$-Ebene) und mittels Ladungsteilung etwa 2 cm in $z$ erreicht wird. Zur Verbesserung der $z$-Auflösung auf 350 $\mu$m schließen sich innen und außen an die CJC1 dünne Driftkammern CIZ (central inner $z$-chamber) und COZ (central outer $z$-chamber) an, deren Drähte das Strahlrohr annähernd konzentrisch umgeben. Innerhalb der CIZ und außerhalb der COZ befinden sich dünne Vieldrahtproportionalkammern (MWPCs, multi wire proportional chambers), die der Vertexbestimmung für das Triggersystem dienen. Die zentrale Spurkammer deckt den Polarwinkel $\Theta$ in einem Bereich von $25^0 - 155^0$ ab, sie wird im Bereich $155^0 - 170^0$ durch die rückwärtige Driftkammer BDC (backward drift chamber) ergänzt [17]. Der Winkelbereich von $5^0 - 25^0$ wird durch das Vorwärts–Spurkammersystem FTD (forward tracking detector) abgedeckt, das sich aus drei Modulen, bestehend aus Driftkammern, Proportionalkammern und Übergangsstrahlungsdetektoren, zusammensetzt.

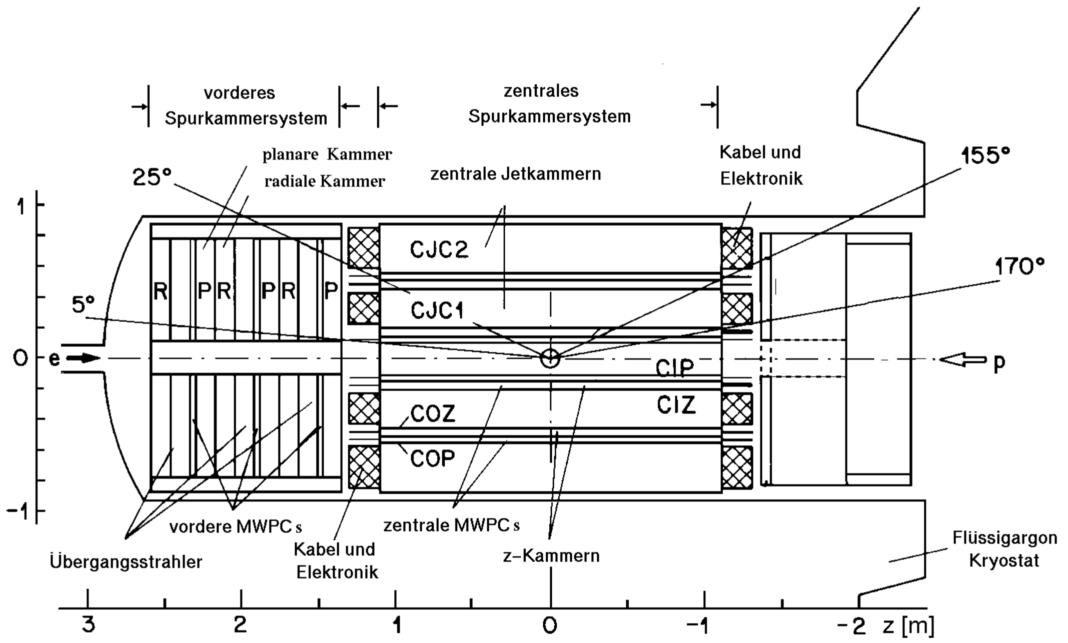



**Abbildung 2.5:** Seitenansicht des H1–Spurkammersystems

### 2.3.2 Die Kalorimeter

Das Spurkammersystem wird von mehreren Kalorimetern umgeben, so daß die Energien der Elektronen, Photonen und der hadronischen Jets gemessen werden können.

Um die Kinematik von CC Ereignissen bestimmen zu können, bei denen die Energie des auslaufenden Leptons (Neutrino) nicht gemessen werden kann, muß das Kalorimeter in der Lage sein, die Energien aller übrigen Teilchen genau zu bestimmen

#### Das Flüssigargon Kalorimeter

Um genaue Energiemessung von Elektronen und Photonen mit guter Energiemessung der hadronischen Jets zu vereinen, ist das Flüssigargon Kalorimeter LAC (Liquid Argon Calorimeter) in einen inneren elektromagnetischen Teil und einen äußeren hadronischen Teil aufgeteilt. Im elektromagnetischen Teil werden 2,4 mm dicke Bleiplatten[3] als Absorber verwendet, zwischen denen 2,35 mm dicke Schichten flüssigen Argons (LAr) als aktivem Medium dienen. Es wird eine Energieauflösung von $12\%/\sqrt{E_e} \oplus 1\%$ erreicht [19]. Im hadronischen Teil werden 16 mm starke Platten aus Edelstahl als Absorber eingesetzt, zwischen denen sich 2,4 mm dicken Schichten LAr befinden, die Energieauflösung beträgt $50\%/\sqrt{E_h} \oplus 2\%$. Um eine genaue Winkelbestimmung und die Unterscheidung elektromagnetischer und hadronischer Schauer zu ermöglichen, besitzt das LAC eine hohe Granularität der Auslesezellen, die über 45000 elektronische Kanäle ausgelesen werden. Das LAC deckt den Polarwinkel im Bereich von $4^0 - 153^0$ ab. Im extremen Vorwärtsbereich von $0,6^0 - 4^0$ wird der hadronische Energiefluß durch ein Kupfer–Silizium Sampling–Kalorimeter PLUG mit einer Energieauflösung von etwa $150\%/\sqrt{E_h}$ gemessen [20].

Das LAr Kalorimeter wird von einer supraleitenden Solenoidspule mit 5,75 m Länge und 6,08 m innererem Durchmesser umgeben. Diese Spule erzeugt ein homogenes Magnetfeld von 1,15 T parallel zur Strahlröhre, um die zur Bestimmung der Transversalimpulse $P_\perp$ und Ladungsvorzeichen nötige Bahnkrümmung der gestreuten Teilchen zu gewährleisten [5].

#### Das Blei-Szintillator Kalorimeter

Der rückwärtige Bereich wird seit 1995 von einem elektromagnetisch–hadronischen Kalorimeter, bestehend aus Bleiabsorbern und Szintillationsfasern, das wegen der Geometrie der Fasern als Spacal („spaghetti calorimeter“) bezeichnet wird, mit einer Energieauflösung von $7\%/\sqrt{E_e} \oplus 1\%$ für elektromagnetische und $\sim 29\%/E_h$ für hadronische Schauer, abgedeckt [21][22].

#### Das instrumentierte Eisenjoch

Hadronische Schauer, die nicht vollständig im LAC absorbiert werden, werden in Plastik–Streamerkammern nachgewiesen, die sich im Eisenjoch befinden, das den ma-

[3] Der Wirkungsquerschnitt der maßgeblichen Effekte, Bremsstrahlung und Paarbildung, steigt etwa quadratisch mit der Kernladungszahl [18, S.37ff].

gnetischen Fluß der Solenoidspule rückführt. Das Eisenjoch ist zusätzlich mit ortsauflösenden Kammern instrumentiert, die dem Nachweis von Myonen dienen.

### 2.3.3 Die neuen Silizium–Minivertex–Detektoren

Dem gesteigerten Interesse an Physik bei niedrigem $x_{Bjorken}$ und $Q^2$ einerseits und dem aktuellen Schwerpunkt bei der Erforschung von Schwerquarksystemen andererseits, wurde durch ein umfangreiches Erweiterungsprogramm Rechnung getragen [14]. Dabei wurde im Bereich des Detektors eine neue Strahlröhre mit nur 45 mm statt bisher 95 mm Radius [23][24] eingesetzt, um die herum kleine Spurdetektoren noch innerhalb der CIP plaziert wurden.

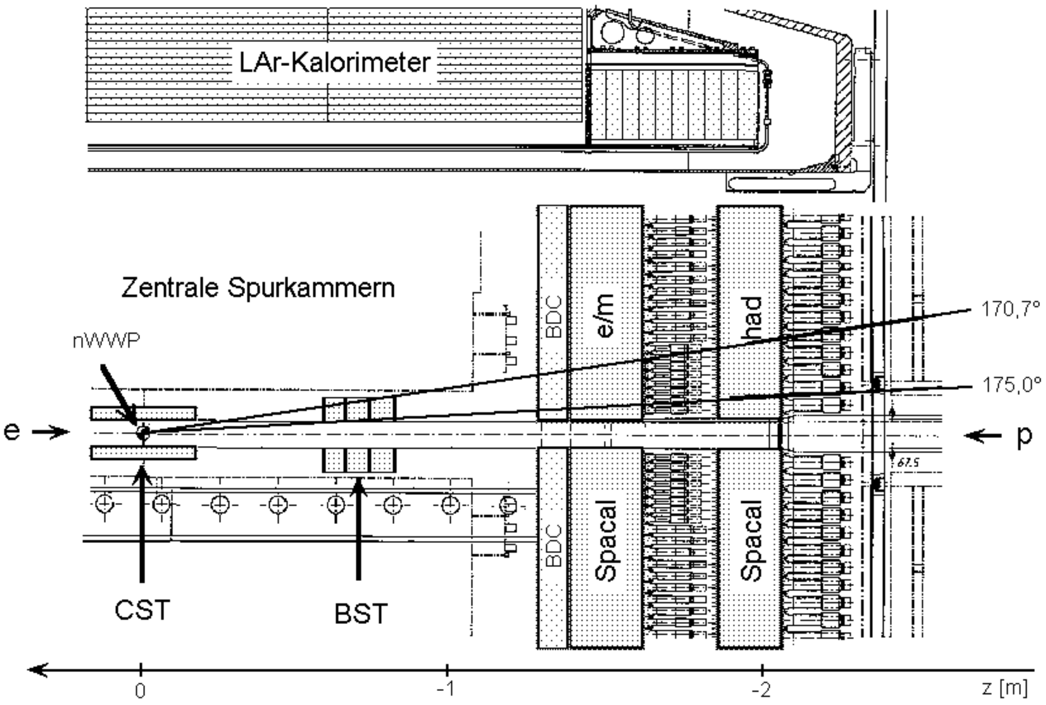



**Abbildung 2.6:** Die Erweiterungen des H1–Detektors, CST, BST, BDC und Spacal

In größtmöglicher Nähe des nWWP befindet sich der zentrale Si–Minivertex–Detektor CST (central silicon tracker). Die Geometrie des CST wird durch zwei reguläre Prismen beschrieben, wobei das innere, zwölfseitige Prisma das Strahlrohr mit einem Radius von 57,5 mm umgibt, das äußere, zwanzigseitige im Radius von 97 mm. Jede Seite besteht aus sechs doppelseitigen Si–Streifendetektoren mit einer Gesamtlänge von 356 mm. Es soll eine Ortsauflösung von 60 $\mu$m in $z$ erreicht werden, so daß Primär- und Sekundärvertex des Zerfalls von Teilchen mit $c$- und $b$-Quarks getrennt werden können [23].

Zur Beobachtung der in Kollisionen mit niedrigem $x_{Bjorken}$ und $Q^2$ nur schwach gestreuten Elektronen dient der rückwärtige Si–Minivertex–Detektor BST (backward silicon tracker), der den Polarwinkelbereich $170,7^0 - 175,0^0$ abdeckt. Dieser Detektor wird ausführlicher in Abschnitt 2.6 beschrieben.

## 2.4 Das Triggersystem

Die gesamte Detektorinformation wird über 270000 elektronische Kanäle ausgelesen und erreicht einen Datenumfang von etwa 3 Mbyte pro Ereignis. Durch Nullunterdrückung (zero–suppression) der von leeren Auslesezellen gelieferten Signale, werden die Daten auf 50–100 Kbyte pro Ereignis komprimiert.

Dies limitiert die Rate, mit der die Ereignisse dauerhaft auf Magnetbändern gespeichert werden können bei einer maximalen Schreibgeschwindigkeit von 1,2 Mbyte/s auf etwa 10 Hz. Die Rate der interessanten physikalischen Reaktionen wird bei Designluminosität im Bereich von etwa 1 kHz für weiche Photoproduktion bis weit unter 1 Hz für Ereignisse mit hohem $Q^2$, wie z.B. CC–Ereignisse, liegen.

### 2.4.1 Untergrundereignisse

Demgegenüber steht eine erwartete Rate an Untergrundereignissen von mehreren 100 kHz, die im wesentlichen von Kollisionen des Protonenstrahls mit Kernen des Restgases im Strahlrohr (Strahl–Gas–Reaktion) und mit Komponenten des Strahlführungssystemes (Strahl–Wand–Reaktion) herrühren.

#### Strahl–Gas–Reaktionen

Bei einer Strahl–Gas–Reaktion wechselwirkt das einfallende Proton mit einem der Nukleonen im Atomkern des Restgases im Strahlrohr. Bei der resultierenden Schwerpunktsenergie von 39 GeV kann in erster Näherung die Isospineinstellung des Nukleons vernachlässigt und der Stoß als pp-Kollision verstanden werden ($\sigma_{pp} \simeq \sigma_{np}$). Die Rate der Strahl–Gas–Ereignisse pro Meter $\mathcal{R}_{SGE}$ ist gegeben durch das Produkt des Wirkungsquerschnittes des Kernes mit der aus der allgemeinen Gasgleichung zu berechenden Flächendichte der Kerne und der Zahl der einfallenden Protonen pro Zeiteinheit.

$$\mathcal{R}_{SGE} = \sigma_{pp} A^{2/3} \frac{P}{kT} n_p f \tag{2.10}$$

| | |
|---|---|
| $\sigma_{pp}$ | totaler Wirkungsquerschnitt einer 39 GeV pp-Kollision |
| $A$ | Massenzahl des Kerns |
| $P, T$ | Gasdruck, Temperatur |
| $n_p, f$ | Zahl der Protonen im Ring, Umlauffrequenz |

Mit $\sigma_{pp} = 40$ mb und einem Druck von $3 \cdot 10^{-9}$ Torr ergibt sich eine Rate von grob

2 kHz/m bei Designluminosität[4]. Der Detektor ist noch auf Strahl–Gas–Ereignisse empfindlich, deren Vertex bis zu 100 m in rückwärtiger Richtung liegt ($z \geq -100$ m). Ab einer Entfernung von etwa 10 m erreichen nicht mehr Fragmente des primären Strahl–Gas–Ereignisses den Detektor, da sie von Kollimatoren an der Detektorfront abgefangen werden, sondern die in den Kollimatoren oder der Wand des Strahlrohres erzeugten sekundäre Teilchenschauer. Eine Abschätzung der Rate des Strahl–Gas–Untergrundes mittels Simulationen ergab eine obere Grenze von 45 kHz, wobei eine Unsicherheit von 500% angegeben wurde [26].

#### Strahl–Wand–Reaktionen

Während die Schwingungen der Elektronen um die Sollbahn durch die Abgabe von Synchrotronstrahlung beständig gedämpft werden, weitet sich der Protonenstrahl durch Strahl–Strahl Wechselwirkung mit der Zeit auf. An Wolframkollimatoren kontrollierbarer Apertur, die sich im westlichen geraden Teilstück von HERA befinden, werden die am weitesten außerhalb der Strahlmitte befindlichen Protonen zum überwiegenden Teil aus dem Halo entfernt. Die Rate, mit der Haloprotonen den Kollimator durchdringen läßt sich unter der Annahme, daß die Teilchenverluste im Protonenring lediglich vom Abstreifen an den Wolframkollimatoren herrührt, gemäß

$$\mathcal{R}_{offmom} = \frac{n(1-f)}{T_{1/2}\ln 2} \tag{2.11}$$

| | | |
|---|---|---|
| $n$ | Anzahl der Protonen im Ring | $2,1 \cdot 10^{11}$ |
| $T_{1/2}$ | Halbwertszeit des Protonstrahles | 20 h |
| $f$ | Wirkungsgrad der Kollimatoren | $99,5\%$ |

zu 10 MHz berechnen. Die gestreuten Protonen besitzen zu 90% Polarwinkel zwischen $0,05$ und $0,2$ mrad, so daß sie weite Strecken zurücklegen können, bevor sie aus der Strahlröhre austreten und Teilchenschauer hervorrufen. Die Rate, mit der diese Strahl–Wand–Ereignisse zu Signalen im Detektor führen, hängt empfindlich von der Aperturgeometrie im detektornahen Bereichab und ist daher schwer abzuschätzen. Simulationen zufolge sind 40–150 kHz zu erwarten [26]. Die beobachteten Raten liegen in diesem Bereich, aufgrund wechselnder Strahlbedingungen sind sie großen Schwankungen unterworfen.

### 2.4.2 Die Triggerstufen

Da bei jeder, der alle 96 ns stattfindenden Bündeldurchdringungen (bunch cross, BC) eine ep-Kollision auftreten kann, ergibt sich eine Rate der „Ereigniskandidaten“ von 10,4 MHz. Die Aufgabe des Triggersystems ist es, aus der überwältigenden Anzahl der pro Sekunde anfallenden Ereigniskandidaten ungefähr zehn auszuwählen, die für die weitere Analyse auf Magnetbändern gespeichert werden können. Diese Auswahl kann allerdings nicht innerhalb von 96 ns getroffen werden, da bereits die Signallaufzeiten

[4] 1 Torr = $133,3224$ Pa [25, S.146]

in den Kabeln mehrere BC betragen können. Darüber hinaus betragen die Signalformungszeiten einiger Detektorkomponenten viele BC:

13 BC beim LAC, bestimmt durch die Integrationszeiten der Vorverstärker,

11 BC bei den zentralen Driftkammern, verursacht durch Driftzeiten bis zu 1 $\mu$s.

Aus diesem Grund speichert jeder Subdetektor seine Daten in einem nach dem first-in-first-out Prinzip organisierten Puffer (Pipeline) zwischen, wobei dessen Tiefe $27-35$ BC beträgt, entsprechend einer Zeitspanne von $2,6-3,4$ $\mu$s [5]. Während dieser Zeitspanne getroffene Triggerentscheidungen erfolgen totzeitfrei, allerdings können in dieser kurzen Zeit nur sehr einfache Entscheidungskriterien benutzt werden.

Die geforderte Reduktion der Eingangsrate läßt sich nicht ausschließlich durch diese einfachen Kriterien erreichen, es werden komplexere Algorithmen benötigt, die entsprechend mehr Zeit benötigen. Daher wird das H1–Triggersystem in vier Stufen (Level) aufgeteilt, denen zunehmend mehr Zeit für eine Triggerentscheidung zur Verfügung steht. Bisher wurden nur die erste und die vierte Triggerstufe eingesetzt, die zweite Triggerstufe wurde in den letzten Wochen der Datennahme von 1995 erfolgreich getestet.

#### Erste Triggerstufe – L1

Nach Synchronisation aller Subdetektoren verbleibt der ersten Triggerstufe (L1–Trigger) eine Tiefe der Pipeline von 24 BC, entsprechend 2,3 $\mu$s. In dieser Zeit werden durch Diskriminierung und logische Verknüpfung der Triggersignale der einzelnen Subdetektoren (Triggerelemente) sogenannte Subtrigger gebildet. Gibt einer dieser Subtrigger ein positives Signal (L1-Keep), so wird die Pipeline angehalten und die Auslese des Ereignisses begonnen. Ab diesem Zeitpunkt können keine weiteren Ereignisse aufgenommen werden, die Totzeit beginnt.

Falls die Rate der von L1 getriggerten Ereignisse zu groß wird, müssen Subtrigger mit besonders hoher Rate herabskaliert werden. In diesem Fall wird nur jede $n$-te positive Triggerentscheidung jenes Subtriggers beachtet, die Rate und die Effizienz des Subtriggers werden also den Faktor $n$ (Prescale-Faktor) herabgesetzt.

Da die Auslese eines Ereignisses etwa 1,5 ms benötigt, können die von L1 getriggerten Ereignisse während der Datenauslese einer wesentlich genaueren Analyse unterzogen werden. Dies ist die Aufgabe der Stufen zwei und drei, die von L1 getriggerte Untergrundereignisse erkennen und durch ein L2- bzw. L3-Reject Signal die Datenauslese stoppen sollen.

#### Zweite Triggerstufe – L2

Ab Auftreten einer positiven Triggerentscheidung der ersten Stufe stehen der zweiten Triggerstufe 20 $\mu$s zur Verfügung, um durch „intelligentere“ Nutzung der Triggerdaten Untergrundereignisse zu erkennen und deren Auslese zu stoppen, wodurch die Totzeit reduzieren wird. Hierfür sind zwei Konzepte vorgesehen:

1. ein auf zweidimensionalen topologischen Relationen basierender „L2–Topo–Trigger“ [27],

2. ein auf Mustererkennung im hochdimensionalen Raum der Triggergrößen basierender „L2–Neuro–Trigger“, der KNN Algorithmen benutzt [28][29][30].

#### Dritte Triggerstufe – L3

Wird ein Ereignis von L2 akzeptiert, beginnen die zeitintensiven Ausleseoperationen wie Nullunterdrückung der digitalen Driftkammersignale und Digitalisierung der analogen Kalorimetergrößen. Aufgabe des L3–Triggers ist es, in dieser Zeit Untergrundereignisse herauszufiltern, die von L2 nicht verworfen wurden. Dazu wird ein frei programmierbares System von AM 29000 RISC Prozessoren benutzt, für die Berechnungen stehen maximal 800 $\mu$s zur Verfügung [31]. Der L3–Trigger wurde bisher nicht eingesetzt.

#### Vierte Triggerstufe – L4

Nach beendeter Datenauslese steht der vierten Triggerstufe die gesamte Detektorinformation als Rohdaten zur Verfügung, um die letzte Filterung der Ereignisse vor der Speicherung auf Magnetband vorzunehmen. Dazu werden über 30 Prozessoren vom Typ MIPS R3000 benutzt, die in einer sogenannten Prozessorfarm parallel arbeiten [32]. Durch Zwischenspeicherung der Rohdaten der einzelnen Ereignisse kann L4 asynchron betrieben werden und somit eine optimale Auslastung der Prozessoren erreichen. Bleibt die Eingangsrate unter der maximal verarbeitbaren Durchschnittsrate von derzeit ungefähr 50 Hz, so arbeitet L4 totzeitfrei. Die Ausgangsrate ist durch die Schreibgeschwindigkeit auf Magnetband auf etwa 10 Hz limitiert.

Um die Funktionsweise des Triggersystems und die Untergrundbedingungen überwachen zu können, wird etwa ein Prozent der von L4 verworfenen Ereignisse auf Monitorbändern gespeichert.

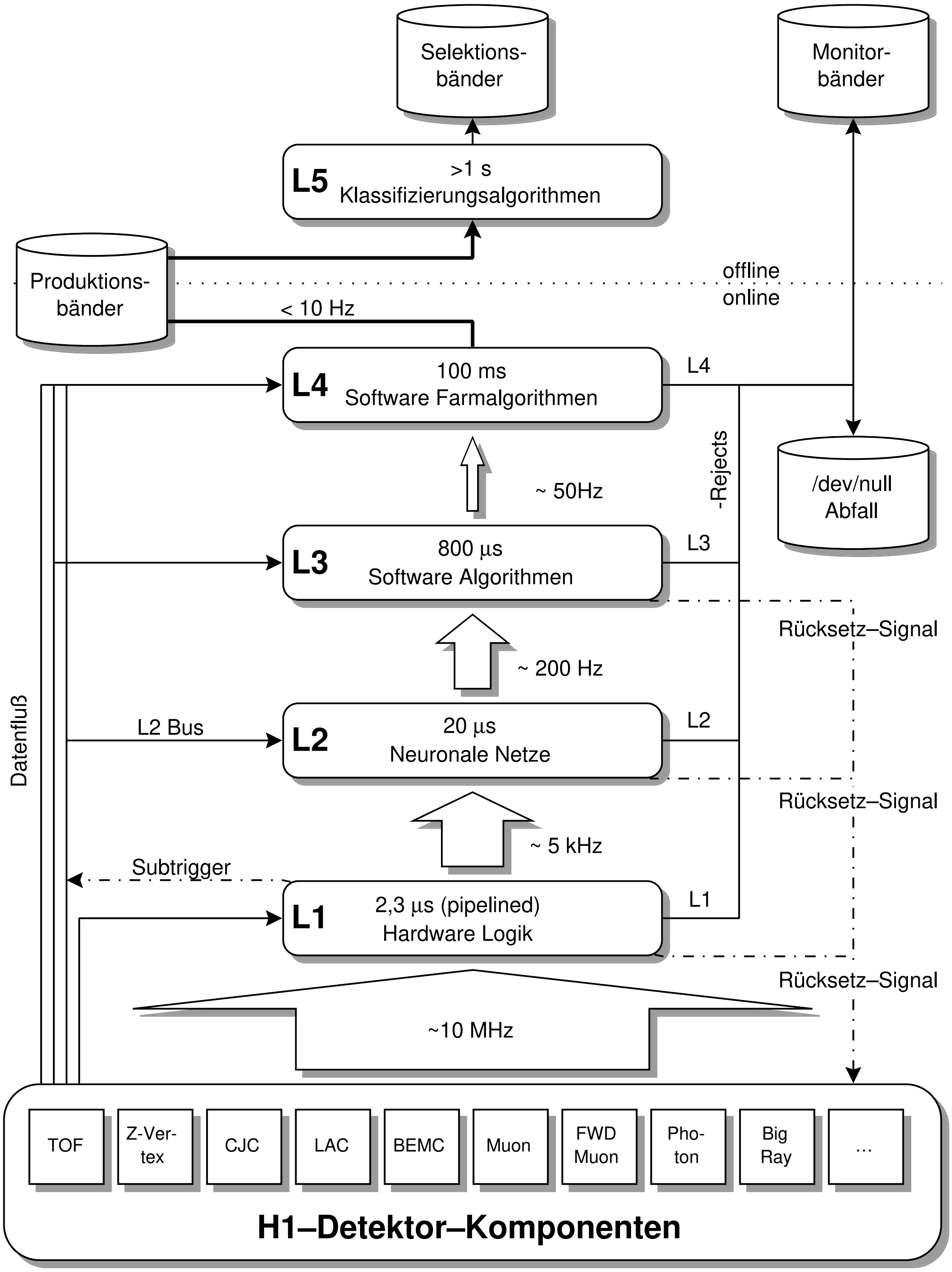



**Abbildung 2.7:** Das H1–Triggersystem. Dargestellt sind die Triggerstufen L1–L4 und deren maximale Eingangsrate und Verarbeitungszeit. Zur Vollständigkeit ist die erste „offline"–Verarbeitungsstufe (L5) dargestellt, die die auf Band geschriebenen Ereignisse klassifiziert (Abbildung aus [33]).

## 2.5 Der L2–Neuro–Trigger

Der zweiten Triggerstufe stehen im wesentlichen die gleichen Daten wie der ersten Stufe zur Verfügung. Im Gegensatz zu L1 sind diese Daten für L2 gleichzeitig verfügbar, so daß eine Filterung der von L1 getriggerten Ereignisse anhand hochdimensionaler Korrelationen der Daten möglich wird. Die Beschränkung der Bearbeitungszeit auf 20 $\mu$s erfordert das Parallelisieren der Berechnungen.

KNN Algorithmen sind aufgrund ihrer inhärenten Parallverarbeitung und ihrer Eignung für die Mustererkennung eine problemgerechte Methode, eine verbesserte und dennoch schnelle Klassifikation zu erreichen. KNN stellen einen sehr allgemeinen Ansatz dar, so daß die Methodik weitgehend unabhängig von den Details der jeweiligen Problemstellung ist. Untersucht wurden unter anderem:

- Die Reduktion der Rate einzelner Subtrigger, bei hoher Effizienz im Erkennen einer Physikklasse, mit dem Ziel ein „Prescaling“ überflüssig zu machen [34][33][35].
- Eingrenzen des Untergrundes (Background Encapsulation), um dadurch sensitiv auf Ereignisse zu sein, die weder als Untergrund noch durch bekannte Physik erklärt werden können („neue Physik“)[36].
- Auswahl spezieller Ereignisklassen, die wegen untergrundähnlicher Signatur durch den Background Encapsulator verworfen würden.

