Untersuchungen zur Verbesserung des neuronalen Triggers beim H1-Experiment durch Datenaufbereitung mit Hilfe topologischer Korrelationen

Diplomarbeit an der Fakultät für Physik
der
Technischen Universität München

vorgelegt von
Sebastian Schmidt

München
April 2000

Erstkorrektor: Prof. Dr. Gerd Buschhorn
Zweitkorrektor:

Inhaltsverzeichnis

Einleitung

Wenn man spezielle Elektron-Proton-Reaktionen mit kleinem Wirkungsquerschnitt an einem Beschleunigerring wie HERA untersucht, ist man darauf angewiesen, die Ereignisse, bei denen diese Reaktionen stattfinden, schnell und trotzdem mit hoher Wahrscheinlichkeit zu erkennen. Es ist nicht zielführend, alle im Detektor gesammelten Daten für jede mögliche Reaktion auszuwerten (oder auch nur aus dem Detektor auszulesen), da dafür jedesmal Zeit benötigt wird, in der der Detektor nicht für die Aufzeichnung von Reaktionen zur Verfügung steht. Auch kann nur eine begrenzte Anzahl von Ereignissen pro Zeit zur weiteren Analyse auf Magnetbändern gespeichert werden.

Man benötigt deshalb einen Mechanismus, der entscheidet, ob erstens überhaupt ein Ereignis im Detektor stattgefunden hat, ob zweitens das Ereignis durch eine Kollision eines Elektrons mit einem Proton ausgelöst wurde (also kein Untergrund ist) und ob es drittens durch eine Reaktion erzeugt wurde, für die man sich interessiert. Ein solcher Mechanismus wird Trigger (engl. für Auslöser) genannt.

Beim H1-Detektor am Beschleunigerring HERA wird ein mehrstufiges Triggersystem benutzt. Die erste Stufe trifft innerhalb von $2,3\mu s$ die Entscheidung, ob der Detektor angehalten wird. Wenn dies der Fall ist, tritt die zweite Triggerstufe in Aktion: Ihr stehen im Gegensatz zur ersten Triggerstufe die Triggerinformationen aller Detektorkomponenten gemeinsam zur Verfügung. Die zweite Triggerstufe besteht bei H1 aus zwei parallel arbeitenden Systemen. Eines davon ist der neuronale Trigger (L2NN), der künstliche neuronale Netze auf die Triggerinformationen anwendet und anhand von Korrelationen zwischen diesen Informationen seine Entscheidung fällt. Er hat dafür $20\mu s$ Zeit. Wenn ein Ereignis die zweite Triggerstufe passiert hat, wird auf der nächsten Stufe endgültig entschieden, ob das Ereignis auf den Magnetbändern gespeichert wird.

Vom Herbst 2000 an wird bei HERA und damit auch bei H1 ein Upgrade durchgeführt. Durch eine verbesserte Strahlführung wird die Dichte der Elektronen und Protonen am Wechselwirkungspunkt erhöht. Es werden dann etwa fünfmal mehr Elektron-Proton-Reaktionen auftreten als jetzt. Im Rahmen dieses Upgrades werden auch an verschiedenen Teilen des H1-Detektors Verbesserungen durchgeführt und teilweise sogar neu entwickelte Detektorkomponenten eingesetzt. Da sich mit diesem Upgrade zwar die Rate der Elektron-Proton-Reaktionen erhöht, aber die Rate, mit der die Ereignisse auf Magnetband gespeichert werden können, gleich bleibt, wird der Bruchteil der aufgezeichneten Ereignisse kleiner. Daraus ergibt sich, dass die Auswahl interessanter Ereignisse besser sein muss, um voll von der Erhöhung der Reaktions-Rate profitieren zu können. Deshalb werden auch die Trigger einem Upgrade unterzogen. Ein Teil dieses Upgrades besteht darin, den neuronalen Trigger bei H1 mit einer verbesserten Datenaufbereitung auszustatten. Diese Datenaufbereitung orientiert sich an topologischen Korrelationen zwischen den von den Detektorkomponenten bereitgestellten Triggerinformationen. Es werden beispielsweise Signale in verschiedenen Detektorkomponenten kombiniert, die sich, vom Wechselwirkungspunkt aus gesehen, in der gleichen Richtung befinden. Damit wird es möglich, Signalen im Detektor bereits auf Triggerebene physikalischen Teilchen (oder auch Jets) zuzuordnen und Triggerentscheidungen von den Eigenschaften einzelner solcher Teilchen oder sogar vom Verhältnis mehrerer Teilchen zueinander (hier sei als Beispiel nur der Winkel zwischen den Teilchen erwähnt) abhängig zu machen.

Die Hardwarekomponente, mit der diese verbesserte Datenaufbereitung realisiert werden soll, befindet sich noch in der Planungsphase. Sie wird als DDB2 bezeichnet. Die voliegende Arbeit beschäftigt sich mit der Entwicklung von Konzepten für die DDB2 und deren Umsetzung in einer Simulation. Besonders wichtig ist dabei die Realisierbarkeit der Konzepte in der Hardware, zum Beispiel in Hinblick auf die von den Algorithmen benötigte Zeit.

Zudem wird in dieser Arbeit die Anwendbarkeit der Konzepte in der Praxis überprüft, indem ein neuronales Netz, das unter Verwendung der verbesserten Datenaufbereitung erstellt wurde, mit einem neuronalen Netz verglichen wird, das die bisherige Datenaufbereitung verwendet. Der Vergleich wird für Netze zum Triggern von ϕ-Mesonen in Photoproduktion durchgeführt.

Ebenfalls zum Triggern von ϕ-Mesonen in Photoproduktion dient ein weiteres neuronales Netz, das zwar teilweise die Konzepte der „intelligenten“ Datenaufbereitung nutzt, aber auch ohne Erweiterungen der Hardware einsetzbar ist. Mit den Daten, die von diesem Netz gesammelt wurden, wird der proton-dissoziative Wirkungsquerschnitt von ϕ-Mesonen in Photoproduktion untersucht.

In den ersten beiden Kapiteln werden zunächst die wesentlichen Komponenten des H1-Detektors (Kapitel 1) und des L2-Triggers (Kapitel 2) beschrieben. Die Konzepte der „intelligenten“ Datenaufbereitung und deren Umsetzung in die Simulation werden in Kapitel 3 diskutiert. In Kapitel 4 wird das neuronale Netz zum Triggern von ϕ-Mesonen in Photoproduktion, das „intelligente“ Datenaufbereitung nutzt, mit den bisher für den neuronalen Trigger erstellten Netzen für die gleiche Reaktion verglichen. Anschließend (Kapitel 5) wird das neuronale Netz vorgestellt, das zwar die Prinzipien, auf denen die „intelligente“ Datenaufbereitung beruht, verwendet, aber bereits mit der jetzt verfügbaren Hardware eingesetzt werden kann. Im selben Kapitel werden physikalische Ergebnisse, die aus den von diesem Netz gesammelten Daten gewonnen wurden, präsentiert.

Die Arbeit schließt mit der Dokumentation zur Software, die in ihrem Rahmen erstellt wurde (Anhang A).

Kapitel 1

H1-Detektor

Dieses Kapitel stellt zunächst die Hadron-Elektron-Ring-Anlage (HERA) vor, an der der H1-Detektor betrieben wird. Dann folgt eine Beschreibung der einzelnen Subdetektoren, d.h. der verschiedenen Teile des Detektors. Besonderes Augenmerk wird dabei auf diejenigen Komponenten und Eigenschaften gelegt, die im späteren Verlauf der Arbeit im Triggerkonzept des neuronalen Triggers eine Rolle spielen.

1.1 HERA

HERA ist ein Speicherring, in dem Elektronen oder Positronen mit einer Energie von 27,5 GeV und Protonen mit einer Energie von 920 GeV zur Kollision gebracht werden (s. Abbildung 1.1). Die Schwerpunktsenergie beträgt damit $\sqrt{s} = 318$ GeV. Im Folgenden wird nur noch von Elektronen gesprochen, da in dieser Arbeit der Unterschied zwischen Elektronen und Positronen keine Rolle spielt.

Die Elektronen und Protonen sind in Bündeln, sogenannten *bunches* mit jeweils n_e bzw. n_p Teilchen angeordnet. Wenn ein Elektron- und ein Proton-*bunch* aufeinander treffen, spricht man von einem *bunch crossing*. Der zeitliche Abstand zwischen zwei *bunch crossings* beträgt bei HERA 96 ns.

Die Anzahl ΔN der Ereignisse pro Zeiteinheit Δt, bei denen eine bestimmte Reaktion zwischen dem Elektron und dem Proton auftritt, hängt von der Luminosität L und dem Wirkungsquerschnitt σ der Reaktion ab:

$$\Delta N = \sigma L \Delta t. \tag{1.1}$$

Die Luminosität ist eine Eigenschaft des Speicherrings. Sie bestimmt sich aus der Strahlfläche A, der Bündeldurchdringungsfrequenz f und der Zahl von Elektronen und Protonen pro Bündel wie folgt:

$$L = f \frac{n_e n_p}{A}. \tag{1.2}$$

Zur Bestimmung des (physikalisch interessanten) Wirkungsquerschnitts σ muss die Luminosität genau bekannt sein. Auch ist eine möglichst hohe integrierte Luminosität erforderlich, damit auch Reaktionen mit kleinem Wirkungsquerschnitt mit ausreichender Statistik betrachtet werden können.

An HERA werden vier verschiedene Experimente betrieben: H1 und ZEUS betrachten Elektron-Proton-Kollisionen bei hohen Schwerpunktsenergien, HERMES dient der Bestimmung der spinabhängigen Strukturfunktion mit Hilfe des polarisierten Elektronstrahls, der auf ein polarisiertes Gas-Target gelenkt wird und bei HERA-B werden Zerfälle von B-Mesonen durch Reaktionen von Protonen aus dem Halo des Strahls mit Draht-Targets untersucht.

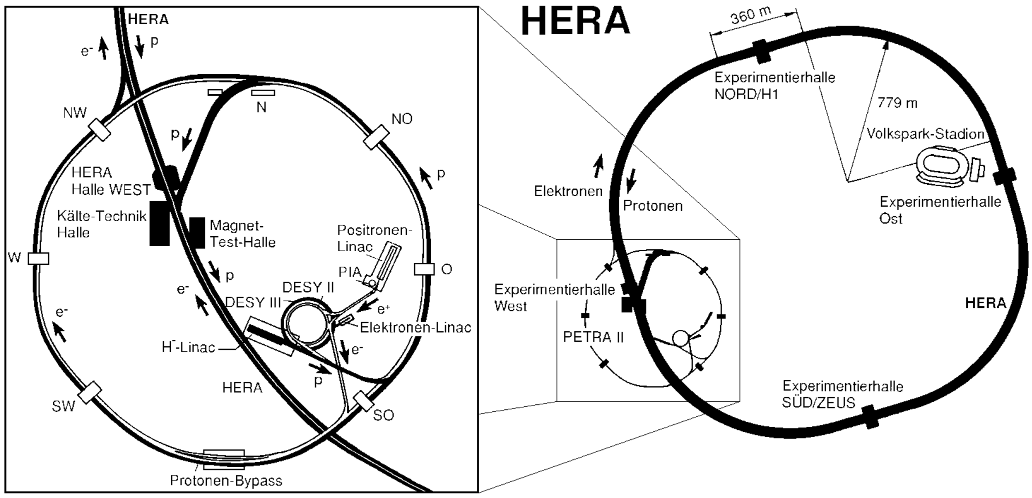


Abbildung 1.1: HERA

1.2 H1

In Abbildung 1.2 ist der Aufbau des Detektors schematisch dargestellt.

Der gesamte innere Teil des Detektors befindet sich in einem homogenen, parallel zum Strahlrohr ausgerichteten Magnetfeld von 1,2 T, das von einer supraleitenden Spule $\boxed{6}$ erzeugt wird. Dadurch werden geladene Teilchen im Detektor abgelenkt, was zur Messung ihres Impulses genutzt wird.

Der Ursprung des H1-Koordinatensystems liegt im nominellen Wechselwirkungspunkt, dem Punkt, an dem normalerweise die *bunches* aufeinandertreffen. In Abbildung 1.2 ist er durch ein Kreuz gekennzeichnet. Die z-Achse verläuft in Richtung Protonstrahl von rechts nach links, die y-Achse nach oben und die x-Achse nach vorne. Weiter wird ein den Symmetrien angepasstes Polarkoordinatensystem mit den Winkeln θ und ϕ sowie dem Radius r eingeführt. θ ist bezüglich der Protonflugrichtung definiert, ϕ und r wie üblich in der xy-Ebene.

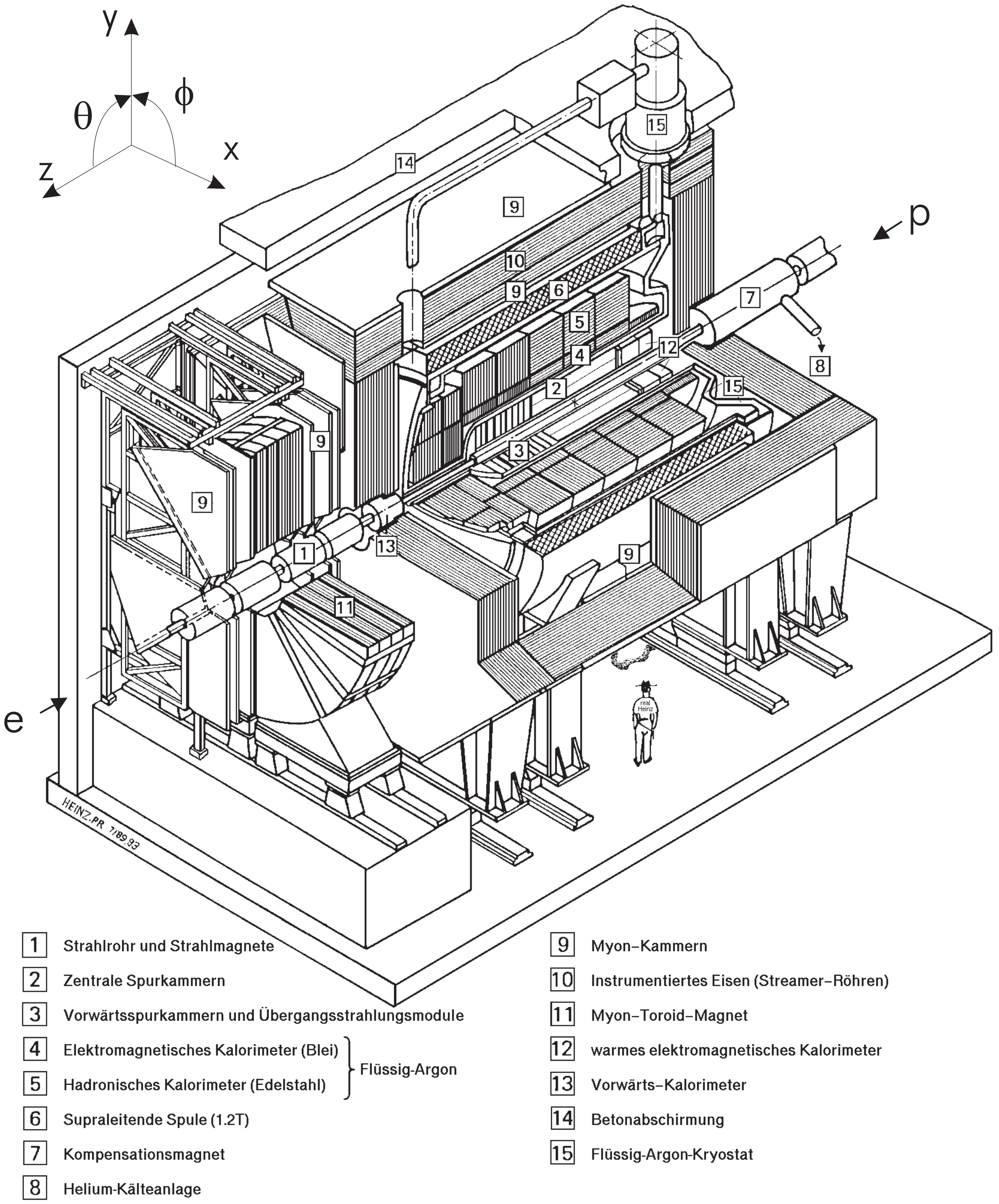


Abbildung 1.2: Schematische Darstellung des H1-Detektors. In der Darstellung fehlen die im Laufe der Zeit nachgerüsteten Komponenten: das SpaCal, die *backward drift chambers* (BDC), das VLQ sowie die Siliziumtracker (CST und BST). Die Protonen durchlaufen den Detektor von rechts nach links, die Elektronen von links nach rechts.

Im Folgenden werden die Detektorkomponenten genauer beschrieben, die für diese Arbeit relevant sind. Für Informationen den gesamten Detektor oder nicht erwähnte Subdetektoren betreffend, sei auf [1] und [2] verwiesen.

1.3 Spurkammern

Aufgabe der Spurkammern ist die Bestimmung von Richtung und Impuls geladener Teilchen. Außerdem dienen sie zur Festlegung von Ereignistopologie und Position des Wechselwirkungspunktes (Vertex).

Abbildung 1.3 zeigt den Aufbau des Spurkammersystems.

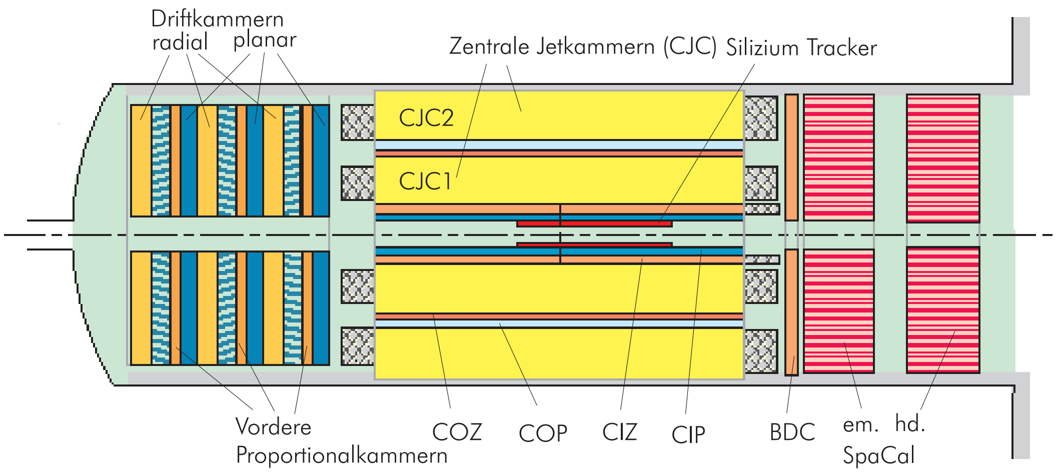


Abbildung 1.3: Spurkammersystem bei H1, Seitenansicht

Die Zentralen Spurkammern $\boxed{2}$ decken einen Bereich von $20° < \theta < 160°$ ab. Zur Rekonstruktion der Spuren werden vier Kammern verwendet. Mit den Zentralen Jetkammern (*Central Jet Chambers*, **CJC1** und **CJC2**) wird die Projektion der Spuren in die ϕ-r-Ebene bestimmt. Die Signaldrähte verlaufen in Richtung z-Achse. Sie sind so in Reihen angeordnet, dass 30 (in der CJC1) oder 60 Zellen (in der CJC2) entstehen. Diese Reihen sind gegenüber der radialen Richtung um 30° geneigt. Dies vermeidet Mehrdeutigkeiten bei der Spurbestimmung, indem Spursegmente in verschiedenen Reihen miteinander kombiniert werden können.

Zur Bestimmung der z-Koordinate der Spur dienen die Z-Kammern (*central inner z-chamber*, **CIZ** und *central outer z-chamber*, **COZ**). Die Signaldrähte verlaufen hier jeweils in vier Lagen ringförmig um die Strahlachse. Die CIZ ist in 15 Segmente aufgeteilt, die COZ in 24. Einen Radialschnitt dieser Komponenten zusammen mit den Zentralen Proportionalkammern findet man in Abbildung 1.4.

Der Winkelbereich $5° < \theta < 25°$ wird durch die vorderen Spurkammern $\boxed{3}$, der Bereich $151° < \theta < 177,5°$ durch die rückwärtige Driftkammer (*Backward Drift Chamber*, **BDC**) abgedeckt (vgl. auch Abbildung 1.7).

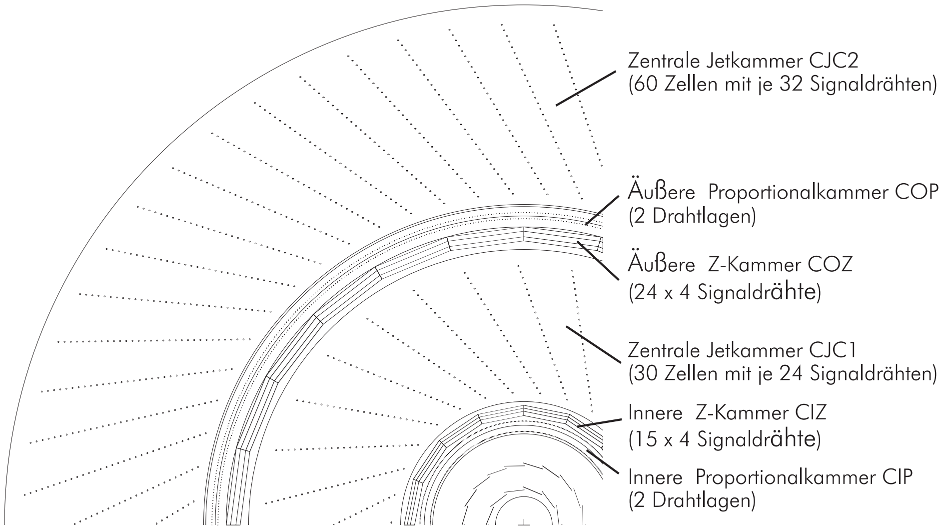


Abbildung 1.4: Zentrales Spurkammersystem bei H1, Radialschnitt

1.4 Proportionalkammern

An der Innenseite der CIZ bzw. der Außenseite der COZ befinden sich die Zentralen Proportionalkammern (*central inner proportional chamber*, **CIP** und *central outer proportional chamber*, **COP**). Diese werden hauptsächlich zum Triggern eingesetzt. Beide Kammern bestehen aus einer Doppellage von Drähten. In z-Richtung sind sie 60-fach (CIP) bzw. 18-fach (COP) segmentiert. Die Proportionalkammern werden auch als MWPC (*multi wire proportional chamber*) bezeichnet.

Die MWPCs sind in ϕ-Richtung in 16 Sektoren geteilt. Wenn drei oder vier Pads (Drahtlagensegmente) innerhalb eines Sektors ein Signal aufweisen, bilden sie einen sogenannten *ray*. Die Extrapolation aller *rays* auf die z-Achse wird in das z-Vertex-Histogramm [3] eingetragen (s. Abbildung 1.5). Dieses ist in z-Richtung 16-fach gebinnt. Die Inhalte dieser 16 Bins stehen auf der zweiten Triggerstufe zusammen mit der Position des Bins mit dem maximalen Wert, diesem maximalen Wert und der Summe über die Einträge aller Bins zur Verfügung. Die Position des Maximums kennzeichnet den vermutlichen Wechselwirkungspunkt zwischen Elektron und Proton.

Vor der Zentralen MWPC liegen die Vorderen Proportionalkammern (*forward proportional chambers*, **FPC**). Auch von ihr werden Daten, kombiniert mit denen der CIP, in das z-Vertex-Histogramm eingetragen. Zusammen decken die Proportionalkammern einen Winkelbereich von $7° < \theta < 175°$ ab.

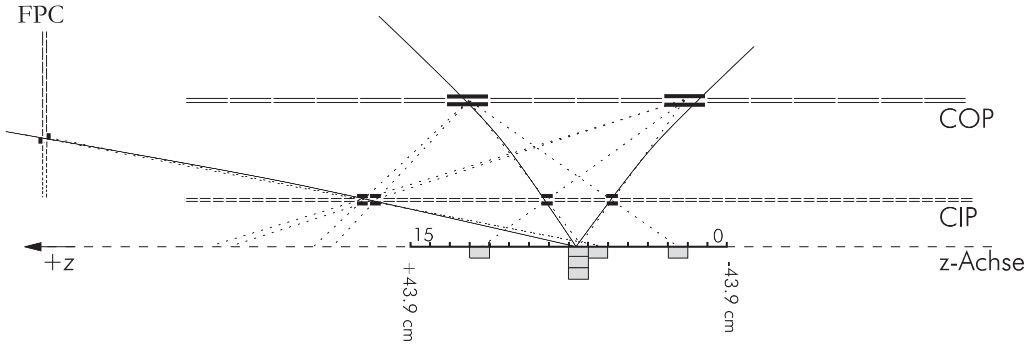


Abbildung 1.5: Bildung des z-Vertex-Histogramms durch Extrapolation der *rays* auf die z-Achse.

1.5 Kalorimeter

Die Kalorimeter dienen einerseits der Bestimmung der Energie der auftretenden Teilchen, andererseits auch der Teilchenidentifikation.

Das **Flüssig-Argon-Kalorimeter** (*Liquid Argon Calorimeter*, LAr) umschließt das Spurkammersystem und deckt den Winkelbereich $4° < \theta < 155°$ ab [4] [5].

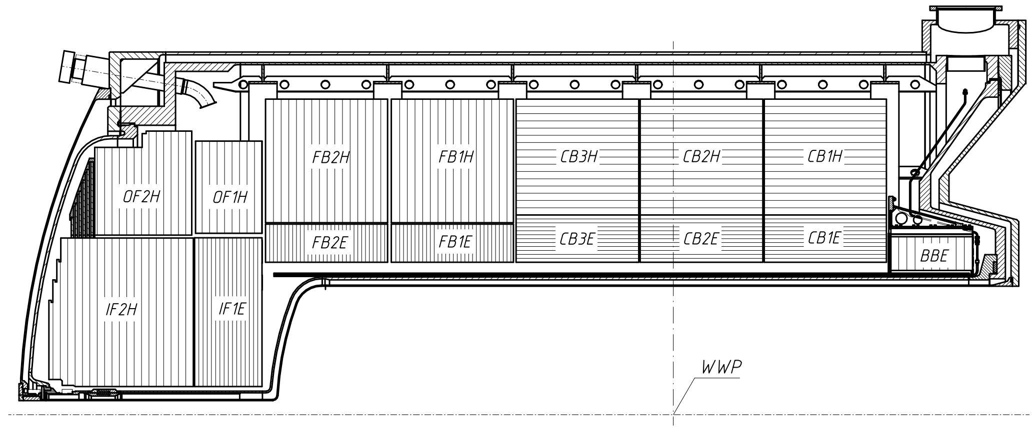


Abbildung 1.6: Flüssig-Argon-Kalorimeter bei H1, Seitenansicht

Es besteht aus schichtweise angeordneten Absorberplatten, zwischen denen sich flüssiges Argon als aktives Ionisierungsmaterial befindet. Aufgebaut ist es aus acht Rädern (*wheels*), die mit *inner forward* (IF), *outer forward* (OF), *forward barrel* (FB), *central barrel* (CB) und *backward barrel* (BB) bezeichnet werden. Die meisten *wheels* bestehen aus einem inneren elektromagnetischen und einem äußeren hadronischen Teil. Im hadronischen Teil werden Absorberplatten aus Edelstahl, im elektromagnetischen solche aus Blei verwendet. Die Lage der einzelnen Komponenten sowie die Orientierung der Platten ist aus Abbildung

1.6 ersichtlich. Insgesamt besteht das Kalorimeter aus rund 44.300 Zellen, davon 32.700 im elektromagnetischen Teil und 11.600 im hadronischen. Je nach Teilchenart unterscheiden sich die Signaturen im hadronischen und elektromagnetischen Teil. Dies wird zur Identifikation der auftretenden Teilchen benutzt. Ein Elektron deponiert beispielsweise relativ viel Energie im elektromagnetischen und relativ wenig im hadronischen Bereich. Sein Cluster ist zudem nur wenig ausgedehnt.

An der Vorderseite des Flüssig-Argon-Kalorimeters schließt sich ein Kupfer-Silizium-Kalorimeter, das ***plug*** an [13]. Es deckt den Winkelbereich $0,6° < \theta < 3,5°$ ab und besteht aus insgesamt 336 Kanälen.

Im rückwärtigen Bereich ($155° < \theta < 177,5°$) wird das sogenannte Spaghetti-Kalorimeter (**SpaCal**) zur Energiemessung eingesetzt (s. Abbildung 1.7).

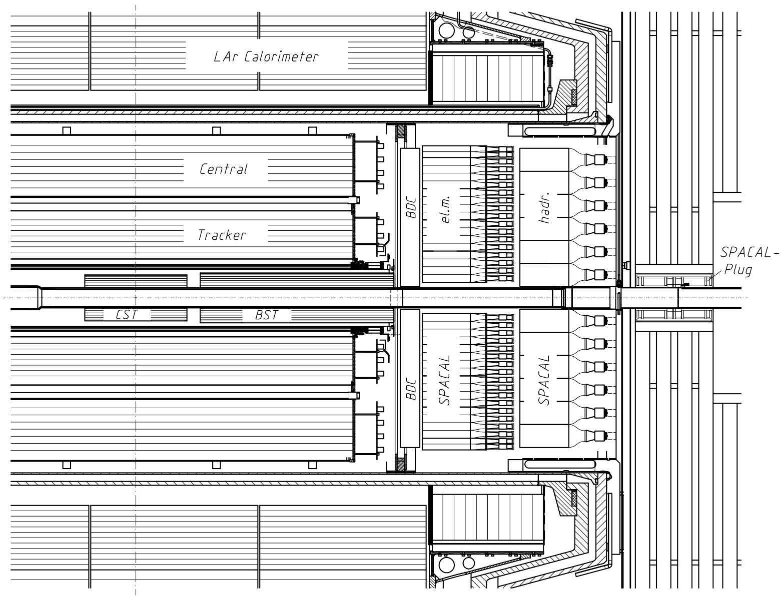


Abbildung 1.7: Darstellung der hinteren Detektorregion mit CST, BST, BDC und SpaCal

Es handelt sich dabei um ein Blei-Szintillations-Kalorimeter, das ebenfalls aus einem elektromagnetischen und einem hadronischen Teil besteht. Die beiden Teile unterscheiden sich im Verhältnis von Absorbermaterial zu Szintillationsfasern sowie in der Zellgröße, zu der die Fasern zusammengefasst werden. Im hadronischen Teil befinden sich 128 Zellen, im elektromagnetischen 1192. Durch ringförmige Szintillatoren direkt am Strahlrohr wird festgestellt, ob die gesamte Energie im SpaCal erfasst wurde. Falls dies nicht der Fall ist, führt dies ebenso zu einem Veto, wie eine durch die Flugzeitfunktion (*Time of Flight System*, ToF) des hadronischen Teils des SpaCals festgestellte Proton-Untergrund-Reaktion: Die Reaktionsprodukte einer Reaktion am nominellen Wechselwirkungspunkt während eines *bunch crossings* müssen das SpaCal innerhalb eines bestimmten Zeitfensters erreichen.

Winkel sehr nahe am rückwärtigen Strahlrohr deckt das **VLQ-Spektrometer** (*very low* Q^2) [4] ab. Sein Aufbau ist aus Abbildung 1.8 ersichtlich. Das VLQ-Spektrometer besteht aus Tracker, Kalorimeter und einem *Time of Flight System* (VLQ-ToF). In dieser Arbeit wird nur das VLQ-Kalorimeter eine Rolle spielen. Es besteht aus zwei Teilen, die symmetrisch zur z-x-Ebene über und unter dem Strahlrohr angeordnet sind, d.h., die Effizienz ist von ϕ abhängig.

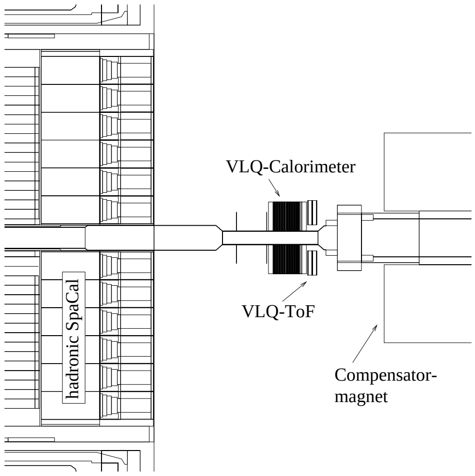


Abbildung 1.8: VLQ-Spektrometer

Zur Beobachtung noch kleinerer Ablenkwinkel von Elektronen und Protonen werden Luminositätsdetektoren (für die Elektronen) und *proton remnant tagger* (für die Protonen) verwendet, die sich in einem Abstand von bis zu 100 m vom Detektor entfernt befinden.

1.6 Myonsysteme

Das Eisenjoch, das den inneren Teil des Detektors umschließt und der Rückführung des magnetischen Flusses dient, ist zusätzlich mit Streamerrohrkammern instrumentiert $\boxed{10}$. Diese werden zur Identifikation von Myonen und zur Bestimmung der Restenergie hadronischer Schauer verwendet. Diese Detektorkomponente wird daher instrumentiertes Eisen (*instrumented iron*) oder **Zentrales Myonsystem** genannt. Der abgedeckte Winkelbereich ist $4° < \theta < 171°$.

Nur für den vorderen Detektorbereich zuständig sind die **Vorderen Myonkammern** $\boxed{9}$. Sie decken den Winkelbereich $3° < \theta < 17°$ ab und bestehen aus zwei Abschnitten, die durch einen Eisen-Toroid-Magneten voneinander getrennt und selbst wiederum aus jeweils drei doppelten Driftkammern aufgebaut sind. Im Toroid-Magneten ist ebenfalls ein Flugzeitsystem (*Forward Time of Flight System*, FToF) implementiert.

1.7 Luminositätsmesssystem

Die Luminosität wird über den in der QED sehr gut verstandenen Bethe-Heitler-Prozess [5] $ep \to ep + \gamma$ bestimmt. Mit Hilfe eines Elektrontaggers (ET) bei $z = -33$ m und eines Photondetektors (PD) bei $z = -103$ m wird ein solches Ereignis festgestellt (vgl. Abbildung 1.9).

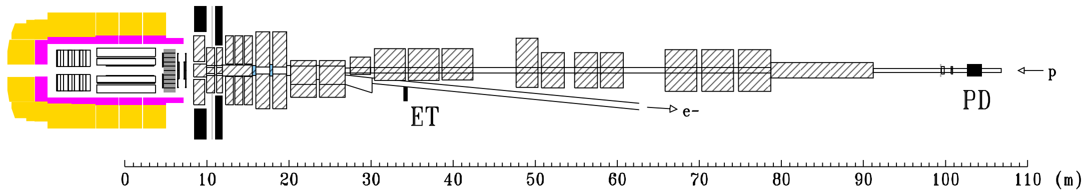


Abbildung 1.9: Schematische Darstellung des Luminositätsmesssystems

Kapitel 2

Trigger bei H1

Dieses Kapitel gibt zunächst einen Überblick über das Triggerkonzept von H1 und geht dann näher auf den neuronalen Trigger ein, mit dem sich diese Arbeit hauptsächlich beschäftigt.

2.1 Gesamtkonzept

Wie bereits erwähnt, beträgt bei HERA die Zeit zwischen zwei *bunch crosses* (BC), d.h. zwischen zwei potentiellen Reaktionen, 96 ns. Dies ergibt eine Frequenz von 10,4 MHz. Die im Detektor beobachtbare Wechselwirkungsrate zwischen den Proton- und Elektronbündeln liegt bei einer Luminosität von 10^{31} cm^{-2}s^{-1} bei etwa 100 Hz. Daneben existieren die nicht erwünschten Reaktionen (Untergrund), deren Rate die der Elektron-Proton-Kollisionen um einige Größenordnungen (etwa 10^3) übertrifft. Die wichtigsten derartigen Reaktionen werden im Folgenden aufgeführt:

- Strahl-Gas-Wechselwirkungen

  Es finden Reaktionen der Protonen mit Kernen der Restgasmoleküle (bei einem Vakuum von circa 10^{-7} Pa) im Strahlrohr statt. Die Untergrundrate von einigen kHz wird durch diese Ereignisse dominiert.

- Strahl-Wand-Wechselwirkungen

  Weiterhin können auch Strahlrohr oder Detektorkomponenten von Protonen aus dem Halo des Strahls getroffen werden. Topologisch entsprechen diese Ereignisse im Wesentlichen den Strahl-Gas-Wechselwirkungen.

- Synchrotronstrahlung

  Eine andere Ursache für Untergrund ist die Synchrotronstrahlung. Sie entsteht durch die Adjustierung des Elektronenstrahls mit Hilfe von Fokussiermagneten. Obwohl sie durch Kollimatoren größtenteils unterdrückt wird, kann die verbleibende Strahlung Signale in verschiedenen Detektorkomponenten auslösen.

- Kosmische Strahlung

  Auch Myonen aus der kosmischen Strahlung führen zu Signalen im Detektor. Die Rate von ca. 1 kHz kann auf einfache Weise durch die Anwendung einer Vertexbedingung reduziert werden. Damit spielt diese Art von Untergrund keine wesentliche Rolle mehr.

Zur Unterscheidung zwischen den interessanten physikalischen und den unerwünschten Ereignissen braucht man einen Trigger (engl. Auslöser), der entscheidet, bei welchen Ereignissen der Detektor ausgelesen wird. Dies hat zwei Gründe: Zum einen benötigt die Auslese des gesamten Detektors eine relativ lange Zeit (etwa 1 ms), in der der Detektor nicht für weitere Ereignisse bereit ist (Totzeit). In dieser Zeit müssen über eine viertel Million Kanäle, insgesamt ca. 3 MB an Rohdaten pro Ereignis, digitalisiert und ausgelesen werden. Zum anderen ist die Rate, mit der die Ereignisse auf Magnetbändern aufgezeichnet werden können, technisch limitiert. Pro Ereignis werden auch mit Nullunterdrückung noch etwa 50 bis 100 kB auf Magnetband weggeschrieben. Das beschränkt die Aufzeichnungsrate auf etwa 10 Hz.

Obwohl ein Trigger also den Untergrund (oder auch nicht interessierende physikalische Ereignisse) ausreichend unterdrücken muss, soll natürlich seine Effizienz in Bezug auf die physikalischen Reaktionen möglichst groß sein.

Um diesen Anforderungen gerecht zu werden, verwendet man bei H1 ein mehrstufiges Triggersystem, durch das die anfängliche Ereignisrate sukzessive auf die erforderliche Rate reduziert wird. Diese mehrstufige Struktur hat den Vorteil, dass Ereignisse, die bereits mit schnell zur Verfügung stehender Information als Untergrund erkannt werden können, die Totzeit des Detektors nicht erhöhen.

Ein Überblick über das Triggersystem wird in Abbildung 2.1 gegeben.

Bei H1 sind drei Triggerstufen implementiert: L1, L2 und L4. L3 wird zurzeit nicht verwendet. Manchmal bezeichnet man die offline stattfindende Ereignisrekonstruktion als L5. Im Folgenden wird genauer auf die verschiedenen Triggerlevel eingegangen.

2.1.1 Erste Triggerstufe L1

Die erste Stufe des Triggersystems bei H1 wird mit L1 („*level 1*“) bezeichnet. Fast alle Subdetektoren stellen dafür Informationen bereit. Dies geschieht in Form von zurzeit 196 Triggerelementen (TE). Beispiele für TE sind überschrittene Energieschwellen in bestimmten Detektorkomponenten oder eine bestimmte Anzahl von Spuren in einer Spurkammer, die einer bestimmten Maske genügen und damit einem bestimmten Impulsbereich entsprechen.

