CDF Grid-Computing und das $X(3872) \to J/\psi\,\pi^+\pi^-$ mit $J/\psi \to e^+e^-$

Zur Erlangung des akademischen Grades eines
DOKTORS DER NATURWISSENSCHAFTEN
von der Fakultät für Physik der
Universität Karlsruhe (TH)

genehmigte

DISSERTATION

von

Ulrich Kerzel
aus Essen

Tag der mündlichen Prüfung: 11. November 2005
Referent: Prof. Dr. M. Feindt, Institut für Experimentelle Kernphysik
Korreferent: Prof. Dr. G. Quast, Institut für Experimentelle Kernphysik

Deutsche Zusammenfassung

Einführung

Das Hauptziel physikalischer Forschung ist es, ein konsistentes Bild der Natur zu erlangen, das es uns ermöglicht, die in Experimenten beobachteten Phänomene zu verstehen. Die Teilchenphysik beschäftigt sich mit der Analyse der Elementarteilchen und deren Wechselwirkung. Mathematisch läßt sich der derzeitige Kenntnisstand durch das sog. "Standard-Modell" beschreiben. In dieser Theorie werden die Elementarteilchen durch quantenmechanische Felder beschrieben, die durch die elektromagnetische, schwache und starke Kraft miteinander wechselwirken. Diese Wechselwirkung geschieht unter Austausch von Eichbosonen wie dem Photon (als Träger der elektromagnetischen Kraft), den $W^{\pm}$ und Z^0 Bosonen (schwache Wechselwirkung) und den Gluonen als Austauschteilchen der starken Kraft. Die Gültigkeit dieser Herangehensweise wird experimentell dadurch getestet, dass Teilchen wie z.B. Elektronen oder Protonen (und deren Anti-Teilchen) bei hoher Schwerpunktsenergie zur Kollision gebracht werden. Die dabei entstehenden Reaktionen werden mit fortschrittlichen Detektoren aufgezeichnet und analysiert. Im Vergleich dieser Analysen und der Vorhersagen aus der Theorie zeigt sich, daß das Standard-Modell eine sehr genaue Beschreibung der Natur ermöglicht. Präzisionsmessungen, die von den vier Experimenten (Aleph, Delphi, Opal, L3) am Large Electron Posititron Collider (LEP) des Teilchenphysiklabors CERN bei Genf gemacht wurden, konnten viele der in der Theorie enthaltenen Parameter mit enormer Präzision vermessen. Dennoch sind noch viele Fragen offen, z.B. ob das Standard-Modell eine gültige Beschreibung bei noch höheren Energien liefert oder ob wir derzeit das niederenergetische Verhalten einer noch fundamentaleren Theorie parametrisieren. Weiterhin ist die Frage nach dem Ursprung der Masse der Teilchen und die Suche nach Phänomenen, die vom Standard Modell nicht erklärt werden können oder sogar seinen Aussagen widersprechen, von besonderer Bedeutung für weitere Experimente. So wurde von der Belle-Kollaboration Ende 2003 das neue Teilchen $X(3872)$ entdeckt, dessen Eigenschaften bisher von der Theorie nicht zufriedenstellend erkärt werden können.

Schwerpunkt dieser Dissertation ist daher die experimentelle Analyse dieses neuen Teilchens mit Daten, die vom CDF-Experiment am Tevatron-Collider aufgezeichnet wurden. Moderne Teilchenphysik-Experimente stellen enorme Datenmengen im Umfang vieler Terabyte zur Verfügung. Um physikalische Analysen durchführen zu können, ist die Entwicklung fortschrittlicher Grid-Computing-Technologien eine grundlegende Voraussetzung, da nur so ausreichende Rechenkapazitäten bereitgestellt werden können und ein effizienter Zugriff auf die Daten möglich ist. Die in dieser Arbeit geleisteten Beiträge erlauben es, das deutsche Grid-Kompetenzzentrum "GridKa" für die deutsche CDF-Beteiligung nutzbar zu machen und dort Physikanalysen durchzuführen. Die komplexen Reaktionen, die vom CDF-Experiment aufgezeichnet werden, stellen besonders hohe Anforderungen an die verwendeten Analysewerkzeuge. Ein weiterer Schwerpunkt dieser Arbeit liegt daher in der Entwicklung einer hoch effizienten und genauen Methode zur Identifikation von Elektronen im Detektor. Die Verbindung

dieser beiden Komponenten ermöglicht es erstmals, den Zerfall des $X(3872)$ unter Verwendung von Elektronen in Hadronkollisionen zu beobachten.

Das CDF-Experiment befindet sich am Tevatron-Collider am Fermi National Laboratory in der Nähe von Chicago, der Protonen und Anti-Protonen bei der derzeit weltweit höchsten Schwerpunktsenergie von $\sqrt{s} = 1.96$ TeV zur Kollision bringt. Die bei dieser Reaktion entstehenden Ereignisse werden mit hoher Rate aufgezeichnet; zur Zeit stehen der internationalen Kollaboration mehr als 2000 TB prozessierte Daten für Physikanalysen zur Verfügung. Diese enormen Datenmengen sind nicht mehr manuell handhabbar und erfordern die Entwicklung von Grid-Technologien, die es ermöglichen, diese Daten in den weltweit entstehenden Rechenzentren zu analysieren. Dazu wird im Rahmen des FermiGrid Projekts die Software "SAM" (Sequential Access via Metadata) entwickelt und eingesetzt, um die große Datenmengen effizient zu den Rechenzentren zu transferieren und dort den Benutzern für Analysen zur Verfügung zu stellen. Das deutsche Grid-Kompetenzzentrum "GridKa" befindet sich am Forschungszentrum Karlsruhe und stellt große Rechenkapazitäten für Mitglieder der Tevatron-Experimente CDF und DØ, sowie aller LHC-Kollaborationen, BaBar und Compass bereit. Die in Kapitel 3 diskutierten Entwicklungen ermöglichen es der deutschen CDF-Beteiligung, diese Kapazitäten effizient zu nutzen. Unter Ausnutzung der Synergie-Effekte, die sich durch den gemeinsamen Betrieb des Rechenzentrums ergeben, konnten so bisher ≈ 500 TB Daten analysiert werden. Die dabei gewonnenen Erkenntnisse ermöglichen sowohl den Aufbau weiterer Rechenzentren z.B. in Italien, Asien und Amerika, als auch der zentralen Produktions- und Analysecluster am Fermilab selbst. Somit nimmt das GridKa eine Vorreiterrolle für CDF ein und kann als Prototyp für ähnliche Zentren gesehen werden.

Ende 2003 wurde von der Belle-Kollaboration [2] der neue Zustand $X(3872)$ im Zerfallskanal $B^{\pm} \to K^{\pm}X(3872) \to K^{\pm}(\pi^+\pi^- J/\psi)$ entdeckt. Kurz darauf wurde diese Entdeckung von der CDF-Kollaboration [1], sowie von der BaBar- [3] und DØ- Kollaboration [4] bestätigt. Abbildung 1 zeigt die Verteilung der invarianten $J/\psi\pi^+\pi^-$ Masse im Kanal $J/\psi \to \mu^+\mu^-$ aus der CDF-Veröffentlichung. Neben der bekannten Resonanz $\psi(2S)$, die in den gleichen Endzustand zerfällt, ist ein deutlicher Überschuß bei $m = 3872$ MeV/c^2 zu sehen. Die experimentell ermittelte Breite des Zustands ist mit der Auflösung der jeweiligen Detektoren verträglich. Eine detaillierte Messung der Masse des $\pi^+\pi^-$ Systems durch die CDF Kollaboration [5] zeigt eine deutliche Anreicherung bei hohen Werten nahe der kinematischen Grenze an, was auf die Bildung der Zwischenresonanz ρ^0 zurückzuführen sein könnte.

Aufgrund der extrem komplexen hadronischen Umgebung und des damit verbundenen enormen Untergrunds in Proton-Antiproton Kollisionen konnte der Zerfall des $X(3872)$ bisher bei der CDF und DØ Kollaboration nur im Kanal $J/\psi \to \mu^+\mu^-$

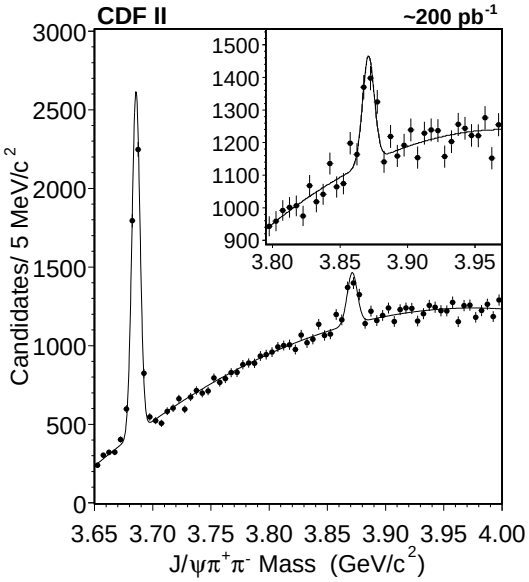


Abbildung 1: Verteilung der invarianten $J/\psi\pi^+\pi^-$ Masse im Kanal $J/\psi \to \mu^+\mu^-$, aus der ersten CDF-Veröffentlichung [1], die die Existenz des neuen Teilchens $X(3872)$. bestätigt.

beobachtet werden. Dies führte zur Entwicklung einer hocheffizienten Methode zur Identifikation von Elektronen, da der Zerfall $J/\psi \to e^+e^-$ mit gleicher Rate auftritt wie der Zerfall $J/\psi \to \mu^+\mu^-$. Wie in Kapitel 4 dargestellt, werden mittels fortschrittlicher neuronaler Netzwerke Informationen aus verschiedenen Bereichen des Detektors (wie z.B. Energiemessungen aus den Kalorimetern, Flugzeitmessung, Spurparameter) kombiniert, so daß es nun möglich ist, Elektronen mit sehr großer Reinheit und Effizienz vom mehr als 10 mal größeren Untergrund zu trennen. Der linke Graph in Abbildung 2 zeigt die so erreichte Verbesserung im Vergleich zur bisher verfügbaren schnittbasierten Analyse. Verlangt man, daß jede rekonstruierte Spur mindestens 2 GeV/c Transversalimpuls[1] hat, so kann mittels der hier vorgestellten Methode die Effizienz von 63% auf 96% bei gleicher oder besserer Reinheit gesteigert werden. Kapitel 5.1 erläutert, wie mittels dieser Methode der Zerfall $J/\psi \to e^+e^-$ fast untergrundfrei beobachtet werden kann. Dabei wird der verbleibende Hauptuntergrund, die Konversion eines Photons in ein Elektron-Positron-Paar im Innern des CDF-Detektors, durch ein weiteres neuronales Netzwerk unterdrückt. Nach einer Korrektur, die die Energieverluste aufgrund von Bremsstrahlung ausgleicht, ist es somit erstmals möglich, den Zerfall $X(3872) \to J/\psi\pi^+\pi^-$ mit $J/\psi \to e^+e^-$ an einem Hadron-Collider zu

[1]Dies entspricht der Schwelle des $J/\psi \to e^+e^-$ Triggers bei der Datennahme

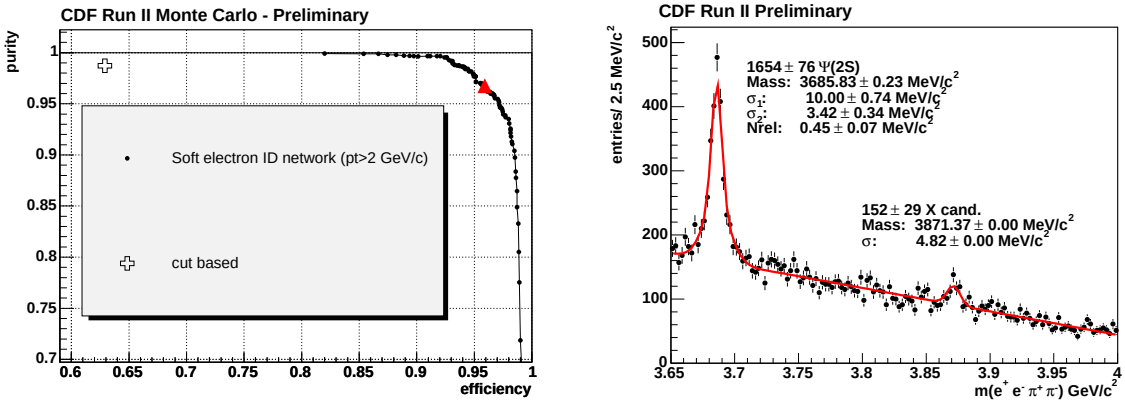


Abbildung 2: Der linke Graph veranschaulicht die Verbesserung der Elektronen-Identifikation, die mit der hier vorgestellten Methode im Vergleich zu den Standardschnitten (dargestellt durch das Kreuz) erreicht wird. Der Dreieck (rot) repräsentiert den besten Arbeitspunkt, bei dem der Abstand vom idealen Arbeitspunkt, der durch 100% Reinheit bei gleichzeitig 100% Effizienz gekennzeichnet ist, minimal ist. Das Bild rechts zeigt die Verteilung der invarianten $J/\psi\pi^+\pi^-$ Masse im Kanal $J/\psi \to e^+e^-$. Die Resonanzen des $\psi(2S)$ und des $X(3872)$ sind deutlich zu erkennen. Das Spektrum wird durch ein Polynom zweiten Grades (für den Untergrund), einer doppelten Gauß-Funktion für das $\psi(2S)$ und einer Gauß-Funktion für das $X(3872)$ beschrieben. Bei der Anpassung der Parameter wurden Masse und Breite der Gauß-Funktion für das $X(3872)$ aus dem Kanal $J/\psi \to \mu^+\mu^-$ übernommen und festgehalten.

beobachten, wie der rechte Teil der Abbildung 2 zeigt. Die hier vorgestellte Methode zur effizienten Identifikation von Elektronen ist von großer Bedeutung bei allen B-Physik-Analysen. In Kapitel 5.2 wird das Potential der Methode illustriert, indem sie zur Trennung von $b/\bar{b}$ Quarks in semi-leptonischen B-Meson Zerfällen eingesetzt wird. Dazu wurde sie in die Analyseumgebung der Messung von Δm_s, der Teilchen-Antiteilchen-Oszillationsfrequenz von B_s Mesonen, integriert. Im Vergleich zur bisher verwendeten Methode, die auf einem Likelihoodverfahren beruht, wird eine signifikante Steigerung der Identifikationskraft erreicht.

Aus theoretischer Sicht bieten sich mehrere Modelle zur Erklärungen der Natur des $X(3872)$ an: Ähnlich wie das $\psi(2S)$, das in den gleichen Endzustand zerfällt, könnte es sich beim $X(3872)$ um ein angeregtes Charmonium, also ein gebundenes $c\bar{c}$-Quark-Paar handeln. Genaue Rechnungen zeigen jedoch (siehe z. B. [6]), daß die vorhergesagten Massen der noch nicht beobachteten Zustände um etwa 100 MeV/c^2 von der beobachteten Masse abweichen, so dass Vorhersage und Beobachtung auf-

grund der geringen gemessenen Breite des Zustands nicht miteinander verträglich sind. Darüber hinaus müßten für viele der vorhergesagten Charmoniumzustände weitere Zerfallskanäle existieren, die in experimentellen Suchen bisher nicht beobachtet werden konnten. Die Nähe zur $D^0\bar{D}^{*0}$-Massenschwelle bei 3871.3 MeV/c^2 eröffnet die Möglichkeit, dass es sich beim $X(3872)$ um ein exotisches Teilchen handeln könnte: Bereits 1977 wurden molekülartige $D^*\bar{D}^*$- und $D\bar{D}^*$-Zustände von DeRújula, Georgi and Glashow [7] vorgeschlagen, die entweder durch Dissoziation (d.h. das D^* und das $\bar{D}^*$, aus denen das Molekül aufgebaut ist, fallen auseinander) oder durch Emission von Pionen oder ρ^0-Mesonen zerfallen. Törnqvist [8] stellt die These auf, dass das $X(3872)$ ein molekülartiger $D\bar{D}^*$ Zustand ist, der, ähnlich wie das Deuteron, durch Pionaustausch gebunden ist. Seine Rechnungen zeigen, dass ein bindendes Potential für einen solchen Zustand nur dann möglich ist, wenn das $X(3872)$ die Quantenzahlen $J^{PC} = 1^{++}$ oder 0^{-+} aufweist. Aufgrund der Isospinverletzung ist die Bildung einer ρ^0-Zwischenresonanz möglich. Die Rechnung von Swanson [9] zeigt, dass eine Beimischung der ω-Resonanz möglich ist, wenn das $X(3872)$ die Quantenzahlen $J^{PC} = 1^{++}$ besitzt.

Die Bestimmung der Quantenzahlen J^{PC} ist daher eine wichtige Voraussetzung zur Interpretation des $X(3872)$. In Zusammenarbeit mit einer Diplomarbeit [10] werden Spin, Parität und Ladungsparität mittels der Methode der Helizitätsamplituden ermittelt. Wie in Kapitel 6.3 erläutert, sagt diese Methode die Verteilung und Korrelation kinematischer Größen wie den Winkeln zwischen den beteiligten Zerfallsprodukten und der invarianten $\pi^+\pi^-$ Masse in Abhängigkeit von den Eigenschaften der involvierten Teilchen und der eventuellen Bildung einer Zwischenresonanz der Pionen vorher. Ein Vergleich dieser Vorhersagen mit den experimentell gewonnenen Messungen dieser Größen ermöglicht Aussagen über die Quantenzahlen J^{PC} des $X(3872)$. Der Zerfall des bekannten Teilchens $\psi(2S)$ in den gleichen Endzustand erlaubt einen präzisen Test der Methode, da hier sowohl die Quantenzahlen $J^{PC} = 1^{--}$ und das Verhalten des $\pi^+\pi^-$-Systems aus anderen Messungen bekannt sind. Um die Helizitätsanalyse durchführen zu können, werden zunächst sowohl das $\psi(2S)$ und das $X(3872)$ im exklusiven Endzustand $J/\psi\,\pi^+\pi^-$ rekonstruiert (siehe Kapitel 6.1). Insgesamt 21 mögliche Zustände mit Spin $J = 0, 1, 2, 3$ werden analysiert, bei denen das $\pi^+\pi^-$ entweder keine Zwischenresonanz (s-Wellenzustand) oder ein ρ^0 oder f_2 bildet. Angewendet auf das $\psi(2S)$ wird die korrekte Zuweisung $J^{PC} = 1^{--}$, bei der sich die Pionen in einem s-Wellenzustand befinden, mit großer statistischer Signifikanz ermittelt; nur ein speziell für das $\psi(2s)$ entwickeltes Modell [11], das zusätzlich einen geringen d-Wellenanteil berücksichtigt, liefert eine noch bessere Übereinstimmung zwischen Vorhersage und Messung. Für das $X(3872)$ beschreiben die Vorhersagen für die Quantenzahlen $J^{PC} = 1^{++}_{\rho}$ die gemessenen Verteilungen am besten, wobei die Pionen eine ρ^0 Zwischenresonanz bilden. Der Zerfall verläuft also gemäß $X(3872) \rightarrow J/\psi\rho^0 \rightarrow J/\psi\pi^+\pi^-$. Die nächstbeste

Zuweisung ($J^{PC} = 2^{++}_{\rho}$) ist $\approx 1.4\sigma$ schlechter, alle anderen getesteten Hypothesen können mit hoher statistischer Signifikanz von $\gtrapprox 5\sigma$ verworfen werden. Um mögliche Unsicherheiten bei der Modellierung des s-Wellenzustands zu vermeiden, wurde die Analyse zusätzlich nur unter Verwendung der Winkel durchgeführt. In diesem Fall lässt sich ebenfalls die Zuweisung Spin $J = 0$ ausschließen, während die Zustände mit Spin 1 oder Spin 2 des s-Wellenzustands zusätzlich zu obigem Ergebnis verbleiben. Die mittels der Helizitätsamplituden bestimmten Quantenzahlen und das Verhalten des Pionsystems stimmt somit mit den Vorhersagen aus der Molekülinterpretation von Törnqvist und Swanson überein.

CDF Grid computing and the decay $X(3872) \to J/\psi\,\pi^+\pi^-$ with $J/\psi \to e^+e^-$

PhD Thesis
Faculty for Physics
University of Karlsruhe (TH)

by

Ulrich Kerzel

Supervisor: Prof. Dr. M. Feindt, Institut für Experimentelle Kernphysik,
University of Karlsruhe (TH)

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

The main aim of physics research is to obtain a consistent description of nature leading to a detailed understanding of the phenomena observed in experiments. The field of particle physics focuses on the discovery and understanding of the fundamental particles and the forces by which they interact with each other. Using methods from group theory, the present knowledge can be mathematically described by the so-called "Standard Model", which interprets the fundamental particles (quarks and leptons) as quantum-mechanical fields interacting via the electromagnetic, weak and strong force. These interactions are mediated via gauge particles such as the photon (for the electromagnetic force), $W^{\pm}$ and Z^0 (for the weak force) and gluons (for the strong force). Gravitation is not yet included in this description as it presently cannot be formulated in a way to be incorporated in the Standard Model. However, the gravitational force is negligibly small on microscopic levels. The validity of this mathematical approach is tested experimentally by accelerating particles such as electrons and protons, as well as their antiparticles, to high energies and observing the reactions as these particles collide using sophisticated detectors. Due to the high energy of the particles involved, these detectors need to be as big as a small house to allow for precision measurements. Comparing the predictions from theory with the analysed reactions observed in these collisions, the Standard Model has been established as a well-founded theory. Precision measurements from the four experiments (Aleph, Delphi, Opal, L3) the Large Electron Positron collider (LEP), operated at CERN during the years 1989 - 2000, allow the determination of the Standard Model parameters with enormous accuracy. However, many questions remain unanswered, e.g. : How do particles obtain their measured masses? Can the fundamental forces be described by one unified force at higher energies? Is the description provided by the Standard Model valid even at the highest energy or do we observe the low-energy behaviour of a more fundamental

theory? Are there phenomena which cannot be explained by the Standard Model or even contradict its predictions? Multiple experiments operating at higher energies are needed to find answers to these (and many other) open questions. Furthermore, the analysis of experimental data occasionally provides surprises: The new particle $X(3872)$ was discovered by the Belle Collaboration at the end of the year 2003. So far, its properties do not fit well into our present understanding of bound quark states.

The work of this thesis is concerned with the analysis of this new particle using data recorded by the CDF[1] detector at the Tevatron collider. The collider is located at the Fermi National Laboratory in the vicinity of Chicago (USA). Protons and anti-protons are accelerated to an centre-of-mass energy of $\sqrt{s} = 1.96$ TeV which is the highest energy available until the start of the LHC at CERN. The CDF collaboration consists of about 800 physicists from 53 institutes in 11 countries. Data-taking restarted after a five-year shutdown with a major accelerator and detector upgrade end of 2001.

The high event rates and the complex hadronic environment of these reactions lead to unique challenges in both managing the large amounts of data produced and the physics analyses. More than 2000 TB processed data are currently available, which corresponds to approximately 3 million CDs. Sophisticated Grid computing methods are needed to analyse these enormous amounts of data. Multiple large computing facilities are currently being built around the world. The German Grid computing centre "GridKa" is located at the Forschungszentrum Karlsruhe and offers large amounts of computing power to members of the Tevatron experiments CDF and DØ, as well as BaBar, Compass and the LHC experiments. Through the work of this thesis it has been possible for the German CDF group to efficiently use this facility and analyse $\approx$ 500 TB up to now. The developed methods discussed in chapter 3 are of vital importance for the operation of similar computing farms in e.g. Italy, Asia and the USA, as well as the central analysis facility (CAF) and the production farm on-site Fermilab.

The new particle $X(3872)$ was discovered by the Belle collaboration [2] at the end of the year 2003 in the decay channel $B^{\pm} \to K^{\pm}X(3872) \to K^{\pm}(\pi^+\pi^- J/\psi)$ where the J/ψ decays either to $\mu^+\mu^-$ or e^+e^-. The existence of this particle was confirmed shortly afterwards by the CDF collaboration [1]. However, due to the extremely complex hadronic environment only the decay channel $J/\psi \to \mu^+\mu^-$ was considered as muons can be cleanly identified by dedicated muon detectors. This has stimulated the development of a highly efficient method to identify electrons with high purity since the J/ψ decays into $\mu^+\mu^-$ and e^+e^- with equal rates. As discussed in chapter 4, sophisticated neural networks are deployed which optimally combine correlated input variables. These variables are derived from multiple parts of the CDF detector such as energy measurements in the calorimeters, measurements of the specific energy loss dE/dx in the central drift chamber, change of curvature of reconstructed tracks as

[1] Collider Detector at Fermilab

the particles traverse the detector, etc. The electron identification package developed has large impact on all B physics analyses, which is demonstrated in chapter 5.2 by integrating it into the CDF measurement framework of the oscillation frequency between B_s and $\bar{B}_s$ mesons. Using the neural network based approach developed in this thesis results in a significant improvement of the discriminating power between b- and $\bar{b}$-quarks compared to the likelihood based approach currently used. As shown in detail in section 6.2, this work has made it possible to observe the decay $X(3872) \to J/\psi\pi^+\pi^-$ with $J/\psi \to e^+e^-$ for the first time in the complex environment of a hadron collider.

Despite the present knowledge about the $X(3872)$ reviewed in the following section, the exact nature of the particle is still unknown. As attempts to explain the $X(3872)$ as a conventional bound quark-antiquark pair have their shortcomings, the close proximity to the $D\bar{D}^*$ mass-threshold has raised the question whether the $X(3872)$ is an exotic form of matter, a mesonic molecule. The determination of the quantum numbers spin J, parity P and charge-parity C of the $X(3872)$ are of vital importance as the theoretical calculations of these exotic models predict specific assignments for these quantum numbers. Using the method of helicity amplitudes introduced in chapter 6.3, the quantum numbers J^{PC} are measured in a joint effort of a separate diploma thesis [10] and this work. Establishing the exact nature of the $X(3872)$ may also be of help in the interpretation of the well established states $a_0(980)$ and $f_0(980)$. They are not well described by conventional models as well, but their interpretation in terms of exotic models remains inconclusive so far (see e.g. [12, 13]).

1.2 Experimental status of the $X(3872)$

Searching for new charmonium states, i.e. bound $c\bar{c}$ systems, the Belle collaboration announced the observation of a previously unknown particle [2] end of 2003. Figure 1.1 shows the spectrum of the known charmonium states. The recently discovered narrow resonance called $X(3872)$ is produced in the exclusive decay $B^\pm \to K^\pm X(3872) \to K^\pm(\pi^+\pi^- J/\psi)$. 35.7 ± 6.8 events were observed, leading to a statistical significance of 10.3σ. The mass of the new state was measured as:

$$m(X) = 3872.0 \pm 0.6\ (\text{stat})\ \pm 0.5\ (\text{syst})\ \text{MeV/c}^2$$

which is very close to the $\bar{D}D^*$ mass threshold of $m(\bar{D}D^*) = 3871.3\ \text{MeV/c}^2$. The observed width of the state is compatible with the detector resolution. It was found that the $\pi^+\pi^-$ invariant masses tend to cluster near the kinematic boundary around the ρ^0 meson mass of $\approx 770\ \text{MeV/c}^2$ indicating that the $\pi^+\pi^-$ may form an intermediate resonant state. The relative $B \to KX \to KJ/\psi\pi^+\pi^-$ branching fraction with respect to the $\psi(2S)$ which decays into the same final state has been determined to be $0.063 \pm 0.012\ (\text{stat})\ \pm 0.007\ (\text{syst})$.

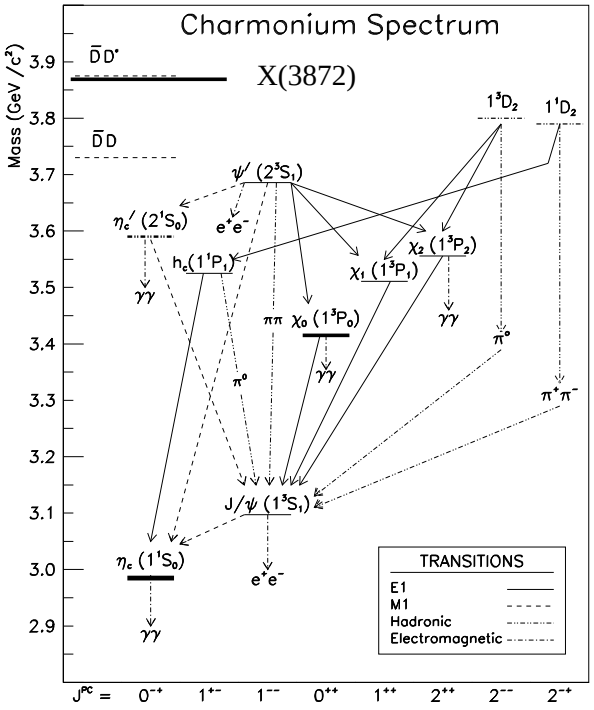


Figure 1.1: Spectrum of the charmonium ($c\bar{c}$ states) system [14]. The states are labelled with their names and characterised by their angular momentum and spin. The notation follows the convention used in spectroscopy: $n^{2s+1}L_J$. The horizontal lines illustrate the mass thresholds of the $D\bar{D}$, $\bar{D}D^*$ and the measured value of the $X(3872)$.

The observation of the $X(3872)$ was quickly confirmed by the CDF collaboration[1]. As illustrated by figure 1.2, a significant excess of 730 ± 90 candidates was found in the invariant $J/\psi\pi^+\pi^-$ mass spectrum. The mass of the particle was measured to be 3871.3 ± 0.7 (stat) ± 0.4 (syst) MeV/c^2, the reported width of $\sigma = 4.9 \pm 0.7$ MeV/c^2 is compatible with the detector resolution. No signal is observed in either isospin 2 combination $\mu^+\mu^-\pi^+\pi^+$ and $\mu^+\mu^-\pi^-\pi^-$.

DØ [4] also searched for this state in the $\Delta m = m(\mu^+\mu^-\pi^+\pi^-) - m(\mu^+\mu^-)$ invariant mass spectrum and found 522 ± 100 $X(3872)$ candidates, which corresponds to a statistical significance of 5.2 standard deviations. The mass difference is determined to be $\Delta m = 774.9 \pm 3.1$ (stat) ± 3.0 MeV/c^2, again the observed width of $\sigma = 17 \pm 3$ MeV/c^2 is compatible with the detector resolution. Exploiting the large muon coverage of the DØ detector, the $X(3872)$ is reconstructed separately in the central ($|y| < 1$) and forward direction ($|y| > 1$) where y is the rapidity[2] of the $X(3872)$ candidate.

[2]The rapidity y is defined by $y = \frac{1}{2}\ln\frac{E+p_l}{E-p_l}$ where E of the particle is the energy p_l is the

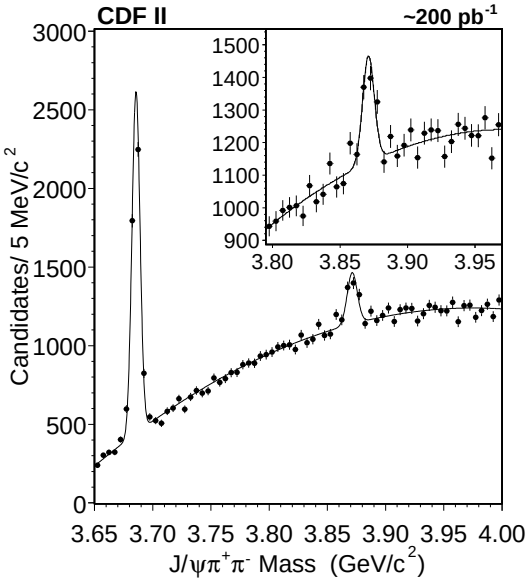


Figure 1.2: Distribution of the invariant $J/\psi\pi^+\pi^-$ mass confirming the existence of the $X(3872)$[1].

The observed behaviour is found to be very similar in both regimes.

BaBar [3] also confirms the observation of this state and measures its mass to be $m(X) = 3873.4 \pm 1.4$ MeV/c^2. The branching fraction is determined to be $\mathcal{B}(B^- \to X(3872)K^-) \cdot \mathcal{B}(X(3872) \to J/\psi\pi^+\pi^-) = (1.28 \pm 0.41) \cdot 10^{-5}$.

BES [15] used initial state radiation data to estimate the e^+e^- partial width in the channel $X(3872) \to J/\psi\pi^+\pi^-$, $J/\psi \to \ell^+\ell^-$ where the lepton ℓ represents either electrons of muons. However, no peak has been observed in the invariant mass spectrum $M(\ell^+\ell^-\pi^+\pi^-) - M(\ell^+\ell^-)$ at the expected value for the $X(3872)$. From this result an upper limit on the e^+e^- partial width and the branching ratio $X(3872) \to J/\psi\pi^+\pi^-$ has been set to $\Gamma_{e^+e^-}\mathcal{B}_{J/\psi\pi^+\pi^-} < 10$ eV at 90% C.L. for an assumed $J^{PC} = 1^{--}$ state. By comparing this result to the ψ and Υ families and considering predictions from potential models, the authors concluded that the $X(3872)$ is unlikely to be a vector state.

CLEO [16] searched for the $X(3872)$ in untagged $\gamma\gamma$ fusion events. Also using initial state radiation data, $X(3872)$ production in e^+e^- annihilation for a $J^{PC} = 1^{--}$ state was investigated. No signal was found in either case.

Many properties of the $X(3872)$ have already been determined. Comparing the distribution of the angle between the J/ψ and the $K^\pm$ momentum vectors in the

longitudinal momentum of the particle along the direction of the incident particle.

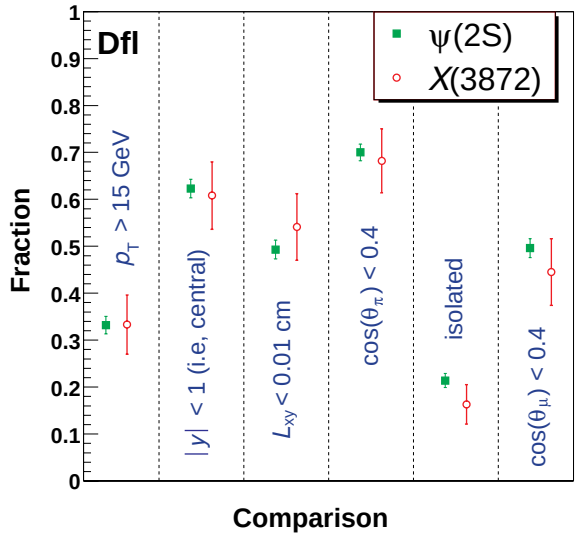


Figure 1.3: Comparison of $X(3872)$ properties to the $\psi(2S)$ by DØ [4].

$X(3872)$ rest frame to predictions from theory [17], Belle disfavours the assignment that the $X(3872)$ is the $h_c'(1^{+-})$ state [18]. Following this approach, Belle [19] also disfavours the assignments $J^{PC} = 0^{-+}$, 0^{++}, whereas the assignment $J^{PC} = 1^{++}$ is compatible with the data.

DØ [4] compares various properties of the $X(3872)$ to the $\psi(2s)$ by applying a cut on the respective quantity and comparing the obtained yield to the yield obtained with the default selection. In detail, the following properties are examined: $p_t > 15$ GeV/c, limit to the central region of the detector ($|y| < 1$), $L_{xy} < 0.01$cm where L_{xy} is defined as the distance of the primary vertex to the decay vertex scaled by M/p_t, helicity angle $\cos(\theta_{\pi,\mu}) < 0.4$ and isolated candidates. The helicity angle $\theta_{\pi,\mu}$ is obtained by boosting one of the pions (muons) in the the di-pion (di-muon) rest frame. Isolation is defined as p(X)/p(X + charged tracks) where the charged tracks are selected from a cone with $\Delta R = 0.5$ around the reconstructed $X(3872)$ direction. The $X(3872)$ behaves very similarly to the $\psi(2S)$ in all comparisons as illustrated by figure 1.3.

In order to determine the production fraction of the $X(3872)$ from B decays CDF [20] defines the following pseudo-proper lifetime $c\tau = m(X)/p_t(X) \cdot L_{xy}$ (where $L_{xy} = (\vec{x}_{decay} - \vec{x}_{prim})\vec{p}_t/|\vec{p}_t|$). Using this definition, the data is fitted simultaneously for $c\tau$ and $m(X)$ in an unbinned log-likelihood fit. The mass signal is described by a Gaussian distribution, whereas a second-degree polynomial is used to describe the background shape of the $m(\pi^+\pi^-\mu^+\mu^-)$ distribution. The signal of the pseudo-proper lifetime of the fit function is described by an exponential distribution, two positive exponential distributions are used to model the background for $c\tau > 0$, a further

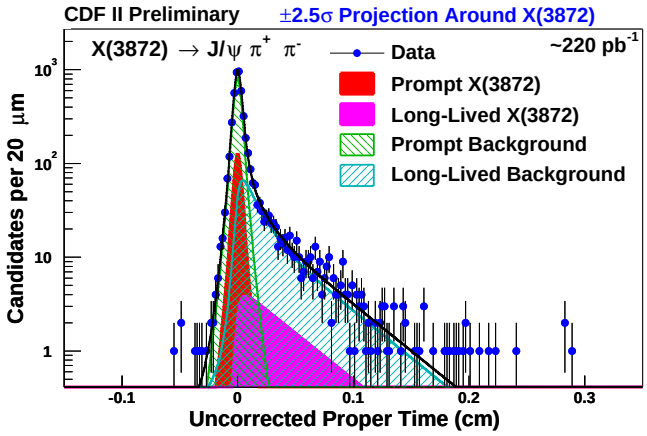


Figure 1.4: $X(3872)$ production fraction from B decays measured by CDF [20].

exponential distribution is used for the background description with $c\tau < 0$. All exponential distributions are folded with Gaussian distributions to account for the limited detector resolution. Performing the fit, a production fraction from B decays of 16.1 ± 4.9 (stat) ± 1.0 (syst) % is found for the $X(3872)$. Although this is a bit lower than the production fraction of the $\psi(2S)$ extracted in the same way determined to 28.3 ± 1.0 (stat) ± 0.7 (syst) % the two values are compatible within given uncertainties. The result of the fit for the $X(3872)$ is shown in figure 1.4 which also illustrates the enormous background which has to be taken care of in all analyses of the $X(3872)$.

The determination of the $m(\pi^+\pi^-)$ mass spectrum plays a vital role in understanding the nature of the $X(3872)$ since the shape of the distribution depends on whether the $\pi^+\pi^-$ sub-system forms an intermediate sub-resonance such as $\rho^0 \to \pi^+\pi^-$ or is in a relative s-wave state. Furthermore, it also depends on the relative angular momentum between the J/ψ and the $\pi^+\pi^-$ systems, as well as on detector effects due to acceptance, inefficiencies, etc. as discussed in detail in [21].

The large background is a major challenge in all $X(3872)$ analyses. CDF [5] uses a "slicing technique" instead of a standard method based on sideband-subtraction: The invariant $m(\pi^+\pi^-\mu^+\mu^-)$ mass-distribution is reconstructed in bins of $m(\pi^+\pi^-)$ (in addition to the default selection criteria). Figure 1.5 shows the $X(3872)$ yields obtained this way as a function of $m(\pi^+\pi^-)$. The spectrum peaks at high values of $m(\pi^+\pi^-)$ and is compatible with zero below $m(\pi^+\pi^-) \leq 0.5$ GeV/c^2. Several theoretical predictions are shown together with the extracted data points. The distribution obtained from simple phase-space consideration is clearly disfavoured by the data. If the $X(3872)$ is a charmonium state (i.e. a bound $c\bar{c}$ system), predictions from multipole expansions of the binding gluon field (e.g. [22]) can be compared to the spectrum. Within these models, only the $^3S_1$ state is compatible with the data. However, this

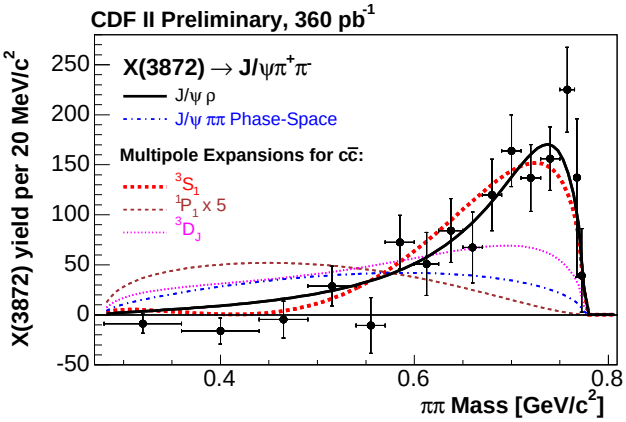


Figure 1.5: Reconstructed invariant mass spectrum of the $\pi^+\pi^-$ system in the decay $X \to J/\psi\pi^+\pi^-$ as measured by CDF[5]

state has the quantum numbers $J^{PC} = 1^{--}$ and should have been observed by BES [15], CLEO [16] and BaBar [23]. The shape of the $m(\pi^+\pi^-)$ distribution is however compatible with the assumption that the pions form an intermediate ρ^0 resonance.

Many searches for other decay channels of the $X(3872)$ have been performed. Belle [2][24] searches for decays to $\gamma\chi_{c1}$ and $\gamma\chi_{c2}$, however, no signal has been observed yet. Recently, Belle[25] reported evidence for the observation of the decay $X(3872) \to J/\psi\gamma$ with a statistical significance of 4.0 σ. 13.6 ± 4.4 events were found in this analysis. This result, if confirmed, establishes the charge conjugation parity of the X(3872) to be $C = +1$ due to the quantum numbers of the J/ψ and the photon. In the same analysis evidence for the decay $X(3872) \to \pi^+\pi^-\pi^0 J/\psi$ is reported with a significance of 4.0 σ which indicates the presence of an intermediate $\omega \to \pi^+\pi^-\pi^0$ resonance. BaBar [26] searched for charged partners $X^\pm$ in the decay $B \to X^- K$, $X^- \to J/\psi\pi^-\pi^0$. If the $X(3872)$ was part of an iso-triplet the expected decay rate of $B \to X^\pm K$ would be twice the decay rate of the neutral $X(3872)$. However, no signal was found. Following a similar approach as in the BES analysis, BaBar searches for the $X(3872)$ in events with initial state radiation [23]. No signal has been found which disfavours the assignment $J^{PC} = 1^{--}$ for the $X(3872)$.

1.3 Theoretical explanations

Many theoretical interpretations of the newly found $X(3872)$ particle are available. A rather straight-forward explanation is that the $X(3872)$ is a charmonium, i.e. a bound $c\bar{c}$ state. Indeed, the particle has been observed in searches aiming to identify the yet unobserved charmonium states. Many predictions of the charmonium spectrum have

been obtained using quark potential models which can be described by [27]:

$$V(r) = \frac{\kappa}{r} + \frac{r}{a^2} \tag{1}$$

and consists of a colour Coulomb potential from gluon exchange and a linear confinement term. This is complicated by spin-spin, spin-orbit and tensor interactions between the c and $\bar{c}$ quarks. Using this Ansatz, numerical predictions for masses of the $1D$ and $2P$ states have been obtained [6]:

State	J^{PC}	pred. mass [MeV/c^2]	pred. width [MeV/c^2]
1^3D_3	3^{--}	3830 — 3884	4.80
1^3D_2	3^{--}	3762 — 3819	0.74
1^3D_1	2^{--}	3762 — 3840	186
1^1D_2	2^{-+}	3765 — 3837	0.86
2^3P_2	2^{++}	3979 — 4020	25.6
2^3P_1	1^{++}	3929 — 3990	1.72
2^3P_0	0^{++}	3854 — 3940	55.8
2^1P_1	1^{+-}	3956 — 3990	1.58

Most predictions for the $1D$ state are about 100 MeV/c^2 below the observed $X(3872)$ mass, the $2P$ states are predicted to lie above the $X(3872)$ by a similar amount.

The close proximity to the $D^0\bar{D}^{*0}$ threshold at 3871.3 MeV/c^2 raises the question whether the $X(3872)$ is a new form of matter: a charmed molecule. As early as 1977, DeRújula, Georgi and Glashow [7] investigated the possibility of four-quark states formed by $D^{(*)}\bar{D}^{(*)}$. These could then decay either by dissociation (i.e. the $D^{(*)}$ and the $\bar{D}^{(*)}$ inside the molecular state break apart) or via molecular transitions by emission of mesons such as π or ρ^0 to an intermediate molecular state as illustrated by figure 1.6.

Törnqvist [8] discusses the interpretation that the $X(3872)$ is a $D\bar{D}^*$ deuteron-like system called "deuson" which is bound by pion exchange. He finds that an attractive potential exists only for the quantum numbers $J^{PC} = 1^{++}$ or 0^{-+}, for other assignments the pion exchange is repulsive or too weak to expect bound states. Due to the large isospin breaking (the $D^0\bar{D}^{*0}$ system is $\approx$ 8 MeV/c^2 lighter than the $D^+\bar{D}^{*-}$ system which is comparable to the expected binding energy) the decay-chain $X(3872) \to J/\psi\rho^0 \to J/\psi\pi^+\pi^-$ is not forbidden. Following this approach, Swanson [9] finds an important admixture of $J/\psi\omega$ if the $X(3872)$ is a $J^{PC} = 1^{++}$ $D\bar{D}^*$ molecular state in addition to the $J/\psi\rho^0$ component. Hence the $X(3872)$ should be visible in the invariant $J/\psi\omega \to J/\psi\pi^+\pi^-\pi^0$ mass spectrum.

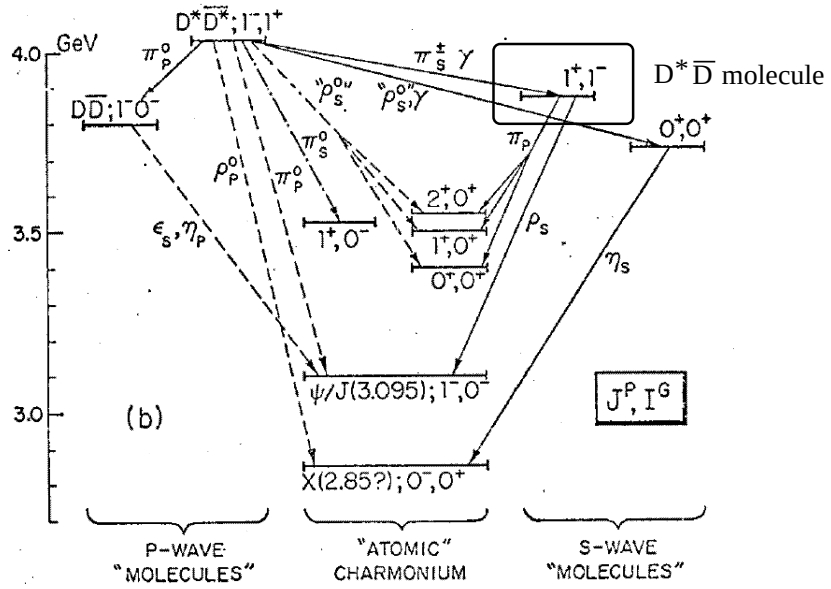


Figure 1.6: Spectrum of $D^{(*)}\bar{D}^{(*)}$ molecules predicted by Glashow *et al.* in 1977 [7].

1.4 Discussion

Many properties of the $X(3872)$ have already been determined, however the exact nature of the newly discovered particle remains unclear. As discussed above, the interpretation of the $X(3872)$ as a charmonium ($c\bar{c}$) state has its shortcomings: The mass of the available candidates is predicted ≈ 100 MeV/c^2 off the observed value. The states 2^3P_2, 2^3P_0 and 1^3D_1 can be eliminated from the list of available assignments due to their large predicted widths [6]. The assignment 2^1P_1 (i.e. the yet unobserved h_c') also seems unlikely as then the ordinary 1^1P_1 (h_c) should be observed as well. Furthermore, this assignment is disfavoured by Belle [18]. If the $X(3872)$ was one of the charmonium candidates 1^3D_2 and 1^3D_3 then the decays $X(3872) \to \chi_{c1,c2}\gamma$ should also be observed, however, no experiment has found evidence for this so far. Any state with quantum numbers $J^{PC} = 1^{--}$ is strongly disfavoured due to the non-observation of the $X(3872)$ in e^+e^- collisions.

The $X(3872)$ may also be an exotic form of matter. One alternative is a so-called 'hybrid state' formed by quarks and gluon s (i.e. $c\bar{c}g$) suggested by [28], however, these states are expected at higher masses. All results presented by the collaborations so far are compatible with the interpretation that the $X(3872)$ is a $D\bar{D}^*$ molecular state as predicted by Törnqvist [8] and Swanson [9]. This state has the possible quantum number assignments $J^{PC} = 1^{++}$ or 0^{-+}. It should be pointed out that if the $X(3872)$ is indeed a molecular state, similar states could exist in the B system. A likely decay

channel is $X_b \to \Upsilon(1S)\pi^+\pi^-$ with $\Upsilon(1S) \to \mu^+\mu^-$, i.e. replacing the charm quarks by beauty quarks.

Chapter 2

Experimental Setup

2.1 The Tevatron

Overview.—The *Tevatron* is a symmetric proton and anti-proton collider ring with a circumference of $\approx$ 4 km, located at the Fermi National Accelerator Laboratory (FNAL or Fermilab) in Batavia/Illinois, USA. In Run 1, the first phase of operation from 1992 to 1996, the *Tevatron* was running at a centre-of-mass energy of $\sqrt{s} = 1.8$ TeV. Among the highlights of physics results from Run 1 are the first experimental evidence for the top quark [29] and a high accuracy measurement of its mass[30]. In the ongoing Run 2, the second phase of *Tevatron* operation started in 2001, the two *Tevatron* experiments CDF 2 and DØ are pursuing physics goals such as measuring the $B_s^0 \bar{B}_s^0$ oscillation frequency Δm_s, Top-Quark physics, Higgs searches and analyses of rare physical processes. The *Tevatron* accelerator was upgraded to achieve a higher instantaneous luminosity and a centre-of-mass energy of $\sqrt{s} = 1.96$ TeV.

The accelerator chain.—Fig. 2.1 shows a schematic view of the Fermilab accelerator chain for Run 2. In the first step of acceleration negatively charged hydrogen ions are produced in a Cockcroft-Walton pre-accelerator and injected into the *Linac*, a linear accelerator 150 m in length. The *Linac* accelerates the ions to an energy of approximately 750 keV. The protons produced by stripping the electrons off the hydrogen ions are then fed into the *Booster*, a 150 m diameter synchrotron. When leaving the *Booster*, the protons have an energy of about 8 GeV. Before injection into the *Tevatron* they undergo a final pre-acceleration in the *Main Injector* which gives them an energy of 120 GeV. The *Main Injector* proton beam is also used to produce the anti-protons by focusing it onto a fixed nickel target. After separating the anti-protons from the numerous different particles emerging from this collision, they are focused and stored in the *Accumulator Ring*. Once a sufficiently large number of anti-protons is stored, they are fed back into the *Main Injector* where they are accelerated to 120 GeV before

injection into the *Tevatron*. In the *Tevatron*, the proton and anti-proton beam get their final energy of 0.98 TeV, yielding the centre-of-mass energy of $\sqrt{s} = 1.96$ TeV.

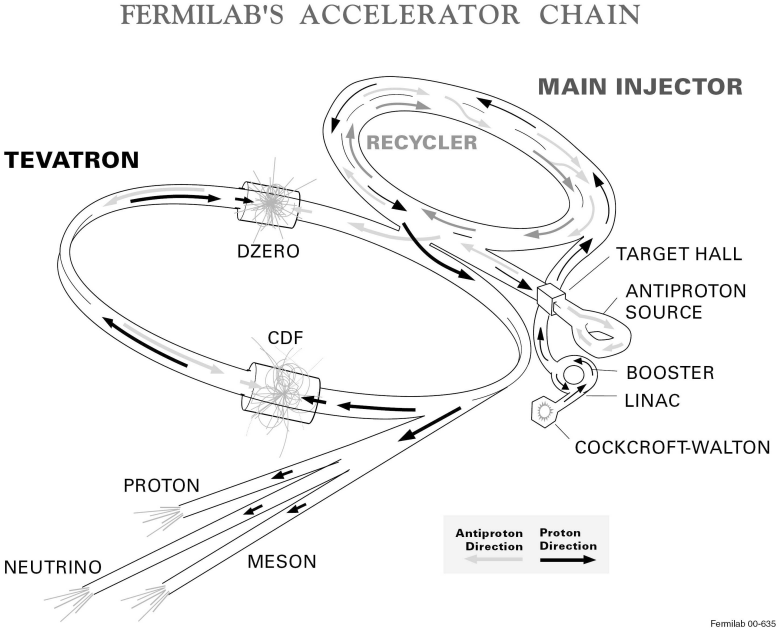


Figure 2.1: The Fermilab accelerator complex for Run 2.

Performance.—Already in Run 1 the anti-proton production was the major limiting factor of the *Tevatron* efficiency. In order to improve the situation for Run 2 the *Recycler* was introduced. The idea is to re-use the remaining anti-protons after a *Tevatron* store. About 75% of the anti-protons are expected to survive a store. These are decelerated down to the energy of 8 GeV in the *Main Injector* and then stored in the *Recycler* for re-use in the next *Tevatron* fill. Unfortunately, in the beginning of Run 2, the *Recycler* could not be commissioned as planned and anti-protons were vanishing at a higher rate than expected. Thus the Run 2 design luminosity of $\mathcal{L} = 10^{32}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$ was not reached immediately. While this has delayed some of the physics goals of *Tevatron* Run 2, the Fermilab Accelerator Division has meanwhile identified and solved the major problems and the *Tevatron* is now working close to Run 2 design specifications. Figures 2.2 and 2.3 show the *Tevatron* Run 2 integrated luminosity and peak luminosity, respectively[1].

[1] As of September 2005.