Dabei wird ein Modulares Konzept verfolgt, das ein KNN pro Physikklasse vorsieht.

In dieser Arbeit wird eine weitere Anwendungsmöglichkeit untersucht, deren Ziel die Untergrundfilterung durch Vertexerkennung ist.

## 2.6 Der rückwärtige Si–Minivertex–Detektor BST

### 2.6.1 Halbleiterdetektoren

Ionisierende Strahlung erzeugt beim Durchgang durch Halbleiter Elektron–Loch Paare, die analog zu den Elektron–Ion Paaren in Gasdetektoren durch ein elektrisches Feld gesammelt werden können. Als Halbleiterdetektoren werden Dioden benutzt, an denen eine Sperrspannung angelegt ist, so daß ein Raumladungsbereich entsteht, der frei von beweglichen Ladungsträgern ist (Depletionszone). Die durch ionisierende Strahlung in der Depletionszone erzeugten Elektron–Loch Paare können nicht rekombinieren und erzeugen ein der Ionisierung proportionales Signal. Die intrinsische Effizienz von Halbleitern liegt für geladene Teilchen bei praktisch 100%, für hochenergetische Photonen oberhalb 1 MeV bei dünnen Halbleiterschichten ($\approx 300\ \mu$m) nur noch bei $0,1\% - 1\%$ [37].

Entsprechend der gegenüber gasgefüllten Drahtkammern erheblich höheren Dichte der Halbleitermaterialien, arbeiten Halbleiterdetektoren mit sehr viel kleineren sensitiven Volumina. Sie können daher sehr kompakt gebaut werden, erreichen Ortsauflösungen $< 10\ \mu$m und Auslesezeiten im Bereich von 10 ns. Da die durchschnittliche Energie zur Erzeugung eines Elektron–Loch Paares in Silizium bei Raumtemperatur nur 3,6 eV beträgt, während sie in Gasen (zur Erzeugung eines Elektron–Ion Paares) typischerweise bei 20–40 eV liegt, lassen sich mit Siliziumdetektoren bessere Energieauflösungen erreichen. Darüber hinaus sind Si–Halbleiterdetektoren einfach zu betreiben, da sie weder Hochspannung noch Kühlung benötigen [18].

Da allerdings das elektronische Rauschen in Si–Halbleiterdetektoren eine starke Temperaturabhängigkeit zeigt (bei 10 K Temperaturanstieg verdreifacht sich der Sperrstrom), ist für konstante Temperatur zu sorgen. Zu beachten sind außerdem mögliche Strahlungsschäden, die zu höherem Sperrstrom und damit verringertem Signalabstand einerseits und schlechterer Energieauflösung durch Fehlstellen im Kristall andererseits führen können [38]. Untersuchungen an einseitigen Si–Streifendetektoren mit kapazitiver Auslese haben ergeben, daß Dosen von ein paar Mrd durch ionisierende Teilchen und Fluenzen von einigen $10^{17}$ m$^{-2}$ durch Neutronenstrahlung toleriert werden können [39]. Demzufolge ist primär die Ausleseelektronik gefährdet, die, Experimenten mit einer radioaktiven $Co^{60}$ Quelle zufolge, eine Dosis von etwa 150 krd toleriert[5] [14].

### 2.6.2 Aufbau und Funktion des BST

In seiner ersten Ausbaustufe besteht der BST aus vier kreisförmigen Scheiben identischer Bauart, die senkrecht zum Strahlrohr angebracht sind. Die Scheiben besitzen eine innere Öffnung mit 56,5 mm Radius, durch die das Strahlrohr verläuft, der äußere Radius der Scheiben wird durch die CIP limitiert und beträgt 123 mm. Jede Scheibe besteht aus 16 gleichartigen Sektoren, auf denen ein $r$–Streifendetektor und ein als Spurtrigger geplanter Si–Flächendetektor (BST-T) angebracht sind.

[5] 1 rd = $10^{-2}$ Gy [25, S.49]

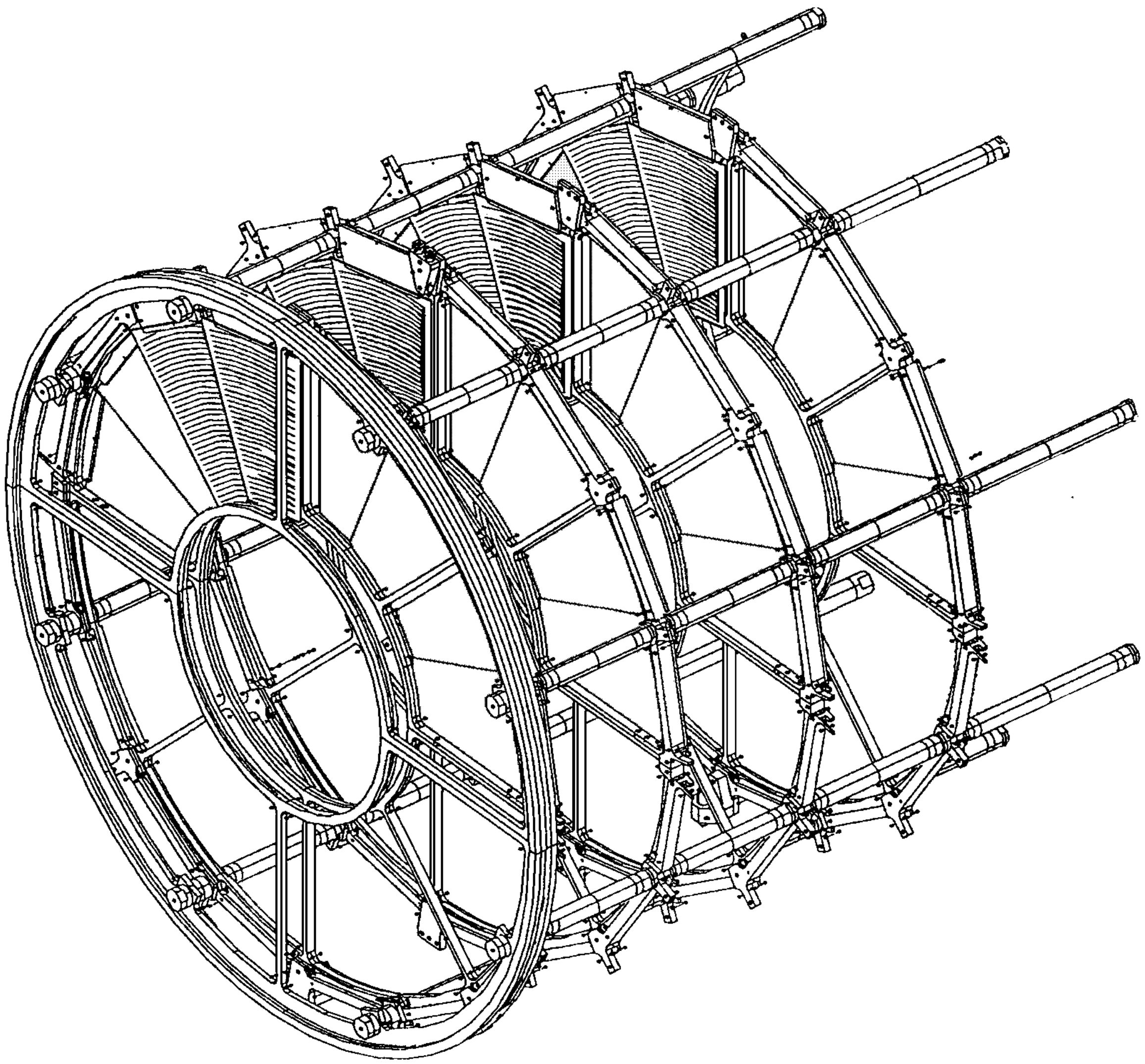


**Abbildung 2.8:** Ansicht des BST in der ersten Ausbaustufe

In der zweiten Ausbaustufe soll der BST aus acht Scheiben bestehen, auf denen sich zusätzlich ein $\phi$–Streifendetektor befindet. Fordert man, daß mindestens drei Scheiben getroffen werden, so deckt der BST den Polarwinkelbereich $170,7^0 < \theta < 175,0^0$ bzw. $164,3^0 < \theta < 175,8^0$ in der zweiten Ausbaustufe ab.

Das erwartete räumliche Auflösungsvermögen des $r$–Streifendetektors liegt bei 30 $\mu$m, so daß sich der Polarwinkel der am nWWP gestreuten Elektronen bis auf etwa 0,5 mrad genau bestimmen läßt. Der $\phi$–Streifendetektor dient der Bestimmung des Ladungsvorzeichens und der Messung des Transversalimpulses, wobei eine Auflösung von 10% bei Impulsen um 10 GeV/c erwartet wird [23][40][41]. Im Jahr 1995 waren nur vier Halbscheiben des BST installiert, die komplette erste Ausbaustufe soll 1996 eingesetzt werden, die zweite Ausbaustufe soll im Jahr 1997 erreicht werden.

### 2.6.3 Der rückwärtige Si–Spurtrigger

Der rückwärtige Si–Spurtrigger BST-T (backward silicon track trigger) soll auf Elektronen ansprechen, die in Kollisionen mit niedrigem $x_{Bjorken}$ und $Q^2$ in den extrem rückwärtigen Bereich ($\theta > 170^0$) gestreuten werden. Der BST-T besteht aus relativ großflächigen (14,3–175,0 mm$^2$), annähernd trapezförmigen Siliziumstücken, die im weiteren „Pixel" genannt werden. Die Pixel bestehen aus 380 $\mu$m starken Siliziumschichten mit direkter Auslese, ihre Kapazität liegt je nach Größe bei 7–50 pF. Der BST-T ist radial in acht Ringe und azimutal in 16 Sektoren aufgeteilt. Sektor für Sektor werden die analogen Signale der 32 Pixel pro Scheibe durch einen integrierten Vierkanal–Chip verstärkt und diskriminiert, die angestrebte Ansprechzeit liegt bei 45 ns [42, S.7]. Die diskriminierten Signale der vier Pixel je Ring werden durch logisches ODER zusammengefaßt. Je vier gemeinsam ausgelesene Pixel werden im weiteren als „Pad" bezeichnet[6].

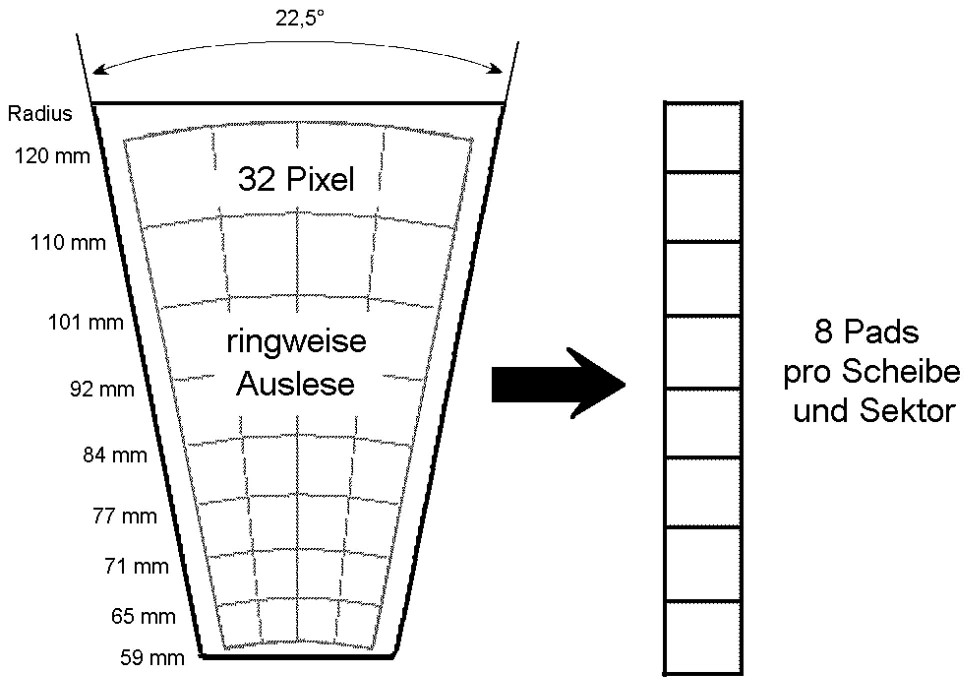



**Abbildung 2.9:** Ein Sektor einer Scheibe des BST-T

Insgesamt werden 512 Pads ausgelesen[7]. Die Radien der Ringe und die Positionen der Scheiben entlang des Strahlrohres wurden so gewählt, daß sich Kanäle bilden, die auf den nWWP zulaufen.

[6] In der Literatur wird die Bezeichnung „Pad" sowohl für die Pixel als auch die Pads verwendet [42, S.5].

[7] 4 Ebenen á 16 Sektoren á 8 Ringe

Da alle Scheiben baugleich sind, bedeutet dies, daß sich die möglichen Positionen $z_i$, an denen die Scheiben postiert werden gemäß

$$z_{i+1} = z_i \sqrt[n]{\frac{r_{max}}{r_{min}}} \tag{2.12}$$

berechnen, dabei bezeichnen $r_{min}$ und $r_{max}$ den Innen- und Außenradius des sensitiven Detektorvolumens und $n$ die Zahl der Ringe ($n = 8$). Die am dichtesten am nWWP gelegene Position wird durch die Länge des CST zu $z_0 = -300$ mm vorgegeben. Die Sequenz der Radien ergibt sich dann zu:

$$r_{j+1} = r_j \sqrt[n]{\frac{r_{max}}{r_{min}}} \qquad \text{mit} \qquad r_1 = r_{min} \tag{2.13}$$

Die gewählten Positionen der ersten und zweiten Ausbaustufe sind Tabelle 2.2 zu entnehmen, die Radien Tabelle 2.3 [43].

| i | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $z_i$ [mm] | 300 | 328 | 359 | 392 | 429 | 469 | 512 | 560 | 612 | 670 | 732 | 800 | 875 | 957 | 1046 |
| BST-1 | | | | | | | | | √ | √ | √ | √ | | | |
| BST-2 | | | √ | √ | √ | √ | | | | | | √ | √ | √ | √ |

**Tabelle 2.2:** $z$-Positionen der BST Scheiben in der ersten und zweiten Ausbaustufe.

| j | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| $r_j$ [mm] | 59,00 | 64,50 | 70,52 | 77,10 | 84,30 | 92,16 | 100,76 | 110,16 | 120,44 |

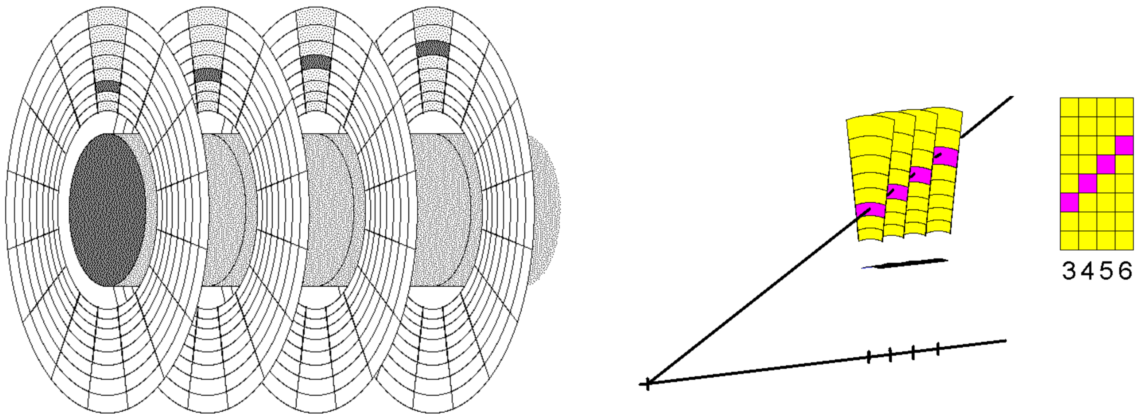
**Tabelle 2.3:** Sequenz der radialen Unterteilungen des BST-Pad–Detektors.

### 2.6.4 L1–Trigger

Die geplante Verwendung des BST-T als L1–Trigger basiert auf der Suche nach geraden Spuren in der $rz$-Ebene, die auf den nWWP zeigen. Monte–Carlo Simulationen zufolge, verläuft ab einem Elektronenimpuls von etwa 5 GeV, die überwiegende Mehrheit der Spuren innerhalb eines Sektors [44]. Daher wurde eine einfache Triggerlogik gewählt, die die einzelnen Sektoren unabhängig voneinander behandelt. Wie in Abbildung 2.10(b) dargestellt, erzeugt eine Spur innerhalb eines Sektors ein Muster von 32 Pads.

### 2.6.5 Padmuster und Masken

Die Radien der Ringe und die Positionen der Scheiben entlang des Strahlrohres wurden so gewählt, daß alle Spuren durch mindestens drei Scheiben des BST, deren Ursprung exakt am nWWP liegt, nur sieben verschiedene Padmuster erzeugen können (oberste




(a) Die 512 Pads des BST-T (b) Eine Spur und das resultierende Padmuster

**Abbildung 2.10:** Schematische Darstellung des BST-T und der Entstehung eines Padmusters

Zeile in Abbildung 2.11). Wegen der Länge der Protonenbündel finden nicht alle ep-Kollisionen genau am nWWP statt, sondern streuen über einen Bereich von etwa $z \in [-25\text{ cm}, 25\text{ cm}]$ (Wechselwirkungszone). Daher können weitere Padmuster auftreten. Dies ist in Abbildung 2.11 dargestellt.

Diese Muster werden als Masken benutzt, so daß der L1–Trigger auf Spuren aus der Wechselwirkungszone anspricht. Dies bedeutet, daß in jedem Sektor BC für BC das aktuelle Padmuster mit einem Satz solcher Masken verglichen wird und eine positive Triggerentscheidung bei Übereinstimmung in mindestens einem Sektor getroffen wird. Zusätzlich wird in einer Scheibe, Sektor für Sektor die Zahl der getroffenen Ringe gezählt. Ein Schnitt auf diese Trefferzahl soll helfen, Untergrundereignisse zu unterdrücken. Für höhere Triggerstufen werden je Sektor die Nummer der ersten passenden Maske und die Trefferzahl zur Verfügung gestellt.

### Benennung der Masken

Die ersten sieben Masken aus Abbildung 2.11 werden durch die Sequenz der Nummern der getroffenen Ringe als (0,1,2,3), (1,2,3,4), (2,3,4,5), (3,4,5,6), (4,5,6,7), (5,6,7,8) und (6,7,8,0) notiert [14]. Die Nummerierung der Ringe beginnt bei eins, null verdeutlicht, daß in dieser Scheibe kein Ring getroffen wurde.

In vorliegender Arbeit wird eine weitere Kurzschrift verwendet, in der die Masken anhand der Differenzen der Ringnummern zwischen zwei benachbarten Scheiben klassifiziert werden. In dieser Notation werden die sieben erwähnten Masken zu einer Klasse zusammengefaßt, die mit $(+1, +1, +1)$ bezeichnet wird. Die weiteren Zeilen in Abbildung 2.11 werden nach diesem Schema als (+2,+1,+1), (+1,0,+1), (+1,+1,0), (+1,+2,+1), und (+1,+1,+2) bezeichnet.

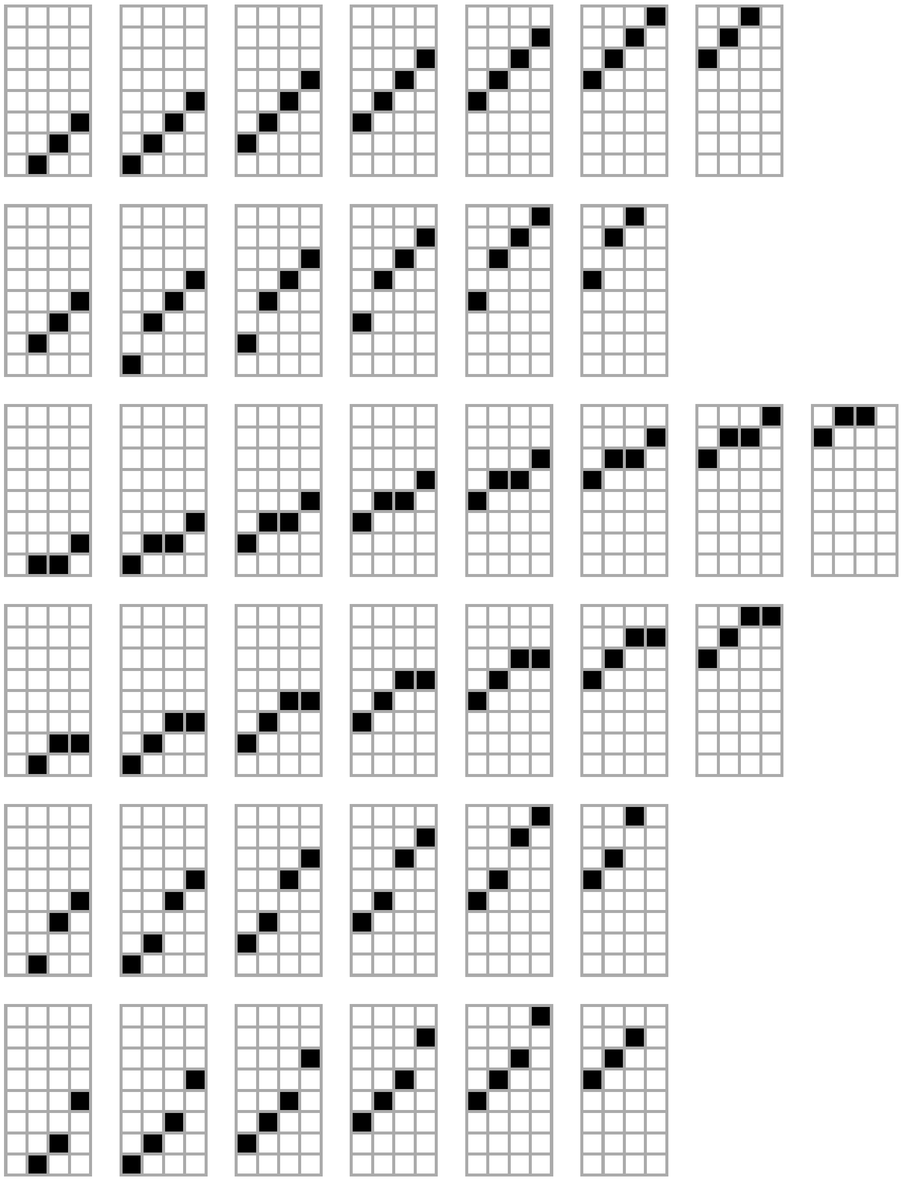


**Abbildung 2.11:** Die von Spuren aus der Wechselwirkungszone am häufigsten generierten Padmuster

# 3. Künstliche neuronale Netze

## 3.1 Biologische Grundlagen

Im Jahre 1888 erhielt der spanische Arzt Santiago Ramón y Cajal erste Hinweise für die Gültigkeit der Neuronentheorie, als er mittels einer um 1880 von Camillo Golgi entwickelten Färbetechnik die synaptischen Spalte in Nervengewebe nachwies[1]. Diese von W. von Waldeyer–Hartz begründete und durch die neurologische Forschung, insbesondere durch Arbeiten von Sherrington, Adrian und Eccles [45][46][47], bestätigte Thoerie besagt, daß das gesamte Nervensystem aus Nervenzellen (Neuronen) als funktionellen Grundeinheiten aufgebaut ist, wobei die Nervenzellen alleinige Träger neuraler Aktivität sind und sich über Synapsen in gegenseitigem Kontakt befinden.

Der Aufbau der Nervenzellen läßt sich in einen Zellkörper (Soma), mehrere Dendriten und ein Axon gliedern, wobei sich das Axon seinerseits verzweigen kann. Während die Dendriten dem Empfang der Signale anderer Nervenzelle dienen, dient das Axon der Weiterleitung der erzeugten neuralen Aktivität an weitere Nervenzellen. Bei der Signalübertragung über die synaptischen Verbindung zwischen dem Axon einer Nervenzelle und einem Dendrit einer anderen, unterscheidet man zwischen hemmender (inhibatorischer) Erregung und verstärkender (exzitatorischer) Erregung, was sich formal als Gewichtung der Signale durch die Synapsen beschreiben läßt.

## 3.2 Das Modellneuron

Basis der künstlichen neuronalen Netze ist ein einfaches mathematisches Modell der Signalverarbeitung einer Nervenzelle, das Modellneuron, in folgenden Abschnitten auch kurz als Neuron oder Knoten bezeichnet. Diese Modellbildung geht auf W. McCulloch und W. Pitts zurück, die 1943 eine Theorie der Informationsverarbeitung auf der Basis binärer Schaltelemente vorstellten [48].

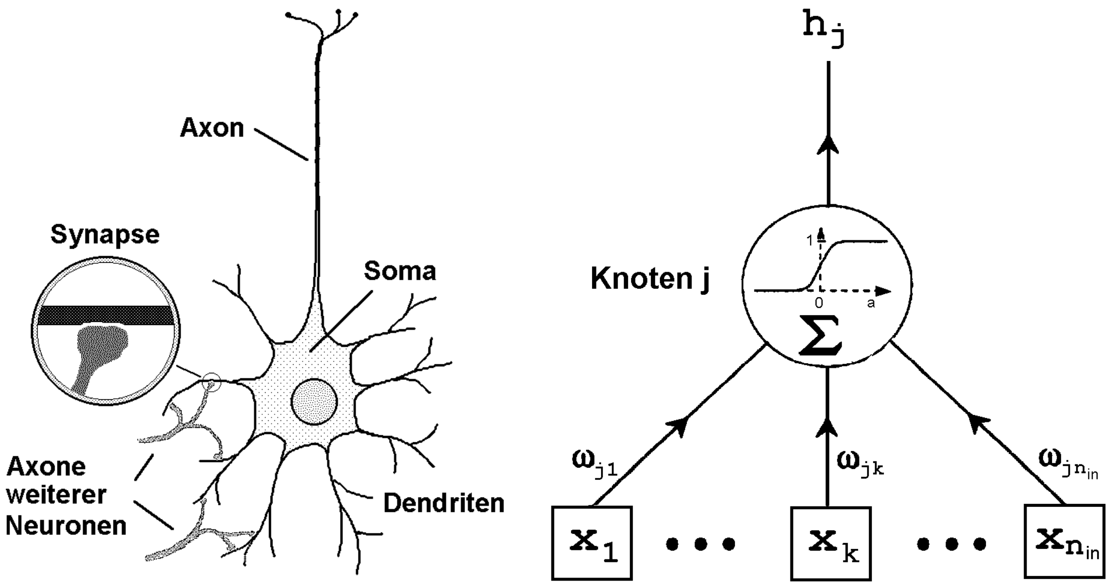
Das Modellneuron erhält einen n-dimensionalen Vektor $\vec{x}$ als Eingabe und berechnet daraus ein als Aktivierung $a$ bezeichneten Skalar, der als Argument der Transferfunktion g dient. Deren Ergebnis ist der Ausgabewert des Neurons $y = \mathrm{g}(a)$, der als Aktivität bezeichnet wird.

Im Gegensatz zu den binären McCulloch–Pitts Neuronen, benutzen die hier betrachteten Modellneuronen durchwegs reellwertige Größen, bzw. deren nährungsweisen Repräsentation durch den Computer. Die Aktivierung wird gemäß

$$a = \vec{x} \cdot \vec{\omega} - \Theta \tag{3.1}$$

als Differenz des Skalarproduktes des Eingabevektors $\vec{x}$ mit dem Gewichtsvektor $\vec{\omega}$

[1] Für diese Entdeckung wurde Golgi und Ramón y Cajal 1906 der Nobelpreis für Medizin verliehen.