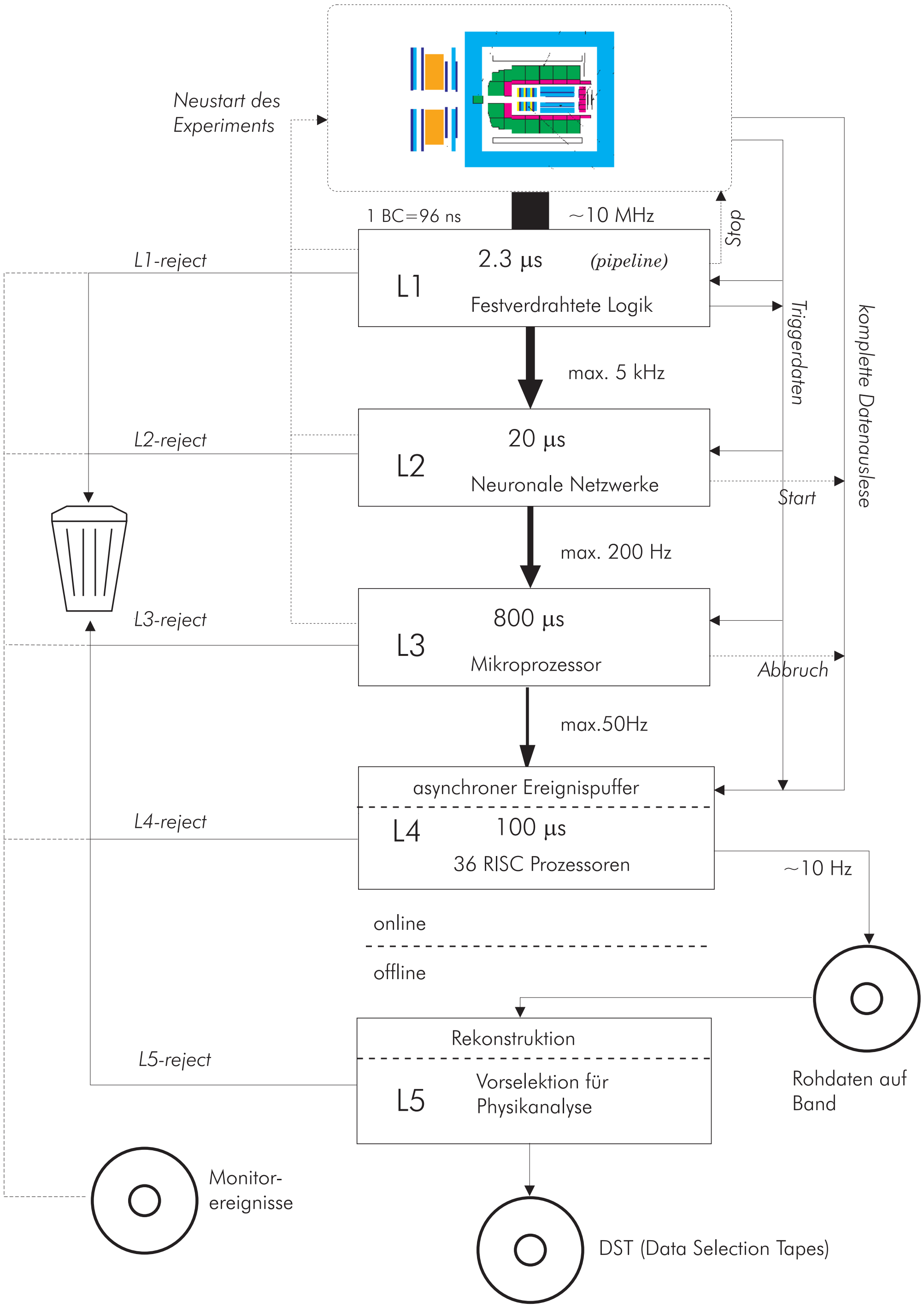


Abbildung 2.1: Schematische Darstellung des H1-Triggersystems

Diese Informationen liegen allerdings teilweise nicht sofort (d.h., innerhalb eines BC) vor: Die Driftzeit der Elektronen in der CJC beträgt bis zu 1 μs (11 BC), die Integration in den Kalorimeter-Vorverstärkern benötigt 1,5 μs (13 BC).

Aus den TE werden durch logische Verknüpfungen bis zu 128 Subtrigger gebildet. Jeder dieser Subtrigger wird speziell für eine physikalische Reaktionsklasse oder einen kinematischen Bereich davon erstellt. Wenn eine dieser Subtriggerbedingungen erfüllt ist, liefert dies das Signal zum Anhalten des Detektors. Dies geschieht mit einer Verzögerung von 24 BC. Während dieser Zeit wird die gesamte Information in einer *pipeline* gespeichert. Dadurch arbeitet H1 bis zu dem Zeitpunkt totzeitfrei, an dem eine positive L1-Triggerentscheidung gefällt wird.

Falls einzelne Subtrigger eine zu hohe Rate liefern, werden sie mit einem *prescale* versehen, d.h. nur jedes n-te Ereignis, das die Triggerbedingung erfüllt, wird auch wirklich getriggert. Dadurch sinkt natürlich die für eine Analyse der entsprechenden Reaktionen zur Verfügung stehende integrierte Luminosität entsprechend.

2.1.2 Zweite Triggerstufe L2

L2 arbeitet im Wesentlichen mit den gleichen Informationen, die bereits auf L1 zugänglich waren. Der Detektor ist angehalten, wird aber noch nicht ausgelesen. Die erlaubte Verarbeitungszeit auf L2 beträgt 20 μs. Durch diese Beschränkung wird die Totzeit des Detektors bei einer L2-Eingangsrate von 1 kHz bei etwa 2 % gehalten.

Zusätzlich zu den trivialen Verknüpfungen der Subtrigger auf L1 können auf L2 komplexere Algorithmen auf die kombinierten Daten aller Subdetektoren angewendet werden. Es wird versucht, topologische Strukturen in den Triggerinformationen auszuwerten. Dazu existieren zwei sich in der verwendeten Hardware und den zugrundeliegenden Algorithmen unterscheidende Ansätze:

- Topologischer Trigger (L2TT)

  Der L2TT wertet die Informationen durch Anwendung von Matrixtransformationen und logischen Verknüpfungen aus [6].

- Neuronaler Netzwerktrigger (L2NN)

  Der L2NN bedient sich sogenannter *feed forward* Netzwerke, die auf schnellen Parallelrechnern ausgewertet werden. Eine eingehende Darstellung dieses Triggers ist in Abschnitt 2.2 zu finden.

Auf L2 werden nur bestimmte L1 Subtrigger verifiziert, die für sich allein eine zu hohe Rate verursachen, beziehungsweise nur mit einem hohen *prescale* betrieben werden könnten. Alle anderen Subtrigger werden immer verifiziert.

2.1.3 Vierte Triggerstufe L4

Für jedes Ereignis, das L2 erfolgreich passiert hat, wird der Detektor vollständig ausgelesen. Damit ist L4 der erste Triggerlevel, auf dem die gesamte Detektorinformation zur Verfügung steht. L4 ist eine Prozessorfarm, bestehend aus 36 Prozessoren. Dort wird eine vorläufige Rekonstruktion der Ereignisse durchgeführt, anhand derer entschieden wird, ob das Ereignis auf Band geschrieben wird. Dafür werden die einzelnen Ereignisse in einem Ringpuffer mit einer Kapazität von 30 Ereignissen gehalten und asynchron auf einem der Prozessoren abgearbeitet. Auch hier kommt wieder das Prinzip der Verifizierung von L1 Subtriggern zum Einsatz.

Außerdem werden bereits auf L4 die verschiedenen Ereignisse in Physikklassen eingeteilt, um einen schnelleren Zugriff auf Ereignisse eines bestimmten Typs zu ermöglichen.

2.1.4 Ereignisrekonstruktion

Alle nach L4 auf Band geschriebenen Ereignisse werden offline nochmals vollständig rekonstruiert. Dabei werden aus den gespeicherten Daten physikalische Objekte gebildet. Es werden beispielsweise einzelne Signale in den Spurkammern zu Spuren oder benachbarte Kalorimeterzellen zu Clustern kombiniert.

Abschließend werden die Ereignisse komprimiert und auf *data summary tapes* (DST) für die Physikanalyse zur Verfügung gestellt.

Auf die Ereignisse kann dann mit Hilfe einer *run* und einer *event* Nummer zugegriffen werden. Ein *run* ist ein Zeitabschnitt, typischerweise zwei Stunden, in dem sichergestellt wird, dass die Detektoreinstellungen unverändert bleiben. Innerhalb eines solchen *run* werden den einzelnen Ereignissen *event* Nummern zugeordnet.

2.2 Neuronaler Trigger auf L2

Wie oben erläutert, werden auf der zweiten Triggerstufe topologische Korrelationen in den Triggerinformationen ausgewertet. Einer der beiden verfügbaren Ansätze dazu besteht in der Anwendung neuronaler Netze auf die Triggerinformation. Dies wird bei H1 vom Neuronalen Netzwerktrigger auf L2 (L2NN) durchgeführt.

In diesem Abschnitt wird zuerst das grundlegende mathematische Modell beschrieben, das neuronalen Netzen zugrunde liegt. Dann wird auf die Hardware-Realisierung des Neuronalen Triggers bei H1 (L2NN) eingegangen. Abschließend wird erläutert, wie diese Hardware durch ein Training an ein bestimmtes physikalisches Problem angepasst wird.

2.2.1 Neuronale Netze

Das Konzept der neuronalen Netze basiert auf dem biologischen Vorbild des Gehirns, das aus vielfach verknüpften Zellen, den Neuronen besteht. Ein einfaches mathematisches Modell für die neuronalen Bausteine wurde in [7] vorgestellt (vgl. Abbildung 2.2).

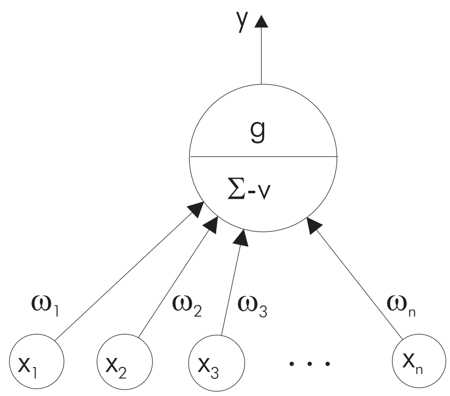


Abbildung 2.2: Schematische Darstellung eines mathematischen Neurons

Der Ausgabewert $y(\vec{x})$ wird durch eine gewichtete Summation über die n Eingabewerte x_i, die Subtraktion einer Schwelle v und die anschließende Anwendung einer nichtlinearen Schwellenfunktion $g(x)$ bestimmt. Die Gewichte der einzelnen Eingabewerte sind ω_i. Sie können „inhibitorisch“ ($\omega_i < 0$) oder „exzitatorisch“ ($\omega_i \geq 0$) wirken:

$$y(x) = g\left(\sum_{i=1}^{n} \omega_i x_i - v\right) \tag{2.1}$$

Als Stufenfunktion wird üblicherweise die Sigmoid-Funktion

$$g(x) = \frac{1}{1+e^{-\beta x}} \quad \text{mit} \quad \beta > 0 \tag{2.2}$$

verwendet (vgl. Abbildung 2.3), da diese im Gegensatz zur Θ-Funktion differenzierbar ist. Dass dies eine notwendige Bedingung für das erfolgreiche Optimieren der Gewichte für ein spezifisches Problem ist, wird unten deutlich.

Wenn nun mehrere solcher Neuronen in Lagen angeordnet werden und jedes Neuron Eingaben nur von den Neuronen der darunterliegenden Schicht erhält, spricht man von einem *feed forward* Netz: Der Informationsfluss in einem solchen Netz verläuft nur in einer Richtung, nämlich von den Eingangswerten der untersten Lage zu den Ausgangswerten der obersten Lage.

Abbildung 2.4 zeigt ein *feed forward* Netz mit n Eingabewerten ξ_i, m versteckten Knoten und einem Ausgabeknoten. Obwohl eigentlich nur zwei der Lagen (Ausgabeknoten und versteckte Schicht) wirklich Neuronen enthalten, spricht

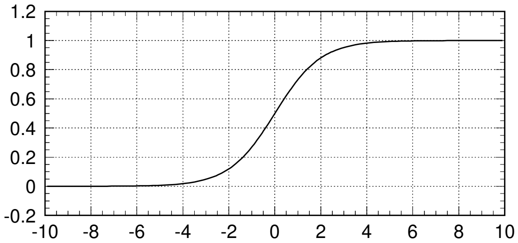


Abbildung 2.3: Sigmoid-Funktion (vgl. Gleichung (2.2))

man hier gebräuchlicherweise von einem dreilagigen neuronalen Netz. Im allgemeinen Fall ist die Zahl der Ausgabeknoten beliebig — im Fall eines Triggers, der nur eine Ja-oder-Nein-Entscheidung zu treffen hat, genügt jedoch einer. Wenn der Ausgabewert des Ausgabeknotens über einer Schwelle (*cut*) liegt, wird das Ereignis akzeptiert, sonst verworfen. Mit geeignet gewählten Gewichten lassen sich mit einem derartigen Netzwerk beliebige Aufgaben der Mustererkennung lösen.

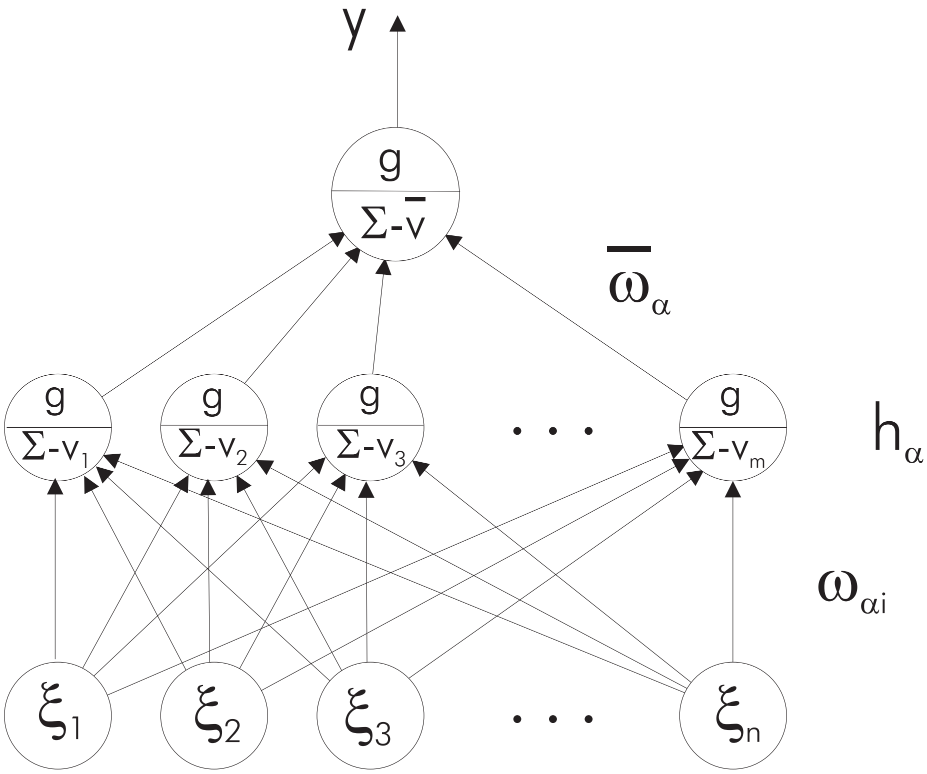


Abbildung 2.4: Schematische Darstellung eines dreilagigen *feed forward* Netzes

In Verallgemeinerung von Gleichung (2.1) berechnet jedes Neuron α in der versteckten Schicht aus den Eingabewerten ξ_i die Eingangswerte $h_\alpha(\xi)$ für den

Ausgabeknoten

$$h_\alpha(\xi) = g\left(\sum_{i=1}^{n} \omega_{\alpha i}\xi_i - v_\alpha\right), \tag{2.3}$$

aus denen dann der Ausgabewert $y(\xi)$ bestimmt wird:

$$y(\xi) = g\left(\sum_{\alpha=1}^{m} \bar{\omega}_\alpha h_\alpha(\xi) - \bar{v}\right). \tag{2.4}$$

Meist werden bei neuronalen Netzen die Parameter $\vec{\omega}$ durch ein „Lernverfahren“ (Training) bestimmt. Dazu müssen im Allgemeinen Beispielmustersätze zur Verfügung stehen. Ziel ist es, die Gewichte so zu wählen, dass für die Eingabemuster ξ^μ der p Beispiele die berechneten Ausgabewerte $y(\xi^\mu)$ möglichst gut mit den gewünschten Ausgabewerten η^μ (Targetwerte) übereinstimmen. Im Falle eines Triggers ist der Targetwert 1 für ein Ereignis, das getriggert werden soll, und 0 für ein Untergrundereignis.

Ein Standardverfahren für das Training liefert der Backpropagation-Algorithmus [8, 9]. Er minimiert eine Fehlerfunktion E mit Hilfe einer Gradientenabstiegsmethode. Üblicherweise wählt man die quadratische Abweichung der Netzausgabe y vom gewünschten Targetwert η als Fehlerfunktion: Für einen Satz ξ^μ von Eingabemustern mit den Targetwerten η^μ erhält man

$$E = \frac{1}{2}\sum_{\mu=1}^{p} \left(\eta^\mu - y(\xi^\mu)\right)^2. \tag{2.5}$$

Je kleiner E ist, desto mehr stimmen die berechneten Ausgangswerte mit den gewünschten Targetwerten überein. Üblicherweise wird während des Trainings nicht das globale Minimum von E, sondern nur ein lokales gefunden.

Sei nun $\vec{\omega}$ ein Vektor, der alle Gewichte eines Netzes ($\omega_{\alpha i}$ und $\bar{\omega}_\alpha$ sowie v_α und $\bar{v}_\alpha$) umfasst. Dann kann dieser Vektor in mehreren Iterationsschritten in Richtung des negativen Gradienten von E angepasst werden:

$$\vec{\omega}_{t+1} = \vec{\omega}_t + \Delta\vec{\omega}_t \quad \text{mit} \quad \Delta\vec{\omega}_t = -\epsilon\frac{\partial E_t}{\partial\vec{\omega}_t}. \tag{2.6}$$

Da die Fehlerfunktion E dazu überall im Raum der Gewichte differenzierbar sein muss, folgt die Notwendigkeit einer differenzierbaren Stufenfunktion $g(x)$.

Der Parameter ϵ ist die Lernrate. Von ihm hängt ab, wie schnell das Verfahren konvergiert. Wird er zu groß gewählt, wird eventuell nicht das ideale Trainingsergebnis erreicht. Wird er zu klein gewählt, nähert man sich dem optimalen Ergebnis nur langsam. Typisch ist ein Wert von 10^{-2}.

Ein komplettes Training besteht aus der wiederholten Anwendung des Iterationsschritts auf einen (pseudo-)zufällig, aber reproduzierbar initialisierten Vektor

$\vec{\omega}$. Theoretische Überlegungen [10] führen dazu, die einzelnen Elemente des Initialisierungsvektors mit Werten aus einem Intervall $[-\zeta, \zeta]$ mit

$$\zeta = \frac{3}{\sqrt{n}} \tag{2.7}$$

zu wählen, wobei n die Zahl der benutzten Eingangsgrößen ist. Die Zahl der Iterationen heißt Epochen. Typische Werte liegen zwischen 500 und 5000.

Der Backpropagation-Algorithmus kann durch einen *momentum term* erweitert werden. Es gilt dann

$$\Delta\vec{\omega}_t = -\epsilon \frac{\partial E_t}{\partial \vec{\omega}_t} + \alpha \Delta\vec{\omega}_{t-1} \tag{2.8}$$

mit einem Momentumparameter $\alpha \in [0, 1]$. Wie der Name sagt, wird der Suchbewegung durch diesen zusätzlichen Term ein gewisser „Impuls" verliehen. Er unterdrückt einerseits starke Oszillationen in Bereichen des Parameterraums, in denen die Fehlerfunktion rasch veränderlich ist, andererseits vergrößert er die effektive Lernrate in Bereichen, in denen sie nahezu konstant verläuft.

Nach dem Training muss man das resultierende Netz mit unabhängigen Testdaten überprüfen. Der Grund ist, dass gerade bei den interessanten, aber schwierig zu triggernden Reaktionsklassen naturgemäß wenige Ereignisse zum Trainieren zur Verfügung stehen. Deshalb besteht die Gefahr des *overtrainings*. Dabei verringert sich E bei einer zunehmenden Zahl von Epochen bezüglich der Trainingsdaten immer weiter, nimmt aber bezüglich eines Satzes unabhängiger Testdaten wieder zu. Das Netz beginnt, die statistischen Fluktuationen der Trainingsdaten „auswendig" zu lernen, die natürlich im Testdatensatz nicht vorhanden sind.

2.2.2 Hardware

Bei H1 wird das Konzept der neuronalen Netze erstmals in der Hochenergiephysik in einem Trigger verwendet [11, 12]. Durch die scharfen zeitlichen Rahmenbedingungen ist eine besondere Hardware nötig, die im Folgenden beschrieben wird.

Die Hardware besteht im Wesentlichen aus *receiver cards* (PQZP oder Quickbus), Data Distribution Boards (DDB) und einer SIMD (Single Instruction Multiple Data) Architektur mit 64 parallel arbeitenden Prozessorknoten, dem CNAPS (vgl. auch Abbildung 2.5).

Die *receiver cards* empfangen die Triggerinformationen der Subdetektoren und leiten sie über den L2-Bus an die DDBs weiter. Der Flüssig-Argon-Kalorimeter-Trigger kommuniziert über das Quickbus-Protokoll mit den *receiver cards*, die anderen Subdetektoren über das PQZP-Protokoll.

Die auf dem L2-Bus liegenden Daten werden von der DDB zu Netzeingabegrößen (*input quantities*) verarbeitet und an die Eingangsknoten eines CNAPS

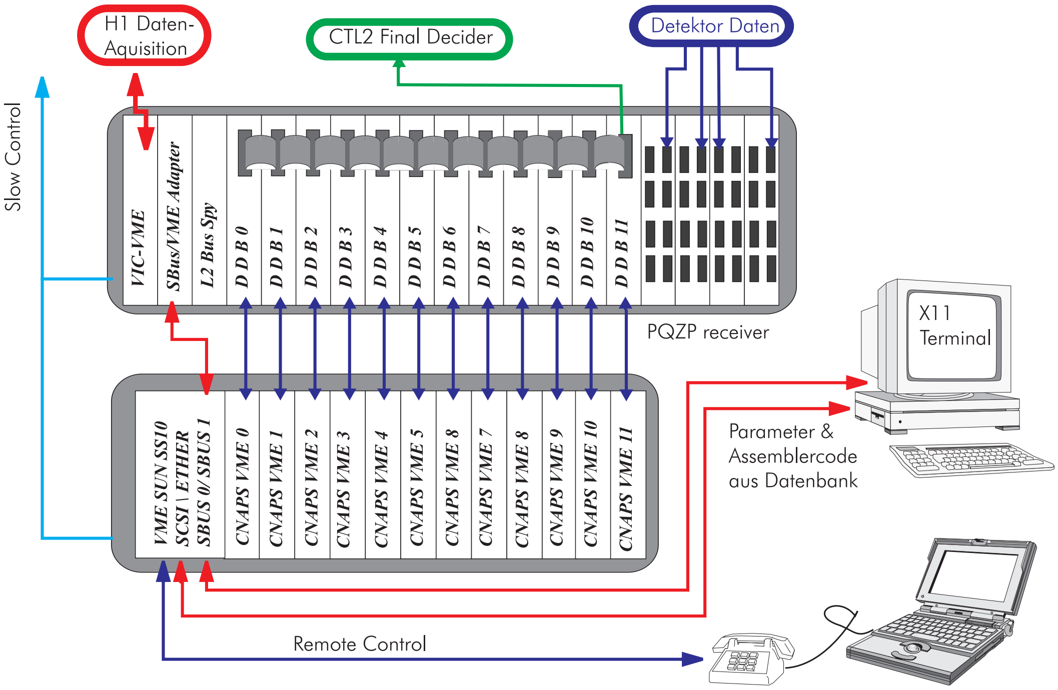


Abbildung 2.5: Überblick über die Struktur der L2-Hardware

weitergeleitet[1]. Bis zu 64 Eingabegrößen können gleichzeitig in einem Netz verwendet werden. Die Gesamtheit aller Eingabegrößen eines Netzes nennt man Eingabevektor $\vec{x}$ (s. Abschnitt 2.2.1).

Der CNAPS realisiert ein dreilagiges *feed forward* Netz und berechnet in nur $8\mu s$ aus dem Eingabevektor zuerst die Werte der Knoten im *hidden layer* und dann den des Ausgangsknotens. Falls dieser eine festgelegte Schwelle, den *cut*, überschreitet (oder exakt dem *cut* entspricht), führt dies zu einer positiven Triggerentscheidung. Die Entscheidung wird über die DDB der Zentralen Trigger Logik (*central trigger logic*, CTL) mitgeteilt, die letztendlich entscheidet (unter Beachtung von *prescale* und *override* zugunsten von Monitorereignissen), ob ein Ereignis getriggert wird. Zu Monitorzwecken übergibt der CNAPS der DDB zusätzlich den vollen Bytewert des Ausgabeknotens.

Von der insgesamt zur Verfügung stehenden Zeit von $20\mu s$ kann ungefähr die Hälfte für die Datenvorverarbeitung und für die Berechnung des Netzes genutzt werden.

Um mehrere unterschiedliche Triggerkanäle gleichzeitig behandeln zu können, werden mehrerer Triggerboxen, bestehend aus jeweils einer DDB und einem CNAPS, parallel auf dem L2-Bus betrieben. Jede Triggerbox mit ihrer DDB und ihrem CNAPS wird für einen Triggerkanal individuell konfiguriert, d.h. die Da-

[1] Die Bezeichnung Data Distribution Board ist nicht sehr treffend gewählt, da die Hauptaufgabe der DDB die Bildung von Netzeingabegrößen ist, der Name sich aber aus der zweiten Funktion, der Verteilung der Daten an die Eingangsknoten eines CNAPS ableitet.

tenvorverarbeitung in der DDB und das neuronale Netz im CNAPS werden dem jeweiligen Problem angepasst. Derzeit sind zwölf Paare von DDB und CNAPS im Einsatz.

Bedingt durch die interne Struktur des CNAPS-Chips wird bei der Berechnung der neuronalen Netze eine Festkommaarithmetik mit einer Genauigkeit von einem Byte für den Eingabevektor und zwei Bytes für die Netzgewichte verwendet. Der Ausgabewert, den der CNAPS an die DDB zurückliefert, hat ebenfalls einen Umfang von einem Byte.

Das gesamte System wird über VME-Interfaces durch eine SUN SPARCstation kontrolliert und konfiguriert.

2.2.3 Netztraining

Für jeden Physikkanal, der mit Hilfe des L2NN getriggert werden soll, muss ein Netz, im Allgemeinen mit Hilfe des Backpropagation-Algorithmus, trainiert werden. Dafür stehen eine Reihe von Programmen zur Verfügung, die sogenannte Trainingskette.

Zuerst werden die möglichen Eingabegrößen für die Physik- und die Untergrundereignisse simuliert. Dies geschieht mit dem Programm `NETSIM` [13]. Die Untergrundereignisse stammen aus *L2L4 transparent runs*[2]. Die Physikereignisse stammen aus einem auf andere Weise getriggerten Datensatz oder aus einem Monte-Carlo-Sample. Beide Ereignistypen müssen natürlich die Subtriggerbedingung erfüllen, die durch das Netz verifiziert werden soll.

Anschließend werden ein Trainings- und zwei Testdatensätze erstellt. Der Trainingsdatensatz enthält eine Mischung aus Physik- und Untergrundereignissen, die dem Backpropagation-Algorithmus als Beispielsatz präsentiert wird. Es erweist sich als günstig, im Trainingsdatensatz möglichst die gleiche Zahl an Physik- und Untergrundereignissen zu verwenden und diese möglichst homogen zu durchmischen. Die Testdatensätze enthalten jeweils einen reinen Satz von Physik- und Untergrundereignissen, mit denen das fertige Netz auf *overtraining* überprüft wird. Das Programm, mit dem die einzelnen Datensätze erstellt werden, heißt `MIXER` [14].

Nach der empirischen Wahl der für das Problem optimalen Eingabegrößen wird das Training selbst vom `TRAINER` [15] durchgeführt. Dabei ist neben der Lernrate ϵ, dem Momentumparameter α und der Zahl der Epochen auch die Zahl der versteckten Knoten n anzugeben. Meist werden mehrere Netze mit verschiedenen Parametersets trainiert, von denen dann das Beste verwendet wird. Der `TRAINER` führt bereits während des Trainings nach jeder Epoche einen Test mit den Testdatensätzen durch. Dadurch ist *overtraining* sofort zu erkennen.

---

[2] Ein *L2L4 transparent run* ist ein *run*, bei dem jedes Ereignis, das von L1 getriggert wird, von L2 und L4 akzeptiert wird. L2 und L4 sind also transparent. Wegen der hohen Untergrundrate sind die getriggerten Ereignisse in guter Näherung ausschließlich Untergrundereignisse.

Als letzter Schritt wird der *cut* festgelegt. Ereignisse mit einem Ausgabewert, der diesen *cut* überschreitet oder ihn exakt erreicht, werden getriggert, die anderen verworfen. Mit einem niedrigerem *cut* steigt die Effizienz, mit der Physikereignisse getriggert werden (*physics efficiency*), es sinkt hingegen der Anteil von Untergrundereignissen, die zurückgewiesen werden (Untergrundreduktion, *background efficiency*). Normalerweise wird ein Netz für eine bestimmte Untergrundreduktion trainiert und der *cut* daran angepasst. Die gewünschte Untergrundreduktion orientiert sich am *prescale*, den der betrachtete Subtrigger ohne zusätzliche Verifizierung durch L2 zugewiesen bekäme.

Ausführliche Untersuchungen eines neuronalen Netzes (z.B. Bestimmung der Relevanz einzelner Eingabegrößen) sind mit dem `TESTER` [16] möglich.

Kapitel 3

Intelligente Datenvorverarbeitung

Da *overtraining* vermieden werden muss, werden bessere Ergebnisse beim Netzwerktraining erzielt, wenn dem Netz wenige physikalisch relevante Größen anstelle einer großen Datenmenge, die nicht mehr physikalische Information enthält, präsentiert werden. Von Nachteil ist die Verwendung unphysikalischer Größen, die mit ihren statistischen Fluktuationen *overtraining* begünstigen. Ein einfaches Beispiel für eine unphysikalische Größe ist der Azimuthwinkel ϕ, der angibt, wie ein bestimmtes Ereignis im Detektor orientiert ist. Dies gilt natürlich nur, falls winkelabhängige Detektoreffizienzen außer Acht gelassen werden können.

In dieser Arbeit wurde daher der Ansatz verfolgt, die dem Netz präsentierten Daten durch eine intelligente Datenvorverarbeitung aufzubereiten und damit die physikalisch relevante Information, die in diesen Daten enthalten ist, dem Netz in möglichst reiner Form zur Verfügung zu stellen.

Im ersten Abschnitt dieses Kapitels wird die bisher verfügbare Datenvorverarbeitung beschrieben und auf deren Grenzen eingegangen.

Danach werden zunächst die Konzepte, die einer intelligenten Datenvorverarbeitung zugrunde liegen, kurz angesprochen und ihr Zusammenspiel sowie ihre prinzipielle Anwendung verdeutlicht. In weiteren Abschnitten werden die Konzepte einzeln vorgestellt.

Realisiert werden diese Konzepte in einer Hardwarekomponente, die im Rahmen des Triggerupgrades bei H1 das bisher vom L2NN verwendete Data Distribution Board (DDB) wie es im letzten Kapitel beschrieben wurde, ersetzen wird. Die Bezeichnung dieser neuen Hardwarekomponente lautet DDB2. Die DDB2 wird neben der neuen Funktionalität auch alle Funktionen der jetzt verwendeten DDB bereitstellen. Derzeit befindet sich die DDB2 noch in der Planungsphase.

Als erster Schritt zur Entwicklung der DDB2 wurde in dieser Arbeit eine Simulation erstellt, anhand derer Algorithmen für die DDB2 entwickelt und getestet werden können. Auch Untersuchungen zur Leistungsfähigkeit der DDB2 werden dadurch möglich. Die Dokumentation der Simulation ist in Anhang A enthalten, ebenso eine Beschreibung aller implementierungsspezifischen Aspekte, die über das reine Konzept hinausgehen.

3.1 Herkömmliche Datenvorverarbeitung

Als erstes wird die von der bisherigen DDB vorgenommene Datenvorverarbeitung genauer beschrieben [11]:

Die DDB liest die Triggerdaten vom L2-Bus. Diese Daten sind eine inhomogene Mischung von Informationen verschiedener Subdetektoren, z.B. Summen der Kalorimeterenergien über Teile des Detektors, Vertex-Histogramme der Spurdetektoren, Bitmuster aus den Proportionalkammern und bitkodierte Trefferanzeigen der Myondetektoren. Die Daten, die die verschiedenen Subdetektoren zur Verfügung stellen, werden in den weiteren Abschnitten dieses Kapitels im Einzelnen vorgestellt, zusammen mit ihrer Behandlung durch eine intelligente Datenvorverarbeitung. Für weitere Informationen über die bereitgestellten Datenformate möge man [17] heranziehen.

Der L2-Bus besteht aus acht je 16 Bit breiten Subbussen. Auf diese verteilen die *receiver cards* die ankommenden Daten wortweise[1]. Dabei ist es möglich, dieselben Daten auch auf mehreren Subbussen zur Verfügung zu stellen.

Von der DDB können Eingabegrößen aus Worten berechnet werden, die zusammen auf einem Subbus liegen. Jedes Wort auf einem Subbus kann nur in eine Eingabegröße einfließen. Die Weiterverarbeitung der Daten beschränkt sich im Wesentlichen auf die Anwendung bestimmter Operationen auf die auf einem Subbus liegenden Worte und die Summierung über die Ergebnisse mehrerer unmittelbar aufeinanderfolgender, gleichartiger Operationen. Die von der Hardware unterstützten Operationen sind teilweise in Tabelle 3.1 zusammen mit ihren Kurzbezeichnungen aufgeführt. Nicht alle dieser möglichen Operationen werden derzeit softwareseitig unterstützt.

Eine wesentliche Beschränkung der herkömmlichen Datenvorverarbeitung ist die Tatsache, dass in jede Eingabegröße normalerweise nur Daten eines Subdetektors eingehen und keine komplexeren Zusammenhänge zwischen den zur Verfügung stehenden Informationen ausgewertet werden. Auch können durch die oben beschriebenen Einschränkungen bei der Bildung von Eingabegrößen nicht alle Informationen, die zur Verfügung stehen würden, gleichzeitig genutzt werden.

3.2 Konzepte für eine verbesserte Vorverarbeitung

Ziel einer intelligenten Datenvorverarbeitung ist es, den neuronalen Netzen möglichst physikalisch relevante Größen zur Verfügung zu stellen. Physikalisch

[1] Als Wort wird hier die Kombination zweier Bytes, also 16 Bits bezeichnet.

`LoBy`	*low* Byte
`HiBy`	*high* Byte
`BiSu`	Summe über alle 16 Bits
`BySu`	Summe über *low* Byte und *high* Byte
`BiPo`	Position des ersten gesetzten Bits
`NaWo`	Transformation Wort zu Byte
`LLxx`	frei definierbare LUT angewendet auf *low* Byte
`LHxx`	frei definierbare LUT angewendet auf *high* Byte
`LSxx`	frei definierbare LUT angewendet auf *low* Byte + *high* Byte
`Tit0/1/2`	Vorprogrammierte LUT für SpaCal-Daten

Tabelle 3.1: Einige Operationen, die in der herkömmlichen DDB auf Worte auf einem L2-Subbus angewendet werden können. Über die Ergebnisse mehrerer gleicher Operationen kann summiert werden. LUT bedeutet *look-up-table*.

relevante Größen sind insbesondere die Eigenschaften von einzelnen bei den Reaktionen entstehenden Teilchen (Impuls, Energie und Art des Teilchens oder Teilchenbündels (Jet)). Diese Eigenschaften werden in verschiedenen Komponenten des Detektors bestimmt: Die Energie wird in einem Kalorimeter gemessen, die Krümmung der Spur in einer Spurkammer und die Art des Teilchens wird durch die Signaturen bestimmt, die das Teilchen in verschiedenen Subdetektoren hinterlässt. Um alle Eigenschaften eines Teilchens in Relation zueinander betrachten zu können, muss also die Information der verschiedenen Detektorkomponenten kombiniert werden. Dies führt zu folgendem Ansatz: Mehrere zusammengehörende Signale aus verschiedenen Subdetektoren mit ihren Eigenschaften bilden ein „Objekt“. Solchen Objekten liegen einzelne isolierte Teilchen oder eng kollimierte Teilchenbündel (Jets) zugrunde.

Voraussetzung für eine derartige Kombination von Signalen aus verschiedenen Subdetektoren ist, dass diese bereits so aufbereitet sind, dass ein Teilchen (oder Jet) nur zu einem Signal führt. Deshalb müssen zum Beispiel nahe beieinander liegende Zellen des Kalorimeters, in denen Energie deponiert wurde (ebenso wie benachbarte Bereiche der Spurkammern, in denen Spuren auftreten), selbst zu einem — definitionsgemäß zu einem einzelnen Objekt gehörenden — Signal kombiniert werden. Dieser Prozess des Zusammenfassens von räumlich benachbarten Signalen innerhalb eines Subdetektors wird *clustering* genannt. Die dabei gebildeten Signale werden hier als „Cluster“ bezeichnet, ihre Komponenten als Zellen. Ein Cluster besitzt ein Zentrum und einen Ring. Das Zentrum eines Clusters ist die Zelle, durch die die Position des gesamten Clusters definiert wird. Um physikalisch sinnvolle Ergebnisse zu erhalten, sollte das Zentrum beispielsweise durch das Energiemaximum oder den Schwerpunkt des Clusters festgelegt sein. Der Ring ist der Bereich des Clusters, der nicht das Zentrum ist. Typische Eigenschaften eines Clusters sind: Gesamte im Cluster deponierte Energie, Energie im Zentrum oder Ring des Clusters und die Zahl der am Cluster beteiligen Zellen. Unabhängig vom verwendeten Clusteralgorithmus kann der Fall auftreten, dass es zwei gleichberechtigte Kandidaten für das Clusterzentrum gibt (je nach Algorithmus charakterisiert durch gleiche Energie oder gleiche Lage relativ zu benachbarten aktiven Zellen). Es wird vereinbart, dass in einem solchen Fall die Zelle mit dem größeren ϕ-Index, oder, wenn auch diese gleich sind, die mit dem größeren θ-Index als Clusterzentrum verwendet wird.

Cluster, die vom selben Teilchen in verschiedenen Detektorkomponenten erzeugt wurden, werden durch ihre räumliche Nähe zueinander in der θ-ϕ-Ebene erkannt und zu einem Objekt kombiniert. Dieser Vorgang wird mit *matching* bezeichnet. Natürlich kann nicht sichergestellt werden, dass alle Cluster, die in der θ-ϕ-Ebene nahe beieinander liegen, vom selben Teilchen (oder Jet) stammen, aber die Wahrscheinlichkeit dafür ist hoch. Ein Objekt kann selbstverständlich auch nur aus einem Cluster in einem Subdetektor bestehen, wenn dieser mit keinem Cluster in einem anderen Subdetektor kombiniert werden konnte. Jeder Cluster wird im Allgemeinen nur in einem Objekt verwendet, da er per definitionem von einem einzelnen Teilchen (oder Jet) stammt. Aus Gründen der Realisierbarkeit in der Hardware wird die Zahl der insgesamt gebildeten Objekte auf 16 beschränkt.

Nun stehen Objekte zur Verfügung, die jeweils aus Clustern in einem oder mehreren Subdetektoren gebildet wurden und mit Teilchen (oder Jets) identifiziert werden können. Die Eigenschaften dieser Objekte werden durch die Eigenschaften der Cluster bestimmt, aus denen sie gebildet wurden. In hochenergetischen Reaktionen, wie zum Beispiel in Elektron-Proton-Kollisionen bei HERA werden meist mehrere Teilchen oder Jets erzeugt. Entsprechend können mehrere Objekte gebildet werden. Von physikalischem Interesse sind allerdings nur bestimmte, die sich oft durch eine besonders herausragende Eigenschaft auszeichnen: Sie haben eine besonders hohe Energiedeposition im Kalorimeter oder ein seltenes Signal in einem Subdetektor (zum Beispiel in den Myonkammern). Auch die Lage zueinander kann zwei Objekte interessant machen, wenn sie beispielsweise in entgegengesetzte Richtungen weisen oder sehr nahe beisammen liegen. Um den Zugriff auf die „interessanten“ Objekte zu ermöglichen, werden die Objekte nach unterschiedlichsten Kriterien sortiert. Als Kriterium kommen alle Eigenschaften in Frage, die die Objekte besitzen, also neben den Eigenschaften, die sie von den Clustern, aus denen sie aufgebaut sind, erhalten, auch ihre räumliche Position. Man nennt diesen Vorgang *ordering*.