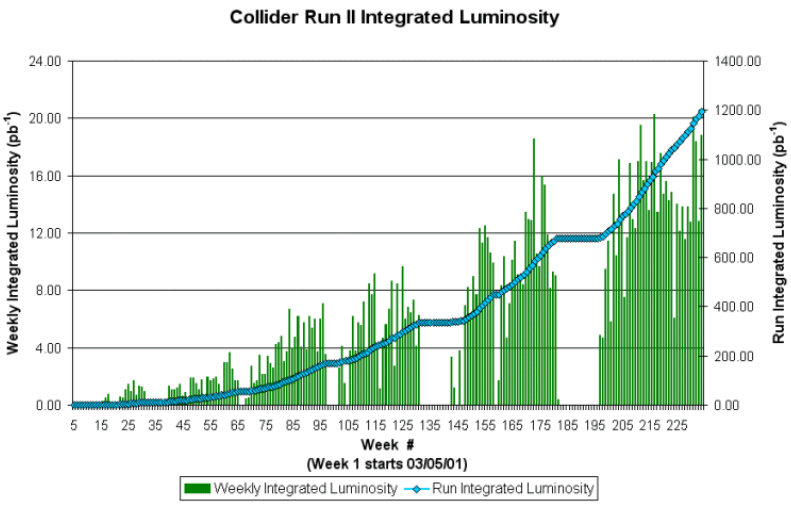


Figure 2.2: Integrated *Tevatron* Run 2 luminosity.

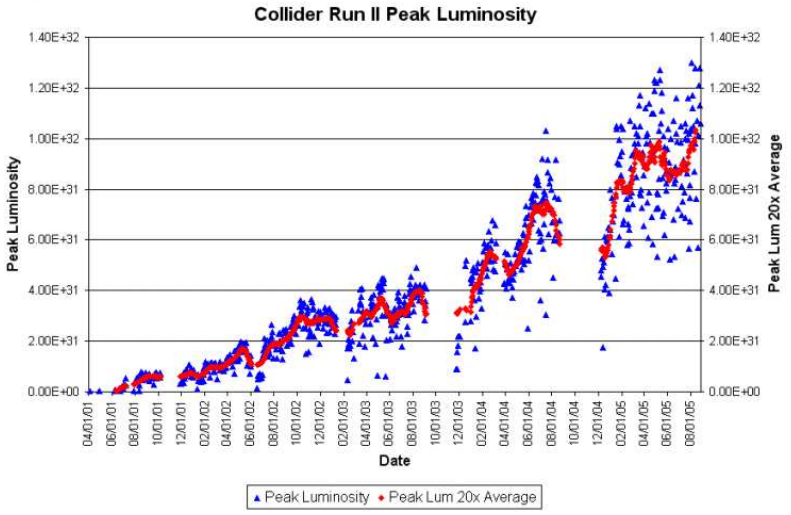


Figure 2.3: *Tevatron* Run 2 peak luminosity.

2.2 The CDF 2 Detector

Overview.—The CDF 2 detector is a general purpose collider detector [31]. It features a vertexing and tracking system, particle identification, a superconducting solenoid generating a 1.4 T magnetic field, calorimetry and muon chambers. The components are arranged in the cylindrical symmetry typical to collider detectors. Figure 2.4 shows a side view of the CDF 2 detector.

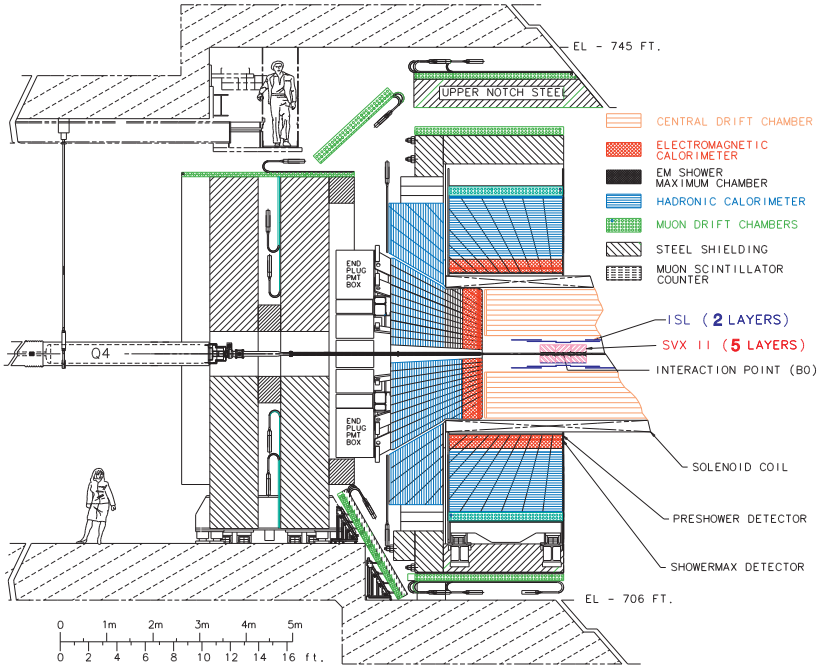


Figure 2.4: The CDF 2 detector.

A right-handed coordinate system in which the positive z direction is defined by the direction of the proton beam. Geographically, the proton beam points east at the location of the CDF 2 detector. The polar angle θ is measured from the positive z direction and the azimuthal angle ϕ is measured from the plane defined by the *Tevatron* ring.

The Tracking System.—Precise and efficient reconstruction of charged particle tracks is crucial to most CDF 2 analyses. The tracking system consists of two major components: the *Central Outer Tracker* (COT) and a silicon vertex detector. The COT is a cylindrical drift chamber 304 cm in length along the z axis, covering the radial region $44\,\text{cm} < r < 132\,\text{cm}$. This corresponds to a pseudo-rapidity coverage of $|\eta| < 1$

where the pseudo-rapidity is defined as $\eta = -\ln(\tan(\theta/2))$ The pseudo-rapidity η has the unique property that in hadron-hadron collisions the particle density is almost constant in equal intervals of η. The COT has eight super-layers in radial direction with twelve measurements in each super-layer. Four out of the eight super-layers are axial layers, measuring only track parameters in the r-ϕ plane. The remaining four super-layers add z information by virtue of a stereo angle of $\pm 3°$.

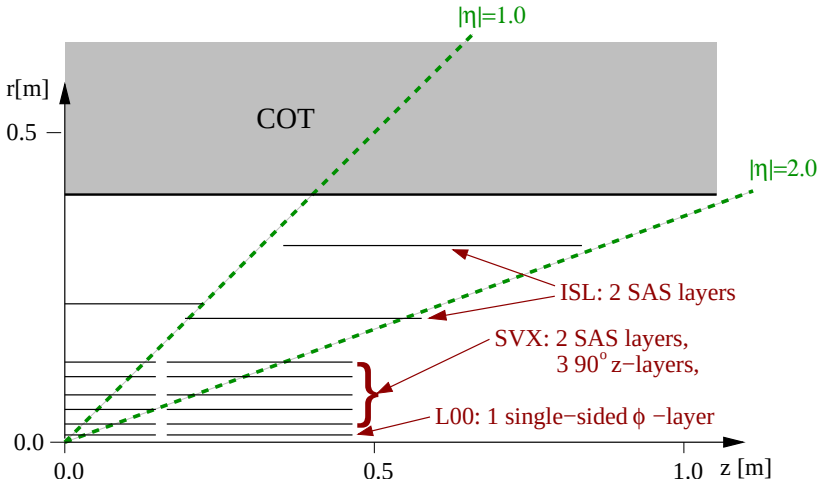


Figure 2.5: Geometry of the CDF 2 silicon detector.

The silicon tracking system consists of three subsystems: the SVX II, the *Intermediate Silicon Layers* (ISL) and the so-called *Layer 00* (L00). The geometry and pseudo-rapidity coverage of the silicon detector is illustrated in fig. 2.5. The SVX II is organised in three barrels with five layers of double-sided silicon micro-strip sensors arranged in twelve wedges. One side of the silicon sensors measure r-ϕ parameters with strips parallel to the z axis. The strips on the other side are tilted by a stereo angle and allow z parameter measurements. There are two different types of sensors with respect to the stereo angle: 90° stereo and *Shallow Angle Stereo* (SAS) sensors with an angle of $\pm 1.2°$. The SVX II has two SAS and three 90° stereo layers.
The ISL is located between the SVX II and the COT. Its main purpose is to provide measurement points close to the COT when pursuing drift chamber tracks into the silicon detector.

The innermost subsystem of the silicon detector is L00. It is composed of single-sided micro-strip silicon sensors, mounted directly onto the beryllium beam-pipe. The purpose of L00 is to improve the the r-ϕ resolution close to the interaction point.

Particle Identification.— Besides the muon chambers, the only detector component with the sole purpose of identifying particles is the *Time of Flight* detector (ToF). The

ToF is mounted just outside the COT inside the solenoid approximately 140 cm away from the beam-pipe as shown in fig. 2.6. It consists of 215 scintillating bars running the length of the COT. A photomultiplier is located at each end of each bar detecting the light emitted by the traversing particle. For a given particle, the combination of the momentum measurement from the COT and the time of flight measurement allows to compute the particles mass: Particles produced at the time t_0 traverse the detector until they reach the ToF-detector, travelling the known distance L. The ToF-detector then measures the arrival time t which can be combined with the measured momentum p to an estimate of the particle's mass: $m = \frac{p}{c}\sqrt{\left(\frac{ct}{L}\right)^2 - 1}$. Given the resolution of the time-of-flight detector of $\approx 100ps$, the different particles can be separated with the following significance:

2σ	K/π	separation for p $<$ 1.6 GeV/c
2σ	K/p	separation for p $<$ 2.7 GeV/c
2σ	p/π	separation for p $<$ 3.2 GeV/c
1.2σ	K/p	separation over all p

Thus one can distinguish particles of different mass, especially protons K-mesons and π-mesons. This only works for charged particles since neutral particles can not be detected in the drift chamber. This information is complemented with the measurement of the specific energy-loss in the COT which is described by the Bethe-Bloch formula (see e.g. [32]). The energy-loss depends both on the velocity $\beta = v/c$ and on the mass of the particle transversing the drift chamber. Hence a precise measurement of the specific energy-loss dE/dx allows to discriminate between different particle types. The new TOF detector is especially powerful in the momentum region between one and two GeV where the drift chamber dE/dx does not give much information as illustrated by figure 2.7.

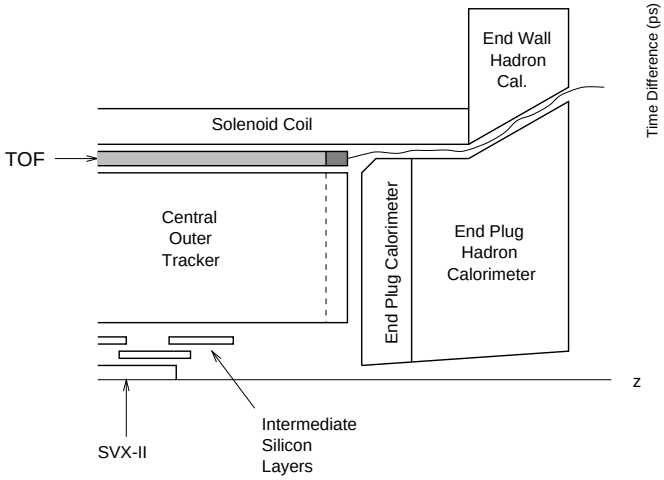


Figure 2.6: Location of the TOF system in the CDF 2 detector.

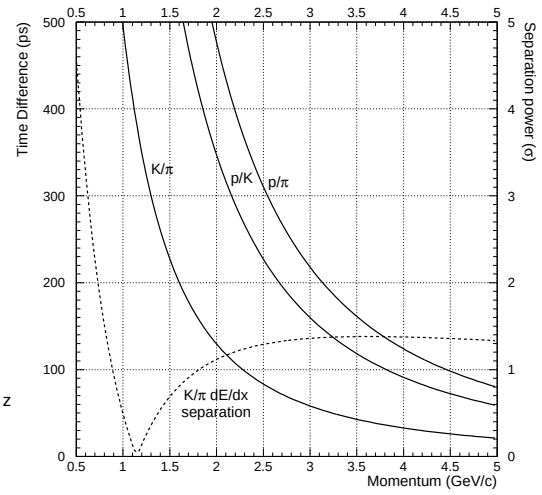


Figure 2.7: Time of flight differences as a function of particle type and momentum. The dashed line indicates K/π separation power of COT dE/dx.

Calorimeters.—The CDF 2 detector features several calorimeters: central electromagnetic and hadron calorimeters, end-wall hadron calorimeters and end-plug electromagnetic and hadron calorimeters. The calorimetry covers the whole azimuth range and the pseudo-rapidity region $|\eta| < 3.64$ and is designed to measure the energy of hadronic jets photons and electrons. The calorimeters are mounted outside the solenoid, as can be seen in fig. 2.4. They are designed as sampling calorimeters, i.e. slices of absorber material alternate with scintilators which allows to measure the total energy of the incident particle as well as the lateral and longitudinal shower development. The calorimeters are unchanged since Run 1, however the read-out electronics has been upgraded to handle the higher luminosity of Run 2.

The *Central Electromagnetic Calorimeter* (CEM) measures the energy of electromagnetic showers in the central region of the detector ($|\eta| < 1.1$) utilising lead (Pb) as absorber material. It consists of 31 layers organised into 24 wedges in ϕ.

Adjoining the electromagnetic calorimeter the *Central Hadronic Calorimeter* (CHA) determines the energy of hadronic showers. This detector covers the region $|\eta| < 0.9$ using iron (Fe) as absorber material. A total of 384 towers are constructed from its 32 layers. This calorimeter is extended by the *Endwall Hadronic Calorimeter* (WHA) covering the region $0.8 \leq |\eta| \leq 1.2$ which consists of 15 layers, also using iron as absorber material.

Additionally, the *Central Electromagnetic Showermax chamber* (CES) provides high precision position measurements at the shower maximum which is utilised to match calorimeter showers to reconstructed tracks. It is located ≈ 6 radiation lengths inside the CEM and consists of 2×24 modules matching the geometry of the CEM calorimeter. The *Central Pre-Radiate chamber* (CPR) mounted on the inside surface of the CEM detector provides additional discrimination power between electrons and hadrons.

Similar calorimeter systems are deployed in the plug-region of the detector to cover the forward region $1.1 \leq |\eta| \leq 3.6$.

The Muon System.—The muon system is the outermost part of the CDF 2 detector. It consists of scintilators drift cells and steel absorbers. Usually only muons reach the muon chambers since all other particles are stopped inside the calorimeter or the steel absorbers. In order to reach the muon chambers a muon must have a momentum of $\sim$ 1.5 GeV. The muon system consists of several components covering in total the range $|\eta| < 1.5$:

detector component	abbreviation	$\|\eta\|_{min}$	$\|\eta\|_{max}$
central muon chamber	CMU	0.0	0.6
central muon upgrade	CMP, CSP	0.0	0.6
central muon extension	CMX, CSX	0.6	1.0
intermediate muon chamber	BMU, BSU-TSU	1.0, 1.0-1.3	1.5, 1.5-1.5

The CMU system was already installed in the first commissioning run (1987), the central muon detectors CMP, CSP, CMX, CSX were installed for Run 1, whereas the intermediate chambers were added during the Run 2 upgrade. Most B-physics analyses using muons include information from the central region ($|\eta| < 1.0$) only.

2.3 The CDF 2 Trigger System

Overview.—The collision rate at the *Tevatron* is much higher than the rate at which data can be stored on mass storage. Thus the trigger plays an important role in selecting the interesting events from a huge background: The huge inelastic cross section of $\sigma \approx 60$ mb is approximately 5000 times higher than the production cross section of B hadron events ($\sigma(p\bar{p} \rightarrow b + X) \approx 20\ \mu b$). Thus rigorous and highly efficient triggers have to be deployed to select the events of physics interest.

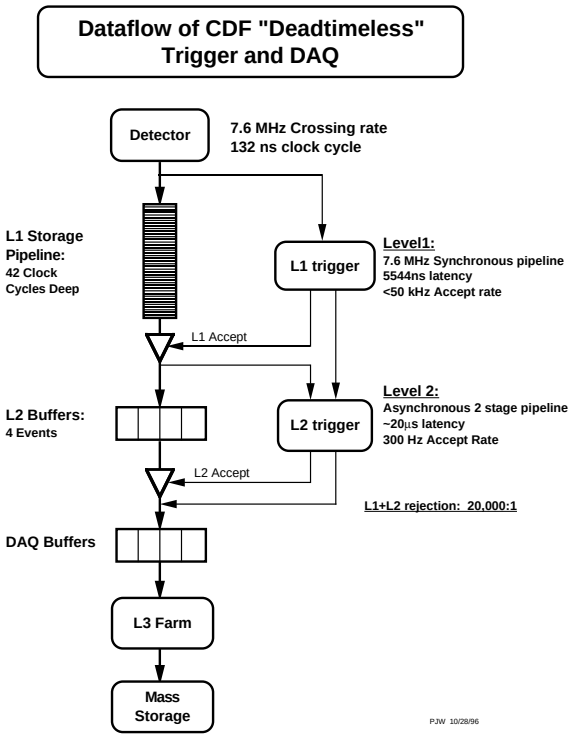


Figure 2.8: Data-flow in the CDF 2 trigger and data acquisition system.

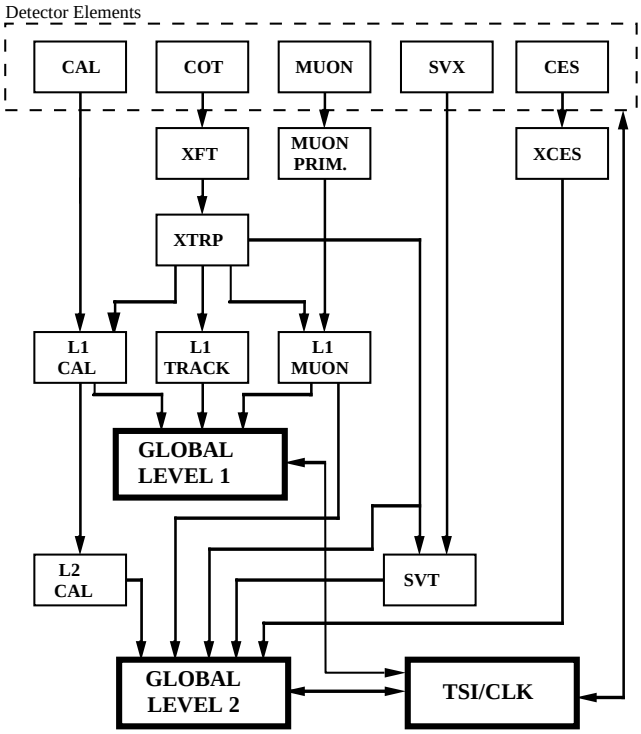


Figure 2.9: The first two trigger levels and the involved detector elements.

The CDF 2 trigger is a three level system with an overall rejection of 120,000:1. This reduces the collision rate to an event output rate of $\sim$ 50 Hz. Given a typical event record size of 200-300 kB this results in $\sim$ 12 MB/s written to mass storage for offline analysis. Fig. 2.8 shows a schematic view of the data-flow in the trigger.

The first (L1) and second (L2) trigger levels are implemented in hardware and make use of several detector components: Tracking information is first obtained by the the *eXtremely Fast Track Finder* (XFT)[33] which finds tracks in the COT in L1. The resulting list of XFT tracks is then passed to the *Silicon Vertex Tracker* (SVT)[34] which is part of L2. The SVT adds silicon hits to the XFT tracks by employing a sophisticated pattern matching algorithm. Information from the calorimeters and muon chambers are included to obtain the trigger decision as illustrated by figure 2.9. It is worth noting that this trigger system works nearly dead-time-less. This is achieved by storing events accepted by L1 in buffers allowing L2 to make its decisions asynchronously. The third trigger level (L3) is implemented in software running on a PC farm. The software environment in L3 is the same as in offline analysis. This allows to base the L3 decisions on event variables reconstructed with offline quality. CDF

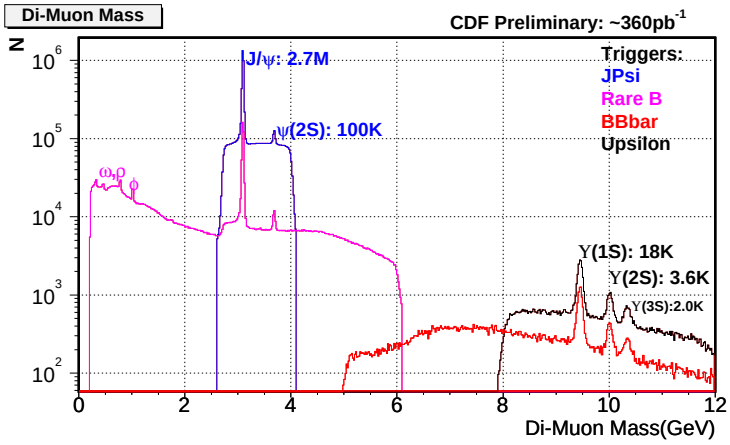


Figure 2.10: Mass spectrum of the various di-muon triggers used in the CDF physics programme [35]. The analysis presented in this thesis uses the dedicated J/ψ trigger.

deploys a large variety of triggers to cover the rich physics programme. The triggers used in this analysis are concerned with the occurrence of J/ψ decaying into either two muons or two electrons. In detail, the trigger $J/\psi \to \mu^+\mu^-$ requires that two muons are observed in the central muon chambers, demanding that the associated reconstructed tracks have a minimal transverse momentum of $p_t > 1.5$ GeV/c which allows the reconstruction of the J/ψ at rest. The invariant mass of the muon pair has to lie in the J/ψ and $\psi(2S)$ mass region. It is required that either both muons are observed in the central muon detector CMU or one muon is detected in the CMU whereas the other transverses the extension CMX. Figure 2.10 shows that $J/\psi \to \mu^+\mu^-$ events can be selected with very low background contamination by this trigger.

Furthermore, a dedicated trigger for the decay $J/\psi \to e^+e^-$ exists which demands that two tracks of opposite charge with a minimal transverse momentum of $p_t > 2$ GeV/c exist and deposit at least an energy $E_t > 2$ GeV in the central electromagnetic calorimeter (CEM). However, events selected by this trigger are dominated by background as will be discussed in section 5.1, thus sophisticated tools are needed to isolate the electrons against a ≈ 10 times higher background. Consequently, this trigger has been granted only very little of the available bandwidth resulting in a much smaller dataset than the di-muon trigger. Furthermore, as this work is (almost) the only analysis using data from this trigger, it has been discontinued early 2005 as more widely used triggers need a larger bandwidth to cope with the rising luminosity of Tevatron.

Chapter 3

Data handling and Grid computing

3.1 Introduction

Overview.— The amount of data produced by high-energy physics experiments has increased significantly in recent experiments. Around July 2005, $\approx$ 350 TB raw data has been stored in the CDF II experiment, all processed datasets which can be used in physics analyses sum up to $\approx$ 2000 TB. It is expected that each of the experiments at the Large Hadron Collider at CERN will produce this amount of data or more annually. A medium sized dataset at CDF II [1] is already about 5 TB big. Processing these large amounts of data is a very difficult task. On one hand, logistical problems have to be solved: As many thousand computers are needed, adequate space needs to be available with adequate supply of electricity and cooling power, to store the data, many file-servers and tape-robots are needed. Also maintenance becomes an issue: As mostly the so called "commodity hardware" found at normal computer vendors is being used for financial reasons, more and more people need to be employed who do nothing but replacing broken hard-disks, cooling fans, etc. when the computing facilities grow bigger and bigger.

On the other hand, the computer technology itself makes it difficult to operate large clusters: Solutions which work very well on a smaller cluster do not scale adequately and cannot be used in large clusters. The example of file-access from several analysis jobs to a file-server illustrates the problem: Typically, a file-server consists of $\approx$ 20 hard-disks which are connected via a RAID controller and provide one large filesystem distributed over all physical disks. Accessing the data stored on such a filesystem with a few jobs (e.g. 20) at the same time works reasonably well, however, to process even the medium sized datasets of a few TB in a reasonable time-scale of a few days, many

[1] A dataset defines which files are to be processed, for example all files containing events from the di-muon trigger, a subset of these files for a given run-period, etc.

more jobs (like 80-100) are needed. If these jobs then access such file-servers, the amount of data delivered to the analysis programs quickly drops to zero as the hard-disks spend then almost all of the time re-adjusting their mechanics in response to the requests for data.

The idea of the Grid (see e.g. [36]) is to build many distributed computing centres across the world avoiding many of these problems. The name "Grid computing" is derived from the analogy of the electrical power grid: The submission of a job and the retrieval of its output should be as easy as plugging in a coffee-machine and switching it on. The user should not have to bother where sufficient resources are to run the job, how the data is being provided, where to find the output of the analysis job and how to get it, etc.

General requirements.— To build the Grid many challenging problems need to be addressed. First of all, many computing clusters (also called the "Fabric") need to be built (e.g. one per participating country) with significant computing power, storage capacity and network connection to the outside world.

The Grid-software then needs to deal with the following aspects:

- *Monitoring participating sites:* Which site has idle computers, how much storage capacity is free, which amount of which data is already there, etc.

- *User interaction:* Can the user be authenticated (i.e. can it be verified that the user really is who he claims to be) and is he authorised to use the facilities? Which type of job does the user want to run (e.g. analysis of data or simulation)? If the user requests that the output is sent back to him (e.g. the resulting ntuples of an analysis program), it either needs to be copied back to the user or the user needs to be notified where and how to retrieve the output.

- *Match-making:* The so called resource-broker "matches" the user's request with the available resources, i.e. it needs to be determined where the job is going to be executed. This decision can be based on the following considerations: If data is needed as input to the user's job, is it already present at some site (i.e. would the overall execution time be quicker if the user has to wait for computers to become free where the data is or is it quicker to copy the data to where free computers are), if the user produces output (e.g. simulated events) is there sufficient storage capacity? If data needs to be copied, is there already a sufficient amount of data that the jobs can be started or is it better to wait until more data arrives and execute other jobs in the meantime?

- *Data handling:* The user specifies which dataset he needs as input to his program, it then needs to be determined which actual files belong to this dataset, where

they are, if data is not yet present at a given site, which other site has the data and where to get it from best.

- *Job execution:* Once it is determined where to run the job, it needs to be sent there and submitted to the local batch system, i.e. an interface between the Grid and the Fabric (i.e. the local facilities) has to exist. This interface needs to be very general as participating sites will differ in their setup, e.g. various batch-systems with different features may be employed to operate the cluster, etc.

Security issues.— Only eligible users may use a given facility, however as the users come from all over the world this is very difficult to verify. A policy where each user has to register with each participating site cannot work given the large amount of sites. As a solution to this problem the concept of "virtual organisations" has been developed: Each member of a collaboration registers with the Certification Authority (CA) for a given site and becomes member of a virtual organisation. If a user then submits a job to the Grid, the virtual organisation determines whether or not the user is authorised to execute the job.

To tighten the security of a computing cluster further means may be employed. Administrators may use a firewall which only allows network traffic for certain network protocols in predetermined port ranges. This has the advantage that certain services can be operated in the cluster (like NFS, etc) without the fear that hackers may use vulnerable code to harm the system. However, this requires that all protocols and ports needed are known to be able to configure the firewall appropriately. Two ways exist to determine the needed information: Whenever a developer writes new code requiring an internet connection, the used protocols and ports are documented, ideally parameters are used to configure the port range needed. In reality, network sniffers (which visualise all communication on the computer) and monitoring tools are used to try to find out which machines, protocols and ports are involved and where the packages are being lost.

Another way to increase the security is to put all computers but some access-nodes in a private network and deploy network address translation to allow outgoing connections. This way user jobs on the worker-nodes can still communicate with the outside world (e.g. to retrieve calibration constants from a remote database) although the cluster itself is not visible from the outside world and hence cannot be attacked. As a drawback, certain programs assuming a public network (e.g. Kerberos) will not work in such an environment.

The Globus toolkit.— This toolkit [37] provides the basic infrastructure for job submission, file-transfer and information management in the grid world. In the past years,

the toolkit has evolved as the de-facto standard for grid-computing and is used both in science and industry.

The Globus toolkit consists of the following basic services:

- *Globus Resource Allocation Manager (GRAM):* This component is intended to provide a uniform interface to grid resources. Individual resources may deploy a large variety of schedulers, batch-systems, etc., interfacing these local services to GRAM generalises these specific implementations such that high-level applications do not have to anticipate each specific setup of a given resource. The main use-case of this component is job submission and control.

- *Grid Security Infrastructure (GSI):* The GSI provides mechanisms for authentification (is the user who he claims to be) and authorisation (is the user allowed to used these facilities). The security structure is based on private/public key cryptographic methods: Each user has a private key, signed by a certification authority (CA) which confirms that the user's identity has been verified by a CA and a public key which uniquely identifies the user. The GSI supports security across organisational boundaries, avoiding a centrally managed system. Instead, multiple CAs can agree to trust each other, if one user is then approved by one CA, he is accepted by all participating CAs (this is called "mutual authentification"). Each user needs to be authenticated only once by creating a grid-proxy: During the lifetime of this proxy, the user can use all grid resources without retyping the password. The credentials are delegated to the respective resources if multiple facilities are involved.

- *Monitoring and Discovery Service (MDS):* Each grid service has a specific set of service data associated with it. The MDS provides means to generate, retrieve and query the information provided. Examples for information managed by MDS are: network status, location of file replicas, unique job ids, etc.

Furthermore, many more specialised tools are provided such as GASS (Global Access to Secondary Storage), a grid-enabled version of FTP for file-transfer, etc.

The Globus Toolkit is very modular, it is possible to use only parts of the provided suite, e.g. install the security infrastructure to identify users or to use gridftp to transfer files while information about resources is gathered otherwise, etc.

3.2 Grid-computing at CDF

Why Grid computing?.— The competitive physics programme of the CDF collaboration requires an enormous amount of computing power. Besides the many analyses

accessing the rapidly growing datasets (e.g. all data taken by the di-muon trigger sums up to a size of currently $\approx$ 4TB), all raw data has to be reprocessed regularly to utilise recent developments and improvements in offline code such as tracking, etc. Furthermore, simulated events are a key ingredience to many analyses. Experience from e.g. LEP shows that one can hardly have enough simulated events. All of these demands cannot be satisfied by one single computing farm, which lead to the decision to build up a significant a mount of computing resources outside of Fermilab.

Challenges in a running experiment.— Building a working computing Grid in a running experiment leads to unique challenges: While the LHC experiments still have a few years of preparations ahead, Run II of the Tevatron experiments has already started in 2002. Since then both CDF and DØ have recorded a significant amount of data – and more data is taken each day. It is therefore imperative that the use of new technologies must not interrupt neither data-taking nor physics analyses. Furthermore, many physics analyses are currently being performed and users are in general not interested in developing or testing new technologies. They will only use the newly developed tools if this leads to a much simpler way to access data and/or more resources being available which speeds up the data analysis drastically.

Ansatz.— The CDF collaboration has chosen to start from working tools and migrate slowly to a full Grid: First, the central analysis farm (CAF) has been deployed on-site Fermilab which can be used by all CDF members. Several off-site computing farms are being set up around the world, most of them initially focus on the production of simulated events first as this requires less online storage. The German facility called "GridKa" is located at the Forschungszentrum Karlsruhe (see sec. 3.3 for details). As a significant amount of both disk and tape storage and computing power are available for CDF, these resources can be used in real physics analyses; hence GridKa is considered as a prototype of a remote analysis farm. The software SAM (see sec. 3.4) is jointly being developed by CDF and DØ and used to manage the data, i.e. copy the data between the centres, deliver the files the user,etc. Within this system, the same approach has been taken: working technologies are taken first and are replaced by more "grid-like" tools as the software evolves. For example, in earlier versions of SAM files were copied using the BaBar FTP implementation [38]. As the Globus toolkit became available and well tested, this method was replaced by the GridFTP protocol. This Ansatz has the advantage that emerging technologies can be immediately tested in "real life" situations, minimising the risk of wrong developments and missing essential features. Furthermore, all resources are available to the users for their analysis. However, not each participating site may be able to cover all functionality during the development: The design of the Grid software may not yet be general enough to be able to function with given constraints (e.g. policies of the participating sites, etc.). These experiences have then to be incorporated into the design and development of

month/year	11/2001	4/2002	4/2003	4/2004	2005	2006	2007
CPU (kSi95)	1	10	25	60	150	325	900
disk (TB)	7	45	113	210	440	850	1500
tape (TB)	7	111	211	350	800	2000	3700

Table 3.1: Planned upgrade plan for GridKa according to the design document [39]. Does not yet include the BaBar TierA upgrade and the DØ changes. 1 kSi95 $\approx$ 24 * 1 GHz PIII or 1 GHz PIII $\approx$ 42 Si95.

later releases.

3.3 GridKa

Overview.— GridKa is the German Grid computing centre located at the research-centre "Forschungszentrum Karlsruhe". Eight high-energy physics experiments (Alice, Atlas, BaBar, CDF, CMS, Compass, DØ, LHCb) share a large cluster. The requirements for this centre are defined in [39, 40]. In the respective terminology, GridKa is a Tier1-centre for the LHC experiments (Alice, Atlas, CMS, Compass, LHCb) and TierA for BaBar, i.e. a major computing facility to store significant amount of data, process a large number of user jobs, reprocess data and provide simulated data. For both CDF and DØ GridKa is used to process a significant amount of user jobs and to evaluate and test new Grid and data-handling technologies. GridKa is managed by two boards: the Overview Board (OB) focusing on management questions and the Technical Advisory Board (TAB) discussing technical aspects such as operations, recommendations for future upgrades, etc.

Resources.—

Each experiment has access to a dedicated access computer which is used to both provide the experiment specific software and to allow users to submit jobs to the cluster and retrieve the output.

The worker-nodes are shared between all experiments and hence no experiment specific software is installed there. PBSPro [42] is used to submit and run jobs. A fair-share manager ensures that each experiment can use the resources it paid for. Figure 3.1 illustrates the setup.

Sharing the worker-nodes between the experiments has the advantage that other experiments may use computing time not used by an experiment. For example, LHC experiments require a large amount of computing power during the so-called data-challenges which test operational issues of the LHC Grid and simulates data-taking,

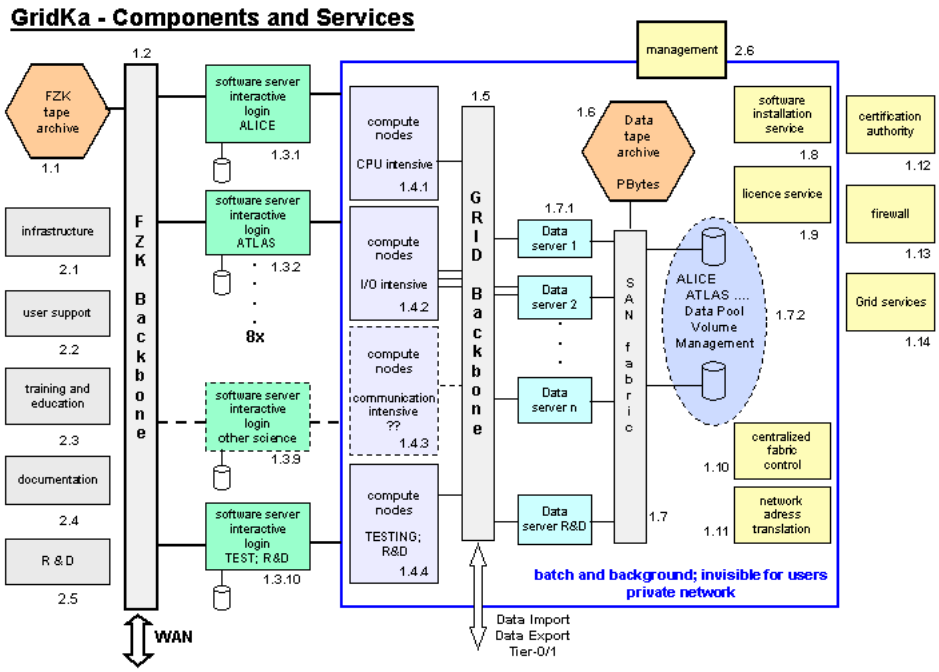


Figure 3.1: Schematic overview of the GridKa setup [41].

reprocessing, etc. However, once these data-challenges are completed the need for computing power drops significantly as the results need to be analysed. These otherwise idle computers can than be used e.g. by the CDF group as illustrated by figure 3.2.

Disk storage is organised in a storage area network (SAN) [43], using GPFS [44] as the underlying filesystem. The tape robot is controlled by the Tivoli Storage Manager (TSM) [45] and can be accessed both via the TSM tools (for backup and archive) and dCache [46].

A ticket-based helpdesk system called "Global Grid User Support" [47] is used to report problems and interact with the users.

3.4 The SAM data-handling system

Overview.— The software SAM has been developed at Fermilab to access the data, distribute files around the various resources and provide an user-interface to define which data should be processed. The name SAM is an abbreviation for Sequential

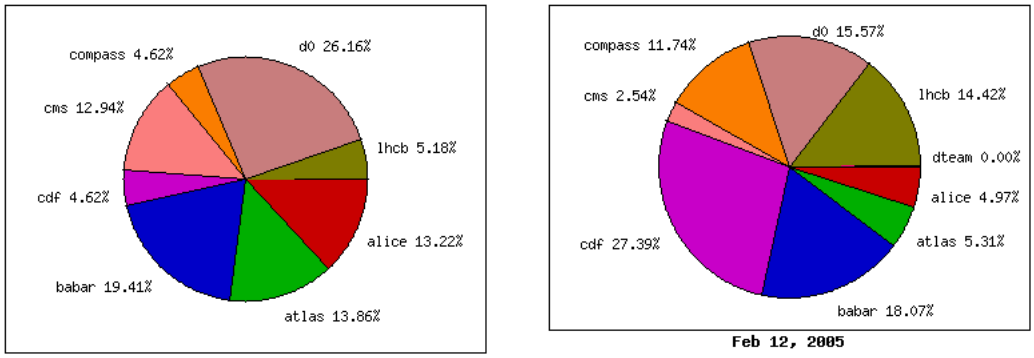


Figure 3.2: These graphs show how the computing power at GridKa is shared between the experiments. The left diagram shows the nominal distribution (i.e. according to the design), the right diagram reflects the actual usage as of February 2005. Due to the common setup at GridKa, CDF users were able to utilise computers not needed by the other experiments (e.g. between LHC data-challenges). Figures courtesy GridKa administration.

Access via Metadata which illustrates the basic principle: Data is being processed sequentially (i.e. one file after the other) and is associated with meta-data which can be used to select the desired files, describe the content or who created the files, etc. The development was started by DØ, later CDF joined the efforts, more recently MINOS also decided to use it.

Data and Metadata.— Each file containing physics data is associated with metadata which describes it uniquely and can be used to select the files of interest. This metadata contains information about:

- *Physical file information:* file-size, creation date, name of creator, check sum, etc.
- *General information:* data or simulation, parent files (if this file is the result of a merge of several other files), run information (which runs and run-sections), event information (first event, last event and number of events), a free-text string describing the dataset this file belongs to, etc.
- *Data information:* which trigger stream the file originates from, etc.
- *Simulation information:* Name and version of generator used, settings of the simulation, etc.

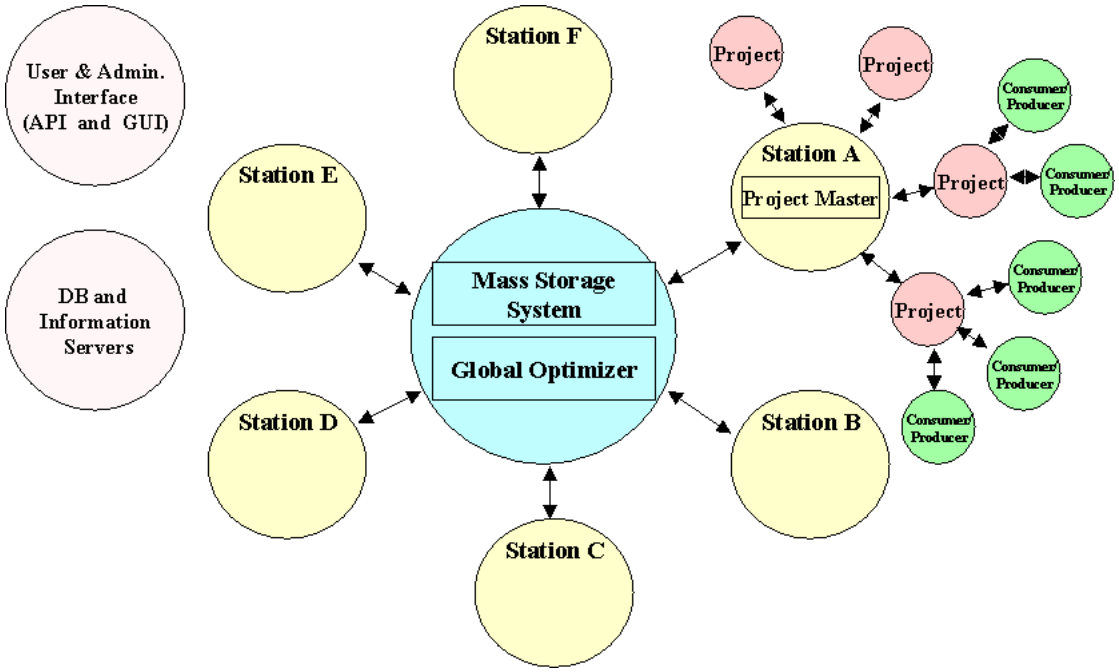


Figure 3.3: Illustration of the SAM architecture [48].

The files of interest can be selected using the metadata, e.g." "select all files originating from a given trigger" or "select all files from a given trigger in a certain run-range", etc. These selection criteria are then passed to the SAM system which takes care of determining which files fulfil these criteria and delivers them to the user job.

Components of SAM.— The whole SAM system can be separated into different components described in detail below. They can be grouped into parts which need to run only once per experiment and components which need to run at each participating site.

The following components are needed only once and are operated at Fermilab. SAM makes extensive use of a central database, among other things the following items are logged: user starting or ending a project, opening or closing file, the location of each file, a history which file has been stored where, when it was last accessed, etc. The `dbServer` is used to access the database and "translates" the SAM communication into `SQL` statements the database understands. The `Optimiser` authorises file-transfers, whereas the naming-service ensures that all parts of the SAM system can communicate with each other.

Each facility runs a station master, a file storage server with an associated stager and a Grid-FTP daemon. Project masters and stagers are started automatically on demand by the station master.

- *Station master:* This is the main component the user mainly deals with when analysing files. It controls data import from other facilities, delivery of data to the user job and keeps track of the user's activity.

- *Project master:* The project master directly interacts with the user job, keeps track of which files have already been processed by the job and which files are still being requested.

- *File Storage Server (FSS):* If new files are to be stored in the SAM system, the FSS ensures that the file is being transfered to the storage location. If the final location is not directly accessible, the file is routed to other FSS running at other sites.

- *Naming service:* Most internal communication between the processes is based on CORBA [49, 50], each application (like station master, project master, etc.) registers with one or several naming services and keeps track of which process runs where. If a process needs to communicate with other processes, the naming service provides the information where these processes can be found. Although it is not necessary to have several naming services, it may be beneficial for large off-site resources to set up an own naming service in addition to the central one. Then small network glitches do not interrupt the communication between the processes local to the off-site cluster.

- *Grid-FTP Daemon:* This daemon handles requests for outbound file-transfers. All file-transfers are done using the grid-ftp implementation provided by the Globus Toolkit.

- *Stager:* A stager controls one cache area, it is responsible for e.g. ensuring that the area is writable, no unknown files are present or files are missing. One stager has to be started per cache area, usually, this is done automatically by the station itself. A further stager is associated with the FSS to handle files which are going to be stored.

- *Optimiser:* The optimiser decides where new files are being imported from and in which order. The basic strategy is to avoid tape-based systems as these are considered to be slow and to prefer disk-based caches, e.g. at other facilities. As the optimiser schedules the file-transfers between different sites globally, only one instance runs per experiment.

- *dbServer:* All activity is logged in an Oracle database. This includes which files are available where, which files belong to which dataset, who opened which file at what time, etc. The `dbServer` translates all requests into SQL and uses the resulting query to retrieve the answer from the database. This is then translated

back for further processing by SAM components. If information needs to be stored in the database, the `dbServer` executes the necessary SQL commands. In principle, several dbServers could be deployed at different remote facilities. However, extensive data needs to be exchanged between the `dbServer` and the central database, the resulting network latencies make a successfully operation of remote `dbServers` impossible. Several of these servers are used to increase robustness and isolate e.g. user interaction from station use.

Figure 3.3 illustrates how these components work together, section 3.4 describes the interaction using the example that a user wants to analyse some data at the (remote) SAM station A. Generally speaking, all components form a "star-shaped" topology. The services running at Fermilab form the centre of the star, the various off-site resources operate a SAM station and form the "rays" of the star. There is no direct communication between the different stations, all communication is performed by contacting the central database and services. Although this is a single point of failure (i.e. if the central services are not available the whole system is inoperable) this is in practice not so much of a problem since these services are intensely monitored and experts are available each day of the year around the clock to take the necessary actions in case of an incident.

Using SAM to analyse data.— Several interfaces exist to interact with SAM: A command line interface, a Python interface for more complex scripts, a C++ API to use SAM in own programs or ROOT [51] macros and an extension to the `DHInput` [52] module used in the CDF analysis framework `AC++`. Most users will use SAM to deliver files to analysis jobs running in the `AC++` framework accessing directly the files containing the output of the (re-) processing of the raw data. However, several physics groups base their analyses on derived files such as `ROOT` StNtuples.

The following example describes the necessary steps to run an analysis project with SAM and illustrates how the various components of SAM involved work together.

1. First, a "dataset" needs to be defined. The smallest unit handled by SAM is a single file, the dataset defines criteria to select the files of interest [2]. To create a SAM dataset, either a web-based editor [3] or the command line or Python interface can be used. Appendix A.6 gives a few examples.

2. Using this dataset, a project has to be started, e.g using the command `sam start project`. The station master then contacts the central database via the `dbServer`

[2] Prior to using SAM, CDF used a tool called DFC (data file catalogue) to associate a name to files belonging together (e.g. all files from a specific trigger with a given reprocessing). This was also called a "dataset", hence the same terminology can have the same meaning - but does not necessarily have to.

[3] `http://cdfdb.fnal.gov/sam_project_editor/DatasetEditor.html`

to determine which files are going to be analysed based on the definition of the dataset defined in the previous step. To determine these files and their current location the associated metadata is used. The list of all files involved and their present location is then sent back to the station. If not all files are already cached at the station, the optimiser is contacted to determine which files are to be copied from which other location. The start of the project is then recorded in the database and the station master starts another process called the "project master" which handles the communication between the clients receiving the files and the station master. The name of a project is unique and it cannot be used again once a project is finished. Files can now be delivered to the user.

3. Analysis jobs can be submitted to the cluster. Several jobs can connect to the same project and analyse the files.

4. Each job participating in the project registers with the project master (and with the station) and requests files from SAM. Each job obtains a unique internal identification number which is also recorded in the database. SAM starts to hand out files to the jobs on the worker-nodes and first delivers the files already in the cache areas, as soon as only a few files are left the import of the missing files is started. Once all files are processed, the special stream `''end of file''` is sent to the job which should then terminate itself. SAM records into the database the time when the file was opened by a job, when it was closed and which status it was closed with.

5. When all files are processed the project ends itself and records this in the database.

It is important that the project is started *before* the first job on the worker-node starts. Depending on the number of files to process the initial startup can take several minutes. Although the name of the project is already known in the database during this time, the project master cannot yet react to incoming requests, i.e. other attempts to start a project with the same name will fail. This needs to be kept in mind when using the `DHInput` module in the `AC++` framework which first tries to connect to an existing project and then tries to start a project by itself if no contact can be established.

Using SAM to store new data.— Another important application of SAM is to store a file generated by a user (e.g. from simulation) into the system, typically on a tape-robot or a dedicated disk. To do so, the user needs to specify a minimal set of metadata (filename, filesize, first and last event, number of events, who created the file, a description of the content, run-number and run-sections, etc.) and create a Python dictionary. Using this dictionary, the file can be stored via a call to `sam store`. The FSS then accepts the file and transfers it to the final destination. If

the destination is not directly accessible the file is sent to other SAM stations which have access to the storage location. To simplify this process and retrieve the required metadata automatically the tool `samStoreCdfFile` can be used which is part of the CDF software.

Configuration for off-site use.— This section describes the necessary configurations to operate a SAM station at a remote site. Although the description is valid in general, special attention will be paid to the setup at GridKa and the University of Karlsruhe.

The SAM station at GridKa is called `cdf-fzkka` and has access to multiple TB of disk cache and tape. The integration of tape at GridKa will be discussed in detail in the next section. The SAM station at the university is called `cdf-ekpka` and serves the Institute's cluster on a much smaller scale. It does not have direct access to a tape system and the disk cache is $\approx$ 500 GB big. The idea is to operate this station as a "satellite station" of the GridKa station, i.e. all files not yet present at the university should only be copied from GridKa.

A SAM station is configured via two means: The so called "server-list" file, and by a product [4] called ``sam_config''.

The SAM station is started and stopped by a product called `sam_bootstrap` which parses the server-list file and starts (stops) the parts of the station listed. Additionally it monitors whether the started products are still running and restarts them automatically in case a process crashes.

The server-list file lists which parts of a SAM station should be started and lists the arguments passed to the respective executable. The format of the file is:

```
<service name> <configuration qualifier> <version> <options>
```

Each line describes one service. Typically, a SAM station runs: The station master, a file-storage server with associated stager and a grid-ftp daemon. The configuration qualifier distinguishes between several possible configurations of the product `sam_config` which makes it possible to use e.g. different dbServers for the station and user interaction or to run different parts of the SAM suite on different computers. As all parts of a SAM station communicate via CORBA, it is not necessary that all parts of a SAM station run on the same computer, they can even be distributed globally if necessary. The GridKa SAM station runs in addition an own naming service to circumvent small network problems which last only for a short time. Appendix A.2 focuses in detail on the server list for the SAM stations at GridKa and the University of Karlsruhe.

Configuring the product `sam_config` ensures that other necessary products are being made available and that environment variables needed are set to the appropriate

[4] Software from FermiLab is distributed and maintained by a package management system called `ups/ups` which can handle the installation of different versions of the same software. Software packages distributed this way are commonly referred to as "products".

value. These environment variables determine e.g. which dbServer the SAM station talks to, which naming service to use, etc. A detailed discussion can be found in appendix A.3.

A further important aspect is the configuration of the product `sam_cp` which determines which protocol is chosen for a file transfer. The configuration is done in two places: The basic configuration file `sam_cp_config.py` accessible via the environment variable `SAM_CP_CONFIG_FILE` and via the so called "arbitrators". The configuration file is implemented in Python and defines the usable protocols and static maps between SAM stations, which define which protocol to use. The structure of this file is:

```
import SamCpClasses

 [...]
KNOWN_SAM_CP_CLASSES = {  'dcache'           : SamCpClasses.SamDcache,
                          [...]
                          'sam_gridftp'      : SamCpClasses.SamGridftp,
                          'dcache_gridftp'   : SamCpClasses.SamDcacheGridftp,
                          }

DOMAIN_CAPABILITY_MAP = { 'enstore'      : [ 'dcache_gridftp', \
                                             'dcache',  ],
                           'fnal.gov'     : [ 'sam_gridftp', ],
                           'cdf.fzk.de'   :  [ 'sam_gridftp', ],
                           'some.station' : [ 'protocol1',  \
                                             'protocol2 , ],
                         }
```

First, the Python module `SamCpClasses` is being imported which contains details about the implementation of the various protocols. The dictionary describing the known `sam_cp` classes contains the information about which protocols are available and where to find the implementation. The capability map determines which protocol is chosen for a file transfer. The first part is a mask describing the station (e.g. the mask 'cdf.fzk.de' covers only the CDF head-node at GridKa, whereas 'fnal.gov' refers to all machines at FermiLab), the second part lists all viable protocols. SAM chooses the first protocol common to both stations. The keyword 'enstore' refers to all files which are stored in a tape-location, regardless if the Enstore protocol is used to control the tape-robot or not.

The "arbitrators" provide a much more flexible way to determine which protocol to choose than these static maps which will be discussed in the next section in more detail.

Integrating tape storage at GridKa.— GridKa also operates a tape-robot which is accessible both via the Tivoli Storage Manager (for backup and archives) and via

dCache [46]. The latter provides a POSIX-like interface called "PNFS" (for Perfectly Normal FileSystem) which is mounted to the CDF access-node `cdf.fzk.de` at the path `/grid/fzk.de/mounts/pnfs/sam`. This interface allows to access the internal dCache database via normal Unix commands, i.e. sub-directories can be created via `mkdir`, files can be listed via `ls` and removed via `rm`, etc. To copy files from and to dCache the special program `dccp` has to be used. To integrate this into SAM, a valid tape-location has to be created below the above path using the command `samadmin add tape location` [5]. Then a new `sam_cp` Python class has to be written which uses the `dccp` command to access dCache. It is advised to put all new Python code in a separate location which is not used by any ups/upd products, e.g. `/home/sam/LocalPythonPath`. This new class is called `sam_gridka_dccp` and is stored in the file `LocalSamCpClasses.py`. To use this new method, it has to be added to the dictionary `KNOWN_SAM_CP_CLASSES` described above such that the relevant part of the file `sam_cp_config.py` reads now:

```
import SamCpClasses
import LocalSamCpClasses

 [...]
KNOWN_SAM_CP_CLASSES = {  'dcache'           : SamCpClasses.SamDcache,
                          [...]
                          'sam_gridftp'      : SamCpClasses.SamGridftp,
                          'dcache_gridftp'   : SamCpClasses.SamDcacheGridftp,
                          'sam_gridka_dccp'  : LocalSamCpClasses.SamGridKaDccp,
                          }
```

Now SAM needs to be configured to use this new class when accessing dCache at GridKa. As mentioned above, all tape locations are treated as 'enstore' in the domain compatibility map defined above, hence the static map is not flexible enough to distinguish between access to the tape-system at FermiLab (via `dcache_gridftp`[6]) and access to the tape-system at GridKa (via the newly created class `sam_gridka_dccp`). Instead, the "arbitrators" can be used to determine which protocol is to be used. The desired choice is:

- if accessing dCache at FermiLab, use `dcache_gridftp`
- if accessing dCache at GridKa, use `sam_gridka_dccp`
- use `sam_gridftp` in all other occasions

[5] It is advised to have several such locations each containing not more than a few ten-thousand files to avoid possible operational issues.