(a) Schematische Darstellung einer Nervenzelle (b) Mathematisches Modell der Nervenzelle

**Abbildung 3.1:** Nervenzelle und Modellneuron

und der Schwelle $\Theta$ berechnet[2]. Positive (negative) Werte der Komponenten $\omega_i$ des Gewichtsvektors stellen die Analogie zu exzitatorischen (inhibitorischen) Synapsen her. Man erkennt, daß die Aktivierung der, nicht normierte, Abstand eines durch den Eingabevektor $\vec{x}$ festgelegten Punktes von der durch $\vec{\omega}$ und $\Theta$ festgelegten Hyperebene im n-dimensionalen Eingaberaum ist. Im einfachsten Fall handelt es sich bei der Transferfunktion g um die Heavyside–Stufenfunktion, so daß sich die Ausgabe $y = g(a)$ des Neurons interpretieren läßt als boolsche Aussage: „$\vec{x}$ besitzt bezüglich der Hyperebene $(\vec{\omega}, \Theta)$ einen positiven Abstand“.

Zwar ist das Modellneuron keine exakte Beschreibung der biologischen Nervenzelle, hinsichtlich seiner Einfachheit ist es durchaus vergleichbar. Der Schlüssel zur enormen Leistungsfähigkeit biologischer neuronaler Netze, liegt also nicht in der Komplexität der kleinsten informationsverarbeitenden Einheiten, den Nervenzellen, sondern in ihrer großen Anzahl ($\approx 10^{12}$) und der noch größeren Zahl ihrer Verbindungen ($\approx 10^{15}$) [49, S.391]. Durch Verbinden vieler Neuronen zu einem künstlichen neuronalen Netz wird versucht die typischen Leistungsmerkmale biologischer neuronaler Netze, wie Mustererkennung, Assoziation, Abstraktion, Parallelität, Fehlertoleranz und Lernfähigkeit auf künstliche Systeme zu übertragen [50].

In vorliegender Arbeit wird gezeigt, daß die untersuchten KNN außer der inhärenten Parallelität auch geeignete Algorithmen für die Mustererkennung und Fehlertoleranz verwirklichen.

[2] Häufig wird $\vec{x} \cdot \vec{\omega}$ als Aktivierung bezeichnet, da man aber bestrebt ist, $\vec{\omega}$ und $\Theta$ in einem gemeinsamen Formalismus zu behandeln, wird die hier gegebene Definition bevorzugt.

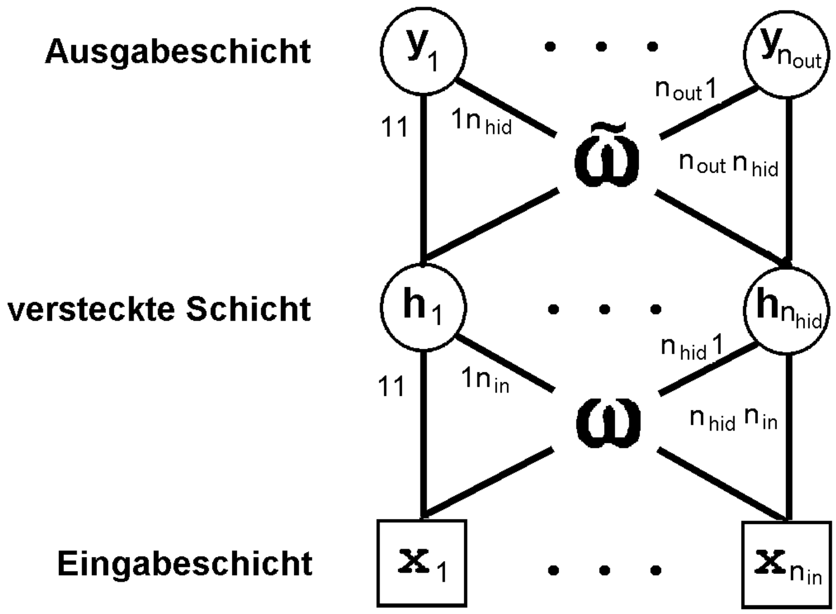



**Abbildung 3.2:** Verknüpfung der Modellneuronen in einem Feedforward Netz.

## 3.3 Das „Feedforward“ Netz

KNN lassen sich hinsichtlich der Art in der die einzelnen Neuronen vernetzt sind, der sogenannten Netztopologie, klassifizieren. Bei Netzen vom Typ „Feedforward Netz“, auch Mehrschicht Perzeptron oder Vorwärtsvermittlungsnetz genannt, lassen sich die Neuronen in mehrere hierarchische Schichten einteilen.

Abbildung 3.2 zeigt den Graph eines dreischichtigen Feedforward Netzes, die Knoten des Graphs symbolisieren die Neuronen, die Kanten[3] des Graphs stellen die Verbindungen der einzelnen Neuronen dar. Alle Knoten innerhalb einer Schicht, mit Ausnahme der Eingabeschicht, erhalten denselben Eingabevektor, der durch die Ausgabewerte der Knoten der tiefer liegenden Schicht gegeben ist. Die Knoten der Eingabeschicht dienen als Datenpuffer für den Eingabevektor und führen ihrerseits keine Berechnungen durch. Die Namen Feedforward- und Vorwärtsvermittlungsnetz drücken die Tatsache aus, daß der Informationsfluß von der Eingabeschicht zur Ausgabeschicht Schicht für Schicht vorwärts vermittelt wird, ohne daß es zu Rückkopplungen kommt.

Um einen mathematischen Ausdruck der Ausgabe eines dreilagigen Feedforward Netzes zu erhalten, wird die Gewichtsmatrix $\omega$ definiert, deren $i$-te Zeile gerade der in Gleichung (3.1) benutzte Gewichtsvektor des $i$-ten Neurons der versteckten Schicht ist. Analog ist die Gewichtsmatrix $\tilde{\omega}$ der Ausgabeschicht definiert. Damit berechnet sich

[3] Während die graphentheoretische Bezeichnung „Knoten“ bereits synonym zum Begriff „Neuronen“ innerhalb künstlicher neuronaler Netze verwendet wird, ist der ebenfalls aus der Graphentheorie stammende Begriff „Kanten“ noch ungebräuchlich.

die Ausgabe eines dreilagigen Feedforward Netzes gemäß:

$$y_i = \mathrm{g}\Big(\sum_{j=1}^{n_{hid}} \tilde{\omega}_{ij}\, \mathrm{g}\Big(\sum_{k=1}^{n_{in}} \omega_{jk}\, x_k - \Theta_j\Big) - \tilde{\Theta}_i\Big) \tag{3.2}$$

| | |
|---|---|
| $y_i$ | Ausgabewert des $i$-ten Ausgabeknotens |
| g | Transferfunktion |
| $n_{hid}$ | Zahl der Knoten der versteckten Schicht |
| $n_{in}$ | Zahl der Eingabeknoten. |

Werden die Gewichtsmatrizen durch eine nullte Spalte erweitert, in der die Schwellen $\Theta_i$ gemäß

$$\omega = \begin{pmatrix} \Theta_1 & \omega_{11} & \dots & \omega_{1n_{hid}} \\ \vdots & \vdots & \ddots & \vdots \\ \Theta_{n_{in}} & \omega_{n_{in}1} & \dots & \omega_{n_{in}n_{hid}} \end{pmatrix} \qquad \text{und} \qquad \tilde{\omega} = \begin{pmatrix} \tilde{\Theta}_1 & \tilde{\omega}_{11} & \dots & \tilde{\omega}_{1n_{hid}} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\Theta}_{n_{in}} & \tilde{\omega}_{n_{in}1} & \dots & \tilde{\omega}_{n_{in}n_{hid}} \end{pmatrix}$$

eingetragen werden und wird in Eingabe- und versteckter Schicht je ein „Biasneuron“ mit Index 0 und einem konstanten Ausgabewert von $-1$ eingeführt, so läßt sich Gleichung (3.2) eleganter schreiben zu

$$\vec{y} = \mathrm{F}(\vec{x}) = \mathrm{g}(\tilde{\omega} \cdot \mathrm{g}(\omega \cdot \vec{x})). \tag{3.3}$$

## 3.3.1 Leistungsfähigkeit der Feedforward Netze

### Modellieren allgemeiner Funktionen

Wie M. Minsky und S. Papert 1969 zeigten, kann ein Feedforward Netz ohne versteckte Schicht (Perzeptron) lediglich linear separierbare Probleme lösen [51]. Beispielsweise kann das logische EXKLUSIV ODER nicht durch ein Perzeptron realisiert werden. Wie in Abschnitt 3.3.2 veranschaulicht, wird diese Einschränkung durch Hinzunahme einer versteckten Schicht aufgehoben. Da es allerdings zu jener Zeit noch keine Möglichkeit gab, die Gewichte von Mehrschicht Perzeptrons an eine gegebene Problemstellung anzupassen, führte die Kritik von Minsky und Papert zu einen enormen Ansehensverlust der KNN.

Anhand eines allgemeinen Theorems von Kolmogorov [52] kann gezeigt werden, daß, sofern die Zahl der Knoten unbegrenzt ist, eine versteckte Schicht ausreichend ist, um beliebige stetige Funktion $\mathrm{R}^n \mapsto \mathrm{R}^m$ zu realisieren[4] [53][54][55][56].

### Optimales Klassifizieren

Sollen Repräsentanten überlappender Wahrscheinlichkeitsverteilungen klassifiziert werden, so läßt sich die Wahrscheinlichkeit einer Fehlklassifikation nicht beliebig reduzieren. Bilden $n$ Klassen $K_i$ eine paarweise disjunkte Zerlegung des Ereignisraumes, so

[4] Als Transferfunktion der Ausgabeknoten muß dazu die Identität verwendet werden.

ist die bedingte Wahrscheinlichkeit $P(K_i|x)$, daß bei Beobachtung von $x$ die Klasse $K_i$ vorliegt nach der Bayesschen Regel gegeben durch:

$$P(K_i|x) = \frac{P(K_i)\,P(x|K_i)}{\sum_{i=1}^{n} P(K_i)P(x|K_i)}\,. \tag{3.4}$$

Die Wahrscheinlichkeit einer Fehlklassifikation wird minimal, wenn Ereignis $x$ der Klasse $K_i$ mit der größten bedingten Wahrscheinlichkeit $P(K_i|x)$ zugewiesen wird (Bayesscher Klassifizierer). Da die Wahrscheinlichkeitsverteilung $P(x|K_i)$ der Zufallsvariable $x$ bei vorliegender Klasse $K_i$ im allgemeinen nicht bekannt ist, kann Gleichung (3.4) nicht direkt für die Klassifizierung benutzt werden.

Feedforward Netze leisten bei geeigneter Wahl der Gewichte eine optimale Approximation des Bayesschen Klassifizierers [57]. Diese Gewichte können durch eine als „überwachtes Lernen“ bezeichnete Prozedur gefunden werden, wobei sich die nötige Kenntnis der Wahrscheinlichkeitsverteilungen $P(x|K_i)$ auf einen Satz von Beispielpaaren $(x, K_i)$ beschränkt. Mit wachsender Zahl der während des „Lernens“ benutzten Beispielpaare erhöht sich die Güte der erreichbaren Approximation.

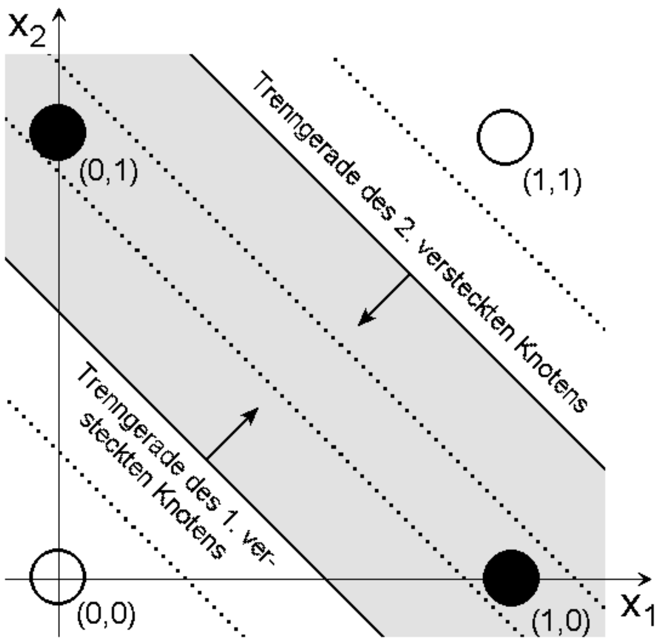
### 3.3.2 Geometrische Interpretation

Für Feedforward Netze mit einem Ausgabeknoten, die zwei Klassen separieren sollen, läßt sich eine einfache geometrische Interpretation der Netzentscheidung angeben. Jedes Neuron der versteckten Schicht definiert im Raum der Eingabevektoren durch:

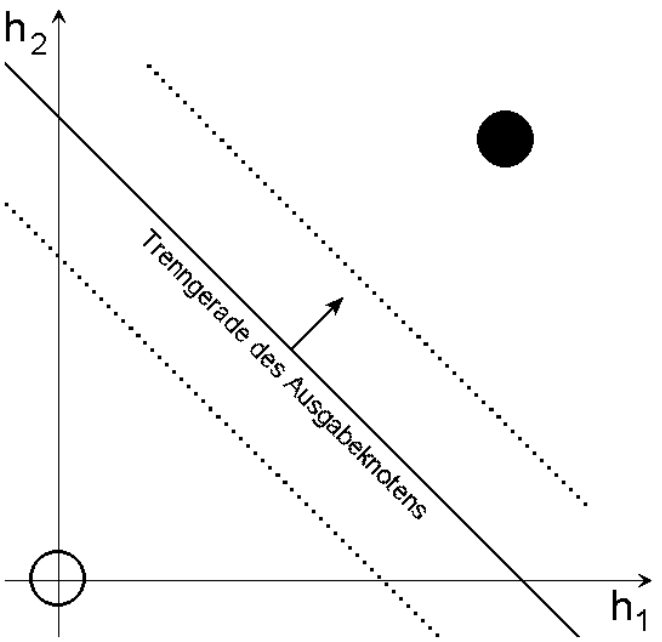
$$\vec{w} \cdot \vec{x} - \Theta = 0 \tag{3.5}$$

einen Unterraum, der den Eingaberaum in zwei disjunkte Teilräume zerlegt. Dies ist in Abbildung 3.3 für zwei Dimensionen am Beispiel des XOR–Problems dargestellt.

Ein einfaches Perzeptron realisiert eine Trenngerade und kann die Menge {(0,0);(1,1)} nicht von {(0,1);(1,0)} trennen. Ein dreischichtiges Perzeptron mit zwei versteckten Knoten, kann, wie dargestellt, zwei Geraden so legen, daß der Eingaberaum in einen „inneren“ und einen „äußeren“ Bereich aufgeteilt wird. Somit wird durch die Aktivitäten $h_1$ und $h_2$ der versteckten Knoten eine interne Darstellung des Problems verwirklicht, die linear separierbar ist und dem Ausgabeneuron die Trennung der Klassen ermöglicht. Die gestrichelten Linien stellen den Übergangsbereich der Transferfunktion dar. Die Ausdehnung des Übergangsbereiches wird durch $1/||\vec{\omega}||$ bestimmt.




(a) Trenngeraden im Eingaberaum




(b) Interne Darstellung der Eingabegrößen in der versteckten Schicht

**Abbildung 3.3:** Lösung des XOR-Problems durch ein dreilagiges Perzeptron

## 3.4 Der Backpropagation–Algorithmus

Aus Gleichung (3.3) folgt, daß der Ausgabevektor bei gegebenem Eingabevektor, eine Funktion der erweiterten Gewichtsmatrizen $\omega$ und $\tilde{\omega}$ („der Gewichte") ist. Als „Training" oder „überwachtes Lernen" wird ein Optimierungsprozeß bezeichnet, der die Gewichte des Netzes an die gestellte Aufgabe anpaßt. Dies geschieht auf Grundlage einer Menge von Paaren $(\vec{x}^\mu, \vec{t}^\mu)$ von Eingabe- und den gewünschten Ausgabevektoren, genannt Trainingsdaten oder Trainingsmuster. Die gewünschten Ausgabevektoren werden „Targetvektoren" genannt.

Im Backpropagation–Algorithmus (BPA), der 1986 von D. E. Rumelhardt und J. L. McClelland [58][59] vorgestellt wurde, wird der globale quadratische Fehler der Ausgabevektoren $\vec{y}^\mu$ in bezug auf die Targetvektoren $\vec{t}^\mu$

$$\mathrm{E}(\omega, \tilde{\omega}) = \sum_\mu ||\vec{y}^\mu - \vec{t}^\mu||^2, \tag{3.6}$$

durch ein in der numerischen Mathematik als Gradientenmethode bezeichnetes Verfahren minimiert. Dazu ist es erforderlich, daß die Transferfunktion differenzierbar ist. Man ersetzt daher die Heavyside–Stufenfunktion durch eine sigmoidale Funktion. Meist wird die Fermifunktion verwendet.

$$\mathrm{g}(a) = \frac{1}{1 + \exp(-\frac{a}{T})} \qquad \text{mit} \qquad \mathrm{g}(a) \in \left]0, 1\right[ \tag{3.7}$$

Diese Funktion enthält die Heavyside–Stufenfunktion als Grenzfall $T \to 0$. Größere Werte des Parameters $T$ ergeben einen „weicheren Verlauf der Stufe" und einen breiteren Übergangsbereich von kleinen zu großen Werten. Wegen

$$\frac{a}{T} = \frac{\vec{x} \cdot \vec{\omega}}{T} - \frac{\Theta}{T} = \vec{x} \cdot \vec{\omega}' - \Theta' = \frac{a'}{1} \tag{3.8}$$

kann der Parameter $T$ ohne Einschränkung zu eins gesetzt werden, die Breite des Übergangsbereiches der Sigmoidfunktion wird dann durch die Norm des Gewichtsvektors bestimmt.

Bei jeder Iteration des BPA werden die Gewichte der einzelnen Schichten, beginnend mit der Ausgabeschicht, proportional zum Gradienten des globalen Fehlers abgeändert. Dazu berechnet das Feedforward Netz für jedes Trainingsmuster $\vec{x}^\mu$ die Aktivitäten $h_j$ der versteckten Schicht und $y_i$ in der Ausgabeschicht. Die Änderung der Gewichte $\tilde{\omega}_{ij}$ errechnet sich dann zu

$$\Delta\tilde{\omega}_{ij} = -\eta \frac{\partial E}{\partial \tilde{\omega}_{ij}} = -\eta \sum_\mu \tilde{\delta}_i^\mu \, h_j^\mu \tag{3.9}$$

mit

$$\tilde{\delta}_i^\mu = (y_i^\mu - t_i^\mu) \cdot \mathrm{g}'(\vec{\tilde{\omega}}_i \, \vec{h}^\mu). \tag{3.10}$$

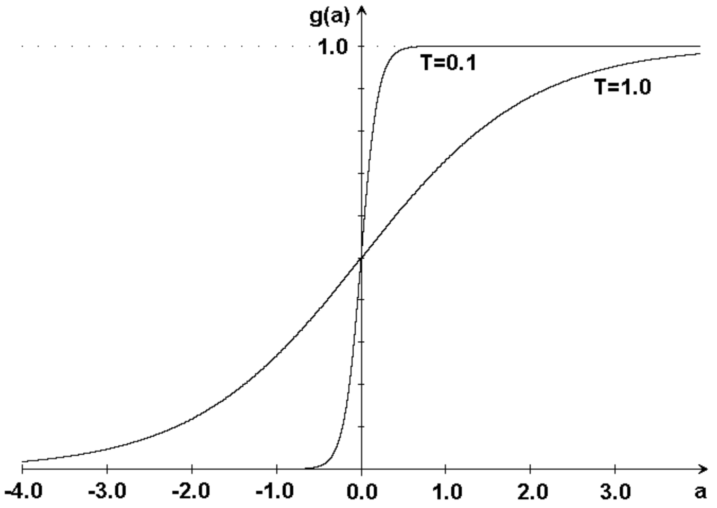



**Abbildung 3.4:** Sigmoidfunktion

Aufbauend auf den Berechnungen in der Ausgabeschicht, läßt sich der Fehler zur versteckten Schicht rückverfolgen und die Änderung der Gewichte $w_{jk}$ in analoger Weise berechnen

$$\Delta\omega_{jk} = -\eta \frac{\partial E}{\partial \omega_{jk}} = -\eta \sum_{\mu} \delta_j^{\mu} \; x_k \tag{3.11}$$

mit

$$\delta_j^{\mu} = \sum_{i=1}^{n_{in}} \tilde{\omega}_{ij} \; \tilde{\delta}_i^{\mu} \;\; \mathrm{g}'(\vec{\omega}_j \, \vec{x}^{\mu}) \tag{3.12}$$

Bei mehr als einer versteckten Schicht, wird der Fehler analog zu den Gleichungen (3.11) und (3.12) „rückwärts" von Schicht zu Schicht propagiert, daher der Name Error–Backpropagation. Der Lernparameter $\eta$, auch Lernrate genannt, bestimmt die Größe der Gewichtsänderungen, anschaulich die „Schrittweite des Suchprozesses im Parameterraum der Gewichte". Der Einfluß des Lernparameters, sowie weiterer Parameter, die in Modifikationen des BPA benutzt werden, wird in Abschnitt 6.3 untersucht.

### 3.4.1 Initialisierung der Gewichte

Zu Beginn des Trainings müssen die Gewichte initialisiert werden. Es ist nicht ausreichend, alle Gewichte auf identische Werte zu setzen, da der BPA nicht in der Lage ist, diese Symmetrie unter den Gewichten zu brechen. In der verwendeten Implementierung des BPA werden die Gewichte $\omega_{jk}$ pseudozufällig, aber reproduzierbar gemäß

$$\omega_{jk} \in [-\epsilon, \epsilon] \quad \text{mit} \quad \epsilon = \frac{3}{\sqrt{n_{in}}} \tag{3.13}$$

gewählt. Analog wird für die Gewichte $\tilde{\omega}_{ij}$

$$\epsilon = \frac{3}{\sqrt{n_{hid}}} \tag{3.14}$$

verwendet [60]. Nach Untersuchungen von L.F.A. Wessels und E. Barnard ist diese Wahl der Gewichte vorteilhaft, um ungünstige lokale Minima zu vermeiden [61].

Der benutzte BPA arbeitet völlig deterministisch, bei gegebenen Startgewichten, Traininsmustern und Trainingsparametern läßt sich ein durchgeführtes Training exakt reproduzieren. Die Auswahl der Startgewichte wird durch einen Parameter bestimmt, der den benutzen Zufallszahlengenerator initialisiert.

### 3.4.2 Modifikationen des BPA

Wie jede Gradientenmethode kann der BPA nicht garantieren, das globale Minimum der Fehlerfunktion zu finden. In der Praxis begnügt man sich daher, ein genügend „tiefes“ lokales Minimum zu erreichen. Um die Zahl der Trainingszyklen minimal zu halten und/oder die Gefahr des Steckenbleibens in einem ungünstig „hohen“ lokalen Minimum zu reduzieren, wurden verschiedene Modifikationen des Standard–BPA untersucht.

#### Lernintervall

Im Standard BPA, wie er in Abschnitt 3.4 beschrieben wurde, wird die Gewichtsänderung durchgeführt, nachdem alle Muster präsentiert wurden (batch learning), das Lernintervall ist in diesem Fall gleich der Zahl der Trainingsmuster. In den meisten Implementierungen des BPA wird eine Gewichtsänderung nach jedem einzelnen Muster durchgeführt entsprechend einem Lernintervall von eins (online learning)[5].

Als Trainingszyklus oder Epoche wird die Zahl der Iterationen des BPA bezeichnet, die benötigt werden, um alle Trainingsmuster zu bearbeiten. Im Falle des batch learning ist eine Epoche gleich einer Iteration, beim online learning ist die Zahl der Iterationen in einer Epoche gleich der Zahl der Trainingsmuster.

#### Momentumfaktor

Um Plateaus in der Fehlerlandschaft schneller zu überwinden, Oszillationen zu dämpfen und ein stabileres Lernverhalten zu erzielen, werden in einer weit verbreiteten Erweiterung des BPA die Gewichtsänderungen $\Delta\tilde{\omega}_{ij}^{alt}$ und $\Delta\omega_{jk}^{alt}$ der letzten Iteration mit berücksichtigt. Dieses Trainingsverfahren wird Momentum–BPA genannt und ergibt sich, wenn die Gleichungen (3.9) und (3.11) des Standard–BPA zu

$$\Delta\tilde{\omega}_{ij} = -\eta \frac{\partial E}{\partial \tilde{\omega}_{ij}} + \underbrace{\alpha\, \Delta\tilde{\omega}_{ij}^{alt}}_{Momentumterm} \tag{3.15}$$

[5] Der Begriff „online–learning“ kann im Rahmen des L2–Neuro–Triggers zu Verwechselungen mit einem Lernprozeß, der Netze während der Triggeranwendung optimiert führen. Ein solcher Lernprozeß ist nicht vorgesehen, da er die Reproduzierbarkeit des Netzverhaltens stark erschweren würde.

und

$$\Delta\omega_{jk} = -\eta \frac{\partial E}{\partial \omega_{jk}} + \underbrace{\alpha \, \Delta\omega_{jk}^{alt}}_{Momentumterm} \tag{3.16}$$

erweitert werden. Die Abklingkonstante $\alpha$ wird als Momentumfaktor bezeichnet und bestimmt den Einfluß der vergangenen Lernschritte auf den aktuellen. Der Momentumfaktor $\alpha$ muß ebenso wie der Lernparameter $\eta$ empirisch an das Problem angepaßt werden.

### Hidden Learning Scale

Im Standard–BPA wird für die Gewichtsänderungen $\Delta\tilde{\omega}_{ij}^{alt}$ in der Ausgabeschicht und $\Delta\omega_{jk}^{alt}$ in der Zwischenschicht derselbe Lernparameter $\eta$ verwendet. Als Erweiterung des Standard–BPA wird die Lernrate $\eta'$ in der Zwischenschicht größer gewählt als die Lernrate $\eta$ in der Ausgabeschicht[6]. Dieses Vorgehen wurde entwickelt, um das Trainingsverhalten von Feedforward Netzen zu verbessern, die wenige Ausgabeknoten besitzen. Die zugrundeliegende Idee wird aus Gleichung (3.12) deutlich. Die gewichtete Summe über die rückvermittelten Fehler der Ausgabeschicht, erstreckt sich über alle Ausgabeknoten $i$. Da die von einem Trainingsmuster $\mu$ hervorgerufene Gewichtsänderung in der versteckten Schicht proportional zu dieser Summe ist, ergeben sich bei geringer Zahl der Ausgabeknoten $n_{out}$, kleinere Gewichtsänderungen relativ zur Ausgabeschicht als im Falle vieler Ausgabeknoten [60].

Dies wird durch den Proportionalitätsfaktor „Hidden Learning Scale“ (HLS) ausgeglichen, der im weiteren mit $\beta$ bezeichnet wird:

$$\eta' = \beta \, \eta \, . \tag{3.17}$$

## 3.5 Feedforward Netze als Trigger

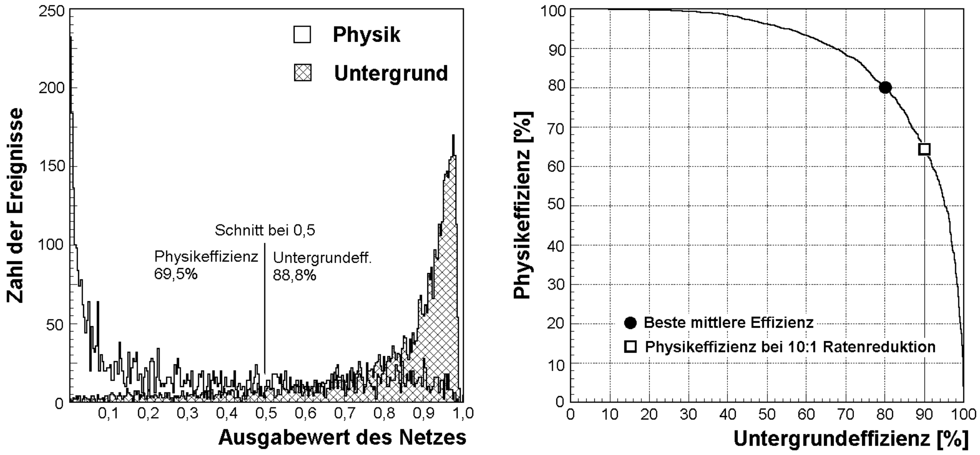
Da ein Trigger eine Ja/Nein Entscheidung trifft, benötigt ein als Trigger eingesetztes Feedforward Netz nur einen Ausgabeknoten. Um einen boolschen Ausgabewert zu erhalten, wird die Netzausgabe mit einer als „Schnitt“ bezeichneten Zahl verglichen.