Die Eigenschaften der sortierten Objekte können nun als Netzeingabegrößen dienen. Ein einfacher Ansatz ist, jeweils acht wichtige Eigenschaften der vier energiereichsten Objekte als Eingabegrößen für ein neuronales Netz zu verwenden. Die Zahl der Eigenschaften und der verwendeten Objekte ist dabei nur durch die maximale Zahl von möglichen Eingabegrößen für ein neuronales Netz begrenzt, allerdings führt eine hohe Zahl von Eingabegrößen sehr schnell zu *overtraining*, so dass eine gezieltere Auswahl der Eingabegrößen anzuraten ist. Um dies zu unterstützen, wird noch eine Nachverarbeitung der Daten der sortierten Objekte durchgeführt. Beispielsweise kann anstelle der Winkel zweier Objekte die Winkeldifferenz zwischen den Objekten verwendet werden. Auch kann statt den aufgelisteten Eigenschaften aller Objekte die Zahl von Objekten, deren Eigenschaften bestimmte Bedingungen erfüllen, als Eingabegrößen verwendet werden. Dadurch wird die Zahl der Eingabegrößen reduziert und *overtraining* vermieden. Dieser abschließende Schritt der Datenvorverarbeitung, der derartige hochintegrierte Information bereitstellt, wird *post-processing* genannt.

Um mehr Flexibilität in der Behandlung der von den Subdetektoren bereitgestellten Daten zu erhalten, werden zusätzlich vor den bisher vorgestellten Schritten auf alle in den Algorithmus eingehenden Daten *look-up-tables* angewandt.

Im Folgenden werden die einzelnen Schritte der eben skizzierten „intelligenten" Datenvorverarbeitung genauer erläutert.

3.3 *look-up-tables*

Der erste Schritt einer intelligenten Datenvorverarbeitung besteht darin, auf alle verfügbaren Daten eine *look-up-table* (LUT) anzuwenden [18], die jedem vorkommenden Eingabewert jeden möglichen Ausgabewert zuordnen kann. Die bisher teilweise schon bei der herkömmlichen Datenverarbeitung eingesetzten LUTs (s. Tabelle 3.1) sind wesentlich eingeschränkter verwendbar: Auf alle Daten, die in eine Eingabegröße eingehen, wird dieselbe LUT angewendet. Außerdem ist die Zahl der insgesamt einsetzbaren LUTs beschränkt.

Wenn man die Datenstrukturen betrachtet, die dem L2-Trigger von den verschiedenen Subdetektoren zur Verfügung gestellt werden, lassen sich drei Kategorien unterscheiden, an deren spezifische Eigenschaften die LUTs angepasst werden müssen.

- Von den *big towers* bzw. *trigger towers* (vgl. Abschnitte 3.4.1 und 3.4.2) wird pro Kanal ein Byte geliefert, das von der im Kanal deponierten Energie abhängt. Eine LUT für einen Kanal besteht aus 256 Bytes, die den möglichen Eingabewerten zugeordnet werden.

- Ähnliches gilt für die Informationen, die vom groben Raster des SpaCals (vgl. Abschnitt 3.4.6) zur Verfügung stehen. Je nachdem, welche der drei Energieschwellen in einem Kanal überschritten werden, wird ein Wert zwischen 0 und 3 geliefert. Hier benötigt man demnach eine LUT mit vier Einträgen pro Kanal.

- Die anderen Detektorkomponenten senden pro Kanal nur ein Bit. Die LUT besteht dann aus zwei Bits, die die Ausgabewerte für die beiden Eingabewerte angeben.

Einige Verwendungsmöglichkeiten solcher LUTs sind:

- Sehr selektive Korrektur von Hardwarefehlern in den Subdetektoren (z.B. von einzelnen hängende Bits).
- Umwandlung der im Flüssig-Argon-Kalorimeter gemessenen Energien E in die Transversalenergien E_T. Selbstverständlich sind auch andere Abbildungen denkbar.
- Vollständige oder teilweise (z.B. in ausgewählten Raumwinkelbereichen) Deaktivierung von Subdetektoren.

3.4 *clustering*

Nach der Anwendung von LUTs folgt als nächster Schritt das *clustering* [18]: Signale aus demselben Subdetektor, die sich in räumlicher Nähe zueinander befinden, werden als physikalische Einheit betrachtet, da sie vermutlich vom selben Teilchen oder zumindestens vom selben Jet erzeugt wurden. Beim *clustering* muss ein Mittelweg zwischen dem Wünschenswerten, nämlich einer mathematisch sauberen Bestimmung von Position, Größe und Grenzen der Cluster, und dem technisch Realisierbaren gefunden werden. Bei der technischen Realisierbarkeit spielt vor allem der Zeit-, teilweise aber auch der Speicherbedarf der Algorithmen eine Rolle.

Wie bei den LUTs ist auch beim *clustering* die Implementierung des allgemeinen Algorithmus stark vom betrachteten Subdetektor abhängig: Zum einen wird dies durch die unterschiedliche Granularität (zum Beispiel: 16 auf 16 Felder in zwei Dimensionen, 45 Felder in einer Dimension, ...) der verschiedenen Subdetektoren verursacht, zum anderen durch die unterschiedlichen von ihnen bereit gestellten Datenstrukturen (zum Beispiel: Bytewert pro Kanal oder Bitwert pro Kanal, ...)

Deshalb muss der allgemein formulierte Algorithmus „Finden von Clustern“ für jeden Subdetektor unterschiedlich implementiert werden. Erste Untersuchungen dazu wurden in [19] durchgeführt. In den folgenden Abschnitten wird nun der Reihe nach auf die bei den verschiedenen Subdetektoren verwendeten Algorithmen eingegangen.

3.4.1 Kalorimeter (*big towers*)

Der gesamte vom Flüssig-Argon-Kalorimeter, dem *plug* und dem SpaCal überdeckte Raumwinkel ist, vom Wechselwirkungspunkt aus gesehen, in *big towers* unterteilt. Ihre Indizierung und ihre Lage in der θ-ϕ-Ebene ist in Abbildung 3.1 dargestellt.

Für die Winkel θ und ϕ werden aus programmtechnischen Gründen Indizes eingeführt: Dabei wird aus $0° \leq \phi < 360°$ nun $0 \leq \phi < 256$. Der θ-Index läuft von 0 für das *plug* bis 25 für den inneren Bereich des SpaCals (s. Tabelle 3.2).

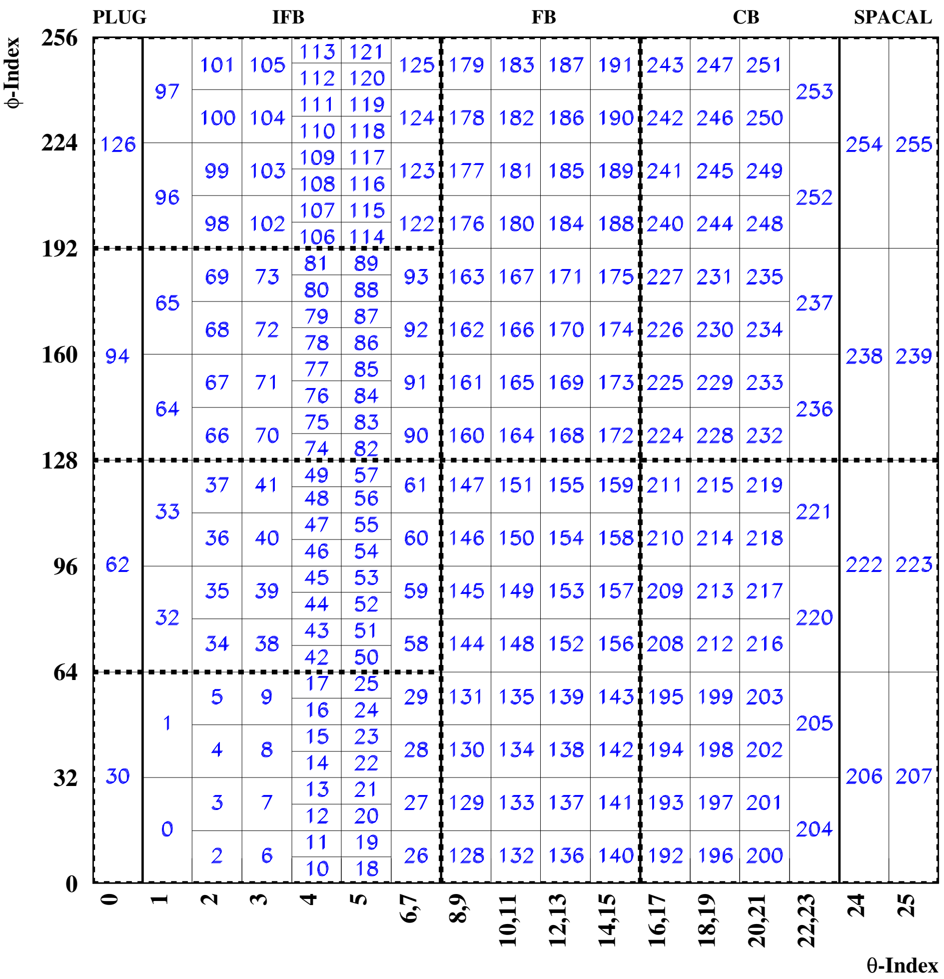


Abbildung 3.1: Überdeckung der θ-ϕ-Ebene durch die *big towers*. Die Zahlen in den Feldern bezeichnen die Nummern der *big towers* (nach [19]). Die Grenzen der Bereiche, in denen die Suche nach Clustern zunächst parallel durchgeführt wird (s. Text), sind durch die gestrichelten Linien gekennzeichnet.

In Abbildung 3.1 ist zu erkennen, dass teilweise einem *big tower* zwei θ-Indizes zugeordnet werden. Dies ist dadurch bedingt, dass die Granularität der *big towers* nicht die feinste im Detektor vorkommende Granularität ist.

Hadronischer und elektromagnetischer Teil des Flüssig-Argon-Kalorimeters liefern jeweils bis zu 240 *big towers*. Hinzu kommen weitere 12 aus *plug* und SpaCal. Das *plug* ist in vier Quadranten aufgeteilt, das SpaCal in je vier innere und äußere Sektoren. Die *big towers* des SpaCal liefern momentan nur elektromagnetische Daten, ebenso wie die hintersten *big towers* des Flüssig-Argon-Kalorimeters (θ-Index 22 und 23). Das *plug* liefert nur hadronische Daten.

plug	0
LAr IF	$1\ldots7$
LAr FB	$8\ldots15$
LAr CB	$16\ldots23$
äußeres SpaCal	24
inneres SpaCal	25

Tabelle 3.2: θ-Indizes der verschiedenen Detektorkomponenten. Die Aufteilung des Flüssig-Argon-Kalorimeter-Triggers in drei Bereiche (IF, FB, CB, vgl. auch Abbildung 3.1) unterscheidet sich von der Bezeichnung der *wheels* in Abschnitt 1.5, da sich die Grenzen der *big towers* am Polarkoordinatensystem orientieren, die der *wheels* aber an den kartesischen Koordinaten.

Die *big towers* im Flüssig-Argon-Kalorimeter [20] sind je nach θ-Index aus bis zu vier *trigger towers* zusammengesetzt, deren Energien bereits vor der Digitalisierung analog addiert werden. Zurzeit wird die Information, die die *trigger towers* liefern, nur auf diese Weise benutzt. Nach dem Upgrade des Detektors wird sie auch an L2 weitergegeben, was jetzt bereits simuliert werden kann. Die in Tabelle 3.2 oder Abbildung 3.1 verwendete θ-Indizierung ist bereits für die Verwendung von *trigger towers* vorbereitet.

Der zur Suche nach Clustern verwendete Algorithmus hängt nicht wesentlich davon ab, welche der beiden Granularitäten verwendet wird. Er wird deshalb zunächst für die *big towers* beschrieben.

Eine Forderung an den Algorithmus ist, dass jeder *big tower* nur zu einem Cluster gehört und somit die Summe aller Clusterenergien die im Kalorimeter deponierte Energie nicht übersteigt. Dies stellt hohe Anforderungen an die Hardware, da eine Parallelisierung so nur in begrenztem Umfang möglich ist.

Von L1 wird eine Liste mit den Adressen und Energien der *big towers*, in denen Energie deponiert wurde, gesendet [17]. Um das Zentrum eines Clusters zu finden, werden die *big towers* noch während des Empfangs der Daten, absteigend nach der Energie sortiert. Der *big tower* mit der höchsten Energie ist notwendigerweise das Zentrum eines Clusters. Er und alle, die zu ihm benachbart sind, werden aus der Liste entfernt. Dies wird solange wiederholt, bis die Liste leer ist, also alle Cluster gefunden sind. Die Nachbarschaften zwischen *big towers* werden über Listen definiert.

Dieser Algorithmus hat einen Nachteil: Falls ein *big tower* nicht benachbart zum Zentrum eines Clusters ist, kann er nicht zu diesem Cluster gehören. Der angewandte Algorithmus bildet aus ihm sogar dann einen eigenen Cluster, wenn die in ihm deponierte Energie nicht größer ist als die seiner Nachbarn. Alternative Lösungsansätze [19] vermeiden dieses Problem, indem ein neuer Cluster nur dann gebildet wird, wenn kein Nachbar mit größerer Energie vorhanden ist. Allerdings führt dies zu Problemen bei der Implementierung in der Hardware (gesamter Zeitbedarf für die Datenvorverarbeitung auf L2 9 μs, Verwendung bestimmter FPGAs mit bestimmten Speichermöglichkeiten), so dass trotzdem der oben vorgestellte Ansatz gewählt wurde.

Noch während der Suche wird für alle Cluster Position (bestimmt durch die Nummer des zentralen *big towers*, daraus ableitbar θ- und ϕ-Index des Clusters), Gesamtenergie (Summe der Energien aller am Cluster beteiligten *big towers*), Zentrumsenergie (Energie des zentralen *big tower*), Ringenergie (Differenz zwischen Gesamtenergie und Zentrumsenergie) und die Zahl der *big towers*, aus denen der Cluster besteht, bestimmt.

Um die Berechnung zu beschleunigen, wird die Suche nach den Clustern zunächst für acht Bereiche in der θ-ϕ-Ebene, in die die Trigger-Elektronik des Flüssig-Argon-Kalorimeters unterteilt ist, parallel durchgeführt und die dabei gefundenen Cluster erst nachträglich kombiniert. In Abbildung 3.1 sind die Grenzen dieser Bereiche eingezeichnet.

Wenn zwei Cluster auf unterschiedlichen Seiten einer Grenze zwischen zwei solchen Bereichen liegen, werden sie nachträglich zu einem verschmolzen. Die Position des neuen Clusters ist die Position des größeren der beiden. Die anderen Eigenschaften werden ebenfalls neu bestimmt: Die Gesamtenergie des neuen Clusters ist die Summe der Gesamtenergien der ursprünglichen Cluster, die Zentrumsenergie ist die Zentrumsenergie des größeren an der Verschmelzung beteiligten Clusters, die Ringenergie ist die Differenz zwischen der Gesamtenergie und der Zentrumsenergie des neuen Clusters und die Zahl der an den Clustern beteiligten *big towers* wird selbstverständlich addiert.

Ein Beispiel für die Funktionsweise des Algorithmus wird in Abbildung 3.2 gegeben.

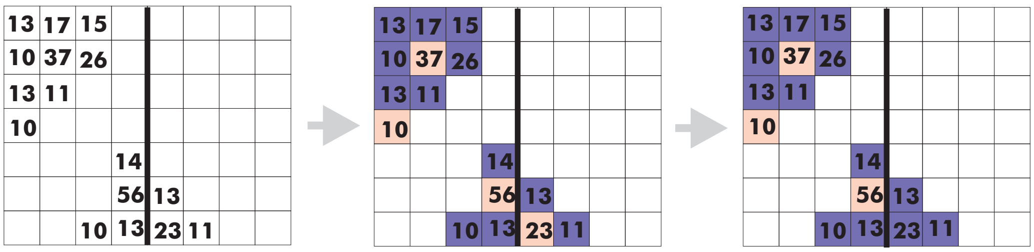


Abbildung 3.2: Bildung von Clustern aus *big towers*. Dargestellt ist ein Ausschnitt aus dem *big tower* Feld, der sich aus zwei Hardwarebereichen zusammensetzt, die zunächst unabhängig voneinander betrachtet werden. Links: Verfügbare Daten; Mitte: In den unabhängig voneinander betrachteten Hardwarebereichen gefundene Cluster; rechts: Kombination der beiden Cluster an der Grenze der Bereiche. Die Zentren der Cluster sind hell markiert, ihre Ringe dunkel. Der kleine Cluster am linken Rand ist ein Beispiel für den im Text angesprochenen Nachteil des Algorithmus.

3.4.2 Kalorimeter (*trigger towers*)

Wie bereits im letzten Abschnitt erwähnt, wird nach dem Upgrade feiner granulierte Information vom Flüssig-Argon-Kalorimeter für L1 und L2 zur Verfügung stehen, in der ebenfalls nach Clustern gesucht werden wird.

Auf die *trigger towers* wird derselbe Algorithmus angewendet, wie auf die *big towers*. In Abbildung 3.3 ist die Lage und die Indizierung der *trigger towers* in der θ-ϕ-Ebene dargestellt. Jetzt wird auch der Grund für die getroffene Wahl des θ-Indexes deutlich. Die *trigger towers* ersetzen nur die *big towers* aus dem Flüssig-Argon-Kalorimeter, die anderen (*plug*, SpaCal) werden weiterbenutzt.

In der Simulation wird auf spezielle Hardwaregegebenheiten bei den *trigger towers* (vergleichbar denen, die bei den *big towers* zur Aufteilung der θ-ϕ-Ebene in acht Bereiche, in denen parallel nach Clustern gesucht wird, führen) keine Rücksicht genommen, da deren genaue Implementierung noch nicht endgültig feststeht. Vermutlich bleibt aber die parallele Bearbeitung von Daten in mehreren Hardwarebereichen bestehen.

3.4.3 Proportionalkammern

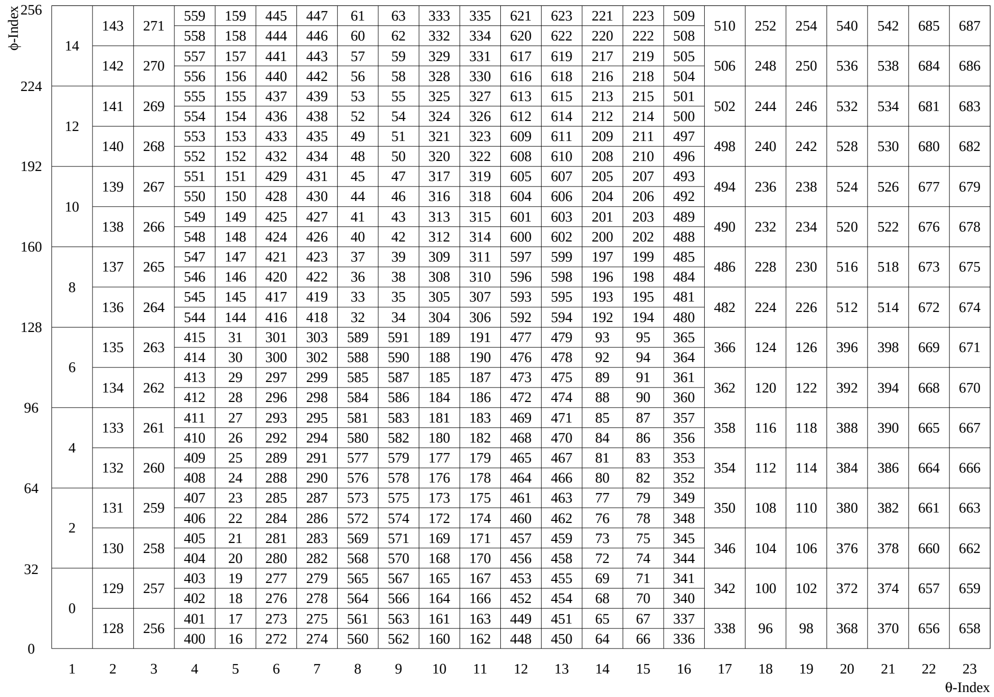
Die Triggerinformation der Proportionalkammern (MWPC) stammt aus der Kombination von Vorderer und Zentraler Proportionalkammer.

Ein *ray* besteht aus Signalen in drei von vier Pads der Proportionalkammern (je nach θ in CIP und COP oder in CIP und FPC) (s. Abschnitt 1.4) im selben ϕ-Sektor. Wenn man die Signale in den unterschiedlichen Kammern verbindet, erhält man die Richtung des *rays*. Ein oder mehrere *rays*, die in ähnliche Richtungen weisen, führen zu einem gesetzten *big ray* in der entsprechenden Zelle der *big ray* Mappe in der θ-ϕ-Ebene, wenn die Signale, aus denen er gebildet wurde, mit dem Maximum des z-Vertex-Histogramms (s. Abbildung 1.5) auf einer Geraden liegen.

Die *big rays* dienen unter anderem zur Validierung von *big towers* [20], die dann erfolgt, wenn im gleichen Raumwinkelbereich sowohl eine Mindestenergie im *big tower*, als auch ein *big ray* vorhanden ist. Die Zahl der validierten *big towers* geht in die Entscheidung des L1 Triggers ein. Deshalb stimmt die Granularität der *big rays* nach Konstruktion im Wesentlichen mit der der *big towers* überein (vgl. Abbildung 3.4).

Jeder von 240[2] *big rays* liefert nur ein Bit Triggerinformation, nämlich ob ein *big ray* vorhanden ist, oder nicht. Insgesamt ergibt das ein Bitfeld, im Gegensatz zu dem bei den *big towers* auftretendem Bytefeld, bei dem jede Zelle acht Bits Information über die Höhe der in ihr deponierten Energie besitzt. Aus diesem Grund wird ein anderer Algorithmus für die Suche nach Clustern benutzt als bei den *big towers* [19].

[2] Für θ-Index 0 wird keine Information gesendet.



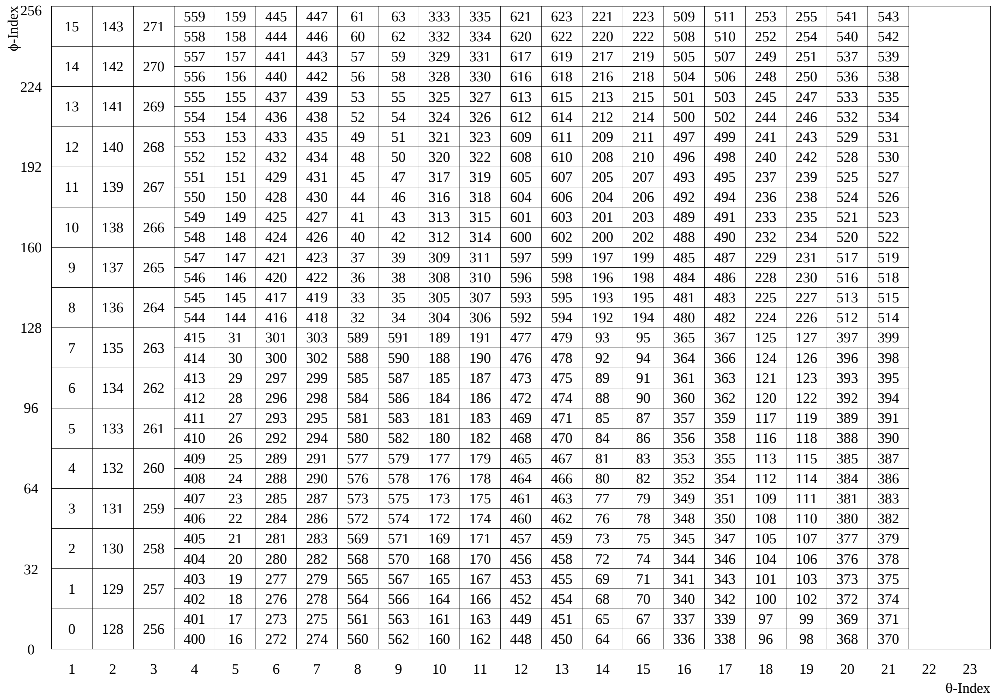


Abbildung 3.3: Abdeckung der θ-ϕ-Ebene durch die elektromagnetischen (oben) und die hadronischen (unten) *trigger towers* des Flüssig-Argon-Kalorimeters (nach [21])

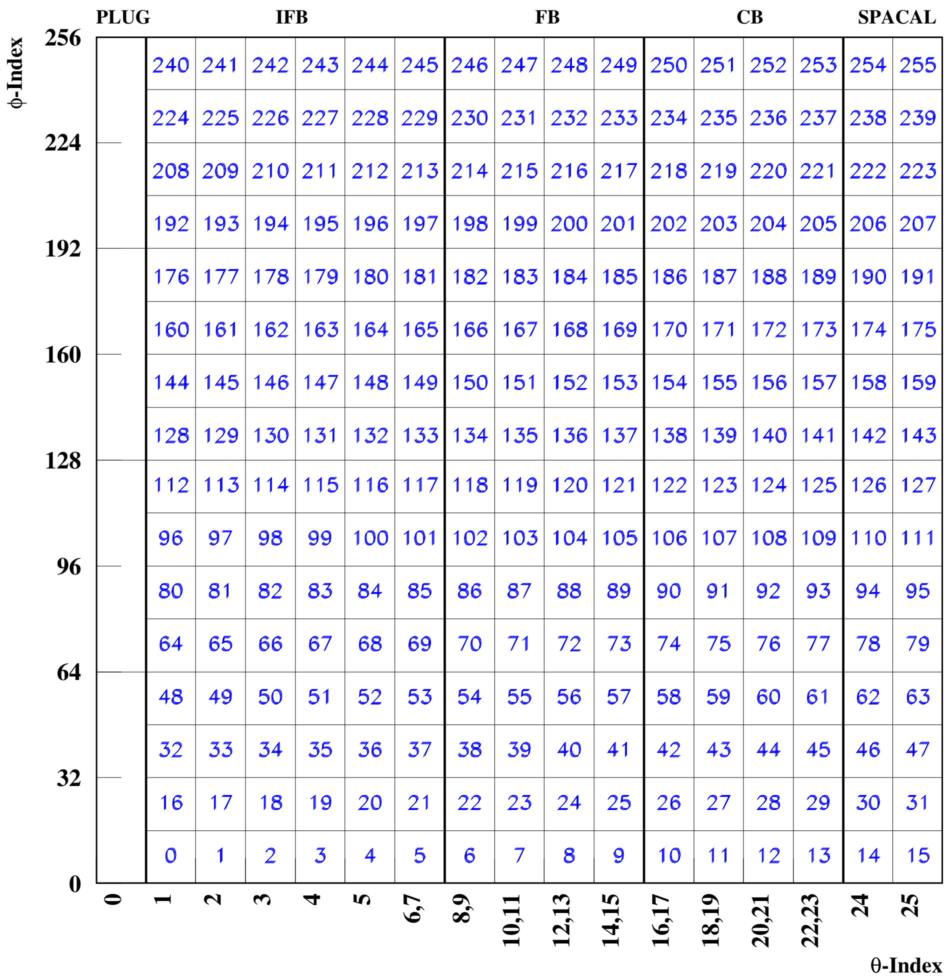


Abbildung 3.4: Abdeckung der θ-ϕ-Ebene durch *big rays*. Die Zahlen in den Feldern sind die Nummern der *big rays* (nach [19]).

Bei einem Bitfeld ist im Vergleich zu einem Bytefeld ein zusätzlicher Schritt nötig, um die Zentren der Cluster zu finden: Für jede Zelle mit einem gesetzte Bit im Bitfeld wird die Zahl der in benachbarten Zellen gesetzten Bits zum Inhalt der Zelle hinzuaddiert. Man erhält ein Bytefeld, ähnlich dem der *big tower* Energien, das man als Bytefeld von Pseudoenergien bezeichnen kann und auf das dann der bei den *big towers* vorgestellte Algorithmus angewendet werden kann. Ein Beispiel für die Funktionsweise dieses Algorithmus wird in Abbildung 3.5 gegeben. Durch die einfache Struktur der *big ray* Mappe kann zur Definition der Nachbarschaften auf Listen verzichtet werden.

Zu jedem Cluster werden folgende Informationen gespeichert: Position, Pseudoenergie, Pseudoenergie im Zentrum des Clusters, Pseudoenergie im Ring, sowie

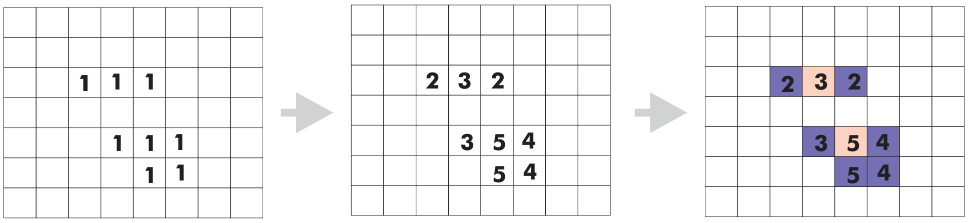


Abbildung 3.5: Bildung von Clustern in der Proportionalkammer. Dargestellt ist ein Ausschnitt der *big ray* Mappe: Links: Verfügbare Daten: Mitte: Nach der Bildung der Summen über die Nachbarzellen; rechts: Die gefundenen Cluster. Die Clusterzentren sind hell markiert, die Randbereiche dunkel.

die Zahl der Zellen, die der Cluster umfasst. Pseudoenergie ist nicht die Energie eines Teilchens im herkömmlichen Sinne. Diese Information lässt sich naturgemäß aus einer Proportionalkammer nicht gewinnen. Pseudoenergie ist, wie oben bemerkt, die Größe, die man als Energie des Clusters erhält, wenn man das nach dem Zwischenschritt erhaltene Bytefeld als Energieverteilung ansieht.

Dies führt zu einigen ungewohnten Eigenschaften dieser Pseudoenergie. Zum Beispiel hängt ihr Wert außer von der Größe eines Clusters auch von seiner Form ab: Drei gesetzte Bits in L-Form ergeben eine höhere Pseudoenergie als drei Bits in einer Reihe. Zudem sind durch diesen Algorithmus nur bestimmte Werte (also nicht jeder einzelne Wert von 0 bis 255) für die Pseudoenergie möglich.

3.4.4 Driftkammern

Von den Zentralen Driftkammern (CJC) werden als Triggerinformation für L2 vier Felder mit je 45 Bits zur Verfügung gestellt [22, 17].

Die vier Felder sind äquivalent zu vier verschiedenen Maskenfeldern in der CJC, die auf Spuren positiv bzw. negativ geladener Teilchen mit hohem ($p_t >$ 800 MeV c^{-1}) bzw. niedrigem (400 MeV $c^{-1} < p_t <$ 800 MeV c^{-1}) transversalem Impuls sensitiv sind. Die 45 Bits entsprechen 45 Winkelsegmenten bezüglich ϕ. In θ-Richtung separiert die CJC auf dem Triggerniveau nicht.

Getrennt für die vier Maskentypen werden aus aufeinanderfolgenden gesetzten Bits Cluster gebildet. Die Position eines Clusters liegt in der Mitte der Bitfolge. Falls ein Cluster aus einer geraden Zahl von Sektoren besteht, hat der größere ϕ-Index Vorrang. Die Pseudoenergie ist hier die Zahl der Sektoren, aus denen der Cluster besteht. Selbstverständlich wird berücksichtigt, dass die CJC einen geschlossenen Ring bildet. Die Funktion des Algorithmus wird in Abbildung 3.6 verdeutlicht.

Die CJC ist diejenige Detektorkomponente, die die genaueste Winkelinformation in ϕ-Richtung liefert. Deshalb bezieht ein Objekt (vgl. Abschnitt 3.5), das einen CJC-Cluster enthält, von diesem seine Positionsinformation bezüglich ϕ. In θ-Richtung steht keine Information zur Verfügung, so dass ein Objekt, das nur

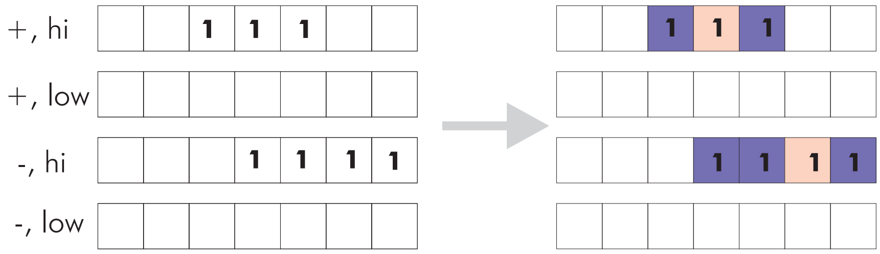


Abbildung 3.6: Bildung von Clustern in der CJC. Dargestellt ist ein Ausschnitt der CJC. Links: Verfügbare Daten für die vier Maskentypen; rechts: Die gebildeten Cluster. Die Zentren der Cluster sind hell markiert, die Randbereiche dunkel.

aus einem CJC-Cluster besteht, keine θ-Information besitzt.

Es wird versucht, die Ladung eines Objekts aus den aktiven Maskentypen zu bestimmen: Wenn mehr Masken für negative Teilchen als für positive ansprechen, wird die Ladung des Objekts als negativ angenommen, im anderen Fall als positiv. Objekte ganz ohne CJC-Masken gelten als neutral.

3.4.5 Myonkammern

Die Triggerhardware des Zentralen Myonsystems ist in 64 Module unterteilt. Ihre räumliche Anordnung ist aus Abbildung 3.7 ersichtlich, die den einzelnen Modulen zuzuordnenden Bereiche der θ-ϕ-Ebene aus Abbildung 3.8. Die Granularität der Myontrigger entspricht diesen Modulen und ist somit relativ grob. Deshalb besteht kein Anlass, bei diesem Subdetektor ein *clustering* durchzuführen.

Jeder Hit in den Myonkammern bildet für sich einen Cluster mit der Pseudoenergie 1. ϕ und θ werden durch die Mitte des getroffenen Segments festgelegt. Die Winkelinformation ist dadurch besonders bei der Bestimmung von θ im *barrel*-Bereich unpräzise. Falls beim *matching* eine Kombination mit einem anderen Objekt erfolgt, wird θ mit dessen Daten präzisiert.

3.4.6 SpaCal

Triggerinformationen vom SpaCal werden von unterschiedlichen Triggersystemen zur Verfügung gestellt: Zum einen liefern die vier, jeweils in einen inneren und einen äußeren Bereich aufgespaltenen Quadranten des SpaCals Daten über die in ihnen deponierte Energie in Form von acht *big towers* (s. Abbildung 3.9 rechts). Diese werden zusammen mit den anderen Kalorimeter-Daten behandelt (s. Abschnitt 3.4.1).

Außerdem existiert ein höher auflösender Trigger, der SpaCal IET (*inclusive electron trigger*) [24]. Er soll Elektronen, deren Energie eine von drei Schwellen überschreitet, mit sehr hoher Effizienz triggern [25]. Übliche Werte für die Schwellen sind $0,5$ GeV, $2,0$ GeV und $6,0$ GeV.

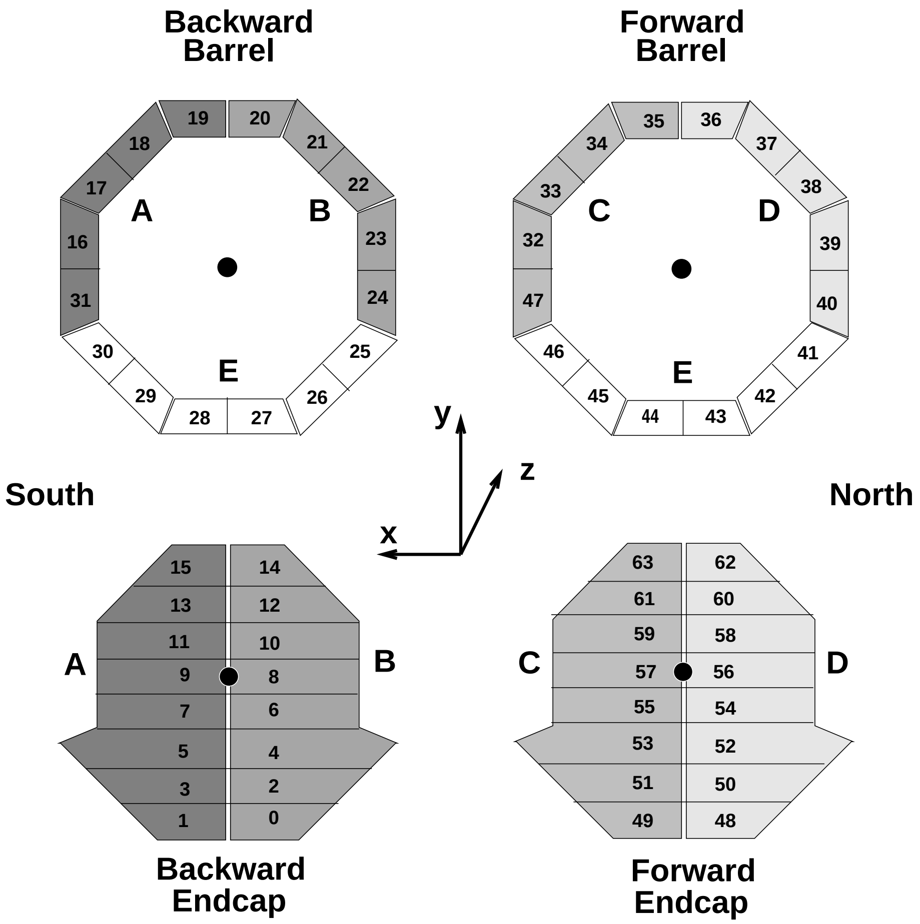


Abbildung 3.7: Räumliche Anordnung der Myonkammer-Module (aus [23]).

Vom Standpunkt des IET aus betrachtet, ist das SpaCal in *towers* aufgeteilt, die jeweils von vier SpaCal-Kanälen gebildet werden. Um auch Elektronen, die ihre Energie an der Grenze zweier SpaCal *towers* deponieren, effizient zu triggern, werden durch Summierung über je vier benachbarte SpaCal *towers* SpaCal *trigger towers* gebildet. Falls die Summe eine der Schwellen überschreitet, wird der Trigger ausgelöst. Diese Methode wird als „*sliding window*" bezeichnet (s. Abbildung 3.9 links).

Der IET liefert Daten vom elektromagnetischen SpaCal in zwei unterschiedlichen Granularitäten. Das feine Raster besteht aus 20 auf 20 SpaCal *trigger towers*, das grobe aus den 5 auf 5 LIET Karten (*local IET*), auf die die Hardware des IET aufgeteilt ist (vgl. Abbildung 3.10).