[6] CDF also operates a dCache system at FermiLab to access the files stored on tape. In contrast to GridKa, Enstore is used as underlying protocol.

site	CPU/GHz	disk/TB
CAF (FNAL)	3200	300.0
Italy	300	7.5
Korea	120	0.6
Taiwan	134	3.0
San Diego	280	4.0
Rutgers	100	4.0
Toronto	576	10.0
Japan	152	5.0
Spain	52	1.5
MIT	110	2.0
GridKa	135	25.0

Table 3.2: Computing resources currently being available for CDF users. The current numbers can be seen at `http://cdfkits.fnal.gov/DIST/doc/DCAF/`

The arbitrator is called `SamCpGridKaProtocolArbitrator` and implemented as a Python function in the file `sam_cp_protocol_arbitrator_gridka.py`.

To use these new classes, the following three variables need to be added to the SAM configuration via tailoring the product `sam_config`:

```
__ENV_PREPEND__PYTHONPATH=/home/sam/LocalPythonPath
SAM_CP_ARBITRATOR_MODULE_FILE=sam_cp_protocol_arbitrator_gridka.py
SAM_CP_ARBITRATOR_CLASS_NAME=SamCpGridKaProtocolArbitrator
```

The first variable includes the local directory in which the files are stored in the local Python search path. The second variable determines the filename holding the code of the arbitrator whereas the third variable sets the name of the arbitrator used. The actual implementation is discussed in appendix A.4.

3.5 Moving towards the Grid

Overview.— Several off-site resources have emerged globally, extending the computing power of the CDF collaboration. The various sites are in different stages of development and deployment. SAM is installed at each site; most sites focus initially on the generation of simulated events as this requires less storage. Table 3.2 summarises the amount of computing power and storage capacity currently being available.

Most of these sites have their resources dedicated to the CDF collaboration at 100 %. However, as the LHC grid efforts progress, most of these sites aim for a general cluster (with a setup similar to the one at GridKa), hence solutions assuming the exclusive usage of some resources will become increasingly difficult to operate.

The technical problems on the way to the Grid are very challenging — however, these are not the only ones on the way: The the different sites are operated by different institutions in different countries - each of them having different policies regarding security, user access, etc. Unfortunately, these policy are often mutually exclusive and a significant amount of time has to be spent to first make each participating party aware of this and then try to settle on a common policy, or at least to agree on a basic set of rules which the individual policies are based on.

The most promising approach to deal with these issues is to develop software which does not explicitly assume that certain services, protocols are available but aims for a general and modular design: The different parts of the software should communicate via defined interfaces, it should be possible to replace sub-components by others to accommodate a given local setup, policy, etc. Experience shows that short-cuts taken in the design often lead to dead ends and require substantial amount of re-design and development if new resources with new constraints have to be accommodated.

The following paragraphs give an overview the currently available options and further development of the CDF grid effort.

Distributed Analysis Facilities.— As a first step towards globally distributed computing the setup of the central analysis facility (CAF) at FermiLab was cloned at several locations (e.g. INFN in Italy, University of Toronto, Rutgers University, University of Japan, University of Taiwan). SAM is used at these sites for data-access users can submit jobs to these clusters via the `CafGui` also used for job-submission at the central farm. As most of these sites do not (yet) have extensive storage capacities to provide datasets for analysis, these sites were mainly used for simulation. Although this approach provides rather straightforward access to off-site facilities it has amongst other others the following drawbacks:

- All nodes have to be in a public network as `Kerberos` is used for authentification and authorisation and the machines have to be part of the `FNAL.GOV` realm
- `CAF` software needs to be installed on each computer which is only possible if the machines are dedicated to CDF use.
- Each user needs his own queue in the batch-system.
- As no grid-like software is involved issues like global scheduling, match-making, etc. discussed in the beginning of the chapter are not taken into account, e.g. fair-share is ensured only on a per-site level by the batch-system.

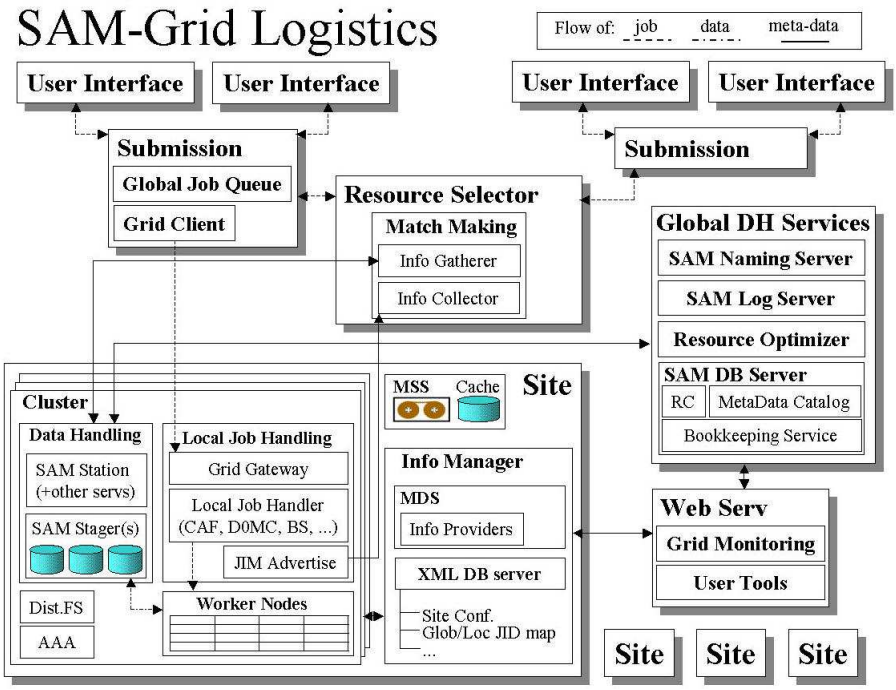


Figure 3.4: Overview of the JIM infrastructure, logistics and integration with SAM [53].

JIM.— JIM is an abbreviation for "Job Information and Managment" and developed together with SAM as part of the "FermiGrid" project. JIM provides global monitoring of participating sites (e.g. free resources, status of running or pending jobs, etc.) and used SAM to determine which (part of a) dataset is already cached at which site. The user provides an "input sandbox" and a job-description file. The input sandbox contains the executable and steering files which are needed to actually run the job on the worker node, the job-description file is evaluated by the Grid software and contains information about the required input files, etc. The JIM client software sends this information a global "match-maker" based on the Condor scheduler [54] which determines where the job should be executed. As a first approach the job is executed where most of the data is already available as the import of data typically takes longer than the duration of the job. Once the so-called "execution site" (i.e. the site where the job is going to be executed) has been determined, the input sandbox is sent there and submitted to the local batch-system. The output of the job can either be stored directly in the SAM system or can be packed up in the so called the output-sandbox which is then sent back to the user or copied to a location accessible via the internet. As users do not directly submit their jobs to a specific cluster, global

fair-share is ensured by the match-maker. Furthermore, users do not need access to the individual sites as JIM provides together with SAM the interface between the Grid and the Fabric. However, the main focus of this project has shifted towards generating simulated events for the DØ collaboration in the past years. Hence at present generic user analyses are beyond the scope of the effort.

Other approaches.— Several other ideas have been presented to utilise computing farms. A quick way to incorporate worker-nodes into an existing setup would be to use the "glide-in" mechanism of the Condor batch system: A special executable is started on a worker node which then contacts a certain Condor scheduler and advertises itself as a free resource accepting executables from the scheduler. Workernodes used this way do not have to be in the same local network as the scheduler, e.g. this mechanism would allow to extend the central analysis farm (CAF) at Fermilab by worker-nodes located in Italy. However, so far only a prototype exists and operational issues arising from the fact that the "cluster" may be spread over several institutions will have to be evaluated. Also, as this mechanism just extends existing computing farms, it can only be an intermediate step on the way to the Grid.

Another option would be to develop a (lightweight) interface to the LHC Grid, i.e. the Grid part of the job management (such as user authentification, match-making, submission to the local batch-system, etc.) would be handled by the LHC Grid software whereas SAM would would be used for data-handling. This way clusters shared with LHC experiments (e.g. GridKa or similar farms like at INFN in Italy) could be easily made available to the CDF community as the LHC Grid software has to be operational anyway. Furthermore, only a single further computer acting as SAM station and CDF software server would be needed to make these farms usable for CDF. However, as this option is presently begin discussed, a working prototype does not yet exist.

3.6 Discussion and concluding remarks

The large amount of data taken by the CDF experiment, the regular reprocessing of raw data and the demand for a huge amount of simulated events result in a need of computing power which cannot be provided by a single big computing farm. This has lead to the decision to provide a significant amount of computing power distributed around the world and make it available to the collaboration using the Grid. As CDF is a running experiment which has already taken lots of good physics data each centre has to remain fully operational at all stages of the development. Many (esp. the smaller) sites have chosen to dedicate (part of) their resources exclusively to CDF. This allows the installation of a clone of the central farm "CAF", thus enabling all CDF users to make use of this facility. However, as there is then no interaction with

other parts of the computing farm, synergy effects resulting e.g. from the collaboration with other experiments cannot be used and as no standard grid tools are used these farms are incompatible with future developments such as the possible interface of LHC Grid with CDF. Owing to the size of disk storage provided, most of these sites focus on generating simulated events rather than user analysis. The big farms at Rutgers and Toronto are also mainly used for simulation as they have mainly taken over the responsibility to generate the "official" samples.

GridKa on the other hand provides a significant amount of both computing power, disk and tape storage. This provides the opportunity to build a prototype of a regional analysis cluster and test the new developments with real users in a production environment. As several components of the Grid such as global job broker, etc. are still being developed, the GridKa cluster can for now only be used by the German CDF group. On the other hand, many potentially severe problems have been discovered at a very early stage. The inclusion of real world users doing analysis in the development and deployment phase of the Grid software has led to the development of many features which were not in the original design but have been proven to be essential in the daily work, an example is the successful integration of dCache read-/write-pools into SAM: Tape access is totally transparent to the user, i.e. the user does not notice (or even know) that SAM needs to access the tape-robot in the background. GridKa is currently the *only* CDF site in the world where this access mode has been deployed. Multiple other key functionalities now routinely used have been made available to CDF through this work, e.g. storing metadata information for new files in the database, storing files on the central Fermilab tape archive via SAM, etc. These functionalities are of vital importance for operating the production farms on-site Fermilab which have recently been upgraded to use SAM.

Figure 3.5 illustrates the successful and stable operation of the GridKa cluster: The cluster has been extensively used in the past year by the Karlsruhe CDF group and processed around 500TB of data, where up to 5TB have been processed on a single day. This corresponds to $\approx$ 20 % of all data being processed at the central farm (CAF) by all CDF users. GridKa has been been the only operational off-site computing facility for almost two years and is still the largest off-site resource being used in analyses (now being closely followed by the Italian effort).

The concept of a shared cluster has proven to be very efficient as all computing resources can be used by each experiment leaving no computer idle while jobs are still pending.

The amount of data processed by the Tevatron experiments is of course much lower than what is expected from LHC where data in the order of few Petabytes are produced each year which need to be copied to the various Tier centres for (re-)processing and analysis. However, extrapolating the amount of data processed by CDF and DØ now and taking into account the numerous efforts by the LHC grid computing community,

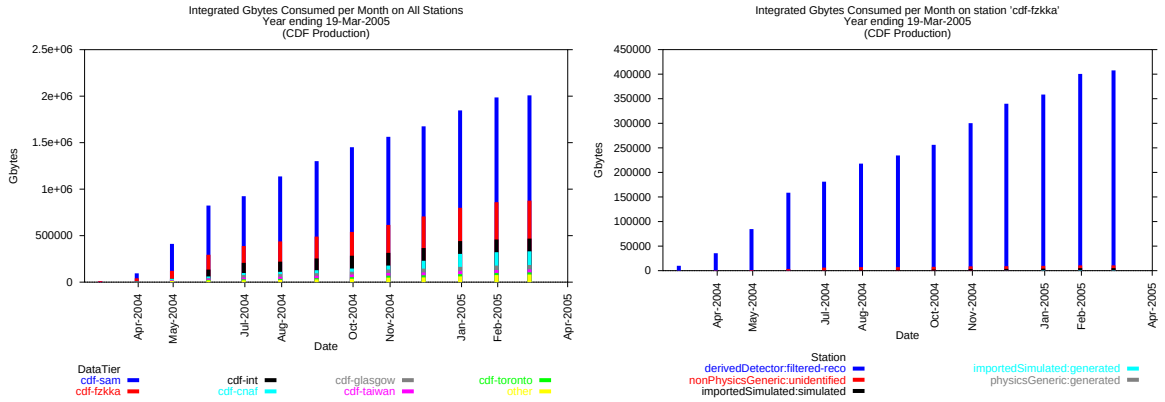


Figure 3.5: Data in GByte processed by SAM in the past year. The plots show the integral of the amount of processed data. The left plot shows the total amount of data globally processed by all stations. The different colours illustrate the contributions from the various stations: The central station `cdf-sam` (in blue) has processed most data, followed the GridKa station `cdf-fzkka` (in red). The third biggest "consumer" is the station `cdf-int` used for deployment tests at Fermilab. All other stations have processed significantly less data. The right plot illustrates the integral of the processed data at the GridKa station `cdf-fzkka` in GByte. The different colour codes correspond to different type of data, e.g. processed data taken by the CDF detector (in blue). The current plots are available at: `http://cdfdb-prd.fnal.gov/sam_local/PlotsAndStats/ConsumptionPlots/SamConsumptionPlots.html`

processing the LHC data seems feasible - though it will be a challenging task keeping the big centres running to retain an acceptable efficiency.

Chapter 4

Electron identification with Neural Networks

4.1 Introduction

Overview.— The identification of electrons plays a vital role in many physics analyses, e.g. semi-leptonic B or D hadron decays. In particular, the lightest charmonium (J/ψ, a $c\bar{c}$ state) decays with the same rate to $\mu^+\mu^-$ and e^+e^-. The J/ψ is used in many exclusive analyses such as $B^\pm \to J/\psi K^\pm$ or $X(3872) \to J/\psi\pi^+\pi^-$.

However, identifying electrons in the hadronic environment of a $p\bar{p}$ collider is a very challenging task. This is illustrated by figure 4.1 which shows a typical event recorded with the CDF detector. The event has been taken from the data-stream of the dedicated $J/\psi \to e^+e^-$ trigger. Only a few percent of all tracks in an event are electrons or positrons, most tracks are pions produced e.g. in the fragmentation process or interactions with the beam remnants after a hard scattering process.

Because of these challenges only very few analyses use electrons, for example, if the J/ψ is reconstructed as an intermediate particle in the decay chain, only the decay to muons is usually considered. Including the electron channel in these analyses can reduce the statistical error due to the higher number of reconstructed particles. This work focuses on the identification of "soft electrons" which are characterised by their small transverse momentum p_t of typically only a few GeV/c.

Development of a general electron ID package.— Up to now, mainly two methods are being used in the CDF collaboration to identify electrons: The `SoftElectronModule`[55] pursues a cut-based approach. This method achieves a rather high purity (i.e. only few of the identified electron candidates are not electrons) - but has only a mediocre efficiency, i.e. it "misses" many of the electrons. The `SoftElectronTagger` [56] uses likelihood methods for the electron identification. However, the latter is not a gen-

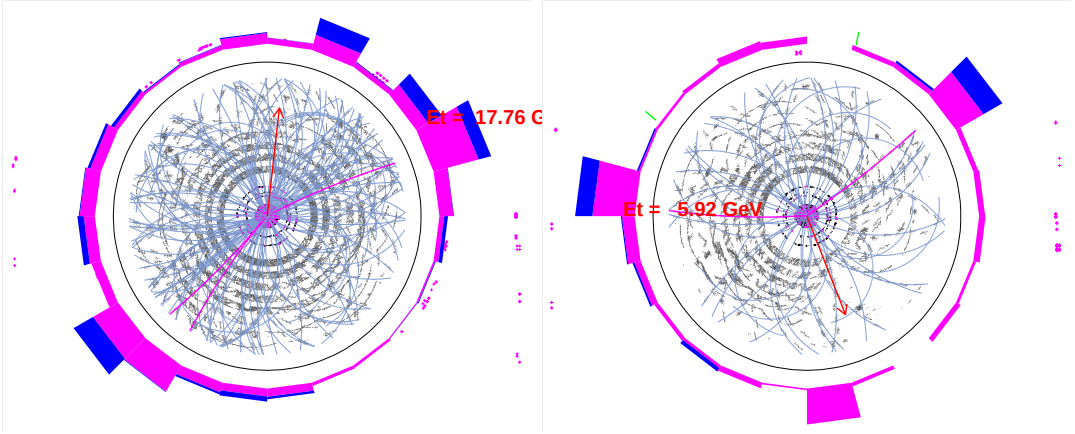


Figure 4.1: Typical event recorded with the CDF detector taken from the dedicated $J/\psi \to e^+e^-$ trigger. The block-like structures on the outside of the graphs illustrate the energy deposited in the calorimeters, the central part of each figure shows each found track and the reconstructed hits in the drift-chamber and silicon vertex detector.

eral tool but has been optimised to identify semi-leptonic B hadron decays and is embedded in the structure of the B_s mixing analysis.

This situation motivated the development of a "general purpose" toolbox for electron identification which is independent of a given analysis framework and provides multiple interfaces to allow efficient and easy integration in the user's analysis code. So far, it is used to reconstruct the exclusive channel $X(3872) \to J/\psi\pi^+\pi^-$ with $J/\psi \to e^+e^-$ and to identify semi-leptonic B hadron decays. It is planned that the toolbox will become part of a larger project which analyses B hadrons in a more inclusive approach following the idea of the `BSAURUS` expert system [57] developed at `DELPHI` at `LEP`.

Neural network based approach.— Using neural networks for the identification of electrons has several advantages as discussed below. An earlier study [58] has shown the high potential of this approach.

Neural networks are superior in several ways: They make use of the correlations between the input variables and are able to learn non-linear dependencies both between the the input variables and the training target and among the input variables themselves. Furthermore, they are able to handle variables with a default value, i.e. variables which are not filled for each candidate. For example, it can happen that (parts of) the calorimeter cannot be read out since there is no activity. In conventional analyses like cut-based or likelihood based approaches precuts have to be applied to these variables such that they are filled with detector information for each candi-

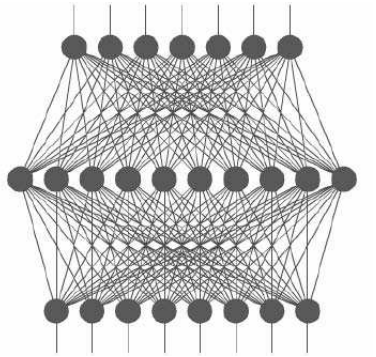


Figure 4.2: General topology of a neural network: The nodes ("neurons") are arranged in layers and connected with each other by weights.

date analysed. However, this often results in a significant loss of efficiency, especially if several such variables are involved. These precuts also remove information since a given variable may have a default value more often for background than for signal, etc. A further advantage of neural networks is that they allow a flexible choice of the working point: Since the network output is a continuous function between e.g. 0 and 1 (when used for classification), purity (i.e. the fraction of correctly identified particles) and efficiency (i.e. the number of particles above a given cut on the network output compared to all particles) can be determined for each cut on the network output. The ideal working point is the one with the closest distance to 100% purity with 100% efficiency - however, depending on the analysis different choices (e.g. higher purity) may be of advantage. Finally, neural networks do not discard information, they provide an output value for each candidate.

4.2 The NeuroBayes® neural network package

General remarks.— The (human) brain is far superior in finding patterns in a high-noise environment, adapting previously learned relations, etc. than today's computers. One of the key features of the brain is its ability to learn and adapt to new situations. In a very simplified picture, the neurons in the brain "fire" if the stimuli from the other connected neurons exceeds a certain threshold. Neural networks are derived from this basic principle: The neurons are replaced by interconnected nodes as illustrated by figure 4.2. These nodes are arranged in layers which are called input layer, hidden layer and output layer. The input layer has one node for each input variable given to the network, additionally a so-called bias-node is often used. There is no clear rule for the number of nodes in the hidden layer; the network has to have sufficient freedom to learn

all important features - but not enough to pick up statistical fluctuations. The number of nodes in the output layer depends on the performed task: For classification (a yes-or-no decision) one node in the output layer is required. The connections between the nodes are called "weights". The size of the weight determines the importance of the corresponding connection.

The basic working principle is then described by:

$$x_j^n = g\left(\sum_k w_{jk}^n \cdot x_k^{n-1} + \mu_j^n\right) \tag{1}$$

where $g(x)$ is a sigmoid function and μ_j^n describes the bias-node. Thus the output of node j in layer n is given by the weighted sum of the output of all nodes in layer $n-1$. Network training (or learning) is done via minimising a loss-function by adjusting the weights such that the actual network output is as close as possible to the desired output.

NeuroBayes®.— NeuroBayes® [59] is a sophisticated neural network package which deploys second generation algorithms and methods to ensure a fast and robust training. The use of regularisation techniques based on Bayesian statistics almost eliminates the risk to get stuck in local minima during the training process. The extremely sophisticated and fully automated preprocessing makes the network package very robust. NeuroBayes® has been developed at the University of Karlsruhe by Prof. Dr. Michael Feindt. The same methods can also be applied outside physics, e.g. to optimise car insurance conditions. These applications have lead to the spin-off company Phi-T (`www.phi-t.de`).

The package is divided into two components: The Teacher and the Expert. The Teacher takes the input data together with the training target from the user. After the preprocessing the network is being trained. The result of the training is stored in the Expertise which is read in by the Expert part of NeuroBayes® which is then used to analyse the data. The NeuroBayes® network can be used both for classification and for shape reconstruction. In classification tasks, NeuroBayes® is used to discriminate between two classes, e.g. is the particle with given calorimeter data, etc. an electron or not. When used for shape-reconstruction, NeuroBayes® reconstructs a complete probability density distribution which can be used in subsequent analyses. For example, NeuroBayes® can be used to reconstruct the energy of an inclusively reconstructed B-hadron. The NeuroBayes® Expert computes the full probability density distribution of the reconstructed quantity for each individual candidate analysed. Then e.g. the median of this distribution is taken as an estimate of the B energy. The 15% and 84% quantiles (which correspond to $\pm 1\sigma$ in case of a Gaussian distribution) can be used as an estimate of the error, to enhance well reconstructed candidates, etc.

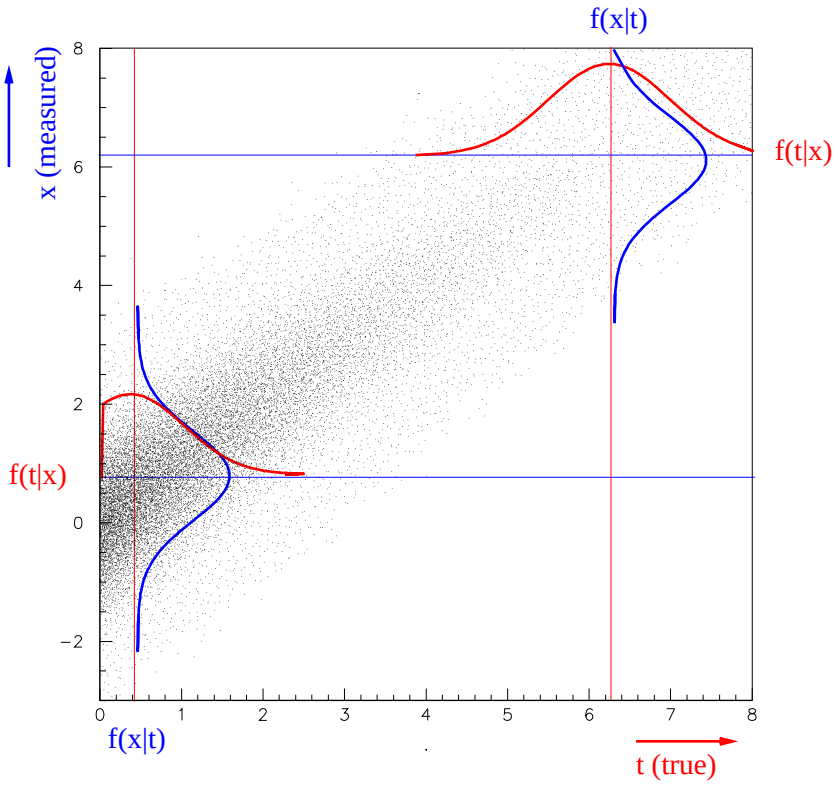


Figure 4.3: The plot illustrates the effect of using *a priori* knowledge using Bayesian statistics.

The Bayesian Approach.— Using Bayes' theorem, NeuroBayes® takes into account *a priori* knowledge. This is illustrated with the example of measuring the lifetime of a particle. The true lifetime of a particle is described by an exponential of the form $f(t) = \frac{1}{\tau}e^{-t/\tau}$ which cannot be negative. Measured quantities, however, can become negative due to resolution effects. This is illustrated by figure 4.3: The distribution of the true lifetime $f(t)$ (on the the x-axis) has been smeared by a Gaussian distribution to simulate resolution effects resulting in the "measured" distribution $f(x)$ (on the y-axis). In the classical approach measurements are taken as an estimate of the true quantity, i.e. the conditional probabilities are treated as being identical: $f(x|t) = f(t|x)$. This is acceptable if the measurements are taken far away from physical boundaries with good resolution (upper right hand side of figure 4.3), as both quantities are approximately Gaussian distributed. However, close to physical boundaries (lower left part of figure 4.3) this approach is wrong: Whilst the true value cannot move into the unphysical region, measurements can. By using Bayesian statistics, this is taken into account such that NeuroBayes® will never produce unphysical results.

A further advantage of the Bayesian approach is that in classification tasks the output of NeuroBayes® can be directly interpreted as a Bayesian *a posteriori* probability, e.g. if the network output is 0.8 this means that the considered candidate has an 80% probability to be a signal event (if the training was performed such that signal events have a desired network output of 1 whereas background events have a network output of 0).

Preprocessing.— A key feature of NeuroBayes® is its automated and robust preprocessing which optimally prepares the input variables for the subsequent training. The importance of this step can be illustrated by the following example: A hiker wants to find the deepest valley in the Swiss alps starting from the top of a high mountain. After having descended into a valley it is impossible to determine whether the next valley would be deeper. The preprocessing steps "guide" the network to a good starting point.

NeuroBayes® has many preprocessing options, the full list is given in [60]. These options can be divided into two groups: One set of options operates on *all* input variables, e.g.

- normalisation and linear decorrelation of the input variables
- automatically recognise binary and discrete variables
- construct derived input variables such that the first variable contains all linear information about the mean, the second about the width, etc.

Other preprocessing options operate on a specific input variable, e.g.

- special treatment of variables with default values or a δ-function
- perform a regularised spline-fit on the input variable
- special treatment for ordered and unordered classes[1]

Using NeuroBayes® with C++.— The NeuroBayes® Teacher and Expert are implemented in Fortran. To allow a large variety of users to use NeuroBayes® interfaces have been developed to `ASCII` files, Fortran with `CERNLIB` and `PAW` and `C++` with `ROOT`. The latter interface enables CDF users to use NeuroBayes® both from the `AC++` analysis framework and from within own `C++` or `ROOT` macros.

Since the NeuroBayes® Teacher internally uses common-blocks to pass information between the various subroutines, the interface to it has been implemented as a singleton class. This insures that only one instance of the Teacher is run at a time.

[1] Ordered classes are lists where the position in the list has a special meaning, e.g. no cut, loose cut, tight cut, whereas in unordered classes the position in the list has no further meaning.

The main routine of the NeuroBayes® Expert has been implemented as a native `C++` class, which allows to run several instances of the Expert simultaneously. Further technical details about the `C++` interface can be found [61].

NeuroBayes® has been made available to the CDF community via ups/upd [62] products as described in detail in appendix D.

4.3 Electron identification toolbox

Overview.— The electron identification has been developed as a general purpose tool which does not assume a specific framework or setup of the user's code. Furthermore, it is self-contained, i.e. it does not require that the user already has to process some information and pass it on to the toolbox but provides multiple interface allowing easy and efficient integration in the analysis code. The toolbox focuses on the central region of the CDF detector defined by $|\eta| < 1$ where the pseudo-rapidity is defined by $\eta = -\ln(\tan(\theta/2))$. This region is used in most b physics analyses. However, there is no implicit assumption that all particles have to be in this region, the toolbox can be extended easily, e.g. by including information from the plug-calorimeters and retraining the networks. Furthermore, the toolbox focuses on the identification of soft electrons with a very small minimal transverse momentum as these play an important role in charm and bottom-quark physics.

The toolbox consists of multiple networks: The `KappaNet` identifies electrons via the change in curvature of the fitted helix when the particle transverses material and radiates off Bremsstrahlung. The `SENet` combines information from the `KappaNet`, calorimeters, time-of-flight detector and dE/dx measurements from the central drift chamber (COT) for an optimal identification of electrons. The conversion of photons into electrons and positrons is the source of a significant background. The `ConversionNet` identifies these electrons which can then be removed from the analysis chain.

Two sets of networks have been trained: one set of networks has been trained for all tracks with a minimal transverse momentum of $p_t > 2$ GeV/c. The other set of networks does not require any minimal p_t value. This choice is based on the main physics use cases: The dedicated $J/\psi \to e^+e^-$ trigger requires the two trigger tracks to have a minimal transverse momentum of $p_t > 2$ GeV/c. On the other hand, there is no such criterion for electrons originating from semi-leptonic B hadron decays, i.e. when reconstructing these decays all possible tracks are used.

Obtaining the calorimeter information.— Two methods exist to retrieve the needed calorimeter information: The `SoftElectronModule` [55] tries to retrieve calorimeter information for each track in the CDF default track selection. Tracks associated with

at least some calorimeter information from either the CES, CPR or CEM detector are accepted for further analysis and are copied to a new collection called SoftElectronColl. Using this collection requires the SoftElectronModule to be run prior to the own analysis code. In detail, the module requires that the candidate track can be matched to a calorimeter cluster in the CPR or the CES detector. Furthermore, the track is extrapolated to the central electromagnetic calorimeter (CEM). The candidate track is accepted if a calorimeter tower associated with this track lies within the limits in pseudo-rapidity η where calorimeter clusters can be found and a minimal amount of energy is deposited. By default, the SoftElectronModule requires that at least 2 GeV are deposited in the calorimeter. Less than 10% of all CDF default tracks with transverse momentum $p_t > 2$ GeV/c fail these requirements and are hence not considered in the subsequent analyses. For candidate tracks with no requirement on the transverse momentum (i.e. $p_t > 0$ GeV/c), about 45 % of all tracks pass the above requirements. Lowering the threshold on the minimal deposited energy to 0.8 GeV results only in a few more percent of accepted tracks. If the threshold is lowered as low as 0.1 GeV, about 60 % of all CDF default tracks are accepted, lowering the threshold even further does not result in a higher number of accepted tracks.

Alternatively, the electron toolbox can start from the calorimeter information stored in the CdfEmObj collection. This collection contains the output of the clustering algorithm used to analyse the response of the calorimeter cell. Reconstructed tracks associated with calorimeter activity are accessible via the collection as well. The calorimeter code generating the CdfEmObj collection is already run during the data processing. However, the precuts chosen there suppress much background - but also reject a significant part of signal candidates. It is therefore necessary to recreate this collection with much lower thresholds. Unfortunately, this results in a rather large increase of execution time of the program.
All information acquired by the electron identification package is stored in a C++ map which allows efficient access to all properties, e.g. for filling histograms or ntuples.
Further technical information about the user interface can be found in appendix B.1.

Training sample.— All neural networks have been trained using simulated events. The neural networks used to identify electrons via the change in curvature (KappaNet), the soft-electron identification network (SENet), as well as conversion identification network (ConvNet) were trained using generic $p\bar{p}$ events generated by Pythia [63]. Events with no $b\bar{b}$ or $c\bar{c}$ pair have been discarded. All relevant production processes for heavy quark flavours (flavour creation, flavour excitation and gluon splitting) have been taken into account. Figure 4.4 shows the Feynman graphs for these processes. The subsequent decay of the produced particles was handled by EvtGen [64] before the events were passed through the full detector and trigger simulation. Further details about the production of the sample can be found in [65]. The final sample consists of

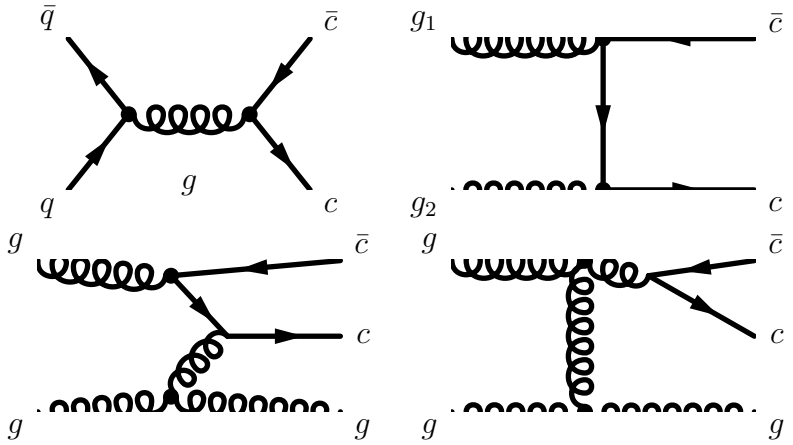


Figure 4.4: Important production mechanisms for heavy quarks ($c\bar{c}, b\bar{b}$) at the Tevatron: flavour creation via $q\bar{q}$ annihilation (top left), flavour creation via gluon fusion (top right), flavour excitation (bottom left), gluon splitting (bottom right)

$\approx 113,000$ events, 75% of the available statistics has been used to train the networks, the control-plots such as the purity versus efficiency plots in this chapter have been obtained by applying the trained network to the remaining 25%. Generally speaking, the distributions of the input variables and the output of the trained network between data and and simulated events agree quite well as shown in appendix B.7. However, the behaviour of several variables in the simulation which are used as input to the soft electron identification network show larger deviations from the respective distribution observed in the data. Since the NeuroBayes® neural network can handle weighted events, an event weight has been constructed based on the quantities χ^2_z(CES), Δ_x(CES), Δ_z(CES) and E_{strip}/E_{wire}(CES) which has been used in the final network training.

Electron identification via change in curvature.— The energy loss of particles due to Bremsstrahlung in material is proportional to $1/m^2$, thus light particles such as electrons and positrons will lose a significant part of their energy due to this process. The particles move on helices with a curvature proportional to the inverse of the momentum due to the magnetic field inside the CDF detector. The energy loss results in a change of curvature as illustrated by figure 4.5. The figure shows an electron which is approaching a material layer from below. While traversing the material, a Bremsstrahlung photon is radiated off and the loss of energy results in a more strongly bent trajectory. The `KalmanFit` tracking algorithm employed in the CDF software [66] is able to determine the curvature at different measurement layers of the silicon vertex detector and the inner wall of the central drift chamber when fitting the track trajectory either from the inside-out direction (i.e. from the interaction point outwards) or outside-in (i.e. towards the interaction point). The measurement of

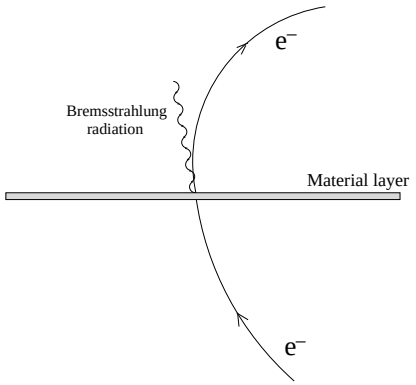


Figure 4.5: The loss of energy due to Bremsstrahlung results in a change of curvature of the trajectory of the charged particle in the magnetic field.

the change in curvature at several measurement layers is combined with additional information in a neural network referred to as `KappaNet`, which implies that only tracks found by the `KalmanFit` can be used as input to the `KappaNet`. The full list of input variables used is given in appendix B.2. Since the `KappaNet` does not depend on calorimeter information, it does not require that the calorimeter clustering module or the `SoftElectronModule` has been run. Consequently, it can be applied to each candidate in the CDF default track collection.

Two networks have been trained which differ in the choice of the minimal transverse momentum: One network requires $p_t > 2$ GeV/c, whereas the other considers all reconstructed tracks, i.e. $p_t > 0$ GeV/c. The resulting purity-efficiency plot is shown in figure 4.6. Purity and efficiency are defined as:

$$\text{efficiency} = \frac{\#\text{ true } e^{\pm} \text{ past cut}}{\#\text{ true } e^{\pm}} \tag{2}$$

$$\text{purity} = \frac{\#\text{ true } e^{\pm} \text{ past cut}}{\#\text{ candidates past cut}} \tag{3}$$

The plots show that the purity of identified electrons can be increased significantly by approximately a factor four. However, this is only achieved with a low efficiency of around 10%. Rejecting background by cutting on the network output will remove most signal candidates. However, since this variable is well correlated to whether the particle is an electron/positron or not it can be used as an input variable in a further neural network.

Identification of soft electrons.— Soft electrons are identified by their distinct signature in various parts of the detectors. The electromagnetic calorimeters and the pre-shower

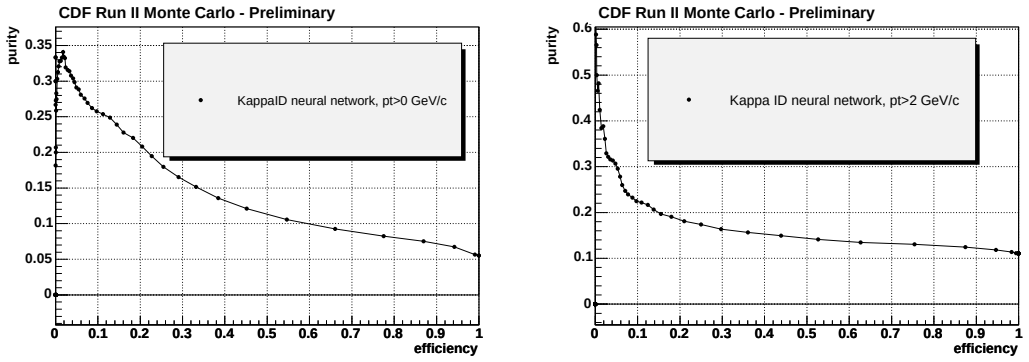


Figure 4.6: The plots show the achieved purity vs. efficiency for the `KappaNet`. The left plot shows the result for all tracks with $p_t > 0$ GeV/c, whereas the right plot illustrates the result when requiring a minimal transverse momentum $p_t > 2$ GeV/c.

detectors CEM (central electromagnetic calorimeter), CPR (central pre-radiator) and CES (central strip chamber) play a vital role in the identification process. However, especially for low momentum particles, measuring the specific energy loss in the central drift chamber (COT) and the time-of-flight (i.e. the time between the primary interaction and the arrival in the `ToF` detector) enhance the discrimination power of the neural network significantly.

The development of the soft electron identification network started using information from CEM, CES and CPR, following the approach in [58] and adding the output of the `KappaNet` as a further input.

The performance of the soft electron identification network is again evaluated in terms of purity and efficiency. The graphs are obtained by analysing all candidates in the `SoftElectronColl` collection obtained with the default values described in detail before. They are hence normalised to the number of all candidates which are associated with minimal information in the `CEM`, `CPR` or `CES` calorimeter. Again purity is plotted against efficiency, hence ideal point is (1,1) corresponding to 100% efficiency (all electrons are selected) at 100% purity (all identifications are correct). The left plot illustrates the obtained result for all candidates in the collection (i.e. no cut on minimal transverse momentum p_t), whereas the right plot shows the obtained discriminating power when requiring $p_t > 2$ GeV/c.

Figure 4.7 illustrates the improvement obtained by adding the output of the curvature change detection network (`KappaNet`) as a further input variable to the soft electron identification network. Each figure contains two graphs: The graph in black shows the purity and efficiency obtained for the neural network based on calorimeter information from the `CEM`, `CES` and `CPR` detector, the graph in blue shows the cor-

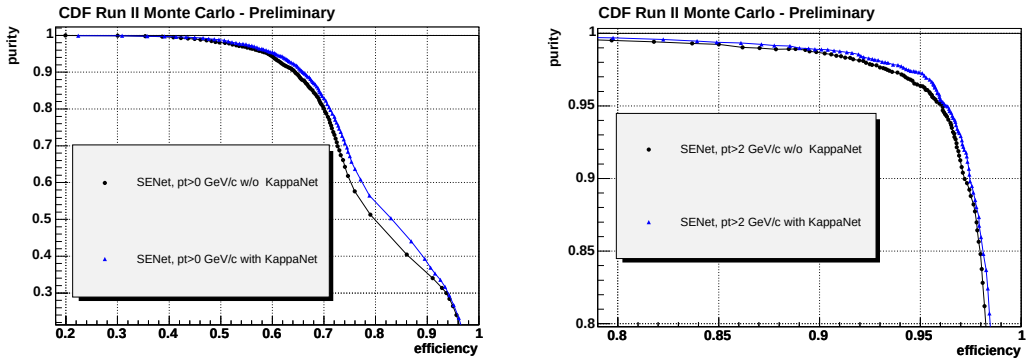


Figure 4.7: The plot shows the improvement achieved by adding the output of the `KappaNet` to the list of input variables of the soft electron identification network. The left plot shows the result for all tracks, the right plot when requiring a minimal transverse momentum of $p_t > 2$ GeV/c. Note the difference scales on the axis.

responding graph after including the output of the `KappaNet` as an additional input variables. Electrons with transverse momentum of $p_t > 2$ GeV/c can be identified efficiently with already high purity using the calorimeter information as illustrated by the right plot in the figure. Improvements to this network will be on the percent level as the purity vs. efficiency curve already is close to the ideal point of (1,1). The identification of the even softer electrons with no cut on p_t is much more difficult as shown by the left graph in the figure. Although a purity of $\approx 100\%$ can be achieved as well, this would require cutting at an efficiency of 50%, i.e. removing half of the candidates. Including the `KappaNet` yields a visible improvement.

Particles with not too high momentum (i.e. moderately relativistic particles) other than electrons lose energy in matter primarily by ionisation and atomic excitation. The mean energy loss is described by the Bethe-Bloch formula. Figure 4.8 illustrates the specific energy loss for different particle types in the momentum range of $\approx 0 \ldots 10$ GeV/c. While the energy loss of electrons is almost independent of the momentum, all other particles show a characteristic behaviour. By measuring the energy loss dE/dx in the central drift chamber (COT) electrons can be separated well from other particles in the region $p_t \lessapprox 2$ GeV/c.

To provide optimal input variables for the neural network the following approach has been taken: The Bethe-Bloch formula is a function of β and γ only if the conditions in the drift chamber (such as gas pressure, mixture, etc.) are constant as discussed in [68] and [69]. The "universal curve" extracted this way predicts the mean energy loss for a given momentum and mass hypothesis. The predicted and measured dE/dx values are combined to Z=log(dEdx(measured)/dEdx(predicted)) which is distributed

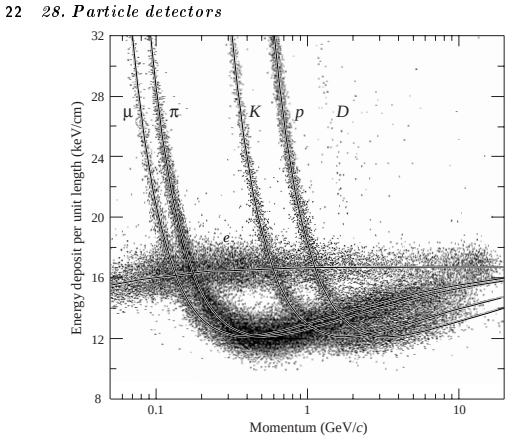


Figure 4.8: The plot illustrates the separation power achieved by measuring the specific energy loss dE/dx in the central drift chamber. The plot has been taken from the Particle Data Group [67]

according to a Gaussian and hence more easily treatable. Together with the *a priori* knowledge of the particle composition (11 % protons, 70% pions, 8% kaons, 4% muons and 7% electrons estimated using simulated events) a likelihood ratio is formed for each of these particles based on the measured and predicted energy loss [70]. These five variables are then used as input variables to the soft electron identification network. The enhancement achieved by this network is shown in figure 4.9. The separation power is improved significantly, most notably for the network not requiring a minimal transverse momentum (left part of the figure).

A further sub-detector sensitive to the particle type is the time-of-flight (ToF) detector as discussed in section 2.2. Figure 4.10 illustrates the separation achieved using ToF information using data. The time-of-flight measurement complements the dE/dx measurements in the central drift-chamber at low momenta. Pulls of the measured vs. expected time-of-flight are computed for several particle hypotheses (electron, muon, pion, kaon, proton) and used as input variables to the neural network. Figure 4.11 shows the improvement achieved by including these variables for the case of the network not requiring a minimal p_t. The network requiring $p_t > 2$ GeV/c contains these variables for consistency as well although the resulting enhancement is very small.

Figure 4.12 shows the final result in terms of purity vs. efficiency combining all information from calorimeters, curvature change, energy loss in the drift chamber and time-of-flight measurements. The cross in each plot reflects the result obtained using a standard cut-based approach [55]. The network based approach is superior to the cut-based approach in several ways: Using the neural network based approach, the purity can be enhanced at the same efficiency. More importantly, the efficiency can be

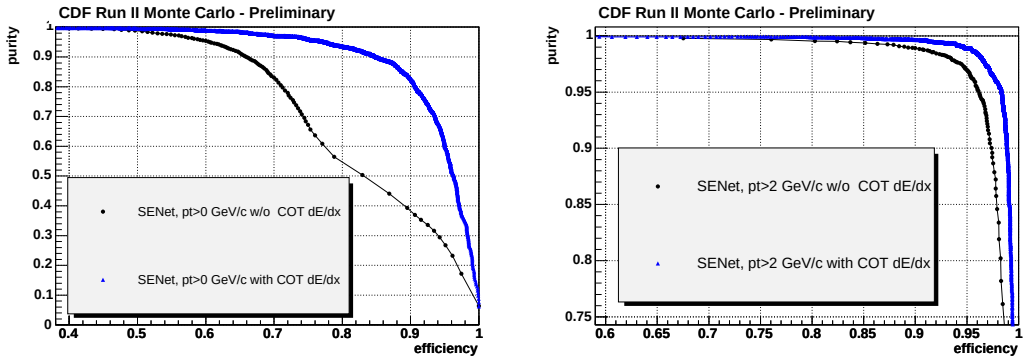


Figure 4.9: The plot shows the improvement achieved by including energy-measurements (dE/dx) in the central drift chamber to the soft electron identification network. The left plot shows the result for all tracks, the right plot when requiring a minimal transverse momentum of $p_t > 2\text{GeV/c}$. Note the different scales.

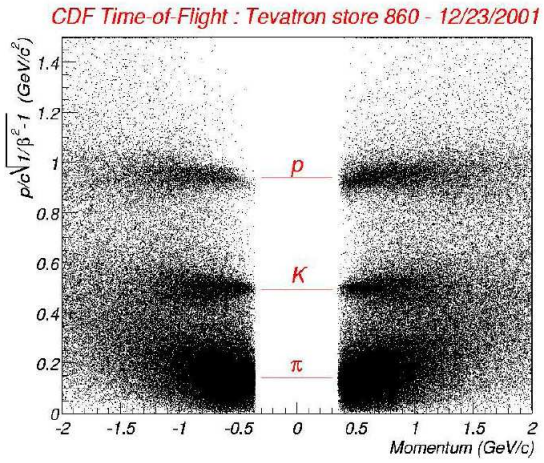


Figure 4.10: The plot illustrates the separation power of the time-of-flight detector between the main sources of background when identifying electrons: Pions, kaons and protons.

increased drastically by the neural network at the same or even higher purity obtained by the cut-based approach. When using all candidates with minimal calorimeter information (i.e. all elements in the `SoftElectronColl` collection) in the central region of the CDF detector, defined by the pseudo-rapidity $|\eta| < 1$, the efficiency can be increased by a factor of three with respect to the cut-based approach. Requiring

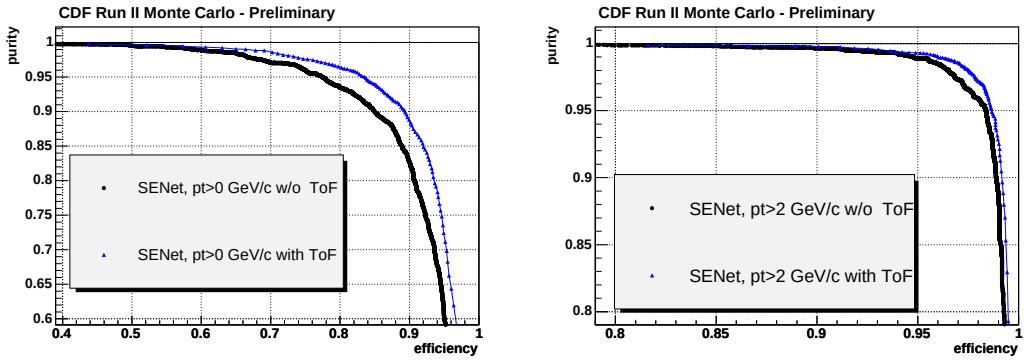


Figure 4.11: The plot shows the improvement achieved by adding the time time-of-flight measurement to the list of input variables of soft electron identification network. This further improves the network with no additional cut on $p_t > 0$ GeV/c, whereas only marginal (if any) improvement is achieved for $p_t > 2$ GeV/c due to the design of the TOF detector. Note the different scale in the right plot.

$p_t > 2$ GeV/c, the neural network yields a factor two better efficiency at the same purity as the cut-based approach. The triangle (red) in the figures correspond to the optimal cut on the neural network yielding the highest purity and efficiency, i.e. the smallest distance to the ideal point with 100% purity and 100% efficiency, as estimated from the simulation. At this point, the following purity and efficiency is obtained:

network	$p_t > 0$ GeV/c	$p_t > 2$ GeV/c
cut	-0.2	-0.1
efficiency	90 %	96 %
purity	92 %	97 %

The networks are optimally trained and and have the best possible discrimination power both in the region $p_t > 2$ GeV/c and in the very challenging region $0 \leq p_t \leq 2$ GeV/c.

Comparison to JetNet.— NeuroBayes is a very sophisticated neural network package as discussed in the beginning of this chapter. It provides many options to optimally preprocess the input variables and ensure an optimally trained network.

An often asked question is whether a similar performance can be obtained with other available networks. This has been tested by comparing the performance of the soft electron identification network (starting from the `SoftElectronCollection` with $p_t > 2$ GeV/c) when trained with NeuroBayes and with JetNet [71] using the ROOT interface from [72]. The same input variables and network architecture have been used,

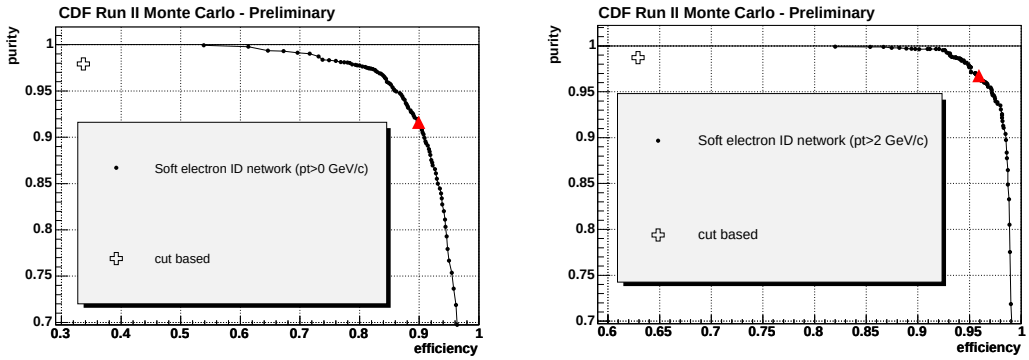


Figure 4.12: The plots show the finally achieved separation power in terms of purity and efficiency for the soft-electron identification network taking the weights for the input variables into account. The left plot shows the obtained result for the network not requiring a minimal transverse momentum, the right plot shows the result when requiring $p_t > 2$ GeV/c. The cross denotes the achieved purity and efficiency for the cut-based selection[55]. Note the different scale on the axis. The red dot illustrates the optimal cut on the neural network.

switching on the default values for the configuration parameters. The trained network based on JetNet performs significantly worse than the one based on NeuroBayes as illustrated by figure 4.13. However, it should be noted that this does *not* mean that a neural network based on JetNet cannot yield the same results as one based on NeuroBayes. The performance depends on many factors such as the complexity of the problem and the preprocessing of the input variables. It is possible that a comparable result can be achieved with JetNet if significant amount of work was put into manually preprocessing the input variables, which is done automatically by NeuroBayes.

Identification of conversion electrons.— Most electrons in a typical CDF event will not originate from decays of charm or bottom hadrons but from pair production which can only occur in the presence of a nucleus: $\gamma + (Z, A) \rightarrow e^+e^- + (Z, A)$. The photons originate e.g. from the decay $\pi^0 \rightarrow \gamma\gamma$ where the pion was produced in the fragmentation process. Conversion electrons are usually considered as background in the analysis as they don't originate from the physics process of interest (e.g. semi-leptonic B decays) and need hence to be removed. Usually, conversion electrons are identified by a series of cuts. To optimally exploit the available information a conversion identification network (`ConvNet`) has been trained. The `ConvNet` has been trained to distinguish whether or not the candidate is a conversion electron, i.e. whether it originated from a photon. A minimal cut on the soft-electron identification network is required to ensure that only good electron candidates are presented to the `ConvNet`.