Das Prinzip der Trennung wird in Abbildung 3.5(a) verdeutlicht. Es ist die Verteilung der Ausgabewerte eines Netzes dargestellt, das trainiert wurde, zwei Klassen zu unterscheiden. Die eine Klasse, bei der es sich um simulierte ep–Kollisionen handelt („Physikereignisse“) soll vom Netz mit einem Ausgabewert von null beantwortet werden, die andere Klasse (simulierte Untergrundereignisse) mit einem Ausgabewert von eins. Die Verteilung zeigt, daß die Aufgabe im wesentlichen erfüllt wird. Setzt man den Schnitt auf den mittleren Wert von 0,5, so werden 69,5% der Physik- und 88,8% der Untergrundereignisse korrekt klassifiziert.

[6] Wenn von „der Lernrate“ gesprochen wird, ist immer die Lernrate in der Ausgabeschicht gemeint, daher wurde die Nomenklatur $\eta, \eta'$ anstatt $\tilde{\eta}, \eta$ gewählt.

Die Anteile der korrekt klassifizierten Ereignisse, im weiteren Physik- bzw. Untergrundeffizienz ($\varepsilon_P$ bzw. $\varepsilon_U$) genannt, können durch Variation des Schnittes verändert werden. Die Abhängigkeit beider Effizienzen ist in Abbildung 3.5(b) als sogenannte Effizienzkurve dargestellt. Aus der Untergrundeffizienz $\varepsilon_U$ ergibt sich der Reduktionsfaktor der Untergrundrate $\mathcal{R}$ zu

$$\mathcal{R} = \frac{1}{1-\varepsilon_U}. \tag{3.18}$$




(a) Verteilung der Ausgabewerte eines trainierten Feedforward Netzes bezüglich der Testmuster

(b) Physikeffizienz als Funktion der Untergrundeffizienz

**Abbildung 3.5:** Veranschaulichung der Klassifizierung

### 3.5.1 Bewertungskriterien

Der BPA bewertet den Erfolg eines Netzes durch den quadratischen Fehler auf den Trainingsdaten. Da dieser von der Zahl der Traingsmuster $M$ abhängt, ist er für den Vergleich zweier Netze ungeeignet. Besser eignet sich daher der *mittlere* quadratische Fehler[7]

$$MSE_{Train} = \frac{1}{M}\sum_{\mu=1}^{M} ||\vec{y}^{\mu} - \vec{t}^{\mu}||^2. \tag{3.19}$$

Die Leistungsfähigkeit eines Triggers läßt sich durch die Effizienzkurve beurteilen. In dieser Arbeit werden zwei Bewertungskriterien benutzt, die sich aus der Effizienzkurve ableiten.

- **Beste mittlere Effizienz** *BME*
  Dies ist der maximale Mittelwert aus Physik- und Untergrundeffizienz, wie er

[7] mean squared error

sich bei geeigneter Wahl des Schnittes ergibt.

$$BME = \max\Big(\frac{\varepsilon_P + \varepsilon_U}{2}\Big) \tag{3.20}$$

Der statistische Fehler der besten mittleren Effizienz, wird durch die Zahl der als Testdaten verwendeten Ereignisse $N$ bestimmt. Er läßt sich unter der Annahme, daß das Netz jedes Ereignis in den Testdaten mit einer festen Wahrscheinlichkeit p=$BME$ korrekt klassifiziert, als statistischer Fehler einer Binomialverteilung zu

$$\sigma(BME) = \sqrt{\frac{\mathrm{p}(1-\mathrm{p})}{N-1}} \tag{3.21}$$

abschätzen.

- **Physikeffizienz bei geforderter Reduktion der Untergrundrate**
  Nach Gleichung (3.18) und Abbildung 3.5(b) läßt sich jedem Reduktionsfaktor der Untergrundrate eine Physikeffizienz zuordnen. Anhand der erreichten Physikeffizienzen bei gleichem Reduktionsfaktor können die Effizienzen verschiedener Netze beurteilt werden.

  Liegt die geforderte Untergrundeffizienz nahe eins (große Reduktion der Untergrundrate), muß der Verlauf der Effizienzkurve (Physik- gegen Untergrundeffizienz) beachtet werden, um den statistischen Fehler der Physikeffizienz korrekt abzuschätzen. In vorliegender Arbeit wird folgendes Verfahren benutzt:
  Die Physikeffizienz wird als Funktion der Untergrundeffizienz $\varepsilon_P = \mathrm{F}(\varepsilon_U)$ betrachtet und die Sekantensteigung $\Delta\mathrm{F}(\varepsilon_U)$ bei der geforderten Untergrundeffizienz $\varepsilon_U$ ermittelt. Der statistischen Fehler der Physikeffizienz wird gemäß

$$\sigma(\varepsilon_P) = \sigma(\varepsilon_U) \cdot \Delta\mathrm{F}(\varepsilon_U) \tag{3.22}$$

  abgeschätzt, wobei $\sigma(\varepsilon_U)$ nach Gleichung (3.21) berechnet wird. Diese Gleichung eignet sich ebenfalls zur Bestimmung von $\sigma(BME)$ und liefert im Grenzfall, in dem F stetig und differenzierbar ist, identische Resultate wie Gleichung (3.21).

Entscheidend ist allerdings nicht die Effizienz des Netzes gegenüber den Trainingsmustern, sondern in der Anwendung, wenn im allgemeinen Muster klassifiziert werden müssen, die nicht im Training präsentiert wurden. Daher werden die Netze nicht anhand des Fehlermaßes $MSE_{Train}$, bzw. der Effizienz $BME_{Train}$ bezüglich der Trainingsmuster bewertet, sondern bezüglich einer Menge von Mustern, die nicht für das Training verwendet wurden, den sogenannten Testmustern.

## 3.6 Implementierung der Netze auf der zweiten Triggerstufe

### 3.6.1 Hardware

Implementiert sind die KNN auf CNAPS/VME-Karten[8], auf denen sich 64 bis 512 einzelne Prozessoren, genannt PN (Processor Node) befinden. Jeder PN verfügt über 4 KByte lokalen Speicher und kann Berechnungen in Festkomma–Arithmetik durchführen. Je nach Bauart sind 16 bzw. 64 PNs in einem Chip integriert [62]. Alle Prozessoren erhalten über einen gemeinsamen Bus identische Instruktionen, werden jedoch mit unterschiedlichen Daten initialisiert, (Single Instruction Multiple Data, SIMD). Die PNs werden mit 20,8 MHz getaktet, der doppelten BC–Frequenz. Die Architektur des CNAPS erlaubt einen Multiplikations–Addier Schritt einer Vektor Martix Multiplikation in einem Taktzyklus. Somit steigt die Rechenzeit linear statt quadratisch mit der Zahl der Knoten des KNN. Die Transferfunktion ist tabelliert und muß daher nicht berechnet werden. Ein Feedforward-Schritt eines $64 \times 64 \times 1$ Netzes kann in 166 Taktzyklen bzw. 8,0 $\mu$s berechnet werden [63]. Dies bedeutet gegenüber sequentiellen Prozessoren moderner Bauart eine Beschleunigung um 2–3 Größenordnungen.

Der Datentransfer zu und von den KNN, sowie eine einfache Vorverarbeitung der Daten wird durch sogenannte „Data Distribution Boards“ DDBs erreicht. Mögliche Vorverarbeitungen sind u.a. das Bilden von Summen oder Bitsummen sowie beliebige Abbildungen Byte $\mapsto$ Byte durch Tabellen. Die maximale Vorverarbeitungszeit beträgt 8 $\mu$s. Die Kombination aus je einer CNAPS-Karte und einem DDB realisiert ein komplettes KNN mit Datenzulieferung, Berechnung und Weiterleitung des Ergebnisses.

Die Steuerung und Kontrolle erfolgt über eine Workstation vom Typ VMESun SPARC 10, die die CNAPS-Karten und DDBs zugreifen kann und über eine als „Spy“ bezeichnete Auslesekarte die direkte Überwachung des Datenflusses ermöglicht.

### 3.6.2 Steuerungs- und Überwachungsprogramme

Die verschiedenen Steuerungs- und Überwachungsaufgaben, wie

- das Schreiben der Gewichte in den lokalen Speicher der PNs,
- das Speichern der benutzen Einstellung in der H1-Datenbank,
- die Überwachung des Datenstroms durch Histogramme einzelner Detektordaten,

sind in einzelne Programme aufgegliedert, die als unabhängige Prozesse gleichzeitig betrieben werden (Multitasking). Die Koordination wird von einem Prozessmanager übernommen, der zusätzlich als Server dient, um via Internet eine Fernüberwachung und -steuerung zu ermöglichen. Die Interprozess-Kommunikation erfolgt über „shared memory“, die Client-Server Kommunikation nach dem TCP/IP Protokoll [64].

Als Beitrag zu diesem Projekt, wurde das Programm zum Erstellen der Überwachungshistogramme beigesteuert. Dabei wurden am CERN entwickelte FORTRAN

[8] CNAPS steht für Connected Network of Adaptive ProcessorS

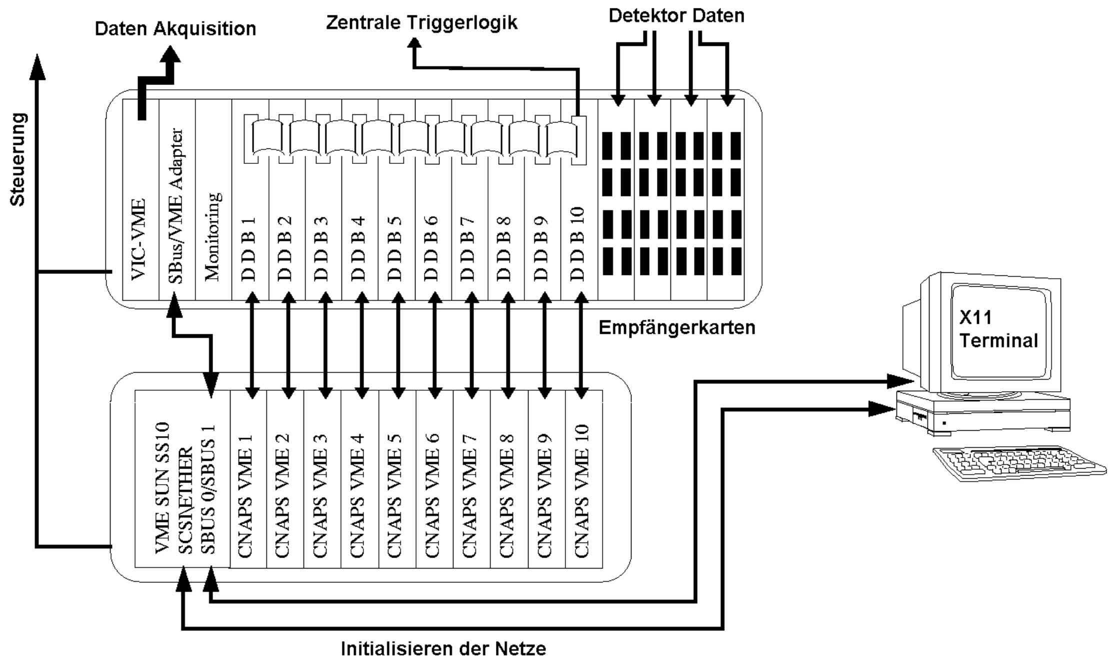



**Abbildung 3.6:** Hardware des L2–Neuro–Triggers

Routinen zum Erstellen, Verwalten und graphischem Darstellen von Histogrammen benutzt, was eine Einarbeitung in diese Programmpakete und die Verbindung von C und FORTRAN Routinen nötig machte.

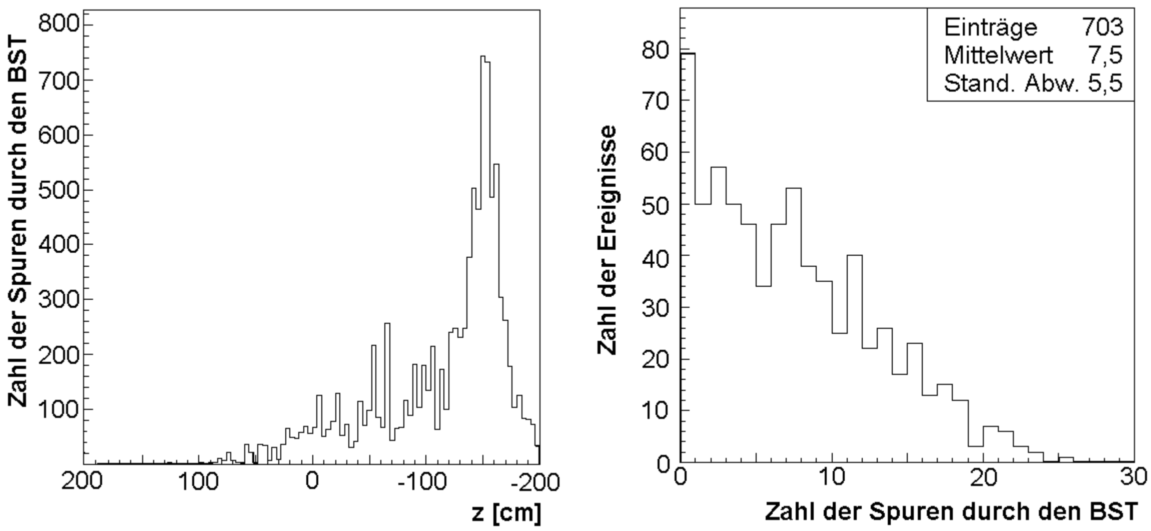
# 4. Der BST-T als L2–Untergrundfilter

In vorliegender Arbeit wird die Leistungsfähigkeit eines auf KNN basierenden Untergrundfilters (Veto-Trigger), der die Daten des BST-T nutzt, diskutiert.

## 4.1 Motivation

Der BST befindet sich aus Sicht der Protonen 80–61 cm vor der Wechselwirkungszone. Dies bedeutet, daß die Mehrheit der Teilchen des protoneninduzierten Untergrundes den BST in entgegengesetzter Richtung der Reaktionsprodukte der ep-Kollisionen durchqueren.

Untergrund, der einige Meter vor dem Detektor entsteht, läßt sich durch das Flugzeitmeßsystem totzeitfrei auf der ersten Triggerstufe unterdrücken. Untergrundereignisse, deren Vertizes sich zu dicht am nWWP befinden, können von L1 nicht mehr anhand der Flugzeitmessung als Untergrund erkannt werden. Abbildung 4.1(a) zeigt eine Verteilung der rekonstruierten $z$-Komponente der Vertizes von Untergrundereignissen des Betriebsjahres 1995, die nicht von der ersten Triggerstufe erkannt wurden.




(a) Rekonstruierte $z$-Verteilung der Spuren (b) Verteilung der Spurmultiplizität

**Abbildung 4.1:** Verteilung der rekonstruierten Vertizes und Spurmultiplizitäten für Untergrundereignisse des Betriebsjahres 1995 mit Spuren durch den BST (L4-Reject)

Bei der Auswahl der Ereignisse wurde gefordert, daß mindestens eine Spur durch den BST verläuft. Alle Ereignisse wurden erst auf der vierten Triggerstufe zurückgewiesen (L4-Reject)[1]. Man erkennt ein ausgeprägtes Maximum bei $z \approx -1,5$ m.

[1] Es handelt sich dabei um Monitor-Ereignisse, die trotz eines L4-Reject auf Band geschrieben

Der BST liegt also zwischen der Wechselwirkungszone der ep-Kollisionen und der Entstehungsszone dieses schwer fassbaren Untergrundes. Es ist daher zu erwarten, daß die Daten des BST-T besonders für die Unterscheidung dieses Untergrundes von Physikereignissen geeignet sind.

## 4.2 Limitierungen

Die Bereitstellung der Daten des BST-T in voller Granularität für höhere Triggerstufen wäre zwar physikalisch sinnvoll, ist jedoch momentan nicht realisiert, da die geringen freien Volumina einfache technische Lösungen unmöglich machen [65]. Daher wird direkt am Detektor eine 4:1 Verdichtung der Daten vorgenommen. Das L1–Konzept sieht vor, das Padmuster jedes Sektors durch feldprogrammierbare Gatter XILINX 3000 mit einem Satz von maximal 63 Masken zu verglichen und die Nummer der ersten „passenden“ Maske in eine 6-Bit Zahl zu kodieren. Zusätzlich wird in einer vorher festgelegten Scheibe (z.B.: der zweiten) für jeden Sektor die Zahl der getroffenen Pads gezählt. Diese Trefferzahl wird in weiteren 2 Bits kodiert, so daß die 32 Bit eines Padmusters zu 8 Bits verdichtet werden. Die Verarbeitungsgeschwindigkeit der XILINX erlaubt einen Vergleich von etwa 100 Masken [14].

## 4.3 Konzeption

Von den KNN wird verlangt, daß sie diesen Untergrund zu 99% korrekt klassifizieren, so daß die Rate bereits auf der zweiten Triggerstufe auf ein Hundertstel reduziert und somit Totzeit gespart werden kann. Untersucht wurde, welche Physikeffizienz sich bei der geforderten Ratenreduktion erreichen läßt.

Neben der Nutzung der Maskennummer und Trefferzahl wurde auch die direkte Nutzung der Padmuster untersucht, wobei verschiedene Möglichkeiten der notwendigen 4:1 Verdichtung betrachtet wurden. Darüber hinaus wurde die Idee untersucht, die vorhandene L1–Hardware besser für die Untergrunderkennung zu nutzen, und zusätzlich zu den „Physik“-Masken auch Masken zu verwenden, die speziell auf Spuren von Untergrundereignissen ausgelegt sind.

werden, um die Funktionsweise des Triggers und die Untergrundbedingungen überwachen zu können.

# 5. Simulation des BST-T

Da geeignete Daten vom BST-T noch nicht verfügbar waren, wurden die Daten, anhand derer die Gewichte der Netze optimiert werden (Trainingsdaten), durch Monte–Carlo Simulation erzeugt. Grundlage dafür ist das in dieser Arbeit erstellte Programm `bstsim`. In diesem Kapitel wird die Funktionsweise der Simulation erläutert, die Wahl der Parameter für die Erzeugung der Trainings- und Testdaten wird zusammen mit den Trainingsergebnissen in Kapitel 6 und 7 diskutiert.

Ein weiterer Zweck der Simulation ist die Analyse der Masken. Dabei wurde die Zahl aller Masken und das geometrische Auflösungsvermögen des BST-T bestimmt. Für die Simulation der Maskierung auf der ersten Triggerstufe, wurde die Häufigkeit des Auftretens der Masken bestimmt.

## 5.1 Beschreibung

Das Programm schneidet gerade Linien, die von einem Punkt $(0, 0, z)$ entlang der $z$-Achse des H1–Koordinatensystems ausgehen mit mindestens einer Scheibe des BST. Das binäre BST-T Signal für den Durchgang eines Teilchens wird simuliert, indem die Durchstoßpunkte berechnet und die getroffenen Pads bestimmt werden. Jedes getroffene Pads wird mit einer Wahrscheinlichkeit $\mathcal{E}_{intrisic}$ auf den Wert 1 gesetzt (intrinsische Detekoreffizienz).

Die Näherung der Teilchenspuren durch gerade Linien ist gerechtfertigt, da die Transversalimpulse $P_\perp$ der untersuchten Teilchen klein gegenüber ihrem Gesamtimpuls $P$ sind. Die Ablenkung durch das Feld des Solenoidmagneten ist proportional zu $P_\perp/P$ und kann gegenüber der durch die Größe der Pads bestimmten radialen Auflösung vernachlässigt werden. Der Einfluß elektronischen Rauschens wird simuliert, indem auch die nicht getroffenen Pads, mit der Wahrscheinlichkeit $\mathcal{N}$ auf 1 gesetzt wird.

In der Simulation sind sowohl $\mathcal{E}_{intrisic}$ als auch $\mathcal{N}$ für alle Pads gleich. Eventuell in der Praxis auftretenden Abhängigkeiten dieser Größen von $r$ oder $z$ (z.B.: verstärktes Rauschen nahe der Strahlachse) führen demgegenüber zu einem geringeren Informationsverlust, so daß hier der ungünstigste Fall betrachtet wird.

## 5.2 Trainings- und Testdaten

Für die Erzeugung der Trainings- und Testdaten wie sie in Kapitel 7 benutzt werden, wurde $z$ zufällig gemäß einer Gauß–Verteilung mit Mittelwert $\mu_P$ und Standardabweichung $\sigma_P$ für simulierte ep–Kollisionen (Physikereignisse) und $\mu_U$ und $\sigma_U$ für Untergrundereignisse gewählt.

Die Polarwinkel der Spuren wurden zufällig aus dem Bereich $[\Theta_{\min(z,2)}$–$\Theta_{\max(z,2)}]$ für Physikereignisse bzw. $[\Theta_{\min(z,3)}$–$\Theta_{\max(z,3)}]$ für Untergrundereignisse gewählt. $\Theta_{\min(z,3)}$ ist der kleinste Winkel, unter dem eine Spur mit Vertex $(0, 0, z)$ mindestens drei Scheiben des BST-T schneidet. Für die Untergrundereignisse wurden mindestens drei getroffene

Scheiben gefordert, denn der Versuch, ein Veto auf Untergrund mit nur zwei getroffenen Scheiben auszusprechen, erhöht die Wahrscheinlichkeit erheblich, Ereignisse ohne Spur im BST-T, aber mit durch Rauschen vorgetäuschten, spurähnlichen Mustern zu verwerfen. Die Zahl der Spuren durch den BST-T, genannt Spurmultiplizität, wurde zufällig gemäß einer Poisson-Verteilung mit Parameter $\lambda_P$ für Physik- und $\lambda_U$ für Untergrundereignisse, gewählt. Somit sind innerhalb einer Ereignisklasse (Physik, Untergrund) lediglich $\Theta$ und $z$ korreliert, gemäß der Forderung, den BST-T zu treffen. Auf die Simulation weiterer Korrelationen wurde verzichtet, um dem KNN möglichst keine weiteren Klassifikationsmerkmale als $z$-Vertex und die Spurmultiplizität zu bieten. Durch Verwendung von Trainingsdaten mit unterschiedlichen Verteilungen der Spurmultiplizität $\lambda_P \neq \lambda_U$ einerseits und identischer Spurmultiplizitätsverteilung $\lambda_P = \lambda_U$ andererseits, wurde der Einfluß der Spurmultiplizitätsverteilung auf die Klassifikationsfähigkeit untersucht (Abschnitt 7.8).

## 5.3 Geometrisches Auflösungsvermögen des BST-T

Um ein besseres Verständnis der möglichen Muster zu erlangen, wurden zunächst per Computer alle Spurmuster innerhalb eines Sektors generiert, die bei gegebener Geometrie des BST-T auftreten können. Dazu wurde $z$ im Bereich von $-75$ m bis 75 m mit einer Schrittweite von 0,2 mm variiert und für jeden Wert von $z$ der Polarwinkel $\Theta$ zwischen $\Theta_{\text{min}}(z, 2)$ und $\Theta_{\text{max}}(z, 2)$ mit einer Schrittweite von 10 Bogensekunden variiert. Dabei bezeichnet $\Theta_{\text{min}}(z, 2)$ den kleinsten Winkel, unter dem eine geradlinige Spur mit Ursprung in $z$ mindestens zwei Scheiben des BST trifft. Auf diese Weise wurde die Zahl der möglichen Spurmuster innerhalb eines Sektors zu 415 bestimmt. Fordert man mindestens drei getroffene Scheiben, so erhält man 297 mögliche Muster bzw. Masken.

Jeder Maske wird durch alle Spuren, die diese Maske generieren eine Verteilung in $z$ und durch deren Standardabweichung ein Auflösungsvermögen zugeordnet. Das geometrische Auflösungsvermögen $\sigma_z(z)$ des BST-T ergibt sich als gewichtetes Mittel der Standardabweichungen aller Masken, die durch Spuren mit Ursprung in $z$ generiert werden können. Dabei ist mit der Häufigkeit des Auftretens der einzelnen Masken zu wichten. Die drastische Verschlechterung des Auflösungsvermögen außerhalb des Intervalls $z \in [-176, 35]$ cm wird durch das Auftreten der Masken der Klasse $(0, +1, 0)$ bzw. $(0, -1, 0)$ verursacht, deren Standardabweichung bei etwa 8 m liegt. Innerhalb dieses Intervalles treten ausschließlich Masken der Form

$$(a, b, c, d) \quad \text{mit} \quad a, b, c, d \in \{0; 1; 2; 3; 4; 5; 6; 7; 8\} \quad \text{und} \quad |a - d| \geq 2$$

auf, bei denen sich Ein- und Austrittsring um mindestens zwei unterscheiden. Die $z$-Verteilung dieser Masken besitzen Standardabweichungen kleiner 30 cm. Masken einer Klasse besitzen bezüglich der Vertizes der Spuren ähnliche Verteilungen in $z$.

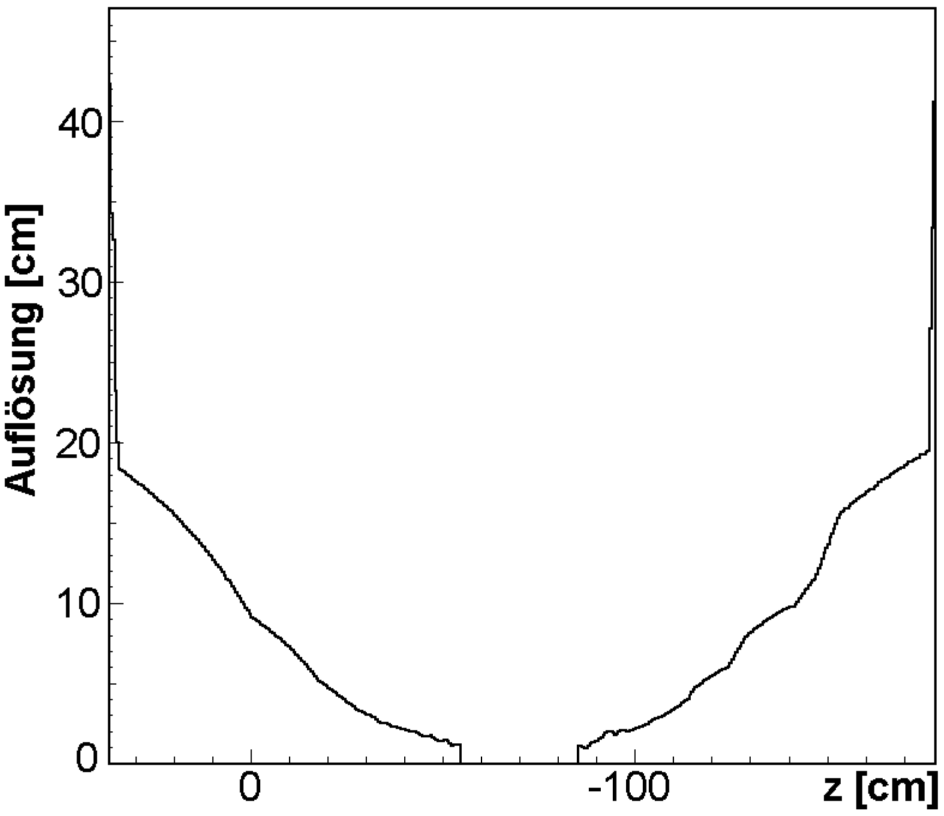



**Abbildung 5.1:** Geometrisches Auflösungsvermögen des BST-T

## 5.4 Triggermasken

Geeignete Masken, um Physik- bzw. Untergrundereignisse zu erkennen, wurden ebenfalls mit hilfe der Simulation bestimmt. Dazu wurden je $10^7$ Spuren durch einen Sektor des BST-T generiert, die mindestens drei Scheiben durchdringen. Analog zur Erzeugung der Trainingsdaten wurde $z$ zufällig gemäß einer Gauß–Verteilung mit $\mu_P$ und $\sigma_P$ bzw. $\mu_U$ und $\sigma_U$ und der Polarwinkel gemäß einer Gleichverteilung über $[\Theta_{\min(z,3)}-\Theta_{\max(z,3)}]$ gewählt. Indem die Häufigkeit für das Auftreten der einzelnen Masken bestimmt wird, läßt sich die Sequenz der Masken bestimmen, die für eine gegebene $z$-Verteilung benötigt werden.

