QCD FITS TO NEUTRINO-IRON STRUCTURE FUNCTIONS AT NUTEV

by

Voica A. Radescu

MS in Physics, University of Pittsburgh, 2001

Submitted to the Graduate Faculty of
the Department of Physics and Astronomy in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

University of Pittsburgh

2006

UNIVERSITY OF PITTSBURGH

DEPARTMENT OF PHYSICS AND ASTRONOMY

This dissertation was presented

by

Voica A. Radescu

It was defended on

August 3, 2006

and approved by

Dr. Donna Naples, Associate Professor, Dept. of Physics and Astronomy

Dr. Paul Shepard,Professor, Dept. of Physics and Astronomy

Dr. Vittorio Paolone, Associate Professor, Dept. of Physics and Astronomy

Dr. Adam Leibovich, Assistant Professor, Dept. of Physics and Astronomy

Dr. Rob Coalson, Professor, Dept. of Chemistry/Dept. of Physics and Astronomy

Dissertation Director: Dr. Donna Naples, Associate Professor, Dept. of Physics and Astronomy

ABSTRACT

QCD FITS TO NEUTRINO-IRON STRUCTURE FUNCTIONS AT NUTEV

Voica A. Radescu, PhD

University of Pittsburgh, 2006

This thesis presents a new determination of Λ_{QCD} from Next-to-Leading Order QCD fits to the Q^2 dependence of neutrino-iron structure functions. This is the first measurement of Λ_{QCD} which uses a theoretical model that fully accounts for heavy quark production. Compared with previous neutrino measurements, the result has improved understanding of the largest systematic uncertainties on the muon and hadron energy scales. These improvements lead to one of the most precise determination of α_S at moderate Q^2.

NuTeV is a neutrino-iron deep inelastic scattering (DIS) experiment that collected data during 1996-97 at Fermilab. The key features of NuTeV include its sign-selected beam which produced separate high purity neutrino and antineutrino beams, and its continuous calibration beam which enabled NuTeV to considerably improve the knowledge of energy scales which have dominated uncertainties in the previous measurements.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

I here would like to thank the people who made this thesis possible. I have the pleasure to thank Donna Naples, my thesis advisor and also my friend who witnessed to my growth during these PhD years. She always encouraged me and provided me support. She taught me how to stand up for myself, and exposed me to several conferences that allowed me to meet extraordinary people in the field.

I am honored for being given the chance of working with Jeff Owens and Fred Olness. I would like to address my special thanks to both of them. This thesis would not have been possible without their remarkable patience and prompt guidance in this complex QCD analysis.

I thank to Jeff McDonald for helping me getting started in a field which was new to me. I am particularly grateful to him for learning various high energy physics computer skills. Many thanks to Steve Boyd, although he might enjoy the darkness, he surely shed a lot of light into the NuTeV analysis. Also, I would like to thank Martin Tzanov who convinced me to join the group and with whom I worked closely on the NuTeV analysis.

I would like to thank the NuTeV collaboration for their monthly feedback. Especially, I thank to Bob Bernstein, Mike Shaevitz, Randy Johnson, Arie Bodek, and Un-Ki Yang for various physics discussions and ideas. Many thanks to Dave Mason who always brought some humor during the collaboration meetings, and moreover strongly encouraged me to contact Fred Olness about the QCD code.

I would like to thank my committee members for careful reading and for suggestions in improving the manuscript. Many thanks to my friend Ted Rogers for going through the thesis and correcting my English.

I am grateful to all my friends who touched my life one way or another giving me strength needed in this lengthy road of PhD.

I could not end this preface without addressing a warm thank you to my parents for inspiring me to aspire, and to Lorenzo for his moral support and belief in everything I do.

1.0 INTRODUCTION

Neutrino-scattering experiments have strongly influenced the development of particle physics. The purely weak nature of neutrino interactions makes them a unique probe of nucleon structure. The focus of this thesis is to determine the coupling strength from the scale dependence of nucleon structure functions. This is a first measurement which takes quark masses into account. The measurement was performed at a low to moderate scale where there are only a few high precision measurements. It is a sensitive test of predictions of the fundamental theory of strong interactions, Quantum Chromodynamics (QCD).