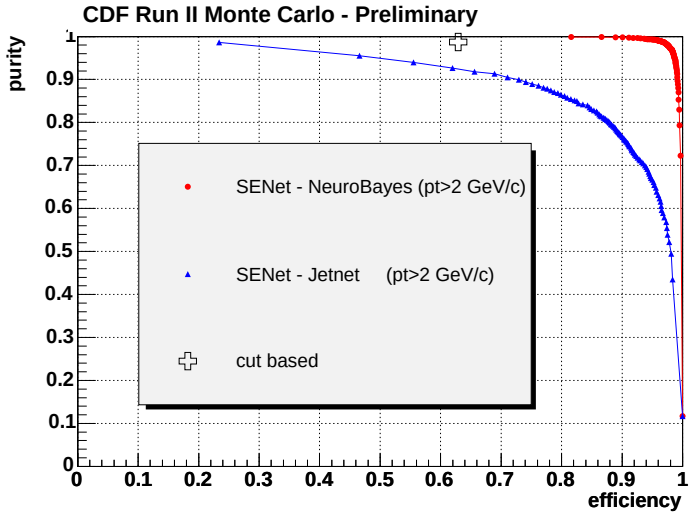


Figure 4.13: Comparison of the soft-electron identification network for $p_t > 2$ GeV/c between NeuroBayes and JetNet using the same network architecture.

The full list of input variables is given in appendix B.4. One of the input variables is the charge-signed impact parameter $q \times d_0$ which is (in absence of tracking resolution effects) positive for conversion electrons as suggested in [56, 73]. Figure 4.14 illustrates the different behaviour of this variables for electrons or positrons originating from a conversion or from the decay of a particle: The photon (originating e.g. from a π^0 decay at the primary vertex) interacts with the detector material and converts to a e^+e^- pair. The impact parameter d_0 itself is a signed quantity. In the case of the positron in the figure d_0 is positive, whereas d_0 is negative for the electron. By multiplying d_0 with the charge of the particle q, the combined quantity $q \times d_0$ has the same sign for electron and positron. This is not necessarily the case for electrons originating from a decay as illustrated by the right part of the figure.

Figure 4.15 shows the achieved purity and efficiency obtained by the conversion identification network as estimated by simulated events. The cross in the plots reflects the result obtained the standard approach based on sequential cuts. The curve illustrates the significant improvement achieved by the `ConvNet`.

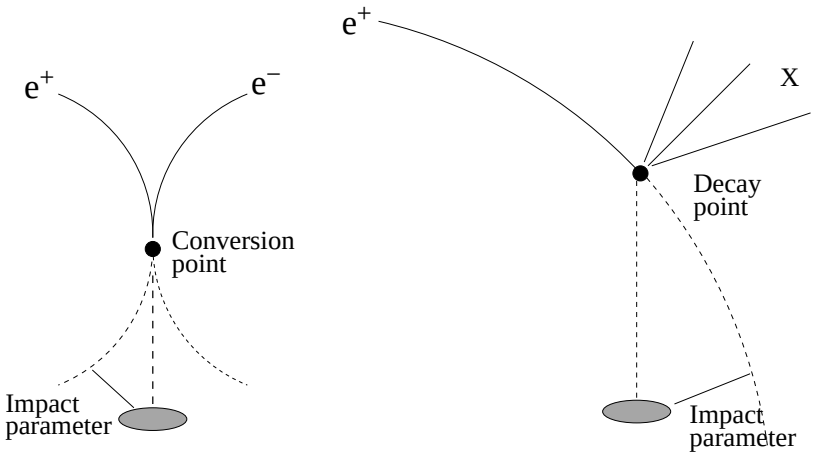


Figure 4.14: The figure illustrates the different behaviour of $q \times d_0$ for conversion electrons (left) and electrons originating from the decay of a particle.

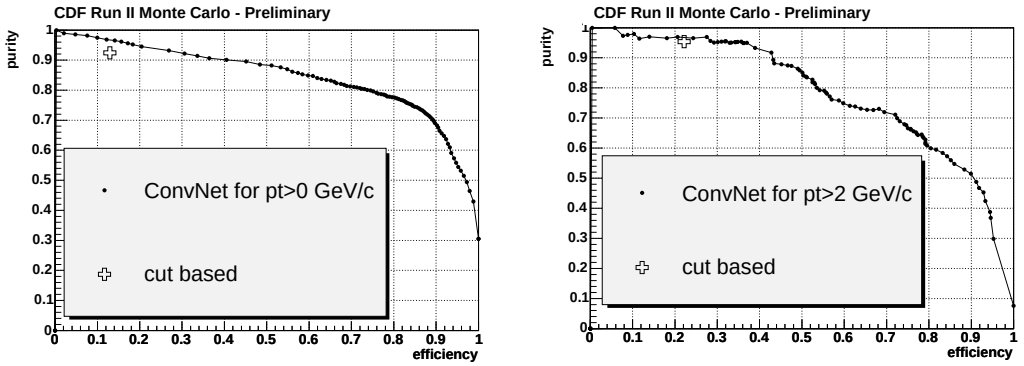


Figure 4.15: The plots illustrate the purity and efficiency achieved by the conversion identification network. The left plot shows the result for the case $p_t > 0$ GeV/c, whereas the right plot shows the result for the network requiring $p_t > 2$ GeV/c. The cross shows the result obtained with the default cuts.

Chapter 5

Applications of the electron identification toolbox

The identification of so-called soft electrons, i.e. electrons with low transverse momentum p_t, plays a vital role in many physics analyses. The method developed has been used in this work to obtain an (almost) pure sample of $J/\psi \to e^+e^-$. These reconstructed particles are then combined with two oppositely charged pions to observe the decay $X(3872) \to J/\psi\pi^+\pi^-$ for the first time in the channel $J/\psi \to e^+e^-$ at a hadron collider. In a second application, the electron identification toolbox is interfaced with the framework used in the CDF measurement of the B_s mixing frequency Δm_s resulting in a significant improvement in discriminating b from $\bar{b}$ quarks in semi-leptonic B meson decays.

The developed electron identification package will play an integral part in many further analyses concerned with semi-leptonic B meson decays or more inclusive reconstruction techniques inspired by the successful `BSAURUS` package [57] used in many DELPHI B physics analyses. Special attention has been paid to an intuitive and versatile user interface of the electron identification toolbox which allows easy integration with any analysis framework.

5.1 Reconstructing exclusive $J/\psi \to e^+e^-$

Overview.— Although the J/ψ decays with the same rate of $\approx 7\%$ both into muons and electrons most analyses focus on the channel $J/\psi \to \mu^+\mu^-$. The di-muon channel can be cleanly identified and triggered by requiring that in a given event two muons identified by the muon chambers originate from the same vertex. CDF deploys a dedicated $J/\psi \to e^+e^-$ trigger which is used in this analysis. However, this case remains much more challenging experimentally even when selecting only events passing

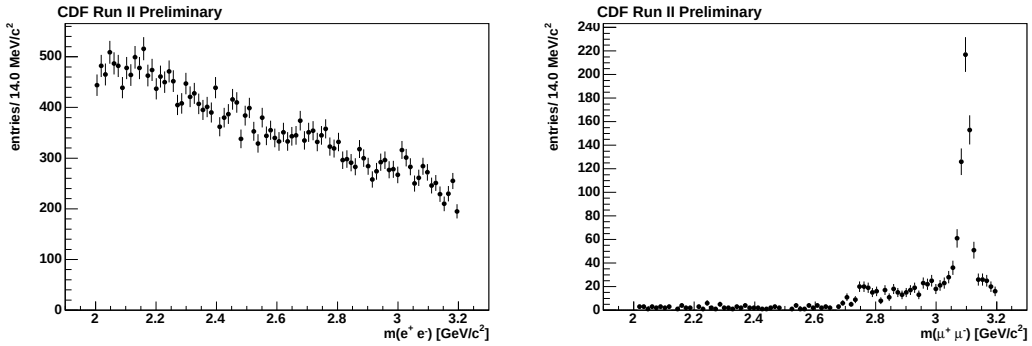


Figure 5.1: Invariant mass of the combination of all potential signal tracks. Left plot: One file from the $J/\psi \to e^+e^-$ trigger. Right plot: Corresponding plot, taking one file from the dedicated $\mu^+\mu^-$ trigger.

the trigger requirements as illustrated by figure 5.1: The left part of the figure shows the invariant mass spectrum of all potential signal tracks (one processed data file has been used). The $J/\psi \to e^+e^-$ signal at $m(J/\psi) = 3.096$ GeV/c^2 is not visible but 'buried' in the combinatoric background. The right plot in the figure shows the corresponding plot from the di-muon trigger in the same mass range. The J/ψ peak is clearly visible on a low background level.

Using `SENet` *to reconstruct* $J/\psi \to e^+e^-$.— Using the soft electron identification network discussed in detail above the background can be almost entirely removed as illustrated in figure 5.2. The figure shows the invariant mass spectrum of all identified e^+e^- pairs. Using the `SENet`, candidate tracks are selected with a purity of $\approx 99\%$ at an efficiency of $\approx 92\%$, as estimated by simulated events. The $J/\psi \to e^+e^-$ signal peak is clearly visible on a low background level. In contrast to the $\mu^+\mu^-$ case energy loss due to Bremsstrahlung plays a vital role for electrons due to their much lower mass. This effect results in a strong radiative tail as illustrated by the figure.

Suppressing background from conversions.— The background still present in figure 5.2 can be further reduced: It turns out that it consists mainly of conversion electrons, i.e. of cases where the `SENet` correctly identified the constituent particles as electrons, but either one or both originate from the conversion of a photon instead of the $J/\psi \to e^+e^-$ decay. These combinations can be rejected using the conversion identification network `ConvNet`. The cut on this network was determined such that the invariant mass spectrum of these candidates is almost flat and contains only a small resonant J/ψ signal, as illustrated by the left part of figure 5.3. To tighten this cut and efficiently remove as much background as possible, all J/ψ candidates are rejected where either

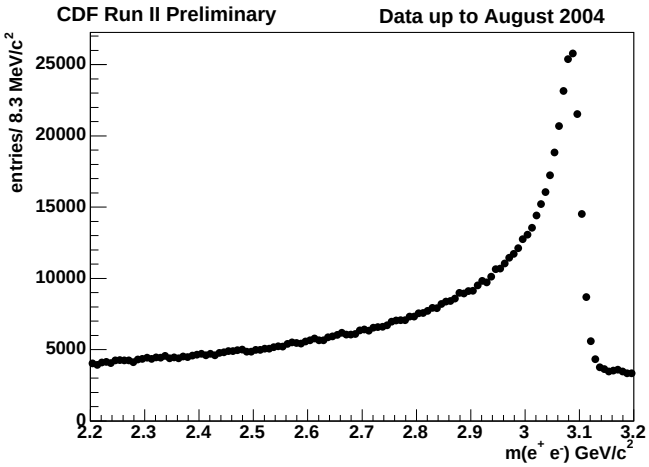


Figure 5.2: The plot shows the invariant mass spectrum of all identified electrons and positrons yielding a clear J/ψ signal. Note the extensive radiative tail as compared to the $J/\psi \to \mu^+\mu^-$ case. The ratio of the height of the peak divided by the level of background taken from the right side-band is ≈ 7.

the e^- *or* the e^+ is identified as a conversion electron by the `ConvNet`. The invariant mass spectrum of all rejected candidates is shown in the right part of figure 5.3: A small $J/\psi \to e^+e^-$ peak is visible on a large background level, i.e. using this tighter rejection only little signal is lost whereas almost all background is removed.

Figure 5.4 shows the $J/\psi \to e^+e^-$ signal after the conversion background has been removed. The left part of the figure shows the signal after rejecting all candidates where both e^+ and e^- are identified as conversion electrons, whereas in the right plot all candidates are rejected where either e^+ or e^- originates from a conversion. This way a very clean signal on a low background level is obtained.

Correcting for Bremsstrahlung.— Due to their low mass electrons mainly lose energy by Bremsstrahlung: The electron interacts with the detector material and emits a photon in forward direction. This is illustrated in figure 5.5: Using the truth information of simulated events, each plotted point represents the position of a material interaction where an electron radiated off a photon. The structure of the CDF detector is clearly visible, note the detailed ladder structure of the silicon vertex detector in the range $0 \leq r \leq 10$ cm. Following the cylindrical symmetry of the detector, the x-axis is the distance r (in cm) from the beam and the angle φ is plotted on the y-axis.

To study the Bremsstrahlung effects a realistic simulation has been performed: $\psi(2S)$ events have been created using `BGen` [74] which are forced to decay into $J/\psi\pi^+\pi^-$. The J/ψs are then forced to decay into e^+e^- pairs using `EvtGen` [64] and the `PHOTOS`

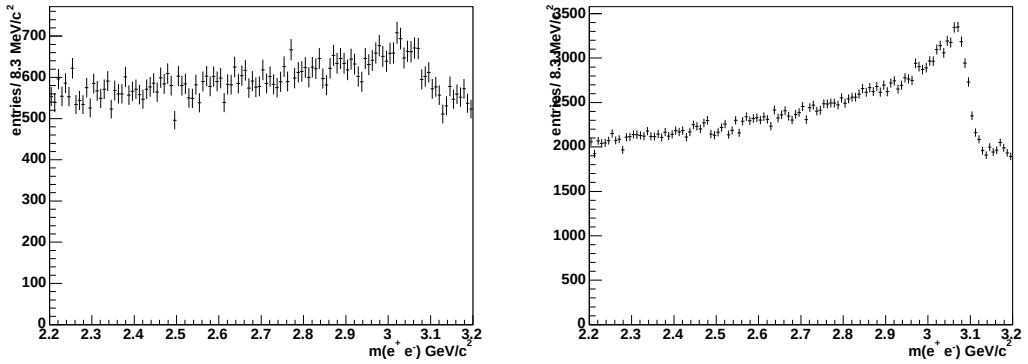


Figure 5.3: The figures illustrate the background from conversion electrons when reconstructing $J/\psi \to e^+e^-$. The left plot shows the invariant mass spectrum when requiring that *both* electrons originate from conversions, the right plot shows the invariant mass spectrum when requiring that either the e^- or the e^+ comes from a conversion.

[75] package to correctly model the radiative effects. The events simulated this way are passed through the full detector and trigger simulation described in detail in appendix E.3.

Effects of Bremsstrahlung on the track measurement.— Since the Bremsstrahlung photon is emitted in forward direction the trajectory of the electron will not show a kink but change curvature. This effect is exploited in the `KappaNet` to identify electrons. However, this effect presents extra challenges in analyses using electrons: Although the detailed simulation has shown that on average only two or three photons are radiated off by the electrons which leads to an energy loss of a few MeV, in more extreme cases up to 10 photons are radiated off and energy losses of more than one GeV can occur which leads to the strong radiative tail of reconstructed $J/\psi \to e^+e^-$. Consequently, the electron trajectory changes quite drastically several times as the particle traverses the detector, resulting in a significant underestimation of the errors of the fitted track helix parameters. The drift-chamber (COT) dominates the determination of the track's helix parameters due to its large volume compared to the silicon vertex detector. However, the electron has already passed a significant amount of material when it reaches the drift-chamber and most of the Bremsstrahlung has already been radiated off, i.e. the helix fit will in general produce a compromise between the electron's true trajectory and the found hits. The impact parameter (d_0), the curvature (κ) and φ_0 are affected most by Bremsstrahlung whereas the z-position (z_0) and $\cot(\Theta)$ are hardly changed. To compensate for this the error assigned to the measurement of

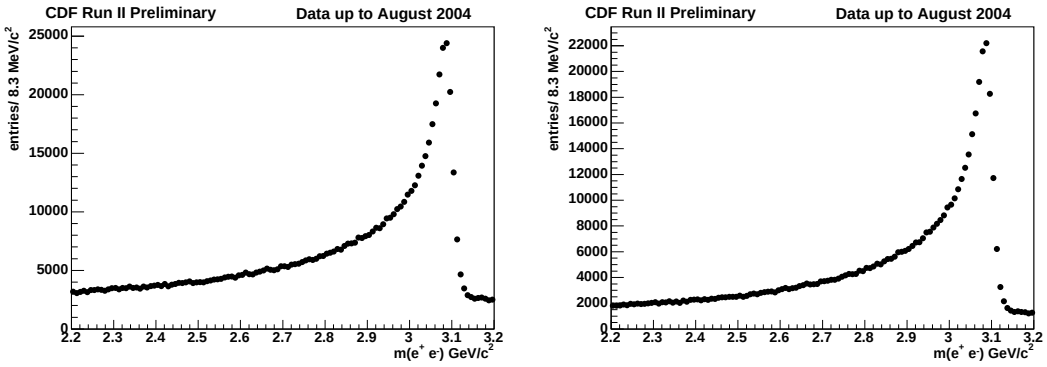


Figure 5.4: Improved $J/\psi \to e^+e^-$ after removing conversions. The left plot shows the achieved signal after removing the J/ψ when requiring that both constituents originate from conversions, the right plot shows the signal after rejecting candidates where either e^+ or e^- comes from a conversion. The ratio of the height of peak divided by the level of background taken from the right side-band is improved to ≈ 11 rejecting candidates where both e^+ and e^- are identified as conversion products (left plot) or to ≈ 22 when J/ψ are rejected where either e^+ or e^- originate from a conversion.

the first three helix parameters should be increased. However, since these parameters are correlated it is not sufficient to just rescale the measurement error of the respective quantity but the full covariance matrix needs to be adopted. This is done in the following way:

1. Obtain the covariance matrix from the track's helix fit.
2. Calculate the correlation matrix between all parameters.
3. Rescale the diagonal elements of the covariance matrix with a sufficiently large parameter[1] and change the off-diagonal terms accordingly *preserving* the correlations.
4. Assign the rescaled covariance matrix obtained this way to the track's helix fit.

This correction is particularly important if the J/ψ candidate is fitted together with other particles to a common vertex: The original errors on the track helix parameters will lead to a very high χ^2 number of the vertex fit (or the fit does not converge at all). Usually, one would discard such candidates as the χ^2 of the vertex fit is a quality measure of how well the particles 'fit together' at a common point. However, in the

[1]good results have been obtained with a scaling factor of 100, i.e. increasing the error on these measurements by a factor 10

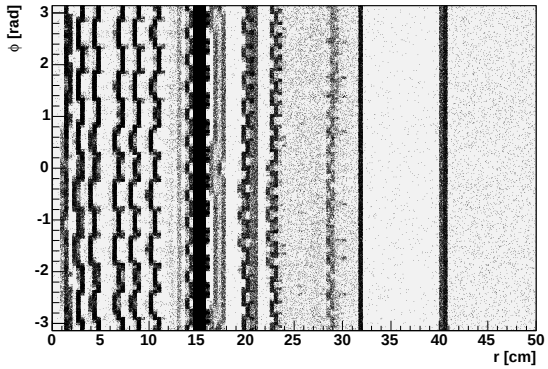


Figure 5.5: The plot illustrates the energy loss due to Bremsstrahlung: It shows where in the detector electrons have radiated off a photon using simulated events. Following the cylindrical symmetry of the detector, the x-axis is the distance from the beam, the angle φ is plotted on the y-axis. The material structure of the CDF detector is clearly visible.

case of electrons, the high χ^2 is dominated by the underestimated errors in the helix fit. By rescaling these numbers the χ^2 can again be interpreted as a measure of how well e.g. two pion candidates fit together to a $J/\psi \to e^+e^-$ vertex. Furthermore, since the overall χ^2 numbers are again in reasonable ranges, those fits which did not converge because the χ^2 hit an upper limit will again be successful.

Figure 5.6 illustrates that indeed Bremsstrahlung is responsible for the strong tail in $J/\psi \to e^+e^-$. To produce the plot, the electron and positron from which the J/ψ is formed are characterised by their energy, i.e. if the electron carries more energy than the positron it would be assigned to the 'higher energy' whereas the positron would go to the 'lower energy' group and vice versa. One immediately sees that the behaviour of the particles is quite different. The threshold at 2 GeV is due to the features of the dedicated $J/\psi \to e^+e^-$ trigger. Three different plots are overlayed for both the 'lower energy' particle and the 'higher energy' particle. The solid line represents the reconstructed energy as measured by $E = \sqrt{m^2 + \vec{p}^2}$. The triangles show the energy originally simulated (i.e. without any detector or measurement effects). The circles then show the originally simulated energy after the energy of all emitted photons has been subtracted (since this uses the simulated "truth" information effects from the reconstruction software are not visible).

To correct for the energy lost by Bremsstrahlung the following approach is taken: It is assumed that one of the electron or positron from the decay of the $J/\psi \to e^+e^-$ will have radiated off more energy than the other. All lost energy is then attributed

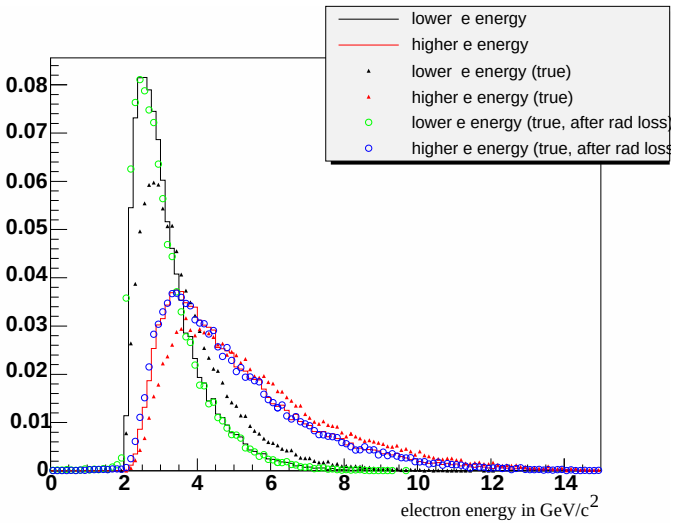


Figure 5.6: The plots show the effect of Bremsstrahlung in $J/\psi \to e^+e^-$: For both constituent particles the reconstructed energy, the energy as in the simulation and the difference due to Bremsstrahlung is shown.

to this particle where the this energy loss is defined from the difference between the reconstructed invariant mass $m(e^+e^-)$ and the nominal J/ψ mass. In detail, the performed corrections are:

1. Identify which of e^+ and e^- has radiated off more energy
2. Determine the amount of energy lost: $m_{J/\psi} - m_{ee}$
3. Determine the 4-vector components of a photon with that energy collinear with the particle identified in step (1)
4. Add this 4-vector to the 4-vector of the identified particle and determine new track helix parameters from this combined 4-vector at the $J/\psi \to e^+e^-$ vertex.
5. Overwrite the parameters of the original track helix fit with the parameters obtained by this way.

A neural network has been trained to determine which of the electron and positron has radiated off more energy. This `BremsID` network uses track based quantities such as the curvature change of the electron's track helix at thick material layers in the detector (these quantities are also used in the `KappaNet`), the maximal residual of a hit in the silicon vertex detector with respect to the fitted track helix, etc. This network has been trained using a dedicated signal simulation: Single $\psi(2S)$ particles are generated

using `BGen` which are then decayed into $\psi(2S) \to J/\psi\pi^+\pi^-$ using `EvtGen` [64], where the J/ψ was decayed into $J/\psi \to e^+e^-$. These events were then passed through the full detector and trigger simulation. More details about the `BremsID` network can be found in section B.5, the simulation process is described in detail in appendix E.3.

The 4-vector of the photon taking all the lost energy can be calculated in the following way: Suppose particle 2 has radiated off more energy, then: $p_1 = (E_1, \vec{p}_1),\ p_2 = (E_2, \vec{p}_2),\ p_\gamma = (E_\gamma, \vec{p}_\gamma)$ with $\vec{p}_\gamma = \lambda\vec{p}_2 \Rightarrow E_\gamma = \lambda|\vec{p}_2|$.

$$\begin{aligned} m^2_{J/\psi} &= (E_1 + E_2 + E_\gamma)^2 - (\vec{p}_1 + \vec{p}_2 + \vec{p}_\gamma)^2 \\ &= (E_{12} + \lambda|\vec{p}_2|)^2 - (\vec{p}_1 + (1+\lambda)\vec{p}_2)^2 \\ &= \lambda^2(|\vec{p}_2|^2 - \vec{p}_2^{\,2}) + 2\lambda(E_{12}|\vec{p}_2| - \vec{p}_1\cdot\vec{p}_2 - \vec{p}_2^{\,2}) + E_{12}^2 - \vec{p}_1^{\,2} - \vec{p}_2^{\,2} - 2\vec{p}_1\cdot\vec{p}_2 \end{aligned}$$

where E_{12} is defined by $E_{12} = E_1 + E_2$. The equation derived this way is linear in λ. Using $(E_1 + E_2)^2 - (\vec{p}_1 + \vec{p}_2)^2 = m^2_{ee}$, it can be solved for λ:

$$\begin{aligned} \lambda &= \frac{m^2_{J/\psi} - [E_{12}^2 - \vec{p}_1^{\,2} - \vec{p}_2^{\,2} - 2\vec{p}_1\cdot\vec{p}_2]}{2(E_{12}|\vec{p}_2| - \vec{p}_1\cdot\vec{p}_2 - \vec{p}_2^{\,2})} \\ &= \frac{m^2_{J/\psi} - m^2_{ee}}{2(E_{12}|\vec{p}_2| - \vec{p}_1\cdot\vec{p}_2 - \vec{p}_2^{\,2})} \end{aligned}$$

Although this approach neglects that the electron and positron each have radiated off several photons and attributes all lost energy to one particle, it has several advantages. The strong radiative tail is removed and the mass of the J/ψ candidate can be constrained to the nominal value in a further vertex fit. This constraint is usually performed in the final vertex fit when the J/ψ is part of the decay chain of another particle (e.g. $X(3872) \to J/\psi\pi^+\pi^-$) to enhance the resolution: The natural width of the J/ψ is much smaller than the detector resolution. Thus fixing the mass of the $\ell^+\ell^-$ to the nominal J/ψ mass in the final vertex fit corresponds to incorporating the a-priori knowledge that the leptons originate from the decay of a J/ψ and all deviations from the nominal mass are due to detector effects. This works very well in the case of $J/\psi \to \mu^+\mu^-$ since there the radiative effects are very small and can be neglected. However, Bremsstrahlung plays a vital role in the case of $J/\psi \to e^+e^-$ and the the mass-constraint cannot be applied: The deviations from the nominal J/ψ mass are now almost entirely due to the Bremsstrahlung process. After correcting for this as discussed in detail above the mass-constraint can be applied safely again in the final fit.

5.2 Using the electron ID toolbox for b-flavour tagging

Introduction.— One of the key measurements of the CDF 2 experiment is the determination of the transition (or *oscillation*) frequency between the neutral B_s^0 meson and its anti-particle $\bar{B}_s^0$. These oscillations originate in the fact that the quark mass eigenstates (d, s, b) differ from the eigenstates of the weak interaction (d', s', b') in the Standard Model. These states are connected via the Cabibbo-Kobayashi-Maskawa matrix [76, 77]:

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

This matrix is unitary for three quark generations, i.e. $V^\dagger V = \mathbb{1}$.
The $B_{d,s}^0 - \bar{B}_{d,s}^0$ mesons can then be described as a decaying two-component system:

$$\begin{aligned} \left|B^0\right\rangle &= \frac{1}{\sqrt{2}}\left(\left|B_H^0\right\rangle + \left|B_L^0\right\rangle\right) \\ \left|\bar{B}^0\right\rangle &= \frac{1}{\sqrt{2}}\left(\left|B_H^0\right\rangle - \left|B_L^0\right\rangle\right) \end{aligned}$$

where $|B^0\rangle$ denotes the flavour eigenstates and $\left|B_{L,H}^0\right\rangle$ are the mass eigenstates. Due to the small mass difference $\Delta m = m_H - m_L$, the "heavy" (H) and "light" (L) states evolve differently with time, which can be expressed as

$$p_\pm(t) = \frac{1}{2}\Gamma e^{-\Gamma t}\left(1 \pm \cos(\Delta m t)\right),$$

describing the probability for an initially pure B^0 meson to decay as a B^0 meson (pos. sign) or $\bar{B}^0$ meson (neg. sign) as a function of time. In terms of Feynman graphs, these oscillations are described by the box-diagrams shown in figure 5.7. The overall lifetime τ of the B mesons is described by the decay width $\Gamma = 1/\tau$.

A precise measurement of Δm_d (Δm_s) allows the determination of CKM matrix element V_{td} (V_{ts}) using predictions from theory calculating these box diagrams [78]. However, these calculations are impaired by large theoretical uncertainties, hence the determination of the ratio $\Delta m_s/\Delta m_d \propto |V_{ts}|^2 / |V_{td}|^2$ plays a vital role in constraining the values of the CKM matrix elements.

The experimentally accessible quantity is the asymmetry of decays which proceed via mixing compared to direct decays:

$$A(t) = \frac{N_{mix}(t) - N_{unmix}(t)}{N_{mix}(t) + N_{unmix}(t)} \quad \propto \cos(\Delta m_{d,s} t) \tag{1}$$

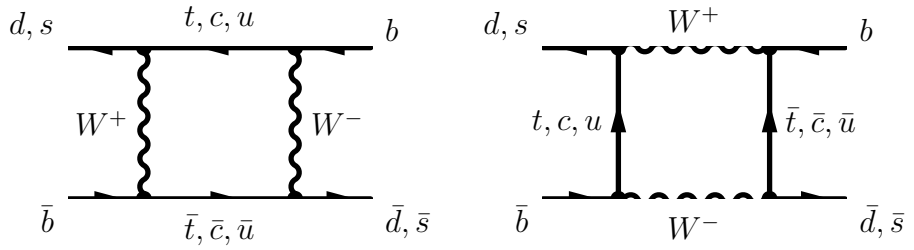


Figure 5.7: Feynman graphs describing the oscillation of the neutral $B^0_{d,s} - \bar{B}^0_{d,s}$ system

Due to their higher mass difference, $B^0_s - \bar{B}^0_s$ meson oscillations are at least 20 times faster than in the B^0_d system, as illustrated by figure 5.8. Highly efficient B^0_s reconstruction at high purity as well as excellent tools discriminating between mixed and unmixed states are essential prerequisites in measuring the oscillation frequency Δm_s.

The oscillation in the B^0_d system has been observed in 1987 for the first time [79, 80]. Combining measurements from LEP, CDF, BaBar and Belle, its frequency is determined to $\Delta m_d = 0.502 \pm 0.004(\text{stat.}) \pm 0.005(\text{syst.})$ ps^{-1} [67].

Results for the B^0_s system are limited by statistics so far, combining the measurements from the above experiments, the lower limit of $\Delta m_s > 14.4\text{ps}^{-1}$ [67] is obtained. Until the start of the LHC, B_s mesons can only be produced at the Tevatron experiments CDF and DØ because they are inaccessible at the B-factories BaBar and Belle due to their high mass.

The measurement of the B^0_s mixing frequency presents several unique challenges. Only ≈ 1.5 in 1000 events contain a $b\bar{b}$ pair. Dedicated triggers such as the TTT (*"two track trigger"*) and the ℓ+SVT (*"lepton plus SVT track"*) trigger exploit the high tracking resolution of the silicon vertex detector already on trigger level and select events with tracks displaced from the primary production vertex. These tracks likely originate from a secondary vertex which are mostly formed by B mesons due to their large lifetime. The TTT requires at least two tracks with transverse momentum $p_t > 2$ GeV/c at large impact parameters 100 μm$< |d_0| <$ 1 mm and enriches events where a B hadron decays into two hadrons, i.e. $B \to h^+h^-$. The ℓ+SVT trigger is sensitive to semi-leptonic B meson decays, i.e. $B \to \ell\ \nu_l\ h\ X$, where the lepton ℓ is either an electron or a muon. Muons have to be detected in either the CMU or the CMP muon chambers, electrons are required to deposit at least $E_T > 4$ GeV in the central electromagnetic calorimeter (CEM). Furthermore, only little energy may be brought into the hadronic calorimeter, i.e. $E_{had}/E_{em} < 0.125$. Both the track associated with the lepton and the displaced SVT track are required to have a minimal transverse momentum of $p_t > 2$ GeV/c. In addition, the impact parameter of the SVT track has to be between 120 μm$< |d_0| <$ 1 mm. The transverse invariant mass $m_T(\ell, SVT)$

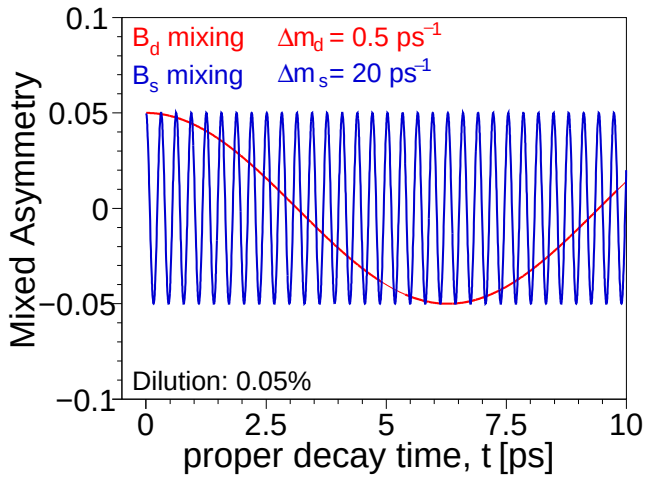


Figure 5.8: Comparison of the oscillation frequency Δm_d of B_d^0 mesons and Δm_s of B_s^0 mesons for an assumed frequency of $\Delta m_s = 20$ ps^{-1}. The figure illustrates the experimental challenge resolving the very high mixing frequency.

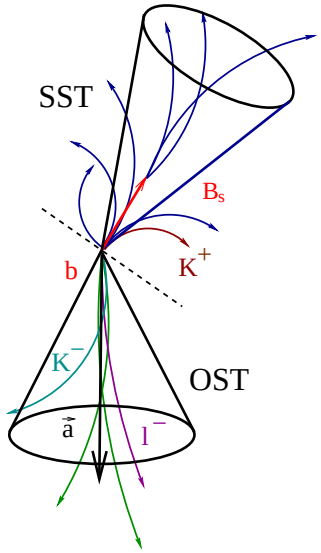


Figure 5.9: The figure illustrates the definition of same side (SST) and opposite side (OST) for reconstructed B mesons.

has to be smaller than 5 GeV/c^2 and the opening angle between the lepton and the SVT candidate has to lie in the interval $5° < \Delta\Phi < 90°$. Additionally, electrons are required to match CES showers with $E_T > 4$ GeV.

The crucial part in the mixing analysis is the so-called "flavour tagging", i.e. the determination whether the B meson contains a b or $\bar{b}$ quark. Figure 5.9 illustrates the topology of a fragmenting $b\bar{b}$ quark system. The b quark on the same side fragments into a B_s meson and is accompanied by a kaon as leading fragmentation particle. Different types of taggers are distinguished depending on which meson they operate on: *Same Side Taggers* (SST) determine the quark type of the exclusively reconstructed B meson, e.g. by exploiting information about the charge of the leading fragmentation particle. *Opposite Side Taggers* (OST) on the other hand use information about the other b quark in the event. One approach is to inclusively determine the weighted sum of the charge of all tracks in the associated jet [81]: $Q_{jet} = \frac{\sum_i w_i Q_i}{\sum_i w_i}$ with an appropriately defined weight w_i, e.g. $w_i = p_{t,i}^{\alpha}(2 - T_p)$ where T_p is the probability of the track to originate from the primary vertex. Q_{jet} is then correlated to the charge of the b quark which the analysed jet originates from and can hence be used to determine the quark flavour.

About 20% of the B mesons decay semi-leptonically, these decays originate in the weak transition of the b quark, e.g. $b \to W^- c$, $W^- \to \ell^- \bar{\nu}$, hence detecting the lepton originating from this decay and determining its charge allows to discriminate between whether the decaying neutral B meson contained a b or a $\bar{b}$ quark.

Construction of the electron tagger.— Developing a tagger based on sophisticated neural networks is expected to result in a significant improvement in the discriminating power of the tagger since the neural network can optimally exploit the correlation between the various input variables. Furthermore, the network can cope with cases where parts of the detector information is not available for some candidates. As traditional methods typically require that all information from the detector has been obtained, using a neural network based approach leads to a higher efficiency of the tagger, i.e. more candidates can be analysed than with other methods.

As the electron identification toolbox described in detail in chapter 4 is implemented as a general purpose toolkit with multiple interfaces, it can be used in a large variety of physics analyses. The development of the neural network based electron tagger is done in the `BottomTagger` and `BottomAnalysis` framework [82] to allow easy and seamless integration into the existing B_s^0 mixing analysis effort. The tagging is done in the following way:

1. The exclusively reconstructed B meson candidate defines the *same side*. Tracks used in the exclusive B meson reconstruction are removed from the track list and hence not considered further.

2. Tracks which are on the same side as the exclusively reconstructed B meson are discarded. This is verified by checking whether the tracks lie within a cone $\Delta R = \sqrt{(\Delta\Phi)^2 + (\Delta\eta)^2} < 0.7$ around the direction of the B meson.

3. Reject the track if it is more than 5 cm away in z direction from any of the tracks associated with the exclusively reconstructed B meson.

4. The output of the soft-electron identification network is obtained for each remaining track.

5. Conversion electrons are rejected using the conversion identification network.

6. If several electron candidates are selected by the above procedure, use the one with the highest probability to originate from a B hadron decay to derive the tagger decision.

In case several electron candidates are found on the opposite side of the exclusively reconstructed B meson, the one with the highest probability to originate from B

meson decays is selected for the further tagging procedure. The probability is determined using a dedicated neural network. This network requires a loose cut on the soft-electron identification network to reject obvious background events and has been trained to separate electrons originating from B hadron decays from other particles. The output of the electron and conversion identification networks, properties of the electron candidate such as the impact parameter and information from the exclusively reconstructed B meson are used in this network. The list of input variables and the correlation between them are given in appendix B.6.

The charge of the electron selected this way is then used to determine the flavour of the B meson decaying on the opposite site of the exclusively reconstructed B meson.

Tagger optimisation.— The optimisation of the tagger is done in the following way: If the flavour of the b-quark in the hadron is known, the tagger is said to give a *right-sign* tag if its decision agrees with the flavour of the b-quark. If the decision disagrees, it gives a *wrong-sign* tag. If for any reason the tagger is not able to make a decision (e.g. because insufficient information is available for a given candidate to run the tagger), this is counted as a *no-tag* candidate.

The performance of the tagger is evaluated in terms efficiency ϵ and dilution $\mathcal{D}$, which are derived from the number of (correctly) tagged candidates: The *efficiency* is defined as:

$$\epsilon = \frac{N_{RS} + N_{WS}}{N_{RS} + N_{WS} + N_{NT}} \tag{2}$$

where N_{RS} is the number of right-sign tags, N_{WS} is the number of wrong-sign tags and N_{NT} is the number of no-tag candidates. The efficiency hence reflects the percentage of candidates with tagger information.

The *dilution* is defined as:

$$\mathcal{D} = \frac{N_{RS} - N_{WS}}{N_{RS} + N_{WS}} \tag{3}$$

and is related to the probability $\mathcal{P}_{rs}$ to give a right-sign tag and the probability $\mathcal{P}_{ws}$ to give a wrong-sign tag via:

$$\begin{aligned} \mathcal{P}_{rs} &= \frac{N_{rs}}{N_{RS} + N_{WS}} = \frac{1 + \mathcal{D}}{2} \\ \mathcal{P}_{ws} &= \frac{N_{ws}}{N_{RS} + N_{WS}} = \frac{1 - \mathcal{D}}{2} \end{aligned}$$

The name "dilution" is therefore a bit counter-intuitive as good taggers are characterised by a high dilution. It results from the fact that the measured asymmetry and the true asymmetry (see eqn. 1) are related via $A_{meas} = \mathcal{D} A_{true}$, i.e. the true asymmetry A_{true} is diluted by the imperfect tagger decision. The statistical uncertainty of

A_{true} is given by (see e.g. [81]):

$$\sigma_A = \sqrt{\frac{1 - \mathcal{D}^2 A^2}{\epsilon \mathcal{D}^2 N}}$$

where N is the number of events before tagging. The uncertainty scales as $1/\sqrt{\epsilon\mathcal{D}^2}$, hence the taggers are optimised to achieve a high tagging power $\epsilon\mathcal{D}^2$.

The tagging power is evaluated using data recorded by the e + SVT track trigger and μ + SVT track trigger. The numbers are obtained using a provided script from the `BottomAnalysis` [82] framework into which this tagger has been integrated. This ensures that all taggers are evaluated in the same way and can be directly compared to each other. The tagger gives a right-sign (RS) tag when the tagger decision (i.e. the charge of the electron identified by the neural networks) and the charge of the trigger lepton have the opposite sign. If the tagger decision and the charge of the trigger lepton have the same sign, the tagger gives a wrong-sign (WS) decision. If the tagger does not give a decision, the event is not tagged. This can happen if either no electron candidate was found or all electron candidates are likely to originate from the conversion of a photon inside the detector material. To remove events where the trigger lepton and the displaced SVT track do not originate from the same vertex (e.g. QCD background events), a background subtraction technique based on the signed impact parameter δ^{SVT} of the SVT track is used, which is described in detail in [83] and appendix B.7. The procedure to compute the uncertainties on the efficiency and the dilution taking into account the background subtraction is described in [84]. The dilution computed from the above numbers is called *raw dilution* $\mathcal{D}_{raw}$. The *true dilution* $\mathcal{D}_{true}$, which is used to determine the tagging power $\epsilon\mathcal{D}^2$, is obtained by scaling the raw dilution: $\mathcal{D}_{true} = \frac{\mathcal{D}_{raw}}{s}$. Using simulation studies, the scaling factor has been determined in ref. [83]:

$$s = \begin{cases} 0.6412 \pm 0.0015\ (\text{stat})^{+0.0141}_{-0.0226}\ (\text{syst}) & \mu + SVT \\ 0.6412 \pm 0.0015\ (\text{stat})^{+0.0215}_{-0.0367}\ (\text{syst}) & e + SVT \end{cases}$$

It takes into account the wrong-sign correlations on the same side (i.e. of the exclusively reconstructed using the trigger lepton and the displaced SVT track) B meson due to mixing and sequential semi-leptonic decays (i.e. when the c quark originating in the b quark decay also decays semi-leptonically).

Results and comparison to other methods.— Currently, an opposite side electron tagger based on a likelihood approach [56] is used in the CDF B_s^0 mixing analysis. A cut-based approach using the `SoftElectronModule` [55] exists but is not used in the mixing analysis.

The neural network based tagger described in detail above is designed to be fully compatible with the likelihood based method. This allows both a direct comparison

dataset	$\epsilon\mathcal{D}^2$ (likelihood based)	$\epsilon\mathcal{D}^2$ (network based)
e + SVT	$0.488 \pm 0.043\%$	$0.677 \pm 0.054\%$
μ + SVT	$0.305 \pm 0.030\%$	$0.365 \pm 0.034\%$

Table 5.1: Tagging power $\epsilon\mathcal{D}^2$ achieved with the neural network based approach compared to the likelihood based method as measured in the data.

of the two methods based on measuring the performance $\epsilon\mathcal{D}^2$ in the data and to switch easily between the two approaches. The performance of the taggers is evaluated separately for the e+SVT and μ+SVT dataset. All available data until the shutdown in August 2004 has been analysed (dataset `xbel0d` and `xbmu0d`) using release 5.3.4 of the CDF software framework. Since both the tagger based on the likelihood approach and the neural network based tagger discussed in detail above run within the same analysis program and thus evaluate *exactly* the same events, the obtained result can be directly compared. Table 5.1 summarises the results achieved with the neural network based tagger and compares the tagging power obtained this way to the tagging power achieved by the likelihood based approach. Using the developed neural network based method to discriminate between b and $\bar{b}$ quarks in semi-leptonic B hadron decays involving electrons results in a significant improvement of the the tagging power $\epsilon\mathcal{D}^2$ as illustrated by the above numbers. The different behaviour on the two datasets is observed in all taggers and is likely due to uncertainties in estimating the opposite-side dilution between the two samples as discussed in [56]. Key aspects of the improvement obtained using the neural network based approach are the optimal combination of correlated variables by the sophisticated neural networks used and the ability of the NeuroBayes® neural network package to use input variables where not all values are available for each candidate, which leads to a high efficiency of the tagger.

Chapter 6

Analysis of the $X(3872) \to J/\psi\pi^+\pi^-$

This chapter focuses on the analysis of the newly discovered particle $X(3872) \to J/\psi\pi^+\pi^-$. Section 6.1 is concerned with the reconstruction of the $X(3872)$ in the case where the J/ψ decays via $J/\psi \to \mu^+\mu^-$.

Section 6.2 then describes how the sophisticated techniques discussed in chapter 4 are used to include the channel $J/\psi \to e^+e^-$. By using these methods it is possible to observe the $X(3872)$ in this exclusive final state for the first time in the very challenging environment of hadronic collisions.

The measurement of the quantum numbers J^{PC} of the $X(3872)$ plays a vital role in the understanding of its nature. Section 6.3 discusses in detail the method of helicity amplitudes employed in the determination of spin, parity and charge parity and summarises the obtained results.

6.1 Reconstruction in the channel $J/\psi \to \mu^+\mu^-$

The analysis in the exclusive final state $\mu^+\mu^-\pi^+\pi^-$ uses all available data until the shutdown in August 2004 recorded by the CDF detector, using the $J/\psi \to \mu^+\mu^-$ trigger (dataset `jpmm0d`). This corresponds to an integrated luminosity of $\mathcal{L} \approx 355$ pb^{-1}. The run ranges $179056 - 182843$ and $184062 - 184208$ of the so called "COT compromised" period have been excluded since during these runs parts of the drift chamber have been switched off while investigation of the rapid aging was in progress.

The events are reconstructed using CDF software release 5.3.4. A "bottom-up" approach has been chosen for the candidate reconstruction: First, two muons are combined to a J/ψ candidate and appended to the event record, this collection is then read in by the $X(3872)$ analysis module and combined with two pions to form a $X(3872)$ candidate.

Each track used in the analysis has to pass the following basic selection criteria:

- The tracks are required to have a minimal number of hits in the $r-\phi$ layers of the silicon vertex detector (SVX) and the drift-chamber (COT). Depending on by which tracking algorithm the track was reconstructed, at least 2 SVX hits (for Inside-Out, Outside-In and SVX-stand-alone algorithms) and 10/10 hits in the axial and stereo layers of the COT (for Inside-Out, Outside-In and COT-stand-alone algorithms) are required.
- Hits from the intermediate silicon layer (ISL), 90Z and the innermost layer of the silicon vertex detector (L00) were included by default.
- Tracks are re-fitted taking into account the appropriate mass assumption.
- All tracks are required to be in the central region of the detector, i.e. $|\eta| < 1$.
- Tracks taken as pion candidates have to fulfil $p_t > 0.35$ GeV/c, no additional p_t cut besides the trigger requirement of $p_t > 1.5$ GeV/c has been applied for the muons.

Muon candidates of opposite charge are fitted to a common vertex to form J/ψ candidates using the CTVMFT [85] vertex fitter. The beam position and covariance is taken from the SVX beam at $z = 0$. It is ensured that candidates with duplicate tracks are not used. Only candidates in the window $3.02 \leq m(J/\psi) \leq 3.16$ GeV/c^2 are appended to the event collection and considered in the subsequent analysis. Around 3.2M J/ψ candidates are obtained after the precuts, cf. figure 6.1.
These are combined with two pion candidate tracks to form a $X(3872)$ candidate. During the construction of the $X(3872)$ candidate it is ensured that candidates with duplicate tracks (e.g. one of the pions tracks is also used as a muon) are not considered. Both right-sign (i.e. $X \to J/\psi\pi^+\pi^-$) and wrong-sign combinations (i.e. $X^{++} \to J/\psi\pi^+\pi^+$ or $X^{--} \to J/\psi\pi^-\pi^-$) are formed. The J/ψ mass is constrained to the nominal value in this fit to improve the resolution.

Following [86] the $X(3872)$ signal is enriched by the following cuts:

- A 60 MeV/c^2 mass window is required for the J/ψ with respect to the nominal value.
- The J/ψ is required to have a transverse momentum p_t of at least 4 GeV/c.
- The χ^2 of the di-muon vertex fit forming the J/ψ is required to be less than 15.
- All $X(3872)$ candidates are required to have a mass in the (broad) region of interest:
  $3.65 \leq m(J/\psi\pi^+\pi^-) \leq 4.0$ GeV/c^2.
- The χ^2 of the final vertex fit is required to be less than 25.

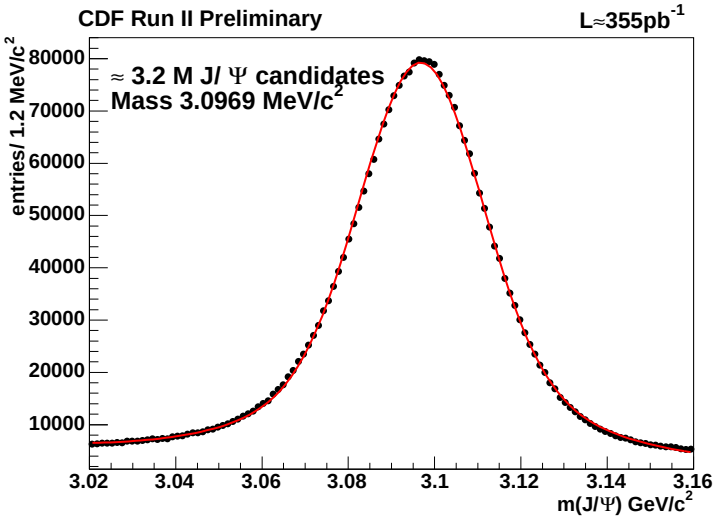


Figure 6.1: Distribution of J/ψ candidate mass with $p_t(J/\psi) \geq 4$ GeV/c, $\chi^2 < 15$ was required for the vertex fit. The fit was performed using two Gaussians with the same mean to parameterise the signal and a linear polynomial for the background

- Both pions are required to lie in a cone around the $X(3872)$ momentum vector with: $\Delta R(\pi) < 0.7$.

where $\Delta R = \sqrt{(\Delta\Phi)^2 + (\Delta\eta)^2}$. Here $\Delta\Phi$ is the azimuthal angle and $\Delta\eta$ is the pseudo-rapidity of the pion with respect to the $X(3872)$ candidate.

The mass spectrum obtained with these cuts is shown in figure 6.2. A clear $\psi(2S)$ signal is observed with 21905 ± 410 events. A significant excess of 959 ± 109 events is observed at the expected mass of about 3871 MeV/c^2. The numbers have been obtained from a simultaneous fit to the $\psi(2S)$, the $X(3872)$ and the background. The $\psi(2S)$ signal is described by a double Gaussian with common mean (following the approach in [87]), the $X(3872)$ signal is described by a single Gaussian distribution and the background is parameterised by a a second order polynomial. No signal is seen in either wrong-sign combination.

$X(3872)$ candidates favour a high mass of the $\pi^+\pi^-$ subsystem. Consequently, $X(3872)$ candidates can be enriched by requiring $m(\pi^+\pi^-) > 0.5$ GeV/c^2. Imposing this additional cut, one obtains 14320 ± 543 $\psi(2S)$ and 1490 ± 353 $X(3872)$ candidates (see figure 6.3).

To enhance the signal but to avoid cutting into the kinematics directly, a cut on the quantity $Q = m(J/\psi\pi^+\pi^-) - m_{PDG}(J/\psi) - m(\pi^+\pi^-)$ is applied, where:

- $m(J/\psi\pi^+\pi^-)$ is the mass of the $X(3872)$ candidate,

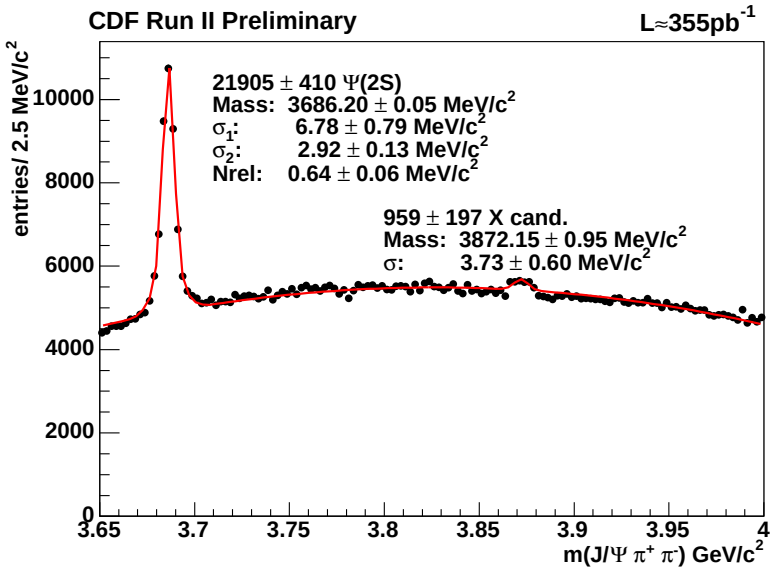


Figure 6.2: $J/\psi\pi^+\pi^-$ invariant mass distribution after cuts. Superimposed is a fit of the distribution, using a Gaussian distribution for each of the $\psi(2S)$ and $X(3872)$ resonant signals and a quadratic polynomial for the background.