# 6. Untersuchungen zum Trainingsverhalten

Um leistungsfähige Feedforward Netze zu erhalten, müssen die Gewichte für die jeweilige Problemstellung optimiert werden. In diesem Kapitel wird das Training der Netze durch den Backpropagation-Algorithmus (BPA) untersucht.

Da die mittels BPA erzielten Ergebnisse von den gewählten Trainingsparametern abhängen, wurde der Einfluß der einzelnen Parameter untersucht. Das Training wurde auf dem CNAPS-Parallelcomputer durchgeführt, was aufgrund der enormen Rechenleistung umfangreiche Parameterstudien des Backpropagation-Algoritmus' ermöglichte.

Das Ziel dieser Untersuchung ist es, die Zahl der Trainingsparameter die variiert werden müssen, um optimale Ergebnisse zu erlangen einzuschränken. Wenn möglich soll ein Parametersatz gefunden werden, der unabhängig von der jeweiligen Problemstellung zu guten Ergebnissen führt. Dies ist für umfangreiche Vergleiche der Leistungsfähigkeit von Feedforward Netzen für verschiedene Problemstellungen, wie sie im nächsten Kapitel durchgeführten werden erforderlich, da der sich Aufwand, alle Parameter der BPA–Modifikationen für jede Trainingsaufgabe neu zu optimieren, verbietet.

Um die Trainingsdauer der Parameterstudie zu reduzieren, wurde eine leicht vereinfachte Problemstellung untersucht, bei der die Netze lernen sollen, Muster mit einer Spur durch einen Sektor des BST-T zu erkennen.

## 6.1 Trainingsdaten

Als Netzeingabe wurden Muster in der Auflösung $4 \times 8$ benutzt, wie sie von den Pads eines Sektors des BST-T geliefert werden. Der Trainingssatz umfaßt alle möglichen Muster mit genau einer Spur, die mindestens zwei Scheiben des BST-T durchdringt. Dies sind alle 415 Masken mit mindestens zwei Treffern. Es wurden verschiedene geforderte Netzausgaben untersucht.

- $z$-Bestimmung
  In dieser Problemstellung war die geforderte Netzausgabe der Mittelwert in $z$ all jener Spuren, die die entsprechende Maske generieren ($\bar{z}$), transformiert auf den Wertebereich der Netzausgabe $y \in [0, 1]$.
- Rechts-Links Trennung
  Hier wurde die Netzantwort $y = 0$ gefordert, falls $\bar{z}$ größer als die Mitte des BST ist, und $y = 1$ sonst.
- Vertex-Nonvertex Trennung
  Diese Problemstellung fordert die Netzantwort $y = 0$, falls $\bar{z}$ weniger als 50 cm vom nWWP entfernt ist, $\bar{z} \in [-500\ \text{mm}, 500\ \text{mm}]$ und $y = 1$ sonst.

Als alternative Problemstellung unter erschwerten Bedingungen wurde jeweils statt der Maske ein Satz von 157 verrauschten Mustern präsentiert, insgesamt also $415 \cdot 157 = 65155$ Trainingsmuster. Dazu wurde jedes Pad mit der Wahrscheinlichkeit $\mathcal{N}$ auf eins

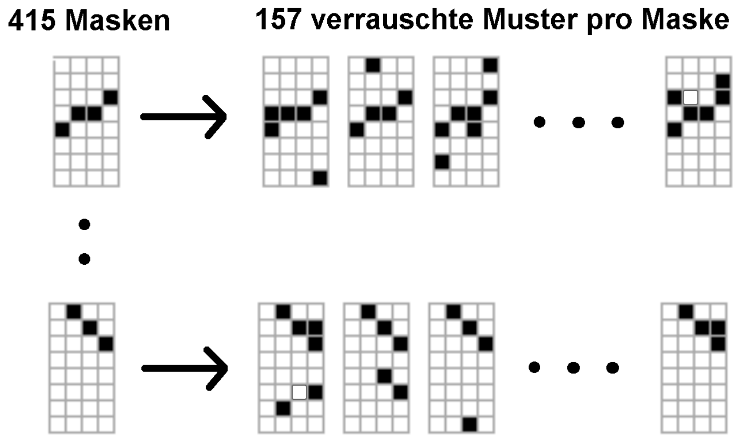



**Abbildung 6.1:** Masken und verrauschte Muster

gesetzt. Der Sachverhalt wird in Abbildung 6.1 verdeutlicht. Der gesamte Satz von Trainingsdaten wurde durchmischt.

Es wurden $n_{in} = 32$ Eingabeknoten benutzt, so daß für jede Eingabegröße $x_k \in \{0, 1\}$ gilt. Die betrachteten Netze besitzen eine versteckten Lage. Die Zahl der versteckten Knoten $n_{hid}$ war Gegenstand der Parameterstudie und wurde daher variiert. Die Zahl der Ausgabeknoten $n_{out}$ ist eins.

## 6.2 Training auf dem CNAPS Parallelrechner

Alternativ zu herkömmlichen Workstations, kann das Training auch auf dem CNAPS–Parallelrechner durchgeführt werden. Dies führt zu einer drastischen Beschleunigung des Trainings, wodurch sehr umfangreiche Studien des Einflusses der Trainingsparameter möglich sind. Im Rahmen dieser Arbeit wurden alle Netze auf dem CNAPS trainiert. Das Trainingsprogramm (`bpexp2` [66]), das die Steuerung des Trainings übernimmt und auf einer herkömmlichen Workstation betrieben wird, wurde dabei weiterentwickelt. Die benutze CNAPS–Karte ist mit 256 PNs bestückt und greift auf 16 MByte externen Hauptspeichers zu. CNAPS kann in zwei Softwaremodi betrieben werden:

- Der 8/16-Bit Modus (low precision)
  Die Feedforward–Schritte der Triggeranwendung werden in diesem Modus berechnet. Es werden 8 Bit Daten und 16 Bit Gewichte verwendet. Da der dynamische Bereich der Triggerdaten ebenfalls 8 Bit beträgt, ist die Genauigkeit für die Berechnung des Feedforward Schrittes ausreichend. Die Gewichtsänderungen während des Trainings $\Delta\tilde{\omega}_{ij}$ und $\Delta\omega_{jk}$ werden mit 16 Bit Genauigkeit berechnet.
- Der 16/32-Bit Modus (high precision)
  Dieser Modus benutzt 16 Bit Daten, 16 Bit Gewichte und Gewichtsänderungen mit 32 Bit Genauigkeit. In Tests wurde festgestellt, daß das Training in diesem Modus um den Faktor $2,0 \pm 0,1$ verlangsamt ist, in der Beschreibung der Herstellerfirma [60] wird der Faktor vier angegeben. Trotz dieser Verlangsamung

eignet sich der 16/32-Bit Modus erheblich besser für das Training, was durch die Fähigkeit, kleine Gewichtsänderungen durchführen zu können erklärt wird [66].

Für das Training der Netze wurde der 16/32-Bit Modus gewählt.

## 6.3 Parameterstudien

Verschiedene Parameter des Trainingsprogrammes wurden variiert, um zu möglichst günstigen Parameterkombinationen zu gelangen.

### 6.3.1 Lernintervall

Netze mit 63 versteckten Knoten ($32 \times 63 \times 1$) wurden auf die Vertex-Nonvertex Trennung bei 20% Rauschen trainiert. Durch den hohen Rauschanteil von 20%, ist diese Problemstellung schwierig zu lösen, was den Vorteil deutlicher Unterschiede im Trainingserfolg bietet. Die Verwendung vergleichsweise vieler Knoten in der versteckten Lage soll sicherstellen, daß die Netze bei Verwendung optimaler Gewichte in der Lage sind, die Problemstellung zu lösen.

Das Training wurde mit den Lernintervallen 65155 („offline learning"), 157, 10 und 1 („online learning") durchgeführt. In Abbildung 6.2 ist die mittlere Effizienz bei einem Schnitt von 0,5 dargestellt. Die „offline" trainierten Netze zeigen keinerlei Lernerfolg (eine mittlere Effizienz von 50% wird durch blindes Raten erreicht), die mit kleinerem Lernintervall trainierten Netze erreichen Effizienzen über 75%. Da das beste Netz mit einem Lernintervall von eins trainiert wurde und die Effizienzen der ohne Momentum trainierten Netze mit sinkender Größe des Lernintervalls steigen, wurde im weiteren ein Lernintervall von eins benutzt.

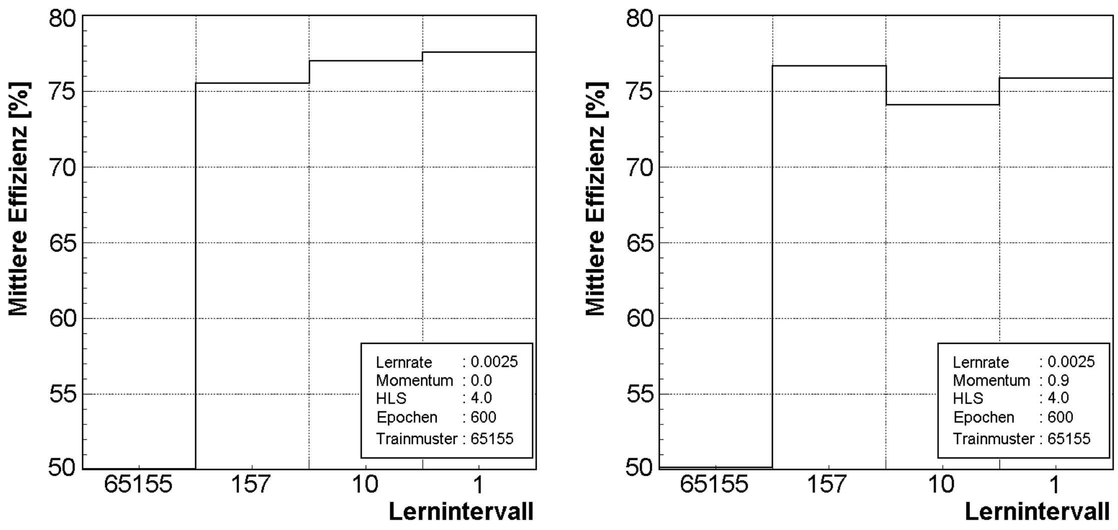



**Abbildung 6.2:** Einfluß des Lernintervalls auf die erreichte Effizienz

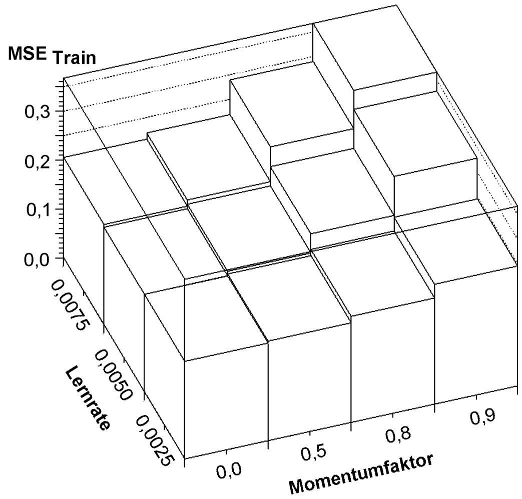



**Abbildung 6.3:** Einfluß des Momentumtermes bei verschiedenen Lernraten

### 6.3.2 Lernrate und Momentumfaktor

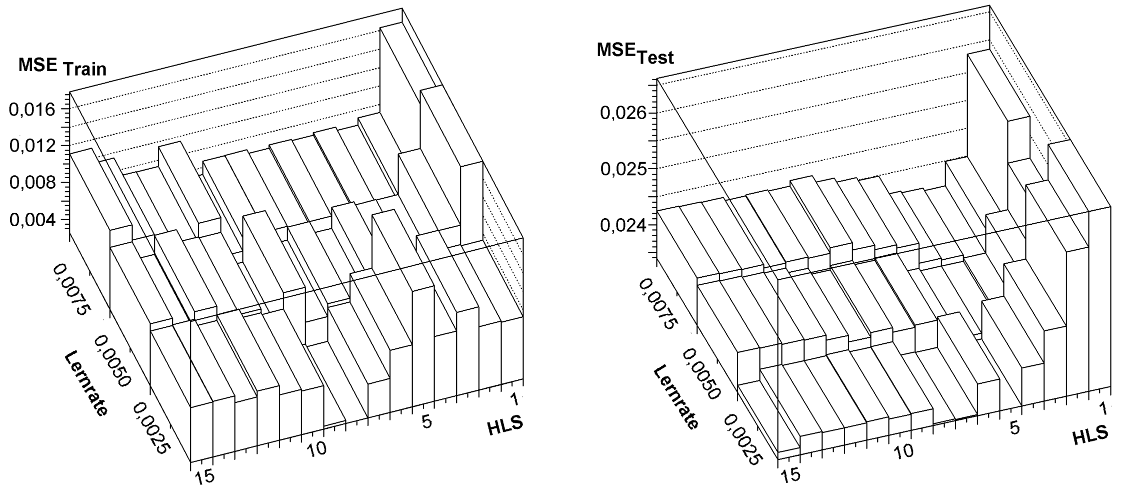
Lernrate $\eta$ und Momentumfaktor $\alpha$ beeinflussen gemeinsam die Schrittweite der Gewichtsänderung während des Trainings. Es wurden alle Parameterkombinationen von $\eta \in \{0,0025; 0,0050; 0,0075\}$ und $\alpha \in \{0,0; 0,5; 0,8; 0,9\}$ untersucht. Trainiert wurden $32 \times 63 \times 1$ Netze auf Links-Rechts Trennung bei 30% Rauschanteil. In Abbildung 6.3 ist der mittlere quadratische Fehler auf den Trainingsdaten $MSE_{Train}$ nach 50 Epochen dargestellt. Aufgrund der großen Zahl der Trainingsvektoren ($M = 65155$) bei nur 32 Bit Netzeingabe, waren 50 Epochen ausreichend — bereits nach 36 Epochen, wurde von keinem Netz noch Verbesserungen des mittleren quadratischen Fehlers auf den Testdaten $MSE_{Test}$ erzielt. Es läßt sich keine Verbesserung der Trainingsergebnisse durch von null verschiedene Momentumfaktoren erkennen, ein Resultat, das bereits in einer früher durchgeführten Studie erzielt wurde [66].

Da die benutzte Implementierung des Backpropagation–Algorithmus' auf dem CNAPS–Parallelcomputer bei Verwendung eines von null verschiedenen Momentumfaktors etwa die doppelte Rechenzeit benötigt, wurde im folgenden auf einen Momentumfaktor verzichtet.

### 6.3.3 Hidden Learning Scale

Die Lernrate $\eta$ und HLS $\beta$ wurden gemeinsam variiert, um eine geeignete Parameterkombination zu finden ($\eta \in \{0,0025; 0,0050; 0,0075\}$ und $\beta \in [1, 15]$). Trainiert wurden $32 \times 4 \times 1$ Netze auf $z$-Bestimmung ohne Rauschen. In Abbildung 6.4(a) ist der mittlere quadratische Fehler über $\beta$ und $\eta$ aufgetragen. Es zeigt sich, daß $\beta = 1$ nicht optimal ist. Dies deckt sich mit den in [66] erzielten Resultaten, die die besten Ergebnisse bei dem höchsten untersuchten Wert $\beta = 3$ lieferten.

Nach Gleichung (3.17) ist zu erwarten, daß der günstigste Wert für $\beta$ mit steigendem $\eta$ abnimmt. Dieser Effekt wird durch die großen Schwankungen des Lernerfolges in




(a) $MSE_{Train}$ der auf $z$-Bestimmung bei 0% Rauschpegel trainierten Netze

(b) $MSE_{Test}$ der auf $z$-Bestimmung bei 10% Rauschpegel trainierten Netze

**Abbildung 6.4:** Abhängigkeit der Fehler $MSE_{Train}$ bzw. $MSE_{Test}$ von $\beta$ und $\eta$

Abbildung 6.4(a) überdeckt. Der mittlere quadratische Fehler auf den Testdaten zeigt bei $32\times4\times1$ Netzen, die bei 10% Rauschanteil trainiert wurden, geringere Schwankungen. Das Absinken des Bereiches günstiger Werte für $\beta$ mit steigender Lernrate wird deutlicher, siehe Abbildung 6.4(b).

Es wurde der Versuch unternommen, einen Zusammenhang zwischen dem optimalen Wert für $\beta$ und der Zahl der versteckten Knoten $n_{hid}$ bei fester Zahl der Eingabe- und Ausgabeknoten und fester Lernrate nachzuweisen. In Abbildung 6.5 ist $MSE_{Train}$ der auf $z$-Bestimmung ohne Rauschen trainierten Netze dargestellt.

Neben der drastischen Verringerung des Fehlers bei steigendem $n_{hid}$, ist eine Begünstigung von hohen Werten für $\beta$ zu erkennen.

### 6.3.4 Zahl der versteckten Knoten

Bei steigender Zahl der versteckten Knoten werden die Trainingsmuster immer besser durch das Netz beschrieben. Wie in Abbildung 6.6(a) zu sehen ist, gilt dies nicht für die Testmuster. Dargestellt sind die quadratischen Fehler auf den Trainingsmustern $MSE_{Train}$, den Testmustern am Ende des Trainings $MSE_{Test}^{End}$ und der niedrigste quadratische Fehler auf den Testmustern $MSE_{Test}^{Best}$, der während des ganzen Trainings auftrat. Untersucht wurde $z$-Bestimmung bei 10% Rauschen. Die zunehmende Differenz zwischen $MSE_{Test}^{End}$ und $MSE_{Train}$ wird als „Auswendiglernen" der Trainingsdaten verstanden („overtraining"). Dieser Effekt tritt auf, wenn ein Netz hinreichend viele freie Parameter in Form der Gewichte $\omega$ und $\tilde{\omega}$ besitzt, um die durch die Traingsdaten vorgeschrieben Abbildung $\vec{x}^{\mu} \rightarrow \vec{t}^{\mu}$ Muster für Muster zu nachzubilden. Das Erlernen der zugrundeliegende Systematik, die sogenannte Generalisierung, wird dann nicht erreicht.

Um diesen Effekt zu verhindern, werden die Gewichte benutzt, die während des

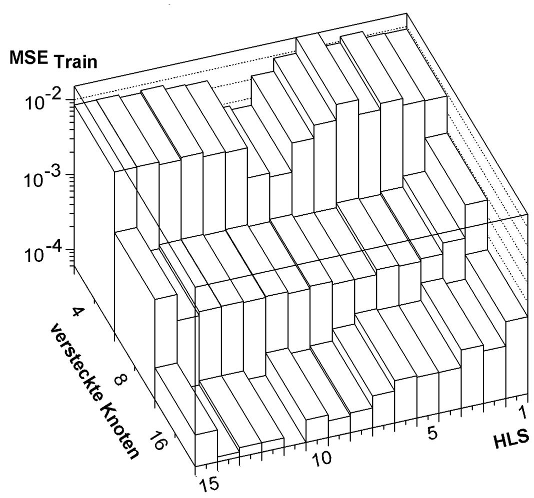
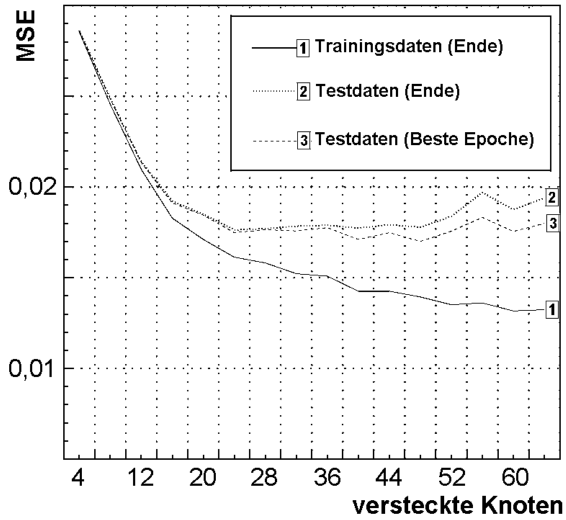
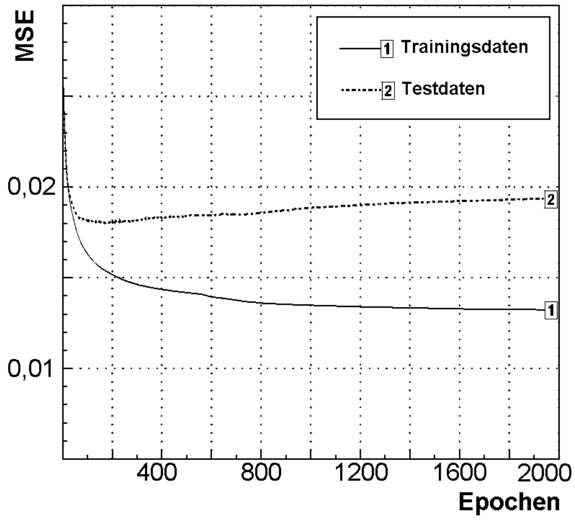



**Abbildung 6.5:** Abhängigkeit von $MSE_{Train}$ von $\beta$ und $n_{hid}$

Trainings den geringsten quadratischen Fehler auf den Testmustern $MSE_{Test}$ erreicht haben. Das benutzte Trainingsprogramm ermittelt die Epoche in der $MSE_{Test}$ minimal war und beginnt das Training erneut, somit ist es nicht nötig, die aktuellen Gewichte in jeder Epoche zu speichern. In Abbildung 6.6(b) ist der Verlauf von $MSE_{Test}$ während des Trainings, am Beispiel des $32 \times 64 \times 1$ Netzes dargestellt. Durch die Verwendung der bezüglich $MSE_{Test}$ optimalen Gewichte, bleibt $MSE_{Test}$ bei steigender Zahl der versteckten Knoten nahezu konstant. Dies zeigt, daß die optimale Zahl der versteckten Knoten nicht ermittelt werden muß, um die erreichbare Leistungsfähigkeit der Feedforward Netze hinsichtlich einer gegebenen Problemstellung zu untersuchen.




(a) Auseinanderlaufen von $MSE_{Train}$, $MSE_{Test}^{End}$ und $MSE_{Test}^{Best}$ bei wachsendem $n_{hid}$




(b) Auseinanderlaufen von $MSE_{Train}$ und $MSE_{Test}$ während des Trainings bei $n_{hid} = 64$

**Abbildung 6.6:** Verdeutlichung des „overtraining“

## 6.4 Zusammenfassung der Untersuchung

Aus der Parameterstudie werden folgende Schlüsse für das Training der Netze im nächsten Kapitel gezogen:

- Als Lernintervall wird eins gewählt („online learning").
  Es wurde eine drastische Verbesserung der Trainingsergebnisse im Vergleich zu der Verwendung der maximalen Größe des Lernintervall („batch learning") beobachtet.

- Es wird kein Momentum-BPA verwendet.
  Bei drei verschiedenen Werten der Lernrate, wurde eine systematische Verbesserung der Trainingsergebnisse mit sinkenden Werten für $\alpha$ beobachtet.

- „Hidden Learning Scale" wird größer eins gewählt.
  In allen untersuchten Fällen, konnten die Trainingsergebnisse durch die Wahl $\beta > 1$ verbessert werden.

- Die Zahl der versteckten Knoten wird nicht optimiert.
  Zwar wurde beobachtet, daß unnötig viele versteckte Knoten den Fehler auf den Testdaten verschlechtern, durch die Verwendung der Gewichte mit niedrigstem Fehler auf den Testdaten wird dieser Effekt jedoch gering gehalten. Daher kann auf die Suche nach der optimalen Zahl der Knoten verzichtet werden.

- Es wird ein versteckter Knoten pro binärer Eingabegröße verwendet.
  Werden zu wenig versteckte Knoten verwendet, verschlechtern sich die Trainingsergebnisse deutlich. Daher wird versucht die Zahl der versteckten Knoten hinreichend groß zu wählen. In den untersuchten Problemstellungen erwies sich ein versteckter Knoten pro binärer Eingabegröße als ausreichend.

Somit bleiben als anzupassende Parameter $\eta$ und $\beta$.

Obgleich nicht auszuschließen ist, daß die Resultate dieser Untersuchungen problemspezifisch sind, können sie doch als Faustregeln für das Training der Netze auf dem CNAPS dienen.

## 6.5 Vergleich mit einem genetischen Algorithmus

Der BPA und die besprochenen Erweiterungen sind in zweierlei Hinsicht nicht optimal:

- Es besteht keine Möglichkeit, ein lokales Minimum wieder zu verlassen und die Suche nach dem globalen fortzusetzen.
- Die empirische Bestimmung der Trainingsparameter $(\eta, \alpha, \beta)$ ist aufwendig.

Ein Versuch, beide Probleme zu umgehen wurde mit der Implementierung eines adaptiven genetischen Algorithmus für das Training dreilagiger Feedforward Netze, unternommen. Genetische Algorithmen sind dem durch die biologische Evolution verwirklichten Optimierungsprozess nachempfunden. Dabei wird ein Parametersatz P, hinsichtlich einer sehr komplexen Bewertungsfunktion F(P) optimiert, wobei nur die Werte $F(P_i)$ für jede untersuchte Parameterkombination $P_i$ bekannt sind. In der biologischen Evolution spielt die Erbinformation (DNS) die Rolle des Parametersatzes, die Bewertungsfunktion ist der Grad der Anpassung an die Lebensbedingungen. Beim Training neuronaler Netze bilden die Gewichte $\omega$ und $\tilde{\omega}$ den Parametersatz, die Bewertungsfunktion wird durch die Anforderungen festgelegt, denen das trainierte Netz genügen soll. Um den Trainingserfolg direkt mit dem BPA vergleichen zu können, wurde der mittlere quadratische Fehler als Bewertungsfunktion gewählt.

Das Training eines Feedforward Netzes erfolgt nach folgendem Schema:

1. Initialisierung
   Zunächst werden die Gewichte von $n$ Netzen pseudozufällig initialisiert. Für jedes der $n$ Netze wird der mittlere quadratische Fehler bezüglich allen Trainingsdaten berechnet.
2. Belohnung des besten Netzes
   Die Gewichte eines zufällig gewählten Netzes („Mutter“) werden kopiert. Dabei wird in der Kopie mindestens ein Parameter gegenüber dem Original abgeändert. Die kopierten und abgeänderten Gewichte bilden ein neues Netz („Kind“).

   Die Belohnung des besten Netzes (kleinster Wert für MSE) besteht in einer gegenüber den anderen Netzen bevorzugten Auswahl. Das beste Netze wird mit $\sqrt{(n)}$ höherer Wahrscheinlichkeit gewählt.

   Die in der Kopie auftretende Abweichung wird erreicht, indem mindestens eine der 16 implementierten Operationen auf die Gewichte des ausgewählten Netzes angewandt wird.