Wenn in Bitfeldern, die mit dem „*sliding window*" Algorithmus erstellt wurden, nach Clustern gesucht wird, sind die Pseudoenergien der Cluster nicht besonders aussagekräftig: Im ungünstigsten Fall unterscheiden sich zwei Cluster, die von Elektronen gleicher Energie erzeugt wurden, in der Gesamtpseudoenergie

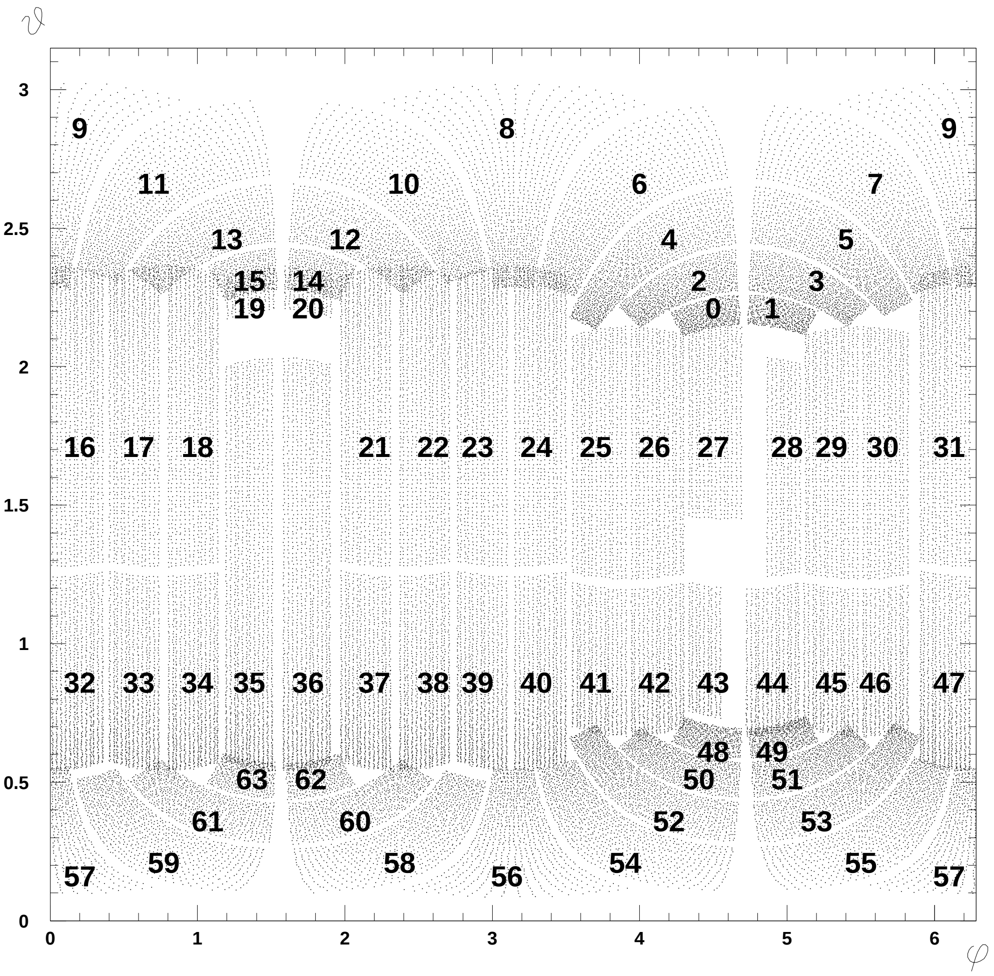
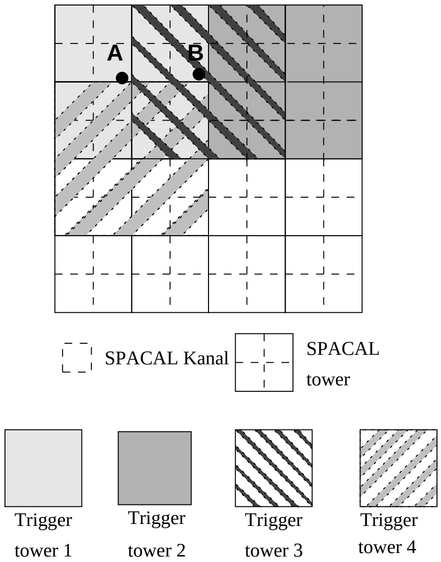


Abbildung 3.8: Bedeckung der θ-ϕ-Ebene durch Module des Zentralen Myondetektors (aus [23]). Deutlich zu sehen sind die Löcher für den Kryostat des Flüssig-Argon-Kalorimeters.

um den Faktor vier, je nachdem, ob das Elektron seine Energie in der Mitte eines oder am Berührungspunkt von vier SpaCal *towers* deponiert.

In der groben Granularität (LIET Karten) stehen Informationen über alle drei Schwellen (1,2,3) zur Verfügung. Diese Informationen werden aus denen der feinen Granularität durch logische inklusive ODER-Verknüpfungen gewonnen. Daten der feinen Granularität (SpaCal *trigger towers*) sind von der zentralen LIET Karte vollständig und von den anderen Karten nur für die mittlere Schwelle (2) verfügbar [26].

Ein Teil der Information wird vor dem *clustering* aus dem groben Raster in das feine übertragen: Dabei wird davon ausgegangen, dass wegen der starken Winkelabhängigkeit des Wirkungsquerschnitts näher am Strahl liegende Bereiche



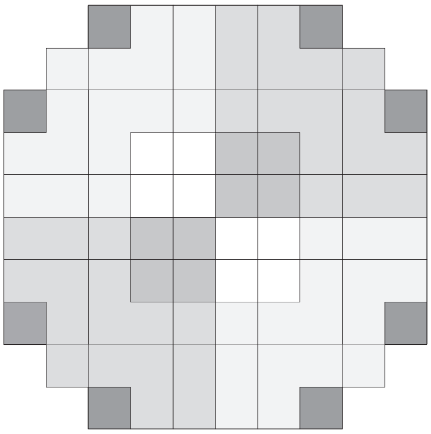


Abbildung 3.9: Links: Überlapp der SpaCal *trigger towers* („*sliding window*"). Für Elektronen, die das SpaCal am Punkt A treffen, findet man die Energie vollständig im SpaCal *trigger tower* 1. Elektronen, die am Punkt B auftreffen, deponieren ihre Energie sowohl in *trigger tower* 1 als auch 2. Die vollständige Energie ist aber in *trigger tower* 3 ablesbar. *Trigger tower* 4 zeigt, dass diese Methode in beiden Dimensionen angewendet wird (aus [25]). Rechts: Zuordnung der SpaCal *towers* zu acht *big towers*. Die äußeren (dunkelsten) Bereiche sind in keinem *big tower* enthalten (aus [24]).

(„innen" im SpaCal) mit höherer Wahrscheinlichkeit getroffen werden, als weiter außen liegende. Deshalb wird folgender Algorithmus angewendet: Wenn die niedrigste Schwelle (1) auf einer LIET Karte überschritten wird, wird der innerste SpaCal *trigger tower* auf dieser Karte entsprechend gesetzt. Wenn auf einer LIET Karte die höchste Schwelle (3) überschritten wird, wird der innerste SpaCal *trigger tower* der Karte, der die mittlere Schwelle (2) überschritten hat, auf die höchste Schwelle (3) hochgesetzt. In beiden Fällen gilt: Falls sich zwei SpaCal *trigger towers* innerhalb einer Karte gleich weit vom Mittelpunkt des SpaCals entfernt befinden, wird der Algorithmus auf beide angewendet.

SpaCal *trigger towers* liegen in der Dynamik mit ihren vier Werten (drei Schwellen) zwischen den *big towers* (256 Werte) und den *big rays* (zwei Werte). Deshalb wird auf das Feld mit den 2-Bit-Werten, die die Zahl der in einem Kanal überschrittenen Schwellen angeben, eine Kombination beider bisher vorgestellter Algorithmen angewendet. Damit orientiert sich die Bestimmung des Clusterzentrums sowohl an der Umrandung des Clusters (wie bei den *big rays*), als auch an lokalen Maxima in der Energieverteilung (wie bei den *big towers*). Der Algorithmus arbeitet analog zu dem, der auf die *big rays* angewendet wird. Bei der Summierung über die Nachbarn einer Zelle wird der Wert in der zentralen Zelle

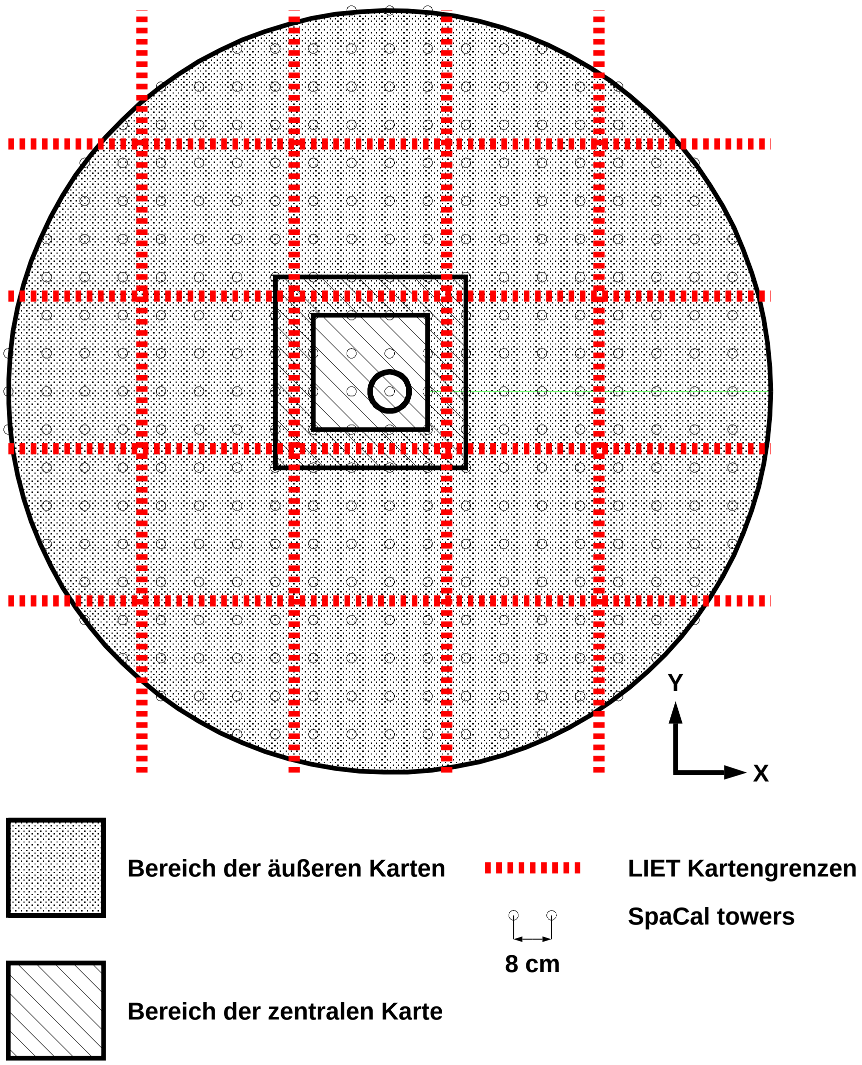


Abbildung 3.10: Gliederung des IET im elektromagnetischen SpaCal.

mit einem höheren Gewicht als eins[3] mit aufsummiert. Für ein beliebig hohes Gewicht geht der Algorithmus in den über, der auf die *big towers* angewendet wird. Analog zu Abschnitt 3.4.3 wird auch eine Pseudoenergie der Cluster definiert.

Die Funktionsweise des gesamten Algorithmus wird in Abbildung 3.11 veranschaulicht.

Die Position von Clustern im SpaCal wird durch den Winkel ϕ und den Abstand zum Strahlrohr (Radius) festgelegt. Außerdem werden die Gesamtpseudoenergie des Clusters, die Pseudoenergie im Clusterzentrum, die Pseudoenergie im Ring, sowie die Zahl von SpaCal *trigger towers*, aus denen der Cluster besteht, gespeichert.

[3]Im Augenblick wird ein empirisch bestimmtes Gewicht von 2 verwendet.

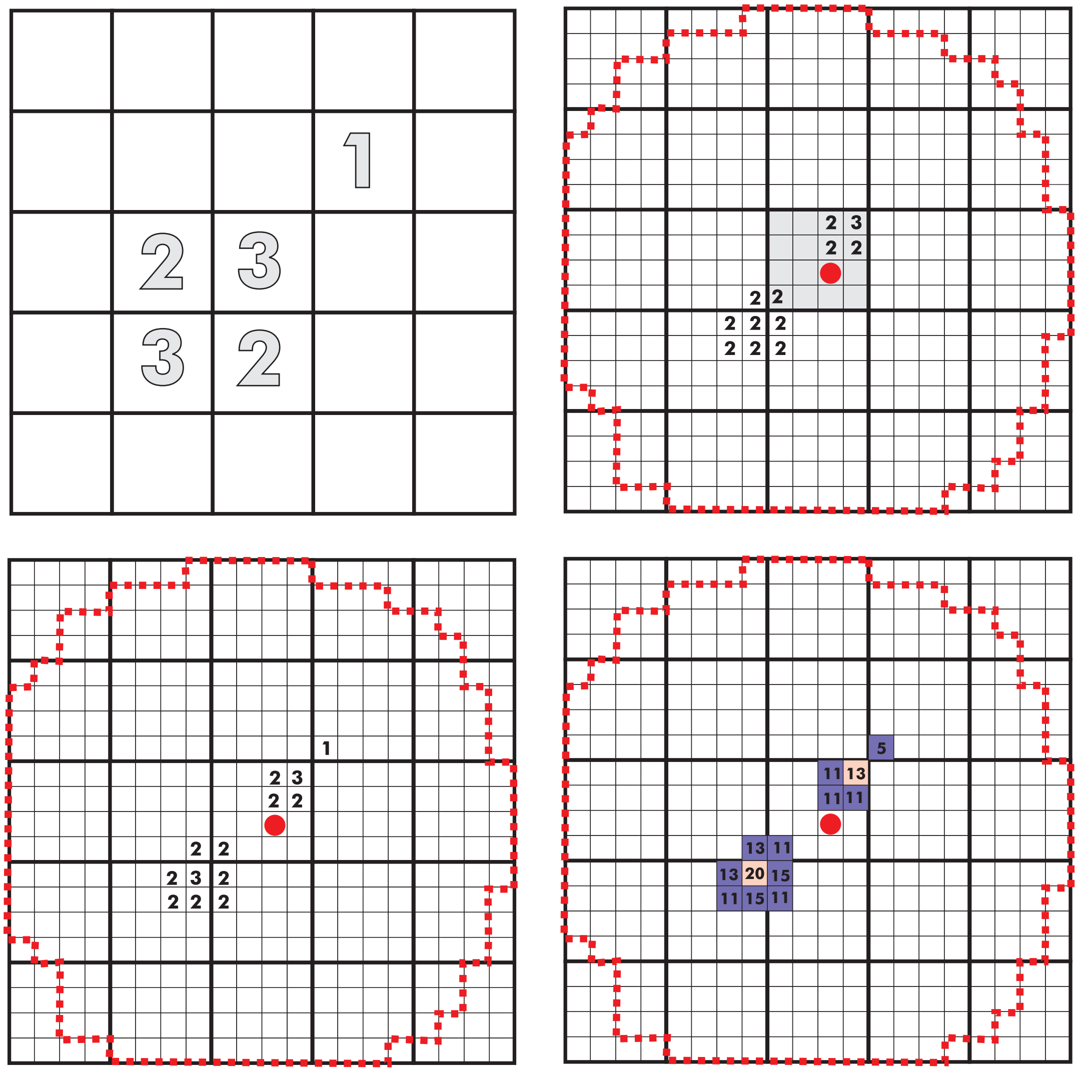


Abbildung 3.11: Bildung von Clustern im SpaCal: Verfügbare Information aus dem groben (links oben) und feinen (rechts oben) Raster. Im feinen Raster sind im Bereich der zentralen Karte (schattiert) ebenso drei Schwellen verfügbar wie im gesamten groben Raster. Teile des Rasters liegen außerhalb der geometrischen Akzeptanz des SpaCals, die durch die gepunktete Linie markiert wird. Außerhalb dieser Linie sind die gelieferten Daten immer Null. Zuerst werden die Daten, wie im Text beschrieben, vom groben Raster in das feine übertragen (links unten). Dann wird für jede Zelle die Summe über diese Zelle und alle Nachbarn gebildet. Die Zelle selbst wird mit dem Faktor zwei gewichtet (rechts unten). Die gefundenen Clusterzentren sind hell markiert, die Randbereiche dunkel.

Die Pseudoenergie der Cluster ist selbstverständlich nicht linear von der deponierten Energie abhängig: Erstens sind bereits die Werte 0, 1, 2 und 3, die angeben, welche Schwelle überschritten wurde, nicht linear von der Energie abhängig. Zweitens tritt der oben beschriebene Effekt durch das „*sliding window*“ Verfahren auf und drittens ist natürlich wieder auf den Unterschied zwischen Energie und Pseudoenergie (s. Abschnitt 3.4.3) hinzuweisen.

3.5 *matching*

Beim *matching* werden Cluster aus mehreren Subdetektoren zu physikalischen Objekten kombiniert [18, 27]. Ein Objekt besteht beispielsweise aus einem Cluster im Flüssig-Argon-Kalorimeter, einem Signal im instrumentierten Eisen und einer aktiven Maske für negativ geladene Teilchen mit hohem Transversalimpuls in der CJC. Die Eigenschaften der Cluster, aus denen das Objekt aufgebaut ist, werden Elemente des Objekts.

Informationen, die in das *matching* eingehen, sind zurzeit verfügbar vom elektromagnetischen und hadronischen Teil der Kalorimeter (wahlweise *big towers* oder *trigger towers*, für Testzwecke auch beide Granularitäten zusammen), von den Proportionalkammern (*big rays*), von den Myonkammern und von den vier verschiedenen Maskentypen der CJC. Letztere werden beim *matching* als eigenständige Subdetektoren behandelt. Information vom SpaCal wird getrennt von den anderen Subdetektoren behandelt, da der abgedeckte Raumwinkelbereich ein anderer ist.

Da die verschiedenen Subdetektoren unterschiedliche Granularitäten besitzen, wurde als gemeinsames Koordinatensystem das schon zuvor vorgestellte System von ϕ- $(0 \dots 255)$ und θ-Index $(0 \dots 25)$ eingeführt. Damit ist eine globale Granularität definiert.

Cluster in verschiedenen Subdetektoren können zu einem Objekt kombinierbar sein, obwohl sie nicht die exakt gleichen Koordinaten haben. Ursache der Abweichungen sind unter anderem die durch das Magnetfeld gekrümmten Teilchenbahnen und die Fluktuationen innerhalb der die Cluster erzeugenden Schauer. Bei der Kombination von Clustern unterschiedlicher Detektorkomponenten zu einem Objekt werden deshalb Abweichungen der Koordinaten von einer Einheit in θ- und ϕ-Richtung, bezogen auf die Granularität der Detektorkomponente, toleriert.

Es werden maximal 16 Objekte aus Clustern in allen Subdetektoren außer dem SpaCal gebildet. Aus den Clustern im SpaCal werden ebenfalls 16 Objekte gebildet, die getrennt von den anderen behandelt werden, da sich durch die unterschiedlichen abgedeckten Raumwinkel kaum Möglichkeiten ergeben, Cluster im SpaCal mit Clustern in anderen Subdetektoren zu kombinieren.

Um Objekte zu finden, wird jeweils ein Cluster in einem Subdetektor als Ausgangspunkt genommen und in anderen Subdetektoren nach Kombinationspartnern für diesen Cluster gesucht. Die Subdetektoren werden in einer empirisch bestimmten Reihenfolge abgearbeitet: Als erstes werden Kombinationsparter für die Cluster im hadronischen Teil des Kalorimeters gesucht. Danach kommen die Cluster im elektromagnetischen Teil des Kalorimeters, in der MWPC, in den vier CJC Maskenfeldern und den Myonkammern an die Reihe. Da eine Verwendung von Clustern in mehreren Objekten unphysikalisch und daher nicht gewünscht ist, werden bei jedem Schritt nur Cluster betrachtet, die noch keinem Objekt zugeordnet sind[4].

Nach Kombinationspartnern wird nur für eine maximale Zahl von Clustern pro Subdetektor gesucht. Diese Zahl ist für jeden Subdetektor individuell festlegbar. Nicht berücksichtigt werden die (pseudo-)energieärmsten Cluster, die nur dann in einem Objekt verwendet werden, wenn sie als Kombinationspartner eines ausreichend großen Clusters in einem anderen Subdetektor dienen. Damit wird vermieden, dass beispielsweise viele Objekte gebildet werden, die nur aus einem rauschenden *trigger tower* bestehen, derentwegen dann für interessantere Objekte ohne Signatur in einem *trigger tower* kein Platz mehr ist (vgl. Anhang A.2.2).

Um diesen Algorithmus in einer vom Zeit- und Ressourcenbedarf der Hardware her adäquaten Art durchzuführen, müssen einige Kunstgriffe angewendet werden (s. auch Abbildung 3.12):

Es werden Mappen der oben definierten globalen Granularität eingeführt, die den gesamten Bereich von θ- und ϕ-Index abdecken. Für jeden Subdetektor existiert eine derartige Mappe. Die globale Granularität wird so fein gewählt, dass die Granularitäten aller Detektorkomponenten darauf abgebildet werden können. Jeder in einem Subdetektor gefundene Cluster wird in die Mappe dieses Subdetektors eingetragen, und zwar sowohl in die Zellen, die dem Clusterzentrum entsprechen, als auch in alle Zellen, die, bezogen auf die Granularität des Subdetektors (nicht auf die globale) benachbart sind. Wenn dann zu einem Cluster C im Subdetektor $S1$ Kombinationspartner im Subdetektor $S2$ gesucht werden sollen, muss die Position von C nicht mit den Positionen aller in $S2$ gefundenen Cluster verglichen werden (unter Berücksichtigung erlaubter Abweichungen, deren Größe vom Subdetektor und der Lage der Cluster im Detektor abhängen kann). Es genügt hingegen, festzustellen, ob in der Zelle der Mappe des Subdetektors $S2$, die sich an der Position von C befindet, ein Cluster eingetragen wurde.

Falls bei einem Subdetektor $S3$ (beispielsweise den Myonkammern) die Granularität sehr grob ist, wird die Zahl der Zellen in der globalen Granularität, in die ein Cluster in $S3$ eingetragen werden muss, sehr groß. Falls dies aus zeitlichen Gründen nicht durchgeführt werden kann, kann der Algorithmus erweitert werden: Es wird bei der Suche nach Kombinationspartnern für C in diesem Subdetektor $S3$ nicht nur die eine Zelle überprüft, die sich an der Position von C

[4]Die Ausnahme sind Cluster in einem der vier Maskenfelder der CJC. Da bei diesen keine θ-Position bestimmt werden kann, wird eine Verwendung in mehreren Objekten zugelassen.

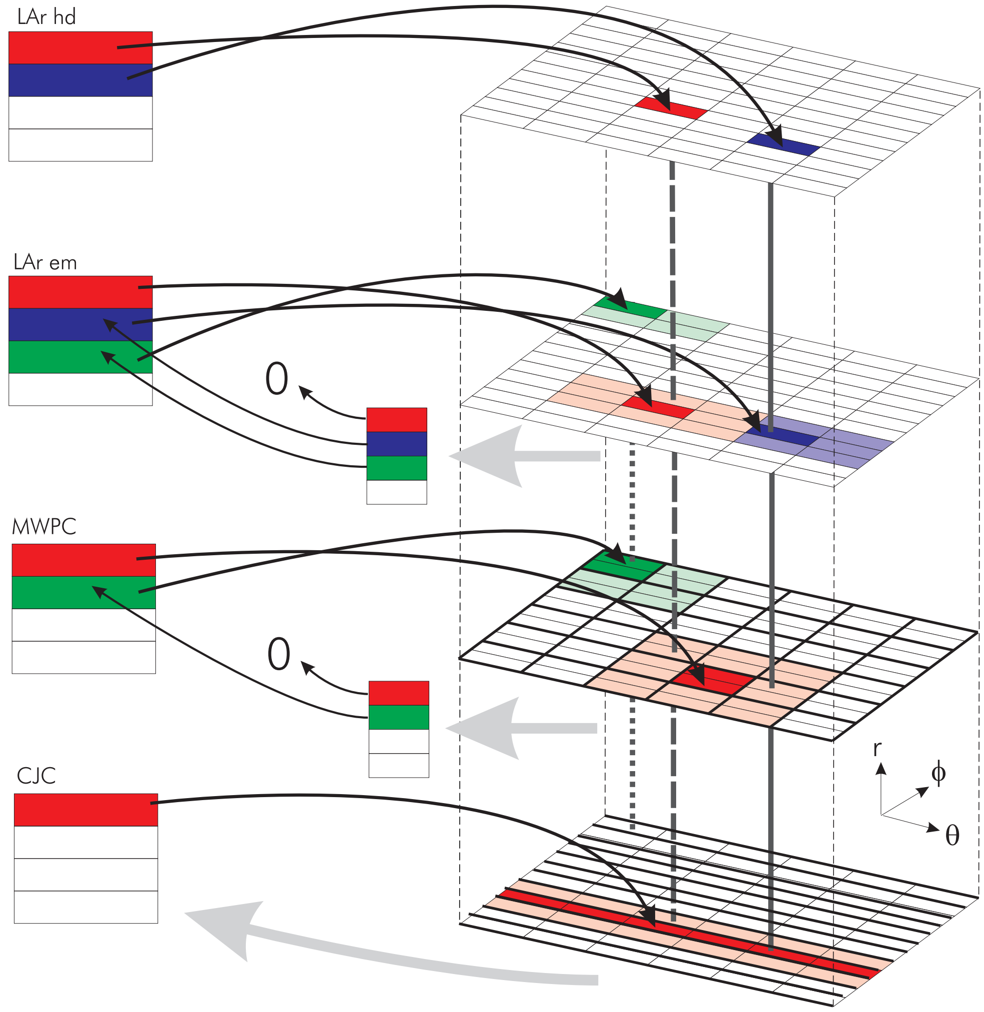


Abbildung 3.12: Algorithmus zur Bildung von Objekten aus Clustern unterschiedlicher Subdetektoren: Dargestellt sind rechts in vier Ebenen die Mappen von vier Subdetektoren, die anderen Subdetektoren werden analog behandelt. Bei den Mappen von MWPC und CJC ist der Unterschied zwischen der feinen globalen Granularität und der gröberen Granularität des Subdetektors erkennbar. In die Mappen wurde die Position aller gefundenen Cluster eingetragen, und zwar sowohl die Clusterzentren (dunkle Farben), als auch die Nachbarschaften der Cluster (helle Farben). Die Nachbarschaften geben den Bereich an, in dem der Cluster zum Kombinieren verwendet werden kann. Links sind die „Tabellen" mit den Clustern in den Subdetektoren dargestellt, in der Mitte der Skizze die Auflösung der Indizes (rot, blau, grün) durch eine „Liste". Dargestellt ist folgende Situation: Es wurde, ausgehend vom größten Cluster in der obersten Ebene (LAr hd), ein Objekt gebildet (gestrichelte Linie). Die verwendeten Cluster (außer dem CJC-Cluster) wurden gelöscht. Die zu den entsprechenden Indizes gehörenden Listeneinträge haben deshalb keinen Inhalt mehr (sie weisen auf Null). Bei der Bildung des nächsten Objekts, ausgehend von zweitgrößten Cluster in der obersten Ebene, (durchgezogenen Linie) würde deshalb kein Cluster aus der MWPC verwendet, obwohl er von der Position her passen würde. Als drittes und letztes Objekt käme, ausgehend vom verbleibenden Cluster in der zweiten Ebene (LAr em), das mit der gepunkteten Linie markierte an die Reihe.

befindet, sondern auch deren acht Nachbarn in der globalen Granularität. Dann müssen beim Einlesen der Cluster in die Mappe von Subdetektor $S3$ nicht mehr so viele Zellen gefüllt werden: Es genügt, wenn genau die normalerweise gefüllten Zellen mindestens einen gefüllten Nachbarn in der globalen Granularität haben.

Wie oben bemerkt, ist eine Mehrfachverwendung von Clustern in verschiedenen Objekten nicht gewünscht. Bereits verwendete Cluster müssen deshalb aus der Mappe gelöscht werden, damit sie für den weiteren *matching*-Prozess nicht mehr zur Verfügung stehen. Eine effektive Methode dafür ist die folgende: In die Zellen der Mappe werden nicht die Eigenschaften des Clusters, der sich an der entsprechenden Stelle befindet, eingetragen, sondern nur Indizes, die durch eine Liste aufgelöst werden. Um einen Cluster als benutzt zu markieren, müssen so nicht in mehreren Schritten alle Zellen des Clusters gelöscht werden. Es genügt, die Eigenschaften, die dem Index durch die Liste zugeordnet werden, auf Null zu setzen.

Die Suche nach Kombinationspartnern zu einem Cluster C in einem Subdetektor $S1$ kann in allen anderen Subdetektoren parallel durchgeführt werden. Außerdem wird, wie bereits oben angemerkt, die Zahl der insgesamt gebildeten Objekte auf 16 limitiert. Zusammengenommen führt dies zu einem akzeptablen Zeitverhalten des Algorithmus: Die Zahl der Operationen zum Füllen der Mappen ist linear von der Zahl der auftretenden Cluster abhängig und durch die maximale Zahl der von jedem Subdetektor zur Verfügung gestellten Cluster nach oben beschränkt. Die Zahl der Operationen zur Bildung von Objekten ist durch die maximale Zahl von Objekten nach oben beschränkt und (im Wesentlichen) nicht von der Zahl der auftretenden Cluster oder der Zahl der Subdetektoren abhängig.

Zu den vorhandenen Subdetektoren werden nach dem Upgrade von H1 weitere Komponenten hinzukommen, die Daten mit θ- und ϕ-Information für L2 bereitstellen. Hier seien nur der Jet-Trigger [28], der bereits auf L1 den hier verwendeten Clustern sehr ähnliche „Jets“ aus den Flüssig-Argon-Kalorimeter *trigger towers* bildet, und der *Fast Track Trigger* (FTT) [29, 30], der Informationen über Position und Krümmung von Spuren liefert, genannt. Allein die Tatsache, dass deren Spezifikationen teilweise noch unklar sind, führt dazu, dass das *matching* als der Punkt, an dem die Datenströme der Subdetektoren zusammenfließen, flexibel gegenüber Änderungen und Erweiterungen sein muss, indem beispielsweise die Möglichkeit offengehalten wird, weitere Subdetektoren hinzuzufügen und die Kommunikation des *matching*-Algorithmus mit den *clustering*-Algorithmen der Subdetektoren standardisiert angelegt wird: Jeder Cluster liefert zum einen die Positionsinformation, die vom *matching*-Algorithmus benutzt wird, um den Cluster in die Mappe einzutragen, zweitens eine Information, anhand derer entschieden wird, ob der Cluster so (pseudo-)energiereich ist, dass in anderen Subdetektoren nach Kombinationspartnern für ihn gesucht wird und drittens die restlichen Eigenschaften des Clusters, die unverändert in ein gebildetes Objekt eingebracht werden.

3.6 *ordering*

Die beim *matching* erstellten Objekte reichen von einem Objekt mit einem einzigen rauschenden *trigger tower* bis hin zu einem mit Myonsignatur, großer Energiedeposition im Flüssig-Argon-Kalorimeter, Signal in der Proportionalkammer und Clustern in mehreren Maskentypen der CJC. Um die weniger relevanten Objekte von den wichtigen zu trennen, werden die Objekte sortiert [18]. Es sind drei Sortierungen gleichzeitig verfügbar. Damit können beispielsweise das Objekt mit dem kleinsten θ-Index und das Objekt mit der höchsten Energie im Flüssig-Argon-Kalorimeter dem gleichen Netz Eingabegrößen liefern.

Die Zahl von Elementen, die ein Objekt haben kann, ist hoch. Es sind dies vor allem die Positions-Information des Objekts, sowie die Eigenschaften der Cluster, die in das Objekt eingebracht wurden (s. Anhang A.1 für eine vollständige Auflistung aller möglichen Elemente). Die Objekte können nach beliebigen Elementen sortiert werden, entweder in auf- oder absteigender Reihenfolge. Statt einem Element kann auch das Signum[5] eines Elements als Sortierkriterium verwendet werden. Um auch dabei eine eindeutige Reihenfolge sicherzustellen, wird immer nach zwei Kriterien sortiert. Das zweite Kriterium wird benutzt, wenn zwei Objekte bezüglich des ersten Kriteriums gleich sind. Durch eine derartige Sortierung mit zwei Kriterien kann zum Beispiel das bezüglich des Flüssig-Argon-Kalorimeters energiereichste Objekt ausgewählt werden, das gleichzeitig einen Hit in der MWPC hat. Die SpaCal-Objekte werden analog dazu ebenfalls auf drei Arten mit je zwei Kriterien sortiert, unabhängig von den anderen Objekten. Weitere Informationen über die spezifische Steuerung des *orderings* findet man im Anhang A.2.2.

Falls dies aus Zeitgründen erforderlich ist, ist es vorstellbar, nur die nach der Sortierung ersten n der 16 möglichen Objekte zu sortieren und die Objekte $n+1$ bis 16 unsortiert zu belassen.

Als Eingabegrößen stehen dann letztendlich Größen wie „*Energie im Ring des Clusters im hadronischen Teil des Flüssig-Argon-Kalorimeters bei dem Objekt, das, wenn alle Objekte absteigend nach ihrer Energie im gesamten Flüssig-Argon-Kalorimeter sortiert werden, an zweiter Stelle steht*“ zur Verfügung. Die vollständige Liste aller verfügbaren Eingabegrößen ist in Anhang A.1 aufgeführt, ebenso wie ihre Nomenklatur.

3.7 *post-processing*

Der letzte Schritt der Datenvorverarbeitung ist das *post processing* [18]. Möglich sind im Prinzip alle Berechnungen, die in der zur Verfügung stehenden Zeit mit den gebildeten Objekten durchgeführt werden können.

[5] 1 für $x > 0$, -1 für $x < 0$, 0 sonst

Die DDB2 verwendet zwei Arten der Nachverarbeitung: Zum einen werden Winkeldifferenzen (bezüglich θ- oder ϕ-Index) zwischen dem jeweils ersten Objekt in einer der sortierten Listen und den folgenden gebildet. Damit ist es zum Beispiel möglich, den Winkel zwischen den beiden energieintensivsten Objekten zu betrachten. Eine typische Anwendung ist die Erkennung eines ϕ-Mesons, das unter bestimmten energetischen Voraussetzungen in zwei Kaonen zerfällt, die damit einen bestimmten Zwischenwinkel haben. Die Winkeldifferenzen stehen direkt als Eingabegrößen zur Verfügung. Zur Nomenklatur möge man wiederum Anhang A.1 heranziehen.

Der andere Ansatz besteht darin, Objekte zu zählen, deren Elemente bestimmte Bedingungen erfüllen (*counting*). Ein Beispiel ist die Zahl von Objekten mit Elektronsignatur: Die Bedingung dafür lautet: Es existiert ein Cluster im elektromagnetischen Teil des Flüssig-Argon-Kalorimeters über einer bestimmten Energieschwelle und einer im hadronischen Teil, der eine bestimmte Energie nicht übersteigt. Auch die Zahl der insgesamt gebildeten Objekte lässt sich auf diese Weise als Eingabegröße verwenden. Zu den — ziemlich implementierungsspezifischen — Einzelheiten der dafür notwendigen Bedingung wird auf den Anhang A.2.2 verwiesen. In der Simulation stehen 32 verschiedene Zähler zur Verfügung. Sie sind in gewissen Grenzen frei programmierbar (vgl. Anhang A.1 und A.2.2).

Weder das Berechnen von Differenzwinkeln noch das Zählen von Objekten, deren Elemente bestimmte Bedingungen erfüllen, ist speicher- oder zeitintensiv, auch weil für das Zählen keine sortierten Listen benötigt werden und dies somit parallel zum *ordering* erledigt werden kann.

Selbstverständlich sind noch weitere Arten des *post processing* vorstellbar und technisch realisierbar, die aber keinen Eingang in die Simulation gefunden haben:

- Die Definition der Bedingungen an die Elemente beim *counting* kann komplexer werden.
- Statt Objekte, deren Elemente bestimmte Bedingungen erfüllen, nur zu zählen, kann über ihre Clusterenergien summiert werden.
- Die Berechnung von Winkeldifferenzen kann auf Objekte aus unterschiedlichen Sortierungen erweitert werden.

3.8 Behandlung der übrigen Daten

Mit der DDB2 stehen auch die Triggerinformationen als Eingabegrößen zur Verfügung, die nicht direkt im neuen Konzept verwendet werden. Dazu gehören vor allem das z-Vertex-Histogramm mit seinen abgeleiteten Eingabegrößen (Position und Wert des Maximums im z-Vertex-Histogramm, Summe über den Inhalt aller z-Bins im z-Vertex-Histogramm) und die vom Flüssig-Argon-Kalorimeter-Trigger berechneten und teilweise gewichteten Energiesummen (E_t, E_x, E_y, Energiedeposition in IF, FB oder CB). Andere Größen (wie die Elektronsignatur im

Flüssig-Argon-Kalorimeter) können ersetzt und besser an den Einzelfall angepasst werden (zum Beispiel: für unterschiedliche Physikklassen unterschiedliche Energieschwellen).

3.9 Beispiel für intelligente Datenvorverarbeitung

Abschließend soll die Funktionsweise von *clustering* und *matching* an einem realen Beispiel dargestellt werden. Dazu wurde ein Ereignis ausgewählt, bei dem ein Elektron und ein Jet nachgewiesen werden kann. Es ist in Abbildung 3.13 dargestellt. Außer dem SpaCal liefern in diesem Ereignis alle Triggerkomponenten, die im Konzept der DDB2 Verwendung finden, ein Signal. Zunächst sollen diese Signale vorgestellt werden:

Die aktiven *big rays* in der Proportionalkammer sind in Abbildung 3.14 dargestellt, die aktiven hadronischen und elektromagnetischen *big towers* des Kalorimeters in den Abbildungen 3.15 und 3.16. Diese drei Abbildungen zeigen Debug-Ausgaben der in dieser Arbeit erstellten Simulation. Dargestellt sind die Inhalte der Bit- und Bytefelder der *big rays* und *big towers* in der θ-ϕ-Ebene. Die Anordnung der Zellen entspricht der in den Abbildungen 3.1 und 3.4. Jeweils am unteren Rand der Darstellungen befindet sich eine Liste aus Wertepaaren. Jedes Wertepaar entspricht einem gefundenem Cluster. Die zweite Zahl in einer Klammer gibt die Nummer des *big towers* oder *big rays* an, der das Clusterzentrum bildet, die erste Zahl die gesamte (Pseudo-)Energie des Clusters. Die Cluster sind mit verschiedenen Farben markiert. Cluster mit derselben Farbe werden beim *matching* kombiniert[6]. Zusätzlich zu den Clustern in den bis jetzt angesprochenen Detektorkomponenten existiert in der CJC noch eine Clustergruppe, die aus Clustern mehrerer Maskentypen besteht, sowie ein alleinstehender Cluster aus zwei aktiven Masken für negativ geladene Teilchen mit hohem Transversalimpuls. In den Myonkammern wurde ein Treffer nachgewiesen. Diese und alle anderen Triggersignale sind auch in Abbildung 3.13 dargestellt.

In Abbildung 3.17 ist wiederum eine Debug-Ausgabe der Simulation zu sehen. In ihr sind, ebenfalls farblich markiert, alle Eigenschaften der gebildeten Objekte aufgelistet. Zur Benennung der Eigenschaften möge man Anhang A.1 heranziehen. Die Objekte sind entsprechend der Sortierung `U` (absteigend nach `LET`, zweites Sortierkriterium absteigend nach `PET`) angeordnet. Von den 16 verfügbaren Objekten sind nur sechs belegt. Von den 16 verfügbaren Objekten im SpaCal ist keines belegt. Die Lage der Objekte wird wie gewohnt mit θ- und ϕ-Index angegeben (zweite und dritte Spalte in Abbildung 3.17).

Es ist deutlich zu erkennen, dass die beiden wichtigsten Objekte an den ersten

[6]Eine Ausnahme sind die dunkelblauen Cluster. Diese Farbe bedeutet, dass die betreffenden Cluster zu klein sind, um beim *matching* verwendet zu werden.