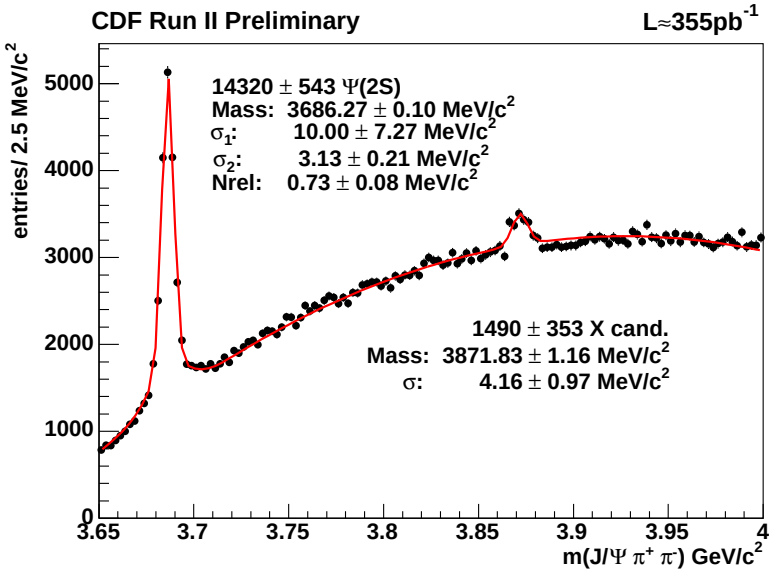


Figure 6.3: $J/\psi\pi^+\pi^-$ mass spectrum after additionally requiring $m(\pi^+\pi^-) > 0.5$ GeV/c^2

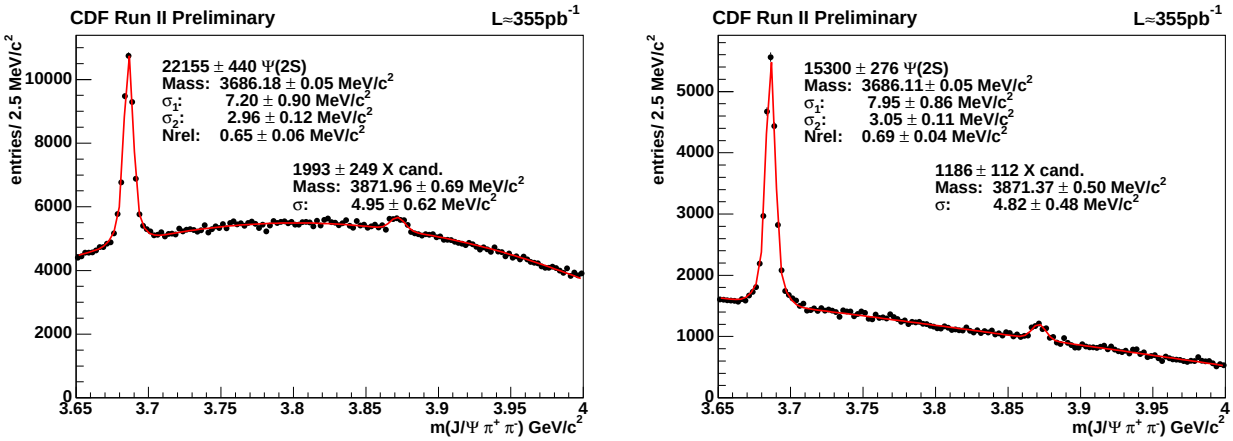


Figure 6.4: $J/\psi\pi^+\pi^-$ mass spectrum with additional cuts on $Q < 0.49$ (left) and $Q < 0.0098$ (right)

- $m_{PDG}(J/\psi)$ is the current world average mass of the J/ψ (taken from the Particle Data Group[67]),
- $m(\pi^+\pi^-)$ is the mass of the di-pion system where the 4-vectors of the two pions have simply been added, no vertex fit, etc. has been performed to combine the pions to a sub-resonance such as e.g. a ρ^0.

Cutting on this quantity can improve the signal to background ratio significantly, as illustrated in figure 6.4

6.2 Reconstruction in the channel $J/\psi \to e^+e^-$

The reconstruction of the $X(3872)$ in the exclusive final state $e^+e^-\pi^+\pi^-$ uses all data until the shutdown in August 2004 recorded by the CDF detector the dedicated $J/\psi \to e^+e^-$ trigger (dataset `jpee0d`). This trigger requires that two tracks of opposite charge with $p_t > 2$ GeV/c deposit at least $E_t > 2$ GeV in the central electro-magnetic calorimeter. Additionally, the ratio of energy deposited in the hadronic calorimeter and the electro-magnetic calorimeter is required to be below 0.125 to suppress hadronic background (mainly from pions). Again, all available data until the August 2004 shutdown has been analysed and the "COT compromised" period has been removed from the dataset. The reconstruction of the $X(3872)$ signal follows the same approach as the case where the J/ψ decays into two muons. All tracks have to meet the same basic selection criteria described in section 6.1. The p_t requirement on the muon tracks is replaced by a cut of $p_t > 2$ GeV/c on the electron tracks following the requirements of the $J/\psi \to e^+e^-$ trigger.

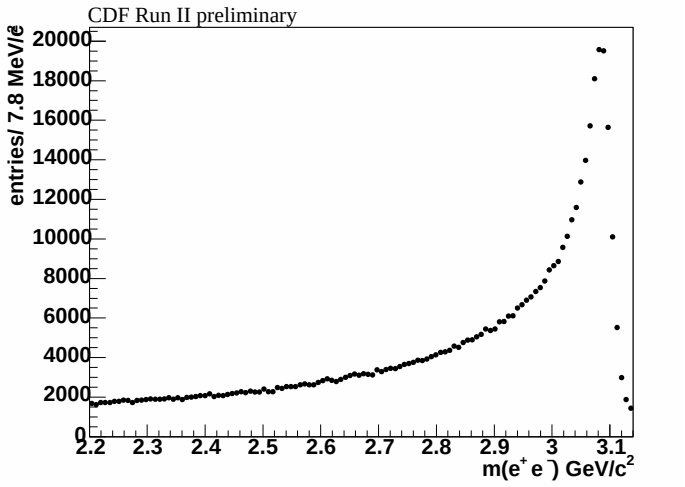


Figure 6.5: Invariant mass spectrum of the $J/\psi \to e^+e^-$ candidates used in the subsequent $X(3872)$ reconstruction. Successful candidates are required to have minimal transverse momentum $p_t(J/\psi) \geq 4.0$ GeV/c and $\chi^2 < 20$ for the vertex fit. Background from conversions has been removed using the `ConvNet`. 537441 J/ψ candidates remain after the selection cuts have been applied. Note the strong radiative tail as opposed to the case $J/\psi \to \mu^+\mu^-$.

The main difficulty in this part of the analysis is to obtain the $J/\psi \to e^+e^-$ candidates: Electrons and positrons are identified using the NeuroBayes® neural network at high purity and efficiency as described in detail in section 5.1. J/ψ candidates are then formed from the electrons and positrons identified this way. The main source of background originates from conversion electrons, i.e. when at least one of the electron or positron originates from the process $\gamma \to e^+e^-$, as discussed in section 5.1. J/ψ candidates where either constituent particle is likely to originate from a conversion are rejected. J/ψ candidates are further required to have a minimal transverse momentum $p_t(J/\psi) \geq 4.0$ GeV/c and the χ^2 value of the e^+e^- vertex fit was required to be below 20. After these selection cuts 537441 J/ψ candidates remained, as illustrated by figure 6.5. The figure shows the candidates prior to the corrections for Bremsstrahlung.

The strong radiative tail in the J/ψ mass spectrum illustrates that Bremsstrahlung plays a vital role in the reconstruction of the J/ψ signal. Following the discussion in section 5.1 the following corrections have been applied:

- Due to the Bremsstrahlung, the uncertainties on the track helix parameters d_0 (impact parameter), κ (curvature) and φ_0 are greatly underestimated. These uncertainties have been rescaled by a factor 10 *preserving* the correlations between

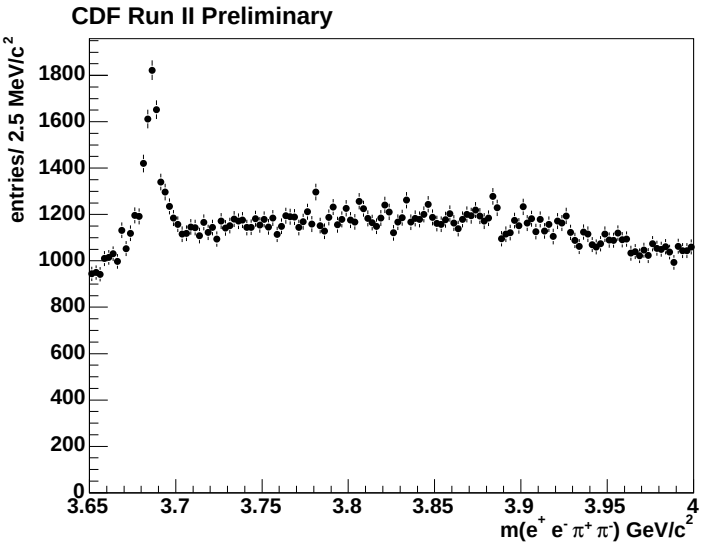


Figure 6.6: Inv. $J/\psi\pi^+\pi^-$ mass spectrum for the case $J/\psi \to e^+e^-$ after the basic $X(3872)$ selection cuts have been applied.

all fitted parameters.

- Using the `BremsID` neural network it is determined whether the electron or positron forming the J/ψ candidate has radiated off more energy by Bremsstrahlung. All lost energy is then attributed to this particle by determining the four-vector of a photon at the J/ψ vertex parallel to the identified $e^\pm$. The four-vector calculated by this approach is added to the four-vector of the $e^\pm$ and the helix-parameters are changed accordingly.

After the above corrections have been applied these J/ψ candidates are combined with two pions to form a $X(3872)$ candidate. Only right-sign candidates (i.e. $X \to J/\psi\pi^+\pi^-$) have been considered in this part of the analysis. Again it is ensured that candidates with duplicate tracks (e.g. one of the pions is also used as an electron candidate) have been rejected. The J/ψ mass is again constrained to the nominal value in the fit of the common $(e^+e^-\pi^+\pi^-)$ vertex. The $X(3872)$ candidates are then enriched by requiring:

- The J/ψ is required to have a transverse momentum p_t of at least 4 GeV/c.
- The χ^2 of the e^+e^- vertex fit forming the J/ψ is required to be less than 20.
- All $X(3872)$ candidates are required to have a mass in the broad region of interest:

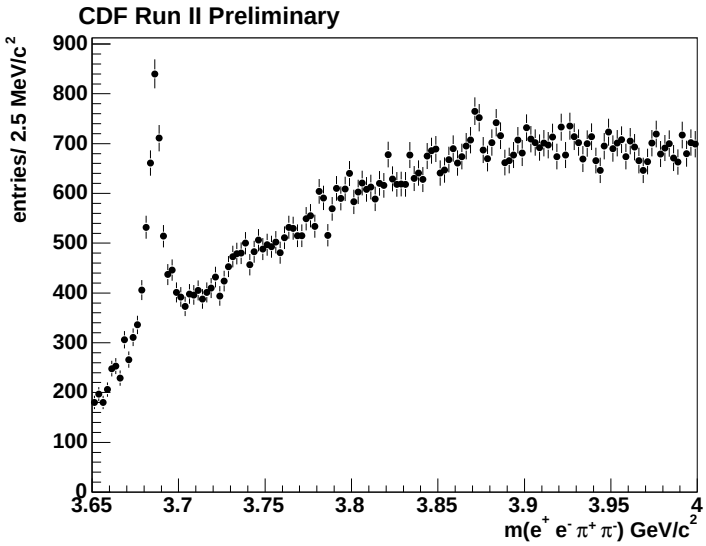


Figure 6.7: Inv. $J/\psi\pi^+\pi^-$ mass spectrum for the case $J/\psi \to e^+e^-$ with the additional cut $m(\pi^+\pi^-) > 0.5$ GeV/c^2. The inv. $J/\psi\pi^+\pi^-$ mass spectrum spectrum behaves as in the case where the J/ψ decays into two muons. A small excess is observed at the X signal region of $m(X) = 3.872$ GeV/c^2

$3.65 \leq m(J/\psi\pi^+\pi^-) \leq 4.0$ GeV/c^2.

- The χ^2 of the final vertex fit is required to be less than 20.
- Both pions are required to lie in a cone around the $X(3872)$ momentum vector with: $\Delta R(\pi) < 0.7$.

The cone is defined by $\Delta R = \sqrt{(\Delta\Phi)^2 + (\Delta\eta)^2}$, where $\Delta\Phi$ is the azimuthal angle and $\Delta\eta$ is the pseudo-rapidity of the pion with respect to the $X(3872)$ candidate.

The resulting invariant mass spectrum in shown in figure 6.6. A clear $\psi(2S)$ signal is observed, however, the $X(3872)$ is not clearly visible in this plot but is "covered" by the background. Note the overall much lower statistics as compared to the case where the J/ψ decays into two muons.

Figure 6.7 shows the invariant $J/\psi\pi^+\pi^-$ mass spectrum where the requirement $m(\pi^+\pi^-) > 0.5$ GeV/c^2 has been applied in addition to the basic selection cuts. Similar to the case where the J/ψ decays into two muons the background is suppressed and an excess becomes visible at the X signal region of $m(X) = 3.872$ GeV/c^2.

To further enrich the $X(3872)$ signal an additional cut on the quantity $Q = m(X) - m_{PDG}(J/\psi) - m(\pi^+\pi^-)$ is applied instead of a cut on $m(\pi^+\pi^-)$. Owing to the challenging task to reconstruct the small signal and suppress the much larger

hadronic background, a hard cut of $Q < 0.054$ is required to observe a significant excess as shown in figure 6.8. The invariant mass distribution is fitted with the same parametrisation as in the case where the J/ψ decays into two muons to allow a direct comparison. The left part of the figure shows the fit with all properties floating, whereas in the right part the mass and width of the $X(3872)$ are constrained to the values obtained from the fit with $J/\psi \to \mu^+\mu^-$. 1652 ± 75 $\psi(2S)$ candidates are obtained from the fit where the X properties are allowed to float and a significant excess of 135 ± 30 $X(3872)$ candidates is found at the mass $m(X) = 3872.18 \pm 0.84$ MeV/c^2. The obtained width of the Gaussian fitted to the signal is $\sigma = 3.44 \pm 0.89$ MeV/c^2. Fixing the mass and width of the $X(3872)$ to the respective values obtained in the case where the J/ψ decays into two muons 1654 ± 76 $\psi(2S)$ and 152 ± 29 $X(3872)$ candidates are obtained.

The shape of the mass spectrum in figure 6.8 is similar to the shape of the right part of figure 6.4 showing the corresponding plot with $J/\psi \to \mu^+\mu^-$. Thus by comparing the ratio of the obtained $\psi(2S)$ yield in either case it can be estimated which yield for the $X(3872)$ can be expected: 15301 ± 276 $\psi(2S)$ candidates are obtained in the case $J/\psi \to \mu^+\mu^-$ compared to a yield of 1654 ± 76 in the case $J/\psi \to e^+e^-$. Multiplying the obtained $X(3872)$ yield of 1186 ± 112 from the di-muon case with this ratio of ≈ 0.11 about 130 $X(3872)$ are expected in the case $J/\psi \to e^+e^-$, which is well compatible with the numbers obtained from the fit to the invariant $J/\psi\pi^+\pi^-$ mass spectrum. Figure 6.9 illustrates the behaviour of the observed $X(3872)$ signal for the case $J/\psi \to e^+e^-$ when varying the main reconstruction cuts. The significance is defined as the yield obtained from the fit to the invariant $e^+e^-\pi^+\pi^-$ mass spectrum divided by the uncertainty on this number as returned from the fit. The mass and width of the Gaussian distribution describing the $X(3872)$ signal are constrained to the values obtained in the case $J/\psi \to \mu^+\mu^-$.

6.3 Helicity analysis of the $X(3872)$

Overview.— A crucial point in understanding the nature of the $X(3872)$ is the determination of its quantum numbers J^{PC}. This can be accomplished using the method of helicity amplitudes (see e.g. [88], [89], [90]) exploiting the fact that kinematic variables such as the mass of the di-pion system and the angles between the various particles involved in the decay-chain $X(3872) \to J/\psi\pi^+\pi^-$ depend on the quantum numbers J^{PC} of the $X(3872)$ and the involved sub-resonances as well as on the orbital angular momentum between them. The behaviour of these variables can be predicted for numerous assumptions using helicity amplitudes. The correct assignment can then be determined by comparing the predicted behaviour to the distributions measured from the data.

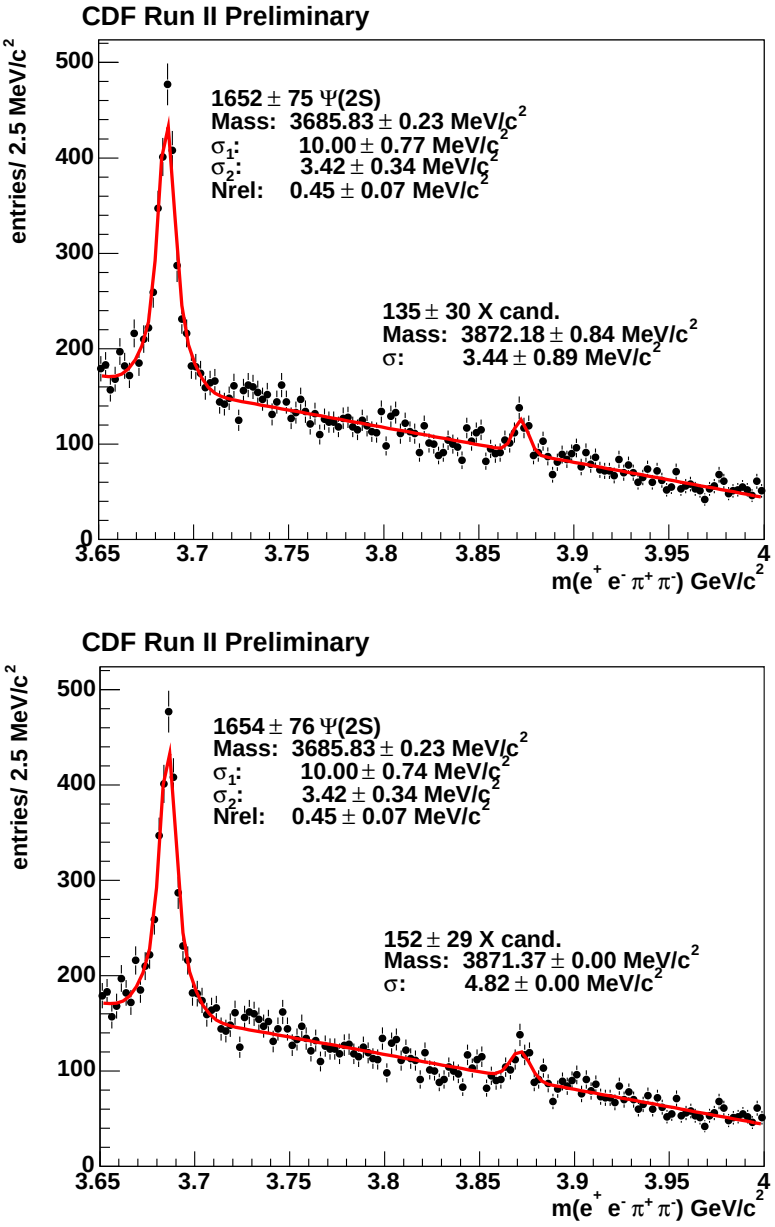


Figure 6.8: Inv. $J/\psi\pi^+\pi^-$ mass spectrum for the case $J/\psi \to e^+e^-$ with the additional cut $Q < 0.054$ GeV/c^2. All values are allowed to float in the fit shown in the upper part of the figure whereas in the lower part the mass and width of the $X(3872)$ are constrained to the values obtained from the fit with $J/\psi \to \mu^+\mu^-$.

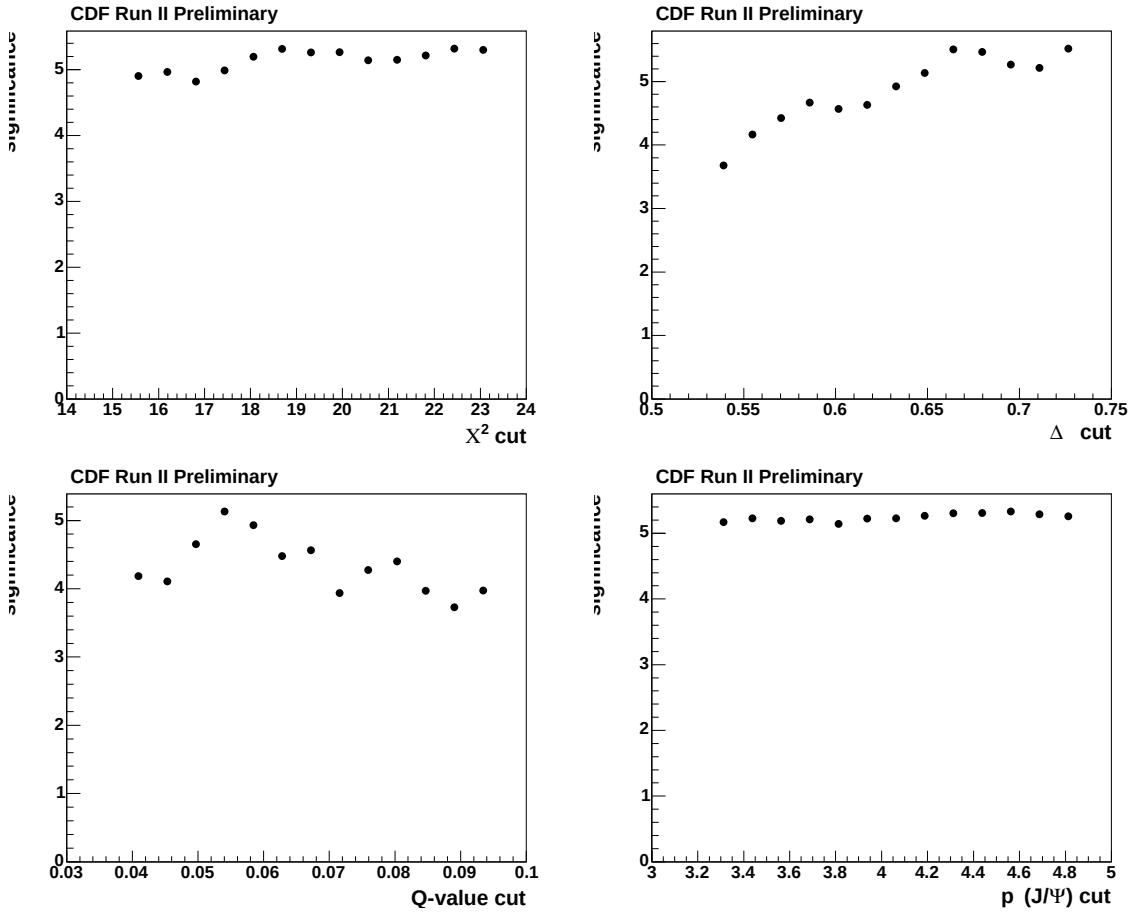


Figure 6.9: The figure illustrates the stability of the significance with respect to varying the cuts on χ^2 of the $X(3872)$ vertex fit, the cone radius Δ_R, the Q-value and the transverse momentum of the reconstructed $J/\psi \to e^+e^-$. The significance is defined as the $X(3872)$ yield obtained from the fit of the $J/\psi\pi^+\pi^-$ invariant mass spectrum divided by its error. The mass and the width of the Gaussian distribution describing the $X(3872)$ signal have been fixed to the vales obtained in the full fit for the case $J/\psi \to \mu^+\mu^-$.

This section describes in detail the analysis steps and summarises the main results. The work of this thesis forms the basis of the Karlsruhe $X(3872)$ helicity analysis [91]. The actual coding of the helicity amplitudes and the comparison to the data are done in a separate diploma thesis [10].

Theoretical framework.— All particles can be characterised by their spin and helic-

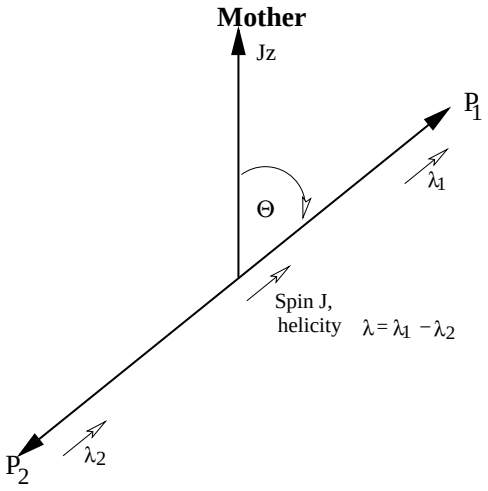


Figure 6.10: Illustration of the decay topology where particle A with parity P and spin projection J_z decays into daughter particles $1, 2$ with parities $P_{1,2}$ and helicities $\lambda_{1,2}$

ity, defined by $\lambda = \frac{\vec{s}\cdot\vec{p}}{|\vec{p}|}$, i.e. the projection of the spin on the momentum axis. To understand the formalism, the decay of particle A into two particles is considered:

$$\begin{array}{rcll} A & \to & 1 & 2 \\ J^P_{J_z} & \to & J^{P_1}_{1,\lambda_1} & J^{P_2}_{2,\lambda_2} \end{array}$$

The mother particle A is characterised by its parity P, its spin J and its spin projection J_z. Each daughter particle is described by its parity $P_{1,2}$ and helicity $\lambda_{1,2}$. Figure 6.10 illustrates the decay topology. The original z-axis is defined as the direction of particle A, i.e. $J_z = \lambda_A$. In the helicity formalism the decay matrix element is defined by the frame defined by the outgoing particles, i.e. rotated by the polar angle θ and the azimuthal angle φ with respect to the original system. The decay $A \to 1\ 2$ is a two-body decay, hence in its rest-frame the decay products are moving away back-to-back, i.e. $\vec{p}_{1,c.m.s.} = -\vec{p}_{2,c.m.s.}$ The orbital angular momentum $\vec{L} = \vec{r} \times \vec{p}$ is perpendicular to the momentum $\vec{p}_1$ of the outgoing particle 1. Therefore the final state has a component of angular momentum along the direction of particle 1 of $\lambda = \lambda_1 - \lambda_2$ (n.b. particle 2 is also quantised along its own direction of motion but this is not a new axis since $\vec{p}_1 = -\vec{p}_2$ in the centre-of-mass system of the decaying mother particle).

Instead of using the momentum vectors $\vec{p}_1$ and $\vec{p}_2$, the final state can be characterised by the direction (specified in terms of the polar angle θ and the azimuthal angle φ) of the decay axis with respect to the z-axis of the decaying particle and the magnitude k^* of either final state particle's momentum in the centre-of-mass frame.

The descriptions of any state with spin J and spin projection J_z in different frames of reference are related via Poincaré transformations. In absence of translations, rotations by angles θ and φ are described by:

$$|J, J_z\rangle_{0,0} = \sum_{\lambda=-J}^{J} D^J_{J_z,\lambda}(\varphi, \theta, -\varphi) \cdot |J, J_z\rangle_{\theta,\varphi}$$

where the rotation functions $D^J_{J_z,\lambda}(\varphi, \theta, -\varphi)$ are called Wigner functions which can be expressed in terms of the so-called reduced d functions:

$$D^J_{J_z,\lambda}(\theta, \phi) = e^{i\phi(J_z-\lambda)} d^J_{J_z,\lambda}(\theta)$$

They are listed in the PDG book [67] for low spins. In particular, these transformations relate the description of the states in the frames of reference of the initial state and the final state. Using this approach, the angular dependence of the transition (or helicity) matrix element describing the decay can then be expressed in terms of the Wigner D functions.

The transition amplitude is given by the full helicity matrix element:

$$\mathcal{M}^J_{\lambda_A,\lambda_1,\lambda_2}(p^\mu_1, p^\mu_2) \propto c_{LS}(\lambda_1, \lambda_2) \cdot D^J_{\lambda_A,\lambda_1-\lambda_2}(\phi, \theta, -\phi) \cdot {k^*}^L \cdot f_L(k^*) \tag{1}$$

composed of the Wigner D functions to describe the angular dependence and the term ${k^*}^L \cdot f_L(k^*)$ to account for the the magnitude of either decay particle's momentum in the centre-of-mass frame (given by k^*). L is the angular momentum of the transition. The form-factor $f_L(k^*)$ counters the divergence of the matrix element for rising k^* and needs to be modelled. A widely used model has been suggested by Blatt and Weiskopf [92] or Jackson [93]. The constant c_{LS} can be taken from comparison to the LS formalism as the product of two Clebsh-Gordan coefficients: The first factor combines the two daughter particle spins to the total spin S and the second factor couples spin S and angular momentum L to the total angular momentum J:

$$c_{LS}(\lambda_1, \lambda_2) = \left(\begin{array}{cc|c} J_1 & J_2 & S \\ \lambda_1 & -\lambda_2 & \lambda_1 - \lambda_2 \end{array}\right) \cdot \left(\begin{array}{cc|c} L & S & J \\ 0 & \lambda_1 - \lambda_2 & \lambda_1 - \lambda_2 \end{array}\right)$$

So far, only decays with two particles in the final state have been considered. However, this is not a limitation. Using the isobar model, the decay of the $X(3872) \to \ell^+\ell^-\pi^+\pi^-$ can be separated into several steps: The isobar model assumes that the decay of a given particle proceeds via intermediate states, e.g. the decay $A \to B{+}C{+}D$ is interpreted as the decay $A \to B + Y$ with the subsequent decay $Y \to C + D$. In the case of the $X(3872)$, the decay chain is constructed in the following way as illustrated by figure 6.11:

$$\begin{aligned} X(3872) &\to J/\psi\, R \\ J/\psi &\to \ell^+\ell^- \\ R &\to \pi^+\pi^- \end{aligned}$$

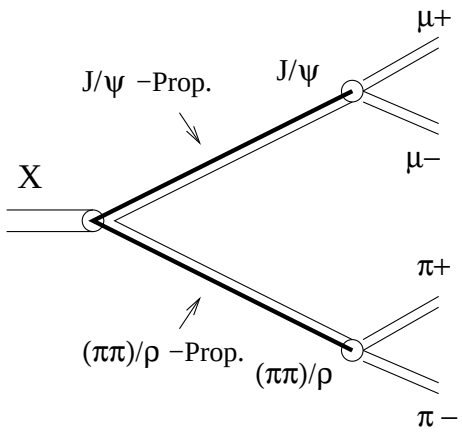


Figure 6.11: Schematic illustration of the $X(3872) \to J/\psi\pi^+\pi^-$ decay topology. The decay proceeds via the intermediate sub-resonances $J/\psi \to \mu^+\mu^-$ and $(\pi^+\pi^-)_{s,p,d} \to \pi^+\pi^-$.

The following three cases are considered for the di-pion system denoted by R: The pions are either in a relative s-wave state, form an intermediate ρ^0 (p-wave state) or intermediate $f_2(1270)$ (d-wave state). Due to the very limited statistics for the case $J/\psi \to e^+e^-$, only the di-muon channel is considered in the helicity analysis.
The complete decay chain is then given by:

$$\begin{aligned}\widetilde{\mathcal{M}}_{J_z,\lambda_{\mu^+},\lambda_{\mu^-}} = \sum_{\lambda_{J/\psi}} \sum_{\lambda_R} \quad & \mathcal{M}_{J_z\lambda_{J/\psi}\lambda_R}(X \to J/\psi\ R) \\ & \cdot\ \mathrm{Prop}_{J/\psi}\ \ \mathcal{M}_{\lambda_{J/\psi}\lambda_{\mu^+}\lambda_{\mu^-}}(J/\psi \to \mu^+\mu^-) \\ & \cdot\ \mathrm{Prop}_R\ \ \mathcal{M}_{\lambda_R}(R \to \pi^+\pi^-)\end{aligned}$$

where propagators $\mathrm{Prop}_{J/\psi}$ and Prop_R have been included for the J/ψ and the $\pi^+\pi^-$ system to connect the vertices. Since the helicities are not measured directly, they have to be summed over. A complete treatment for the example $X(J^P = 0^+) \to J/\psi\ \ (\pi^+\pi^-)_{s-wave}$ is given in appendix F.

The natural width of the J/ψ is much smaller than the experimental resolution and its mass can be treated as a constant fixed to the central value. The $\pi^+\pi^-$ resonances R however may have a broad mass distribution. If the pions form an intermediate ρ^0 or f_2, their propagator is given by a relativistic Breit-Wigner distribution:

$$\mathrm{Prop}_R(m_{\pi\pi}) = BW_R(m_{\pi\pi}) = \frac{1}{m_{\pi\pi}^2 - m_R^2 + i m_R \Gamma_R}$$

To account for the energy dependence of the broad widths, the Breit-Wigner formula

has to be modified, since kinematic factors vary across the width:

$$\Gamma_{A\to 12}(m) = \Gamma_{0,A\to 12} \cdot \left(\frac{k^*}{k_0^*}\right)^{2L+1} \left(\frac{f(k^*)}{f(k_0^*)}\right)^2 \left(\frac{m_0}{m}\right)$$

The index 0 indicates the nominal value as opposed to the actual value of the respective particle. An extensive treatment for the case that the pions form an intermediate ρ^0 resonance is also given in [21].

The treatment of the case where the $\pi^+\pi^-$ system forms an intermediate s wave state is more challenging as this state is not dominated by a single resonance. This issue can be faced by modelling the propagator by a flat mass distribution modified by an "Adler-Zero" [94]:

$$\mathrm{Prop}_{(\pi^+\pi^-)_s}(m_{\pi^+\pi^-}) = m^2_{\pi^+\pi^-} - \lambda m^2_\pi$$

with $\lambda = 4.35$ [95]. This describes the behaviour of the $\pi^+\pi^-$ system very well in case of the decay $\psi(2S) \to J/\psi\pi^+\pi^-$ and may also be valid for other cases. Yet, as the exact nature of the $X(3827)$ is unknown, this Ansatz may not be entirely correct and is hence complemented by the very conservative approach to disregard any information from the $\pi^+\pi^-$ mass spectrum and rely on the discriminating power of the angular variables only.

The complete differential cross section is given by

$$\begin{aligned} d^{11}\sigma/(d\omega^{11}) = \quad & d\sigma/(dm_X^2 && J/\psi\pi^+\pi^- \text{ invariant mass} \\ & dp_T^2 d\eta d\phi && \text{X(3872) production in laboratory frame} \\ & dm^2_{\pi\pi} d\cos\theta_X d\phi_X && \text{X(3872) decay into } J/\psi R \\ & d\cos\theta_{J/\psi} d\phi_{J/\psi} && J/\psi \text{ decay into } \ell^+\ell^- \\ & d\cos\theta_R d\phi_R) && J/\psi \text{ decay into } \pi^+\pi^- \end{aligned}$$

where p_T^2, η and ϕ describe the X(3872) resonance in the lab frame. The $X(3872)$ decays into J/ψ and the $\pi^+\pi^-$-resonance R is described by the decay angles $\cos\theta_X$ and ϕ_X (of the J/ψ in the $X(3872)$ centre of mass frame relative to the $X(3872)$ production axis). The subsequent decay of the J/ψ is determined by the angles $\cos\theta_{J/\psi}$ and $\phi_{J/\psi}$, and that of R by $\cos\theta_R$ and ϕ_R.

From the rotational symmetry of the system it can be deduced that for unpolarised $X(3872)$ production only the variables $m_{\pi\pi}$, $\cos\theta_{J/\psi}$, $\cos\theta_R$, $\Delta\phi = \phi_{J/\psi} - \phi_R$ depend on the J^{PC} of the $X(3872)$ and sub-resonances R. The definition of these angles is illustrated by figure 6.12.

Predictions for these distributions based on specific assumptions of the spin and parity of the $X(3872)$ and of the decay mode of the di-pion system are obtained by first simulating the decay of the $X(3872)$ using a simple phase-space generator. Then

each simulated event is reweighted according to the prediction of the helicity matrix element:

$$w = \frac{1}{2J+1} \sum_{J_z} \sum_{\lambda_{\mu^+}} \sum_{\lambda_{\mu^-}} \left| \widetilde{\mathcal{M}}_{J_z,\lambda_{\mu^+},\lambda_{\mu^-}} \right|^2$$

taking also into account acceptance effects, etc. of the detector. The correct J^{PC} assignment can then be determined by measuring these distributions in the data and comparing them to the numerous predictions using a χ^2 based technique. The distributions are extracted from the data in the following way: Each of the angular variables is split into several bins. To exploit the given symmetries, $|\cos\theta_{J/\psi}|$, $|\cos\theta_{\pi^+\pi^-}|$, $||\Delta\phi - \pi| - \pi/2|$ are used. Instead of using the $\pi^+\pi^-$ mass distribution directly, the Q value is used since cutting on $m_{\pi\pi}$ directly affects the shape of the $J/\psi\pi^+\pi^-$ invariant mass spectrum. 9 bins are used for the Q distribution, 15 bins are used for the angular variables in the case of the $\psi(2S)$. Due to the much lower statistics, only 8 bins can be used for the angular variables when analysing the $X(3872)$. To measure how the $\psi(2S)$ and $X(3872)$ signal depend on these variables, the bin borders defined this way are used as further cuts in addition to the ones discussed in section 6.1. Each of the invariant $J/\psi\pi^+\pi^-$ mass distributions obtained by this approach is then fitted using a double Gaussian with common mean to describe the $\psi(2S)$ signal and a single Gaussian distribution to describe the $X(3872)$ signal. The background is parameterised by a polynomial. The resulting yields of the $\psi(2S)$ or the $X(3872)$ obtained from these fits as a function of the binned variable are then compared to the behaviour of the respective distribution obtained from the simulated events weighted for a specific J^{PC} assumption using the definition

$$\chi^2_j = \sum_{bins} \frac{(y_{i,data} - y_{i,simulation})^2}{\sigma_i}$$

for each variable j. The uncertainty σ_i is given by the uncertainty on the yield of the $\psi(2S)$ or $X(3872)$ signal obtained from the fit in bin i. Figure 6.13 illustrates how the predicted behaviour changes for different quantum number assignments for the example of the angles $\Delta\Phi$ and $\cos\Theta_{J/\psi}$. The three different decay modes chosen in the illustration lead to very different behaviour in the decay angles and allow to discriminate between the various assumptions.

Results of the helicity analysis.— Applying the helicity analysis described above to the data, constraints on the quantum numbers J^{PC} of the $X(3872)$ are obtained. The actual coding of the helicity amplitudes and the subsequent χ^2 based determination of the most likely J^{PC} assignment are subject of a separate diploma thesis [10] which uses the provided plain `ROOT` ntuples containing the 4-vectors of the reconstructed particles in the various reference frames as a starting point.

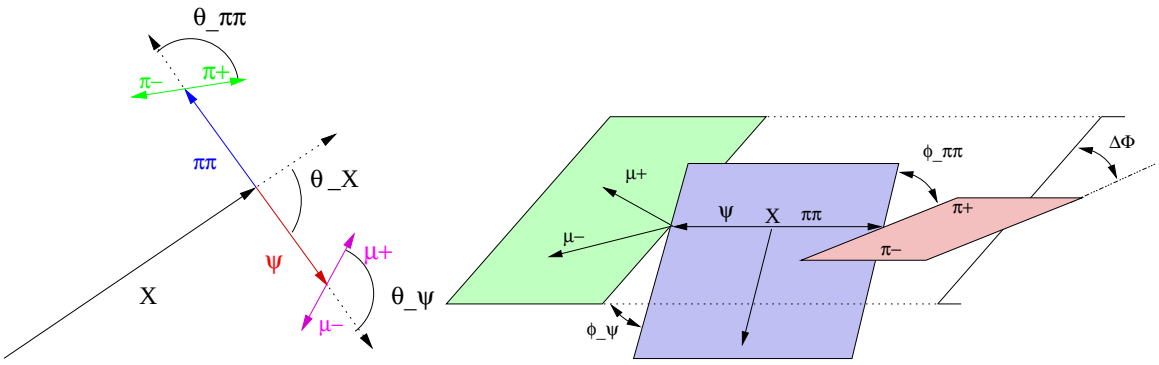


Figure 6.12: Angles between the various particles involved in the decay of the $X(3872)$. The left figure illustrates the angles θ between the mother particles and the respective decay particles, whereas the right figure shows the angle ϕ between the various decay planes.

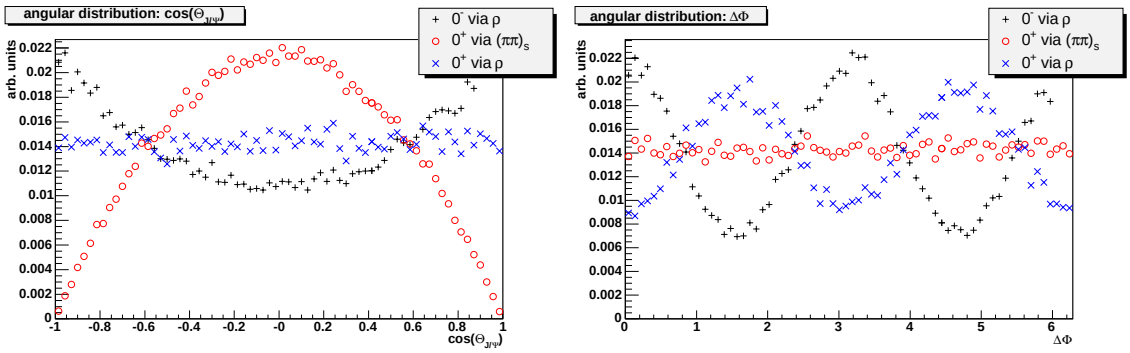


Figure 6.13: The figure illustrates the different predicted behaviour of the angular variables $\cos\Theta_{J/\psi}$ (left) and $\Delta\Phi$ (right) for the three assumptions that the $X(3872)$ is a 0^- or 0^+ state where the $\pi^+\pi^-$ forms an intermediate ρ^0 resonance, or a 0^+ state where the pions are in a relative s wave state.

A total of 21 different assignments covering the states with $J = 0, 1, 2, 3$ are tested. States where the $\pi^+\pi^-$ system is in a relative s- or d- wave state can only have negative charge parity C, whereas states where the pions form an intermediate ρ^0 have positive charge parity.

An important test of the method is the measurement of the known quantum numbers of the $\psi(2S)$ which decays into the same exclusive final state as the $X(3872)$. As expected, the correct assignment $J^{PC} = 1^{--}$ where the pions are in a relative s-wave state yields the best result, only a dedicated $\psi(2S)$ model [11] taking into account

a small d-wave contribution in the $\pi^+\pi^-$ system describes the data even better. All other assignments are rejected by more than 5σ. The same conclusion holds if using angular information only, fixing the information obtained from the $\pi^+\pi^-$ mass distribution such that it describes the data.

The same technique is then applied to determine the quantum numbers J^{PC} of the $X(3872)$. In a first step, J^{PC} assignments are considered where the $\pi^+\pi^-$ system forms either an intermediate ρ^0 or f_2 meson. The corresponding propagators are given by the appropriate Breit-Wigner distribution and hence the matrix element can be computed unambiguously. Performing the χ^2 based comparison to the data, the assignment $J^{PC} = 1^{++}_{\rho^0}$ describes the data best, separated by $\approx 1.4\sigma$ from the next best assignment $J^{PC} = 2^{++}_{\rho^0}$. States where the $\pi^+\pi^-$ system is in a d-wave state (i.e. forms an intermediate f_2 meson) are rejected by more than 6σ. Furthermore, states where the di-pion system forms an intermediate ρ^0 meson and the decay proceeds with orbital angular momentum $L > 0$ are also disfavoured by more than 6σ.

Following a very conservative approach, information about the $\pi^+\pi^-$ mass distribution is removed from the analysis in a second step. This is done by fixing the $\pi^+\pi^-$ propagator in the construction of the helicity weights such that the resulting $\pi^+\pi^-$ mass distribution agrees with the data. Thus only information originating from the angular distributions is used. This allows to reject the assignment $J^{PC} = 0^{+-}_s$ by more than 6σ and $J^{PC} = 0^{++}_{\rho^0}$ by $\approx 4\sigma$. Hence, the following eight candidates remain: 1^{+-}_s, 1^{--}_s, 2^{+-}_s, 2^{--}_s, 3^{+-}_s, 3^{--}_s, 1^{++}_ρ, 2^{++}_ρ.

In a final step, the propagator of the $\pi^+\pi^-$ system in an s-wave state is modelled by an Adler-Zero which has also been used to describe the decay of the $\psi(2S)$. Comparing the resulting distributions to the data, all assignments with negative charge parity ($C = -1$) with the $\pi^+\pi^-$ system in an s-wave state are rejected by more than 5σ. Hence, all assignments but $J^{PC} = 1^{++}_\rho$ and 2^{++}_ρ with no orbital angular momentum ($L = 0$) in the decay $X(3872) \to J/\psi\rho^0$ are clearly disfavoured with large statistical significance. The assignment $J^{PC} = 1^{++}_\rho$ describes the data best which supports the molecular interpretation proposed by Törnqvist and Swanson.

Chapter 7

Conclusions

The CDF Collaboration pursues a rich physics programme. The complex hadronic environment of $p\bar{p}$ collisions and the high event rates lead to unique challenges in both managing the large amounts of resulting data and the physics analyses. Grid computing technologies are needed to provide the high amount of computing power for the physics analyses and efficient access to the data. Through the work of this thesis is has been possible for the German CDF group to efficiently use the German Grid competence centre "GridKa" and analyse ≈ 500 TB of data up to now. The developed methods and accomplishments are of vital importance for similar computing centres in e.g. Italy, Asia and the USA, as well as the facilities on-site Fermilab.

The observation of the $X(3872) \to J/\psi\pi^+\pi^-$ by the Belle Collaboration and the quick confirmation by the CDF, DØ and BaBar collaborations mark the beginning of a new era in hadron spectroscopy. Conventional models explaining this particle as a charmonium state (i.e. a bound $c\bar{c}$ system) cannot satisfactorily explain the nature of the particle, e.g. the predicted mass is off by $\gtrsim 100$ MeV/c^2 from the observed value. The close proximity to the $D^0\bar{D}^{*0}$ mass threshold raises the question whether the $X(3872)$ is an exotic form of matter: Either a quark-gluon hybrid ($c\bar{c}g$) or molecular state first proposed by DeRújula, Georgi and Glashow in 1977.

Due to the extremely complex hadronic environment of $p\bar{p}$ collisions the decay $X(3872) \to J/\psi\pi^+\pi^-$ had been observed at the Tevatron experiments CDF and DØ only in the channel $J/\psi \to \mu^+\mu^-$ before this work. This motivated the development of a highly efficient electron identification tool, since the J/ψ decays to $\mu^+\mu^-$ and e^+e^- with equal rates. The electron ID tool uses the NeuroBayes® neural network package to separate a very clean sample of electrons and positrons from the more than a factor 10 higher background mainly consisting of pions. Neural networks are superior to other approaches as they allow to optimally combine information from calorimeters, specific energy loss in the central drift-chamber (dE/dx) and time-of-flight. Comparing to a standard cut-based approach and using simulated events, the

efficiency of identifying electrons is increased from $\approx 63\%$ to 97% at same or higher purity. The main remaining background originates from conversion electrons of the process $\gamma \to e^+e^-$ which can be removed using another dedicated neural network. Applying the methods developed, it has been possible to observe the decay $X(3872) \to J/\psi\pi^+\pi^-$ with $J/\psi \to e^+e^-$ for the first time at a hadron collider.

The development of this electron identification tool has high impact on all B physics analyses and will play a vital role in the analysis of semi-leptonic B decays, i.e. $B \to \ell X$ (where X denotes all other particles produced in the decay). To illustrate the high potential of this method, the electron identification tool has been integrated in the B_s oscillation analysis framework to tag the b quark flavour in semi-leptonic B decays involving an electron or positron. The performance of the new tagger leads to a significant improvement compared to the likelihood-based method currently used and contributes an important enhancement in the capability of the oscillation frequency Δm_s measurement.

Törnqvist and Swanson developed the original idea of charmed molecules further and suggest that the $X(3872)$ is a deuteron like molecular state where the D^0 and $\bar{D}^{*0}$ are bound by pion exchange. They find that such a state has the quantum numbers $J^{PC} = 1^{++}$ or 0^{-+}. The determination of the quantum numbers J^{PC} are thus vital to the understanding of the nature of the $X(3872)$. Using the method of helicity amplitudes, the spin, parity and charge parity of the $X(3872)$ are determined in a joint effort of a separate diploma thesis and this work. A total of 21 different assignments are tested, covering the possible states with $J = 0, 1, 2$. The assignment $J^{PC} = 1^{++}$ where the pions form an intermediate ρ^0 meson describes the data best, closely followed by the assignment $J^{PC} = 2^{++}$. All other states are disfavoured with high statistical significance. In order to avoid complications arising from modelling the s-wave state of the di-pion system, the analysis as been repeated using angular information only. Following this approach, spin $J = 0$ assignments can be excluded, whereas s-wave states with spin 1 or 2 remain. The obtained result strongly supports the molecular interpretation of the $X(3872)$, leading to the exciting possibility that the conventional classifications of hadrons into baryons and mesons have to be extended by exotic states such as meson molecules.

If the $X(3872)$ is indeed a molecular state, other states could exist bound by the same mechanism. An obvious candidate would be the b sector, replacing the charm quarks by beauty quarks. If such a state exists, it should decay similarly to the $X(3872)$, where the J/ψ (a $c\bar{c}$) state would be replaced by the $\Upsilon(1S)$ (the lowest $b\bar{b}$ vector state), i.e. a likely decay channel would be $X_b \to \Upsilon(1S)\pi^+\pi^-$. This state is expected to have a mass of $m(X_b) \approx 10.5$ GeV/c^2, i.e. close to the $B^0\bar{B}^{*0}$ threshold. Investigations are in progress to determine the existence of such a state, however, due to the high mass the available number of events containing an $\Upsilon(1S)$ is very limited. Keeping in mind the large combinatorial background the observation of such a state

will be even more challenging than for the case of the $X(3872)$.

Appendix A

Technical details for the SAM configuration

A.1 Overview

This appendix discusses the actual configuration of the SAM stations at GridKa and the University of Karlsruhe on a more technical level. Please keep in mind that it will be outdated at some point as the software is evolving.

The configurations described below only show the differences to the "default" installation provided by the `init_sam` command.

A.2 Details of the server-list file

The server-list file determines which SAM services are started and lists the arguments passed to the respective executable. The general layout of a configuration line is

```
product config-qual version station-name parameters
```

where `product` is the product to configure (e.g. station master) `config-qual` is the configuration qualifier taken from `sam_config`, `version` is the version to set up, `station-name` is the name of the sam station (e.g. cdf-fzkka) and then the parameters passed to the executable. Each product is set up by exactly one line (for presentational purposes, line breaks have been introduced here).

Note that options related to the Orbacus naming service start with a single dash (e.g. `-OAhost`) whereas all other options start with a double dash (e.g. `--min-delivery`).

```
nameservice ns_gridka_prd v3_3_4r -ORBtrace_level 5

station local_station_prd v6_0_1_12  cdf-fzkka -OAhost cdf.fzk.de  \
```

```
  -OAport 6789 --min-delivery=1k                    \
  --pmaster-arg=--consumption-map=\.\*::cdf.fzk.de \
  --log-file=station.log --max-prefetched-files=10 \
  --preferred-loc=/grid/fzk.de/mounts/pnfs/cdf/sam \
  --common-timeout=360 --intrastation-timeout=360  \
   --omit-loc=dcap --consumer-wait-timeout=15000

stager local_station_prd  v6_0_1_12   cdf-fzkka         \
 --with-fss --without-sm --max-transfers=5         \
 -OAhost cdf.fzk.de --node-name=cdf.fzk.de         \

fss prd v6_0_1_12 cdf-fzkka -OAhost cdf.fzk.de
gridftp prd v2_1_2
```

The station cdf-fzkka at GridKa runs an own naming service. Although this is not required, it enhances the robustness with respect to small network outages as the station and all projects can contact each other and do not have to rely on the central naming service being reachable. The `IOR` identifying the naming service is generated by the program automatically and can be found in its trace file.

In detail, the main options for the SAM station are:

- `-OAhost, -OAport` : tells the ORBACUS sub-system which machine the SAM station runs on and which port to bind to (needed in case the central naming service cannot be reached, e.g. due to firewall configuration).
- `--min-delivery=1k` : minimal file-size at which SAM should start file imports
- `--pmaster-arg=` : indicates that the following option should be passed on to each started project master and not be interpreted by the station itself.
- `--pmaster-arg=--consumption-map=\.\*::cdf.fzk.de` : all worker-nodes which can reach the SAM station cdf-fzkka will be served by it.
- `--log-file=station.log` : write a per-day log-file of all activity following the convention `station.log_-<date>`
- `--max-prefetched-files=10` : Limit the number of files imported by SAM when all files already cached have been analysed. This option is interpreted per project running.
- `--preferred-loc=/grid/fzk.de/mounts/pnfs/cdf/sam` : First try to retrieve files from the GridKa dCache system, if the files are not yet there, contact other locations. Note that all locations containing the above path will be preferred.

- `--omit-loc=dcap` : determines where *not* to retrieve files from. In this case, all locations containing the string `dcap` are ignored as these storage locations belong to a special setup of the central CDF SAM station.
- `--common-timeout=360` : set the general timeout to 360 minutes.
- `--intrastation-timeout=360` : SAM cache areas can be distributed globally but are associated to a specific SAM station. This option sets the timeout for transfers between two cache areas of the same station which are located at different physical locations to 360 minutes.
- `--consumer-wait-timeout=15000` : All SAM projects are ended automatically after a period of inactivity determined by this parameter, regardless if the requested files have been processed or not. This cleans up old processes which are no longer active, e.g. because the user's executables have crashed or have been ended by other means. However, since it is strongly advised to start a project before the first executable starts on a worker node, there may be (depending on the setup of the local site,etc.) an extended period between the start of the project and the start of the first executable. If this timeout is too short, the project is ended automatically before the first executable had a chance to run.

The stager configured in the GridKa server list file runs together with the FSS by specifying `--with-fss --without-sm` (the station automatically starts the necessary number of stagers to operate the cache areas). The options `-OAhost` and `--node-name` again helps to prevent network/firewall related issues.

A GridFTP daemon is started as well to allow outbound file transfer, i.e. files can be retrieved by other SAM stations from GridKa.