   Bei drei Operationen, die dem erfolgreichen biologischen Konzept der zweigeschlechtlichen Fortpflanzung nachempfunden sind, werden zusätzlich die Gewichte eines weiteren, zufällig gewählten, Netzes („Vater“) verwendet. Die Operationen sind teilweise der Natur nachempfunden, teilweise an die Funktionsweise des Feedforward Netzes angepaßt:

   - Veränderung einzelner Gewichte (Mutation)
   - Kombination der Gewichte des „Mutter“–Netzes mit denen des „Vater“-Netzes durch Austausch einzelner Bereiche der Gewichtsmatrizen (Genaustausch, Cross–over),

- Rotation des Gewichtsvektors eines oder mehrerer Knoten
- Skalierung des Gewichtsvektors eines oder mehrerer Knoten
- Veränderung der Schwelle eines oder mehrerer Knoten
- Kopieren des Gewichtsvektors eines oder mehrerer Knoten des „Vater"-Netzes auf andere Knoten des „Mutter"–Netzes
- Erneute Initialisierung des Gewichtsvektors eines oder mehrerer Knoten.

Die Wahrscheinlichkeit für das Auftreten einer bestimmten Operation wird während des Trainings an die Häufigkeit einer Verbesserung durch diese Operation angepaßt[1].

3. Bestrafung des schlechtesten Netzes
   Das Netz, das den kleinsten Fehler geliefert hat, wird gelöscht, so daß Speicherplatz für das „Kind"–Netz entsteht.

4. Bewertung
   Für das „Kind"–Netz wird der mittlere quadratische Fehler bezüglich allen Trainingsdaten berechnet.

5. Endlosschleife
   Es wird mit Punkt 2 fortgesetzt.

In dieser Implementierung müssen die Zahl der versteckten Knoten und die Zahl der Netze $n$ vorgegeben werden. Im Prinzip können diese ebenfalls durch Adaption an die Trainingsfortschritte bestimmt werden, wodurch der Algorithmus frei von äußeren Parametern wäre. Angewendet auf $z$-Bestimmung bei unverrauschten Mustern zeigt sich, daß der Algorithmus in der Lage ist, Feedforward Netze zu trainieren, allerdings werden ungleich mehr Lernzyklen benötigt als beim Training durch den BPA. Ob die

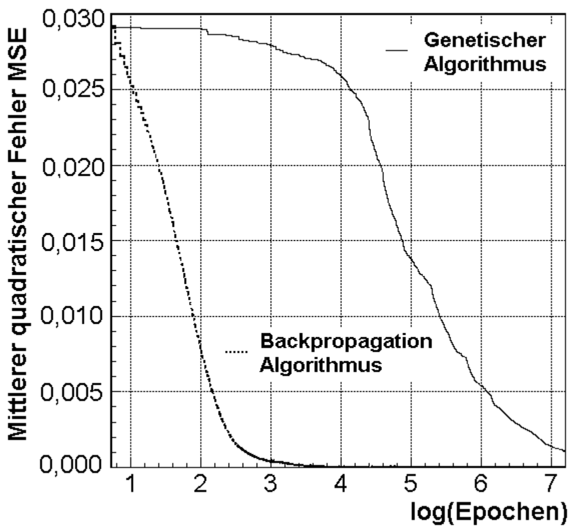



**Abbildung 6.7:** Vergleich der Lerngeschwindigkeiten

[1] Daher die Bezeichnung „adaptiver genetischer Algorithmus".

letzten Endes erreichten Fehler kleiner sind als, die durch BPA–Training erreichten läßt sich nicht feststellen, da der genetische Algorithmus auch nach $1,5 \cdot 10^6$ Lernzyklen (Rechenzeit der benutzen Workstation: 2 Monate) noch Verbesserungen zeigt. Der BPA erreichte sein Minimum bereits nach $\approx 10^4$ Epochen. In Abbildung 6.7 sind die Lernerfolge beider Algorithmen im Vergleich dargestellt. Zu beachten ist, die Abszisse vom Logarithmus der Trainingszyklen (Epochen) gebildet wird. Man sieht, daß der untersuchte genetische Algorithmus keine Alternative zum BPA darstellt.

# 7. Ergebnisse

Aufbauend auf den Ergebnissen des vorigen Kapitels wird nun die Leistungsfähigkeit der KNN bei verschiedenen Methoden der 4:1 Datenverdichtung untersucht.

Das Ziel dieser Untersuchung ist einerseits, die günstigste Datenverdichtung zu finden, andererseits, die Leistungsfähigkeit der Kombination aus BST-T Daten und KNN als Untergrundfilter zu zeigen.

Dazu wurden simulierte ep-Kollisionen und simulierte Untergrundereignisse verwendet, um die Netze zu trainieren und ihre Klassifikationseffizienz zu testen.

## 7.1 Wahl der Simulationsparameter

Im ersten Teil der Untersuchung, wurden die Parameter der Simulation so gewählt, daß das Simulationsprogramm (im Rahmen seiner Möglichkeiten) möglichst realitätsnahe Ergebnisse liefert.

Dazu wurde in der Simulation eine intrinsische Detekoreffizienz $\mathcal{E}_{intrisic}$ von 97% verwendet, wie sie in [14, S.11] für minimal ionisierende Teilchen in Si-Streifen–Detektoren angegeben ist. Einer, in [44, S.11] als „pessimistisch“ bezeichneten Schätzung zufolge, liegt das Verhältnis von effektivem Signal zu elektronischem Rauschen für minimal ionisierende Teilchen in den Pixeln des BST-T bei 6/1. Bei Setzen einer Schwelle bei der dreifachen Standardabweichung des Rauschens, erwartet man 0,1% vorgetäuschte Treffer in den Pixeln und entsprechend 0,4% in den Pads.

Als Parameter der Vertexverteilung wurden für Physikereignisse $\mu_P = 0$ mm und $\sigma_P = 90$ mm gewählt, um die Vertexverteilung der ep–Kollisionen zu approximieren [14, S.11]. Die Verteilung aus Abbildung 4.1(a) wurde durch $\mu_U = -150$ mm und $\sigma_U = 130$ mm approximiert. Als Parameter der Spurmultiplizität wurde $\lambda_P = 2,5$ in Anlehnung an Simulationsergebnisse [44, S.9] verwendet. Die Verteilung der Spurmultiplizität des Untergrundes, wie in Abbildung 4.1(b) dargestellt, wurde mit $\lambda_P = 7,5$ modelliert.

Im zweiten Teil der Untersuchung wurde der Einfluß einzelner Simulationsparameter untersucht. Dabei wurde die Stabilität der untersuchten Datenverdichtungen gegenüber gesteigertem elektronischen Rauschen in den Pixeln des BST-T und der Einfluß der Spurmultiplizität geprüft.

## 7.2 Trainingsparameter und Netzarchitektur

Es wurden dreilagige Feedforward Netze mit 128 Knoten in der versteckten Schicht verwendet. Nach Abbildung 6.6(a) ist zu erwarten, daß die resultierende Verwendung eines versteckten Knotens pro binärerer Eingabegröße ausreichend ist[1]. Bei Verwendung

[1] Die der Abbildung 6.6(a) zugrundeliegenden Netze verwenden 32 Bit Eingabe.

von Eingabewerten, die mehr als ein Bit an Information tragen, wurde ebenfalls 128 versteckte Knoten verwendet, um einen „fairen" Vergleich zu ermöglichen.

Trainiert wurde ohne Momentumterm, bei Lernintervall eins, Lernrate $\eta = 0,005$ und $\beta = 8,0$. Es wurden 30000 Trainings- und 7500 Testmuster verwendet, wobei die Summe der Trainings- und Testdaten lediglich durch die Größe des externen Hauptspeichers, auf den die CNAPS-Karte zugreifen kann, limitiert wird. Es wurde ein 1:1 Verhältnis von Physik- zu Untergrundereignissen in Trainings- und Testdaten gewählt.

## 7.3 Bewertung der 4:1 Datenreduktionen bei 0,4% Rauschen

In diesem Abschnitt werden die untersuchten Methoden, die geforderte 4:1 Verdichtung des Datenvolumens zu erreichen, vorgestellt und hinsichtlich ihrer Eignung als Netzeingabegrößen untersucht. Um den Einfluß der Datenreduktion zu quantisieren, wurde die Klassifikationsfähigkeit der Netze bei Verwendung der vollen Padmuster als Referenz untersucht. Als Maß für die Klassifikationsfähigkeit wurde die beste mittlere Effizienz der Netze bei 0,4% Rauschpegel gewählt.

### 7.3.1 Padmuster als Netzeingabe

Die untersuchten Datenreduktionen gliedern sich in Methoden, die auf Maskierung der Padmuster beruhen und solche, die das gesamte Padmuster mit verminderter Granularität übermitteln.

- **Volles Padmuster als Referenz**
  Als Netz–Eingabegrößen werden die binären Werte des Musters benutzt, was zu insgesamt 512 Eingabegrößen führt. Da die vorhandenen CNAPS-Boards mit maximal 256 Prozessoren bestückt sind, konnten Netze mit der erforderliche Zahl von 512 Eingabeknoten, nicht trainiert werden. Durch Modularisierung der Problemstellung, wie in Abschnitt 7.7 beschrieben, läßt sich die Leistungsfähigkeit der Netze mit 512 Eingabegrößen mit ausreichender Genauigkeit ermitteln.

  Die Modul-Netze wurden auf vier Sektoren trainiert und getestet, die erreichten Effizienzen anhand von Monte-Carlo Berechnungen auf 16 Sektoren hochgerechnet. Es wurde eine Klassifikationseffizienz (*BME*) von $(99,98 \pm 0,02)\%$, Der angegebene Fehler, wurde nach Gleichung (3.21) abgeschätzt.

#### 4:1 reduzierte Padmuster

Aufgrund der limitierten Bandbreite der Frontend–Elektronik, kann nicht das volle, 512 Bit grosse Padmuster, für die L2–Triggerentscheidung benutzt werden, sondern maximal 128 Bit. Daher wurden verschiedene Methoden untersucht, die geforderte 4:1 Datenreduktion auf den Padmustern durchzuführen.

- **Verminderte Auflösung in $z$ und $r$ ($z/2, r/2$)**
  Durch ODER–Verknüpfung je 4 benachbarter Pads wird das Padmuster Sektor

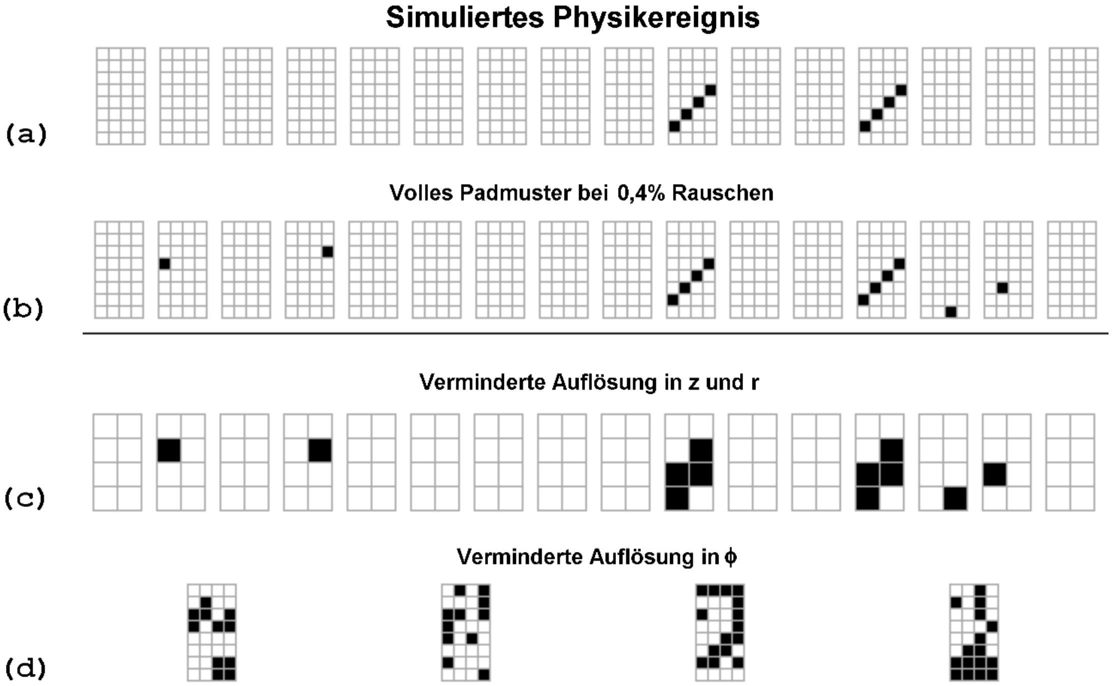



**Abbildung 7.1:** Volles Padmuster und Methoden der 4:1 Datenreduktion am Beispiel eines simulierten Physikereignisses bei 0,4% Rauschpegel

für Sektor auf ein Muster aus 8 Bits reduziert. Dies ist in Abbildung 7.1(c) und 7.2(c) dargestellt. Mit dieser Eingabekodierung erreichen die Netze. eine Effizienz (*BME*) von $(99,38 \pm 0,09)\%$.

- **Verminderte Auflösung in $z$ und Ringkodierung** $(z/2, Ring)$
  In der 2. und 4. Scheibe wird die Nummer des äußersten getroffenen Ringes ermittelt. Da diese Nummer maximal 8 betragen kann[2], werden 4 Bit pro Zahl benötigt. Als Netzeingabegrößen werden diese zwei Zahlen pro Sektor verwendet, was zu 32 Netz–Eingabegrößen führt. Mit einer Effizienz (*BME*) von $(97,1 \pm 0,2)\%$ ist dieses Verfahren deutlich schlechter als das naheliegende $(z/2, r/2)$.

- **Verminderte Auflösung in $\phi$ und $z$** $(\phi/2, z/2)$
  Während die bisher angeführten Methoden die 4:1 Reduktion Sektor für Sektor vornehmen, werden hier die Padmuster zweier in $\phi$ benachbarter Sektoren aufeinander projiziert und gleichzeitig die Auflösung in $z$ verringert. Letzeres wird erreicht, indem nur jede zweite Scheibe ausgelesen wird. *BME* beträgt $(99,55 \pm 0,08)\%$.

- **Verminderte Auflösung in $\phi$ und $r$** $(\phi/2, r/2)$
  Die Projektion in $\phi$ erfolgt wie oben, zusätzlich werden zwei in $r$ benachbarte Ringe gemeinsam ausgelesen. Die Effizienz von $(99,61 \pm 0,07)\%$ ist innerhalb der Fehler identisch mit $(\phi/2, z/2)$.

[2] 0 bedeutet kein Treffer

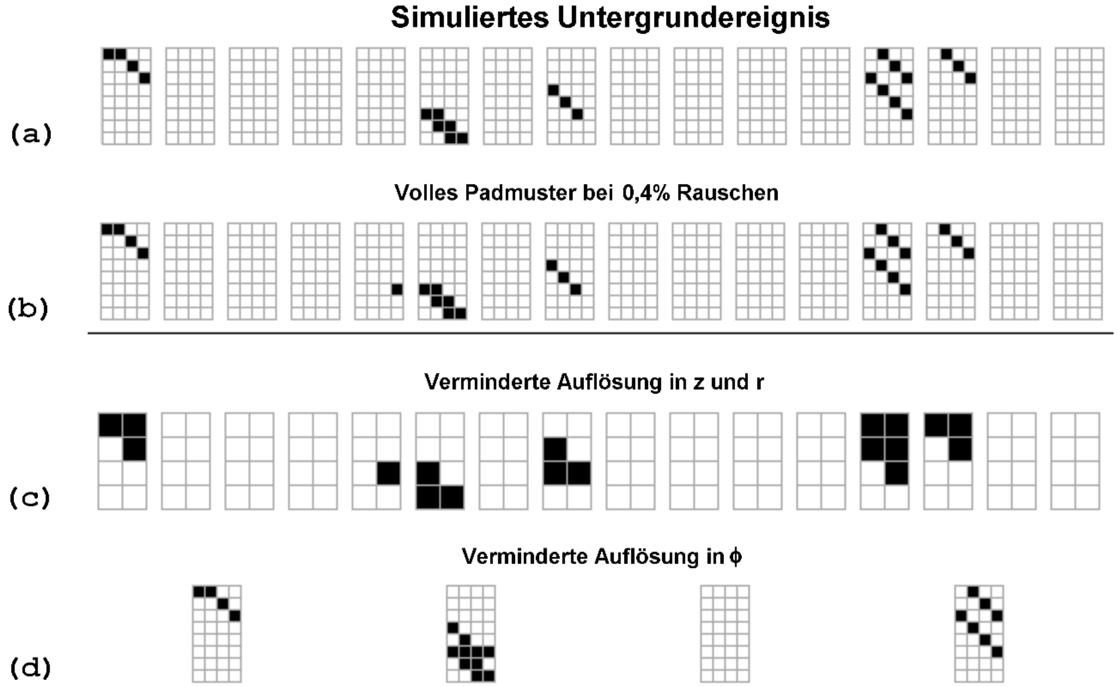



**Abbildung 7.2:** Volles Padmuster und Methoden der 4:1 Datenreduktion am Beispiel eines simulierten Untergrundereignisses bei 0,4% Rauschpegel

- **Verminderte Auflösung in $\phi$ und Ringkodierung** ($\phi/2$, *Ring*)
  Je zwei Sektoren werden aufeinander projiziert, die Nummer des äußersten getroffenen Ringes der Projektion wird Scheibe für Scheibe in eine 4 Bit Zahl kodiert. Als Netz–Eingabegrößen werden diese vier Zahlen pro zwei Sektoren verwendet, was zu insgesamt 32 Eingabegrößen führt. Mit einer Effizienz von $(98,9{\pm}0,1)\%$ ist diese Methode ungünstiger als die rein geometrischen $(\phi/2, z/2)$ und $(\phi/2, r/2)$.

- **Verminderte Auflösung in $\phi$** ($\phi/4$)
  Die Projektion in $\phi$ erstreckt sich über vier benachbarte Sektoren, dies ist in Abbildung 7.1(d) und 7.2(d) schematisch dargestellt. Die trainierten Netze erreichen bei dieser, geometrisch einfachen Methode der Datenreduktion eine Effizienz von $(99,83 \pm 0,05)\%$.

- **Summation über alle 16 Sektoren** ($\sum \phi$)
  Hier werden die Bits in Ring $i$ und Scheibe $j$ in allen Sektoren gezählt und in eine 4 Bit Zahl kodiert[3]. Die resultierenden 32 Zahlen dienen als Netz–Eingabegrößen. Die Effizienz liegt bei $(99,0 \pm 0,1)\%$.

### 7.3.2 Masken als Netzeingabe

Diese Methoden beruhen auf dem Vergleich der Padmuster mit einem vorgegebenem Satz von Masken, für jeden Sektor wird die Nummer der ersten passenden Maske als

[3] 15 und 16 werden gleichermaßen kodiert.

| Modus | *BME* [%] |
|---|---|
| Voll | $99,98 \pm 0,02$ |
| $(z/2, r/2)$ | $99,4 \quad \pm 0,1$ |
| $(z/2, Ring)$ | $97,1 \quad \pm 0,2$ |
| $(\phi/2, z/2)$ | $99,55 \pm 0,08$ |
| $(\phi/2, r/2)$ | $99,61 \pm 0,07$ |
| $(\phi/4)$ | $99,83 \pm 0,05$ |
| $(\sum \phi)$ | $99,0 \quad \pm 0,1$ |

**Tabelle 7.1:** Effizienzen der KNN mit Padmustern als Netzeingabe

Netzeingabe verwendet.

- **„Physik"–Masken und Trefferzahl** (P&T)
  Die derzeit verwirklichte Methode liefert pro Sektor die Nummer der ersten passenden Maske als 6 Bit Zahl, wodurch die Zahl der Masken auf 63 beschränkt ist[4], und die Zahl der getroffenen Pads in einer ausgewählten Scheibe als 2 Bit Zahl. Somit werden insgesamt 32 Netz–Eingabegrößen benutzt.

  Es wurden die 63, bei der durch $\mu_P = 0$ mm und $\sigma_P = 90$ mm festgelegten Vertexverteilung am häufigsten auftretenden Masken verwendet, wobei nur Spuren betrachtet wurden, die mindestens drei Scheiben durchdringen. In der Reihenfolge ihrer Häufigkeit sind dies die Masken vom Typ $(+1, +1, +1)$, $(+2, +1, +1)$, $(+1, 0, +1)$, $(+1, +1, 0)$, $(+1, +2, +1)$, $(+1, +1, +2)$, $(+0, +1, +1)$, $(+2, +1, +2)$, $(+2, +2, +1)$ und $(+2, +2, +2)$. Da es sich dabei um nur 62 Masken handelt, wurde zusätzlich die nächsthäufige Maske $(0, 1, 3, 5)$ verwendet. Durch diese Masken werden 99,997% aller simulierten Physikspuren durch mindestens drei Scheiben gefunden. Da in Trainings- und Testdaten auch Physikspuren durch zwei Scheiben enthalten sind, werden insgesamt 76,838% der Physikspuren gefunden.

  Auf Basis dieser Information erreichen die KNN eine Klassifikationseffizienz von $(98,6 \pm 0,4)\%$.

### Masken für die Untergrunderkennung

Bei der Reduktion des Datenvolumens durch Maskierung der Padmuster wird im Gegensatz zur Reduktion durch verringerte Auflösung Wissen über den physikalischen Sachverhalt benutzt. Dementsprechend ist zu erwarten, daß der Informationsverlust geringer und damit die von den Netzen erreichbare Klassifikationseffizienz höher ist. Für die, in dieser Arbeit untersuchte neue Anwendungsmöglichkeit des BST-T zur Untergrundunterdrückung, ist es daher naheliegend, das Wissen über den Untergrund — geradlinige Spuren aus rückwärtiger Richtung — ebenfalls zu nutzen. Es wird vorgeschlagen, zusätzlich zu den „Physik"–Masken auch speziell auf die Spuren der Untergrundereignisse abgestimmte „Untergrund"–Masken zu benutzen.

[4] Eine der 64 Bitkombinationen muß reserviert bleiben für den Fall, daß keine passende Maske gefunden wurde.

- **Physik-, Untergrundmasken und Trefferzahl** (PU&T)
  Der Satz an benutzen Untergrundmasken wurde analog zu den Physikmasken durch Monte-Carlo Simulation ermittelt. Es wurden die Masken benutzt, die von Spuren durch mindestens drei Scheiben des BST mit Vertex in (0;0;$z$), wobei $z$ um $\mu = -1,5$ m mit Standardabweichung $\sigma = 13$ cm gaußverteilt ist, am häufigsten generiert wurden. Bei Verwendung der Masken der Klassen $(-1,-1,-1)$, $(0,-1,-1)$, $(-1,0,-1)$, $(-1,-1,-2)$, $(-1,-1,0)$, $(0,-1,-2)$, $(-1,0,-2)$ und $(-1,-2,-1)$, werden der Simulation zufolge 98,545% der Untergrundspuren gefunden. Es handelt sich dabei um insgesamt 54 Masken, die Klassen sind in der Reihenfolge ihrer Häufigkeit aufgeführt. Die erreichte Effizienz von $(99,92 \pm 0,03)$% bestätigt die Maskierung als problemgerechte Methode der Datenreduktion.

- **Physik- und Untergrundmasken** (PU)
  Die Verwendung von 63 Physik- und 54 Untergrundmasken erfordert 7 Bit zur Kodierung der Maskennummer. Daher wurden allein die 16 Maskennummern als Netzeingabegrößen verwendet und auf den Versuch, die Zahl der Treffer in nur einem Bit zu kodieren verzichtet. Die Effizienz von (PU&T) wurde exakt reproduziert. Dies zeigt, daß bei Verwendung von Physik- und Untergrundmasken auf die Trefferzahl verzichtet werden kann.

- **Physik-, ausgewählte Untergrundmasken** (PU22)
  Nach dem aktuellen Informationsstand darf die Zahl der Masken 100 nicht weit übersteigen [14]. Somit ist unklar, ob die Anwendung von 117 Masken realisiert werden kann. Bei Verwendung nur der häufigsten Klassen von Masken $(-1,-1,-1)$, $(0,-1,-1)$ und $(-1,0,-1)$ reduziert sich die Zahl der Untergrundmasken auf 22, wobei immer noch 83,723% der Untergrundspuren gefunden werden. Die Effizienz der Netze sinkt geringfügig auf $(99,73\pm0,06)$% ohne Trefferzahl und $(99,76 \pm 0,06)$% bei Verwendung der Trefferzahl.

| Modus | *BME* [%] |
|---|---|
| (P&T) | $98,6 \pm 0,1$ |
| (PU&T) | $99,92 \pm 0,03$ |
| (PU) | $99,92 \pm 0,03$ |
| (PU22&T) | $99,76 \pm 0,06$ |
| (PU22) | $99,73 \pm 0,06$ |

**Tabelle 7.2:** Effizienzen der KNN mit Maskennummern (und Trefferzahlen) als Eingabe

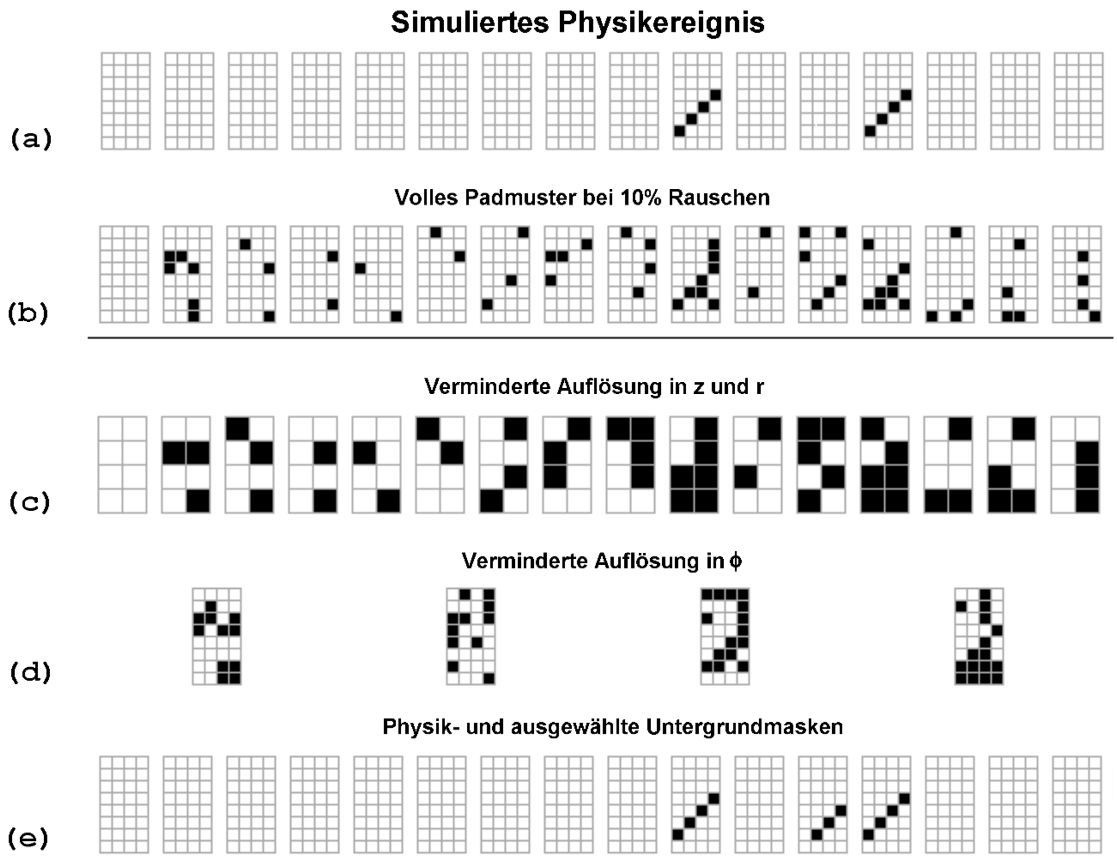



**Abbildung 7.3:** Volles Padmuster und Methoden der 4:1 Datenreduktion am Beispiel eines simulierten Physikereignisses bei 10,0% Rauschpegel