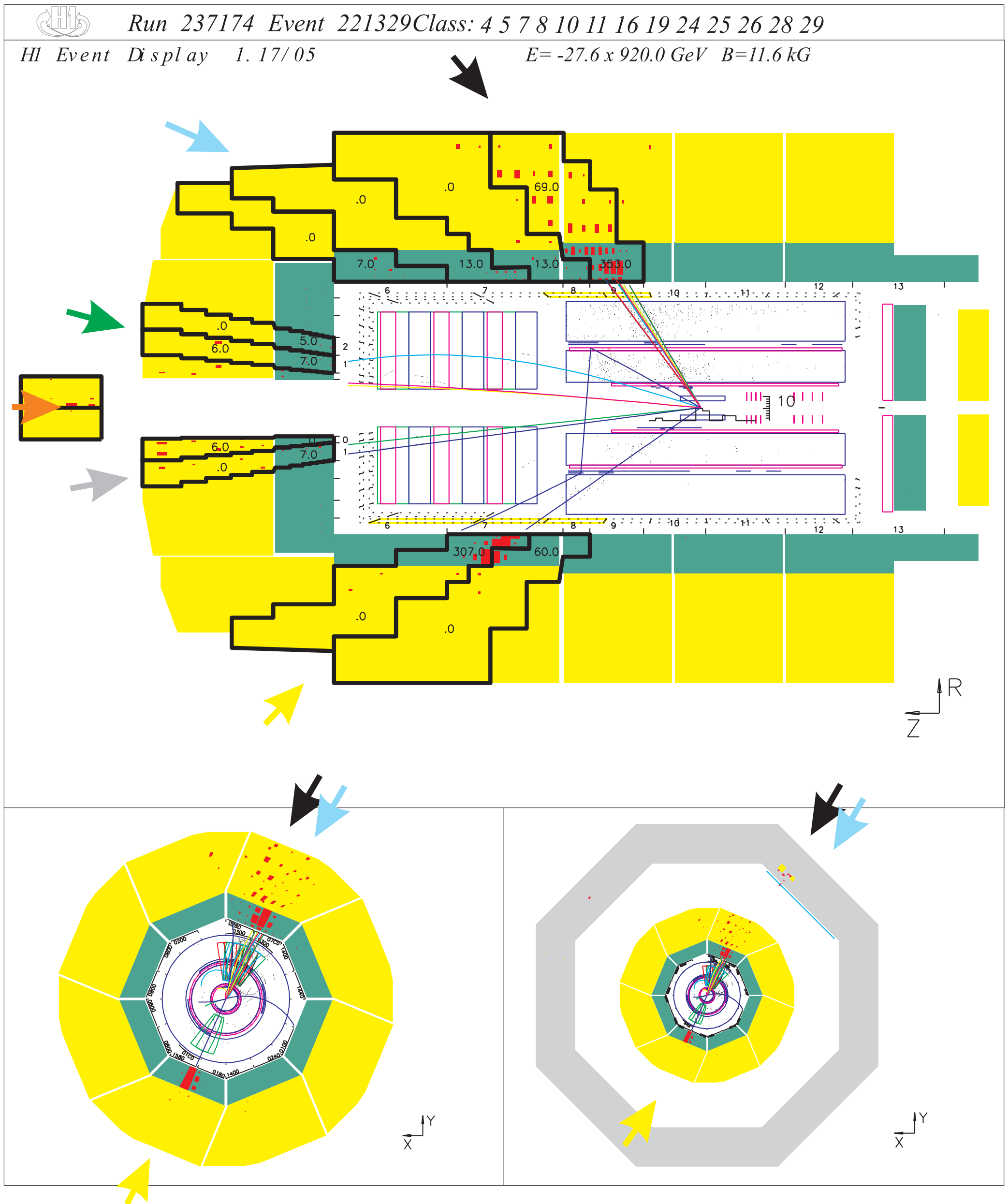


Abbildung 3.13: Graphische Darstellung des verwendeten Beispielereignisses. Sämtliche verfügbare Triggerinformationen sind eingetragen: Aktive *big towers* werden durch schwarze Umrandung und Angabe der deponierten Energie markiert, aktive *big rays* durch eine gelbe Linie. In der Rückansicht unten links sind außerdem noch die aktiven CJC-Masken zu erkennen (Farbe abhängig vom Maskentyp), ebenso wie unten rechts der Treffer in den Myonkammern. Die farbigen Pfeile geben die Lage der gebildeten Objekte an.

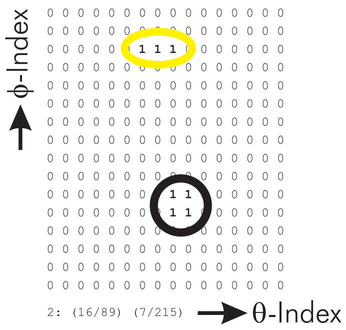


Abbildung 3.14: Bitfeld der *big rays* in der Proportionalkammer im Beispielereignis. Die gefundenen Cluster sind farblich gekennzeichnet.

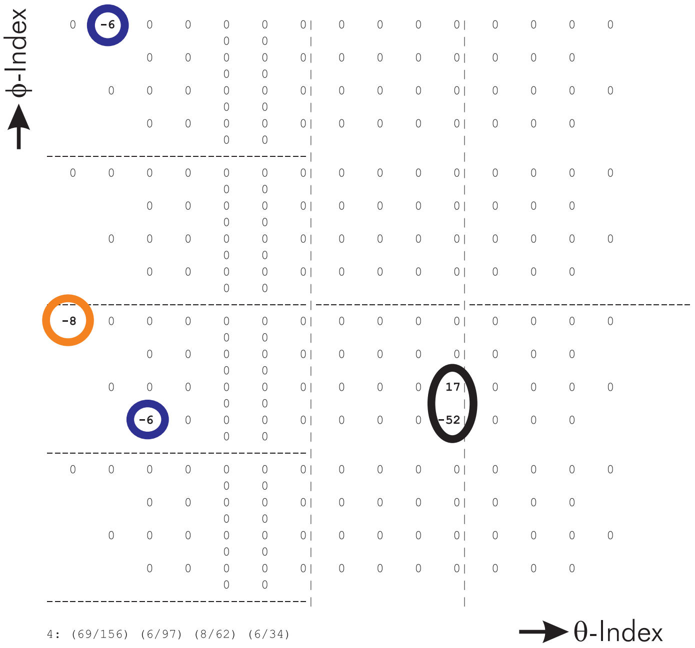


Abbildung 3.15: Bytefeld der Energien der hadronischen *big towers* im Beispielereignis. Dargestellt sind die Energiewerte der einzelnen *big towers*. Die gefundenen Cluster sind farblich markiert, die Clusterzentren mit einem negativen Vorzeichen gekennzeichnet. Die eingezeichneten Linien entsprechen der Aufspaltung in Hardwarebereiche, in denen parallel nach Clustern gesucht wird.

beiden Stellen stehen: Das erste Objekt (schwarz) ist ein Jet mit Signalen im elektromagnetischen und hadronischen Teil des Kalorimeters, Signalen in der Proportionalkammer, in mehreren Maskentypen der CJC und den Myonkammern. Das zweite Objekt stellt das gestreute Elektron dar (gelb), mit einem Signal im elektromagnetischen Teil des Kalorimeters, in der Proportionalkammer, sowie einem Cluster aus zwei aktiven CJC-Masken für negativ geladene Teilchen mit hohem Transversalimpuls. Das Objekt an der dritten Stelle (hellblau) besteht aus elek-

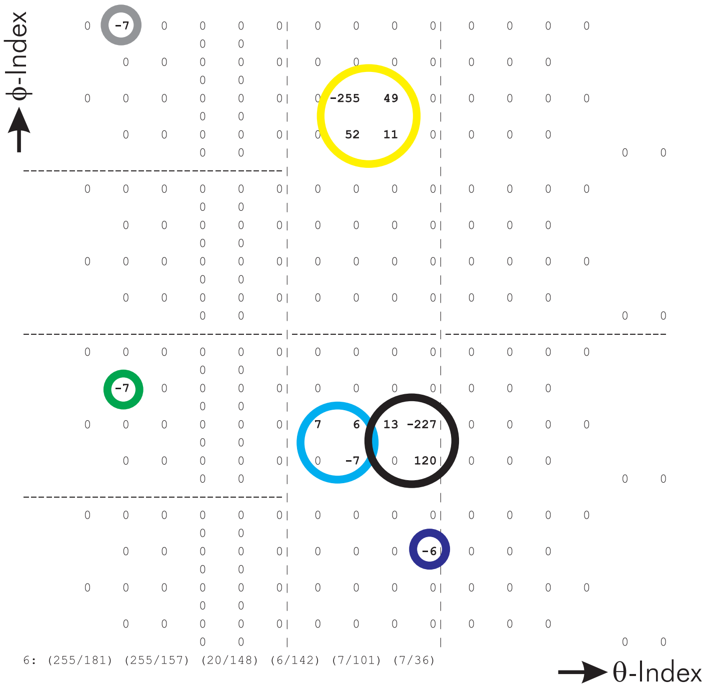


Abbildung 3.16: Energiemappe der elektromagnetischen *big towers* im Beispielereignis. Die Darstellung ist analog zu Abbildung 3.15.

U	>ulet		>upet			LE					LH					L	P		CHP		CLP		CHN		CLN		M	
VEC	THETA	PHI	DTH	DPH	Q	SE	ET	ER	EC	NH	SE	ET	ER	EC	NH	ET	SE	ET	SE	ET	SE	ET	SE	ET	SE	ET	SE	ET
0:	14	80			1	157	255	133	227	3	156	69	17	52	2	255	89	16	13	4	13	3	13	4	0	0	53	1
1:	10	211	4	125	0	181	255	112	255	4	0	0	0	0	0	255	215	7	0	0	0	0	36	2	0	0	0	0
2:	10	80	4	0	1	148	20	13	7	3	0	0	0	0	0	20	0	0	13	4	13	3	13	4	12	3	0	0
3:	0	96	14	16	1	0	0	0	0	0	62	8	0	8	1	8	0	0	0	0	0	0	0	0	0	0	0	0
4:	2	104	12	24	1	36	7	0	7	1	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0
5:	2	247	12	89	1	101	7	0	7	1	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0
6:	255	0	255	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7:	255	0	255	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8:	255	0	255	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9:	255	0	255	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10:	255	0	255	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11:	255	0	255	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12:	255	0	255	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13:	255	0	255	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14:	255	0	255	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15:	255	0	255	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Abbildung 3.17: Liste der aus den Trigger-Signalen des Beispielereignisses gebildeten Objekte.

tromagnetischen *big towers* im Randbereich des ersten Objekts. Die letzten drei Objekte (orange, grün und grau) sind allein aus kleinen Energiedepositionen im vorderen Bereich des Detektors aufgebaut.

Mit diesem Beispiel wird klar, wie durch die Anwendung von *clustering* und *matching* von den Signalen verschiedener Detektorkomponenten auf die beteiligten Teilchen (hier ein Elektron und ein Jet) geschlossen werden kann.

Kapitel 4

Triggern von ϕ-Mesonen in Photoproduktion

Als Anwendung des neuen Konzepts der Datenvorverarbeitung wird in diesem Kapitel unter Verwendung der „intelligenten" Datenvorverarbeitung ein neuronales Netz für den L2NN entwickelt, um ϕ-Mesonen in Photoproduktion zu triggern. Dieses Netz wird dann mit einem Netz verglichen, das ohne diese „intelligenten" Datenvorverarbeitung auskommt. Einführend werden zunächst die theoretischen Grundlagen des betrachteten physikalischen Prozesses und der bisher verwendete Trigger diskutiert.

4.1 Physikalische Grundlagen

4.1.1 Kinematik der Elektron-Proton-Streuung

Als Erstes wird kurz auf die Kinematik der e-p-Streuung eingegangen. Ein Feynman-Diagramm, das alle relevanten Größen enthält, zeigt Abbildung 4.1.

Da bei HERA die Schwerpunktsenergie $\sqrt{s}$ bei der Kollision von Elektronen und Protonen konstant ist, kann eine elastische Streuung durch einen einzigen Parameter beschrieben werden. Dies ergibt sich aus der Energie- und Impulserhaltung. Da die e-p-Streuung im relativistischen Bereich stattfindet, wird sie durch eine lorentzinvariante Variable beschrieben. Hierfür wird meist das negative Quadrat des Viererimpulsübertrages am Elektronvertex Q^2 gewählt:

$$Q^2 \equiv -q^2 = -\left(k' - k\right)^2 . \tag{4.1}$$

Dabei sind k und k' die Vierervektoren des ein- und auslaufenden Leptons.

Die Schwerpunktsenergie $\sqrt{s}$ bestimmt sich zu

$$s = (P + k)^2 \approx 4E_e E_p, \tag{4.2}$$

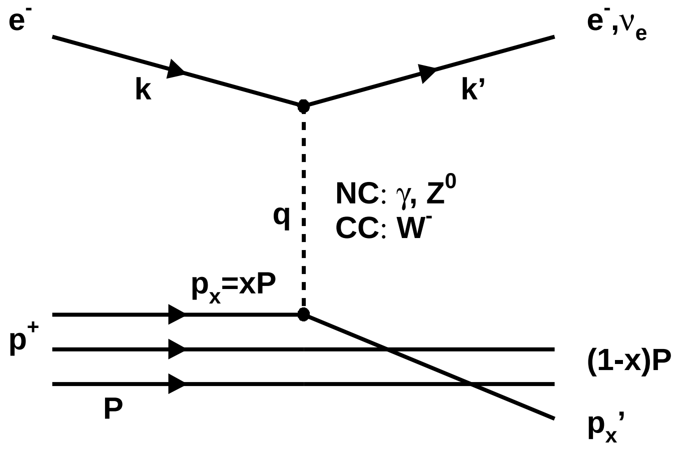


Abbildung 4.1: Feynman-Diagramm einer Elektron-Quark-Streuung für neutrale (*neutral current*, NC) oder geladene Ströme (*charged current*, CC). Angegeben sind die Vierervektoren der beteiligten Teilchen.

wobei als Näherung die Ruhemassen von Elektron und Proton vernachlässigt werden. P ist der Vierervektor des einlaufenden Protons, E_e die Energie des einlaufenden Elektrons, E_p die des einlaufenden Protons.

Für die Beschreibung der inelastischen Streuung benötigt man eine weitere lorentzinvariante Variable, die Inelastizität y, gegeben durch

$$y = \frac{q \cdot P}{k \cdot P} = \frac{2q \cdot P}{s}. \tag{4.3}$$

Im Experiment können die Größen y und Q^2 auch aus dem hadronischen Endzustand bestimmt werden [31], wenn das auslaufende Lepton nicht nachgewiesen werden kann:

$$y = \frac{1}{2E_e} \sum_i \left(E_i - p_i^{||} \right), \tag{4.4}$$

$$Q^2 = \frac{1}{1-y} \sum_i \left(p_i^{\perp} \right). \tag{4.5}$$

Dabei wird über alle Teilchen i des hadronischen Endzustandes (also ohne das auslaufende Lepton) summiert. $p_i^{||}$ und $p_i^{\perp}$ sind die Impulskomponenten des i-ten Teilchens parallel und senkrecht zur Protonstrahlrichtung. Teilchen im Strahlrohr, die nicht nachgewiesen werden, liefern nur einen kleinen Fehler, da ihre Anteile an beiden Summanden klein sind.

Im Quark-Parton-Modell kann ein inelastischer Stoß am Proton als elastischer Stoß an einem Parton dargestellt werden. Dann kann als weitere lorentzinvariante Größe die Bjorkensche Skalenvariable x definiert werden, die angibt, welchen Anteil am Gesamtimpuls des Protons das wechselwirkende Parton hat:

$$x = \frac{Q^2}{2P \cdot k}. \tag{4.6}$$

Die bis jetzt eingeführten lorentzinvarianten Größen hängen zusammen:

$$Q^2 \approx sxy. \tag{4.7}$$

Damit ist klar, dass zwei der drei Variablen Q^2, x und y bei festem s genügen, um den Streuprozess vollständig zu beschreiben. Normalerweise wird mit Q^2 und x gearbeitet.

4.1.2 Photoproduktion und tiefinelastische Streuung

Je nach dem Wert von Q^2 unterscheidet man zwei Ereignisklassen: Für $Q^2 \approx 0$ ist das ausgetauschte Photon quasi reell. Dieser Fall wird als Photoproduktion bezeichnet. Bei Photoproduktion wird das Elektron kaum abgelenkt und ist im Detektor nicht sichtbar, da es im Strahlrohr bleibt. Deshalb ist diese Ereignisklasse, obwohl sie den Wirkungsquerschnitt dominiert, schwer zu triggern. Experimentell gelten bei H1 Ereignisse mit einem Q^2 von bis zu 1 GeV^2 c^{-2} als Photoproduktion, da Elektronen aus Ereignissen mit kleinerem Q^2 im rückwärtigen Kalorimeter nicht nachgewiesen werden können. Reaktionen mit $Q^2 > 1$ GeV werden als tiefinelastische Streuung (*deep inelastic scattering*, DIS) bezeichnet. Bei der Produktion von Vektormesonen ist statt DIS auch der Ausdruck Elektroproduktion gebräuchlich.

4.1.3 Produktion von Vektormesonen

Die Produktion von Vektormesonen kann durch die Kombination von Vektor-Dominanz-Modell (VDM) [32] und Regge-Theorie [33] beschrieben werden. Nach dem VDM fluktuiert ein vom Elektron abgestrahltes virtuelles Photon in ein virtuelles Vektormeson. Das ist möglich, weil das Vektormeson dieselben Quantenzahlen wie das Photon besitzt. Durch starke Wechselwirkung des virtuellen Vektormesons mit dem Proton wird dann ein reelles Vektormeson erzeugt (s. Abbildung 4.2). Die Wechselwirkung zwischen dem Vektormeson und dem Proton kann im Rahmen der Regge-Theorie [33] durch den Austausch von Regge-Trajektorien beschrieben werden. Beim ϕ-Meson erwartet man aufgrund des Quarkinhalts ($s\bar{s}$) nur den Austausch der Pomeron-Trajektorie (Austausch von Vakuumquantenzahlen) für alle Werte der Schwerpunktsenergie, im Gegensatz zu den ρ- und ω-Mesonen, bei denen für niedrige Energien auch ein Reggeon-Anteil (Austausch von nicht-trivialen Quantenzahlen) vorhanden sein wird.

Bei der Wechselwirkung mit dem virtuellen Vektormeson kann das Proton als Ganzes erhalten bleiben (elastischer Prozess, s. Abbildung 4.2 links) oder in einen multihadronischen Endzustand der Masse M_Y übergehen (protondissoziativer Prozess, s. Abbildung 4.2 rechts). Man kann weiter die Photon-Proton-Schwerpunktsenergie $W_{\gamma p}$ definieren:

$$W_{\gamma p}^2 = (P + q)^2 = m_p^2 + ys - Q^2. \tag{4.8}$$

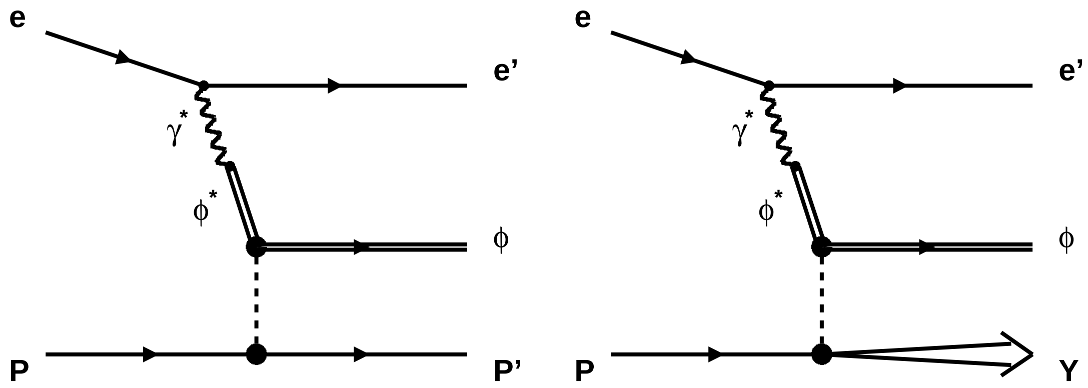


Abbildung 4.2: Feynman-Graph für diffraktive ϕ-Erzeugung, elastischer (links) und proton-dissoziativer Fall (rechts).

Für kleine Impulsüberträge $Q^2 \approx 0$ und zu vernachlässigendes m_p^2 kann dies als

$$W_{\gamma p}^2 \approx ys \tag{4.9}$$

geschrieben werden. Außerdem definiert man t als den Impulsübertrag vom Photon auf das Vektormeson

$$t = (q - p_{VM})^2 = (P - p_Y)^2. \tag{4.10}$$

Dabei ist p_{VM} der Impuls des Vektormesons und p_Y der Impuls des hadronischen Endzustandes.

Für einen inelastischen Stoß, bei dem das Proton nicht intakt bleibt, kann die Inelastizität z definiert werden:

$$z = \frac{P \cdot p_{VM}}{P \cdot q}. \tag{4.11}$$

Die Variablen t und z bei der γ-p-Streuung entsprechen den Variablen Q^2 und y bei der e-p-Streuung.

4.1.4 Perturbative QCD (pQCD)

Um Rechnungen in perturbativer QCD durchführen zu können, wird das virtuelle Vektormeson durch ein Quark-Antiquark-Paar und das Pomeron durch zwei oder mehr Gluonen ersetzt. Es sind mindestens zwei Gluonen nötig, um die Neutralität der Wechselwirkung bezüglich der Farbe sicherzustellen (vgl. Abbildung 4.3).

Für pQCD-Rechnungen existieren unter anderem Modelle von Brodsky et al. [34] und Ryskin [35].

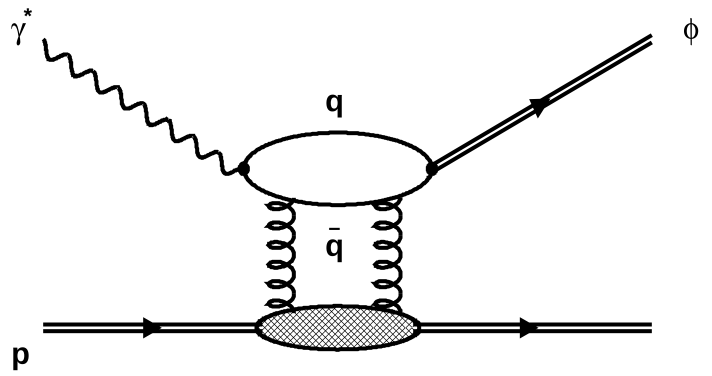


Abbildung 4.3: Ein typischer Beitrag zur Amplitude der Vektormesonenproduktion im Sinne der perturbativen QCD durch Zwei-Gluon-Austausch.

4.1.5 Das ϕ-Meson

Als Anwendung der neuen Konzepte der DDB2 sollen ϕ-Mesonen in Photoproduktion getriggert werden. Das ϕ-Meson hat eine Masse von $1019,413 \pm 0,008$ MeV und eine Gesamtbreite von $4,43$ MeV $\pm$ $0,05$ MeV [36]. Es zerfällt in starker Wechselwirkung, also praktisch sofort. Die wesentlichen Kanäle findet man in Tabelle 4.1.

Zerfall	Verzweigungsverhältnis (%)
$K^+ K^-$	$49,1 \pm 0,6$
$K_L K_S$	$34,1 \pm 0,5$
$\rho\ \pi$	$12,9 \pm 0.7$

Tabelle 4.1: Wesentliche Zerfallskanäle des ϕ-Mesons [36]

Hier wird der Kanal $\phi \to K^+K^-$ genutzt, da bei ihm nur zwei geladene Teilchen, die in Spur- und Proportionalkammern nachweisbar sind, entstehen. Die entstehenden Kaonen sind zwar selbst instabil, besitzen aber eine ausreichend lange Lebensdauer ($1,2386 \pm 0,0024 \cdot 10^{-8}$s [36]), um detektiert werden zu können.

4.2 Trigger auf Level 1

Da das Elektron bei Photoproduktionsereignissen nicht nachweisbar ist, stehen allein die beiden Kaonen zu Triggerzwecken zur Verfügung. Dazu wurde auf L1 der Subtrigger S68 implementiert [37]. Er besteht aus der Verknüpfung verschiedener Triggerelemente, die gleichzeitig erfüllt sein müssen:

- `DCRPh_Ta`

  Der `DCRPhi`-Trigger verwendet dieselben Daten, die später als Driftkammer-Daten (vgl. Abschnitt 3.4.4) an L2 gesendet werden [38]. Das Triggerelement

DCRPh_Ta verlangt das Ansprechen mindestens einer CJC-Maske. Dabei ist der Typ der Maske (für positiv oder negativ geladene Teilchen mit hohem oder niedrigem Transversalimpuls) nicht relevant.

Da die Masken erst für Transversalimpulse $p_t > 400$ MeV c^{-1} effizient werden, schränkt dies den kinematischen Bereich der beobachteten Reaktionen ein. Ohne dieses Triggerelement wäre aber die vom Subtrigger erzeugte Rate zu hoch.

- zVtx_Cls

  Der zVtx-Trigger verwendet dieselben Daten, die auch L2 als z-Vertex-Histogramm (vgl. Abschnitt 1.4) erhält [39]. Das Triggerelement zVtx_Cls fordert, dass sich alle Einträge im z-Vertex-Histogramm in vier benachbarten z-Bins befinden und das Maximum der Einträge nicht in den ersten oder letzten beiden z-Bins liegt. Da ein Eintrag in das Histogramm nur erfolgt, wenn die von einer Spur erzeugten Signale in CIP und COP im selben ϕ-Bin liegen, ergibt sich ein minimaler Transversalimpuls p_t von circa 170 MeV c^{-1}.

- !FwdMu_Fwd_Vtx[1]

  Der FwdMuon-Trigger [40] liefert Informationen über Signalpaare im Vorwärtsmyonspektrometer. Das Triggerelement FwdMu_Fwd_Vtx fordert, dass in der ersten Lage des Vorwärtsmyonspektrometers ein Paar von Signalen auftritt. Falls dieses Triggerelement gesetzt ist, dient es als Veto, da durch die Rapiditätslücke in diesem Bereich keine Signale erwartet werden. Paare von Signalen werden betrachtet, um das Rauschen zu unterdrücken.

- LAr_IF==0

  Der Flüssig-Argon-Trigger bildet Energiesummen über bestimmte Raumwinkelanteile des Detektors (vgl. Abschnitt 3.8), indem über die Energien der *big towers* in den Bereichen IF, FB und CB (vgl. Abschnitt 3.4.1) summiert wird. LAr_IF ist die Summe über den *inner forward* Bereich. Hier wird verlangt, dass sie Null ist, woraus aber wegen der auf L1 implementierten Schwellen nicht notwendig folgt, dass im vorderen Bereich des Flüssig-Argon-Kalorimeters überhaupt kein Signal vorhanden ist.

- Topo_Mult==2

  Dieser Trigger wertet auf L1 dieselben Daten aus, die später auf L2 von der MWPC (vgl. Abschnitt 1.4) verfügbar sind. Topo_Mult ist ein Triggerelement, das angibt, in wievielen der 16 Bins in ϕ-Richtung in der Karte der *big rays* (vgl. Abbildung 3.4) sich Einträge befinden. Hier werden genau zwei Einträge gefordert, entsprechend den zwei Spuren der beiden Kaonen.

[1] !: logisches „nicht“

Da zwei Spuren im gleichen ϕ-Bin nicht genügen, führt dies zu reduzierter Effizienz bei sehr hohem p_t.

- `BGVETO`

  Zusätzlich werden noch einige Triggerelemente gefordert, um Untergrundereignisse auszuschließen. Im Wesentlichen handelt es sich dabei um Forderungen an die Zeitstruktur der Ereignisse mit Hilfe unterschiedlicher *time of flight* Systeme. Solche Veto-Bedingungen werden in jedem Subtrigger benutzt.

Durch diese Triggerelemente wird der kinematische Bereich, in dem die beiden Kaonen und damit das ϕ-Meson erkannt werden, reduziert. Besonders der Zwischenwinkel zwischen den beiden Kaonen wird auf sehr kleine Werte eingeschränkt. Dies ist in Abbildung 4.4 dargestellt.

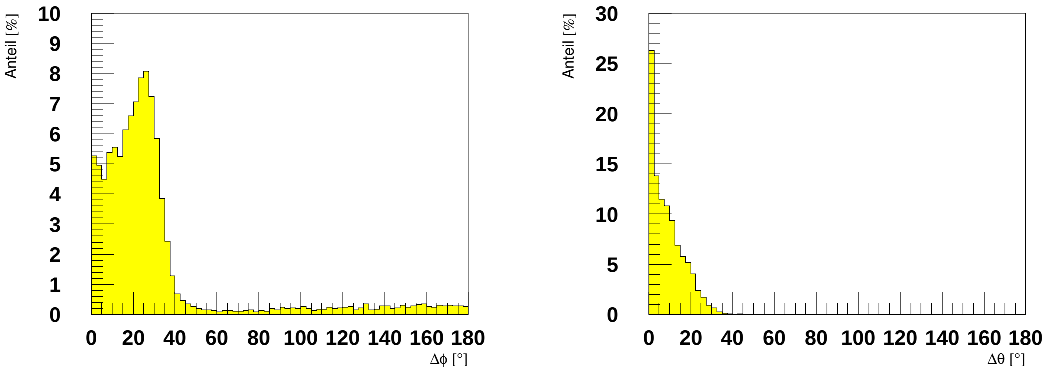


Abbildung 4.4: Zwischenwinkel der beim Zerfall des ϕ-Mesons entstehenden Kaonen. Links bezüglich ϕ, rechts bezüglich θ.

4.3 Netz mit herkömmlicher Datenvorverarbeitung

Das bisher beste bei H1 im L2NN zum Einsatz gekommene neuronale Netz mit herkömmlicher Datenvorverarbeitung (FF54) wurde in [41] vorgestellt. Die Netzbezeichnung wird gebildet aus dem Typ des Netzes (*feed forward*) und einer fortlaufenden Nummer. Die Trainingsdaten setzten sich aus zwei Teilen zusammen: Teil A bestand aus von einem früher verwendeten Netz (FF32) [37] getriggerten Photoproduktionsereignissen des Jahres 1999. Teil B waren durch das Elektron im SpaCal getriggerte Elektroproduktionsereignisse des Jahres 1999. Die Beimischung von Elektroproduktionsereignissen war nötig, da zu wenige Photoproduktionsereignisse, die nicht mit dem Netz FF32 getriggert worden waren, zur Verfügung standen. Ein Netz, das nur mit Teil A trainiert wird und dessen

Trainingsparameter sich nur durch eine erweiterte Auswahl von Eingabegrößen von Netz FF32 unterscheiden, könnte seine Physikeffizienz gegenüber FF32 kaum steigern. Als Untergrund wurden Ereignisse eines *L2L4 transparent runs* verwendet.

Das Netz FF54 verwendet acht versteckte Knoten und folgende Eingabegrößen (zur Definition der Eingangsgrößen s. [42] und die Anmerkungen zu Abbildung 4.5):

- vom z-Vertex-Histogramm die Position des Maximums `CPVPOS`, die Summe über alle Einträge `CPVSUM` und den Wert des Maximums `CPVMAX`
- von den Zentralen Spurkammern (CJC) die Zahl der Masken für negativ und positiv geladene Teilchen mit hohem und niedrigem Transversalimpuls `TRHIPOS`, `TRHINEG`, `TRLOPOS` und `TRLONEG`
- von Flüssig-Argon-Kalorimeter die Energiedeposition in den Bereichen IF, FB und CB, hauptsächlich als Veto gegen Untergrund: `LARIFE`, `LARFBE`, `LARCBE`
- von den Zentralen Myonkammern, die Trefferzahlen in den Bereichen BE, FB und BB, ebenfalls hauptsächlich als Veto gegen Untergrund: `IRONBE`, `IRONFB`, `IRONBB`

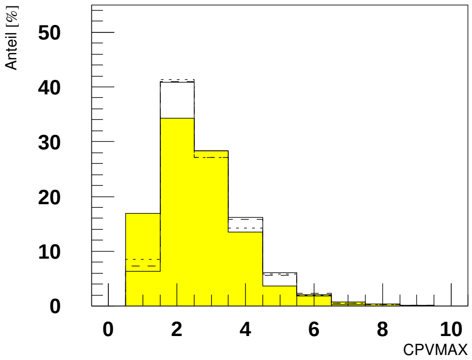
Für eine bessere Effizienz des Netzes FF54 gegenüber dem Netz FF32 wurde in Kauf genommen, dass das Netz von vielen verschiedenen Detektorkomponenten abhängt. Vor allem im Vorwärtsbereich wird dadurch die Beschreibung durch Monte-Carlo (MC) und die Trennung von elastischen und proton-dissoziativen Ereignissen erschwert.

Die Verteilungen der Eingabegrößen für Photoproduktionsdaten, für elastische und proton-dissoziative Monte-Carlo-Ereignisse sowie den Untergrund, der aus einem *L2L4-transparent run* stammt, sind in Abbildung 4.5 dargestellt. Es ist eine gute Übereinstimmung zwischen den Monte-Carlo-Daten und den realen Ereignissen vorhanden.

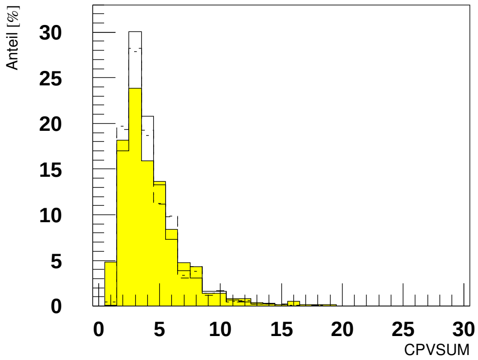
Die Ausgabeverteilungen des Netzes FF54 für die Untergrunddaten und die beiden Monte-Carlo-Samples findet man in Abbildung 4.6, ebenso die Effizienzkurven. Das Netz wird mit einer Schwelle (*cut*) von 0,5 betrieben. Dies entspricht einer Physikeffizienz von 78 % bis 81 % (für elastische bzw. proton-dissoziative Ereignisse) bei einer Untergrundreduktion von 73 %.

4.4 Netz mit intelligenter Datenvorverarbeitung

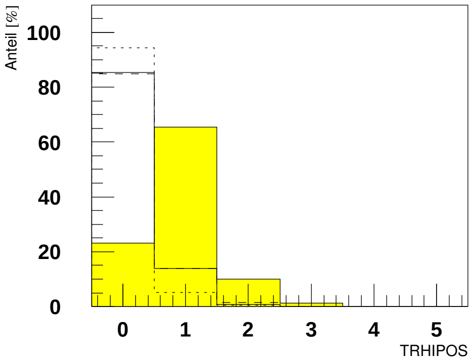
Unter Ausnutzung der im vorigen Kapitel vorgestellten Konzepte der DDB2 wurde ein neues Netz (FF2000) trainiert. Um dabei die Vergleichbarkeit von altem



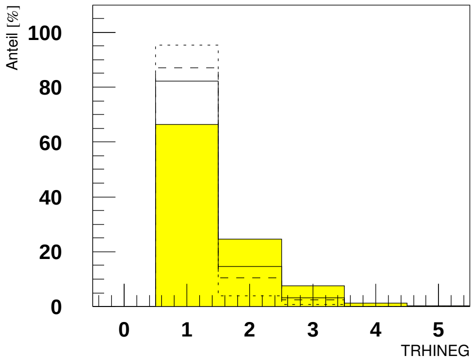
`CPVMAX`: Wert des Maximums im z-Vertex-Histogramm.



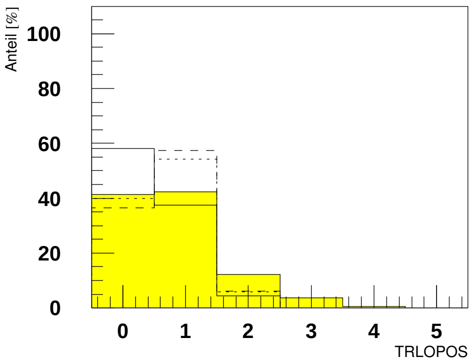
`CPVSUM`: Summe über alle Einträge im z-Vertex-Histogramm.



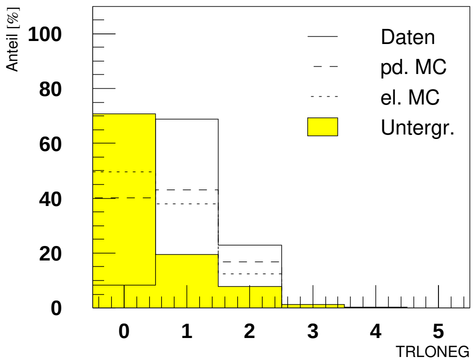
`TRHIPOS`: Zahl von aktiven CJC-Masken für positive Teilchen mit hohem Transversalimpuls.



`TRHINEG`: Zahl von aktiven CJC-Masken für negative Teilchen mit hohem Transversalimpuls. Eine Maske ist fehlerhaft und immer aktiv.

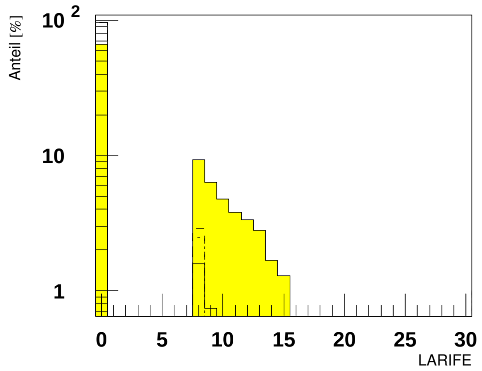


`TRLOPOS`: Zahl von aktiven CJC-Masken für positive Teilchen mit niedrigem Transversalimpuls.

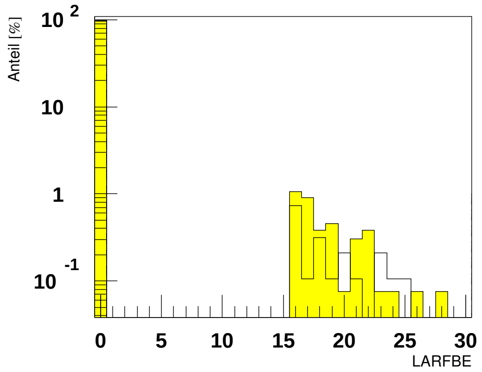


`TRLONEG`: Zahl von aktiven Masken für negative Teilchen mit niedrigem Transversalimpuls.

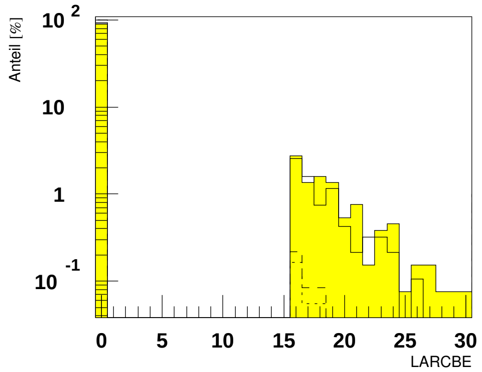
Abbildung 4.5: Verteilungen der Eingabegrößen von FF54. `CPVPOS` wird in FF2000 ebenfalls verwendet. Die Verteilung ist deshalb in Abbildung 4.7 aufgeführt.



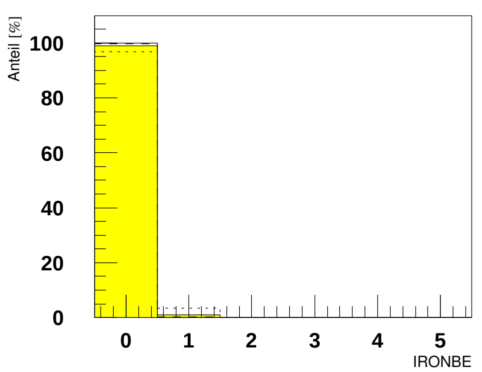
`LARIFE`: Energie im IF (in *counts*). Es ist die Energieschwelle auf L1 zu erkennen. Die Energie pro *count* variiert je nach Detektorbereich (ca. 10 *counts* = 1 GeV).



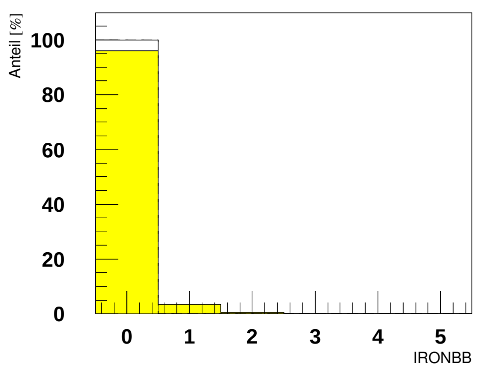
`LARFBE`: Energie im FB (in *counts*).



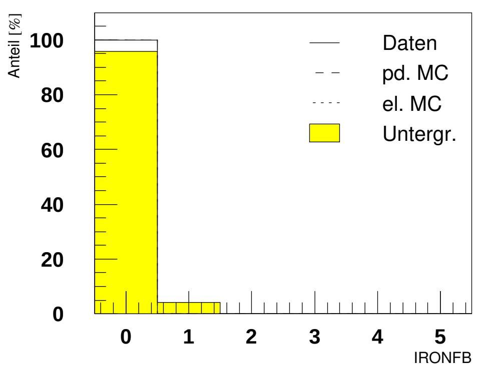
`LARCBE`: Energie im CB (in *counts*).