The complete server list file of the SAM station at the University of Karlsruhe (cdf-ekpka) is given below:

```
station prd v6_0_1_12 cdf-ekpka --min-delivery=1k  \
 --max-prefetched-files=5 --revival=fast                        \
  -OAhost ekpcdf2.physik.uni-karlsruhe.de -OAport 6789          \
 --pmaster-arg=-OAhost --pmaster-arg=ekpcdf2.physik.uni-karlsruhe.de \
 --pmaster-arg=--consumption-map=ekpplus.cluster::ekpcdf2.physik.uni-karlsruhe.de  \
 --log-file=station.log --preferred-loc=cdf.fzk.de                                 \
 --routing-station=\.\*::cdf-fzkka  --routing-user=cdf-ekpka --routing-group=cdf   \
 --common-timeout=180  --intrastation-timeout=180 --omit-loc=dcap                  \
 --consumer-wait-timeout=7200

fss prd  v6_0_1_12  cdf-ekpka  \
 -OAhost ekpcdf2.physik.uni-karlsruhe.de
```

```
stager prd  v6_0_1_12  cdf-ekpka  --with-fss --without-sm   \
   --max-transfers=7   -OAhost ekpcdf2.physik.uni-karlsruhe.de \
   --node-name=ekpcdf2.physik.uni-karlsruhe.de

gridftp prd v2_1_1
```

This server list file configures the station as a "satellite station" of the GridKa station: All files not yet cached at the University station are obtained from GridKa. This is done by using the options: `--routing-station= \.\*::cdf-fzkka --routing-user= cdf-ekpka --routing-group=cdf`. The option `--routing-station` tells the station to retrieve all files not yet in the own cache via GridKa, the next option determines which user name should appear in the log-files, etc. By policy, CDF agreed to use the station name here to be able to trace potential problems. The last option then determines which SAM group should be used for the accounting. The motivation for this approach is that if a user requests a file from a not yet cached dataset it is quite likely that other files are requested as well. As the capacity at GridKa is much higher than at the University, many more files can be kept there. Furthermore, as there is a direct link between the University and GridKa, file-transfers between these two stations are fast and free of charge.

A.3 Details of `sam_config`

The ups/upd product `sam_config` is used to further configure the SAM setup. It sets up other necessary products needed for the operation and sets necessary environment variables used by the various parts of SAM.

Note that the installation of the ups/upd area in the `sam` account and hence decoupled from the installation of the client code in the `cdfsoft` account.

```
__ENV_PREPEND__PYTHONPATH=/grid/fzk.de/home/sam-cdf/LocalPythonPath:
   /home/sam-cdf/products/sam_common_pylib/v6_7_0_5/NULL/SamUtility:
   /home/sam-cdf/products/sam_common_pylib/v6_7_0_5/NULL/SamCorba:
   /home/sam-cdf/products/sam_config/v4_2_34/NULL/conf:
   /home/sam-cdf/products/sam_idl_pylib/v6_7_0_0/NULL/lib
SAM_NAMING_SERVICE_IOR=IOR:010000002a00\ldots (GridKa naming service)
SAM_NAMING_SERVICE_IOR_1=IOR:010000002a00\ldots (FNAL naming service)
SAM_NAMING_SERVICE=cdf.fzk.de:9010
SAM_COMPILER_QUALIFIER=GCC-3.1
SAM_CP_ARBITRATOR_MODULE_FILE=sam_cp_protocol_arbitrator_gridka.py
SAM_CP_ARBITRATOR_CLASS_NAME=SamCpGridKaProtocolArbitrator
SAM_DB_SERVER_NAME=SAMDbServer.station_prd2:SAMDbServer
```

For the SAM station two naming services are used, first the local naming service at GridKa is used (specified by `SAM_NAMING_SERVICE_IOR`), then the central one (specified

by SAM_NAMING_SERVICE_IOR_1) in case the central services or other SAM stations need to be contacted.

The special variable __ENV_PREPEND__PYTHONPATH extends the environment variable PYTHONPATH and prepends the path at which the implementations of the arbitrators can be found. The variable SAM_CP_ARBITRATOR_CLASS_NAME specifies the name of the arbitrator used at GridKa, the variable SAM_CP_ARBITRATOR_MODULE_FILE in which physical file holds the implementation.

If the SAM products were compiled with a different compiler version than assumed in various places, the variable SAM_COMPILER_QUALIFIER can be used.

A.4 Implementation of the Arbitrator

This implementation of the arbitrator returns

- dcache_gridftp if accessing dCache at Fermilab
- sam_gridka_dccp if accessing dCache at GridKa
- sam_gridftp in all other occasions

as the file-transfer protocol. Note that the dCache mount-point /grid/fzk.de/mounts/pnfs/cdf/sam is treated as a mask, i.e. all paths containing this string will be taken as part of dCache at GridKa.

```
#!/usr/bin/env python
# Standard python modules.
import sys
import re
import os
import types
import time
import string

# SAM modules
import SAM
import DbServerProxy
import SamCpFileParser
import ModuleLoader
import CommonUtility
import ExceptionManager
import SAMError

from SamCpProtocolArbitrator import SamCpProtocolArbitrator
from SamCpProtocolArbitrator import SamCpDefaultProtocolArbitrator
import SamCpExceptions
```

```
################################################################################
# Constants.

TRUE_ = 1
FALSE_ = 0

DEFAULT_TIMESTAMP_FORMAT_ = "<%m/%d/%y %H:%M:%S>"

VERBOSE_FLAG_ = "v"
DEBUG_FLAG_ = "d"

PROTOCOL_CONSTRAINT_MAP_ = "ProtocolConstraintMap_"
DEFAULT_PROTOCOL_ = "DefaultProtocol_"

CONFIG_MODULE_ENV_VAR_ = "SAM_CP_ARBITRATOR_CONFIG_MOD"
CONFIG_DIR_ENV_VAR_ = "SAM_CP_ARBITRATOR_CONFIG_DIR"

CDFEN_FLAG_STRING_ = "/pnfs/cdfen/filesets"
GRIDKA_DCACHE_CDF_FLAG_STRING_ = "/grid/fzk.de/mounts/pnfs/cdf/sam"
################################################################################
```

```
    def __init__(self, *args):
        msg = "Using GridKa protocol arbitrator instead of the default"


        SamCpProtocolArbitrator.__init__(self, *args)
        self.srcLoc = self.getSrcLoc()
        self.srcStation = self.srcLoc.station()
        self.srcDir = self.srcLoc.dirname()

        self.destLoc = self.getDestLoc()
        self.destStation = self.destLoc.station()
        self.destDir = self.destLoc.dirname()

        self.fileName = self.srcLoc.basename()


    def getFileName(self):
        return self.fileName

    def getSourceLocation(self):
        return self.srcLoc

    def getDestinationLoction(self):
        return self.destLoc
```

```
def is_CDFEN(self, aPath):
    msg = "\t check if accessing cdfen at Fermilab"
    self.dbgprint(msg)
    returnValue = string.find(aPath, CDFEN_FLAG_STRING_) == 0
    msg = "\t accessing cdfen at Fermilab? %s" % returnValue
    self.dbgprint(msg)
    return returnValue

def is_GRIDKA_DCACHE_CDF(self,aPath):
    msg = "\t check if accessing dCache at GridKa for CDF"
    self.dbgprint(msg)
    returnValue = string.find(aPath,GRIDKA_DCACHE_CDF_FLAG_STRING_) == 0
    msg = "\t accessing dCache at GridKa? %s" % returnValue
    self.dbgprint(msg)
    return returnValue

def selectProtocol(self):
    msg = "Selecting protocol..."
    self.dbgprint(msg)

    sourceLocation = self.srcDir
    destLocation = self.destDir

    msg = "\t will copy file \t %s" % self.fileName
    self.dbgprint(msg)
    msg = "\t from station \t  %s \n \t \t to station \t %s"
             % (self.srcStation, self.destStation)
    self.dbgprint(msg)
    msg = "\t at source path \t %s \n \t \t to destination path \t
             %s" % (self.srcDir, self.destDir )
    self.dbgprint(msg)


    if self.is_CDFEN(sourceLocation):
        returnValue = "dcache_gridftp"
    elif self.is_GRIDKA_DCACHE_CDF(sourceLocation) or
         self.is_GRIDKA_DCACHE_CDF(destLocation):
        returnValue = "sam_gridka_dccp"
    elif self.is_GRIDKA_DCACHE_CDF(sourceLocation) and
         self.is_GRIDKA_DCACHE_CDF(destLocation):
        returnValue = "mv"
        print("cannot copy from dCache to dCache, only move")
    else:
        returnValue = "sam_gridftp"

    msg = "\t selected protocol: %s" % returnValue
    self.dbgprint(msg)
```

```
        return returnValue
```

A.5 Implementation of SamGridKaDccp

The new class SamGridKaDccp allowing SAM to access dCache at GridKa is given below. The main problem was to incorporate the different conventions used by the dCache- and SAM-developers: The dCache developers streamed all non-critical output (e.g. file transfer successful, achieved transmission rate, etc) to std::err and all critical output (e.g. failure, etc) to std::out, whereas the SAM developers treat any output in std::err as a failure. Hence the output of dccp has to be examined to determine if the file transfer was successful.

```
#!/usr/bin/env python
################################################################################
# Project   : SAM
# Package :
#
# Ulrich Kerzel, 2004-April-08 : initial revision
#
# local extensions to SamCpClasses.py (from the sam_cp package)
#
# inherit from SamCpClasses and extend
# functionality to :
# - access dCache at GridKa
#
# You also need to modify $SAM_CP_CONFIG_FILE to
# make this extension known
#
# N.B. You have to tailor sam_config to both extend the Python path
#      to find this module and to tell sam_cp to use it.
#
#
# This work is part of a development project, called SAM, which consists of a
# number of coordinated packages each named sam_xxxx .
#
# Notice of authorship, copyright status,  and terms and conditions, should
# the software eventually become available for use outside Fermilab, can be
# found in the README and LICENCE files in the top level directory of the main
# sam package.
################################################################################

import os
import sys
import string
import time
```

```
import re
import random
import stat

from SamCpFileParser import SamCpFileParser
import SamCpExceptions

from SamCpWrapper import SRC_, DEST_, DEBUG_, VERBOSE_
import CommonUtility
import SamCpOutputHandler
import SamCpEnstoreFileInfoHandler

# get Sam CP classes
import SamCpClasses
#from SamCpClasses import SamCpExecvpShellCommand
from SamCpClasses import SamCpProtocolBaseClass
class SamGridKaDccp(SamCpClasses.SamCpProtocolBaseClass):
    def __init__(self, args):
        SamCpProtocolBaseClass.__init__(self, args)
        self.className = self.__class__.__name__
        self.command = "/usr/local/dcap/bin/dccp"
        self.exitStatus = None
        self.args = args
        self.outputHandler = SamCpOutputHandler.SamCpOutputHandler()

    ##########################################################################
    def copy(self):
        exitStatus = SamCpProtocolBaseClass.copy(self)
        cmd = "sh"
        args = ["%s" % os.path.basename(cmd), "-c", "exit %s" % (exitStatus)]
        self.dbgprint("%s exiting with status %s" % (self.className,
                                                     exitStatus))
        os.execvp(cmd, args)

    ##########################################################################
    def authenticate(self):
        """
        No authentication performed in this class.
        """
        pass

    ##########################################################################
    def prepareFileTransfer(self):
        """
        Convert the file objects to strings that will be used
        in constructing the command
        """
        self.srcFile = self.src.parsedFileName()
```

```
        if self.src.isVirtualDomain():
            self.srcFile = self.src.path()

        self.destFile = self.dest.parsedFileName()
        if self.dest.isVirtualDomain():
            self.destFile = self.dest.path()

        self.dbgprint("prepareFileTransfer: srcFile = %s" % self.srcFile)
        self.dbgprint("prepareFileTransfer: destFile = %s" % self.destFile)

    ############################################################################
    def performFileTransfer(self):
        # we need to expand any environmental variables
        cmd  = os.path.expandvars(self.command)


        args = ["%s" % os.path.basename(cmd)]

        src  = os.path.expandvars("%s" % self.srcFile)
        dest = os.path.expandvars("%s" % self.destFile)

        args.append(src)
        args.append(dest)

        if ( self.getDebugArgs() ):
            args.append(self.getDebugArgs())

        if ( self.getVerboseArgs() ):
            args.append(self.getVerboseArgs())
        exitStatus = 0
        stdErr = ""
        stdOut = ""
        try:
            fullCmd = "%s %s" % (cmd, string.join(args[1:]))
            self.dbgprint("%s exec> %s" % (self.className, fullCmd))
            (stdOut, stdErr, exitStatus) = CommonUtility.ExecSubprocess(fullCmd)
            self.dbgprint("%s perform file transfer: exitStatus: %s" % (
                self.className, exitStatus))
            self.dbgprint("%s perform file transfer: stdOut: %s" % (
                self.className, stdOut))
            self.dbgprint("%s perform file transfer: stdErr: %s" % (
                self.className, stdErr))
            #
            # dccp version at GridKa writes successes to std::err try to
            # intercept and redirect to std::out
            # the output of dccp upon success is something like:
            # 1040960630 bytes in 175 seconds (5808.93 KB/sec)
            temp_1 = string.find(stdErr,"bytes")
```

```
            temp_2 = string.find(stdErr,"seconds")
            if exitStatus == 0 and temp_1 > 0 and temp_2 > 0:
                self.dbgprint("GridKA alert: dccp successfull but writes to std:err")
                self.dbgprint("      instead of std:out")
                self.dbgprint("      standard out returned from dccp: %s" % stdOut)
                self.dbgprint("      standard err returned from dccp: %s" % stdErr)
                self.dbgprint("      exit code    returned from dccp: %i" % exitStatus)
                self.dbgprint("      --> redefining std:out and std:err")
                tempOut = stdOut
                stdOut = "%s \n %s" % (tempOut,stdErr)
                stdErr = ""
            outputDict = {
                SamCpOutputHandler.FILE_TRANSFER_EXIT_STATUS_KEY_ : exitStatus,
                SamCpOutputHandler.FILE_TRANSFER_STD_ERR_KEY_ : stdErr,
                SamCpOutputHandler.FILE_TRANSFER_STD_OUT_KEY_ : stdOut,
                SamCpOutputHandler.ENSTORE_VOLUME_LABEL_KEY_ :  \
                   SamCpOutputHandler.UNKNOWN_DCACHE_VOLUME_LABEL_,
                SamCpOutputHandler.ENSTORE_VOLUME_OFFSET_KEY_ :  \
                   SamCpOutputHandler.UNKNOWN_DCACHE_VOLUME_OFFSET_
                }
            self.outputHandler.add(outputDict)

        except:
            # but if we failed, for any reason, we gotta pass it on up.
            msg = "File transfer failed: %s" % (sys.exc_info()[1])
            raise SamCpExceptions.SamCpFailure(msg)

        if ( exitStatus ):
            msg = "File transfer failed: %s " % (stdErr)
            raise SamCpExceptions.SamCpFileTransferFailure(msg)

        if not self.exitStatus:
            self.exitStatus = exitStatus
        else:
            self.exitStatus = self.exitStatus | exitStatus

    ###########################################################################
    # placeholder to be filled in by subclasses that know
    # how to post-process a file
    def postProcessFileTransfer(self):
        self.dbgprint("%s post process file transfer" % (self.className))
        self.outputHandler.writeOutput()

    ###########################################################################
    # placeholder to be filled in by subclasses that know
    # what to do with a file transfer status
    def setFileTransferStatus(self):
        pass
```

```
    ##########################################################################
    # placeholder to be filled in by subclasses that know
    # what to do with a file transfer status
    def getFileTransferStatus(self):
        self.dbgprint("%s get file transfer status: exitStatus %s" % (
            self.className, self.exitStatus))
        return self.exitStatus

    ##########################################################################
    # placeholder to be filled in by subclasses that know
    # how to turn on debugging
    def getDebugArgs(self):
        return None

    ##########################################################################
    # placeholder to be filled in by subclasses that know
    # how to turn on verbosity
    def getVerboseArgs(self):
        return None
```

A.6 Examples for creating a SAM dataset

This section gives a few examples of how to create a SAM dataset using metadata associated with the files.

The command used follows the pattern:

```
sam create dataset definition --defname=<dataset name> \
 --group=test --defdesc=''some description''           \
 --dim=''selection criteria''
```

The option `--defname` determines the name of the dataset created (e.g. kerzel_dataset_001), since CDF does not yet use groups for accounting, etc., the group "test" is used (for historical reasons), `defdesc` is some free text describing the created dataset, `--dim` contains the actual selection criteria.
It is strongly advised to test the selection criteria using

```
sam tranlslate constraints --dim=''selection criteria''
```

The following examples illustrate how to use and combine several selection criteria. Only very few of the available criteria are used in the examples, however most use-cases can be covered with them.

- `--dim=''cdf.dataset jpmm0d ‘‘` selects all files from the di-muon dataset `jpmm0d` as defined by the DFC

- `--dim=''file_name A or file_name B ‘‘` selects the files A and B (note that this operation is computationally very expensive in the database and should not be use for more than a few files. If too many files are used the central system may be overloaded and the whole SAM system becomes globally inoperative)

- `--dim=''cdf.dataset A and (run_nr > B and run_nr < C) ‘‘` selects all files from the DFC dataset A with runs in the range (B,C).

- `--dim=''__set__ A minus (project_name B and consumed_status consumed and consumer C) ‘‘` selects all files from the previously defined SAM dataset A which have not been processed by the SAM project B by user C. The typical use-case of this selection criteria is define a dataset holding all files which have not yet been processed by the "main" project running over many files, e.g. due to crashes, etc.

- `--dim=''project_name A and cpid B ‘‘` selects all files processed by a specific job on a worker-node with process number B (called "consumer process ID") which was part of project A. The typical use-case is to re-run a specific job, eventually excluding a specific file causing problems.

Appendix B

Further details about electron identification

This appendix summarises further details about the various networks used. For each network, the complete list of input variables is given together with an illustration of the correlation between the variables. Dark colours indicate highly correlated variables, light colours represent variables with low correlation.

B.1 Recreating calorimeter objects

Overview.— The `CdfEmObjectColl` collection contains information about the clusters and towers found in the calorimeters. It is created by default during the production processing of the raw data, i.e. when the "low level" objects such as tracks, calorimeter information, etc. are determined from the raw data. The default settings in the clustering module of the electromagnetic calorimeter already suppress a significant amount of hadronic background. Unfortunately, these settings also discard many of the low energetic signal particles. To be able to identify these particles and include them in the later analyses, the `CdfEmObjectColl` collection needs to be recreated with lower threshold settings.

Needed modules and parameters.— The original `CdfEmObjectColl` collection is discarded by adding it to the `dropList` of the `DHInput` module Then this collection is recreated by running the modules `EmClusterModule` (which rebuilds the calorimeter clusters) and `CdfEmObjectModule` (which creates the collection used in the electron identification toolbox).

The parameters essentially lower the thresholds such that the calorimeter "fires" more often. The results in many more found electrons, the worse signal to background ratio is handled by the neural network.

```
module talk DHInput
   dropList add CdfEmObjectColl
exit

module enable EmClusterModule
module talk EmClusterModule
  verbose         set f
  production      set t
  # default is 2 (GeV)
  seedEMEtMin      set 0.1
  # default is 2 (GeV)
  clusterEMEtMin   set 0.1
  clusterHad2EMMax set 999
exit

module enable CdfEmObjectModule
module talk CdfEmObjectModule
   verbose          set f
   production       set t
exit
```

B.2 Further details for the `KappaNet`

The following input variables have been used in the curvature change detection network `KappaNet`:

1. (target)
2. $\kappa/\sigma(\kappa)$ at PortCard layer 0 ($e^{\pm}$ hypothesis)
3. $\kappa/\sigma(\kappa)$ at PortCard layer 0 ($\pi^{\pm}$ hypothesis)
4. $\kappa/\sigma(\kappa)$ at PortCard layer 2 ($e^{\pm}$ hypothesis)
5. $\kappa/\sigma(\kappa)$ at PortCard layer 2 ($\pi^{\pm}$ hypothesis)
6. $\kappa/\sigma(\kappa)$ at COT inner wall ($\pi^{\pm}$ hypothesis)
7. $\kappa/\sigma(\kappa)$ at COT inner wall ($e^{\pm}$ hypothesis)
8. $\chi^2/ndof$ for the full fit in forward direction ($\pi^{\pm}$ hypothesis)
9. $\chi^2/ndof$ for the full fit in backward direction ($\pi^{\pm}$ hypothesis)

10. radiation length at PortCard layer 2 ($\pi^{\pm}$ hypothesis)
11. χ^2 of fit at COT inner wall ($e^{\pm}$ hypothesis)
12. χ^2 of fit at PortCard layer 0 ($e^{\pm}$ hypothesis)
13. χ^2 of the the full fit in backward direction ($\pi^{\pm}$ hypothesis) - χ^2 of the the full fit in backward direction (forward direction)
14. χ^2 of the the full fit in forward direction ($\pi^{\pm}$ hypothesis) - χ^2 of the the full fit in forward direction (forward direction)

The NeuroBayes® neural network allows to preprocess individual variables to prepare them optimally for the subsequent network training. The preprocessing options are described in detail in the NeuroBayes® user's guide [60]. Variables 2 to 9 are preprocessed with option 34, which applies a regularised spline fit to the distribtion of the input variable. The actual input variable used for the training is then based on the result of the spline fit. Furthermore, the default value at -999, which comes from the fact that these variables are not avialable for all candidates, is treated separately in this preprocessing. The remaining variables are preprocessed with option 14 which uses the same preprocessing steps as option 34 except that these variables are available for each candidate and hence no default value needs to be treated separately.

B.3 Further details for the SENet

Below the full list of input variables for the soft electron identification network SENet is given. For completeness, the names of the actual variables for the calorimeter based quantities are given as well.

1. (target)
2. CES χ^2_z (based on observed vs. fitted charge, strip-cluster, CES_Chi2Z_scaled)
3. CES χ^2_x (based on observed vs. fitted charge, wire-cluster, CES_Chi2X_scaled)
4. distance Δ_x between electron candidate track and CES cluster (wire cluster, CES_DeltaX)
5. distance Δ_z between electron candidate track and CES cluster (strip cluster, CES_DeltaZ)
6. CEM EM energy / $|p|$ (CEM_EEmDivP)
7. CES raw energy / $|p|$ (strip cluster, CES_EStrip/$|p|$)
8. EM transverse energy (CEM_EmEt)
9. corr. CES raw energy (wire cluster, CES_EWireCorr)

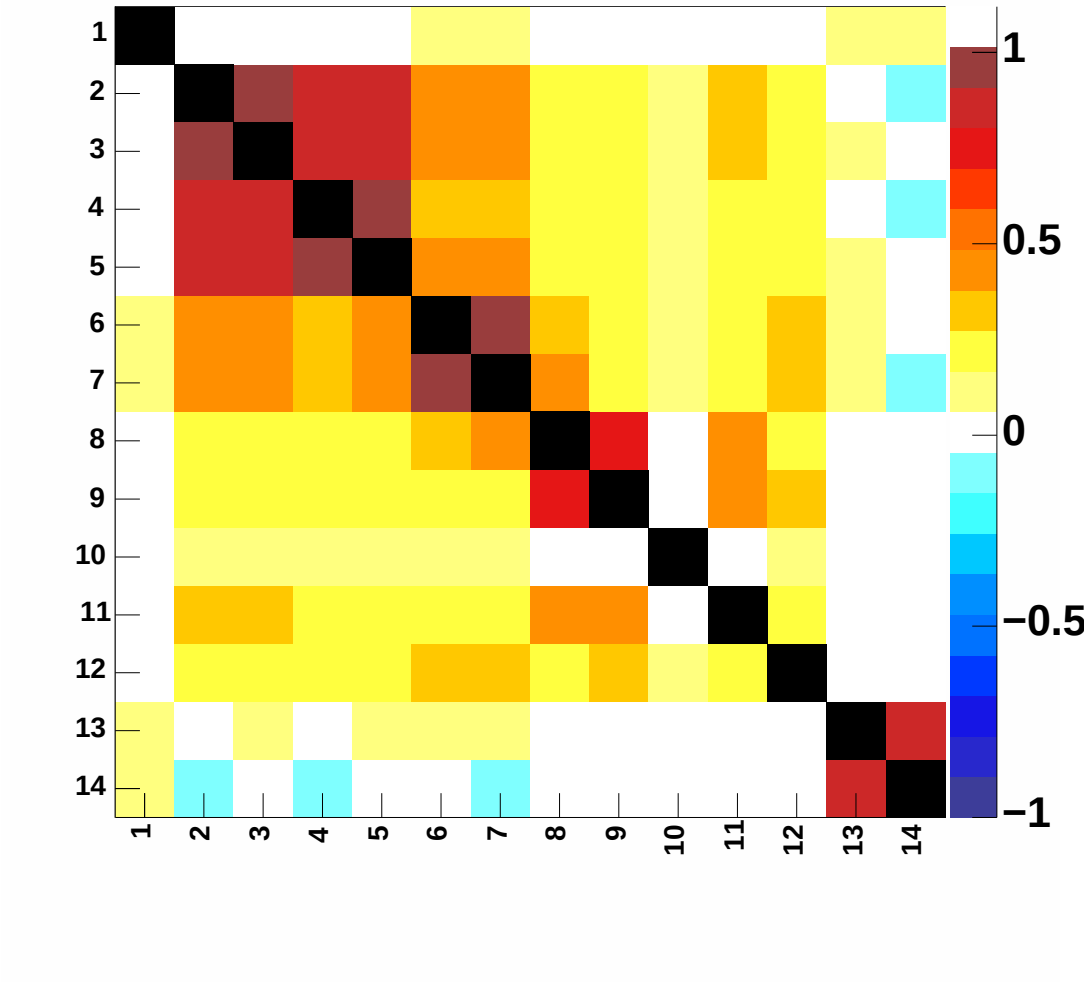


Figure B.1: Correlation matrix for the input variables for the `KappaNet` (for the case $p_t > 2$ GeV/c).

10. hadron energy / EM energy (CEM_HadEm)
11. raw energy(strip) /corr. raw energy (wire) in CES (CES_EStripDivEWireCorr)
12. corr. position of CES cluster (wire cluster) (CES_X)
13. y coordinate of electron candidate track extrapolated to CPR (CPR_xWire)
14. corr. position of CES cluster (strip cluster, CES_Z)
15. z coordinate of electron candidate track extrapolated to CPR (CPR_zWire)
16. $q \times d_0$
17. $\Delta|\coth\theta|$
18. conversion fit probability
19. conversion separation
20. KappaNet output
21. COT dE/dx likelihood ratio for electron hypothesis
22. COT dE/dx likelihood ratio for pion hypothesis
23. COT dE/dx likelihood ratio for proton hypothesis
24. COT dE/dx likelihood ratio for muon hypothesis
25. COT dE/dx likelihood ratio for kaon hypothesis
26. ToF pull for electron hypothesis
27. ToF pull for kaon hypothesis
28. ToF pull for muon hypothesis
29. ToF pull for pion hypothesis
30. ToF pull for proton hypothesis
31. corr. CPR energy (CPR_EwireCorr)

Exploiting the feature of the NeuroBayes® neural network that individual variables can be preprocessed prior to the network training, all variables are first treated with preprocessing option 34, which applies a regularised spline fit to the distribtion of the input variable. The actual input variable used for the training is then based on the result of the spline fit. Furthermore, the default value at -999, which comes from the fact that these variables are not avialable for all candidates, is treated separately in this preprocessing.
The variables CES χ^2_z, CES Δ_x, CES Δ_z and E(strip)/E(wire) are reweighted to achieve better agreement between data and simulated events.

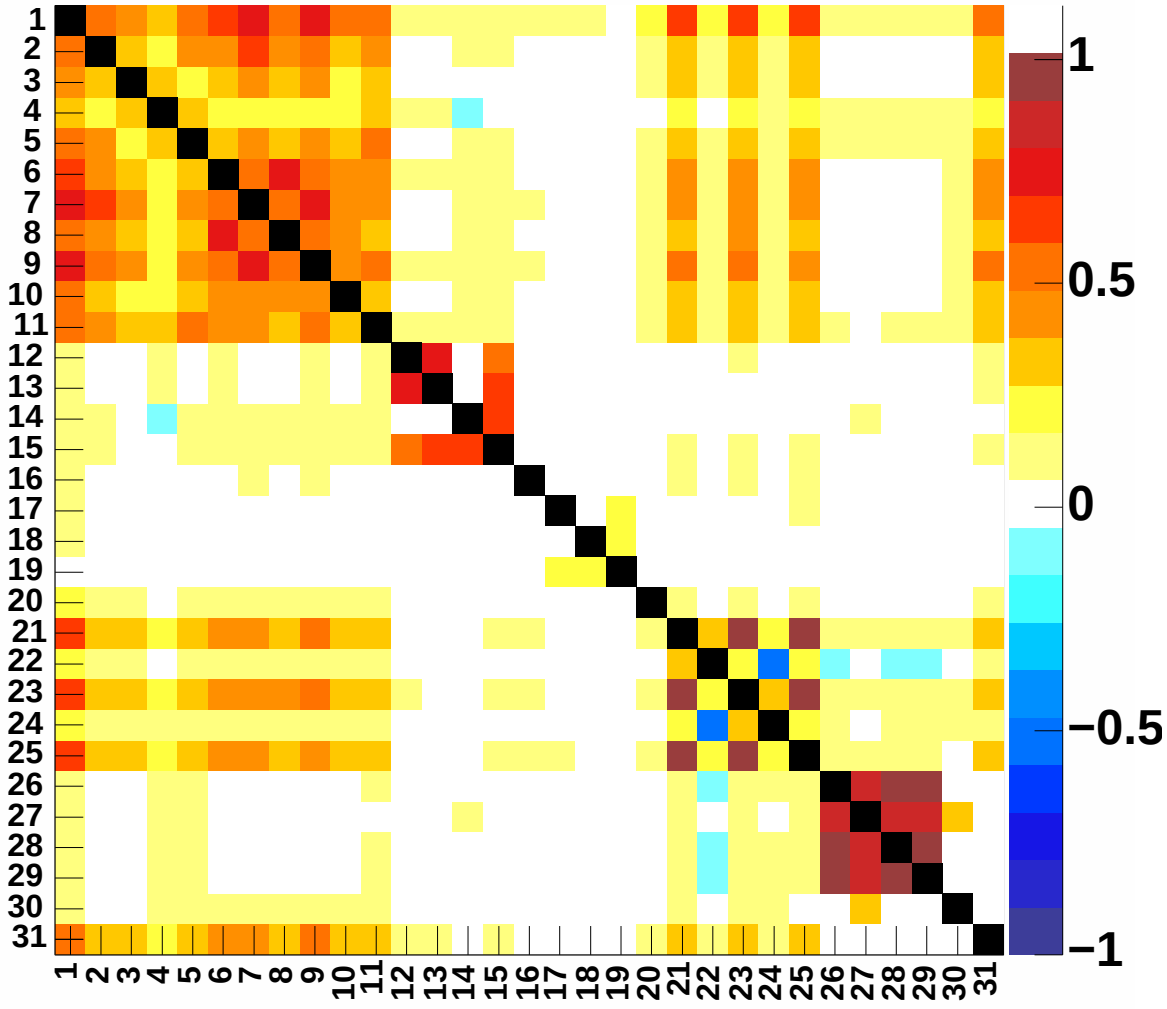


Figure B.2: Correlation matrix for the input variables for the soft electron identification network (for the case $p_t > 2$ GeV/c).

B.4 Further details for the ConvNet

These variables have been used as input to the conversion identification network:

1. (target)
2. network output of the soft electron network
3. $q \times d_0$
4. $\Delta|\coth\theta|$
5. conversion fit probability
6. conversion separation
7. number of hits in the silicon detector

Again exploiting the preprocessing options for individual variables, the following options have been used:

1. (target, no option)
2. option 14 (regularised spline fit, no default value)
3. option 14 (regularised spline fit, no default value)
4. option 34 (regularised spline fit with default value)
5. option 34 (regularised spline fit with default value)
6. option 34 (regularised spline fit with default value)
7. option 18 (ordered class)

Further details about these options can be found in the NeuroBayes® user's guide [60].

B.5 Further details for the BremsIDNet

The full list of input variables for the BremsIDNet is given below. Each quantity appears twice, one is taken from the electron candidate, the other from the positron candidate.

1. (target)
2. impact parameter d_0 significance w.r.t. beamline for e^-

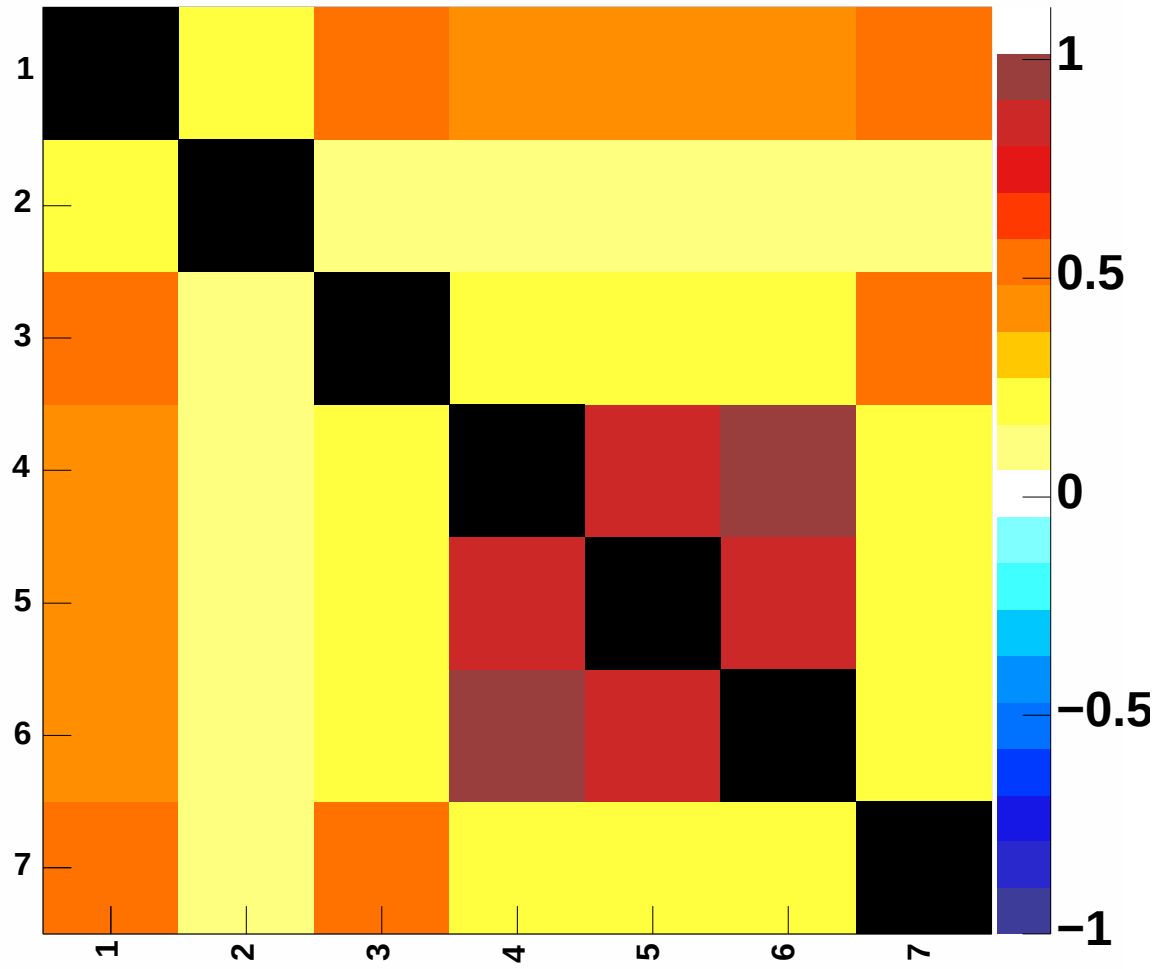


Figure B.3: Correlation matrix for the input variables for the conversion identification network (for the case $p_t > 2$ GeV/c).

3. χ^2 of e^- track fit
4. χ^2 of e^- track fit, SVX part
5. ($\delta\kappa$ at COT inner wall)$\times 10^5$ (e^- hypothesis)
6. ($\delta\kappa$ at PortCard layer 0)$\times 10^5$ (e^- hypothesis)
7. ($\delta\kappa$ at PortCard layer 2)$\times 10^5$ (e^- hypothesis)
8. $\kappa/\sigma(\kappa)$ at COT inner wall (e^- hypothesis)
9. $\kappa/\sigma(\kappa)$ at PortCard layer 0 (e^- hypothesis)
10. $\kappa/\sigma(\kappa)$ at PortCard layer 2 (e^- hypothesis)
11. max. corrected residual of SVX hit w.r.t helix fit for e^- track fit
12. impact parameter d_0 significance w.r.t. beamline for e^+
13. χ^2 of e^+ track fit
14. χ^2 of e^+ track fit, SVX part
15. ($\delta\kappa$ at COT inner wall)$\times 10^5$ (e^+ hypothesis)
16. ($\delta\kappa$ at PortCard layer 0)$\times 10^5$ (e^+ hypothesis)
17. ($\delta\kappa$ at PortCard layer 2)$\times 10^5$ (e^+ hypothesis)
18. $\kappa/\sigma(\kappa)$ at COT inner wall (e^+ hypothesis)
19. $\kappa/\sigma(\kappa)$ at PortCard layer 0 (e^+ hypothesis)
20. $\kappa/\sigma(\kappa)$ at PortCard layer 2 (e^+ hypothesis)
21. max. corrected residual of SVX hit w.r.t helix fit for e^+ track fit

To prepare these variables optimally for the subsequent network training, all variables have been preprocessed with option 14, which applies a regularised spline fit to the distribtion of the input variable. The actual input variable used for the training is then based on the result of the spline fit. The preprocessing options are described in detail in the NeuroBayes® user's guide [60].

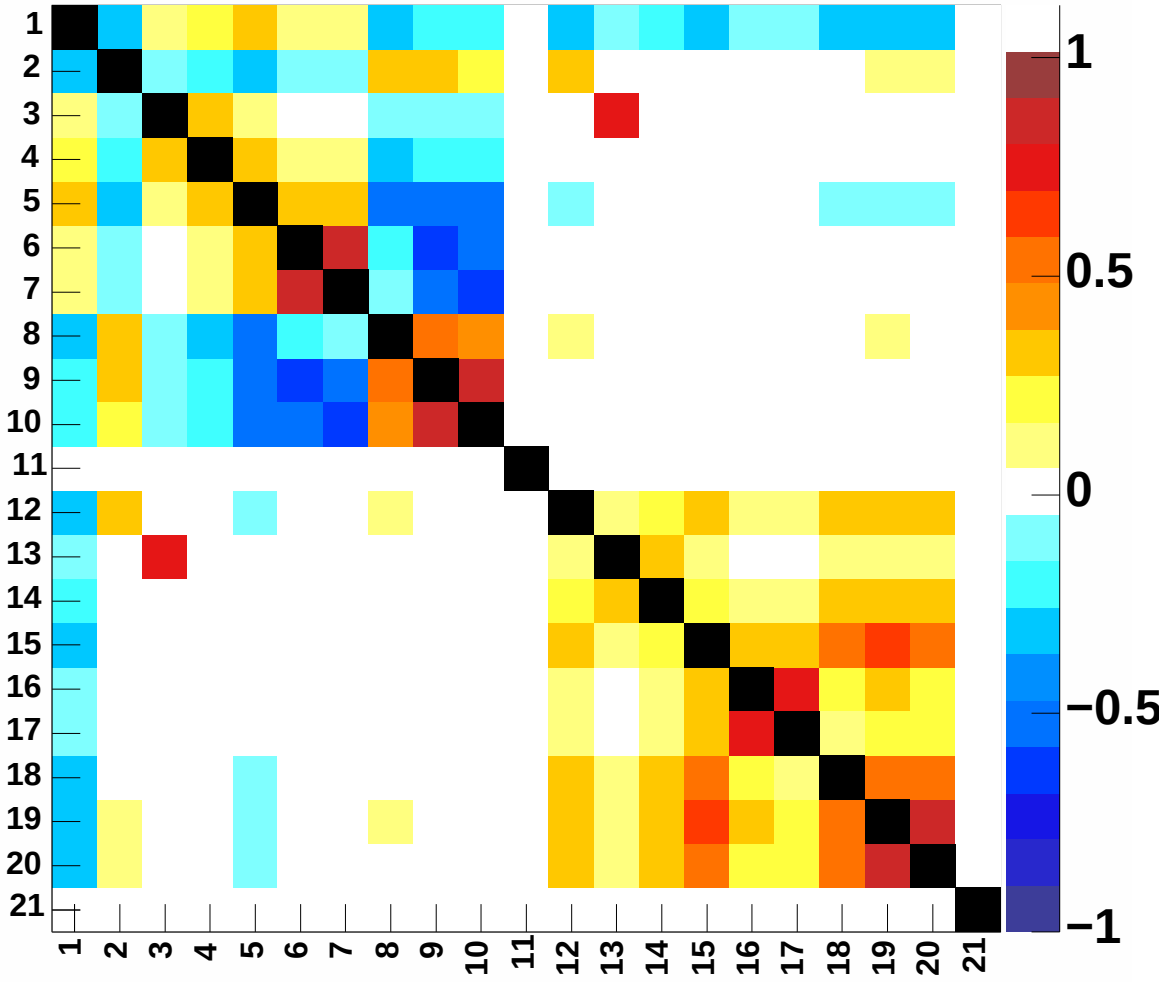


Figure B.4: Correlation matrix for the input variables for the `BremsID` network.

B.6 Further details for the eFromB network

These variables have been used as input to the network identifying whether an electron originates from a B hadron decay:

1. (target)
2. network output of the soft electron network
3. network output of the conversion identification network
4. transverse momentum p_t of the same-side B meson candidate
5. decay length L_{xy} of the same-side B meson candidate
6. mass of the same-side B meson candidate
7. $q \times d_{0,corr}$
8. transverse momentum relative to same side B jet ($p_t(rel)$)
9. Significance of the impact parameter d_0 w.r.t. the beam
10. transverse momentum

To prepare these variables optimally for the subsequent network training, all variables have been preprocessed with option 14, which applies a regularised spline fit to the distribtion of the input variable. The actual input variable used for the training is then based on the result of the spline fit. The preprocessing options are described in detail in the NeuroBayes® user's guide [60].

B.7 Agreement between data and simulation

This section illustrates the agreement between the distributions of the input variables used for the electron identification networks between data and the realistic simulation [65] used to train the networks. All simulated events have been used to obtain the distributions, the data distributions have been obtained by analysing the dataset `xbel0d` which requires a minimal energy in the electromagnetic calorimeter and the presence of a B hadron candidate. Each figure is composed of three distributions:

- variable reconstructed using data (points with error bars)
- variable reconstructed using simulated events for signal and background (blue histogram)

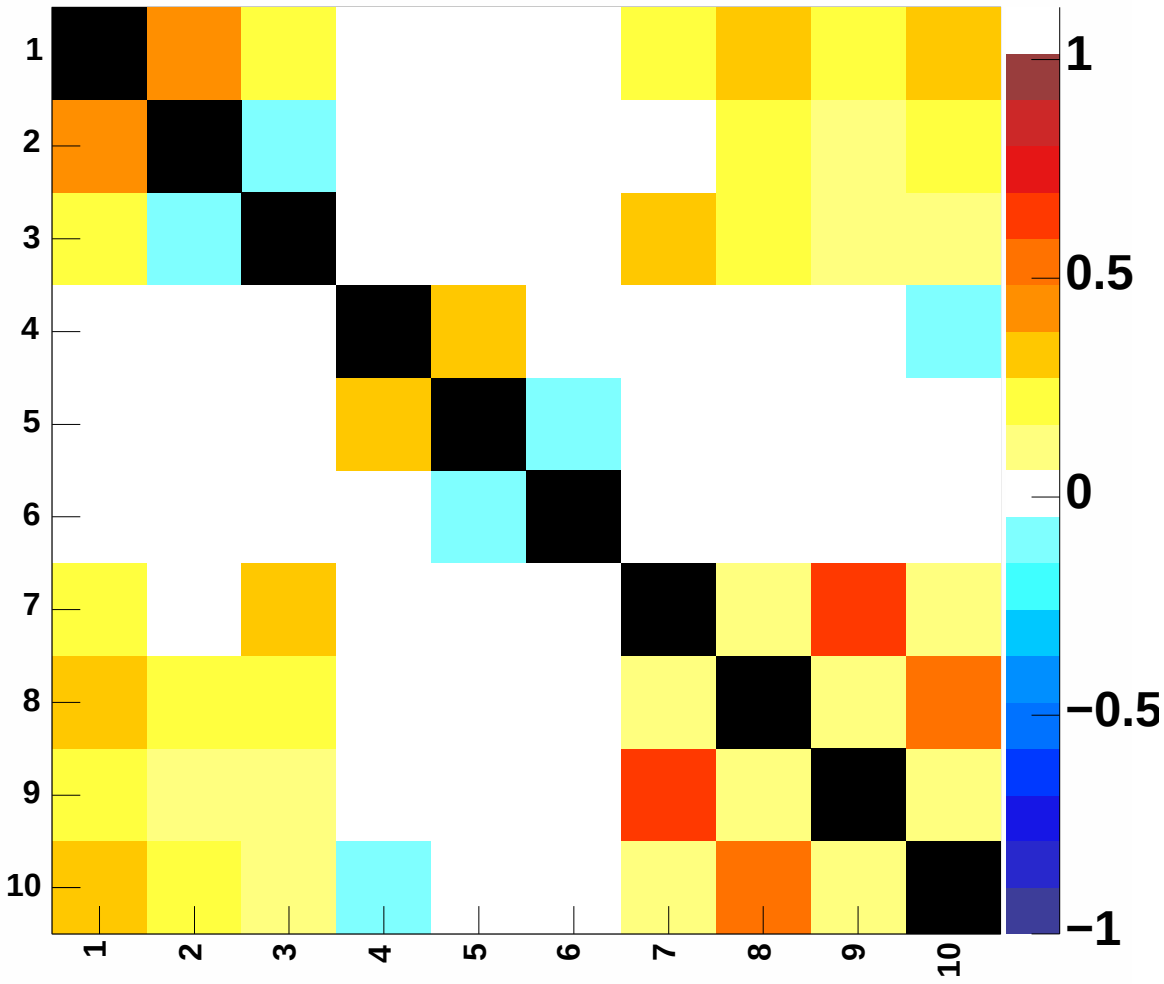


Figure B.5: Correlation matrix for the input variables for the network identifying electrons from B meson decays

- variable reconstructed using simulated events for signal only (green histogram)

This illustrates the different behaviour of the input variables for signal (i.e. $e^{\pm}$) and background (everything else). The distribution taken from simulated events are normalised such that the total number of entries corresponds to the number of entries in the data distribution.

Following the approach developed in [83], the variables are compared in the following way:

1. reconstruct the ℓ+SVT candidate and require $2 < m(\ell + \mathrm{SVT}) < 4$ GeV/c^2
2. fill histogram h_1 for the respective quantity requiring $\delta > 0$
3. fill histogram h_2 for the respective quantity requiring $\delta < 0$
4. subtract: $h_1 - h_2$

The quantity δ is defined as:

$$\delta = |d_0| sign(\vec{d_0} \cdot \vec{p}_{\ell+SVT})$$

where d_0 is the impact parameter of the SVT candidate. This approach is applied to both simulated events and data in the same way prior to the comparison and enriches events containing semi-leptonic B hadron decays. Only after this background subtraction technique has been applied are the simulated events (including the trigger simulation) and the data expected to behave similarly. In detail, events containing charm quarks are suppressed by requiring that the mass of the ℓ+SVT candidate is greater than 2 GeV/c^2 and less than 4 GeV/c^2. Events where the lepton ℓ and the SVT track do not originate from the same vertex are removed by the background subtraction procedure. This assumes that the distributions obtained from these background processes are distributed symmetric around zero. The plots are shown for the case of $p_t > 2$ GeV/c.

Generally speaking, the distribution of the input variables in the simulated events agrees well with the distribution observed in the data. However, the quantities χ^2_z(CES), Δ_x(CES), Δ_z(CES) and E_{strip}/E_{wire}(CES) show larger deviations. To account for this discrepancy and to bring the distribution of these variables in simulated events into agreement with the distribution seen in the data, weights are constructed in the following way: Each of the variables is filled into a histogram to obtain the corresponding distribution. This is done for both simulated events and data using events recorded by the e + SVT trigger. After normalising these distributions to unit area, they are smoothed to decrease the sensitivity to statistical fluctuations. An event weight is then obtained by taking the ratio of these distributions. The resulting distribution is again smoothed to decrease the sensitivity to small fluctuations. As a

cross-check, this procedure has also been applied to events recorded by the μ + SVT trigger which leads to the same conclusion. The individual weights obtained for these three variables are then combined to a global event weight which is then passed to NeuroBayes®, exploiting the feature that the network is able to train with weighted events.

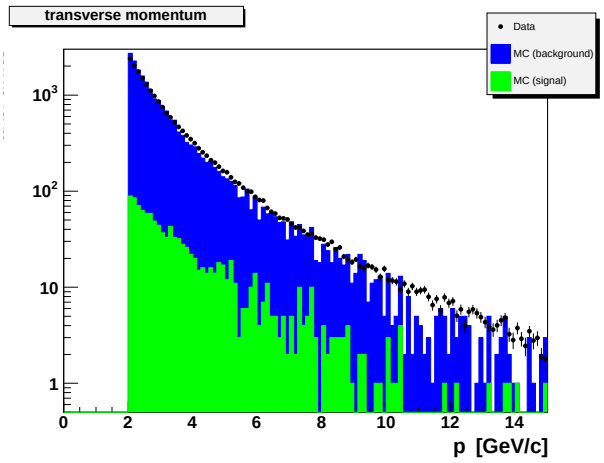
The distributions of the input variables are shown prior to weighting for the quantities χ^2_z(CES), Δ_x(CES), Δ_z(CES) and E_{strip}/E_{wire}(CES), whereas the output of the soft-electron identification network is shown *after* applying the above weights.

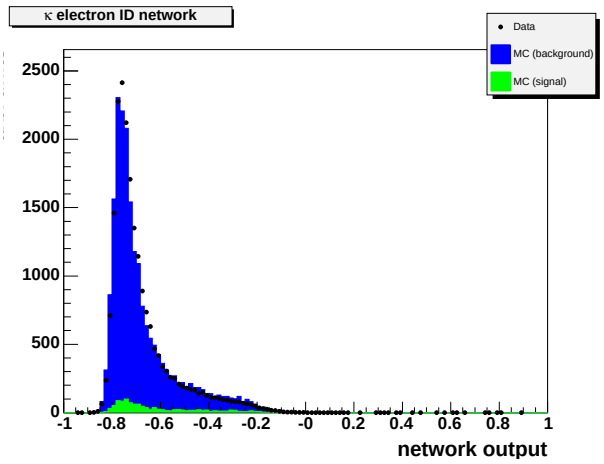
The simulated distribution of the transverse momentum of all particles agrees well with the distribution measured in the data as illustrated by the top left plot on the next page.

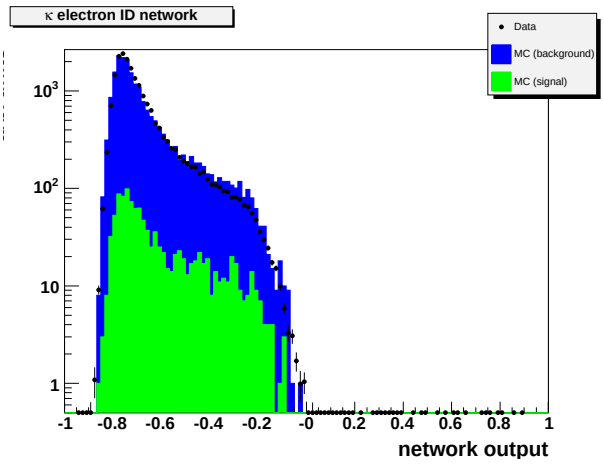
The next two sets of plots show the agreement of the network output between the realistic simulation and the data for the curvature change identification network `KappaNet` (plots in the middle) and the soft electron identification network `SENet` (lower set of plots). A network output close to +1 means that the considered candidate is an electron whereas a network output of −1 signals a background (i.e. not electron) candidate. Furthermore, the different scale on the y-axis of the various plots illustrate one of the features of the set of input variables: Not all variables are filled for all candidates, often only a subset of variables are filled, whereas the others are set to a default value. Conventional methods based on e.g. simple cuts are not able to deal with this kind of information. The NeuroBayes® neural network however offers a dedicated preprocessing for these kind of variables which a allows to optimally exploit the data taken by the detector. Further information about these special preprocessing options can be found in the user's guide [60].

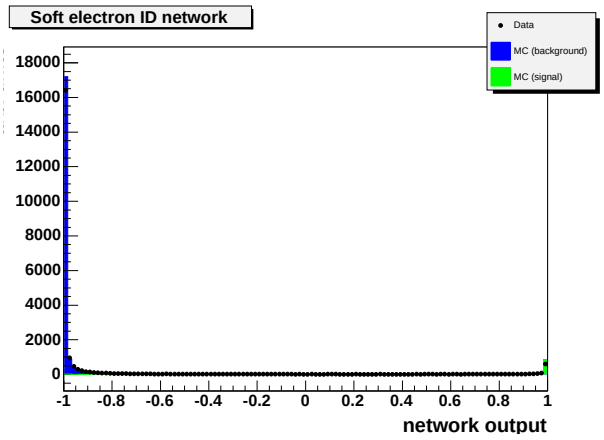
The plots for the `KappaNet` identifying electrons by change of curvature as they traverse material layers illustrate the difficult task the network has to learn: The changes in the input variables are rather small and though they behave differently for electrons and other particles, the differences are rather subtle.