## 7.4 Untersuchung der Rauschstabilität

Um die Eignung der untersuchten Methoden besser beurteilen zu können, wurde die Stabilität der erreichten Effizienzen gegenüber elektronischem Rauschen untersucht. Dazu wurde die Effizienz der Netze anhand von Testdaten ermittelt, die einen Rauschanteil von 0,4%; 4,0%; 8,0%; 12,0%; 16,0% und 20,0% aufweisen. Dies wurde sowohl für die im vorigen Abschnitt besprochenen Netze, die für einen Rauschanteil von 0,4% trainiert wurden, als auch für Netze mit 10,0% und 20,0% Rauschen in den Trainingsdaten durchgeführt. In Abbildung 7.5(a) ist das Verhalten der Effizienzen für die Datenreduktion $(z/2, r/2)$ durch drei Kurven dargestellt. Die steil abfallende, gestrichelte Kurve zeigt die drastische Verschlechterung der Effizienz des mit 0,4% trainierten Netzes, bei steigendem Rauschpegel der Testdaten. Die flacher verlaufende, gepunktete Kurve stellt das Verhalten des anhand von 10,0% Rauschen trainierten Netzes dar, das einerseits eine deutlich gesteigerte Rauschstabilität, andererseits eine geringere Effizienz bei niedrigem Rauschpegel zeigt. Das anhand von 20,0% Rauschen trainierte Netz zeigt bis 8,0% verrauschten Testdaten steigende Effizienz, es erreicht bei 0,4% Rauschen eine Effizienz von nur 51%, was die völlige Unfähigkeit bedeutet,

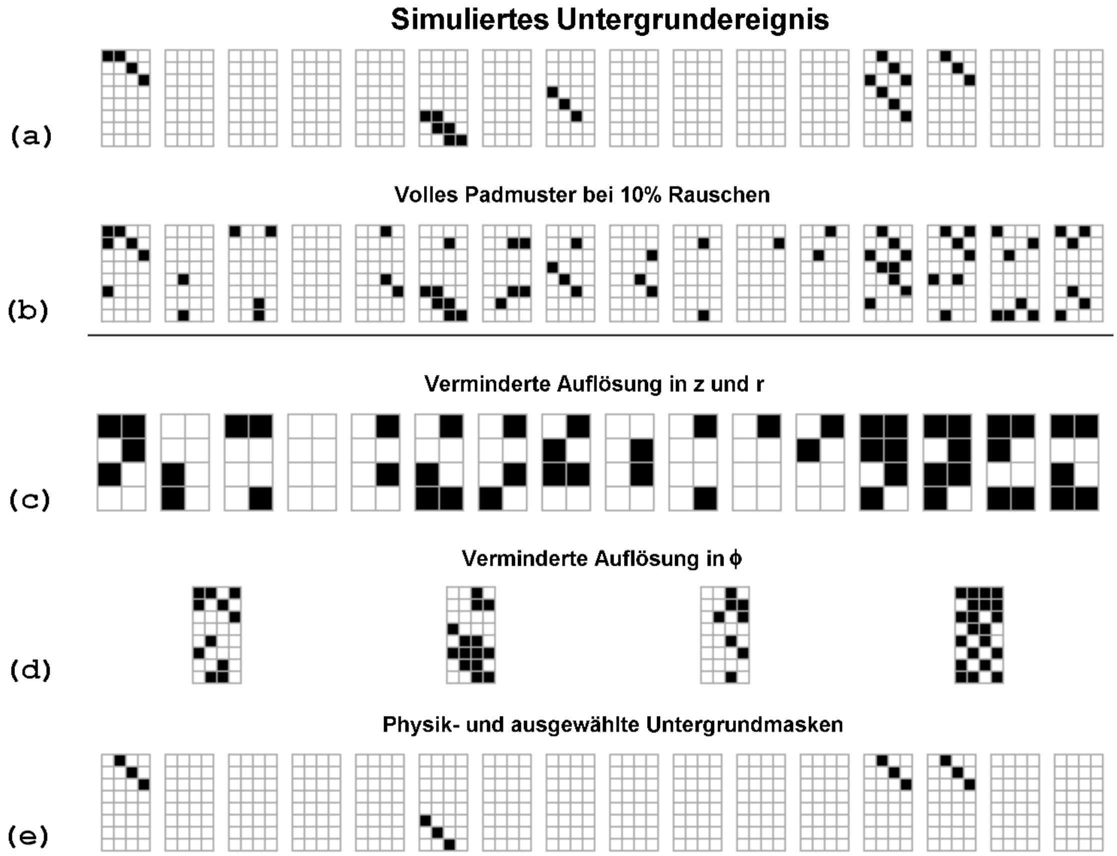



**Abbildung 7.4:** Volles Padmuster und Methoden der 4:1 Datenreduktion am Beispiel eines simulierten Untergrundereignisses bei 10,0% Rauschpegel

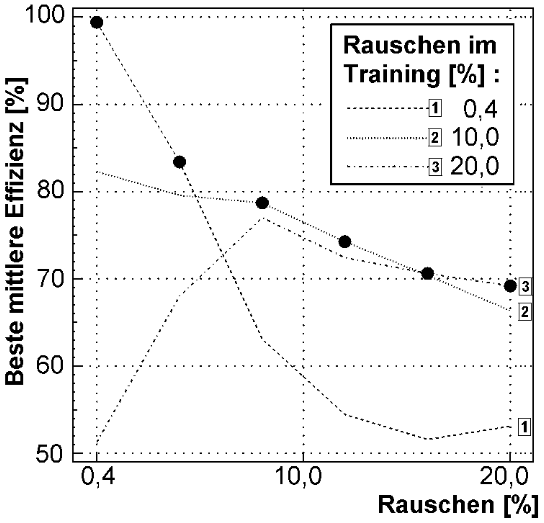
die Ereignisse zu klassifizieren. Wie zu erwarten ist jedes Netz in dem Bereich des Rauschpegels optimal, für den es trainiert wurde. Durch die Punkte ist der Verlauf der Effizienz dargestellt, wie er sich bei Verwendung der geeignet trainierten Netze ergibt.

### 7.4.1 Unsicherheiten durch Trainingseffekte

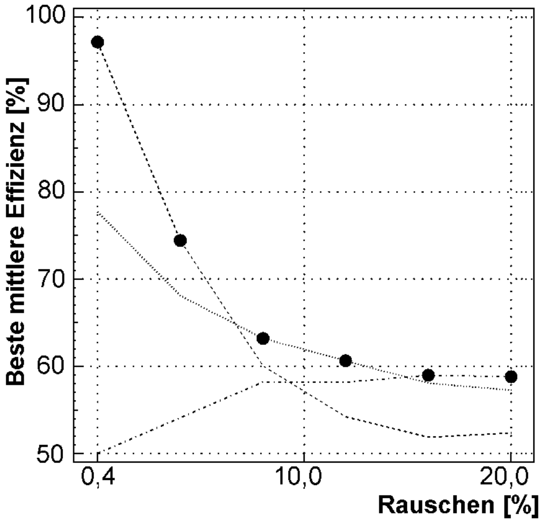
Das Ansteigen der 0,4% Kurve zwischen 16,0% und 20,0% Rauschen, kann nicht als statischer Fehler gedeutet werden (Anstieg um $2,6\ \sigma$). Dieser Effekt wird als Abweichung der Trainingsergebnisse bei 16,0% Rauschen vom Optimum betrachtet. In gleicher Weise wird der Anstieg der 0,4% Kurve bei 20,0% Rauschen in Abbildung 7.5 sowie der Einbruch der 20,0% Kurve bei 12,0% Rauschen in Abbildung 7.6(a) und Abbildung 7.6(c) erklärt.

Da solche Unregelmäßigkeiten der Effizienzkurven nur bei Effizienzen unter 80% beobachtet wurden und die Abweichungen der Effizienzen sowohl zwischen (PU&T) und (PU) also auch (PU22&T) und (PU22) rein statistisch erklärt werden können, wird angenommen, daß die Tendenz, das Training in einem ungünstigem lokalen Minimum zu beenden, mit sinkender erreichter Effizienz zunimmt. Siehe dazu auch Abschnitt 7.5.

Beim Vergleich der Effizienzen der untersuchten Datenreduktionen muß daher bei




(a) $(z/2, r/2)$




(b) $(z/2, Ring)$

**Abbildung 7.5:** Effizienzen der KNN bei verschiedenen Methoden der 4:1 Datenverdichtung

deutlich von 100% abweichenden Effizienzen mit einer Unsicherheit durch nicht optimales Training im Prozentbereich gerechnet werden. Die Tatsache, daß sich das Ansteigen in Abbildung 7.5(a) und Abbildung 7.5(b), sowie der Einbruch Abbildung 7.6(a) und Abbildung 7.6(c) jeweils wiederholen, ist zu verstehen, da die Trainingsdaten bei festem Rauschen für alle untersuchten Datenreduktionen aus denselben Padmustern abgeleitet wurden.

### 7.4.2 Vergleich der Rauschstabilität der Datenreduktionen

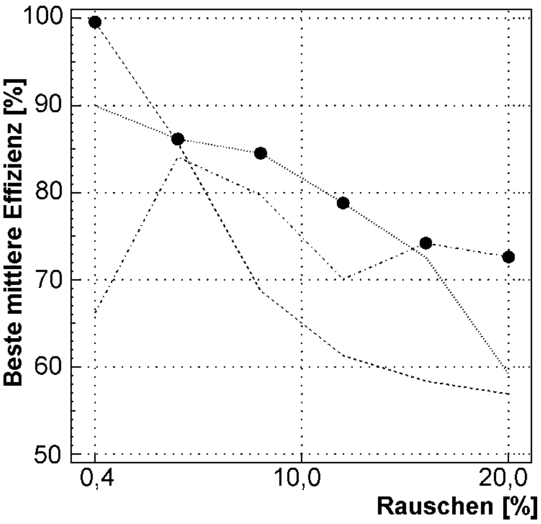
Der Vergleich der Rauschstabilität der einzelnen Möglichkeiten der Datenreduktion basiert auf den Effizienzen der am besten angepassten Netze, in den Diagrammen durch Punkte dargestellt.

#### Padmuster als Netzeingabe

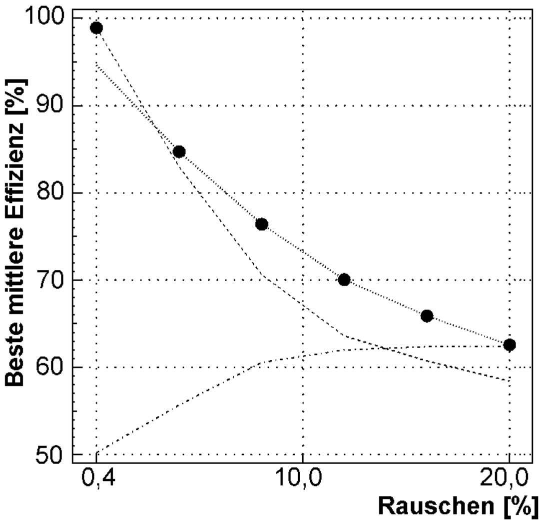
In Abbildung 7.5 ist zu erkennen, daß $(z/2, Ring)$ insgesamt ungünstiger ist, als $(z/2, r/2)$. Analog ist $(\phi/2, r/2)$ insgesamt effizienter als $(\phi/2, Ring)$ (Abbildung 7.6(a)(b)), $(\phi/2, z/2)$ ist nicht dargestellt, da der Unterschied zu $(\phi/2, r/2)$ marginal ist.

Interessant ist der Vergleich von $(\phi/4)$ und $(\sum \phi)$ in Abbildung 7.6(c) und Abbildung 7.6(d). Während $(\phi/4)$ bei 0,4% Rauschen die effizienteste, auf verringerter Auflösung basierende Datenreduktion ist, erweist sich $(\sum \phi)$ als erheblich rauschstabiler.

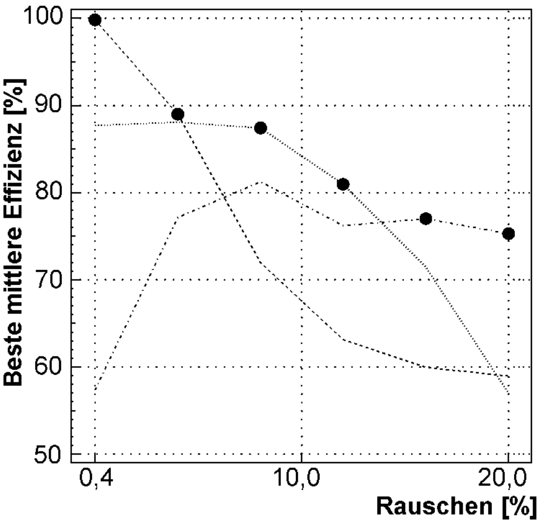
Bei voller Granularität ergibt sich erwartungsgemäß die beste Rauschstabilität, Abbildung 7.6(e). Bemerkenswert ist, daß das bei 10,0% trainierte Netz über einen großen Bereich des Rauschpegels sehr effizient ist, die Effizienz bei 0,4% Rauschen liegt bei $(99,81 \pm 0,05)\%$.



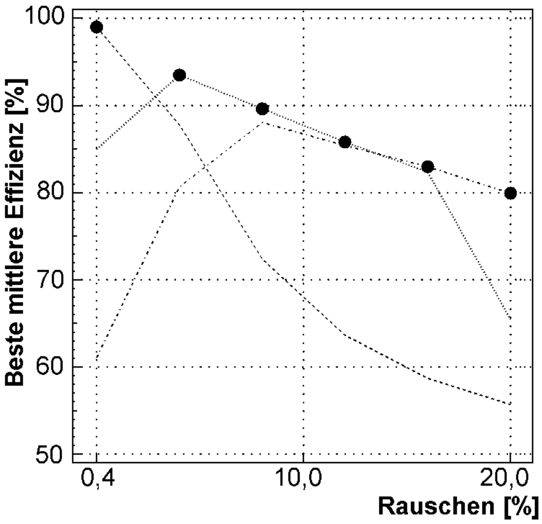

(a) $(\phi/2, z/2)$




(b) $(\phi/2, Ring)$




(c) $(\phi/4)$

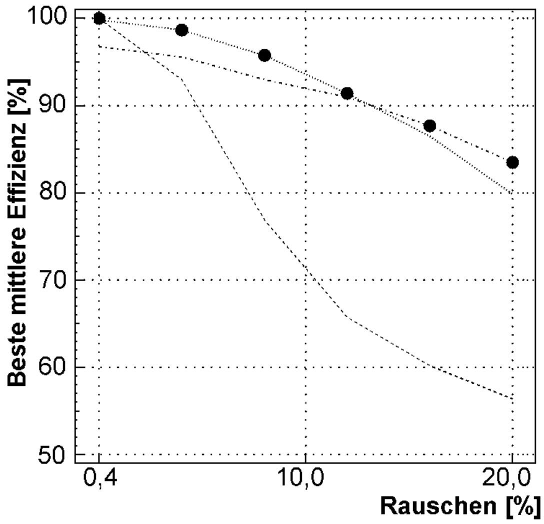



(d) $(\sum\phi)$

## Masken als Netzeingabe

Die derzeit verwirklichte Methode (P&T) ist insgesamt weniger effizient als die besten auf Padmustern basierenden Datenreduktionen $(\phi/4)$ und $(\sum\phi)$, Abbildung 7.7(a). Besonders drastisch ist der Unterschied bei 0,4% und 20,0% Rauschanteil, $(98,6\pm0,1)$% gegenüber $(\phi/4)$ mit $(99,83 \pm 0,05)$% und 70% gegenüber $(\sum\phi)$ mit 80%.

In Abbildung 7.7(b)(c) sind die Effizienzen der auf Maskennummern trainierten Netze, bei Verwendung von Physik- und Untergrundmasken dargestellt. Die Rauschstabilität ist bei Verwendung der 22 häufigsten Untergrundmasken (PU22) größer als bei 54 Untergrundmasken. Um dies zu verstehen, müssen zwei gegenläufige Effekte




(e) Volles Padmuster

**Abbildung 7.6:** Effizienzen der KNN bei verschiedenen Methoden der 4:1 Datenverdichtung

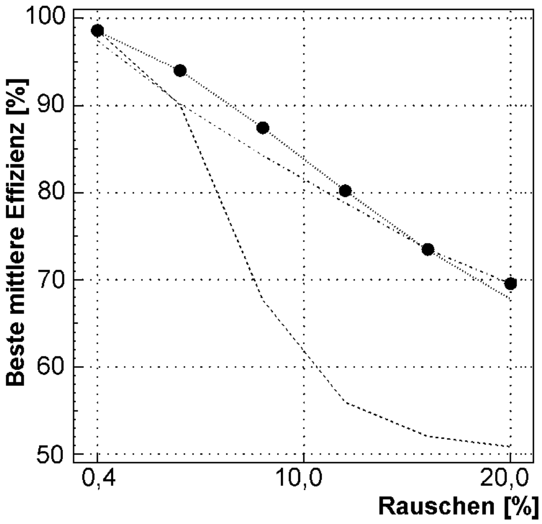
betrachtet werden. Die Wahrscheinlichkeit, ein Untergrundereignis deshalb nicht zu erkennen, weil keine passende Maske vorhanden ist, ist bei Verwendung von nur 22 Untergrundmasken größer als mit 54 Masken. Da es aber ausreicht, eine der durchschnittlich 7,5 Untergrundspuren ($\lambda_P = 7,5$) durch die Maskierung zu erkennen, sinkt die Untergrundeffizienz nur gering. Andererseits steigt mit jeder zusätzlichen Maske das Risiko, daß durch Rauschen zufällig ein passendes Muster entsteht. Da bereits *ein* zufälliges, eine Untergrundspur vortäuschendes Muster in einem der 16 Sektoren ausreicht um ein Physikereignis zu verwerfen, ist bei hohem Rauschen die Physikeffizienz deutlich besser, wenn weniger Untergrundmasken verwendet werden.

Die im Vergleich mit anderen Methoden überlegene Rauschstabilität, die sogar die auf vollem Padmuster trainierten Netze übertrifft, zeigt, das die Maskierung nicht nur die optimale Methode der Datenreduktion ist, sondern auch eine problemgerechte Datenvorverarbeitung ist. Die gute Rauschunterdrückung wird in Abbildung 7.3 und 7.4 verdeutlicht, die ein simuliertes Physik- und Untergrundereignis bei 10,0% Rauschen zeigen.

Man erkennt, daß die Maskierung mit Physik- und Untergrundmasken das Rauschen ausblendet und die wesentliche Information bewahrt. Demgegenüber sind ($\phi/4$) und auch das volle Padmuster deutlich schwieriger zu erkennen.

## 7.5 Einfluß der Trefferzahl

Bei Maskierung mit Physik- und Untergrundmasken ist der Vergleich der Effizienzen mit und ohne Verwendung der Trefferzahl aufschlußreich. In Abbildung 7.8 sind die Differenzen zwischen den Effizienzen der (PU&T) und den Effizienzen der (PU) Netze für 0,4%, 10,0% und 20,0% Rauschen im Training bei unterschiedlichem Rauschanteil der Testdaten dargestellt. Zum Vergleich ist der $1\sigma$-Bereich des statistischen Fehlers




(a) (P&T)

der Differenz der Effizienzen dargestellt.

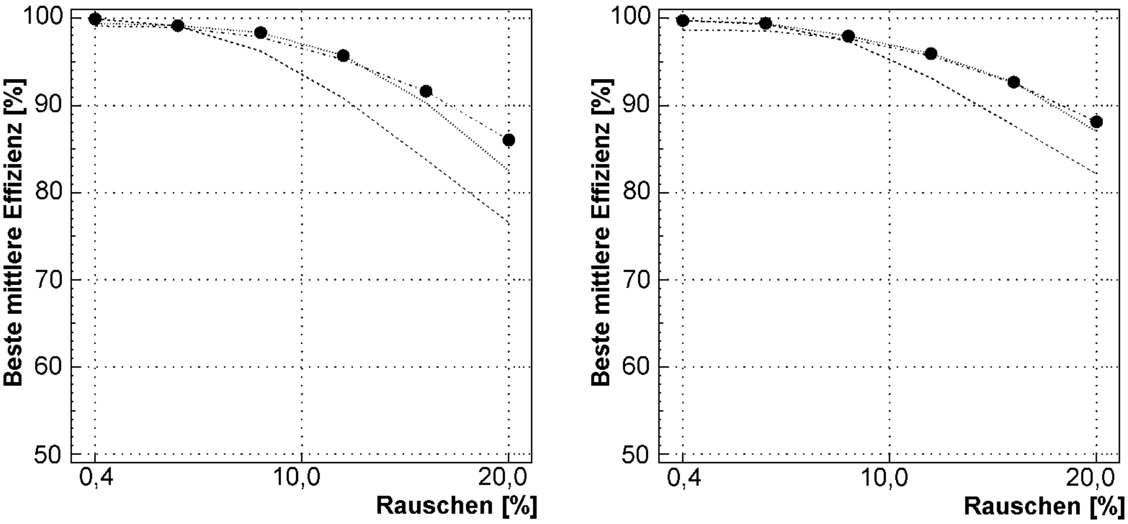
Das steile Abfallen der 0,4% Kurve zeigt, daß das (PU&T) Netz die Trefferzahl benutzt und bei deutlich über dem Trainingsniveau liegendem Rauschpegel irregeführt wird. Bemerkenswert ist, daß das (PU&T) Netz obwohl die Trefferzahl benutzt wird, bei 0,4% Rauschpegel keine bessere Effizienz als das (PU) Netz erreicht, dem die Trefferzahl nicht zur Verfügung steht. Die bei 10,0% und 20,0% Rauschen trainierten (PU&T) Netze erreichen unabhängig vom Rauschpegel der Testdaten ähnliche Effizienzen wie die (PU) Netze, was zeigt, daß diese Netze „gelernt" haben, die Trefferzahl nicht zu benutzen.

Der Mittelwert der Differenzen der bei 10,0% und 20,0% Rauschen trainierten Netze liegt bei $(0,1 \pm 0,2)$%. Dies verdeutlicht einerseits, daß sich die Effizienzen der bei diesem Rauschen trainierten (PU&T) und (PU) Netze nicht unterscheiden. Andererseits zeigt sich, daß die Unsicherheit durch Trainingseffekte, zumindest bei den beobachteten hohen Effizienzen, klein ist.

## 7.6 Vergleich dreilagiger mit vierlagigen Netzen

Die benutzen dreilagigen Netze wurden hinsichtlich der Effizienz mit vierlagigen Netzen, bei einem Rauschpegel von 10% verglichen. In der zweiten versteckten Schicht wurde die gleiche Zahl von Knoten benutzt, wie in der ersten. Für das Training der vierlagigen Netze wurden alle Parameterkombinationen von $\eta \in \{0,0025; 0,0050; 0,0075\}$ und $\beta \in \{4,0; 8,0\}$ untersucht, da nicht a priori davon ausgegangen werden kann, daß die Kombination $(\eta = 0,0050; \beta = 8,0)$, die für die untersuchten dreilagigen Netze günstig ist, auch für vierlagige Netze geeignet ist. Die Lernraten in den beiden versteckten Schichten sind identisch, $\beta$ bestimmt das Verhältnis der Lernrate in der Ausgabeschicht zur Lernrate in den versteckten Schichten [60, S.13].

Die Tabelle 7.3 aufgeführten Effizienzen der vierlagigen Netze sind jeweils die be-




(b) (PU) (c) (PU22)

**Abbildung 7.7:** Effizienzen der KNN bei 4:1 Datenverdichtung durch Maskierung

sten unter allen getesteten Parameterkombinationen. Unter Berücksichtigung der angegeben statistischen Fehler, erreichen die vierlagigen Netze im Mittel eine $(0,2 \pm 0,2)$ Prozentpunkte höhere Effizienz, somit läßt sich keine Überlegenheit der vierlagigen Netze nachweisen.

| Modus | 3 Lagen *BME* [%] | 4 Lagen *BME* [%] |
|---|---|---|
| $(z/2, r/2)$ | $76,0 \pm 0,5$ | $77,2 \pm 0,5$ |
| $(\phi/4)$ | $84,6 \pm 0,4$ | $84,8 \pm 0,4$ |
| (P&T) | $83,9 \pm 0,4$ | $83,8 \pm 0,4$ |
| (PU) | $97,7 \pm 0,2$ | $97,3 \pm 0,2$ |
| (PU22) | $97,1 \pm 0,2$ | $97,2 \pm 0,2$ |

**Tabelle 7.3:** Effizienzen drei- und vierlagiger Netze im Vergleich

## 7.7 Modularisierung der Problemstellung

Beim Training der Netze anhand aller 16 Sektoren, wird die zugrunde liegende azimutale Symmetrie lediglich implizit über die Trainingsdaten vermittelt. Um diese geometrische Information direkt in die Architektur des Netzes einfließen zu lassen, kann anstelle eines großen Netzes, das alle Sektoren bearbeitet, ein Satz kleinerer, identischer Netze verwendet werden, von denen jedes nur einen Teil der Sektoren bearbeitet. Entsprechend der geringeren Zahl der Gewichte, kann ein solches Modul–Netz mit geringer Statistik trainiert werden, was beim Training mit Realdaten bedeutsam werden kann. Die Ausgabe der Module kann anschließend logisch verknüpft werden, um zu einer Triggerentscheidung zu kommen. Die Zahl der Sektoren die ein Modul mindestens bearbeiten muß, wird durch die verwendete Methode der 4:1 Datenreduktion bestimmt:

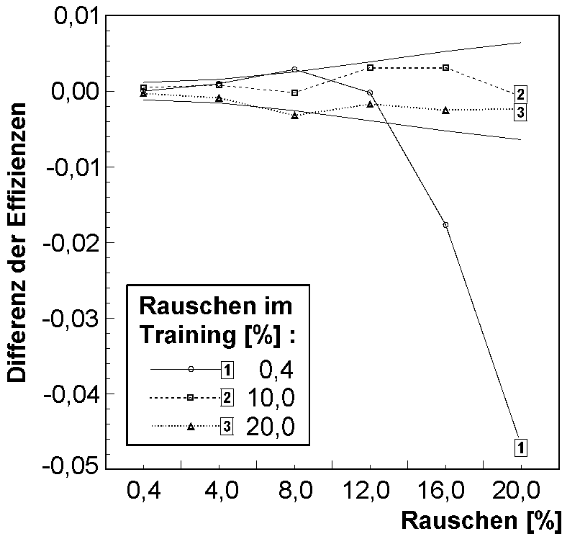



**Abbildung 7.8:** Einfluß der Trefferzahl als Netzeingabegröße

- 1 Sektor
  für $(z/2, r/2)$, $(z/2, Ring)$ und alle Maskierungsmethoden
- 2 Sektoren
  für $(\phi/2, z/2)$, $(\phi/2, r/2)$ und $(\phi/2, Ring)$
- 4 Sektoren
  für $(\phi/4)$
- Nicht modularisierbar
  ist $(\sum \phi)$.