`IRONBE`: Zahl von Hits im *backward endcap* der Zentralen Myonkammern.

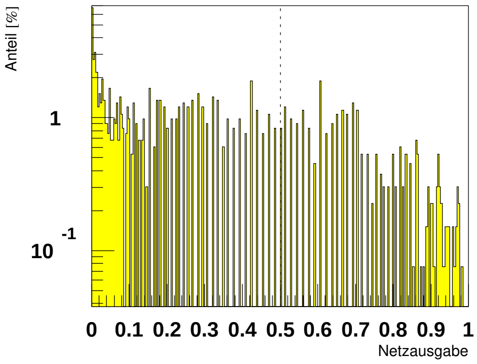


`IRONBB`: Zahl von Hits im *backward barrel* der Zentralen Myonkammern.

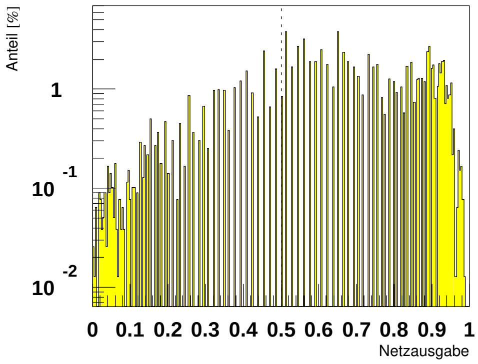


`IRONFB`: Zahl von Hits im *forward barrel* der Zentralen Myonkammern.

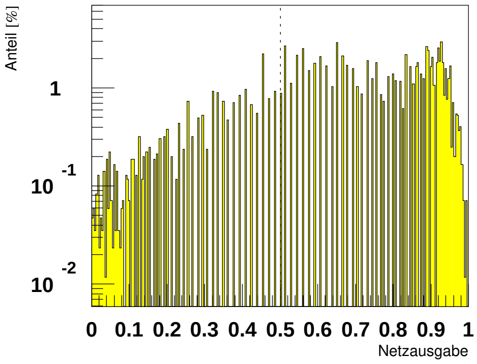
Abbildung 4.5 (Fortsetzung)



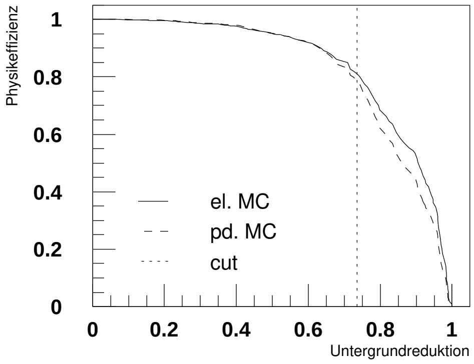
Ausgabeverteilung Untergrund.



Ausgabeverteilung elastisches MC.



Ausgabeverteilung proton-dissoziatives MC.



Effizienzkurven, bestimmt aus Monte-Carlo-Daten und Untergrund.

Abbildung 4.6: Eigenschaften von FF54

und neuem Konzept sicherzustellen, wird auf die gleichen Trainingsdaten wie bei FF54 zurückgegriffen. Allerdings wird darauf verzichtet, Teil B des Trainingssamples (Elektroproduktionsdaten) zu verwenden, weil sich die Signatur von Elektroproduktionsereignissen im Konzept der DDB2, das immer den ganzen Detektor betrachtet, viel stärker von der von Photoproduktionsereignissen unterscheidet, als dies bei der herkömmlichen DDB der Fall ist. Bei der herkömmlichen DDB kann der Unterschied ignoriert werden, indem die Eingabegrößen, die vom SpaCal stammen (und das gestreute Elektron enthalten), nicht verwendet werden. Bei der DDB2 ist etwas derartiges im Prinzip auch möglich. Die Implementierung ist allerdings komplexer: Durch LUTs müsste die vom SpaCal stammende Information explizit auf Null gesetzt werden. Auf die Signatur von Elektroproduktionsereignissen bei intelligenter Datenvorverarbeitung wird nochmals am Ende dieses Abschnitts eingegangen.

Das Training gestaltet sich mit der intelligenten Datenvorverarbeitung (DDB2) komplexer als bei der herkömmlichen (DDB): Bei der DDB stehen „fertige“ Eingabegrößen zur Verfügung, und zwar für alle physikalischen Probleme die selben. Bei der DDB2 hingegen existieren zum einen wesentlich mehr Eingabegrößen, zum anderen können diese Eingabegrößen durch die Adjustierung der in die Algorithmen eingehenden Parameter verändert werden (s. Anhang A.2), Oft ist es sogar so, dass Eingabegrößen erst durch die Festlegung bestimmter Parameter (beispielsweise der Sortierkriterien) sinnvoll bestimmt werden. Deshalb genügt es nicht, nur aus einer Liste von Eingabegrößen auszuwählen. Im Gegenteil: Es ist möglich (aber auch erforderlich), durch Anpassung von LUTs, Sortierkriterien und weiteren Parametern speziell an das jeweilige physikalische Problem angepasste Sätze von Eingabegrößen zu erstellen.

Bei dem Netz für ϕ-Mesonen in Photoproduktion erwiesen sich folgende Einstellungen als nützlich (Zur genauen Beschreibung möglicher Parametersätze s. Anhang A.2):

- Die einzelnen Algorithmen (*clustering*, *matching*, etc.) werden mit den Parametern in Tabelle 4.2 betrieben.

big towers beim *clustering* benutzen	ja
trigger towers beim *clustering* benutzen	ja
unvollständige[2] Objekte beim *clustering* zurückweisen	ja
minimale Clusterenergie eines em./had. *big/trigger towers*	6
maximale Zahl von *big tower* Clustern	10
maximale Zahl von *trigger tower* Clustern	11
maximale Zahl von *big ray* Clustern	10
maximale Zahl von CJC-Clustern	22

Tabelle 4.2: Parameter der DDB2 für FF2000 (vgl. auch Anhang A.2).

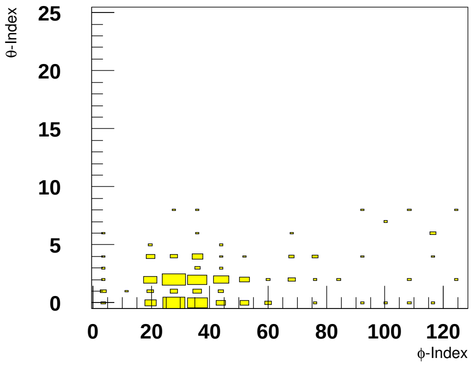
- Im Wesentlichen werden 1:1 LUTs verwendet, nur einige rauschende *trigger towers* und ein hängendes Bit in der CJC werden ausgeblendet[3].

- Es werden nur Eingabegrößen verwendet, die aus einer einzigen Sortierung stammen: Das primäre Sortierkriterium dieser Sortierung `W` ist `PET`, die Pseudoenergie des Clusters in der Proportionalkammer, das sekundäre `LET`, die Gesamtenergie der Cluster im Flüssig-Argon-Kalorimeter. Energiereiche Objekte stehen beide Male vorne.

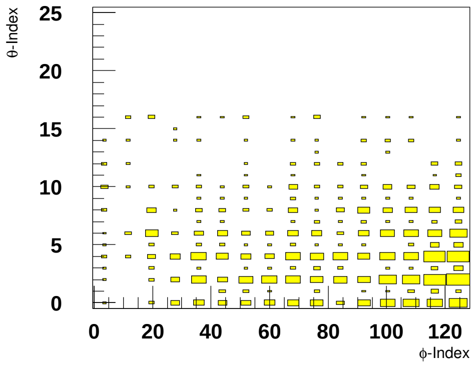
- Außerdem wird der Zähler `COUNT4` eingeführt. Er gibt an, bei wie vielen Objekten die Bedingung `PET` > 0 erfüllt ist.

[2]Unvollständig bedeutet: Es ist keine vollständige Positionsinformation für den Cluster verfügbar, sondern entweder nur der ϕ-Index oder nur der θ-Index.

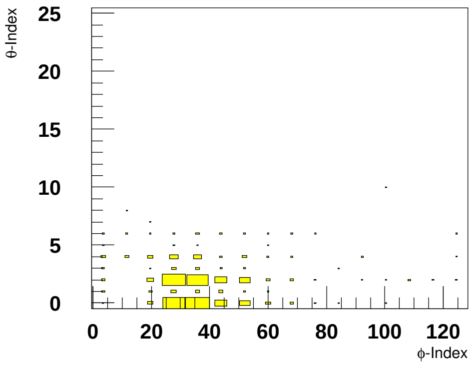
[3]Diese Ausblendung bezieht sich nur auf die von der DDB2 weiterbearbeiteten Daten. Die Eingabegrößen der herkömmlichen DDB bleiben unverändert. Deshalb ist das hängende Bit auch in Abbildung 4.5 in der Größe `TRHINEG` noch sichtbar.



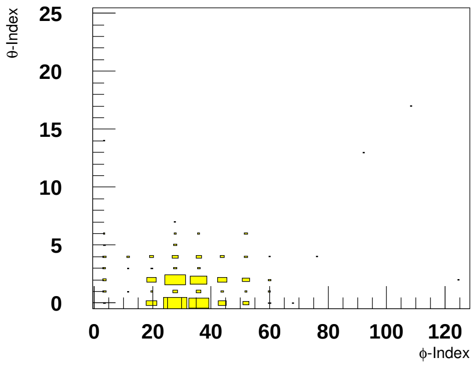
`WDTHETO` vs. `WDPHIO`: Winkeldifferenzen bezüglich θ- und ϕ-Index zwischen den beiden ersten Objekten nach Sortierung `W`, dargestellt für `COUNT4` $= 2$. (Photoproduktion)



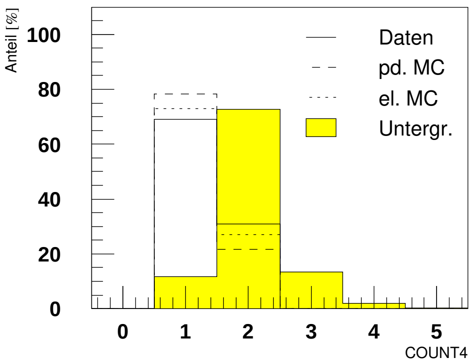
`WDTHETO` vs. `WDPHIO`: s.o (Untergrund)



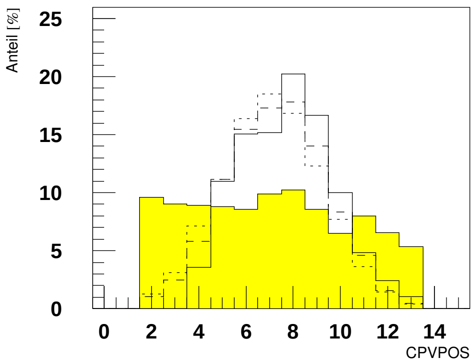
`WDTHETO` vs. `WDPHIO`: s.o. (MC elastisch).



`WDTHETO` vs. `WDPHIO`: s.o. (MC proton-diss.).

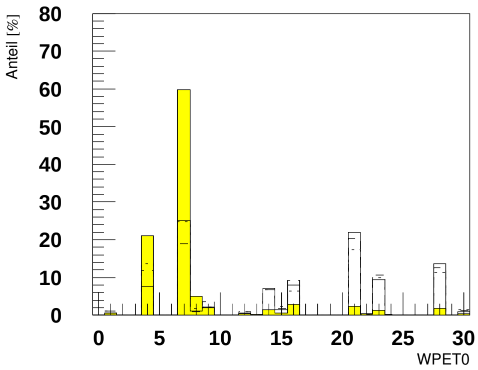


`COUNT4`: Zahl der Objekte mit `PET` > 0.

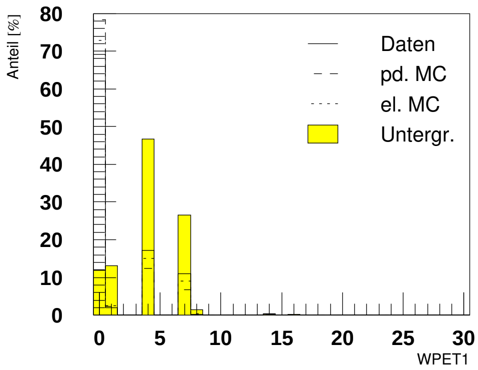


`CPVPOS`: Position des Maximums im z-Vertex-Histogramm.

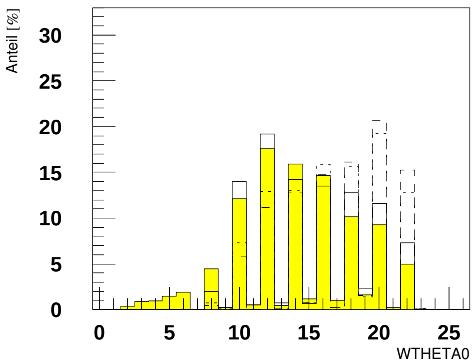
Abbildung 4.7: Verteilungen der Eingabegrößen von FF2000.



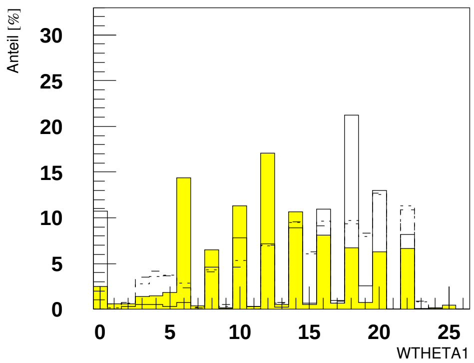
`WPET0`: Pseudoenergie des ersten Objekts nach Sortierung `W` in einem MWPC-Cluster. Deutlich ist zu erkennen, dass auf Grund des Algorithmus nicht alle Werte möglich sind. Der Untergrund dominiert bei niedrigen Werten von `WPET0`.



`WPET1`: Pseudoenergie des zweiten Objekts nach Sortierung `W` in einem MWPC-Cluster. Hier dominiert der Untergrund bei hohen Werten von `WPET1`.



`WTHETA0`: θ-Index des ersten Objekts der Sortierung `W`. Das „Zackenmuster" für θ-Indizes zwischen 6 und 22 entsteht dadurch, dass meist kein Signal in den *trigger towers* vorhanden ist, die Darstellung aber in deren (im Vergleich zur MWPC) feinerer Granularität erfolgt.

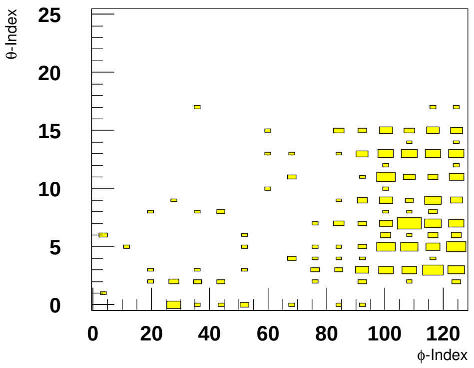


`WTHETA1`: θ-Index des zweiten Objekts der Sortierung `W`. Wie auch bei `WTHETA0` tendieren Physikereignisse zu einem höheren θ-Index als Untergrundereignisse.

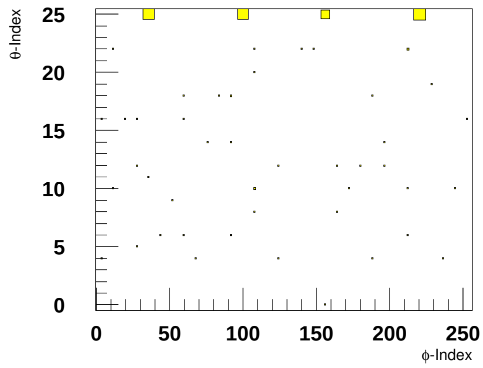
Abbildung 4.7 (Fortsetzung)

Mit der so konfigurierten DDB2 wurde ein Netz mit fünf versteckten Knoten und folgenden Eingabegrößen trainiert (zur Nomenklatur der Eingabegrößen s. Anhang A.1 und [42]):

- Zahl von Objekten mit Proportionalkammerhits: `COUNT4`
- Winkeldifferenzen in θ- und ϕ-Index zwischen den ersten beiden Objekten



`WDTHETO` vs. `WDPHIO`: Winkeldifferenzen bezüglich θ- und ϕ-Index zwischen den beiden ersten Objekten nach Sortierung `W`. Durch die grobe Aufteilung des θ-Bins 25 (inneres SpaCal) in ϕ-Richtung wird die eigentliche *back-to-back*-Anordnung von ϕ-Meson und gestreutem Elektron verschmiert.



`WTHETA1` vs. `WPHI1`: Position bezüglich θ- und ϕ-Index des zweiten Objekts nach Sortierung `W`. Deutlich ist das gestreute Elektron bei einem θ-Index von 25 in den vier Quadranten im inneren Teil des SpaCals zu erkennen.

Abbildung 4.8: Signatur von Elektroproduktionsereignissen.

bezüglich der Sortierung `W`: `WDTHETO`, `WDPHIO`

- jeweils zwei Eigenschaften (Pseudoenergie des Clusters in der Proportionalkammer und θ-Index) der beiden ersten Objekte bezüglich der Sortierung `W`: `WPETO`, `WPET1` und `WTHETAO`, `WTHETA1`

- Zusätzlich wird eine bereits in FF54 verwendete Eingabegröße verwendet, die Position des Maximums im z-Vertex-Histogramm: `CPVPOS`

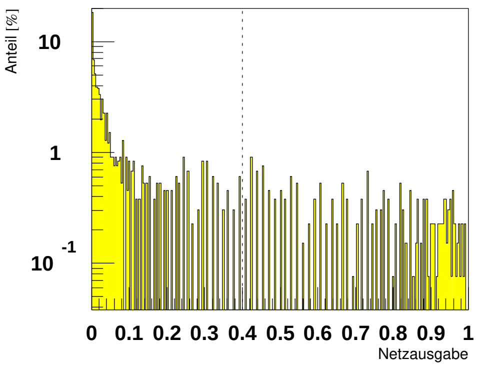
Die Verteilungen für Untergrunddaten, Photoproduktionsdaten und die beiden Monte-Carlo-Samples sind in Abbildung 4.7 dargestellt.

Bei den Physikereignissen (Daten und Monte-Carlo) ist der Anteil von Ereignissen mit einem einzigen Cluster in der Proportionalkammer sehr viel höher als beim Untergrund, da die Spuren der beiden Kaonen oft zusammen in einen Cluster fallen, während beim Untergrund eher Ereignisse dominieren, deren Cluster einen Winkel von 180° einschließen (*back-to-back*-Bedingung, s. Abbildung 4.7, Seite 67, links unten). Dies zeigt sich besonders, wenn man die Winkeldifferenzen zwischen den beiden ersten Objekten nach Sortierung `W` (wie oben definiert mit dem primären Sortierkriterium `PET` und dem sekundären `LET`) betrachtet, unter der Voraussetzung, dass es zwei Objekte mit einem Hit in der MWPC gibt. Dann ist klar erkennbar, dass bei einem Physikereignis die Spuren nahe beisammen liegen (s. Abbildung 4.7, Seite 67, oben links, Mitte links und Mitte rechts für Daten und MC), während sie beim Untergrund (s. Abbildung 4.7, Seite 67, oben rechts) einer *back-to-back*-Bedingung genügen. Dass im Bereich der Winkeldifferenz bis etwa 25° für $\Delta\phi$, bei kleinen Werten für $\Delta\theta$, weder bei den Physikereignissen noch beim Untergrund Einträge vorhanden sind, liegt daran, dass zwei Objekte in diesem Bereich durch *clustering* und anschließendes *matching* nicht mehr auflösbar sind und zu einem verschmolzen werden (s. Abbildung 4.7, Seite 67, oben und Mitte).

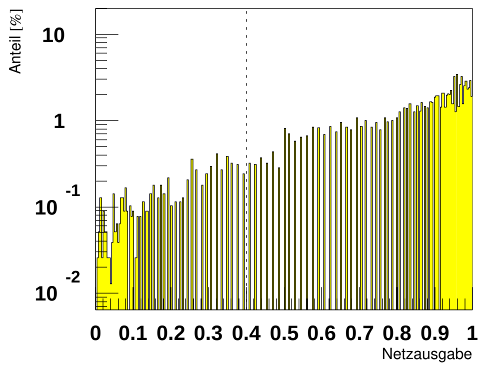
Bei Photoproduktion werden also die ersten beiden Positionen in der Sortierung `W` von den beiden Kaonen belegt, falls die Signale getrennt werden konnten. Falls nicht, enthält die erste Position die Signale, die beide Kaonen zusammen erzeugt haben. Die zweite Position ist entweder leer oder wird durch zufällig erzeugte Signale besetzt.

Wenn man hingegen Elektroproduktion betrachtet, ist folgendes zu bemerken: Durch den höheren Transversalimpuls p_t des ϕ-Mesons, hervorgerufen durch die Transversalenergie des gestreuten Elektrons, liegen hier die Spuren der Kaonen näher zusammen und fallen öfter in einen einzigen Cluster als bei Photoproduktionsereignissen. Wiederum enthält in einem solchen Fall das erste Objekt nach Sortierung `W` die Signale, die von den beiden Kaonen generiert wurden. Die zweite Position wird in diesem Fall aber eindeutig von einem Objekt belegt, das aus dem Signal des Elektrons in einem *big tower* des SpaCals besteht: Ein solches Objekt besitzt aber (aus Gründen des vom SpaCal abgedeckten Raumwinkels) kein Signal in der Proportionalkammer, auf das das primäre Sortierkriterium der Sortierung `W` (absteigend nach `PET`) sensitiv wäre. Deshalb wird dieses Objekt sicher nach dem Objekt, das die beiden Kaonen enthält, angeordnet. Das sekundäre Sortierkriterium der Sortierung `W` (absteigend nach `LET`, der gesamten in *big towers* deponierten Energie) bewirkt, dass das Objekt mit dem Elektron im SpaCal unmittelbar nach allen Objekten, die ein Signal in der Proportionalkammer besitzen, angeordnet wird. Dies ist hier die zweite Position. Wenn man in diesem Fall die Winkeldifferenzen zwischen den beiden ersten Objekten bezüglich der Sortierung `W` aufträgt, bemerkt man einen *back-to-back*-Zusammenhang: Die beiden Kaonen liegen dem gestreuten Elektron gegenüber (vgl. Abbildung 4.8).

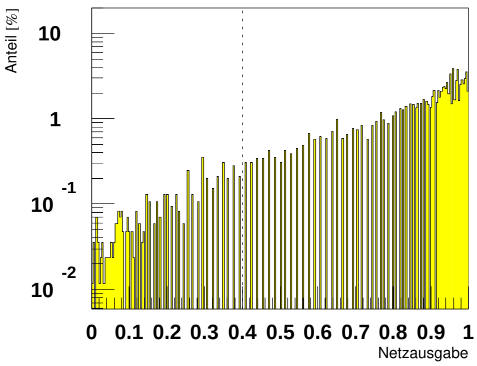
Ausgabeverteilungen und Effizienzkurven des Netzes FF2000 findet man in Abbildung 4.9. Auch die Effizienz in Abhängigkeit von $|t|$ und $W_{\gamma p}$ ist dort dargestellt. Die Schwelle (*cut*) des Netzes wurde so gewählt, dass die Untergrundreduktion bei 80 % liegt. Trotz dieses im Vergleich zu FF54 höheren Wertes für die Untergrundreduktion verbessert sich die Physikeffizienz auf 93 % bis 95 % (für elastische und proton-dissoziative Ereignisse).



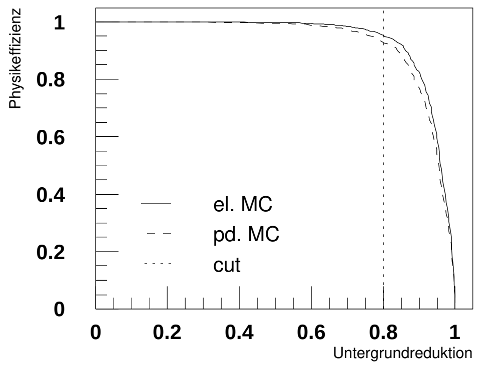
Ausgabeverteilung Untergrund.



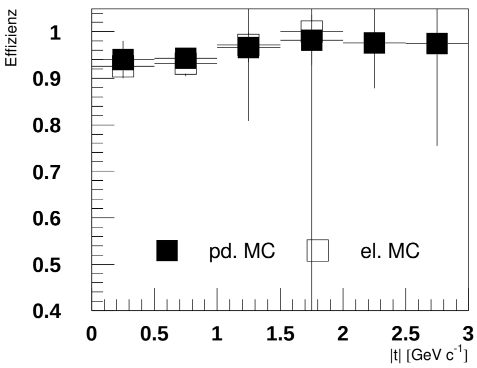
Ausgabeverteilung elastisches MC.



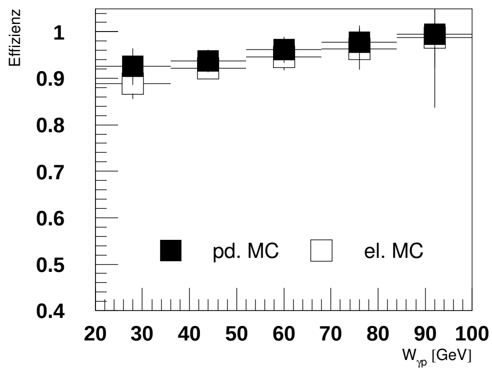
Ausgabeverteilung proton-dissoziatives MC.



Effizienzkurven, bestimmt aus Untergrund und Monte-Carlo-Daten.



Effizienz in Abhängigkeit von $|t|$.



Effizienz in Abhängigkeit von $W_{\gamma p}$.

Abbildung 4.9: Eigenschaften des FF2000

Kapitel 5

Anwendungen

5.1 Emulation der DDB2 durch die DDB

Da die DDB2 bis jetzt nur als Simulation zur Verfügung steht, sind physikalische Ergebnisse, die mit dem neuronalen Trigger unter Zuhilfenahme der DDB2 gewonnen werden, noch nicht möglich. Es besteht allerdings die Möglichkeit, wichtige Eingabegrößen des im letzten Kapitel vorgestellten Netzes mit Hilfe des existierenden neuronalen Netzwerktriggers zu emulieren.

Die eigentlich zur Berechnung eines neuronalen Netzwerkes vorgesehene Hardware (CNAPS) wird benutzt, um die DDB zu erweitern und mit Operationen, die in der DDB allein nicht möglich sind, eine Größe X zu berechnen. Dazu wird im CNAPS derselbe *feed forward* Algorithmus benutzt, der auch sonst zur Berechnung neuronaler Netze verwendet wird. Allerdings werden die Gewichte des verwendeten Netzes nicht durch Trainieren, sondern durch explizites Konstruieren bestimmt.

Die CNAPS-Hardware steht dann natürlich nicht mehr zur Berechnung eines neuronalen Netzes zur Verfügung, das dieses X als Eingabegröße verwendet, d.h., eine Kombination von X mit anderen Eingabegrößen ist nicht möglich: X wird wie der Ausgabewert eines neuronalen Netzes behandelt und die Triggerentscheidung durch Vergleich von X mit einer Schwelle (*cut*) getroffen. Aber auch ohne die Möglichkeit, ein neuronales Netz mit X zu trainieren, leistet das so konstruierte Netz mehr als das bisher [41] verfügbare.

Die aus der DDB2 stammenden Eingabegrößen, die teilweise emuliert werden, sind `COUNT4` und `WDPHI0`. `COUNT4` bezeichnet die Zahl von Objekten, die einen Cluster in der MWPC besitzen, `WDPHI0` die Differenz der ϕ-Indizes zwischen den beiden größten dieser Cluster. Die Emulation dieser Größen ist nur näherungsweise möglich. Die Unterscheidung zwischen verschiedenen Objekten mit unterschiedlichem θ-Index wird fallengelassen, ebenso die Verbindung von Signalen in nebeneinanderliegenden ϕ-Bins der *big ray*-Mappe zu Clustern.

Es wird folgender Algorithmus benutzt: Damit ein Ereignis als Physikereignis akzeptiert wird, darf insgesamt nur in maximal drei aufeinanderfolgenden ϕ-Bins der *big ray*-Mappe ein Signal vorhanden sein. Damit wird Untergrund, dessen Signale oft die *back-to-back*-Bedingung erfüllen, unterdrückt, während Physikereignisse mit ihren nahe beieinander liegenden Spuren akzepiert werden. Die Bedingung kann auch in einer anderen Form formuliert werden, in der sie dann in ein neuronales Netz umgesetzt wird: Zu keinem ϕ-Bin mit einem gesetzten Bit in der *big ray*-Mappe liegt ein weiteres gesetztes Bit außerhalb einer Maske von zwei ϕ-Bins links oder rechts.

Die Realisierung erfolgt in zwei Schritten: Als Erstes werden neue Eingabegrößen (`XBRPHI0`, …, `XBRPHI31`) eingeführt. `XBRPHI` steht für „*e***x***tended* **B***ig* **R***ay* gebinnt bezüglich $\boldsymbol{\phi}$“. Diese 32 Eingabegrößen liefern einen Wert von 1, wenn in der vorderen (θ-Index 1 bis 11) bzw. hinteren (θ-Index 12 bis 25) Hälfte der *big ray*-Mappe (vgl. Abbildung 3.4) in einem der 16 ϕ-Bins ein oder mehrere Kanäle aktiv sind, sonst 0. Die Aufspaltung in den vorderen und hinteren Bereich ist technisch durch die Anordnung der Daten auf dem L2-Bus bedingt.

Danach werden die Gewichte des „neuronalen Netzes“ konstruiert. Das Netz benötigt 32 versteckte Knoten. Jeder dieser Knoten prüft für eine der `XBRPHI`-Eingabegrößen, ob die oben formulierte Bedingung erfüllt ist. Die Funktionsweise soll kurz dargestellt werden:

Die Struktur des Netzes ist in Abbildung 5.1 skizziert. In einem versteckten Knoten i wird mit einer vierwertigen Logik gearbeitet. Falls Eingabegröße i nicht verschwindet, liefert sie den Wert „viel“, sonst „nichts“. Falls eine der Eingabegrößen, die nicht als zu Eingabegröße i benachbart gilt (siehe auch Abbildung 5.1, oben rechts), nicht verschwindet, liefert sie einen Wert „wenig“, sonst ebenfalls „nichts“. Alle anderen Eingabegrößen liefern immer „nichts“. Mit Hilfe der Sigmoid-Funktion wird folgende Algebra realisiert: Um „genug“ zu erhalten, muss mindestens „wenig“ zu „viel“ addiert werden, während „wenig“ im Prinzip beliebig oft aufsummiert werden kann, ohne dass sich „genug“ ergibt.

Falls ein Knoten nach der Aufsummierung den Wert „genug“ hat, sendet er ein Veto an den Ausgabeknoten. Falls der Ausgabeknoten auch nur ein Veto erhält, wird das Ereignis abgelehnt. Ein weiterer Knoten, der 33., wird dazu verwendet, um vollständig „leere“ Ereignisse zurückzuweisen. Dieser Knoten ist in Abbildung 5.1 nicht eingezeichnet.

Der Gewichtsvektor wird automatisch durch ein einfaches C-Programm generiert. Im Wesentlichen wird dabei folgender Algorithmus verwendet (s. auch Abbildung 5.1): Die Gewichte der versteckten Schicht sind 511 zwischen dem Eingangsknoten i und dem versteckten Knoten i. Zwischen dem versteckten Knoten i und allen Eingangsknoten, in denen ein Signal gleichzeitig mit einem Signal im Eingangsknoten i erlaubt ist, sind die Gewichte 0. Die Gewichte zwischen dem verstecken Knoten i und allen anderen Eingabeknoten sind 1. Schwellenwerte werden in der versteckten Schicht nicht angewendet. Die Gewichte zwischen dem Ausgabeknoten und den Knoten der versteckten Schicht sind -1024. Im Ausgabeknoten wird eine Schwelle angewendet, die sich wie in Abbildung 5.1 bestimmt.

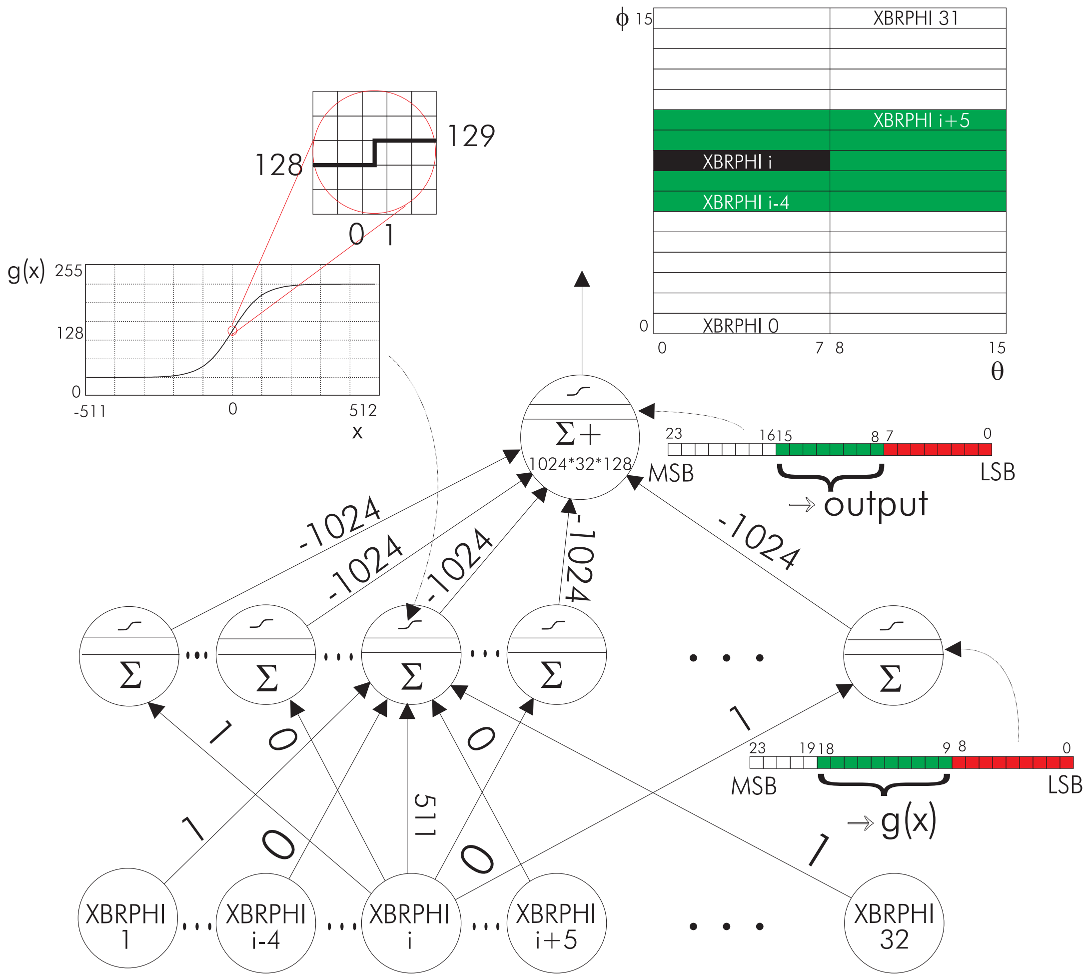
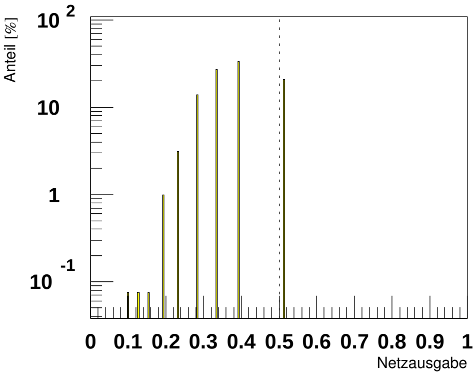
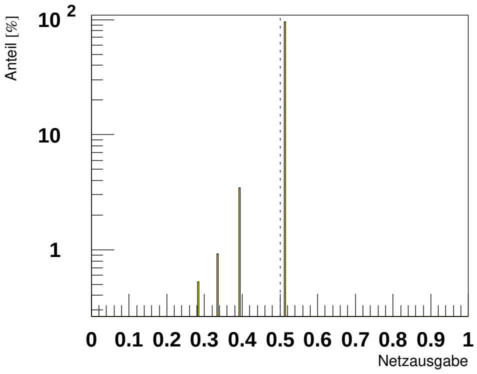


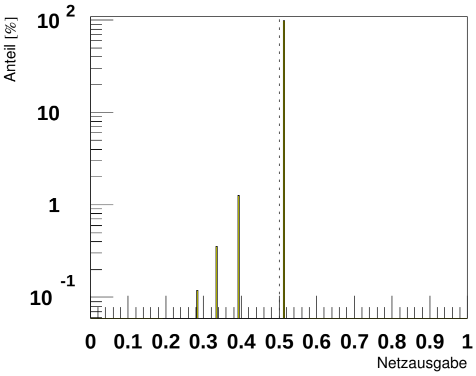
Abbildung 5.1: Struktur des konstruierten Netzes FF59. Es wird deutlich, dass das idealisierte Modell eines neuronalen Netzes unter der Verwendung von Fließkommazahlen wie es in Abschnitt 2.2.1 dargestellt wurde, durch technische Aspekte erweitert werden muss, wenn der Algorithmus auf einer mit Festkommaarithmetik arbeitenden Plattform umgesetzt werden soll. Für die Gewichte stehen 16 Bits zur Verfügung, für die Eingabegrößen 8 Bits. Durch die Multiplikation in den Knoten der versteckten Schicht entstehen 24-Bit-Größen, von denen nur bestimmte Bits (unter Beachtung des Vorzeichens) durch Anwendung der Sigmoid-Funktion $g(x)$ weiterverarbeitet werden. Dies wurde in der Darstellung durch die unterschiedlich eingefärbten Bitreihen kenntlich gemacht. Oben rechts ist die Lage der den 32 Eingabegrößen zugeordneten Anteilen der MWPC bezüglich θ und ϕ dargestellt, oben links der von den Knoten der versteckten Schicht verwendete Teil der Sigmoid-Funktion. Es wurden nur Gewichte eines Knotens der versteckten Schicht und eines Knotens in der Eingabeschicht dargestellt.



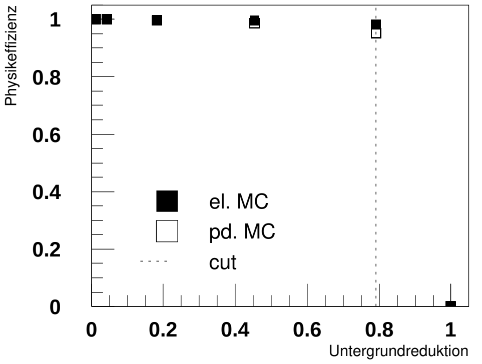
Ausgabeverteilung Untergrund. Die gepunktete Linie markiert den *cut*. Je weiter links vom *cut* ein Ausgabewert liegt, an desto mehr versteckten Knoten wurde die zugehörige Maske nicht erfüllt.



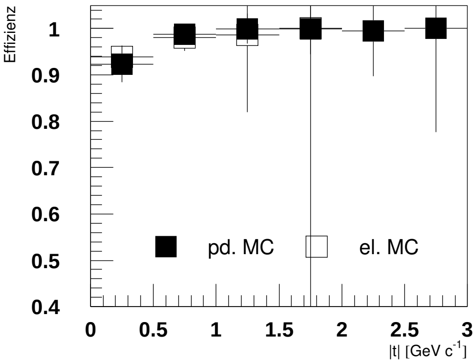
Ausgabeverteilung elastisches MC.



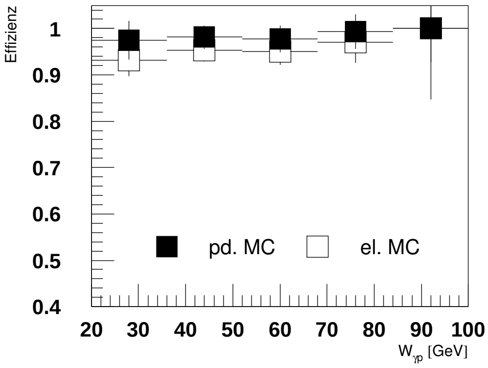
Ausgabeverteilung proton-dissoziatives MC.