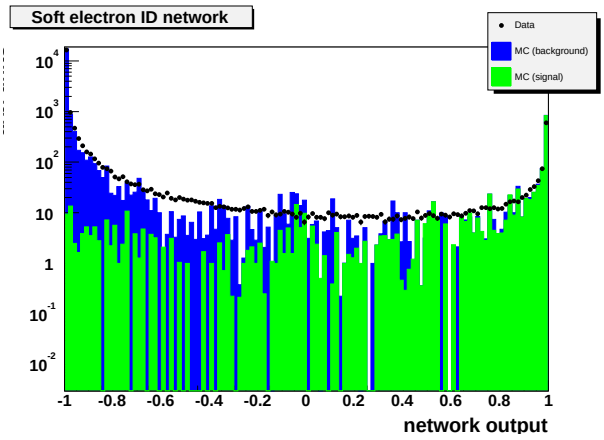
The plots for the soft electron identification network emphasise one of the key challenges the network has to face: The electron signal is dominated by more than a factor 10 higher background. The network reproduces the data well.


transverse momentum
Data
MC (background)
MC (signal)
p [GeV/c]

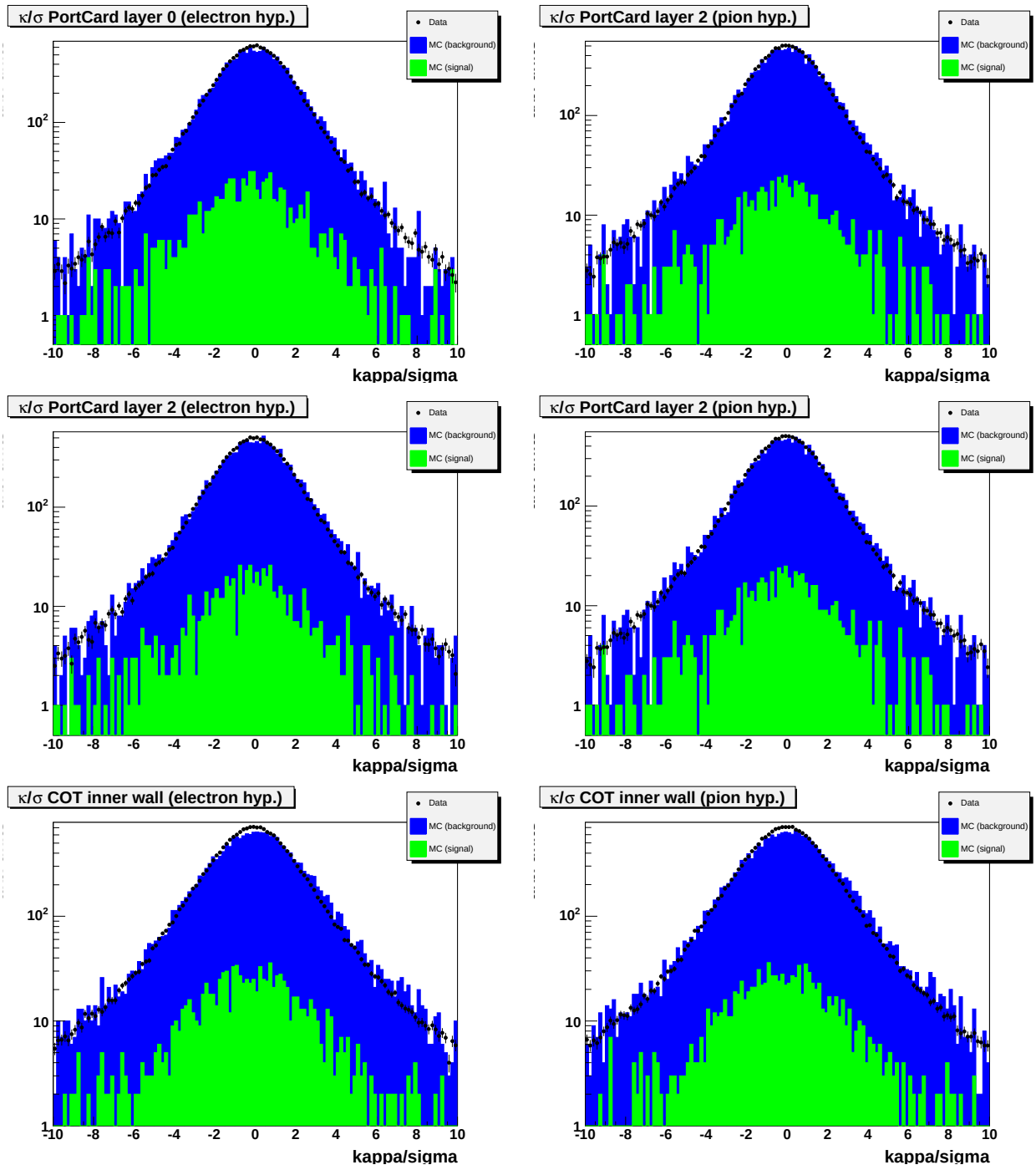

κ electron ID network
Data
MC (background)
MC (signal)
network output

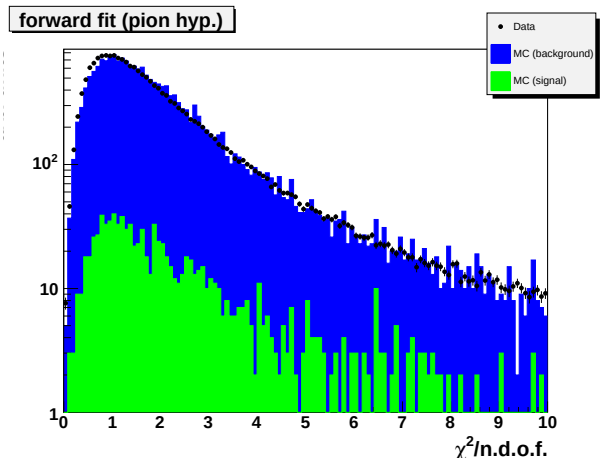

κ electron ID network
Data
MC (background)
MC (signal)
network output

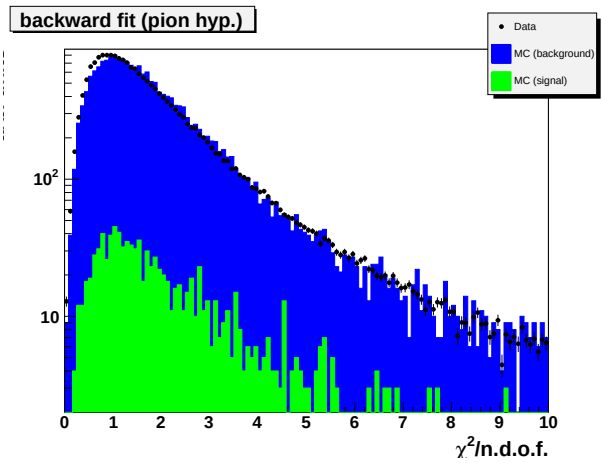

Soft electron ID network
Data
MC (background)
MC (signal)
network output

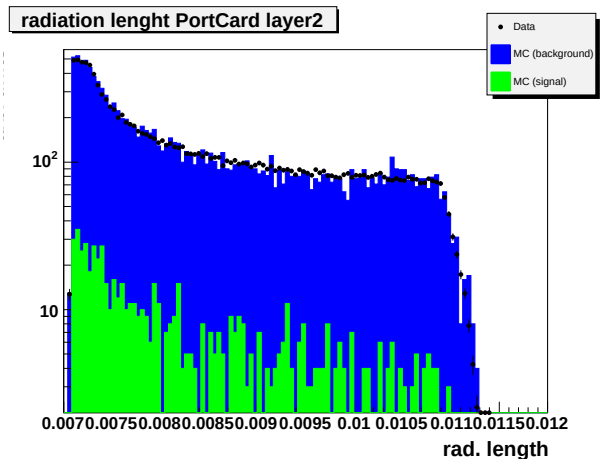

Soft electron ID network
Data
MC (background)
MC (signal)
network output

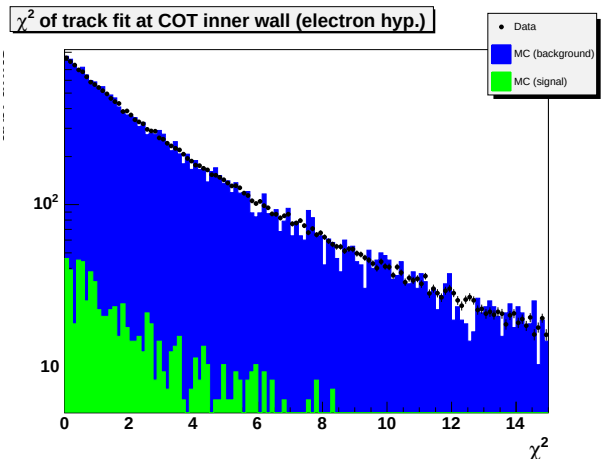
B.8 Variables used for `KappaNet`

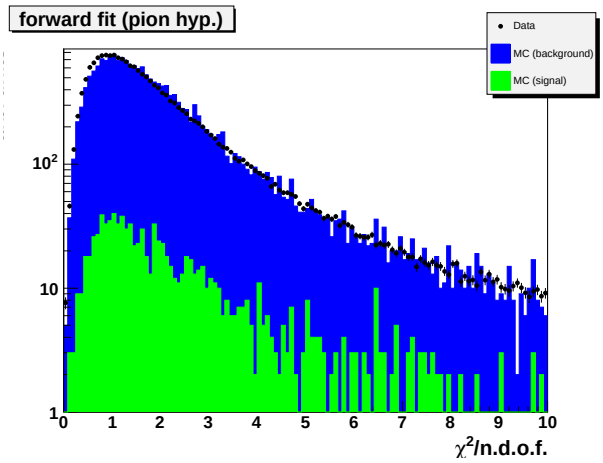


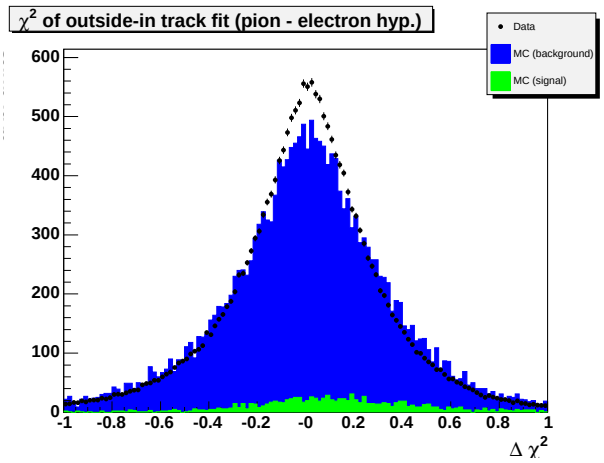

forward fit (pion hyp.)
Data
MC (background)
MC (signal)
10^2
10
1
0 1 2 3 4 5 6 7 8 9 10
χ^2/n.d.o.f.

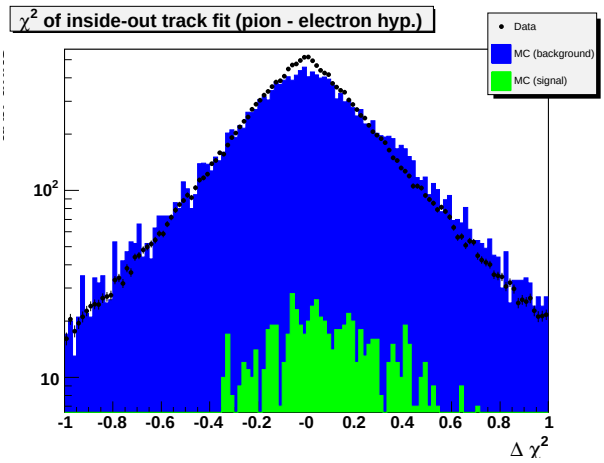

backward fit (pion hyp.)
Data
MC (background)
MC (signal)
10^2
10
0 1 2 3 4 5 6 7 8 9 10
χ^2/n.d.o.f.


radiation lenght PortCard layer2
Data
MC (background)
MC (signal)
10^2
10
0.007 0.0075 0.008 0.0085 0.009 0.0095 0.01 0.0105 0.011 0.0115 0.012
rad. length

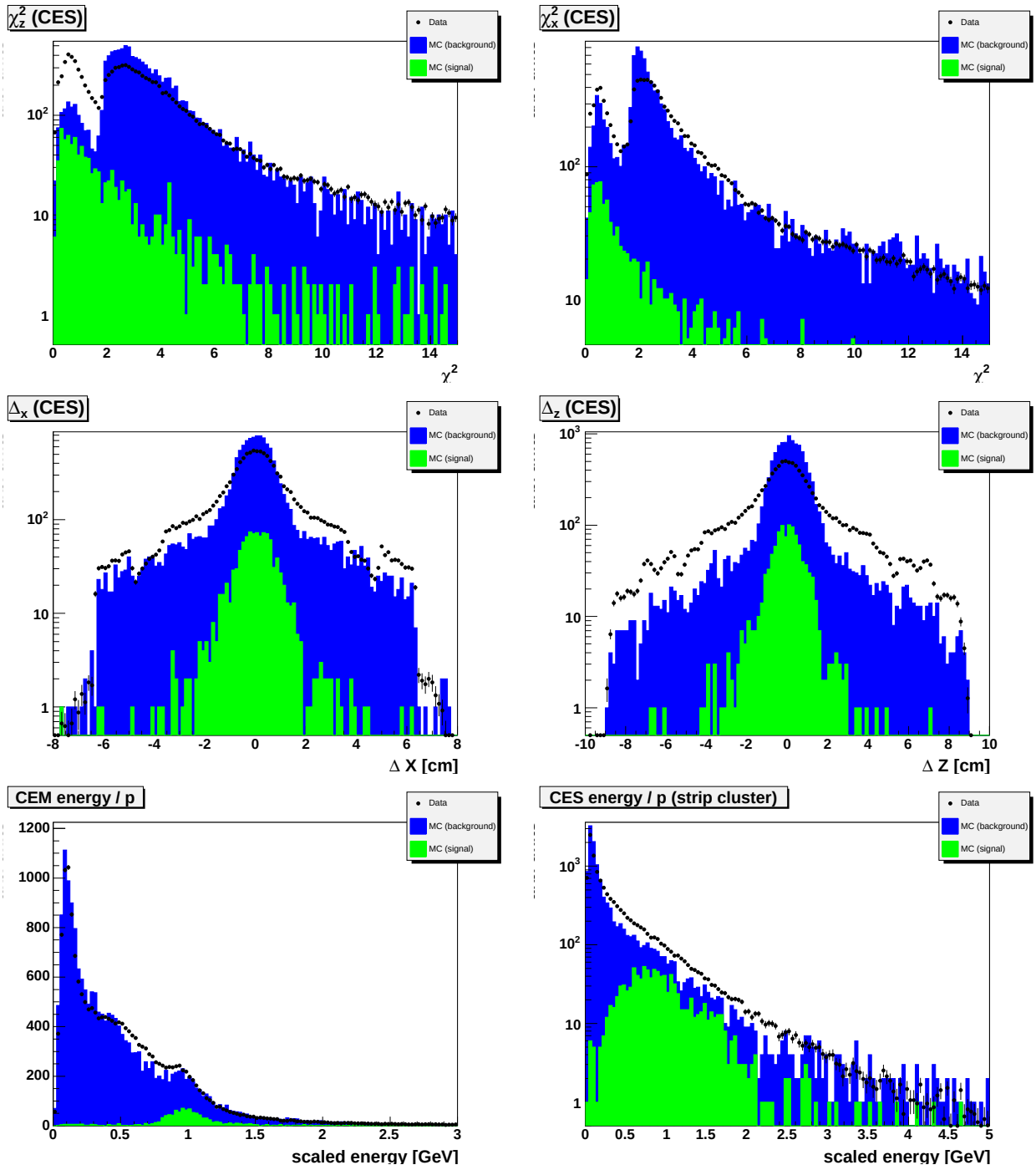

χ^2 of track fit at COT inner wall (electron hyp.)
Data
MC (background)
MC (signal)
10^2
10
0 2 4 6 8 10 12 14
χ^2

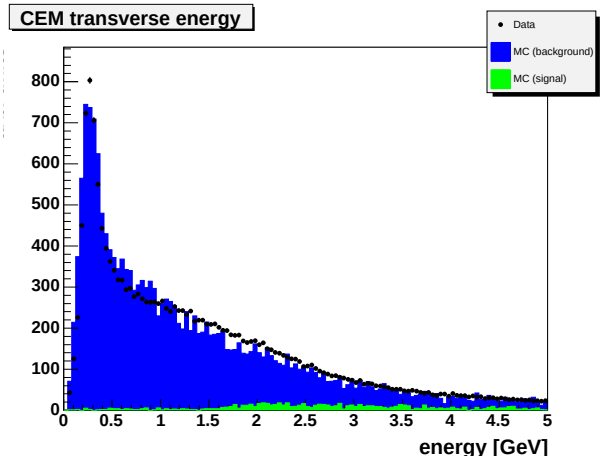

forward fit (pion hyp.)
Data
MC (background)
MC (signal)
10^2
10
1
0 1 2 3 4 5 6 7 8 9 10
χ^2/n.d.o.f.

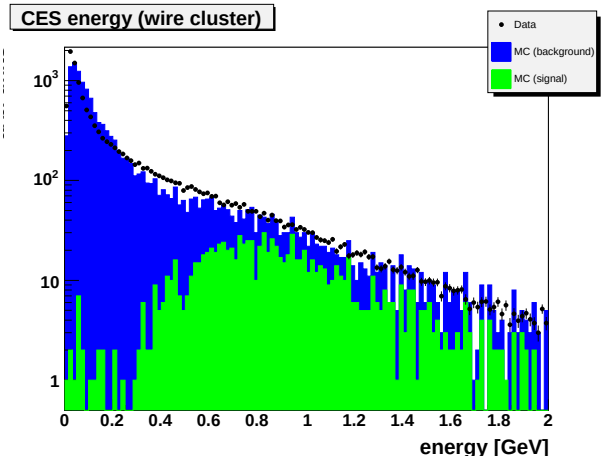

χ^2 of outside-in track fit (pion - electron hyp.)
Data
MC (background)
MC (signal)
600
500
400
300
200
100
0
-1
-0.8
-0.6
-0.4
-0.2
-0
0.2
0.4
0.6
0.8
1
$\Delta\ \chi^2$

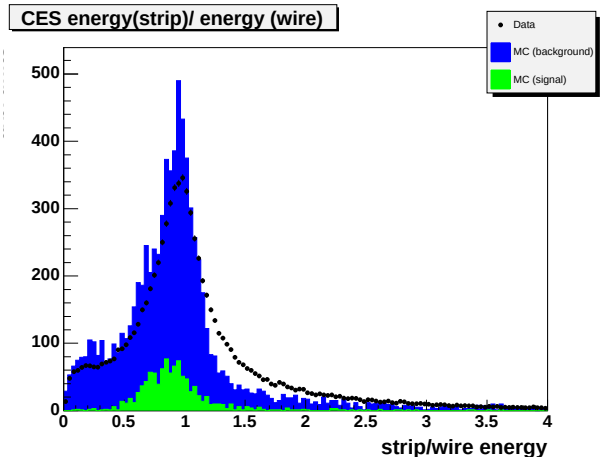

χ^2 of inside-out track fit (pion - electron hyp.)
Data
MC (background)
MC (signal)
10^2
10
-1
-0.8
-0.6
-0.4
-0.2
-0
0.2
0.4
0.6
0.8
1
$\Delta\ \chi^2$

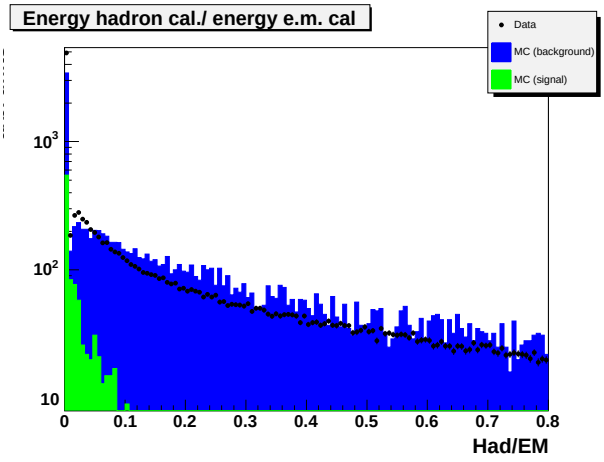
B.9 Variables used for SENet

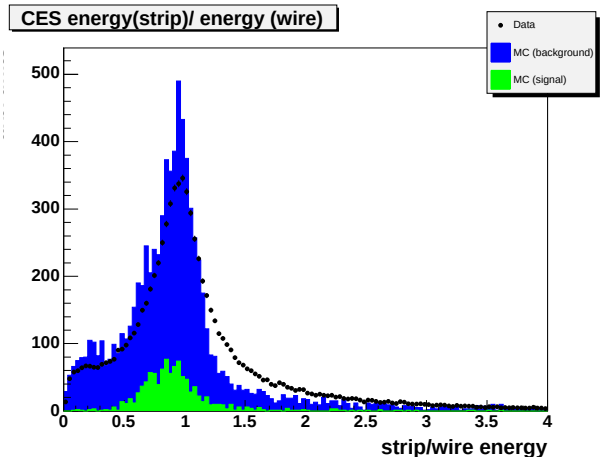


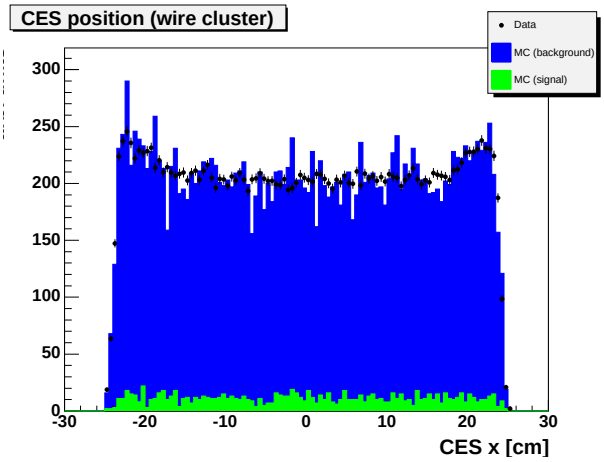

CEM transverse energy
Data
MC (background)
MC (signal)
energy [GeV]

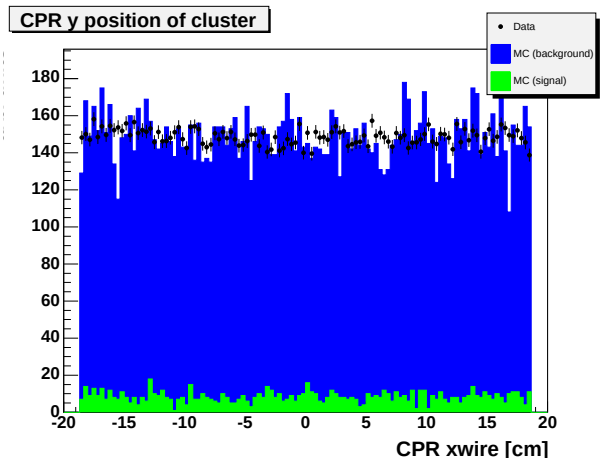

CES energy (wire cluster)
Data
MC (background)
MC (signal)
energy [GeV]

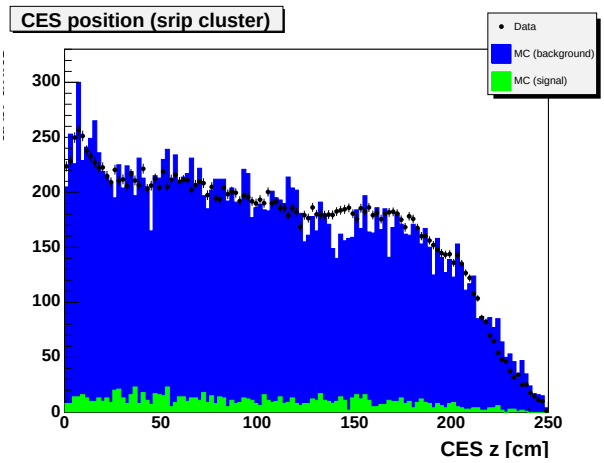

CES energy(strip)/ energy (wire)
Data
MC (background)
MC (signal)
strip/wire energy

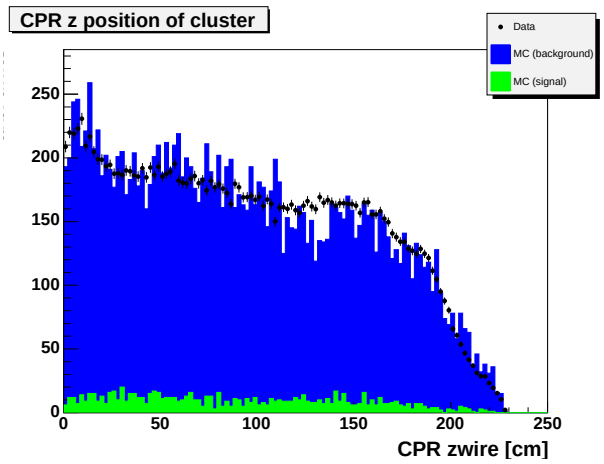

Energy hadron cal./ energy e.m. cal
Data
MC (background)
MC (signal)
Had/EM

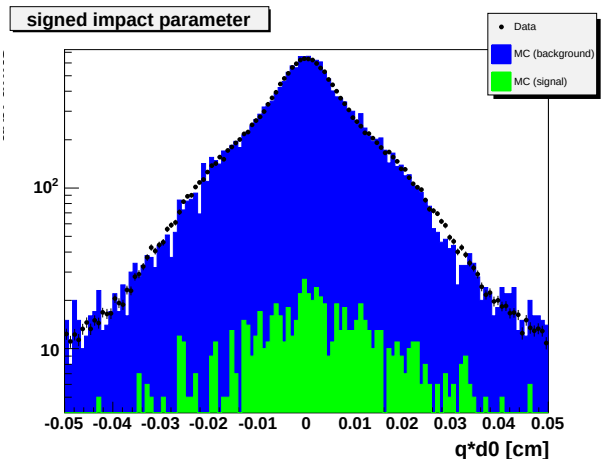

CES energy(strip)/ energy (wire)
Data
MC (background)
MC (signal)
strip/wire energy

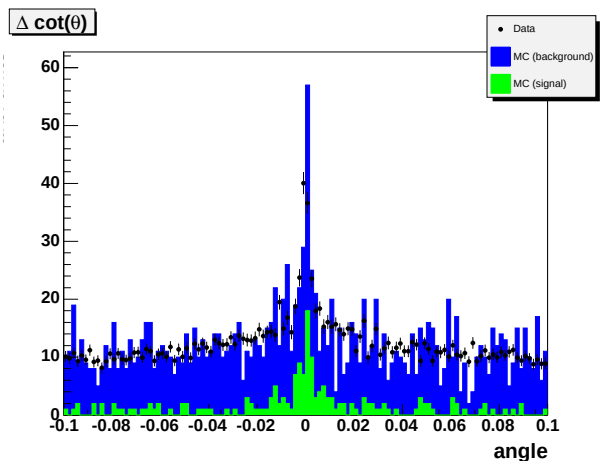

CES position (wire cluster)
Data
MC (background)
MC (signal)
CES x [cm]

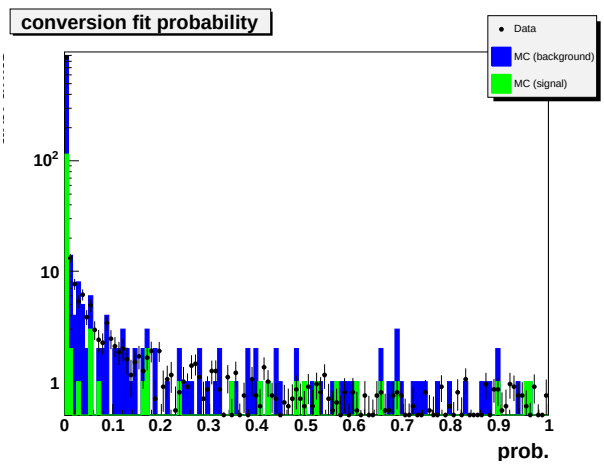

CPR y position of cluster
Data
MC (background)
MC (signal)
CPR xwire [cm]

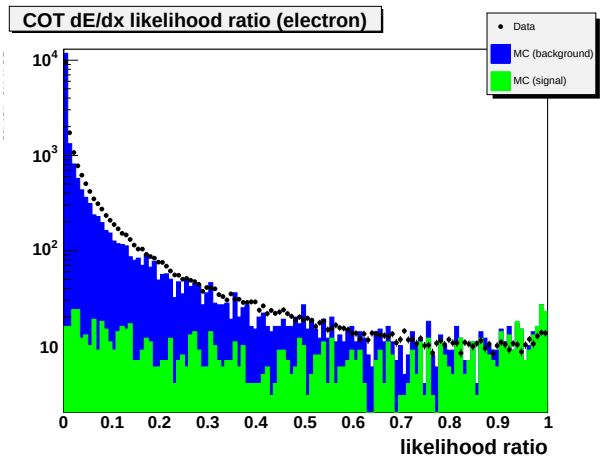

CES position (srip cluster)
Data
MC (background)
MC (signal)
CES z [cm]

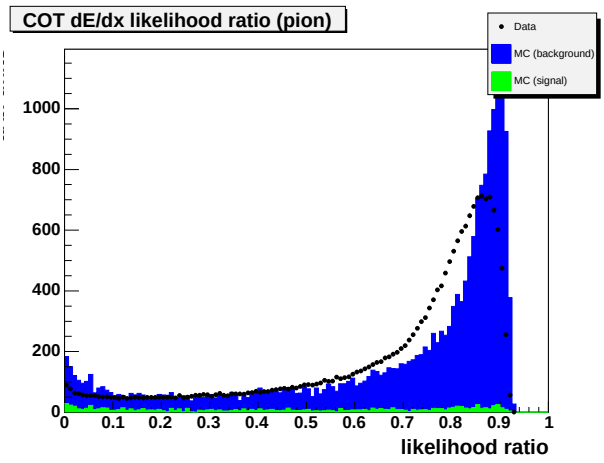

CPR z position of cluster
Data
MC (background)
MC (signal)
CPR zwire [cm]

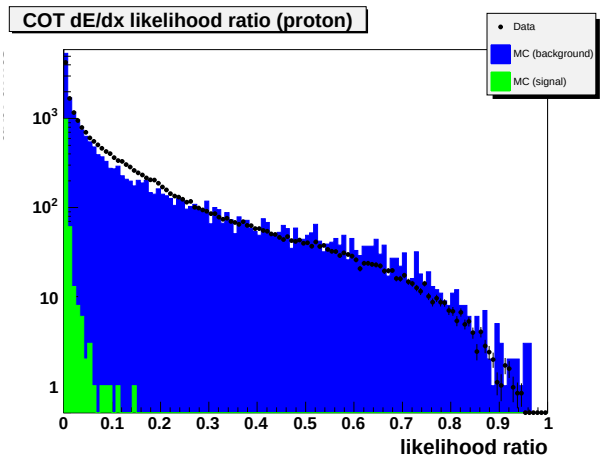

signed impact parameter
Data
MC (background)
MC (signal)
q*d0 [cm]

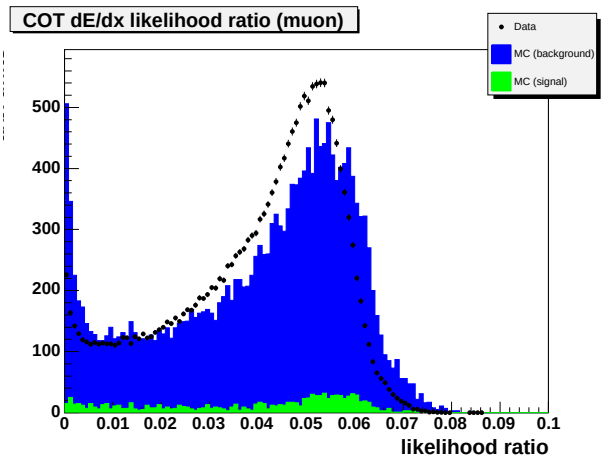

Δ cot(θ)
Data
MC (background)
MC (signal)
angle

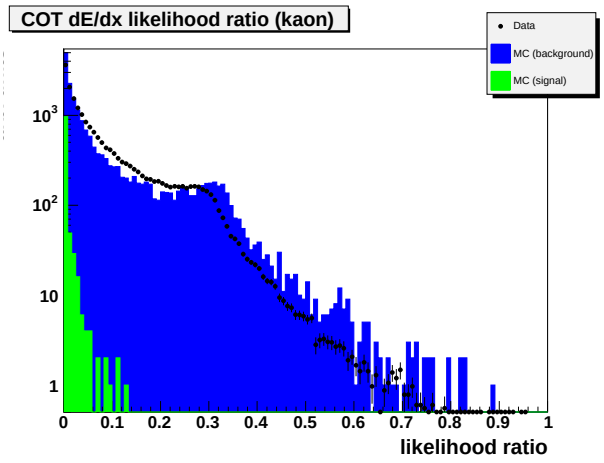

conversion fit probability
Data
MC (background)
MC (signal)
prob.

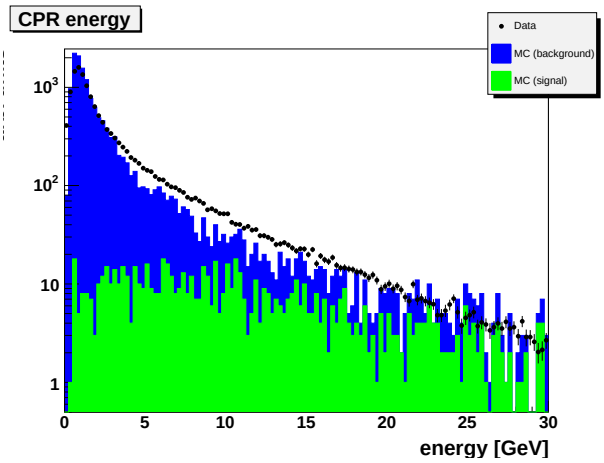

COT dE/dx likelihood ratio (electron)
Data
MC (background)
MC (signal)
likelihood ratio

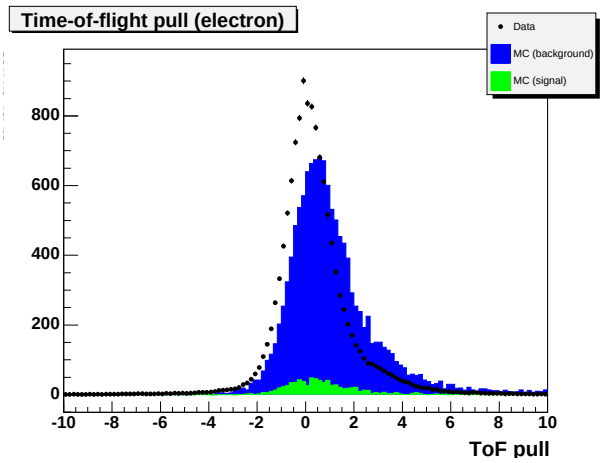

COT dE/dx likelihood ratio (pion)
Data
MC (background)
MC (signal)
likelihood ratio

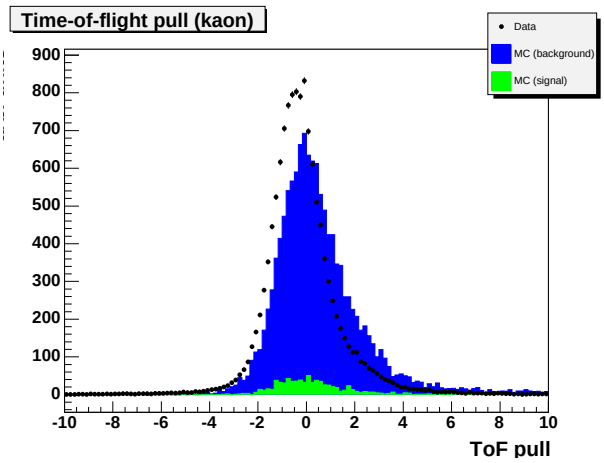

COT dE/dx likelihood ratio (proton)
Data
MC (background)
MC (signal)
likelihood ratio

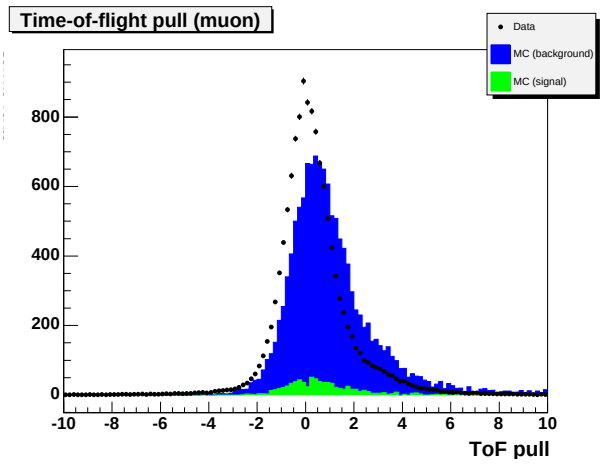

COT dE/dx likelihood ratio (muon)
Data
MC (background)
MC (signal)
likelihood ratio

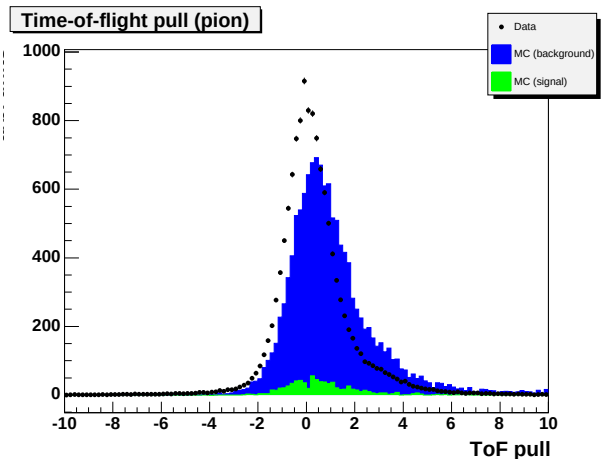

COT dE/dx likelihood ratio (kaon)
Data
MC (background)
MC (signal)
likelihood ratio

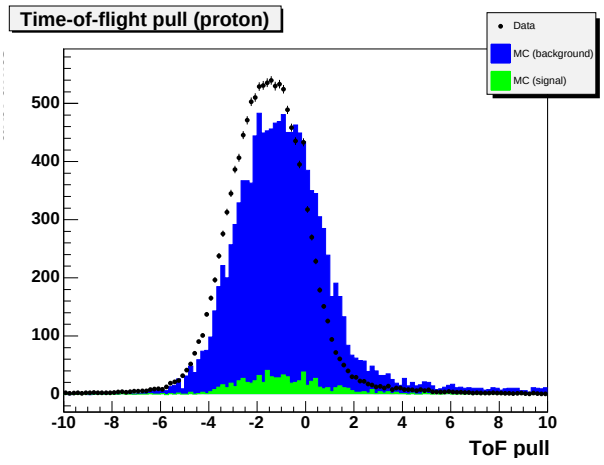

CPR energy
Data
MC (background)
MC (signal)
energy [GeV]


Time-of-flight pull (electron)
Data
MC (background)
MC (signal)
ToF pull


Time-of-flight pull (kaon)
Data
MC (background)
MC (signal)
ToF pull


Time-of-flight pull (muon)
Data
MC (background)
MC (signal)
ToF pull


Time-of-flight pull (pion)
Data
MC (background)
MC (signal)
ToF pull


Time-of-flight pull (proton)
Data
MC (background)
MC (signal)
ToF pull

Appendix C

Technical information about the electron ID toolbox

C.1 Overview

This appendix provides further technical information about the electron identification toolbox and describes the user interface in detail.

The toolbox is implemented as a `C++` singleton class and provides many interfaces to obtain the requested information. This approach has the advantage that the user does not have to know which routines to run in which order, how to set up the neural networks correctly, etc but can directly ask for the physics result.

The toolbox provides two data-types which contain all information gathered. The type `ElectronData` contains all information obtained from reconstructed quantities such as track properties, calorimeters, time-of-flight, dE/dx, etc., whereas the type `MCData` contains information obtained using the truth banks of simulated events.

To efficiently store this information and make it available for easy read-out, standard maps are used where the key the map is the track ID of the electron candidate, the value is then either of the two data-type. This has the advantage that the information needs to be gathered only once and can henceforth be accessed quickly using the maps. *Note:* These maps need to be reset (or "flushed") at the beginning of each event as the track ID is unique only per event. Routines of the toolbox operating on the whole event do this automatically, however, if the user does not want to use these, he has to take care of this himself.

The methods of the toolbox are accessed via:

```
KBSElectronTools::instance()->method(arguments);
```

C.2 General settings

The following three routines initialise the neural networks. The string `Name` sets the name of the NeuroBayes® Expertise file which contains the information about the network topology and training. Different network trainings can be used as long as the input variables are the same and are in the same order. The `debug` flag is used to trace potential problems in the NeuroBayes® part of the toolbox and can only be used with an appropriate license from the company Phi-T[1].

```
void initSENet    (char* Name = "senet.nb"      , int debug=-2);
void initKappaNet (char* Name = "kappann.nb"    , int debug=-2);
void initConvNet  (char* Name = "convnet.nb"    , int debug=-2);
```

The following routines reset all variables currently in use (i.e. not yet stored in the map) to the default value of -999. Normally not used unless looping manually over candidates.

```
void reset ();
void resetMC();
```

These routines clear the information stored in the maps. Has to be called once per event (if not using routines operating on the whole event) as the track ID used as the map key is unique only within an event. It is recommended to call these routines at the beginning of the `event` routine in the `AC++` analysis program.

```
void flushEData();
void flushMCData();
```

To set the overall level of verbosity, the following function can be used. The levels are: 0 (default): no output, 1: indicate begin/end of each called method, write out most important information, 2,3: increase the level of output.

```
void setVerbosity(int verbosity);
```

Further information can be retrieved about the individual hits in the silicon vertex detector such as hit residuals, global or local hit position, etc. As this information is not yet used in a network it is not read out by default. To gather this information, call `setReadoutSIHitResiduals` with `true` as argument. The method `getReadoutSIHitResiduals()` returns the current status of the setting.

```
void setReadoutSIHitResiduals(bool readout);
bool getReadoutSIHitResiduals();
```

[1] For further information email `info@phi-t.de`

C.3 Using the toolbox

Using the `KappaNet`.— These methods are used to identify electrons using the `KappaNet` .

```
bool getKappaElectrons(PartColl_h& ePlusCands,
                       PartColl_h& eMinusCands,
                       const CdfTrackView_ch& tracks,
                       const CdfTrackCut& trackCut,
                       float kappaNetCut);
```

argument	type	purpose
`ePlusCands`	output	collection of identified e^+
`eMinusCands`	output	collection of identified e^-
`tracks`	input	track collection to be analysed (e.g. `defTracks`)
`trackCut`	input	cuts on the tracks, e.g. p_t, η
`kappaNetCut`	input	cut on the `KappaNet` the candidate has to exceed to be accepted as $e^\pm$ (range: -1,...,1, -1 accepts everything)

To obtain the network output for a given track, call:

```
float kappaNetOutput(const CdfTrack_clnk &trackLink);
```

Using the `SENet`.— These methods are used to identify electrons using the `SENet` which combines information from the `KappaNet`, calorimeters, time-of-flight and energy loss in the central drift chamber. Several methods exist operating either on the full event or on specific candidates.

These two methods return particle collections of electrons and positrons found in the event. The method `getSoftElectrons` uses as starting point the `SoftElectronCollection` provided by the `SoftElectronModule` whereas the function `getSoftElectronsCdfEm` starts from the `CdfEmObjColl`.

```
bool getSoftElectrons(AbsEvent* anEvent,
                      PartColl_h& ePlusCands,
                       PartColl_h& eMinusCands,
                       const CdfTrackCut& trackCut,
                       float SENetCut,
                       float ConvNetCut);

bool getSoftElectronsCdfEm(AbsEvent* anEvent,
```

```
                                PartColl_h& ePlusCands,
                                PartColl_h& eMinusCands,
                                const CdfTrackCut& trackCut,
                                float SENetCut,
                                float ConvNetCut);
```

argument	type	purpose
`ePlusCands`	output	collection of identified e^+
`eMinusCands`	output	collection of identified e^-
`trackCut`	input	cuts on the tracks, e.g. p_t, η
`SENetCut`	input	cut on the `SENet` the candidate has to exceed to be accepted as $e^\pm$ (range: -1,...,1, -1 accepts everything)
`ConvNetCut`	input	cut on the `ConvNet` to reject conversion electrons (range: -1,...,1, +1 accepts everything)

The following methods return the output of the softelectron identification network for various methods of specifying the candidate. The argument `useSEColl` determines whether to start from the `SoftElectronCollection` or `CdfEmObjColl`. As several candidate tracks can be associated with a calorimeter cluster, the second method also requires the ID of the candidate track as input. All arguments in these methods are input arguments. The return value is the network output in $-1, \ldots, 1$.

```
float SoftENetOutput(const SoftElectronColl::const_iterator & se_iterator);

float SoftENetOutput(const CdfEmObjectView::const_iterator & emObjIterator,
                     const CdfTrack::Id trackID);

float SoftENetOutput(AbsEvent* anEvent,
                     CdfTrack::Id trackID,
                     bool  useSEColl = true,
                     const CdfTrackCut& trackCut = track_cut::SelectAll());
float SoftENetOutput(AbsEvent* anEvent,
                     const CdfTrack_clnk& trackLink,
                     bool  useSEColl = true,
                     const CdfTrackCut& trackCut = track_cut::SelectAll());
```

The following method returns the ID of the candidate track which has the highest probability in a given event to be an electron.

```
CdfTrack::Id getBestSoftElectron(AbsEvent* anEvent,
```

```
                                const CdfTrackCut& trackCut,
                                float ConvNetCut,
                                float q,
                                bool  useSEColl);
```

argument	type	purpose
`anEvent`	input	the whole event
`trackCut`	input	cuts on the tracks, e.g. p_t, η
`ConvNetCut`	input	cut on the `ConvNet` to reject conversion electrons
		(range: -1,...,1, +1 accepts everything)
`q`	input	select e^- ($q = -1$) or e^+ ($q = +1$)
`useSEColl`	input	start from `SoftElectronCollection (true)`
		or `CdfEmObjColl (false)`

Using the `ConvNet`.— The following methods obtain the probability that a given electron candidate is likely to come from a conversion (netout = +1) or not (netout = -1). Note that the conversion identification networks have been trained on a sample of identified electrons by requiring a minimal cut on the `SENet` of 0, i.e. there has to be a high *a priori* probability that the given candidate is really an electron or positron.

```
float ConvNetOutput(const CdfTrack::Id trackID);
float ConvNetOutput(const CdfTrack_clnk &trackLink);
```

The default precut can be changed by calling:

```
void setConvNetPrecut_SENet(float minNetout);
```

C.4 NTupling

These methods allow convenient storage of all gathered information on ntuples. The `addBranches` methods create the necessary structure on the ntuple. Each variable on the ntuple has the same name as in the provided data structure. The optional argument `branchName` allows to assign the name of the tree created on the ntuple.

```
bool addBranches(HepNtuple* ntuple, KBSElectronTools::ElectronData& eData,
                 const char* branchName = "eData");
bool addBranches(HepNtuple* ntuple, KBSElectronTools::MCData& mcData,
                 const char* branchName = "mcData");
```

The methods below allow access to the internally used maps. The `storeEData` are public to the user, however normally not needed. To retrieve the information for a

given candidate track (using its ID) it should first be tested that there is an entry in the map by calling `existEData(id)` or `existMCData(id)`. If the return value is `true`, the information can be obtained by calling `getEData(id)` or `getMCData`. (See also the example below). The methods `sizeEData()` and `sizeMCData()` return the current size of the map.

```
const  KBSElectronTools::ElectronData* getEData(CdfTrack::Id id);
void storeEData(CdfTrack::Id id);
void storeEData(CdfTrack::Id id,const KBSElectronTools::ElectronData& ed);
bool existEData(CdfTrack::Id id);
int  sizeEData();

const KBSElectronTools::MCData* getMCData(CdfTrack::Id id);
void storeMCData(CdfTrack::Id id);
void storeMCData(CdfTrack::Id id,const KBSElectronTools::MCData& mcd);
bool existMCData(CdfTrack::Id id);
int  sizeMCData();
```

Both provided data-types contain several two dimensional arrays which cannot be captured automatically. For convenience, the necessary code to capture these arrays to the ntuple is coded in the methods:

```
bool captureMCArrays(HepNtuple* ntuple,
                     KBSElectronTools::MCData& mcData,
                     const char* branchName);
bool captureArrays(HepNtuple* ntuple,
                   KBSElectronTools::ElectronData& eData,
                   const char* branchName);
```

C.5 Misc. functions

The following methods provide additional functionality which is not covered by the above sections.
The method returns `true` if a given electron candidate from the `SoftElectronCollection` would pass the cuts of the `SoftElectronModule`, false otherwise.

```
bool passSECut(const SoftElectronColl::const_iterator & seIterator);
```

This method is used to retrieve all information stored in the provided `MCData` data-type, e.g. PID of the candidate as well as its mother particle, the true four-vector, helix parameters, etc.

```
int getMCInfo(AbsEvent* anEvent, const CdfTrack_clnk& track);
```

When the truth banks in simulated events are accessed, an internal flag is checked if this event is simulated. The method `setMCFlag(flag)` sets this flag manually, `getMCFlag` retrieves the current setting.

```
void setMCFlag(bool isMC);
bool getMCFlag();
```

The ID of the track is widely used in the electron identification toolbox. Since several methods exist to obtain the track ID, the helper function

```
CdfTrack::Id getTrackID (const CdfTrack_clnk& trackLink);
```

has been implemented. Its default behaviour is to return the ID obtained via `track → id()`. However, if tracks are refitted the `derived` status is set manually, these IDs may change. The behaviour of the `getTrackID` routine can be switched to use ID of the `COT` ancestor track by calling:

```
void setUseAncestorID(bool useAncestorID);
bool getUseAncestorID()
```

where the latter returns the current setting.
The following two methods determine if the electron track originates from a conversion based on a series of sequential cuts:

```
bool checkConversion (const SoftElectronColl::const_iterator & seIterator);
bool checkConversion (const CdfTrack_clnk& trackLink);
```

The first method uses the conversion collection provided by the `SoftElectronModule`: Each element of the `SoftElectronCollection` which passes some loose conversion criteria is associated with its likely conversion partner(s). The other method takes the track of the electron candidate as input and then loops over all tracks in the event to determine possible conversion partners. This method is used when starting from `CdfEmObjColl`. In detail, the cuts used are:

nr.	cut	value
1	# his in SVX	$= 0$
2	$q \cdot d0$	> 0
3	separation	$<$ cut value (default: 0.02)
4	$\Delta\lvert\cot(\theta)\rvert$	$<$ cut value (default: 0.05)
5	track ID e^-	not equal to track ID e^+
6	fit probaiblity	> 0

where the different criteria are combined as follows: if the conversion collection attached to the `SoftElectronCollection` does not exist for a given candidate (or no

possible conversion partner has been found), criteria (1) and (2) are applied. If potential conversion partners exist, all criteria are evaluated. To set the cut values on the conversion separation and $\Delta|\cot(\theta)|$ the two methods

```
void setSepMax(double sepMax);
void setDcothMax(double dcothMax);
```

are used. The default values `sepMax = 0.02` and `dcothMax = 0.05` have been taken from the `StandardSoftElectronCut.cc` module [96].
Electrons lose a significant part of their energy due to Bremsstrahlung when transversing material in the detector. This leads to a a substantial underestimation of the errors in the track helix fit which are dominated by the behaviour in the central driftchamber, i.e. after most of the Bremsstrahlung has already radiated off. Consequently, vertex fits including electrons will have a bad χ^2 value or not converge at all. The below method allows to manipulate the error-matrix of the helix fit while conserving the correlation between the parameters. The error matrix can be obtained via: `HepSymMatrix &M = track → helixFit().errorMatrix();` The first argument determines which error of the helix fit to operate on: (1) $\cot(\theta)$, (2) curvature, (3) z_0, (4) d_0, (5) ϕ_o. The corresponding error is then enlarged by $\sqrt{factor}$, i.e. $M_{ii} = factor \cdot M_{ii}$. Good results have been obtained by rescaling the errors on curvature and d_0 using a factor 100. Note that the matrix M has to be created such that the changed Matrix returned is used henceforth instead of the original one from the track helix fit.

```
void rescaleErrorMatrix(int var, double factor,
                        HepSymMatrix &errMatrix);
```

Information about the track residuals in the silicon vertex detector (SVX), i.e. information related to the actual hit in the detector and the reconstructed track can be obtained. Since gathering this information is quite time-consuming and only necessary in few analyses, the correspoinding code is disabled by default. The routines can be enabled by calling:

```
void setReadoutSIHitResiduals(bool readout);
bool getReadoutSIHitResiduals();
```

where the latter returns the current setting.

C.6 Example

The example below shows how to use the electron identification toolbox to obtain a collection of electrons and positrons and write the obtained information to the ntuple. It is assumed that the corresponding variables are set up accordingly.

The example uses the methods which use the `SoftElectronCollection` as starting point, thus the `SoftElectronModule` has to run prior to the code described here.

Initialisations.— The following code should be executed during the `beginJob` phase of the `AC++` module:

```
KBSElectronTools::instance()->initSENet(SENetName,-2);
KBSElectronTools::instance()->initKappaNet(KappaNetName,-2);
KBSElectronTools::instance()->initConvNet(ConvNetName,-2);

// set min SENet cut used in ConvNet training
KBSElectronTools::instance()->setConvNetPrecut_SENet(0.0);

KBSElectronTools::instance()->setVerbosity(verbose_);

KBSElectronTools::instance()->addBranches(ntuple, eData, branchName);
KBSElectronTools::instance()->addBranches(ntuple, mcData,branchName);
```

Obtaining the electrons.— The following code should be executed during the `event` phase of the `AC++` module:

```
//
// define basic track cuts, e..
//
 const CdfTrackCut& cMinPt   = track_cut::PtGreaterThan(2.0);
 const CdfTrackCut& cAbsEta  = track_cut::AbsEtaLessThan(1.0);
 const CdfTrackCut& cAllCuts = cMinPt && cAbsEta;

//
// create the collections
//
 PartColl_h       unfittedSoftEPlusColl(new PartColl());
 PartColl_h       unfittedSoftEMinusColl(new PartColl());
 unfittedSoftEPlusColl -> set_description("unfittedSoftEPlusColl");
 unfittedSoftEMinusColl-> set_description("unfittedSoftEMinusColl");


//
// reset electron ID toolbox maps
//
 KBSElectronTools::instance()->flushEData();
 KBSElectronTools::instance()->flushMCData();

//
// run the electron ID toolbox
```

```
//
 KBSElectronTools::instance()->getSoftElectrons(anEvent,
                                                unfittedSoftEPlusColl,
                                                unfittedSoftEMinusColl,
                                                cAllCuts,
                                                SENetCut,
                                                ConvNetCut);

 std::cout << "***   obtained SoftElectrons: " << std::endl;
 std::cout << "*** # electrons (SENet): "
           << unfittedSoftEMinusColl->contents().size()
           << std::endl;
 std::cout << "    # positrons (SENet): "
           << unfittedSoftEPlusColl->contents().size()
           << std::endl;

//
// do the ntupling
//
for (PartColl::const_iterator partIter = unfittedSoftEMinusColl->contents().begin();
      partIter != unfittedSoftEMinusColl->contents().end();
      partIter++) {

    const StablePart_l eMtemp     = (*partIter);
    CdfTrack_clnk      eTrack     = eMtemp->track();
    const CdfTrack::Id etrackID  = eTrack->id();

    if (KBSElectronTools::instance()->existEData(eTrackID)) {
      eData  =  *KBSElectronTools::instance()->getEData(eTrackID);
       KBSElectronTools::instance()->captureArrays(ntuple,
                                                   eData,
                                                   "SENet");

   } // if existEData

   if ( AbsEnv::instance()->monteFlag() ) {
     if (KBSElectronTools::instance()->existMCData(eTrackID)){
      eMCData =  *KBSElectronTools::instance()->getMCData(eTrackID);
      KBSElectronTools::instance()->captureMCArrays(ntuple,
                                                    eMCData,
                                                    "SENet");
     } // if existMCData
   } // if MC
```

```
    ntuple -> capture();
    ntuple -> storeCapturedData();
 } //for

// do the same for positron candidates
```

Appendix D

Packaging NeuroBayes® for CDF

D.1 Overview

CDF software is usually made available via the UNIX Product Support (UPS) [62]. These products are installed using UNIX Product Distribution (UPD). Unlike other package management systems, these tools allow the simultaneous installation of different versions or releases of the same software package, allowing the user to choose between available versions, providing stable releases for the end-users while working on further improvements, etc.
This section describes how to build a UPS/UPD package from the NeuroBayes® libraries provided by `Phi-T`.