Beim Training mit Realdaten wird zu untersuchen sein, ob sich die fehlende Nutzung eventuell vorhandener Korrelationen zwischen den Sektoren durch die Modulnetze nachteilig auswirkt. In Tabelle 7.4 und 7.5 wird die mittlere beste Effizienz der

| Modus | Zahl der Module | Training auf 16 Sektoren | Modulnetz–Architektur |
|---|---|---|---|
| $(z/2, r/2)$ | 16 | $99,3 \pm 0,1$ | $99,1 \pm 0,1$ |
| $(z/2, Ring)$ | 16 | $97,4 \pm 0,2$ | $98,0 \pm 0,2$ |
| $(\phi/4)$ | 4 | $99,9 \pm 0,1$ | $99,9 \pm 0,1$ |
| (PU) | 16 | $99,9 \pm 0,1$ | $99,9 \pm 0,1$ |
| (PU22) | 16 | $99,8 \pm 0,1$ | $99,8 \pm 0,1$ |

**Tabelle 7.4:** *BME* [%] der Modulnetz-Architektur bei 0,4% Rauschpegel

Modulnetz–Architektur mit den auf 16 Sektoren trainierten Netzen verglichen. Der angegeben Fehler ist rein statistisch, wegen der geringen Effizienz bei 20,0% Rauschen, muß zusätzlich eine Trainingsunsicherheit im Prozentbereich angenommen werden. Bei

| Modus | Zahl der Module | Training auf 16 Sektoren | Modulnetz–Architektur |
|---|---|---|---|
| $(z/2, r/2)$ | 16 | $67,6 \pm 0,5$ | $62,4 \pm 0,6$ |
| $(z/2, Ring)$ | 16 | $57,7 \pm 0,6$ | $53,5 \pm 0,6$ |
| $(\phi/4)$ | 4 | $73,9 \pm 0,4$ | $74,2 \pm 0,4$ |
| (PU) | 16 | $86,2 \pm 0,4$ | $75,4 \pm 0,4$ |
| (PU22) | 16 | $88,6 \pm 0,4$ | $83,4 \pm 0,4$ |

**Tabelle 7.5:** *BME* [%] der Modulnetz-Architektur bei 20,0% Rauschpegel

0,4% Rauschen stimmen die Effizienzen der Modulnetz–Architektur gut mit den auf 16 Sektoren trainierten Netzen überein, bei 20,0% Rauschpegel zeigt sich eine systematische Benachteiligung der Modulnetz–Architektur mit 16 Modulen. Für den $(\phi/4)$ Modus, bestehend aus vier Modulen, ergibt sich auch bei 20,0% Übereinstimmung. Diese Beobachtung kann durch den Einfluß der logischen Verknüpfung der einzelnen Module erklärt werden. Dabei wird das Ereignis als Untergrund klassifiziert, falls *ein* Modul Untergrund signalisiert. Um bei 16 Sektoren eine optimale Klassifikation zu erreichen, muß daher die Wahrscheinlichkeit Physik- oder leere Sektoren als Untergrund einzuordnen für die Module minimal sein. Die Untergrundeffizienz der einzelnen Module kann vergleichsweise niedrig sein, da es ausreichend ist, von den durchschnittlich 7,5 Untergrundspuren in 16 Sektoren eine zu erkennen. Netze, die auf nur einem Sektor trainiert werden, können diese Anforderung nicht lernen. Die Beobachtung, daß der Effizienzverlust um so größer ist, je mehr Module verwendet werden und je höher die Wahrscheinlichkeit ist, durch Rauschen untergrundähnliche Muster zu erzeugen, ist somit zu erwarten.

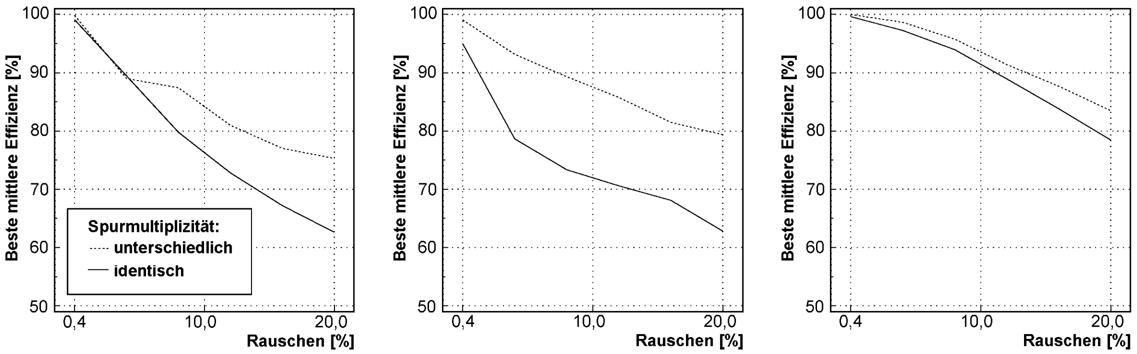
Die geringen Unterschiede im $(\phi/4)$ Modus stützen die Annahme, daß die Aufteilung der bei voller Granularität trainierten Netze in vier Module die Effizienz gegenüber einem auf 16 Sektoren trainiertem Netz nicht wesentlich verändert.

## 7.8 Einfluß der Spurmultiplizität

Die bisher betrachteten simulierten Physik- und Untergrundereignisse unterscheiden sich sowohl hinsichtlich der $z$-Vertex Verteilung, als auch der Verteilung der Spurmultiplizität. Um die Effizienz bei reiner Vertexkennung zu untersuchen, wurde in diesem Abschnitt für Physik- und Untergrundereignisse die gleiche Verteilung der Spurmultiplizität benutzt. Gewählt wurde eine Gleichverteilung über dem Intervall $[1, 9]$. Dargestellt ist der Verlauf der Effizienz bei Verwendung der geeignet trainierten Netze, wie in Abbildung 7.5 und 7.6 durch Punkte symbolisiert, bei unterschiedlicher und identischer Verteilung der Spurmultiplizität.

In Abbildung 7.9(b) erkennt man die deutliche Verschlechterung der Effizienz der $(\sum \phi)$ Datenreduktion. Wie, für einen auf Summation basierenden Algorithmus zu erwarten, ist für diese Methode die Zahl der Spuren für die Klassifikation wesentlich. Demgegenüber sind die $(\phi/4)$ Netze und besonders die auf voller Granularität trainierten Netze wesentlich besser in der Lage, Physik- und Untergrund allein aufgrund der

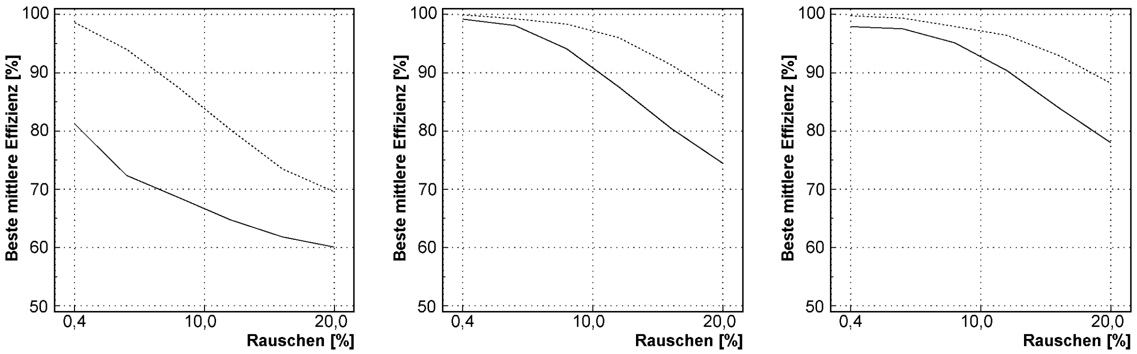
$z$-Vertexverteilung zu trennen, Abbildung 7.9(a)(c).




(a) $(\phi/4)$ (b) $(\sum\phi)$ (c) Volles Padmuster

**Abbildung 7.9:** Padmuster als Netzeingabe

Da die (P&T) Netze die Trefferzahl, eine aus der Spurmultiplizität direkt abgeleitete Größe, als Eingabegröße benutzen, ist der drastische Abfall der Effizienz in Abbildung 7.10(a) qualitativ zu erwarten. Die (PU) und (PU22) Netze verlieren bei identischen Verteilungen der Spurmultiplizitäten weit weniger an Effizienz, die Klassifikation wird somit durch Vertexerkennung getroffen, Abbildung 7.10(b)(c).




(a) (P&T) (b) (PU) (c) (PU22)

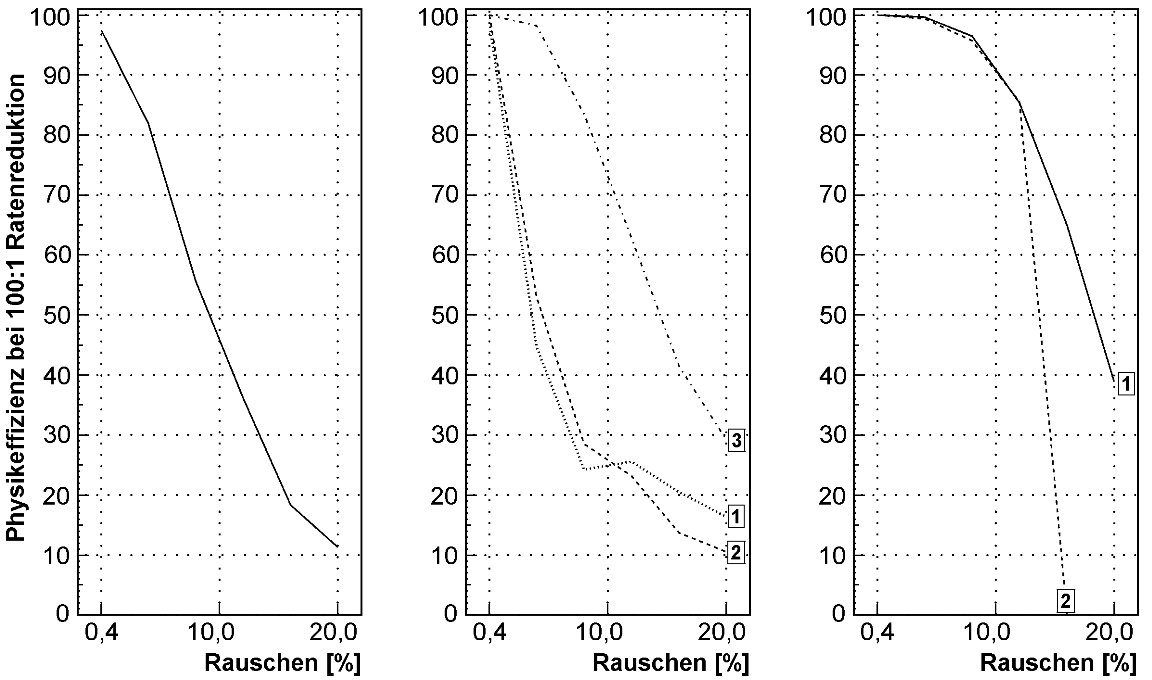
**Abbildung 7.10:** Masken als Netzeingabe

## 7.9 Bewertung der Leistungsfähigkeit

Um die Leistungsfähigkeit eines auf den BST-T Daten basierenden Untergrundfilters zu beurteilen, wurde untersucht, welche Physikeffizienz die Netze erreichen, wenn die Untergrundrate auf ein Hundertstel zu senken ist. Die Untersuchungen basieren auf den in Abschnitt 7.4 beschriebenen Monte-Carlo Ereignissen.

In Abbildung 7.11 ist die erreichte Physikeffizienz bei der geforderten 100:1 Unterdrückung der Untergrundrate gegen den Rauschpegel aufgetragen.

Werden Physikmasken und Trefferzahlen als Netzeingabe verwendet, erreichen die Netze $(75 \pm 8)\%$ Physikeffizienz bei 0,4% Rauschen. Der Abfall der Physikeffizienz ist




(a) (P&T) (b) Padmuster als Eingabe (c) (PU22) mit und ohne KNN

**Abbildung 7.11:** Verhalten der Physikeffizienz bei 100:1 Reduktion der Untergrundrate, bei Rauschpegeln zwischen 0,4% und 20%, dargestellt für verschiedene Methoden der 4:1 Datenverdichtung

bei der geforderten Untergrundeffizienz von 99% sehr steil, so daß der nach Gleichung (3.22) abgeschätzte Fehler sehr deutlich ist.

Erheblich höhere Effizienzen werden durch die ($\phi/4$) und ($\sum \phi$) Kodierung der Padmuster erreicht. Bei verringerter Auflösung in $\phi$ wird $(100,00 - 0,05)$% und bei Summation über alle Sektoren $(98,72 \pm 0,11)$% Physikeffizienz für 0,4% Rauschpegel erreicht, wobei ($\sum \phi$) bei extrem hohem Rauschen relativ effizienter ist. Zum Vergleich ist in Abbildung 7.11(b) der Verlauf der Physikeffizienz bei voller Granularität aufgetragen.

Bei Benutzung der ausgewählten Untergrundmasken (PU22), erreichen die Netze wiederum 100% Physikeffizienz bei 0,4% Rauschen und zeigen eine Rauschstabilität, die sogar besser ist, als bei Netzen mit dem kompletten Padmuster als Eingabegröße. Zum Vergleich wurde die Klassifikation anhand der (PU22) Daten ohne KNN geprüft ($\neg$NN). Dazu wurde jedes Ereignis, bei dem mindestens eine Untergrundmaske gefunden wurde als Untergrund klassifiziert, Ereignisse mit Physik- und ohne Untergrundmaske als Physik. Bis zu einem Rauschpegel von 12% ist die Effizienz identisch der von den KNN gelieferten. Dies zeigt, daß die Nutzung der (PU22) Daten für die Klassifikation bis 12,0% Rauschpegel trivial ist.

Bei 16,0% und 20,0% Rauschen läßt sich dagegen die geforderte 100:1 Reduktion nur durch komplettes Verwerfen aller Ereignisse erreichen (Physikeffizienz 0%). Die Leistungsfähigkeit der KNN bei höchstem Rauschen, wird als die Fähigkeit der KNN interpretiert, mit der Unsicherheit der einzelnen Masken umzugehen (Fehlertoleranz).

| Modus | 4:1 Reduktion | benutzt Masken | benutzt KNN |
|---|---|---|---|
| P&T | $\surd$ | $\surd$ | $\surd$ |
| $(\phi/4)$ | $\surd$ | | $\surd$ |
| $(\sum\phi)$ | $\surd$ | | $\surd$ |
| Voll | | | $\surd$ |
| PU22 | $\surd$ | $\surd$ | $\surd$ |
| $\neg$NN | $\surd$ | $\surd$ | |

**Tabelle 7.6:** Eigenschaften der Modi

| | Rauschpegel der Testdaten | | | | | |
|---|---|---|---|---|---|---|
| Modus | 0,4% | 4,0% | 8,0% | 12,0% | 16,0% | 20,0% |
| P&T | $75 \pm 8$ | $52 \pm 4$ | $25 \pm 2$ | $12,8 \pm 0,9$ | $6,1 \pm 0,5$ | $2,1 \pm 0,3$ |
| $(\phi/4)$ | $100,00 - 0,05$ | $53 \pm 3$ | $29 \pm 2$ | $23 \pm 2$ | $13,7 \pm 1,0$ | $10,5 \pm 0,8$ |
| $(\sum\phi)$ | $98,72 \pm 0,11$ | $45 \pm 3$ | $24 \pm 2$ | $26 \pm 2$ | $20 \pm 1$ | $16,4 \pm 1,1$ |
| Voll | $100,00 - 0,02$ | $98,22 \pm 0,11$ | $84 \pm 5$ | $63 \pm 4$ | $41 \pm 3$ | $29 \pm 2$ |
| PU22 | $100,00 - 0,06$ | $99,29 \pm 0,11$ | $95 \pm 6$ | $85 \pm 5$ | $65 \pm 4$ | $39 \pm 2$ |
| $\neg$NN | $100,00 - 0,06$ | $99,37 \pm 0,11$ | $95 \pm 6$ | $85 \pm 5$ | $0,0 + 0,4$ | $0,0 + 0,2$ |

**Tabelle 7.7:** Physikeffizienzen bei 100:1 Reduktion der Untergrundrate, bei verschiedenem Rauschpegel der Testdaten

# 8. Zusammenfassung

In dieser Arbeit wurde die Möglichkeit diskutiert, die Triggerinformation des neuen rückwärtigen Si–Minivertex–Detektors BST im H1–Experiment bei HERA zur Filterung protoneninduzierten Untergrundes zu nutzen. Untersucht wurde die Leistungsfähigkeit künstlicher neuronaler Netze der zweiten Triggerstufe für diese Aufgabe. Die benötigten Daten für Training und Test der Netze wurden durch Simulationsrechnungen generiert, das benutzte Programm wurde im Rahmen der Arbeit entwickelt.

Es konnte gezeigt werden, daß die Netze bei Verwendung der planmäßig für die zweite Triggerstufe verfügbaren Triggerinformation (Physikmasken und Trefferzahl), eine Physikeffizienz von $(75 \pm 8)\%$ bei 100 :1 Reduktion der Untergrundrate erreichen. Dabei wurde ein Rauschpegel der Pads des BST von 0,4% zugrundegelegt [44].

Es wurden alternative Wege der geforderten 4 :1 Datenverdichtung hinsichtlich ihrer Eignung als Netzeingabegrößen untersucht. Die günstigste Methode ohne Verwendung von Masken, beruht auf einer verringerten azimutalen Auflösung ($\phi/4$). Mit dieser Datenverdichtung erreichen die Netze $(100,00 - 0.05)\%$ Physikeffizienz. Dies verdeutlicht, daß die Kodierung der Padmuster durch Physikmasken und Trefferzahlen einen wesentlichen Informationsverlust bedeutet.

Die vorgestellte Idee einer Maskierung mit Physik- *und* Untergrundmasken erwies sich als günstigste Datenverdichtung, legt man die Physikeffizienz bei Rauschpegeln zwischen 0,4% und 20% zugrunde. Es konnte gezeigt werden, daß bei Verwendung von Untergrundmasken die Kodierung der Trefferzahl unnötig wird, und daß bereits die Verwendung relativ weniger (22) Untergrundmasken ausreichend ist. Dieser Weg der Datenverdichtung ist so vorteilhaft daß bis zu einem Rauschpegel von 12% die Verwendung der KNN keinen Vorteil gegenüber einfachem Auswerten der Masken bietet. Bei extremen Rauschpegeln von 16% und 20% versagt das einfache Auswerten, die KNN erweisen sich in diesem Bereich als genügend fehlertolerant, um eine Effizienz von $(65 \pm 4)\%$ bzw. $(39 \pm 2)\%$ aufrechtzuerhalten.

Die vorgestellten Ergebnisse führen zu der Empfehlung, den BST-T als Untergrundfilter zu nutzen und den Maskensatz durch Untergrundmasken zu ergänzen.

# Literaturverzeichnis


[1] E. Rutherford, *The collected papers*, Hg. J. Chadwick, London (1965)

[2] Ch. Gerthsen, H. Vogel, *Physik*, 17. Auflage, Springer-Verlag, S. 690 (1993)8

[3] M. Gell–Mann, Phys. Lett. **8**, S. 214, (1964)

[4] P. Schmüser, D. Proch, *Der Einsatz von Supraleitung bei der neuen Speicherringanlage HERA*, Herausgeber: Arbeitsgemeinschaft d. Großforschungseinrichtungen, ISSN 0935–2236, Bonn (1990)

[5] H1 Collaboration, *The H1 Detector at HERA*, Interner Bericht, DESY H1-96-01 (März 1996)

[6] D.H. Perkins, *Hochenergiephysik*, Addison-Wesley (Deutschland) GmbH, (1991)

[7] P. Waloschek, Das Forschungszentrum DESY, Praxis d. Naturwiss. Phys. 7/38, (1989)

[8] S.L. Glashow, *Particle-Symmetries of Weak Interactions*, Nucl. Phys. **22**, S. 579 (1961); *Towards a Unified Theory: Threads in a Tapestry*, Rev. Mod. Phys. **52**, S. 539 (1980)

[9] A. Salam, *Renormalizability of Gauge Theories*, Phys. Rev. **127**, S. 331 (1962); *Gauge Unification of Fundamental Forces*, Rev. Mod. Phys. **52**, S. 525 (1980)

[10] S. Weinberg, *A Model of Leptons*, Phys. Rev. Lett. **19**, S. 1264 (1967); *Conceptual Foundations of the Unified Theory of Weak and Elektromagnetic Interactions*, Rev. Mod. Phys. **52**, S. 515 (1980)

[11] J.D. Bjorken, E.A. Paschos, Phys. Rev. **179**, S. 1547 (1969)

[12] R.P. Feynman, Acta Phys. Pol. **24**, S. 697 (1963); Phys. Rev. Lett. **23**, S. 1415 (1969)

[13] F. Jacquet, A. Blondel, *Proceedings of the study of the ep facility for Europe 79/48*, U. Amaldi ed., S. 391 (1979)

[14] H1 Kollaboration, *Technical Proposal to build silicon tracking detectors for H1*, interner Bericht, DESY PRC 92/01 (1992)

[15] V.N. Gribov, L.N. Lipatov, Sov. J. Nucl. Phys. *15*, S. 438 (1972)

[16] G. Altarelli, G. Parisi, Nucl. Phys. *126*, S. 297 (1977)

[17] H1 Kollaboration, *Technical proposal to upgrade the backward scattering region of the H1 detector*, interner Bericht, DESY PRC 93/02 (1993)

[18] W.R. Leo, *Techniques for Nuclear and Particle Physics Experiments*, zweite Auflage, Springer–Verlag, (1994)

[19] H1 Kalorimetergruppe, *The H1 Liquid Argon Calorimeter System*, Nucl. Instr. Meth. **A 336**, S. 460 (1993)

[20] M. Ruffer, *Implementierung des siliziuminstrumentierten PLUG–Kalorimeters in den H1–Detektor*, Dissertation, Univ. Hamburg (1992)

[21] H1 SPACAL Gruppe, *Performance of an Electromagnetic Lead/Scintillating-Fibre Calorimeter for the H1 Detector*, DESY 95-250 (1995)

[22] H1 SPACAL Gruppe, *Hadronic Response and $e/\pi$ Separation with the H1 Lead/Fibre Calorimeter*, DESY 95-165 (1995)

[23] J. Bürger et al., *The Readout System of the New H1 Silicon Detectors*, DESY 95/049 (1995)

[24] *Status Report on H1 Silicon Trackers*, interner Bericht, H1-10/93-324 (1993)

[25] H. Kuchling, Taschenbuch der Physik, Fachbuchverlag Leipzig, (1991)

[26] D. Handschuh, *Studien zum Protoneninduzierten Untergrund am H1 Detektor und Entwicklung von Filteralgorithmen zu dessen Unterdrückung*, Dissertation, Univ. Hamburg (1993)

[27] J.C. Bizot et al., *Status of simulation for a topological level 2 trigger*, H1-Note H1-02/92-212, DESY, (1992)

[28] A. Gruber, *Neuronale Netze als L2–Trigger im H1–Experiment*, Diplomarbeit, LMU München (1992)

[29] J. Möck, *Einsatz neuronaler Netze als intelligente Trigger im H1–Experiment*, Diplomarbeit, TU München, (1994)

[30] J. Fent et al., *A neural network second level trigger for the H1–Experiment at HERA*, interner Bericht, H1-10/95-457 (1995)

[31] E. Barrelet et al., *The hardware implementation of L3 triggers in H1*, interner Bericht, H1-12/88-100, DESY (1988)

[32] A. Campbell, *A RISC multiprozessor event trigger for the data aquisition system of the H1 experiment at HERA*, Int. Conf. Real Time '91, Jülich (1991)

[33] D. Westner, *Ein neuronaler Netzwerktrigger für die Produktion von $J/\psi$– Teilchen in ep–Streuung ($J/\psi \to e^+e^-$)*, Diplomarbeit, LMU München (März 1996)

[34] H. Getta, *Untersuchungen eines auf neuronalen Netzen basierenden Triggers für Jet-Ereignisse im H1–Experiment*, Diplomarbeit, Univ. Dortmund, (1995)

[35] T. Zobel, *Neuronale Netzwerke als Trigger für die $J/\psi$-Meson-Produktion in ep–Kollisionen am Beispiel des Zerfalles $J/\psi \rightarrow \mu^+\mu^-$*, Diplomarbeit, LMU München (März 1996)

[36] P. Ribarics, *L2-L3 Background Rejector Based on the Encapsulation of Background Clusters*, interner Bericht H1-08/93-310 (1993)

[37] M. Schreckenberg, *Testmessungen an Prototypen der H1–$\phi$–Streifen– Silizium–Detektoren*, Diplomarbeit, Univ. Hamburg (1995)

[38] H.W. Kraner, Nucl. Instr. and Meth. in Phys. Res. **A225**, S. 615 (1984); IEEE Trans. Nucl. Sci. NS-**29**, No. 3, S. 1088 (1982)

[39] D. Pitzl et al., Nucl. Phys. B. (Proc. Suppl.) **23A**, S. 340 (1991); H. Ziock et al., IEEE Trans. Nucl. Sci. NS-**38**, S. 269 (1991)

[40] N. Wulff, Design studies for the BST $\phi$-strip detector, interner Bericht, H1-06/92-227 (1992)

[41] N. Wulff, Energy measurement with the BST, interner Bericht, H1-09/91-191 (1991)

[42] J. Bürger et al., *Design of a Silicon Backward Tracking Detector and Trigger for the H1 Experiment at the ep Collider HERA*, DESY 93/002 (1993)

[43] M. Klein, schriftliche Mitteilung vom 14. Nov. 1995

[44] H. Henschel, H.H. Kaufmann, M. Klein, W. Lange, J. Meißner, *A Silicon Pad Detector L1 Trigger for Deep Inelastic Scattering at Low $x$ and $Q^2$*, interner Bericht, H1 04/92-221 (1992)

[45] *Selected writings of Sir Charles Sherrington*, London (1939)

[46] E.D. Adrian, *The basis of sensation* (1928); *The mechanism of nervous action* (1932); *The physical background of perception* (1947)

[47] J.C. Eccles, *The neurophysiological basis of mind* (1953); *The physiology of synapses* (1964)

[48] W.S. McCulloch, W. Pitts, *A Logical Calculus of the Ideas Immanent in Nervous Activity*, Bull. Math. Biophys. **5**, (1943)

[49] R. Wehner, W. Gehring, *Zoologie*, 23. Auflage, Thieme Verlag, (1995)

[50] B. Müller, J. Rheinhardt, M.T. Strickland, *Physics of Neural Networks*, 2. Auflage, Springer Verlag (1995)

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

[52] A.K. Kolmogorov, *On the Representation of Continuous Functions of Several Variables by Superposition of One Variable and Addition*, Dokl. Akad. Nauk. **114**, S. 953 (1957)

[53] R.J.P. de Figueiredo, *Implications and Applications of Kolmogorov's Superposition Theorem*, J. MAth. Anal. **38**, S.1227 (1980)

[54] R. Hecht–Nielsen, *Kolmogorov's Mapping Neural Network Existence Theorem*, IEEE First Annual Int. Cponf. on Neural Networks, Band III (1987)

[55] K. Hornik, M. Stinchcomb, H. White, *Multilayer FeedForward Networks are Universal Approximators*, Neural Networks **2**, S.359 (1989)

[56] G. Cybenko, *Approximation by superpositions of sigmoidal functions*, Math. of Control, Signals, and Systems **2**, S.303 (1989)

[57] M.D. Richard, R.P. Lippman, *Neural Network classifiers estimate bayesian a posteriori probabilities*, Neural Computation **3**, S.461 (1991)

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

[59] D.E. Rumelhart, G.E. Hinton, R.J. Williams, *Learning Representations by Back–propagating Errors*, Nature **323**, S. 533 (1986)

[60] CNAPS Back-Propagation 2.0 Release Notes, Adaptive Solutions, Inc. , Beaverton Or. (1993)

[61] L.F.A. Wessels, E. Barnard, *Avoiding False Local Minima by Proper Initialization of Connections*, IEEE Transactions on Neural Networks 3 (6), S. 899 (1992)

[62] D. Hammerstrom, *The CNAPS Hardware Architecture.* Vortrag am DESY, Hamburg (1995)

[63] J. Möck et al., Artificial Neural Networks as a Second-Level Trigger at the H1 Experiment, In. Conference of Artifical Neural Networks, AIHENP (1995)

[64] T. Krämerkämper, *The H1 Level-2 Triggersystem Using Artificial Neural Networks — Online Software Documentation*, in Vorbereitung

[65] H. Henschel, Schriftliche Mitteilung vom 25. Okt. 1995

[66] S. Udluft, *Training und Einsatz neuronaler Netze auf dem CNAPS–Parallelcomputer*, Fortgeschrittenen Praktikum, LMU München (1994)