Effizienzen für proton-dissoziatives (volle Quadrate) und elastisches (leere Quadrate) MC an den wenigen möglichen Betriebspunkten.

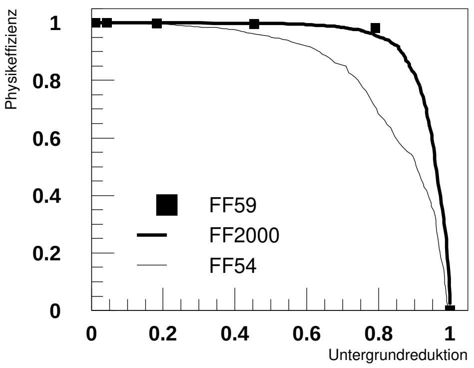


Effizienz in Abhängigkeit von $|t|$.

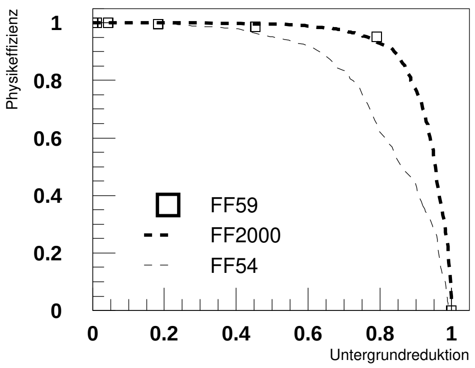


Effizienz in Abhängigkeit von $W_{\gamma p}$.

Abbildung 5.2: Eigenschaften von FF59



Vergleich der Netzeffizienzen für elastisches MC.



Vergleich der Netzeffizienzen für proton-dissoziatives MC.

Abbildung 5.3: Vergleich der Effizienzen aller angesprochenen Netze

Das Ergebnis ist das Netz FF59. In Abbildung 5.2 sind die Ausgabeverteilungen dieses Netzes für Untergrund und Monte-Carlo-Ereignisse dargestellt, ebenso die möglichen Kombinationen von Untergrundreduktion und Physikeffizienz. Die Ausgabeverteilungen unterscheiden sich beträchtlich von denen eines normalen neuronalen Netzes.

Das Netz FF59 ist das beste derzeit verfügbare Netz zum Triggern von ϕ-Mesonen in Photoproduktion. Neben der besten Untergrundreduktion im Vergleich aller angesprochenen Netze ist vor allem die gleichbleibende Physikeffizienz für unterschiedliche Werte von $|t|$ hervorzuheben. Konstruktionsbedingt können Untergrundreduktion und Physikeffizienz bei FF59 nur auf die in Abbildung 5.2, Mitte links, markierten Werte eingestellt werden. Deshalb ist eine feinere Abstimmung auf den *prescale* des S68 nicht möglich. Ein Vergleich der Effizienzen aller in dieser Arbeit angesprochenen Netze ist in Abbildung 5.3 durchgeführt. Die unterschiedlichen Effizienzen für elastische und proton-dissoziative Monte-Carlo-Daten sind im wesentlichen auf die unterschiedlichen p-Verteilungen zurückzuführen. Konstruktionsbedingt ist das Netz für Ereignisse, bei denen der Winkel zwischen den beiden Kaonen unter $45°$ liegt, vollständig effizient und für Winkel über $67,5°$ vollständig ineffizient. Dazwischen nimmt die Effizienz linear ab.

Das Netz FF59 wird seit August 1999 (*run* 249312) eingesetzt. Es reduziert die Rate des S68 um etwa 80 % bei einer Physikeffizienz von etwa 95 % bis 98 % (für elastische bzw. proton-dissoziative Ereignisse).

5.2 Analyse der Daten

Eine Analyse von Photoproduktionsdaten von ϕ-Mesonen bei H1 wurde in [43] durchgeführt. In der vorliegenden Arbeit wird, über ihren ursprünglich geplanten Rahmen hinausgehend, diese Analyse auf alle bis jetzt mit dem im letzten

Abschnitt vorgestellten neuronalen Netz gewonnenen Daten angewendet. Dies schließt neben den bereits in [43] verwendeten Daten der zweiten Hälfte des Jahres 1999 auch Daten aus dem Jahr 2000 ein. Aus Gründen, die unten erläutert werden, werden andere von [43] verwendete Daten nicht benutzt.

Um die Effizienzen aller in der Analyse angewandten Schnitte bestimmen zu können, wurden Monte-Carlo-Datensätze für elastische und proton-dissoziative Ereignisse generiert und rekonstruiert. Dazu wurde der DIFFVM-Generator verwendet, die Fragmentation wurde nach JETSET durchgeführt. Die weiteren Parameter des Monte-Carlo-Datensatzes wurden aus [43] übernommen.

Um die photoproduzierten ϕ-Mesonen anzureichern, werden auf die getriggerten (und auf DST verfügbaren) Ereignisse geeignete Schnitte angewendet:

- Nur „gute“ oder „mittlere“ *runs*[1] werden analysiert, in denen alle Detektorkomponenten, die in der Analyse verwendet wurden, aktiv waren. Die zu einem *run* gehörende Luminosität wird mit dem *prescale* des S68 in diesem *run* gewichtet. Bereiche eines *runs*, in denen bestimmte Subdetektoren kurzfristig ausfallen (beispielsweise durch kurzfristig zusammengebrochene Hochspannung an den Spurkammern, „*tripp*“), werden bei der Bestimmung der Luminosität berücksichtigt. Monte-Carlo-Ereignisse gehören per definitionem immer zu „guten“ *runs*.

- Damit ein Ereignis analysiert werden kann, muss es von Subtrigger S68 getriggert und von L2 und L4 verifiziert worden sein. Von L4 wird es in die Ereignisklasse 13 eingeteilt.[2]

- Die erste Forderung an ein Ereignis ist die nach genau zwei Spuren. Falls weniger Spuren existieren, können die Kaonen nicht rekonstruiert werden, bei mehr Spuren wird das Ereignis als Untergrund verworfen. Um auch Ereignisse zu unterdrücken, in denen ungeladene Teilchen auftreten, die zu keinem Signal in den Spurkammern führen, darf die Energiedeposition im Flüssig-Argon-Kalorimeter außerhalb des Bereichs der beiden Kaonen $0,5$ GeV nicht überschreiten.

- Die beiden geladenen Teilchen werden durch ihren differentiellen Energieverlust dE/dx in den Spurkammern mit einer bestimmten Wahrscheinlichkeit als Kaonen identifiziert (vgl. Abbildung 5.4, links). Ihre invariante Masse soll in einem Fenster um die ϕ-Masse (hier $0,98$ GeV $c^{-2} < m_\phi < 1,06$ GeV c^{-2}) liegen (vgl. Abbildung 5.4, rechts). Das hinreichend breite

[1] Alle *runs* werden in „gut“, „mittel“ oder „schlecht“ eingeteilt, je nachdem, wie viele der essentiellen Detektorkomponenten aktiv waren. Daten „schlechter“ *runs* werden nicht auf DSTs zur Verfügung gestellt.

[2] Klasse 13 enthält (semi-)exklusive Endzustände in diffraktiven Ereignissen, die von neun verschiedenen L4-Findern selektiert wurden. Ein L4-Finder ist ein Unterprogramm von L4, das Ereignisse einer bestimmten Physikklasse selektiert.

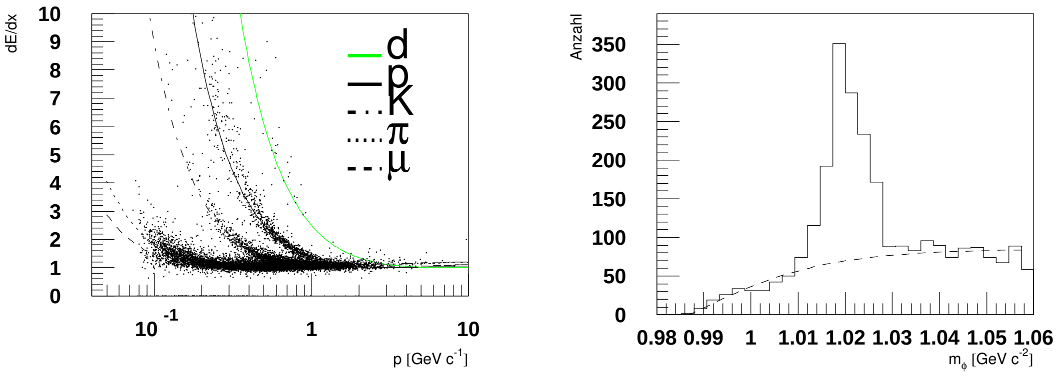


Abbildung 5.4: Links: Unterscheidung verschiedener Teilchen durch ihren differentiellen Energieverlust dE/dx in den Spurkammern. Rechts: Massenverteilung des verwendeten Datensamples aus den Jahren 1999 und 2000. Dabei wurde bereits der erweiterte Schnitt gegen Elektroproduktion angewandt, der unten eingeführt wird.

Fenster wird für die Bestimmung des nach allen Schnitten verbleibenden Untergrunds benötigt.

- Photoproduktion wird in [43] definiert durch die Abwesenheit eines Signals im SpaCal. Zugelassen werden nur Cluster mit einer Energie unter $0,5$ GeV, wenn die gesamte im SpaCal deponierte Energie unter $1,0$ GeV liegt. Dieser Schnitt wird in dieser Arbeit durch die zusätzliche Verwendung des VLQ-Kalorimeters verschärft (s. Seite 83).

Ein typisches Ereignis, das alle diese Schnitte erfüllt, ist in Abbildung 5.5 dargestellt. Es wurde Ende Januar 2000 durch die Kombination S68 und FF59 getriggert.

Für die im Datensample verbliebenen angereicherten ϕ-Meson-Kandidaten muss nun die Ereigniskinematik bestimmt werden: Die Variable y kann gemäß Gleichung 4.4 bestimmt werden und daraus dann aus Gleichung 4.8 die Schwerpunktsenergie im γp-System $W_{\gamma p}$. Der Transversalimpuls p_t des entstandenen ϕ-Mesons wird aus den in den Spurkammern gemessenen Transversalimpulsen der Kaonen errechnet.

Damit können die Ereignisse nun sowohl nach $W_{\gamma p}$ als auch nach p_t^2 in Bins eingeteilt werden. Die Zahl der Ereignisse in einem Bin wird durch einen Fit an eine von Monte-Carlo-Ereignissen generierte Verteilung, die sowohl die Breite des ursprünglichen Zustands (entsprechend einer Breit-Wigner-Funktion) als auch die weitere Verschmierung des Signals durch die endliche Detektorauflösung (resultierend in einer Gaußfunktion) berücksichtigt, bestimmt. Der Untergrund wird durch eine einfache Funktion (Tangens hyperbolicus) parametrisiert und dann subtrahiert.

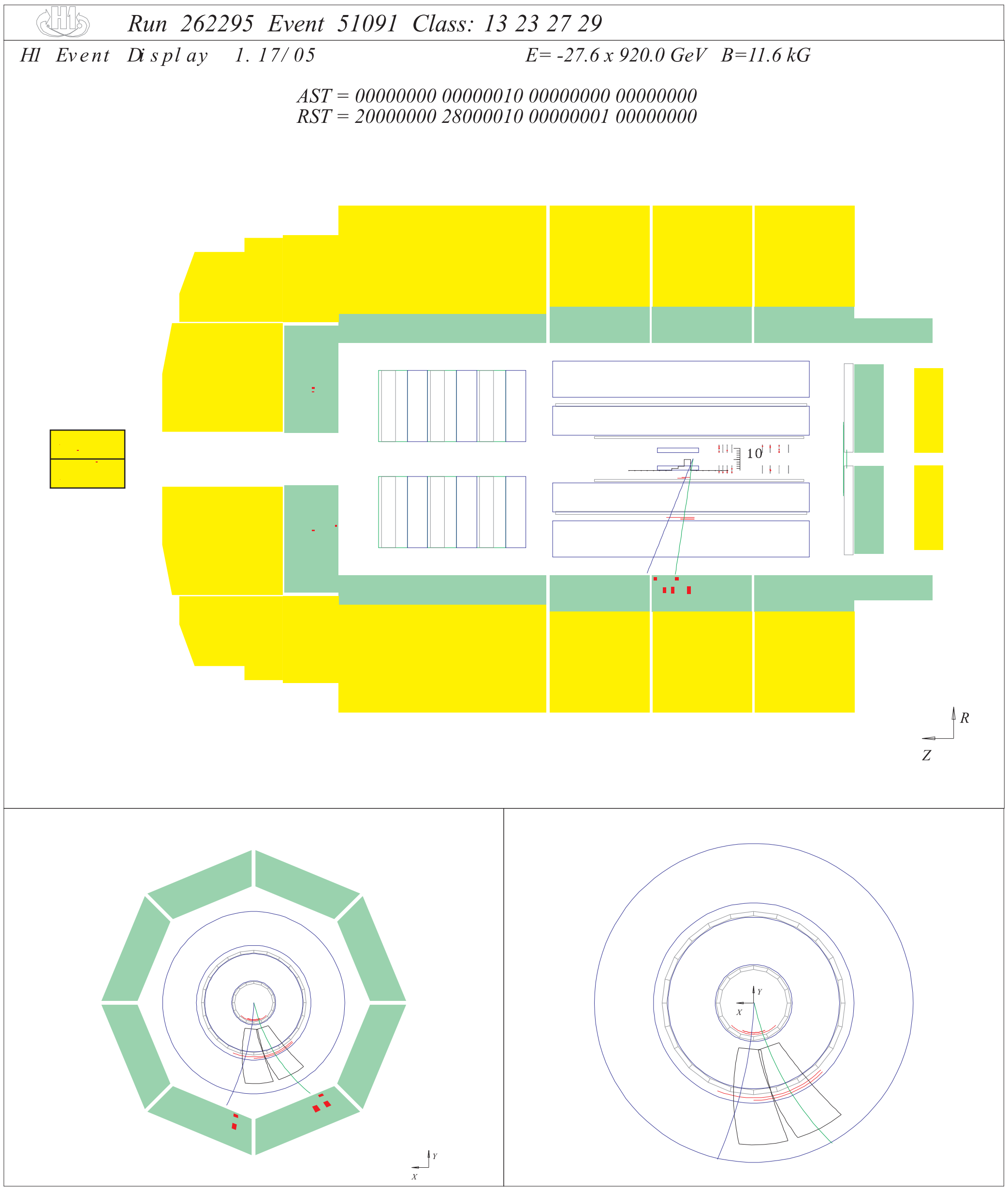


Abbildung 5.5: Darstellung eines Ereignisses mit einem in zwei Kaonen zerfallenden ϕ-Meson. Zu erkennen sind die Spuren der beiden Kaonen, die von ihnen aktivierten Masken in der CJC und die ausgelösten *rays*. Die Energiedepositionen im Kalorimeter sind zur Verdeutlichung vergrößert abgebildet. Weiter ist das generierte z-Vertex-Histogramm dargestellt, ebenso die Physikklassen (unter anderem die hier interessierende Klasse 13), in die das Ereignis eingeordnet wurde. RST bezeichnet die *raw* Subtrigger, auf die *monitoring* und *prescale* keinen Einfluss haben, AST die *actual* Subtrigger, die diese Effekte berücksichtigen. Die gesetzten Subtrigger werden durch gesetzte Bits in hexadezimaler Darstellung angezeigt.

Zwischen elastischen und proton-dissoziativen Ereignissen kann nicht eindeutig unterschieden werden. Sie erzeugen aber mit unterschiedlicher Wahrscheinlichkeit ein Signal in einem der Vorwärtsdetektoren. Deshalb können die beiden Anteile durch eine Entfaltung bestimmt werden. Als Entfaltung wird hier folgende Methode bezeichnet: Durch Multiplikation mit einer 2×2-Matrix wird der Vektor $\vec{x} = \begin{pmatrix} N_{w/fwd} \\ N_{wo/fwd} \end{pmatrix}$, dessen Komponenten angeben, wie viele Ereignisse mit und ohne Vorwärtsaktivität beobachtet wurden, in den Vektor $\vec{y} = \begin{pmatrix} N_{el} \\ N_{pd} \end{pmatrix}$ übergeführt, der angibt, wieviele elastische und proton-dissoziative Ereignisse stattfanden. Der Inhalt der Entfaltungsmatrix wird aus Monte-Carlo-Daten bestimmt.

Ein Ereignis gilt als vorwärtsaktiv, wenn eine der folgenden Bedingungen erfüllt ist:

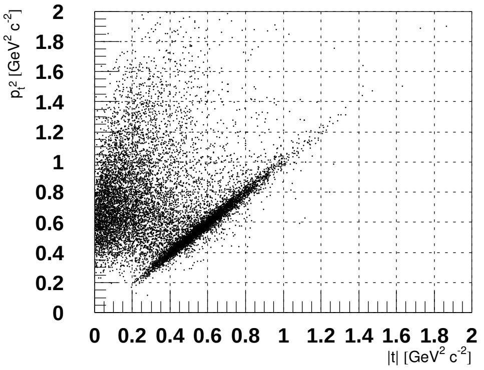
- Es gibt mehr als ein Trefferpaar in den inneren drei Modulen des Vorwärtsmyonspektrometers. Ein Trefferpaar wird noch als Rauschen gewertet. Es werden nur die Module des Vorwärtsmyonspektrometers benutzt, die innerhalb des Toroidmagneten liegen, da die äußeren durch Synchrotronstrahlung beeinflusst werden.
- Es gibt einen Treffer in mindestens einem der *proton remnant tagger* 0, 1 oder 2.
- Die im IF Bereich des Flüssig-Argon-Kalorimeters deponierte Energie übersteigt $0,75$ GeV.

Das Ziel der Analyse ist die Bestimmung des differentiellen Wirkungsquerschnitts von ϕ-Mesonen in Photoproduktion bezüglich t (also $d\sigma/dt$) für verschiedene $W_{\gamma p}$-Bereiche, separat für elastische und proton-dissoziative Ereignisse. Der Wert von t ist keiner direkten Messung zugänglich, da das auslaufende hadronische System nicht vollständig beobachtet wird. Allerdings kann t für kleine Q^2 gemäß

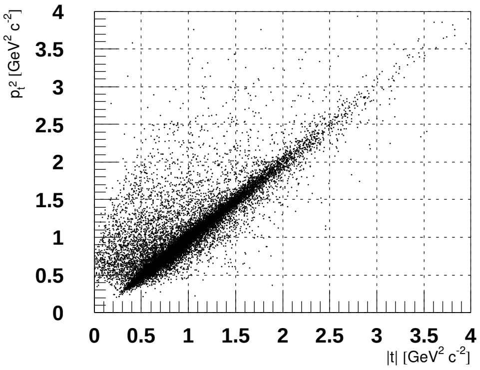
$$\begin{aligned}
t &= (q-V)^2 = -Q^2 - 2q \cdot V + m_{VM}^2 \\
&\approx -2E_\gamma \left(E_{VM} + p_{zVM}\right) + m_{VM}^2 \\
&\approx -\left(E_{VM} - p_{zVM}\right)\left(E_{VM} + p_{zVM}\right) + m_{VM}^2 \\
&= -\left(E_{VM}^2 - p_{zVM}^2\right) + m_{VM}^2 = -p_t^2 \qquad (5.1)
\end{aligned}$$

durch $-p_t^2$ genähert werden [43]. Dabei ist q der Viererimpuls und E_γ die Energie des Photons, V der Viererimpuls, m_{VM} die Ruhemasse, p_{zVM} die z-Komponente des Impulses und E_{VM} die Energie des Vektormesons.

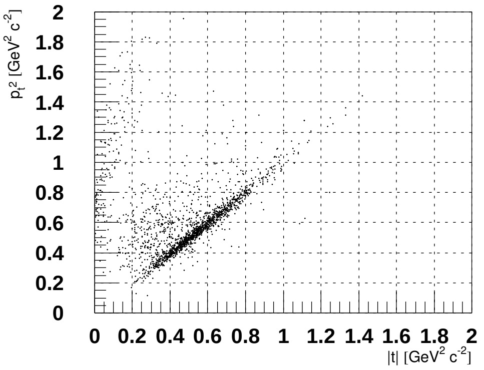
Der Transversalimpuls p_t des ϕ-Mesons kann aus den Impulsen der Zerfallskaonen bestimmt werden. Deshalb muss das t-Spektrum genau wie oben die Anteile von elastischen und proton-dissoziativen Ereignissen durch eine Entfaltung



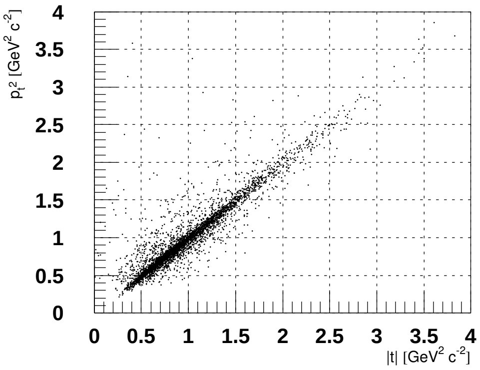
Elastische Ereignisse ohne VLQ-Schnitt.



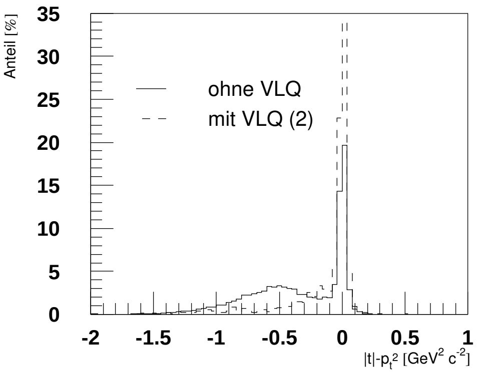
Proton-dissoziative Ereignisse ohne VLQ-Schnitt.



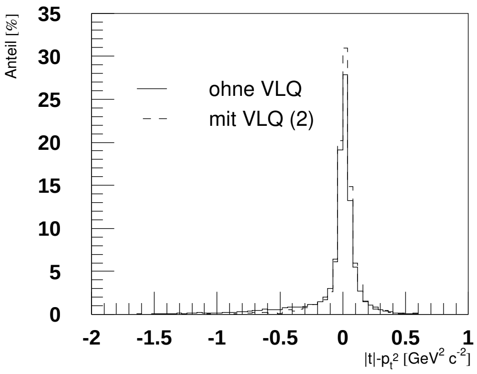
Elastische Ereignisse mit VLQ-Schnitt und Einschränkung von ϕ_ϕ.



Proton-dissoziative Ereignisse mit VLQ-Schnitt und Einschränkung von ϕ_ϕ.



Vergleich der Projektionen für elastische Ereignisse.



Vergleich der Projektionen für proton-dissoziative Ereignisse.

Abbildung 5.6: Zusammenhang zwischen p_t^2 und $|t|$ für verschiedene Monte-Carlo-Datensätze und unterschiedliche Schnitte gegen Ereignisse mit $Q^2 > 0$. Oben und in der Mitte jeweils Scatterplots, unten Abweichung des zu messenenden $|t|$ vom wirklichen Wert p_t^2. Die Scatterplots können als Anhaltspunkt dienen, wie diagonal die Entfaltungsmatrizen, die aus ihnen durch Binning in p_t^2 und $|t|$ gebildet werden, sind.

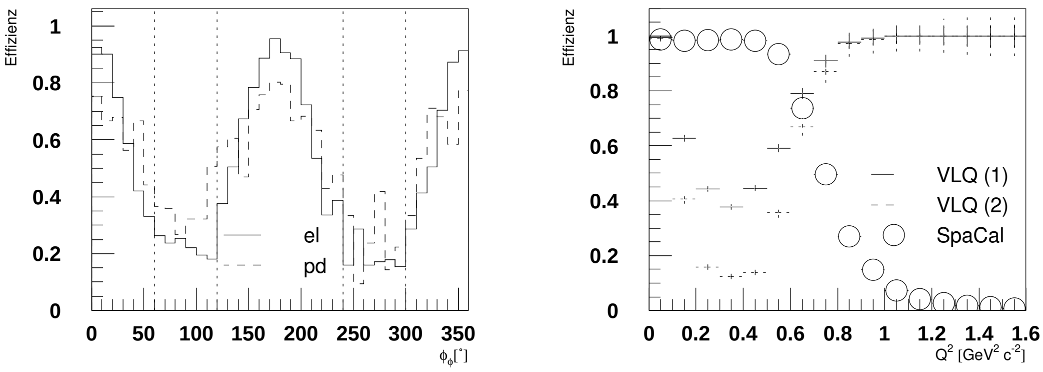


Abbildung 5.7: Links: Winkelabhängige Effizienz des VLQ-Schnitts für Ereignisse mit $0,1\ \mathrm{GeV}^2c^{-2} \leq Q^2 \leq 1,0\ \mathrm{GeV}^2c^{-2}$. Rechts: Effizienz der im Text angesprochenen Schnitte gegen Ereignisse mit $Q^2 > 0$. Die Legenden bedeuten: VLQ (1): Es werden alle Ereignisse verwendet. VLQ (2): Es werden nur Ereignisse verwendet, deren ϕ_ϕ auf Werte eingeschränkt ist, in denen das VLQ-Kalorimeter effizient ist.

aus dem gemessenen p_t-Spektrum bestimmt werden. Dazu werden die Ereignisse in sieben p_t^2-Bins eingeteilt. Aus dem Vektor, der die Zahl der Ereignisse in den einzelnen p_t^2-Bins enthält, wird durch Multiplikation mit einer 7×7-Matrix der Vektor bestimmt, der die Zahl der Ereignissen in den (ebenfalls sieben) $|t|$-Bins enthält. Die Entfaltungsmatrizen werden wiederum aus Monte-Carlo-Daten bestimmt.

Es kann zu Problemen führen, wenn diese Entfaltungsmatrizen nicht diagonalisiert werden können, wie das in [43] bei den elastischen Ereignissen auftritt (s. Abbildung 5.6, oben). Die wesentliche Ursache für die Nichtdiagonalisierbarkeit der Matrix ist die zu starke Verunreinigung des Datensatzes durch Ereignisse mit nicht verschwindendem Q^2. Dies rührt daher, dass mit dem SpaCal nur auf die Bedingung $Q^2 < 1,0\ \mathrm{GeV}^2c^{-2}$ getestet werden kann. Die Effizienz dieses Schnitts gegen Ereignisse mit von Null verschiedem Q^2 ist in Abbildung 5.7, rechts dargestellt. Die Linie ist mit „SpaCal“ bezeichnet.

In dieser Arbeit wurde daher zusätzlich das erst seit kurzer Zeit verfügbare VLQ-Kalorimeter verwendet. Analog zum SpaCal werden alle Ereignisse, die einen Cluster im VLQ-Kalorimeter besitzen, aus dem Datensatz entfernt. Dies sind Ereignisse mit $0,1\ \mathrm{GeV}^2c^{-2} < Q^2 < 1,0\ \mathrm{GeV}^2c^{-2}$. Die Effizienz dieses Schnitts ist ebenfalls in Abbildung 5.7, rechts dargestellt und mit „VLQ (1)“ bezeichnet.

Das VLQ-Kalorimeter ist nicht radialsymmetrisch aufgebaut, sondern deckt nur zwei ϕ-Bereiche im oberen und unteren Bereich des Detektors ab. Das bedeutet, dass die Effizienz des angewendeten Schnitts von der ϕ-Koordinate des Elektrons abhängt. Diese ist natürlich nicht bestimmbar, wenn das Elektron nicht nachgewiesen wird. Sie ist aber mit der ϕ-Koordinate des ϕ-Mesons korreliert, die

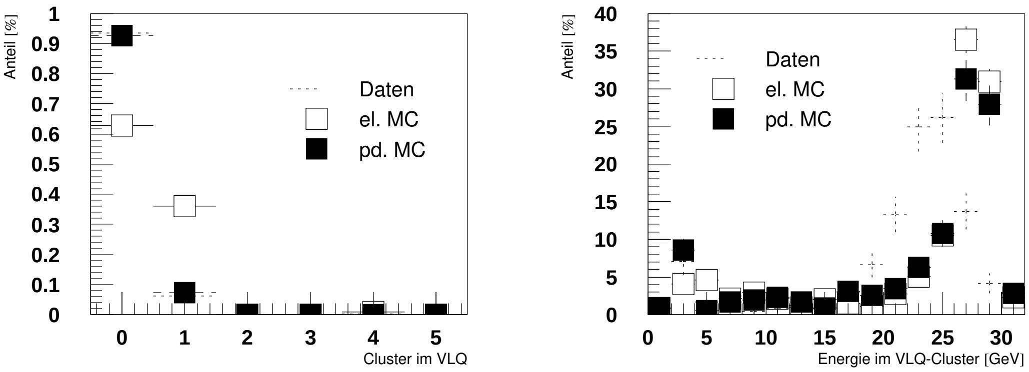


Abbildung 5.8: Vergleich von Monte-Carlo-Ereignissen mit Daten. Links: Anzahl der auftretenden Cluster im VLQ. Bedingt durch den Typ der Ausleseelektronik des VLQ werden statt zwei Clustern immer vier rekonstruiert. Rechts: Signal im VLQ. Zu erkennen ist der kinematische Peak der gestreuten Elektronen.

rekonstruiert wird. Diese Abhängigkeit der Effizienz des Schnittes von ϕ_ϕ ist in Abbildung 5.7, links dargestellt. Die Effizienz des Schnitts kann also erhöht werden, wenn nur Ereignisse betrachtet werden, bei denen das ϕ-Meson in einem der Winkelbereiche $60° \leq \phi_\phi \leq 120°$ oder $240° \leq \phi_\phi \leq 300°$ rekonstruiert wird. Die resultierende Effizienz ist wiederum in Abbildung 5.7, rechts dargestellt und mit „VLQ (2)“ bezeichnet. Die Anwendung dieses letzen Schnitts reduziert die verfügbare Statistik um den Faktor drei, erhöht aber die Reinheit des Datensatzes nochmals.

Die Signale im VLQ-Kalorimeter werden für proton-dissoziative Ereignisse durch das Monte-Carlo in akzeptabler Weise beschrieben (s. Abbildung 5.8). Für elastische Ereignisse ist die Übereinstimmung deutlich schlechter.

Durch die Verwendung von VLQ-Spektrometer und zusätzlichem Schnitt auf bestimmte ϕ_ϕ-Bereiche wird die aus den Monte-Carlo-Daten erhaltene p_t^2-t-Matrix wesentliche diagonaler (s. Abbildung 5.6, Mitte), so dass eine Analyse der elastischen Ereignisse möglich scheint.

Da die Daten des VLQ-Kalorimeters erst seit Mitte des Jahres 1999 auf DST verfügbar sind, muss darauf verzichtet werden, die in [43] ebenfalls analysierten Daten aus der ersten Hälfte des Jahres 1999 und dem Jahr 1996 zu verwenden.

5.3 Differentieller Wirkungsquerschnitt für ϕ-Mesonen in Photoproduktion

Trotz der Einbeziehung des VLQ-Kalorimeters zur Unterdrückung von Ereignissen mit $Q^2 > 0$ und trotz der sich damit ergebenden Verbesserung durch eine diagonalere p_T^2-t-Matrix ist eine Analyse des elastischen Wirkungsquerschnitts von ϕ-Mesonen in Photoproduktion nicht möglich, da das Verhalten der Analyse bei Tests mit Monte-Carlo-Daten nicht hinreichend stabil war[3]. Vermutlich ist dies auf die Überlagerung der sehr unterschiedlich großen elastischen und proton-dissoziativen Wirkungsquerschnitte für größere Werte von $|t|$ zurückzuführen. Zudem ist die Beschreibung der Daten durch das elastische Monte-Carlo-Sample nicht ausreichend. Deshalb beschränkt sich auch diese Arbeit (wie [43]) auf Ergebnisse, die den differentiellen Wirkungsquerschnitt proton-dissoziativer Photoproduktion von ϕ-Mesonen betreffen.

Der Wirkungsquerschnitt der untersuchten Reaktion σ_{ep} kann für kleine Q^2 faktorisiert werden in die Wahrscheinlichkeit für die Emission eines Photons durch das einlaufende Elektron (Flussfaktor F) und den γp-Wirkungsquerschnitt $\sigma_{\gamma p}$. Näherungsweise kann diese Faktorisierung durch

$$\sigma_{\gamma p} = \frac{\sigma_{ep}}{F} \tag{5.2}$$

formuliert werden, wo F mit Hilfe der Weizäcker-Williams-Approximation bestimmt wird. Die Schwerpunktsenergie $W_{\gamma p}$ lässt sich mit der Schwerpunktsenergie $\sqrt{s}$ der Elektron-Proton-Streuung identifizieren.

Mit Hilfe von VDM und optischer Analogie kann in Einklang mit den experimentellen Beobachtungen für den differentiellen Wirkungsquerschnitt $d\sigma/dt$ folgende Parametrisierung verwendet werden [44]:

$$\frac{d\sigma}{dt} = \left.\frac{d\sigma}{dt}\right|_{t=0, W_{\gamma p}=W_0} e^{b(W_{\gamma p})t} \left(\frac{W_{\gamma p}}{W_0}\right)^{4\epsilon} \tag{5.3}$$

mit

$$b(W_{\gamma p}) = b_0 + 4\alpha' \ln \frac{W_{\gamma p}}{W_0} \tag{5.4}$$

Wie oben ausgeführt, wird die Statistik im Vergleich zu [43] dadurch reduziert, dass, um den zusätzlich angewendeten VLQ-Schnitt effizient zu machen, $60° \leq \phi_\phi \leq 120°$ oder $240° \leq \phi_\phi \leq 300°$ gefordert wird. Dadurch wird der Datensatz auf ein Drittel reduziert. Eine Aufteilung des Datensatzes in einem t-Bereich auf mehrere $W_{\gamma p}$-Bereiche erscheint dann nicht mehr sinnvoll. Deshalb wurde darauf und auf die damit mögliche Bestimmung von α' verzichtet und nur $b(\langle W \rangle)$ bestimmt.

Eingang in die Messung fanden Daten, die von September 1999 bis März 2000 aufgezeichnet und rekonstruiert wurden. Die gesamte Luminosität betrug unter Berücksichtigung der *prescales* des S68 und nach Korrekturen wegen kurzfristig ausgefallener Hochspannung $6,84\ \text{pb}^{-1}$. Die Daten wurden mit der in [43] vorgestellten Analyse ausgewertet, wobei diese Analyse durch den oben beschriebenen VLQ-Schnitt mit Einschränkung des Winkelbereichs von ϕ_ϕ erweitert wurde. Alle verwendeten Daten wurden vom neuronalen Netz FF59 getriggert.

[3] Beim Test der Analyse mit Monte-Carlo-Daten gelang es nicht, immer die Parameter, die der als „Datensatz" verwendetete Monte-Carlo-Datensatz besaß, mit der Analyse zu reproduzieren.

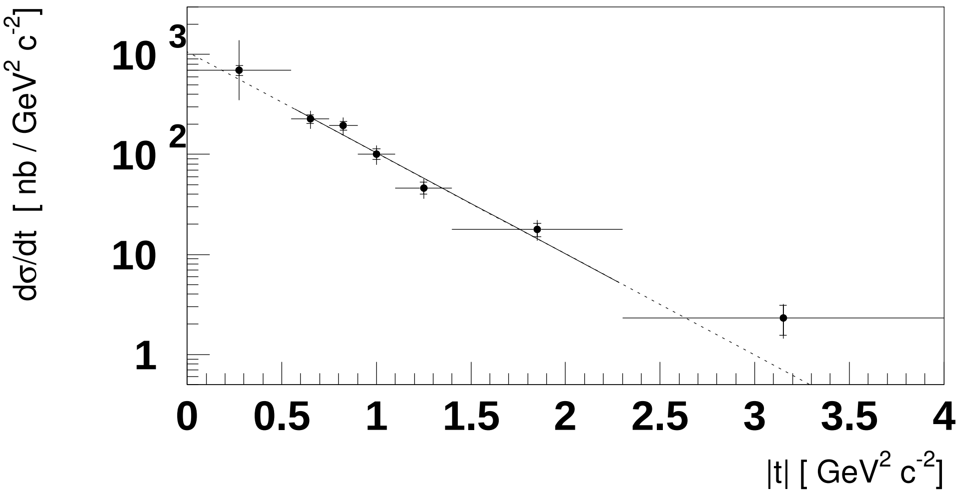


Abbildung 5.9: Bestimmung des b-Parameters beim proton-dissoziativen Wirkungsquerschnitt von ϕ-Mesonen in Photoproduktion. Die inneren vertikalen Fehlerbalken bezeichnen den statistischen Fehler, für die äußeren Fehler wurde statistischer und systematischer Fehler quadratisch addiert. Es wurden beim Fit nur die Bins 2 mit 6 berücksichtigt, da im erste Bin Unsicherheiten bezüglich der Triggereffizienz bestehen und das letzte Bin nicht mit einer exponentiellen Abhängigkeit verträglich ist.

Die Bestimmung sowohl des statistischen als auch des systematischen Fehlers der einzelnen Datenpunkte wird aus [43] übernommen. Wesentlich zum systematischen Fehler tragen demnach bei der Bestimmung von b die verwendete Methode, die Trennung von proton-dissoziativen und elastischen Ereignissen sowie die Subtraktion des Untergrundes bei. Dies führt zu einem gesamten systematischen Fehler von 10 %. Dazu ist außer dem statistischen auch noch ein Fehler zu addieren, der sich aus der in gewissen Grenzen willkürlichen Wahl der Parameter des verwendeten Monte-Carlo-Modells ergibt. Er liegt bei etwa 10 %.

Für den Parameter b ergibt sich im betrachteten Bereich von $0,55\ \mathrm{GeV}^2c^{-2} < |t| < 2,30\ \mathrm{GeV}^2c^{-2}$ und $35\ \mathrm{GeV} < W < 100\ \mathrm{GeV}$ (entspricht $\langle W\rangle = 57$ GeV) für den proton-dissoziativen Wirkungsquerschnitt ein Wert von (vgl. auch Abbildung 5.9)

$$b = 2,28 \pm 0,31(\mathrm{stat.}) \pm 0,23(\mathrm{syst.}) \pm 0,23(\mathrm{mod.})\ \mathrm{GeV}^{-2}c^2. \qquad (5.5)$$

In [43] wurde für denselben W- und t-Bereich $b = 2,89 \pm 0,10$ (stat.) $\pm$ $0,29$ (syst.) $\pm$ $0,29$ (mod.) $\mathrm{GeV}^{-2}c^2$ erhalten, eine vergleichbare Messung von ZEUS für $\langle W\rangle = 94$ GeV und $0,50\ \mathrm{GeV}^2c^{-2} < |t| < 1,30\ \mathrm{GeV}^2c^{-2}$ ergab einen Wert von $2,1 \pm 0,5$ (stat.) $\pm\, 0,3$ (syst.) $\pm\, 0,4$ (mod.) $\mathrm{GeV}^{-2}c^2$ [45].

Damit steht diese Messung in guter Übereinstimmung mit bisherigen Messungen.

Zusammenfassung

Es ist geplant, den neuronalen Trigger auf der zweiten Triggerstufe beim H1-Experiment (L2NN) zu verbessern, indem die Triggerdaten, bevor sie dem neuronalen Netz präsentiert werden, nach physikalischen Kriterien vorverarbeitet werden. Diese Datenaufbereitung wird vor allem topologische Korrelationen der Daten untereinander („intelligente“ Datenvorverarbeitung) berücksichtigen.

Im Rahmen dieser Arbeit wurde eine Simulation dieser „intelligente“ Datenvorverarbeitung erstellt, mit der die Leistungsfähigkeit dieses Ansatzes nachgewiesen wurde. Die Hardwarekomponente, die dieses Konzept nutzt, befindet sich zurzeit, basierend auf den hier niedergelegten Untersuchungen, in der Entwicklungsphase. Sie wird mit DDB2 bezeichnet.

Das zugrundeliegende Konzept der Datenaufbereitung ist, aus den einzelnen Signalen der unterschiedlichen Detektorkomponenten physikalische „Objekte“ zu bilden, die die in den Reaktionen erzeugten Teilchen und Teilchenbündel (Jets) charakterisieren. Die Eigenschaften dieser Objekte werden dann dem neuronales Netz als Eingabedaten präsentiert.