Two packages are available to the CDF community:

- The package `neurobayes` contains both the NeuroBayes® Teacher and the NeuroBayes® Expert and can thus be used to both train new neural networks and to use already trained network in the physics analysis. This package requires a license which can be obtained from `Phi-T`. Currently, licenses are available on the machine `fcdflnx2.fnal.gov` and at the University of Karlsruhe.

- The package `neurobayes_expert` contains the NeuroBayes® Expert and can be used in physics analysis running a previously trained network. This package does not require any license and can be freely installed from the CDF software server.

D.2 Useful UPS/UPD commands

Using the provided packages.— To use the NeuroBayes® packages, the software needs to be setup. This can be done via:

```
source ~cdfsoft/cdf2.shrc          (for bash users)
source ~cdfsoft/cdf2.cshrc         (for tcsh users)
setup <package> <version> -f <flavour> -q <qualifier>
```

where `<package>` is either `neurobayes` or `neurobayes_expert`, `<version>` describes the version of the package, `<flavour>` the architecture (so far, only Linux is supported) and `<qualifier>` denotes any additional specification discriminating between different NeuroBayes® releases.
To determine which packages are available, use the following command:

```
ups list -aK+ neurobayes_expert
```

which lists all available packages, e.g.

```
ups list -aK+ neurobayes_expert
"neurobayes_expert" "v1_1" "Linux+2.4" "KCC_4_0" ""
"neurobayes_expert" "v1_1" "Linux+2.4" "GCC_3_4_3" ""
```

i.e. this package is available in version `v1_1` for the compilers `KCC_4_0` and `GCC_3_4_3` under Linux. Setting up one of the packages gives the following output:

```
setup neurobayes v1_1 -f Linux+2.4 -q KCC_4_0
--------------------------------------------------------------
       <<           hh                   tt  >>
     <<             hh       ii          tt    >>
   <<               hh                 tttttt    >>
 <<        ppppp     hhhhhh   ii          tt        >>
   <<      pp  pp    hhh  hh  ii  -----    tt      >>
     <<    pp  pp    hh   hh  ii  -----    tt    >>
       <<  ppppp     hh   hh  ii          tt  >>
           pp
           pp  ////////////////////////////
           pp  \\\\\\\\   Phi-T(R) NeuroBayes(R)

  Algorithms by Michael Feindt
  Implementation by Phi-T Project 2001-2005
  Copyright Phi-T GmbH
  Usage granted for scientific purposes only
  This software is provided as is with no warranty
  For further information contact info@phi-t.de
--------------------------------------------------------------
```

Verifying dependencies.— UPS package can have multiple dependencies, i.e. package which need to be available and set up to use the given package. To verify these dependencies, the command `ups depend <package>` can be used, e.g.

```
source ~cdfsoft/cdf2.shrc
ups depend  ups depend neurobayes_expert v1_1 -f Linux+2.4 -q GCC_3_4_3
neurobayes_expert v1_1 -f Linux+2.4 -z /home/cdfsoft/products/upsdb -q GCC_3_4_3
|__gcc v3_4_3 -f Linux+2.4 -z /home/cdfsoft/products/upsdb
|  |__gdb v5_2_1 -f Linux+2.4 -z /home/cdfsoft/products/upsdb
|__root v4_00_08g -f Linux+2.4 -z /home/cdfsoft/products/upsdb -q GCC_3_4_3
|__cern 2000 -f Linux+2.2 -z /home/cdfsoft/products/upsdb
```

I.e. this package requires the availability of the gcc compiler version v3_4_3 (which in turn depends on the debugger gdb), ROOT version v4_00_08g and the CERN library. If any of these packages is not available, UPS will issue a warning.

Installing new versions.— Phi-T improves the NeuroBayes® neural network continuously and releases new versions regularly. The package neurobayes will be installed on licensed sites when new packages are available. The package neurobayes_expert is distributed with the nightly updates of the CDF software. However, if your off-site installation does use this feature, the package may be installed manually. To check whether new versions are available, execute:

```
source  ~cdfsoft/cdf2.shrc
setup upd
upd list -aK+ neurobayes_expert -h cdfkits.fnal.gov
```

If new versions are available, they can be installed as user cdfsoft via:

```
source  ~cdfsoft/cdf2.shrc
setup upd
upd install neurobayes_expert <version> -f <flavour> \
  -q <qualifier> -h cdfkits.fnal.gov
```

To remove old versions no longer needed, use:

```
source  ~cdfsoft/cdf2.shrc
ups undeclare -Y eurobayes_expert <version> -f <flavour> \
  -q <qualifier>
```

as user cdfsoft.

D.3 Structure of UPS packages

Once UPS packages are installed, they consist of a directory structure below ~cdfsoft/products and an UPS database entry. The directory structure reflects version, qualifier and flavour of the product. Below the directory ~cdfsoft/products, the product directory neurobayes_expert exists containing all installed versions of the package. The directory v1_1GCC_3_4_3/ within this directory contains all available flavours of version v1_1 with qualifier GCC_3_4_3/. Currently, only Linux is supported. Thus the actual installation is then located in ~cdfsoft/products/neurobayes_expert/v1_1GCC_3_4_3/Linux+2.4. The actual installation directory has then the following structure:

```
RELEASE_NOTES  doc      include  licence  make_fragment
bin            example  lib      macros   ups
```

The directory `doc` contains the NeuroBayes® documentation provided by `Phi-T`, `include` contains all needed header files to use the network in own programs, examples how to use the package are provided in the directory `example`, `lib` contains the NeuroBayes® libraries provided by `Phi-T` and `macros` any provided ROOT analysis macros (e.g. a macro to evaluate the training process). The directory `make_fragment` contains a part of a `Makefile` to assist the user in linking the libraries to the analysis program. The directory `ups` contains files internal to UPS, in particular the file `neurobayes.table` which is used by UPS to setup the package correctly. The directory `licence` contains the NeuroBayes® Teacher license needed to train new networks.

D.4 Creating new UPS packages

Cutting a new release.— If new libraries are obtained from `Phi-T` a new version both the `neurobayes` and `neurobayes_expert` package have to be created. This is done in the following way:

1. log in as user `cdfsoft`
2. Create a new directory structure for the new release following the scheme described in appendix D.3.
3. Copy the files obtained from `Phi-T` in the appropriate directories
4. Create the following new files:
   - `neurobayes.mk` in the directory `make_fragment`
   - `neurobayes.table` in the directory `ups`

   and make sure their content agrees with the new release structure. Examples for these files are given below.
5. Declare the thus created release to the local UPS installation to test if everything works. **Cave:** If this is not done properly, the local CDF software installation will be corrupt! Be sure to understand the UPS manual [62]!
   The declaration is done via:

```
ups declare -z <UPS DB directory> \
 <package> <version> \
-f <flavour> -q <qualifier> \
-r <installation directory> \
-m neurobayes.table
```

where:

`<UPS DB directory>`	directory of the UPS database e.g. ∼`cdfsoft/products/upsdb`
`<package>`	name of the package to be installed, e.g. `neurobayes_expert`
`<version>`	version to be installed
`<flavour>`	flavour to be installed e.g. `Linux+2.4`
`<qualifier>`	additional means to discriminate between packages, e.g. the compiler version
`<installation directory>`	directory where the new files have been copied to.

6. Try to setup the package.

7. If everything works, arrange with CDF code management the installation on the central servers at FNAL and the distribution of the package `neurobayes_expert` via UPD. Prove code management with a `tar` archive of the newly created versions.

Sample `Makefile` *fragment.—*

```
# neurobayes.mk
#
# Original version:
# Michael Feindt, Ulrich Kerzel
# June 2005: for NeuroBayes Expert only

extpkg := neurobayes

arch_spec_warning :=
NEUROBAYES_EXPERT_DIR_DEFAULT = /home/cdfsoft/products/neurobayes/v1_1GCC_3_4_3/Linux+2.4
ifndef NEUROBAYES_EXPERT_DIR
    arch_spec_warning:=\
    "Using default value NEUROBAYES_EXPERT_DIR = $(NEUROBAYES_EXPERT_DIR_DEFAULT)"
   NEUROBAYES_EXPERT_DIR = $(NEUROBAYES_EXPERT_DIR_DEFAULT)
endif

NEUROBAYES_LIB = -L$(NEUROBAYES_EXPERT_DIR)/lib -lNeuroBayesExpertCPP  \
         -lNeuroBayesInterfaceDummy

override LOADLIBES += $(NEUROBAYES_LIB)
override CPPFLAGS += -I$(NEUROBAYES_EXPERT_DIR)/include
override CXXFLAGS += -I$(NEUROBAYES_EXPERT_DIR)/include
```

Sample UPS table file (neurobayes.table).—

```
FILE=TABLE
PRODUCT=neurobayes_expert

GROUP:

  FLAVOR=ANY
  QUALIFIERS="GCC_3_4_3"

COMMON:

  # Internal action, used by all other actions.
  ACTION=SetEnvironment
    setupEnv()
    proddir()
  # Setup action.
  ACTION=Setup
    exeActionRequired(SetEnvironment)
    setupRequired(gcc v3_4_3 -f Linux+2.4)
    setupRequired(root v4_00_08g  -f Linux+2.4 -q GCC_3_4_3)
    setupRequired(cern 2000)
    envSet(PHIT_LICENCE_PATH,${UPS_PROD_DIR}/licence)
    pathPrepend(LD_LIBRARY_PATH,${UPS_PROD_DIR}/lib)
    execute("echo "------------------------------------------------------------" ",NO_UPS_ENV)
    execute("echo "       <<           hh                    tt  >>            " ",NO_UPS_ENV)
    execute("echo "     <<             hh       ii           tt    >>          " ",NO_UPS_ENV)
    execute("echo "   <<               hh                  tttttt    >>        " ",NO_UPS_ENV)
    execute("echo " <<        ppppp    hhhhhh   ii           tt        >>      " ",NO_UPS_ENV)
    execute("echo "   <<      pp  pp   hhh  hh  ii  -----    tt      >>        " ",NO_UPS_ENV)
    execute("echo "     <<    pp  pp   hh   hh  ii  -----    tt    >>          " ",NO_UPS_ENV)
    execute("echo "       <<  ppppp    hh   hh  ii           tt  >>            " ",NO_UPS_ENV)
    execute("echo "           pp                                              " ",NO_UPS_ENV)
    execute("echo "           pp  ////////////////////////////                 " ",NO_UPS_ENV)
    execute("echo "           pp  \\\\\\\\\\\\\\\   Phi-T(R) NeuroBayes(R)     " ",NO_UPS_ENV)
    execute("echo "                                                           " ",NO_UPS_ENV)
    execute("echo "  Algorithms by Michael Feindt                             " ",NO_UPS_ENV)
    execute("echo "  Implementation by Phi-T Project 2001-2003                " ",NO_UPS_ENV)
    execute("echo "  Copyright Phi-T GmbH                                     " ",NO_UPS_ENV)
    execute("echo "  Usage granted for scientific purposes only               " ",NO_UPS_ENV)
    execute("echo "  This software is provided as is with no warranty         " ",NO_UPS_ENV)
    execute("echo "  For further information contact info@phi-t.de            " ",NO_UPS_ENV)
    execute("echo "------------------------------------------------------------" ",NO_UPS_ENV)
```

Appendix E

Realistic simulation of $\Psi(2S) \to J/\psi\pi^+\pi^-$

E.1 Overview

In order to study the decay $\psi(2S) \to J/\psi\pi^+\pi^-$ in more detail it has been simulated in a realistic simulation employing both the full detector and the trigger simulation. Two separate studies were performed where the J/ψ decayed either to $\mu^+\mu^-$ or to e^+e^-, i.e. these studies differ by the decay channel of the J/ψ and the trigger used to detect these events. This section is organised as follows: First the aspects common to both studies are described, then the later sections focus on the specific aspects of each study.

Each simulated events consists of a single signal particle which is then subsequently decayed. No background events were simulated as the description of background in simulation programs does still not sufficiently describe the data and should thus be taken from e.g. side-band regions (i.e. regions $\geq 2\sigma$ from the signal region). However, secondary reactions where properly simulated, e.g. the conversion of a photon to an e^+e^- pair, etc. Figure E.1 shows a few examples.

Common aspects.— The simulation has been done using version `5.3.3` of the `CDF` offline software. To include the latest development in the simulation of the time-of-flight detector, the simulation executable `cdfSim` has been patched according to [97].

The events were generated by `BGen` [74] which generates a single $\psi(2S)$ particle per event. The particles were generated in the range $-1.1 \leq \eta \leq 1.1$ with a minimal $p_t > 4.0$ GeV/c. The input p_t spectrum has been taken from [98] and was assumed to

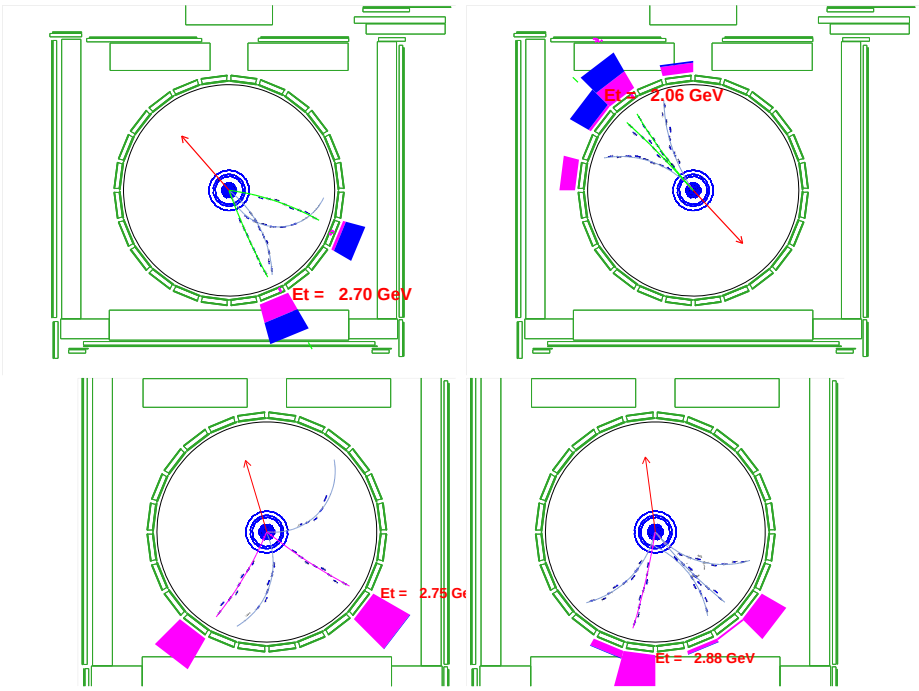


Figure E.1: A few example events from the realistic simulation. The upper two plots show events with the decay $J/\psi \to \mu^+\mu^-$, the lower two plots illustrate events with $J/\psi \to e^+e^-$. The muons are plotted in green, pions in blue and electrons in magenta. Note the extra activity in the lower right plot caused by the conversion of a Bremsstrahlungs-photon into an additional e^+e^- pair. The display of the central drift chamber (COT) has been enlarged.

be flat in η. It can be parametrised as [99]:

$$\frac{d\sigma}{d\,p_t} = \frac{A}{(p_t^2 + m^2)^n}\ \text{nb/GeV/c} \tag{1}$$

where $A = 4345$, $M = 3.69$, $n = 2.60$ in the relevant region of $3.8 \leq p_t(\psi(2S)) < 18$ GeV/c.

The subsequent decay was performed by `EvtGen` [64]. The branching ratio of the $\psi(2S)$ has been modified to decay only to $J/\psi\pi^+\pi^-$ using the specialised `VVPIPI` decay mode where the amplitude of the mass of the $\pi\pi$ sub-system given by $A \propto (m^2_{\pi\pi} - 4m^2_\pi)$. The resulting J/ψ was then forced to decay only to either $\mu^+\mu^-$ or e^+e^- pairs, using the `PHOTOS VLL` mode. The `VLL` mode correctly models the decay of a vector particle into two leptons, whereas the `PHOTOS` package [75] takes care of radiative effects.

To prevent that many candidates the following cuts were imposed at generator (`HepG`) level:

particle	$p_t >$ (GeV/c)	$\lvert\eta\rvert <$
l^+, l^-	1.45	1.5
π^+, π^-	0.3	1.5
J/ψ	2.0	1.5

Since all analyses were performed in the central region of the CDF detector (defined by $|\eta| < 1$) all generated particles were required to be in the same region.

Only muons with a minimal transverse momentum of $p_t > 1.5$ GeV/c can reach the muon chambers and pass the trigger requirements. The electrons need an even higher minimal transverse momentum of $p_t > 2$ GeV/c to pass the trigger. A lose cut of $p_t > 0.3$ GeV/c has been imposed on the pions in the decay of the $\psi(2S)$ as tracks with lower transverse momentum are unlikely to be reconstructed by the tracking software. The above cuts reject events which will not pass the requirements from the trigger or the basic selection cuts in the analyses but are sufficiently lose to avoid a potential bias.

The output of the generation process and the subsequent detector simulation was then filtered through the trigger simulation. The trigger simulation (`TrigSim++`) removes events which would not be recorded by the CDF detector.

In a last step, the program `ProductionExe` is run which performs the tracking, calorimeter clustering, etc. The output of this program then "looks" the same as normal data-files and can be treated in the same way.

E.2 Special requirements for the decay $J/\psi \to \mu^+\mu^-$

In this simulation the J/ψ originating from the $\psi(2S)$ was forced to decay to $\mu^+\mu^-$. The thus generated decays were passed through the trigger simulation for the di-muon dataset. The specific trigger requirements for this dataset are:

- `L1_TWO_CMU1.5_PT1.5`
- `L1_CMU1.5_PT1.5_&_CMX1.5_PT2`
- `L1_CMUP6_PT4`

Figure E.2 illustrates the agreement between data and simulation for important quantities in the $\psi(2s)$ signal region. The plots are obtained in the following way: Each quantity (e.g. the p_t of the reconstructed π^+) is divided into n bins. The borders of these bins were then used as additional cuts in the reconstruction of the $\psi(2S)$. By fitting the thus obtained $\mu^+\mu^-\pi^+\pi^-$ mass spectrum the $\psi(2S)$ yield and corresponding error are extracted and filled into the corresponding bin of the histogram. Then the

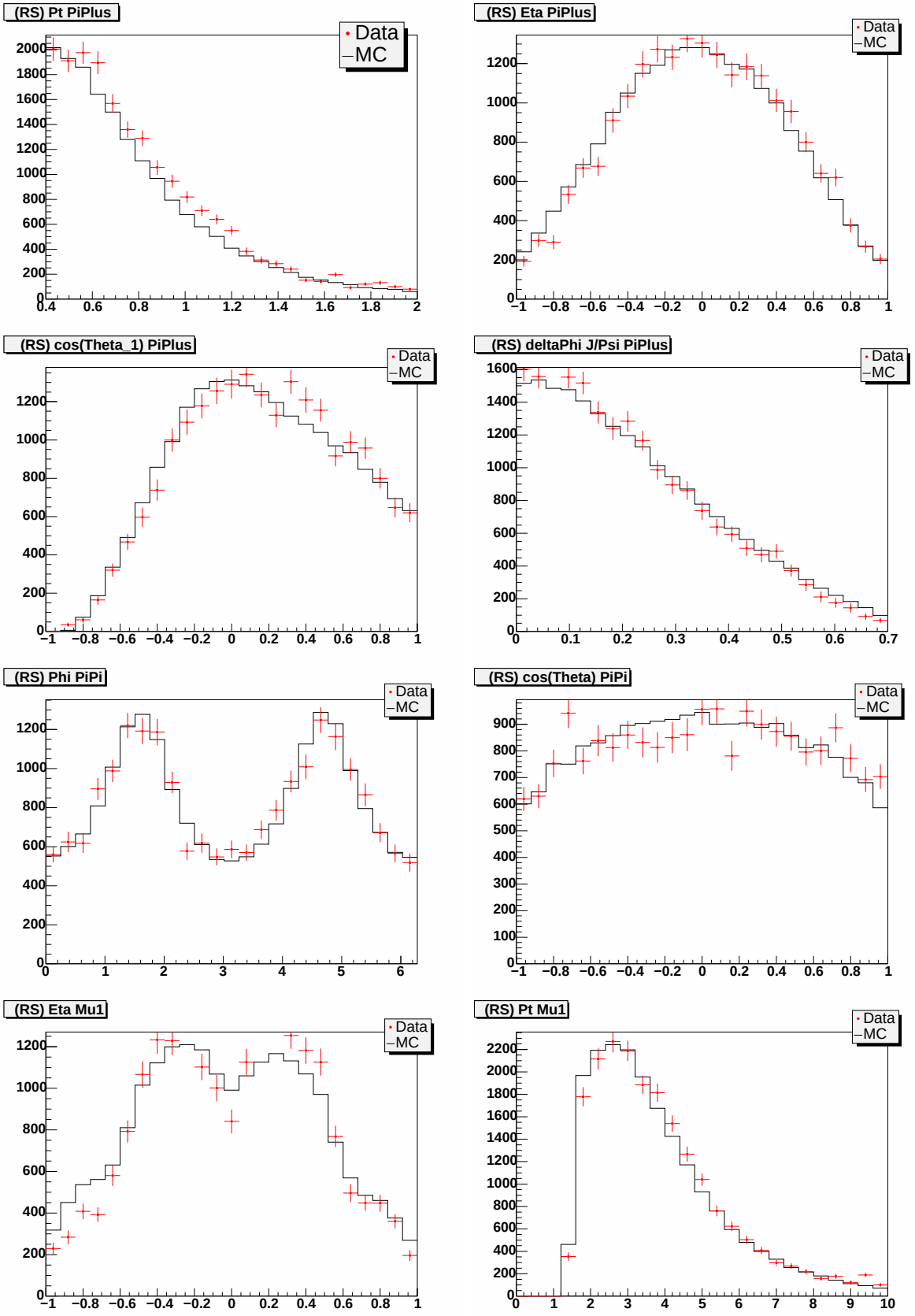


Figure E.2: Agreement between data taken from the $\psi(2S)$ signal region and realistic simulation. The units for $p_t(\pi)$ and $p_t(\mu)$ are GeV/c. "(RS)" indicates that the right-sign combination (i.e. $X(3872) \to J\psi\pi^+\pi^-$) has been analysed

.

distribution of the variable in the simulation is overlaid normalised to the same area. It can be seen that the simulation accurately describes the data.

The upper two plots of figure E.1 show two typical events obtained by this simulation. The muons are shown in green, pions are plotted in blue.

E.3 Special requirements for the decay $J/\psi \to e^+e^-$

In order to study the behaviour of electrons in the detector in detail a dedicated simulation has been run. This has been done in the following way: Starting from the simulation described above the decay of the J/ψ was changed such that it decays exclusively to e^+e^-, again using the `PHOTOS` package to properly handle radiative effects. The trigger simulation has been changed to only select events which would pass the requirements of the dedicated $J/\psi \to e^+e^-$ trigger:

- `L1_TWO_CEM2_PT2_OPPQ`
- `L1_TWO_CEM2_PT2_OPPQ L2_JPSI_TWO_CEM2`

These requirements select events where two tracks of opposite charge with $p_t > 2$ GeV/c deposit at least $E_t > 2$ GeV in the central electro-magnetic calorimeter. Additionally, the ratio of energy deposited in the hadronic calorimeter and the electromagnetic calorimeter is required to be below 0.125 to suppress hadronic background (mainly from pions).

The lower two plots of figure E.1 illustrate the different topology of these events. The electrons are shown in magenta whereas the pions are again plotted in blue. The lower left plot shows an event with the same topology as found in the case where the J/ψ decays to two muons (although in this event both pions have much lower transverse momentum). The lower right plot however shows an event with additional activity: One of the electrons originating from the decay of the J/ψ interacted with the detector material and radiated off a Bremsstrahlungs-photon which in turn converted into into an additional e^+e^- pair. Due to the light mass of the electron this process is much more pronounced for electrons than for muons and presents one of the major challenges during the reconstruction and identification of electrons in the detector.

Appendix F

Explicit calculation of a helicity matrix element

F.1 Example : $X(0^+) \to J/\psi \ (\pi^+\pi^-)_s$

The helicity formalism discussed in section 6.3 is illustrated by applying it in detail to the decay $X(3872) \to J/\psi\,(\pi^+\pi^-)_s$. It is assumed that the $X(3872)$ is a scalar (i.e. $J = 0$) with positive parity $P = 1$ and negative charge parity $C = -1$. The $\pi^+\pi^-$ system is in a relative s-wave state, thus the quantum numbers of the involved particles are:

$$\begin{array}{ccccc} X & \to & J/\psi & (\pi^+\pi^-)_{s-wave} \\ 0^+ & & 1^- & 0^+ \end{array}$$

The possible helicities of the involved particles can be deduced from the above assignments: The $X(3872)$ is a scalar (i.e. $J = 0$), hence the total angular momentum is zero, consequently its helicity is $\lambda_X = 0$ where the z-axis has been chosen again as the direction of the $X(3872)$, i.e. $J_z = \lambda_X$.
The same argument holds for the $\pi^+\pi^-$ system in a relative s-wave state leading to $\lambda_{\pi\pi} = 0$ where now the z' axis is defined along the flight direction of the $(\pi^+\pi^-)$ and J/ψ system in the centre-of-mass system, i.e. $\hat{z}' \parallel \vec{p}_{\pi\pi} = -\vec{p}_{J/\psi}$.

The J/ψ however is a vector particle with $J = 1$ and has hence the possible helicities: $\lambda_{J/\psi} = +1,\ 0,\ -1$. The helicities of the daughter particles are related via:

$$\lambda = \lambda_{J/\psi} - \lambda_{\pi\pi}$$

as discussed in section 6.3. Since $\lambda_{\pi\pi}$ is zero, λ is the same as $\lambda_{J/\psi}$. As the $X(3872)$ is assumed to be spinless in this example, its spin J and its helicity λ_X are zero as well. In this special case the latter is true for *any* quantisation axis, in particular for

the axis $\hat{z}'$ defined by the decay particles:

$$J_z \equiv \lambda_X = 0 = J'_z \equiv \lambda = \lambda_{J/\psi} - \lambda_{\pi\pi}$$

As discussed above, $\lambda_{\pi\pi} = 0$, to fulfil the above relation, the J/ψ can only be produced with zero helicity along $\hat{z}'$, i.e. $\lambda_{J/\psi} = 0$.

To compute the full matrix element, the decay of the $(\pi^+\pi^-)_s$ and the J/ψ need to be considered. The treatment of the di-pion system is straightforward for the s-wave case:

$$\begin{array}{ccccc} \pi^+\pi^- & \rightarrow & \pi^+ & \pi^- \\ 0^+ & & 0^- & 0^- \end{array}$$

i.e. all particles are spin-less and have zero helicity.

The consideration for the J/ψ is more complex since the J/ψ is a vector particle with spin $J = 1$ and the muons are fermions with spin $J = \frac{1}{2}$:

$$\begin{array}{ccccc} J/\psi & \rightarrow & \mu^+ & \mu^- \\ 1^- & & \frac{1}{2} & \frac{1}{2} \end{array}$$

Hence all possible helicity combinations are:

$\lambda_{\mu\mu}$	λ_{μ^+}	λ_{μ^-}
0	$+\frac{1}{2}$	$+\frac{1}{2}$
0	$-\frac{1}{2}$	$-\frac{1}{2}$
1	$+\frac{1}{2}$	$-\frac{1}{2}$
-1	$-\frac{1}{2}$	$+\frac{1}{2}$

where the helicities of the muons are again combined via $\lambda_{\mu\mu} = \lambda_{\mu^+} - \lambda_{\mu^-}$. Due to the transversality of the photon in the decay of the J/ψ ($c\bar{c} \rightarrow \gamma^* \rightarrow \mu^+\mu^-$) the combination with $\lambda_{\mu\mu} = 0$ does not exist and only the matrix elements $\mathcal{M}_{-\frac{1}{2},+\frac{1}{2}}$ and $\mathcal{M}_{+\frac{1}{2},-\frac{1}{2}}$ leading to $\lambda_{\mu\mu} = 1$ remain. They are related via:

$$\mathcal{M}_{-\lambda_1,-\lambda_2} = \eta\mathcal{M}_{\lambda_1,\lambda_2}$$

where η is the *naturality* defined by

$$\eta = P(P_1P_2) * (-1)^{J-J_1-J_2}$$

from which it follows that $\mathcal{M}_{-\frac{1}{2},+\frac{1}{2}} = \mathcal{M}_{+\frac{1}{2},-\frac{1}{2}}$. Note that these considerations are valid along the decay axis of the J/ψ determined by the muons boosted into the rest-frame of the J/ψ (which in turn is boosted into the rest-frame of the $X(3872)$). In

the construction of the complete helicity matrix element the helicities $\lambda_{J/\psi} = 0$ and $\lambda_{\mu\mu} = 1$ are related via a Wigner D function.

As the J/ψ is a vector particle, whereas both the $X(3872)$ and the $\pi^+\pi^-$ are assumed to be scalars in this example, the J/ψ and $\pi^+\pi^-$ system will be in a $L = 1$ state with respect to each other to conserve the total angular momentum which contributes a factor ${k^*}^{L=1}$.

According to the prescription discussed in detail in section 6.3 the matrix element is then constructed in the following way:

$$\begin{aligned}
\mathcal{M}^{0^+} &\propto \sum_{\lambda_{J/\psi}} \sum_{\lambda_{(\pi\pi)_s}} \mathcal{M}^{X}_{\lambda_X,\lambda_{J/\psi},\lambda_{(\pi\pi)_s}} \cdot \mathcal{M}^{J/\psi}_{\lambda_{J/\psi},\lambda_{\mu^+},\lambda_{\mu^-}} \cdot \mathcal{M}^{(\pi\pi)_s}_{\lambda_{(\pi\pi)_s},\lambda_{\pi^+},\lambda_{\pi^-}} \cdot c_{11}(\lambda_{J/\psi},\lambda_{(\pi\pi)_s}) \\
&\propto k^*_{J/\psi\ in\ X\ cms}\ f_1(k^*)\ D^0_{0,0} \cdot Prop_{J/\psi}\ D^1_{0,1} \cdot Prop_{(\pi\pi)_s}\ D^0_{0,0} \\
&\propto k^*_{J/\psi\ in\ X\ cms}\ f_1(k^*) \cdot cos(\theta_{J/\psi}) \cdot Prop_{(\pi\pi)_s}
\end{aligned}$$

where k^* is the magnitude of the momentum of the J/ψ calculated in the $X(3872)$ centre-of-mass system, $D^J_{J_z,\lambda}$ are the Wigner D functions and $Prop(...)$ are the propagators of the J/ψ and the $\pi^+\pi^-$ system, respectively. The sum over the final states has been omitted which (after squaring $\mathcal{M}^{0^+}$) contributes a factor two in this case (from the treatment of the muons). As the natural width of the J/ψ is much lower than the detector resolution, the J/ψ mass is treated as constant in the formalism and fixed to the nominal PDG value. The propagator for the $(\pi\pi)_s$ however needs to be modelled as discussed in section 6.3.

Bibliography

[1] D. Acosta et al. *Observation of the Narrow State $X(3872) \to J/\psi\,\pi^+\pi^-$ in $\bar{p}p$ Collisions at $\sqrt{s} = 1.96$ TeV. Phys. Rev. Lett.*, 93:072001, 2004.

[2] S. K. Choi et al. *Observation of a New Narrow Charmonium State in Exclusive $B^\pm \to K^\pm\pi^+\pi^- J/\psi$ decays. Phys. Rev. Lett.*, 91:262001, 2003.

[3] B. Aubert et al. *Study of the $B \to J/\psi K^-\pi^+\pi^-$ decay and measurement of the $B \to X(3872)K^-$ branching fraction. Phys. Rev.*, D71:071103, 2005.

[4] V. M. Abazov et al. *Observation and properties of the $X(3872)$ decaying to $J/\psi\pi^+\pi^-$ in p anti-p collisions at $\sqrt{s} = 1.96$ TeV. Phys. Rev. Lett.*, 93:162002, 2004.

[5] G. Bauer et al. *Dipion mass spectrum in $X(3872)$ and $\psi(2S)$ decays*, 2005. CDF/DOC/BOTTOM/CDFR/7502.

[6] Ted Barnes and Stephen Godfrey. *Charmonium options for the X(3872). Phys. Rev.*, D69:054008, 2004.

[7] A. De Rujula, Howard Georgi, and S. L. Glashow. *Molecular charmonium: a new spectroscopy? Phys. Rev. Lett.*, 38:317, 1977.

[8] Nils A. Tornqvist. *Isospin breaking of the narrow c harmonium state of Belle at 3872 MeV as a deuson. Phys. Lett.*, B590:209–215, 2004.

[9] Eric S. Swanson. *Short range structure in the X(3872). Phys. Lett.*, B588:189–195, 2004.

[10] J. Heuser. *Spin-Paritätsanalyse des X(3872)- Zustandes bei CDF Run II.* Master's thesis, Institut für Experimentelle Kernphysik, University of Karlsruhe, 2005. IEKP-KA/2005-4.

[11] V. A. Novikov and Mikhail A. Shifman. *Comment on the $\psi' \to J/\psi\pi\pi$ DECAY. Zeit. Phys.*, C8:43, 1981.

[12] Rui Zhang, Yi-Bing Ding, Xue-Qian Li, and Philip R. Page. *Molecular states and 1^{-+} exotic mesons. Phys. Rev.*, D65:096005, 2002.

[13] M. Buscher, F. P. Sassen, N. N. Achasov, and L. Kondratyuk. *Investigation of light scalar resonances at COSY.* 2003.

[14] Charmonium spectrum (e835 collaboration). http://www.e835.to.infn.it/images/charm_spectrum_e835.ps.

[15] C. Z. Yuan, X. H. Mo, and P. Wang. *The upper limit of the e+ e- partial width of X(3872). Phys. Lett.*, B579:74–78, 2004.

[16] P. Zweber. *Search for X(3872) in gamma gamma fusion and ISR at CLEO.* 2005.

[17] Sandip Pakvasa and Mahiko Suzuki. *On the hidden charm state at 3872-MeV. Phys. Lett.*, B579:67–73, 2004.

[18] B. Aubert et al. *Properties of the $X(3872)$ at Belle.* 2004. hep-ex/0408116.

[19] F. Mandl. *Spectroscopy and new particles.* Beauty 2005.

[20] G. Bauer et al. *The 'Lifetime' Distribution of X(3872) Mesons* , 2004. CDF/PUB/BOTTOM/PUBLIC/7159.

[21] M. Feindt, J. Heuser, and U. Kerzel. *On the $\pi^+\pi^-$ mass spectrum in $X(3872) \to J/\psi\rho^0$ decays*, 2005. CDF/DOC/BOTTOM/CDFR/7662.

[22] Tung-Mow Yan. *Hadronic transitions between heavy quark states in quantum chmomodynamics. Phys. Rev.*, D22:1652, 1980.

[23] *$e^+e^- \to \pi^+\pi^-\pi^+\pi^-$, $K^+K^-\pi^+\pi^-$ and $K^+K^-K^+K^-$ cross sections at center-of-mass energies 0.5–4.5 GeV measured with initial-state radiation. Physical Review D (Particles and Fields)*, 71(5):052001, 2005.

[24] S. K. Choi. *Properties of the X(3872).* 2004.

[25] K. Abe et al. *Evidence for $X(3872) \to \gamma J/\psi$ and the sub-threshold decay $X(3872) \to \omega J/\psi$.* 2005.

[26] B. Aubert et al. *Search for a charged partner of the $X(3872)$ in the meson decay $B \to X^-K$, $X^- \to J/\psi\pi^-\pi^0$. Phys. Rev.*, D71:031501, 2005.

[27] E. Eichten, K. Gottfried, T. Kinoshita, K. D. Lane, and Tung-Mow Yan. *Charmonium: The Model. Phys. Rev.*, D17:3090, 1978.

[28] A.M. Zaitsev. *Hybrid Mesons.* Proceedings of hte XXII workshop on high energy physics and field theory, June 1998.

[29] F. Abe et al. *Evidence for top quark production in anti-p p collisions at* $\sqrt{s} = 1.8$ *TeV. Phys. Rev. Lett.*, 73:225–231, 1994.

[30] T. Affolder et al. *Measurement of the top quark mass with the Collider Detector at Fermilab. Phys. Rev.*, D63:032003, 2001.

[31] R. Blair et al. *The CDF-II detector: Technical design report.* FERMILAB-PUB-96-390-E.

[32] W.R. Leo. *Techniques for Nuclear and Particle Physics Experiments.* Springer, 1994.

[33] C. Ciobanu et al. *Online Track Processor for the CDF Upgrade*, 1999. IEEE Trans. Nucl. Sci., vol. 46, pp. 933-939.

[34] A. Bardi et al. *The CDF-II Online Silicon Vertex Tracker*, 2001. eConf C011127:THBT003, hep-ph/0112141.

[35] M. Herndon. *The Di-Muon Spectrum.* `http://www-cdf.fnal.gov/internal/people/links/MatthewHerndon/dimuon/dimuon.html` .

[36] Ian Foster and Carl Kesselman. *The Grid 2 : Blueprint for a New Computing Infrastructure.* Morgan Kaufmann, 2^{nd} edition, 2003.

[37] *The Globus Alliance.* `http://www.globus.org`.

[38] *The* bbftp *file transfer protocol.* `http://doc.in2p3.fr/bbftp`.

[39] *Anforderungen an ein "Grid Computing Centre Karlsruhe".* `http://grid.fzk.de/LHCComputing-1july01.pdf`.

[40] *Antwort auf die Anforderungen an ein "Grid Computing Centre Karlsruhe".* `http://grid.fzk.de/RDCCG-answer-v8.pdf`.

[41] *GridKa hardware overview.* `www.gridka.de` → GridKa Info → Hardware.

[42] *PBS Professional.* `http://www.altair.com/software/pbspro.htm`.

[43] J. van Wezel, H. Marten, B. Verstege, and A. Jaeger. *First experiences with large SAN storage and Linux. Nucl. Instrum. Meth.*, A534:29–32, 2004.

[44] IBM. *General Parallel File System.* `http://www.ibm.com/servers/eserver/clusters/software/gpfs.html`.

[45] IBM. *Tivoli Storage Manager*. `http://www.ibm.com/software/tivoli/products/storage-mgr`.

[46] *dCache*. `http://www.dcache.org`.

[47] *Global Grid User Support*. `http://www.ggus.org`.

[48] *The SAM architecture*. `http://cdfdb.fnal.gov/sam/doc/architecture/sam_architecture.html`.

[49] Object Management Group. *The common object request broker: Architecture and specification*. Technical Report 97.09.01, Object Management Group, August 1997. `www.corba.org`.

[50] Object Managment Group. *The Common Object Request Broker: Architecture and Specification (revision 2.2)*. Technical report, Object Managment Group, February 1998. `ftp://ftp.omg.org/pub/docs/formal/98-02-01.ps`.

[51] R. Brun and F. Rademakers. *ROOT: An object oriented data analysis framework. Nucl. Instrum. Meth.*, A389:81–86, 1997.

[52] F. Ratnikov. *Input and Output Modules user guide*, 2001. CDF/DOC//PUBLIC/5336.

[53] *SAM-Grid Logistics*. `http://www-d0.fnal.gov/computing/grid/JIM_V1_Logistics.jpg`.

[54] M. Litzkow, M. Livny, and M. Mutka. *Condor - A Hunter of Idle Workstations*. In *Proceedings of the 8th International Conference of Distributed Computing Systems*, June 1988.

[55] T. Moulik, M. Tanaka, and B. Wicklund. *B Flavor tagging using opposite side electrons*, 2003. CDF/ANAL/BOTTOM/CDFR/6793.

[56] V. Tiwari, G. Giurgiu, M. Paulini, J. Russ, and B. Wicklund. *Likelihood Based Electron Tagging*, 2004. CDF/ANAL/BOTTOM/CDFR/7121.

[57] T. Allmendinger, G. J. Barker, M. Feindt, C. Haag, and M. Moch. *BSAURUS: A package for inclusive B reconstruction in DELPHI*. 2001.

[58] M. Milnik. *New methods for electron identification with the CDFII-Detector*. Master's thesis, Institut für Experimentelle Kernphysik, University of Karlsruhe, 2003.

[59] M. Feindt. *A Neural Bayesian Estimator for Conditional Probability Densities*, 2004. http://arxiv.org/abs/physics/0402093.

[60] Phi-T Physics Information Technologies. *The NeuroBayes User's Guide*, 2002-2005. `www.phi-t.de`.

[61] Phi-T Physics Information Technologies. *How to use NeuroBayes in Root*, 2004. `www.phi-t.de`.

[62] Fermilab Computing Division. *UNIX Product Support*. `http://www.fnal.gov/docs/products/ups`.

[63] Torbjorn Sjostrand et al. High-energy-physics event generation with pythia 6.1. *Comput. Phys. Commun.*, 135:238–259, 2001.

[64] W. Bell et al. *User Guide For EvtGen CDF*, 2001. CDF/DOC/BOTTOM/CDFR/5618.

[65] G. Barker, M. Feindt, C. Lecci, et al. *Monte Carlo study of lepton SVT sample*, 2004. CDF/PHYS/BOTTOM/CDFR/7318.

[66] M Feindt, S. Menzemer, and K. Rinnert. *TrackingKal - A Tracking and Alignment Software Package for the CDFII Silicon Detector*, 2003. CDF/THESIS/TRACKING/PUBLIC/5968.

[67] S. Eidelman, K.G. Hayes, K.A. Olive, M. Aguilar-Benitez, C. Amsler, D. Asner, K.S. Babu, R.M. Barnett, J. Beringer, P.R. Burchat, C.D. Carone, C. Caso, G. Conforto, O. Dahl, G. D'Ambrosio, M. Doser, J.L. Feng, T. Gherghetta, L. Gibbons, M. Goodman, C. Grab, D.E. Groom, A. Gurtu, K. Hagiwara, J.J. Hernández-Rey, K. Hikasa, K. Honscheid, H. Jawahery, C. Kolda, Kwon Y., M.L. Mangano, A.V. Manohar, J. March-Russell, A. Masoni, R. Miquel, K. Mönig, H. Murayama, K. Nakamura, S. Navas, L. Pape, C. Patrignani, A. Piepke, G. Raffelt, M. Roos, M. Tanabashi, J. Terning, N.A. Törnqvist, T.G. Trippe, P. Vogel, C.G. Wohl, R.L. Workman, W.-M. Yao, P.A. Zyla, B. Armstrong, P.S. Gee, G. Harper, K.S. Lugovsky, S.B. Lugovsky, V.S. Lugovsky, A. Rom, M. Artuso, E. Barberio, M. Battaglia, H. Bichsel, O. Biebel, P. Bloch, R.N. Cahn, D. Casper, A. Cattai, R.S. Chivukula, G. Cowan, T. Damour, K. Desler, M.A. Dobbs, M. Drees, A. Edwards, D.A. Edwards, V.D. Elvira, J. Erler, V.V. Ezhela, W. Fetscher, B.D. Fields, B. Foster, D. Froidevaux, M. Fukugita, T.K. Gaisser, L. Garren, H.-J. Gerber, G. Gerbier, F.J. Gilman, H.E. Haber, C. Hagmann, J. Hewett, I. Hinchliffe, C.J. Hogan, G. Höhler, P. Igo-Kemenes, J.D. Jackson, K.F. Johnson, D. Karlen, B. Kayser, D. Kirkby, S.R. Klein, K. Kleinknecht, I.G. Knowles, P. Kreitz, Yu.V. Kuyanov, O. Lahav,

P. Langacker, A. Liddle, L. Littenberg, D.M. Manley, A.D. Martin, M. Narain, P. Nason, Y. Nir, J.A. Peacock, H.R. Quinn, S. Raby, B.N. Ratcliff, E.A. Razuvaev, B. Renk, G. Rolandi, M.T. Ronan, L.J. Rosenberg, C.T. Sachrajda, Y. Sakai, A.I. Sanda, S. Sarkar, M. Schmitt, O. Schneider, D. Scott, W.G. Seligman, M.H. Shaevitz, T. Sjöstrand, G.F. Smoot, S. Spanier, H. Spieler, N.J.C. Spooner, M. Srednicki, A. Stahl, T. Stanev, M. Suzuki, N.P. Tkachenko, G.H. Trilling, G. Valencia, K. van Bibber, M.G. Vincter, D. Ward, B.R. Webber, M. Whalley, L. Wolfenstein, J. Womersley, C.L. Woody, O.V. Zenin, and R.-Y. Zhu. Review of Particle Physics. *Physics Letters B*, 592:1+, 2004.

[68] S. Yu, J Heinrich, et al. *COT dE/dx Measurement and Corrections*, 2003. CDF/DOC/BOTTOM/PUBLIC/6361.

[69] S. D'Auria, D. Lucchesi, et al. *Track-based calibration of the COT specific ionization*, 2004. CDF/ANAL/BOTTOM/CDFR/6932.

[70] A. Scheurer, COT dE/dx studies private communication.

[71] C. Peterson, Th. Rognvaldsson, and L. Lonnblad. *JETNET 3.0: A Versatile artificial neural network package. Comput. Phys. Commun.*, 81:185–220, 1994.

[72] Ph. Koehn and C. Ciobanu. *Quick root_to_jetnet Tutorial.* `http://www.hep.uiuc.edu/home/catutza/root_to_jetnet`.

[73] A. Affolder. *A Measurement of Bottom Quark-Antiquark Azimuthal Production Correlations*, 2003. CDF/THESIS/BOTTOM/PUBLIC/6263.

[74] K. Anikeev, P. Murat, and Ch. Paus. *Description of Bgenerator II* , 1999. CDF/DOC/BOTTOM/CDFR/5092.

[75] E. Barberio and Z. Was. *PHOTOS: A Universal Monte Carlo for QED radiative corrections. Version 2.0. Comput. Phys. Commun.*, 79:291–308, 1994.

[76] N. Cabibbo. *Unitary symmetry and leptonic decays. Phys. Rev. Lett.*, 10:531–532, 1963.

[77] M. Kobayashi and T. Maskawa. *CP violation in the renormalizable theory of weak interaction. Prog. Theor. Phys.*, 49:652–657, 1973.

[78] A. J. Buras, W. Slominski, and H. Steger. *B0 anti-B0 mixing, CP violation and the B meson decay. Nucl. Phys.*, B245:369, 1984.

[79] C. Albajar et al. *Search for B0 anti-B0 oscillations at the CERN proton - anti-proton collider. (paper 2.). Phys. Lett.*, B186:247, 1987.

[80] H. Albrecht et al. *Observation of B0 - anti-B0 mixing. Phys. Lett.*, B192:245, 1987.

[81] C. Lecci. *A Neural Jet Charge Tagger For The Measurement Of The $B_s^0 - \bar{B}_s^0$ Oscillation Frequency At CDF*. PhD thesis, Institut für Experimentelle Kernphysik, University of Karlsruhe, 2005.

[82] `http://cdfkits.fnal.gov/CdfCode/source/BottomTaggers`, `http://cdfkits.fnal.gov/CdfCode/source/BottomAnalysis`.

[83] M. Jones, D. Usynin, J. Kroll, and B. Wicklund. *Sample Composition of the ℓ+SVT Triggers*, 2003. CDF/ANAL/BOTTOM/CDFR/6480.

[84] B. Wicklund. *Evaluation of errors on ϵ, $\mathcal{D}$, $\epsilon\mathcal{D}^2$*, 2003. CDF/DOCL/BOTTOM/CDFR/6716.

[85] P. Marriner. *Secondary Vertex fit with mass and pointing constraints (*`CTVMFT`*)*, 1996. CDF/DOC/SEC_VTX/PUBLIC/1996.

[86] G. Bauer, Ch. Paus, and K. Sumorok. *Observation of the Charmonium State $\psi(3870)$ in $J/\psi\pi^+\pi^-$ with Run II Data*, 2003. CDF/DOC/BOTTOM/CDFR/6669.

[87] G. Bauer, Ch. Paus, A. Rakitin, and K. Sumorok. *Measurement of the Dipion Mass Spectrum in $X(3872) \to J/\psi\pi^+\pi^-$ Decays*, 2005. CDF/DOC/BOTTOM/PUBLIC/7570.

[88] M. Feindt. *An Amplitude Construction Primer for Experimentalists.* private communication.

[89] J. D. Richmann. *An experimenter's guide to the helicity formalism.* Technical Report CALT-68-1148, Caltech, 1968.

[90] S. U. Chung. *A General formulation of covariant helicity coupling amplitudes*. *Phys. Rev.*, D57:431–442, 1998.

[91] M. Feindt, J. Heuser, and U. Kerzel. *Analysis of the quantum numbers of the $X(3872)$*, 2005. CDF/DOC/BOTTOM/CDFR/7311.

[92] J.M. Blatt and V. Weisskopf. *Theoretical Nuclear Physics,*. John Wiley, New York, 1952.

[93] J. D. Jackson. *Remarks on the phenomenological analysis of resonances. Nuovo Cim.*, 34:1644–1666, 1964.

[94] Mikhail B. Voloshin and Valentin I. Zakharov. *Measuring QCD anomalies in hadronic transitions between onium states. Phys. Rev. Lett.*, 45:688, 1980.

[95] J. Z. Bai et al. *$\psi(2S) \to \pi^+\pi^- J/\psi$ decay distributions. Phys. Rev.*, D62:032002, 2000.

[96] Standard soft electron cuts. `http://cdfkits.fnal.gov/CdfCode/source/SoftElectronObjects/src/StandardSoftElectronCut.cc`.

[97] S. Menzemer. *Latest optimisations of the time-of-flight detector simulation.* private communication.

[98] F. Abe et al. *J/ψ and $\psi(2S)$ production in p anti-p collisions at $\sqrt{s} = 1.8$ TeV. Phys. Rev. Lett.*, 79(572), 1997.

[99] K. Sumorok. *The RunI $\psi(2S)$ p_t spectrum.* private communication.

Acknowledgements

I would like to thank my supervisor Professor Dr. Michael Feindt for accepting me in this group and offering me this interesting and challenging topic. His motivating support, the many fruitful discussions and ideas have lead me to a deeper understanding of physics during the course of this work.
I would also like to thank Professor Dr. Günter Quast for his help and support with the Grid-computing related part of this work and for co-supervising this thesis.
Furthermore, I would like to thank Professor Dr. Thomas Müller and Professor Dr. Michael Feindt for making it possible for me to stay an entire year at Fermilab and present the work at several conferences. Many thanks also to our secretaries Mrs Haas and Mr Fuchs for their help with the practical matters of contracts and travel-allowance.
I am also deeply indebted to Dr. Richard St.Denis, Dr. Stefan Stonjek, Dr. Sinisa Veseli, Lauri Loebel-Carpenter, Andrew Baranowski and Gabriele Garzoglio for their invaluable help with the Grid software.
I express my thanks to Dr. Gary Barker for the many fruitful discussions, his help during my time at the institute and for the nice time we had sharing an office. I'm also indebted to Dr. Kurt Rinnert for both introducing me to the depths of CDF software and sharing the responsibility of representing the CDF experiment in the technical advisory board of the GridKa computing centre.
I would also like to thank Joachim Heuser for the close collaboration regarding the helicity analysis and reading parts of this manuscript.
I thank Dr. Claudia Lecci for her help, many of the simulated events used in this work were prepared by her. I would also like to thank Michael Milnik for the close collaboration while developing a common framework for lepton tagging, integrating both our methods.
I would also like to express my gratitude to my fellow admins for keeping the many computers going at all times.
I am also indebted to Dr. Thomas Kuhr for carefully reading and commenting on this manuscript and the resulting CDF notes.
The members of the CDF B group provided a nice atmosphere in the discussion of the work presented here. I'd like to thank them for the insights gained in these

discussion and meetings.
This work was supported by scholarships of the Land of Baden-Würtemberg and the "Graduiertenkolleg Hochenergiephysik und Teilchenastrophysik" promoted by the German Research Community and the Federal Ministry for Education and Research. I thank them for my scholarship.
Special thanks also Michael Milnik and Thorsten Scheidle for the nice time we had sharing an office. Many thanks also to Dr. Stephanie Menzemer, Dr. Else Lytken, Dr. Farrukh Azfar, Dr. Jörgen Sjölin, Dr. Jonatan Piedra and Jan Ehlers for the nice time during the activities at Fermilab not related to physics. I would also like to thank my colleagues at the institute and the "EKP AllStars" for creating a very pleasant atmosphere to work in and the many activities not related to work.
Last but not least I would like to express my gratitude to my parents for their continuous help and support.