Im Einzelnen besteht die implementierte Datenaufbereitung aus folgenden Schritten: Innerhalb der Detektorkomponenten werden Signale zusammengefasst, die von einem Teilchen (oder Jet) stammen. Dann werden Signale der verschiedenen Detektorkomponenten zu Objekten kombiniert, wenn sie sich räumlich nahe sind, also zum Wechselwirkungspunkt projizieren. Um wichtige Objekte von unwichtigen zu trennen, werden die Objekte nach ihren physikalischen Eigenschaften sortiert. Die Eigenschaften dieser sortierten Objekte können dann entweder direkt einem neuronalen Netz präsentiert werden oder nochmals weiterverarbeitet werden. Dabei können zum einen die Winkel zwischen zwei Objekten bestimmt werden, zum anderen kann gezählt werden, wie viele Objekte mit spezifizierten Eigenschaften gebildet wurden.

Bei der Umsetzung dieses Konzeptes in die Simulation wurde darauf geachtet, dass die verwendeten Algorithmen gut in die geplante Hardwarekomponente umgesetzt werden können. Insbesondere die Zeit, die für die Ausführung der Algorithmen benötigt wird und der Bedarf an Hardwareressourcen (wie zum Beispiel Speicher) wurde berücksichtigt.

Um nachzuweisen, wie leistungsfähig das Konzept ist, wurde mit Hilfe der erstellten Simulation ein neuronales Netz trainiert und seine Leistung mit der eines

Netzes verglichen, das die „intelligente“ Datenvorverarbeitung nicht verwendet. Als Physikkanal für diesen Vergleich wurde die Photoproduktion von ϕ-Mesonen gewählt. Das Netz mit „intelligenter“ Datenvorverarbeitung erreicht eine Effizienz bei Physikereignissen von bis zu 95 % bei einer Reduktion des Untergrunds um 80 %, während das bisherige Netz nur eine Effizienz bei Physikereignissen von 81 % bei gleichzeitig niedrigerer Reduktion des Untergrunds von nur 73 % erreicht.

Weiterhin wurden die Konzepte der „intelligenten“ Datenvorverarbeitung leicht abgewandelt, um sie bereits heute, also ohne neue Hardwarekomponente, einsetzen zu können. Es gelang, für den bereits zuvor untersuchten Physikkanal, die Photoproduktion von ϕ-Mesonen, ein weiteres neuronales Netz zu erstellen und im Herbst 1999 zum Einsatz zu bringen. Bei diesem neuronalen Netz wurden die Gewichte nicht durch Präsentation eines Beispieldatensatzes, sondern aufwendiger durch analytische Berechnung festgelegt. Es erreicht eine Effizienz bei Physikereignissen von bis zu 98 % bei einer Reduktion des Untergrundes von 80 %, übertrifft also noch das Netz mit „intelligenter“ Datenvorverarbeitung.

Aus den Daten, die mit diesem Netz vom Herbst 1999 bis zum Frühling 2000 gewonnen wurden, wurde die Abhängigkeit des proton-dissoziativen Wirkungsquerschnitts von ϕ-Mesonen vom Quadrat des Impulsübertrags $|t|$ auf das Proton bestimmt. Dazu wurde eine bestehende Analyse weiterentwickelt, indem eine vor kurzem bei H1 neu hinzugekommene Detektorkomponente, das VLQ, benutzt wurde, um Ereignisse mit einem negativen Quadrat des Impulsübertrages am Elektronvertex Q^2 von weniger als $1,0\ \mathrm{GeV}^2c^{-2}$ aus dem Datensatz zu entfernen. Es war nicht möglich, die auf diese Weise erweiterte Analyse auch zur Bestimmung des elastischen Wirkungsquerschnitts von ϕ-Mesonen in Photoproduktion anzuwenden.

Eine Anpassung für den proton-dissoziativen Wirkungsquerschnitt gemäß $d\sigma/dt \propto e^{bt}$ im Bereich $|t| \in [0,55\ \mathrm{GeV}^2c^{-2}, 2,30\ \mathrm{GeV}^2c^{-2}]$ ergab für die exponentielle Abhängigkeit des Wirkungsquerschnitts $b = 2,28 \pm 0,31$ (stat.) $\pm$ $0,23$ (syst.) $\pm\ 0,23$ (mod.) $\mathrm{GeV}^{-2}c^2$, was in guter Übereinstimmung mit den bis jetzt bei H1 und ZEUS durchgeführten Messungen steht.

Anhang A

DDB2-Simulation

Ein wichtiger Bestandteil dieser Arbeit war die Erstellung einer Simulation der DDB2. Die mehr programmspezifischen Aspekte dieser Simulation sind in diesem Anhang dokumentiert.

Die DDB2-Simulation ist in das Programm `NETSIM` eingebunden. Dieses Programm berechnet unter anderem die Eingabegrößen für ein Netzwerktraining. Die Gesamtheit aller möglichen Eingabegrößen wird im Folgenden als Eventvektor bezeichnet.

A.1 DDB2-Elemente des Eventvektors

Der herkömmliche Eventvektor der DDB enthält im Wesentlichen einfach strukturierte, voneinander unabhängige Größen [42]. Im Gegensatz dazu hängen die durch das Konzept der DDB2 neu hinzukommenden Größen in vielerlei Hinsicht voneinander ab und sind oft sogar „mehrdimensional“ angeordnet.

Ein großer Teil der von der DDB2 bereitgestellten Eingabegrößen repräsentiert nicht globale Zustände im Detektor (wie z.B. Energie im vorderen Bereich), sondern physikalische Objekte. Ein physikalisches **Objekt** hat mehrere Eigenschaften, seine **Elemente**. Maximal 16 Objekte bilden zusammen mit ihrer Anordnung einen **Vektor**. Es sind drei verschiedene Vektoren `U`, `V`, `W` verfügbar, denen jeweils eine eigene Sortierung zugeordnet ist. Die Objekte werden zu zwei unterschiedlichen Typen von Vektoren kombiniert: Zum einen Vektoren mit SpaCal-Objekten, zum anderen Vektoren mit allen anderen Objekten. Der Grund für diese Trennung sind die unterschiedlichen topologischen Regionen dieser Objekte.

Bei der Bildung des Namens einer Eingabegröße wird der Name des Vektors mit dem Namen des Elements und der Nummer des Objekts innerhalb des Vektors kombiniert. In einem Vektor sind die Objekte von 0 bis 15 durchnummeriert. Eine Ausnahme sind die Elemente `DPHI` und `DTHETA` (vgl. Tabelle A.2), deren Nummern aus den weiter unten ersichtlichen Gründen nur von 0 bis 14 reichen.

Die Namen der Elemente bestehen aus bis zu vier Teilen (vgl. Tabelle A.1):

Subdetektor	1	2	3	4
Kalorimeter *big towers*	L	E, H		SE, ET, EC, ER, NH
	L			ET
LAr *trigger towers*	T	E, H		SE, ET, EC, ER, NH
Proportionalkammern	P			SEED, ET
Driftkammern (CJC)	C	H, L	N, P	SE, ET
Myonkammern	M			SEED, ET
SpaCal	S	E		SE, ET, EC, ER, NH

Tabelle A.1: Elemente der Objekte aus den einzelnen Subdetektoren.

Der erste Teil (1) leitet sich vom Namen des **Subdetektors** her, die Teile (2) und (3) werden für feinere **Differenzierungen** verwendet: Mit ihnen wird die Unterscheidung zwischen den hadronischen (H) und elektromagnetischen (E) Kalorimetern und die Unterscheidung zwischen Masken für Spuren positiver (P) und negativer (N) Teilchen mit einem hohen (H) oder niedrigen (L) Impuls in der CJC getroffen. Der letzte Teil (4) bezeichnet die verschiedenen **Eigenschaften**, die ein Cluster hat: Solche Eigenschaften sind der Index des Clustermittelpunkts im Subdetektor (SEED oder abgekürzt SE), die gesamte Energie des Clusters (*total energy*, ET), die Energie im Zentrum des Clusters (EC), die Energie im Ring des Clusters (ER) und die Zahl von Zellen, aus denen der Cluster besteht (*number of hits*, NH). Der Begriff Energie kann je nach Subdetektor auch Pseudoenergie im Sinne der Abschnitte 3.4.3, 3.4.4 oder 3.4.5 bedeuten.

Weiterhin besitzen die Objekte Elemente, die globale Eigenschaften des ganzen Objekts festlegen. Diese sind in Tabelle A.2 und A.3 aufgeführt. Winkelindizes und Winkeldifferenzen wurden bereits in 3.4.1 definiert. Die Ladung Q des Objekts kann entweder negativ (Wert 0), neutral (1) oder positiv (2) sein.

Element	Beschreibung
THETA	θ-Index des Objekts
PHI	ϕ-Index des Objekts
DTHET	THETA0 - THETA$(i+1)$
DPHI	PHI0 - PHI$(i+1)$
Q	Ladung des Objekts

Tabelle A.2: Globale Elemente eines Objektes: Indizes, die die Position festlegen, Differenzen zwischen diesen Indizes und die Ladung.

Außerdem existieren unabhängig von den Vektoren die Eingabegrößen COUNT0 bis COUNT31. Diese stehen für die 32 möglichen Zähler für Objekte, deren Elemente bestimmte Bedingungen erfüllen.

Die Simulation stellt neben den oben beschriebenen Größen noch zusätzliche bereit, die entweder direkt die von den Subdetektoren bereitgestellten Datenformate oder die Zwischenergebnisse bestimmter Algorithmen enthalten. Für eine

Element	Beschreibung
SEPHI	ϕ-Koordinate des Objekts
SEDPH	PHI0 - PHI$(i+1)$
SERAD	r-Koordinate des Objekts

Tabelle A.3: Globale Elemente eines Objekts aus dem elektromagnetischen Spa-Cal: Koordinaten zur Positionsbestimmung sowie die Differenz zwischen zwei solchen Koordinaten.

Beschreibung dieser Größen sei auf [42] verwiesen. Wesentlich, da sehr gut als Netzeingabegrößen geeignet, sind `SPMAPB0` bis `SPMAPB24`. Diese Größen entsprechen den überschrittenen Schwellen in der groben Granularität des SpaCals.

Da die verwendeten Datenstrukturen (u.a. HBOOK [46]) gewisse Beschränkungen bezüglich ihrer Maximalgröße haben, werden nicht alle Elemente des Eventvektors von der Simulation ausgegeben. Allerdings betrifft dies nur einige der im letzten Abschnitt angesprochenen zusätzlichen Größen.

A.2 Steering der DDB2-Algorithmen

Das Steering der DDB2-Simulation wird wie allgemein üblich über BOS-Bänke [47] durchgeführt.

A.2.1 Z2NU (Aktivierung)

Zur Einsparung von Rechenzeit kann die DDB2-Simulation deaktiviert werden, indem eine Z2NU-Bank mit dem Wert 0 angegeben wird. Die Anteile des Eventvektors, die durch die DDB2-Simulation gefüllt werden, sind dann nicht gültig.

Ein Beispiel für eine Z2NU-Bank:

```
Z2NU
 1   ! 0: deactivate DDB2-simulation
     ! 1: activate DDB2-simulation
```

A.2.2 Z2NP (Parameter)

Die Z2NP-Bank setzt Parameter für die in der DDB2 verwendeten Algorithmen. Die Möglichkeiten der Einflussnahme auf *ordering*, *clustering*, *matching*, usw. werden in eigenen Abschnitten diskutiert. Die dort angegebenen Beispiele sind zusammengenommen eine gültige Z2NP-Bank.

Parameter für *ordering*

Die Parameter 1 mit 24 der Bank Z2NP definieren die Art der anzuwendenden Sortierungen. Sowohl für die „normalen“ Objekte, als auch für die der SpaCal-

Objekte sind drei Sortierungen vorgesehen. Die sortierten Objekte stehen dann in den Vektoren `U`, `V` und `W` zur Verfügung. Jede Definition einer Sortierung besteht aus vier Werten, und zwar dem Index des primären Sortierkriteriums, der Sortierreihenfolge bezüglich des primären Sortierkriteriums, dem Index des sekundären Sortierkriteriums und der Sortierreihenfolge bezüglich des sekundären Sortierkriteriums. Das sekundäre Sortierkriterium wird nur dann angewendet, wenn das primäre Sortierkriterium alleine keine Aussage über die Reihenfolge der Objekte zulässt.

Zur Definition der Indizes der „normalen" Sortierkriterien, der Sortierkriterien für die SpaCal-Objekte und der Sortierreihenfolgen vgl. Tabelle A.4, A.5 und A.6.

THETA	0	LET	13	TESE	26
PHI	1	PSEED	14	TEET	27
Q	2	PET	15	TEEC	28
LESE	3	CHPSE	16	TEER	29
LEET	4	CHPET	17	TENH	30
LEEC	5	CLPSE	18	THSE	31
LEER	6	CLPET	19	THET	32
LENH	7	CHNSE	20	THEC	33
LHSE	8	CHNET	21	THER	34
LHET	9	CLNSE	22	THNH	35
LHEC	10	CLNET	23		
LHER	11	MSEED	24		
LHNH	12	MET	25		

Tabelle A.4: Indizes für „normale" Sortierkriterien

SEPHI	0	SEET	3	SENH	6
SERAD	1	SEEC	4		
SESE	2	SEER	5		

Tabelle A.5: Indizes für SpaCal Sortierkriterien

Sortierreihenfolge	Vergleichsoperator	Index
aufsteigend nach Signum		2
aufsteigend	$<$	1
keine Sortierung		0
absteigend	$>$	-1
absteigend nach Signum		-2

Tabelle A.6: Indizes zur Festlegung der Sortierreihenfolge beim *ordering* und des Vergleichsoperators beim *counting*

Ein Beispiel für gültige Einstellungen:

```
Z2NP
 13  -1  15  -1  ! main ordering: >LET, >PET (1 - 4)
  0  -1   1   1  ! main ordering: >THETA, <PHI (5 - 8)
 15  -1  13  -1  ! main ordering: >PET, >LET (9 - 12)
  0  -1   3  -1  ! spacal ordering: >SEPHI, >SEET (13 - 16)
  3  -1   0  -1  ! spacal ordering: >SEET, >SEPHI (17 - 20)
  1  -1   0  -1  ! spacal ordering: >SERAD, >SEPHI (21 - 24)
```

Parameter für *clustering*

Die Parameter 25 mit 35 steuern die Clusteralgorithmen. In diese Clusteralgorithmen kann an zwei Stellen eingegriffen werden: Zum einen wird die Obergrenze für die Zahl von Clustern gesetzt, die von einem bestimmten Subdetektor (elektromagnetischem oder hadronischem Flüssig-Argon-Kalorimeter, jeweils *trigger towers* oder *big towers*, *big rays*, CJC) in den *matching*-Algorithmus eingebracht werden. Damit kann beispielsweise verhindert werden, dass durch rauschende *trigger towers* viele Objekte gebildet werden, die nur aus einem *trigger tower*-Signal bestehen, während andere Objekte, die zwar kein *trigger tower*-Signal besitzen, dafür aber ein CJC- und *big ray*-Signal durch die maximale Zahl von Objekten nicht berücksichtigt werden.

Parameter 30 wirkt auf einen alternativen Clusteralgorithmus für *big rays*, der im Augenblick nicht verwendet wird, ist also obsolet (vgl. auch [19]).

Die andere Eingriffsmöglichkeit ist die Festlegung einer minimalen Gesamtenergie (*threshold*) für Cluster aus *big* oder *trigger towers*. Die Forderung minimaler Energien in einzelnen *big* oder *trigger towers* vor dem Clustern ist durch die Einführung von LUTs zusätzlich möglich und sinnvoll.

Sinnvolle Einstellungen sind beispielsweise die hier gezeigten:

```
 10  ! Number of used LAr em BT jets (25)
 10  ! Number of used LAr hd BT jets (26)
 11  ! Number of used LAr em TT jets (27)
 11  ! Number of used LAr hd TT jets (28)
 10  ! Number of used Big rays (29)
 10  ! Number of used Big rays (alt. alg.) (30)
 22  ! Number of used CJC clusters (31)
  6  ! Threshold for LAr em BTs (32)
  6  ! Threshold for LAr hd BTs (33)
  6  ! Threshold for LAr em TTs (34)
  6  ! Threshold for LAr hd TTs (35)
```

Parameter für *matching*

Über die Parameter 36 mit 38 können bestimmte Eigenschaften des *matching*-Algorithmus gesteuert werden.

Zum einen kann gewählt werden, ob Objekte mit unvollständiger Winkelinformation zurückgewiesen werden sollen (1) oder nicht (0). Unvollständige Winkelinformation tritt auf, wenn ein Objekt nur aus einem oder mehreren Signalen in der CJC besteht. In diesem Fall ist die Winkelbestimmung in ϕ-Richtung zwar möglich, es kann aber keine Aussage über θ gemacht werden.

Außerdem kann festgelegt werden, ob die grob strukturierten *big towers* oder die mit höherer Auflösung arbeitenden *trigger towers* zum *matching* verwendet werden sollen. Es ist auch möglich beide zu verwenden. Wenn die *trigger towers* nicht verwendet werden, wird die Simulation dieser noch nicht vorhandenen Detektorkomponente nicht durchgeführt, was zu Zeitersparnis beim Programmablauf führt. Ein Wert von 1 steht für die Verwendung der Detektorkomponente, bei einem Wert von 0 wird die Komponente nicht verwendet.

Falls bestimmte Teile von Detektorkomponenten nicht verwendet werden sollen, empfiehlt sich die Verwendung von LUTs, um deren Eingangsdaten auf Null zu setzen.

Eine mögliche Einstellung zeigt das folgende Beispiel:

```
0  ! Do not reject clusters w/o complete coords (36)
0  ! Use trigger towers for matching (37)
1  ! Use big towers for matching (38)
```

Parameter für *post-processing*

Die letzte Gruppe von Einstellmöglichkeiten bezieht sich auf das *post processing* und dabei auf die Definition der 32 Zähler. Der Wert eines Zählers ist die Anzahl von Objekten, deren Elemente eine frei definierbare Bedingung erfüllen. Die Bedingungen werden über die Z2NP-Bank festgelegt, und zwar durch die Parameter ab Position 39 einschließlich. Jede Bedingung wird durch insgesamt zwölf Parameter definiert, die sich in vier Gruppen zu je drei aufteilen. Jede der Gruppen definiert eine Teilbedingung. Die vier Teilbedingungen $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$ und $\mathcal{D}$ werden durch eine logische Verknüpfung miteinander zur Gesamtbedingung $\mathcal{Z}$ kombiniert[1]:

$$\mathcal{Z} = (\mathcal{A} \&\& \mathcal{B}) \,||\, (\mathcal{C} \&\& \mathcal{D}) \tag{A.1}$$

Jede der Teilbedingungen ist von der Form

$$\mathcal{A} = (A \leq a) \qquad \text{oder} \qquad \mathcal{A} = (A \geq a)\,, \tag{A.2}$$

wobei A ein Element des Eventvektors mit dem Index α und a eine Zahl zwischen 0 und 255 ist.

Der erste Parameter ist der Index α. Er kann aus den Tabellen A.4 und A.5 abgelesen werden. Wenn ein Element aus Tabelle A.5 verwendet werden soll, muss zu dem dort angegebenen Index 36 addiert werden. Der zweite Parameter, der den Vergleichsoperator bestimmt, kann aus Tabelle A.6 abgelesen werden. Der dritte Parameter ist a.

[1] Es werden die in C üblichen logischen Operatoren && und || verwendet.

Im folgenden Auszug aus einer Z2NP-Bank sind einige Beispiele angegeben. Die Beschreibung der einzelnen Kriterien folgt nicht immer wörtlich der mathematischen Definition des Kriteriums, sondern will eher den physikalischen Sinn vermitteln. Deshalb ist z.B. oft das Zeichen $\leq$ durch ein $<$ ersetzt, natürlich unter Anpassung der jeweiligen Grenzen. Außerdem ist zu beachten (z.B. bei der ersten definierten Bedingung), dass ein Wert von 255 in `THETA` bedeutet, dass das jeweilige Objekt in einem Vektor nicht belegt ist.

```
 0   1 254   0   1 254   0   1 254   0   1 254  ! # of used objects (39-...
 0   1   7   0   1   7   0   1   7   0   1   7  ! THETA <= 7
 0   1  15   0   1  15   0   1  15   0   1  15  ! THETA <= 15
13  -1 100  13  -1 100  13  -1 100  13  -1 100  ! LET >= 100
15  -1   1  15  -1   1  15  -1   1  15  -1   1  ! PET > 0
15  -1   2  15  -1   2  15  -1   2  15  -1   2  ! PET > 1
15  -1   3  15  -1   3  15  -1   3  15  -1   3  ! PET > 2
15  -1   4  15  -1   4  15  -1   4  15  -1   4  ! PET > 3
25  -1   1  25  -1   1  25  -1   1  25  -1   1  ! MET > 0
39  -1   1  39  -1   1  39  -1   1  39  -1   1  ! SEET > 0
25  -1   1   4   1  30  25  -1   1   4   1  30  ! [MET > 0]&&[LEET <= 30]
25  -1   1   4   1  60  25  -1   1   4   1  60  ! [MET > 0]&&[LEET <= 60]
25  -1   1   4   1  90  25  -1   1   4   1  90  ! [MET > 0]&&[LEET <= 90]
 4  -1  30   7   1   2   4  -1  30   7   1   2  ! [LEET <= 30]&&[LENH <= 2]
 4  -1  60   7   1   2   4  -1  60   7   1   2  ! [LEET <= 60]&&[LENH <= 2]
 4  -1  90   7   1   2   4  -1  90   7   1   2  ! [LEET <= 90]&&[LENH <= 2]
 4  -1  30   7   1   3   4  -1  30   7   1   3  ! [LEET <= 30]&&[LENH <= 3]
 4  -1  60   7   1   3   4  -1  60   7   1   3  ! [LEET <= 60]&&[LENH <= 3]
 4  -1  90   7   1   3   4  -1  90   7   1   3  ! [LEET <= 90]&&[LENH <= 3]
 4  -1  30   4  -1  30   4  -1  30   4  -1  30  ! LEET >= 30
 4  -1  60   4  -1  60   4  -1  60   4  -1  60  ! LEET >= 60
 4  -1  90   4  -1  90   4  -1  90   4  -1  90  ! LEET >= 90
 4  -1  30   4  -1  30   9  -1  30   4  -1  30  ! [LEET >= 30]||[LHET >= 30]
 4  -1  60   4  -1  60   9  -1  60   4  -1  60  ! [LEET >= 60]||[LHET >= 60]
 4  -1  90   4  -1  90   9  -1  90   4  -1  90  ! [LEET >= 90]||[LHET >= 90]
 2   1   0   2   1   2   2  -1   0   2  -1   2  ! Q not 0
 2   1   0   2   1   0   2   1   0   2   1   0  ! Q negative
 2  -1   2   2  -1   2   2  -1   2   2  -1   2  ! Q positive
 1   1 128   0   1 254   1   1 128   0   1 254  ! half one of the hemisphere
 1  -1 127   0   1 254   1  -1 127   0  -1 254  ! half two of the hemisphere
27  -1   1   4   1   0  27  -1   1   4   1   0  ! [TEET > 0]&&[LEET <= 1]
32  -1   1   9   1   0  32  -1   1   9   1   0  ! [THET > 0]&&[LHET <= 1]
```

A.2.3 Z2NL (*look-up-tables*)

Die Bank Z2NL bestimmt den Inhalt der LUTs, die auf alle Daten, die von der DDB2 verarbeitet werden, im ersten Verarbeitungsschritt angewendet werden. Durch die unterschiedliche Konzeption der Subdetektoren existieren mehrere Typen von LUTs. Für jeden *trigger tower* und jeden *big tower*, deren Signale jeweils aus Bytewerten bestehen, liefert eine LUT zu jedem der 256 möglichen Eingangswerte einen frei bestimmbaren Ausgangswert.

Die den 256 möglichen Eingangswerten zugeordneten Ausgangswerte folgen in der Z2NL-Bank direkt aufeinander.

Für die Subdetektoren, die bitweise Information liefern, existieren für jeden Kanal zwei Bits x_0 und x_1, die die LUT definieren. Bei der Anwendung der LUT errechnet sich der Ausgabewert wie in Tabelle A.7.

Eingabe	Bit x_0	Bit x_1	Ausgabe
0	0	?	0
0	1	?	1
1	?	0	0
1	?	1	1

Tabelle A.7: LUT für Bitwerte

Damit kann ein Bit sowohl dauerhaft ein- oder ausgeschaltet, als auch invertiert werden.

Die Bits x_0 und x_1 sind jeweils für mehrere Kanäle zusammen in Integer-Parameter gepackt. Die Aufteilung der Kanäle auf die Integer-Parameter ist dieselbe wie bei den entsprechenden Elementen des Eventvektors (`BIGRAY`, `CJCS`, `MUON`, `SPAMAP`).

Die Integer-Parameter, die sich auf die gleichen Kanäle beziehen (und einmal x_0 und einmal x_1 enthalten), folgen in der Z2NL-Bank direkt aufeinander.

Bei den grob gerasterten Informationen des SpaCals sind in jedem Kanal vier Werte möglich, da drei Schwellen überschritten werden können. Die LUT ist deshalb analog zu denen der *big towers* und *trigger towers* konstruiert.

Die Reihenfolge der LUTs für die verschiedenen Subdetektoren sowie ein Überblick über den Gesamtumfang der Z2NL-Bank ist aus Tabelle A.8 ersichtlich.

Subdetektor	Kanäle $\div \frac{\text{Kanäle}}{\text{Parameter}} \times$ LUT-Umfang	Anzahl [Byte]
em. *trigger towers*	688×256	176128
hadr. *trigger towers*	688×256	176128
em. *big towers*	256×256	65536
hadr. *trigger towers*	256×256	65536
big rays	$256 \div 32 \times 2$	16
CJC	$445 \div 4 \times 2$	90
Myonkammern	$64 \div 32 \times 2$	4
Spacal (fein)	400×2	800
Spacal (grob)	25×4	100
		484338

Tabelle A.8: Aufbau der Z2NL-Bank

Parameter für nur von der Nummerierung her vorhandene *trigger* und *big towers* müssen zwar in der Z2NL-Bank angegeben werden, um die Position nachfolgender Parameter nicht zu verschieben, ihr Inhalt hat aber keine weitere Bedeutung.

Angesichts der Größe der Z2NL-Bank ist es sinnvoll, sie von einem Programm generieren zu lassen. Derzeit stehen zwei Programme zum Erstellen einer solchen Bank zur Verfügung. Das erste erstellt eine „dummy“ Bank, deren LUTs den jeweiligen Eingabewert auf denselben Ausgabewert abbilden, das zweite eine Bank, in der bei den Subdetektoren, deren Kanäle einen Bytewert zurückliefern, eine Abbildung der Energie auf die Transversalenergie vorgenommen wird.

Letzteres wird im Beispiel wieder aufgegriffen. Man erkennt, wie beim *trigger tower* mit dem Index 0, der im vorderen Bereich des Detektors liegt, die Transversalenergie wesentlich schwächer zunimmt, als dies die gesamte Energie tun würde. Auslassungen sind durch `! ...` gekennzeichnet.

```
Z2NL
*  LAr TT em
!   0
  0   0   0   0   0   0   0   0   0   0   0   0   0   1   1   1
  1   1   1   1   1   1   1   1   1   2   2   2   2   2   2   2
  2   2   2   2   2   2   3   3   3   3   3   3   3   3   3   3
  3   3   4   4   4   4   4   4   4   4   4   4   4   4   4   5
  5   5   5   5   5   5   5   5   5   5   5   6   6   6   6   6
  6   6   6   6   6   6   6   6   7   7   7   7   7   7   7   7
  7   7   7   7   8   8   8   8   8   8   8   8   8   8   8   8
  8   9   9   9   9   9   9   9   9   9   9   9   9  10  10  10
 10  10  10  10  10  10  10  10  10  11  11  11  11  11  11  11
 11  11  11  11  11  11  12  12  12  12  12  12  12  12  12  12
 12  12  13  13  13  13  13  13  13  13  13  13  13  13  13  14
 14  14  14  14  14  14  14  14  14  14  14  15  15  15  15  15
 15  15  15  15  15  15  15  15  16  16  16  16  16  16  16  16
 16  16  16  16  17  17  17  17  17  17  17  17  17  17  17  17
 17  18  18  18  18  18  18  18  18  18  18  18  18  19  19  19
 19  19  19  19  19  19  19  19  19  20  20  20  20  20  20  20
!   1
! ...
*  LAr TT hd
! ...
*  LAr BT em
! ...
*  LAr BT hd
! ...
* MWPC
    0 -1 ! 0
    0 -1 ! 1
! ...
    0 -1 ! 7
* CJC
    0 -1 ! 0
```

```
    0 -1 ! 1
! ...
    0 -1 ! 44
* muons
     0 -1 ! 0
     0 -1 ! 1
* fine spacal
     0  1 ! 0
     0  1 ! 1
! ...
     0  1 ! 399
* coarse spacal
     0  1  2  3 ! 0
     0  1  2  3 ! 1
! ...
     0  1  2  3 ! 24
```

Wenn keine Z2NL-Bank angegeben wird, werden die Voreinstellungen verwendet. Diese sind im Wesentlichen äquivalent zu der „dummy" Bank, korrigieren aber ein „hängendes" Bit in der CJC und einige rauschende *trigger towers*, indem bei diesen alle Eingangswerte kleiner als zehn auf null gesetzt werden.

A.2.4 Kontrolle der Parameter

Im Gegensatz zur bisherigen DDB-Simulation, bei der jeder möglichen Eingabegröße eine eindeutige Berechnung zugeordnet war, ist dies nach der Einführung der DDB2-Simulation nicht mehr gewährleistet. Im Gegenteil: Ein Element des Eventvektors wie z.B. `UTHETA0` ist definiert als Winkel θ des nach der Sortierung `U` an erster Stelle stehenden Objekts. Damit ist eine Abhängigkeit zu der Definition der Sortierung `U` impliziert, ebenso aber zu allen anderen Parametern, angefangen bei den LUTs, bis hin zu den Eigenschaften des Cluster-Algorithmus. Alle diese Abhängigkeiten dadurch zu verdeutlichen, dass der Name `UTHETA0` durch Indizes oder Ähnliches erweitert wird, scheitert schon an der Längenbegrenzung der Namen durch einige Programme.

Deshalb wird ein anderer Weg gewählt: Im Eventvektor werden zwei neue Elemente eingeführt, die die gesamten Einstellungen widerspiegeln, mit denen der Eventvektor erstellt wurde: `MD5VERS` und `MD5CHKSM`. Beide sind jeweils ein Feld mit vier 32-Bit Werten. In `MD5VERS` ist die Version der Simulation (bzw. deren Compile-Zeit) kodiert, in `MD5CHKSM` eine Prüfsumme über alle Parameter in den Bänken Z2NP und Z2NL. MD5[2] ist ein Algorithmus um digitale Signaturen zu erstellen [48, 49]. Wenn zwei Eventvektoren dieselben Signaturen `MD5VERS` und `MD5CHKSM` besitzen, ist davon auszugehen, dass sie mit derselben Version der Simulation und demselben Parameterset erstellt wurden und damit vergleichbar sind.

[2] RSA Data Security, Inc. MD5 Message Digest Algorithm

Literaturverzeichnis

[1] H1 Collaboration. The H1 detector at HERA. *Nuclear Instruments and Methods in Physics Research A*, 386:310–347, 1996.

[2] H1 Collaboration. The tracking, calorimeter and muon detectors of the H1 experiment at HERA. *Nuclear Instruments and Methods in Physics Research A*, 386:348–396, 1996.

[3] Hanspeter Beck. Principles and operation of the z-vertex-trigger. H1–Note H1–IN–479, DESY, Mai 1996.

[4] H1 Collaboration. Technical proposal to build a special spectrometer covering very small momentum transfers. Technical proposal, DESY, 1996.

[5] H. Bethe and W. Heitler. On the stopping of fast particles and on the creation of positive electrons. *Proc. Roy. Soc.*, A146:83, 1934.

[6] J. C. Bizot et al. Status of simulation for a topological level 2 trigger. H1–Note H1–IN–212, DESY, Februar 1992.

[7] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943.

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

[9] D. E. Rumelhart and J. L. McClelland. *Parallel Distributed Processing: Foundations*, volume 1. MIT Press Cambridge, 1986. 318–362.

[10] E. Barnard and L. F. A. Wessels. Avoiding false local minima by proper initialisation of connections. *IEEE Transactions on Neural Networks*, 3(6):899–905, 1992.

[11] J. Fent et al. A neural network second level trigger for the H1 experiment at HERA. H1–Note H1–IN–457, DESY, Oktober 1995.

[12] J. Koehne et al. Realization of a second level neural network trigger for the H1 experiment at HERA. H1–Note H1–IN–509, DESY, Januar 1997.

[13] Max-Planck-Institut für Physik, Werner-Heisenberg-Institut. *Netsim User Manual*, November 1997.

[14] Tobias Zobel. *Mixer Documentation*. Max-Planck-Institut für Physik, Werner-Heisenberg-Institut, März 1995.

[15] Dominik Westner. *Improving the Aspirin-Training Software*. Max-Planck-Institut für Physik, Werner-Heisenberg-Institut.

[16] Dominik Westner. *Tester Documentation*. Max-Planck-Institut für Physik Werner-Heisenberg-Institut, Juni 1995.

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

[18] Christian Kiesling et al. Design specifications for the DDB2 (revision 3). Technical report, Max-Planck-Institut für Physik, Werner-Heisenberg-Institut, November 1998.

[19] Enrico Concetti. Untersuchungen zu neuronalen Triggern für J/ψ-Erzeugung in Elektron-Proton-Stößen bei HERA. Diplomarbeit, Ludwig-Maximilians-Universität München, April 1998.

[20] T. Carli et al. Performance of the LAr trigger in 1994. H1–Note H1–IN–445, DESY, Juli 1995.

[21] Burkard Reisert. Einsatz neuer Vorverstärkerelektronik im Flüssig-Argon-Kalorimeter des H1-Detektors am HERA-Speichering. Diplomarbeit, Ludwig-Maximilians-Universität München, August 1997.

[22] R. Riedlberger. The H1 trigger with emphasis on tracking triggers. H1–Note H1–IN–419, DESY, Januar 1995.

[23] Heiko Itterbeck. *Techniques and Physics of the Central m-Trigger System of the H1-Detector at HERA*. Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, September 1997.

[24] Stephan Spielmann. *L'électronique rapide de déclenchement du calorimètre SpaCal plomb/fibre-scintillante de l'expérience H1 à HERA: Mise en œuvre, résultats en faisceaux test du CERN et premiers résultats à HERA*. Dissertation, École Polytechnique, Palaiseau, Juli 1996.

[25] V. Boundry. The inclusive electron trigger for SPACAL: Design and CERN-test results. H1–Note H1–IN–430, DESY, März 1995.

[26] Jürgen Möck. *Untersuchungen diffraktiver J/Ψ-Ereignisse im H1-Experiment bei HERA und Entwicklung neuronaler Trigger.* Dissertation, Technische Universität München, September 1997.

[27] Markus Eberl. Untersuchungen zum Triggern von ϕ-Mesonen mit neuronalen Netzen bei HERA. Diplomarbeit, Ludwig-Maximilians-Universität München, März 1999.

[28] T. Carli et al. Proposal to upgrade the LAr calorimeter trigger/ the jet trigger. H1–Note H1–IN–560, DESY, Januar 1999.

[29] S. A. Baird et al. A fast track trigger with high resolution for H1. H1–Note H1–IN–573, DESY, Juni 1999.

[30] S. A. Baird et al. Addendum to the proposal 'A fast track trigger with high resolution for H1'. H1–Note H1–IN–576, DESY, September 1999.

[31] F. Jacquet and A. Blondel. Report from the study group on detectors for charged current events. In U. Amaldi, editor, *Proceedings of the study of an ep facility for Europe*, 79/48, page 391. DESY, 1979.

[32] J. J. Sakurai. Vector-meson dominance and high-energy electron-proton inelastic scattering. *Phys. Rev. Lett.*, 22:981–984, 1969.

[33] P. D. B. Collins. *An Introduction to Regge-Theory and High-Energy-Physics.* Cambridge University Press, Cambridge, England, 1977.

[34] S. J. Brodsky, L. Frankfurt, J. F. Gunion, A. H Mueller, and M. Strikmann. Diffractive leptoproduction of vector mesons in QCD. *Phys. Rev. D*, 50:3134–3144, 1994.

[35] M. G. Ryskin. Diffractive J/Ψ electroproduction in LLA QCD. *Z. Phys. C*, 57:89–92, 1993.

[36] C. Caso et al. *The European Physical Journal C*, volume 3. 1998.

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

[38] T. Wolff et al. A drift chamber track finder for the first level trigger of the H1 experiment. *Nuclear Instruments and Methods in Physics Research A*, 323:537, 1992.

[39] S. Eichenberger et al. A fast pipelined trigger for the H1 experiment based on multiwire proportional chamber signals. *Nuclear Instruments and Methods in Physics Research A*, 323:532, 1992.

[40] T. Ahmed et al. A pipelined first level trigger for the H1 forward muon spectrometer. *Nuclear Instruments and Methods Physics Research A*, 364:456–472, 1995.

[41] Daniela Ellgaß. Untersuchungen zum Triggern photoproduzierter ϕ-Mesonen in ep-Kollisionen bei HERA mit Hilfe neuronaler Algorithmen. Zulassungsarbeit, Ludwig-Maximilians-Universität München, September 1999.

[42] Andreas Gruber. *The Event Vector.* Max-Planck-Institut für Physik, Werner-Heisenberg-Institut. `http://wwwh1.mppmu.mpg.de/projects/neuro/-doc/eventvector/eventvector.ps`.

[43] Steffen Udluft. *Photoproduktion von ϕ-Mesonen am H1-Experiment bei HERA unter Berücksichtigung des Neuronalen Triggers.* Dissertation, Ludwig-Maximilians-Universität München, in Vorbereitung.

[44] A. B. Kaidalov. Diffractive interactions of hadrons at high energy. *Phys. Rep.*, 101:157–226, 1979.

[45] ZEUS Collaboration. Measurement of diffractive photoproduction of vector mesons at large momentum transfer at HERA. hep-ex/9910038.

[46] HBOOK – statistical analysis and histogramming. CERN program library entry Y250, CERN, Geneva, 1993.

[47] V. Blobel. The BOS-system, dynamic memory management. Internal Report R1–88–01, DESY, Januar 1988.

[48] Ronald L. Rivest. The MD5 message-digest algorithm, RFC 1321. 1992. `http://info.internet.isi.edu:80/in-notes/rfc/files/rfc1321.txt`.

[49] Theodore C. Belding. Numerical replication of computer simulations: Some pitfalls and how to avoid them. 2000. arXiv:nlin: AO/0001057.

Danksagung

Diese Arbeit wurde in der H1-Gruppe am Max-Planck-Institut für Physik, München durchgeführt.

Herrn Prof. Dr. Gerd Buschhorn danke ich für seine gute und engagierte Betreuung bei diesem interessanten und vielseitigen Thema.

Besonderer Dank gilt Herrn Prof. Dr. Christian Kiesling für viele motivierende Anregungen und interessante Diskussionen rund um die DDB2 und die Physik der Vektormesonen.

Mein Dank gilt auch Herrn Prof. Dr. Andrzej Buras und Herrn Dr. Leo Stodolsky, die mich auf der Suche nach einer interessanten Diplomarbeit mit Erfolg unterstützt haben.

Bei Herrn Steffen Udluft möchte ich mich für die gute Zusammenarbeit bei der Analyse der ϕ-Daten und für viele gute Ratschläge bedanken.

Ich danke den Elektronikern des MPI München für die gute Zusammenarbeit bei der Anpassung der verschiedenen Algorithmen an die Möglichkeiten der Hardware.

Bei allen Mitgliedern der Münchner und Hamburger MPI H1-Gruppe bedanke ich mich für die hilfreiche Unterstützung bei der Lösung des einen oder anderen Problems und die angenehme Arbeitsatmosphäre. Beides hat wesentlich zum Gelingen dieser Arbeit beigetragen.

Meinen Eltern danke ich für ihre Unterstützung.