In this Chapter the formalism for neutrino deep inelastic scattering (DIS) and the theory of strong interactions is discussed. This thesis is organized as follows:

- **Chapter 2** provides a description of the NuTeV detector and neutrino beamline. The calibration of the detector and event reconstruction is also described here.

- **Chapter 3** focuses on the method of extracting the differential cross sections, the structure functions, and the neutrino flux. This Chapter also presents a discussion of the systematic errors.

- **Chapter 4** presents the QCD fits used to determine Λ_{QCD} with error analysis, and compares the results with existing measurements.

- **Chapter 5** concludes and discusses future prospects for DIS measurements.

1.1 STANDARD MODEL AND ELEMENTARY PARTICLES

Particle physics aims to describe the fundamental particles of the universe and their interactions. The Standard Model, developed in the 1970's, is a triumph of particle physics describing concisely the electromagnetic, weak, and strong interactions, as well as explaining the large variety of experimental observations.

In the Standard Model, the elementary particles are grouped into two categories of gauge bosons and fermions. Bosons are the integer spin particles which act as carriers of forces. Fundamental bosons – gluons, Z^0, $W^{\pm}$, and photons – mediate the strong, weak, and electromagnetic interactions, respectively. Table 1.1 summarizes these particles and their interactions.

Fermions are half-integer spin particles that obey Pauli's exclusion principle and are the primary constituents of matter. They are divided further into two groups depending on which interactions they participate in: quarks and leptons.

Quarks are the only fundamental particles to experience the strong force. They also couple via all of the other fundamental interactions because they carry all of the possible charges: color, electric and weak isospin. Quarks have never been observed in free isolated states, but alway in bound states. For example, baryons represent the bound state of three quarks, and mesons are quark-antiquark pairs. Baryons and mesons are collectively named hadrons.

Leptons participate in weak interactions, and if they carry electric charge, also in electromagnetic interactions. They can exist and be observed as free particles.

Quarks and Leptons can be classified into three generations of doublets. To complete the picture of the Standard Model, each particle is associated with an anti-particle having the same mass and carrying the same charge, but of opposite sign. Fundamental fermions of the Standard Model are listed in Table 1.2 divided in three generation by successively increasing mass (from the top row down to the bottom row of the table). All the visible matter is made of the first generation particles, i.e. protons and neutrons are made of u and d quarks. Each generation of leptons consists of a charged and neutral lepton. The electron is the only stable particle among the charged leptons. Neutral leptons are called neutrinos

Table 1.1: Fundamental forces described in the Standard Model with relative strengths evaluated for $Q = 1$ GeV

Force	Relative strength	Participating particles	Mediating boson
Strong	1	quarks	gluon
Electromagnetic	1.4×10^{-2}	quarks and charged leptons	photon
Weak	2.2×10^{-6}	all	$Z^0,\ W^\pm$

and have very small mass. Quarks occur in six varieties of flavors: up, down, strange, charm, bottom, top. They carry fractional electric charge.

1.2 DEEP INELASTIC SCATTERING AND QUARK PARTON MODEL

The first convincing evidence for the existence of point-like constituents in the nucleon came from deep inelastic scattering (DIS) experiments performed at SLAC in the 1970's [1]. DIS is the term for the scattering of a lepton off constituents of a nucleon producing a lepton and a shower of hadrons in the final state. The scattering is called "deep" because high energy particles probe deep within the proton, and "inelastic" because the target is changed, i.e. the proton is disrupted. The kinematics of a lepton-nucleon DIS experiment is sketched in Figure 1.1.

1.2.1 Kinematics of $\nu - N$ Deep Inelastic Scattering

The interaction of a neutrino with a nucleon occurs via exchange of an electrically charged boson $W^\pm$ (charged-current interaction) or an electrically neutral boson Z^0 (neutral-current interaction). A muon-neutrino charged-current interaction has a distinctive signature that

Table 1.2: Fermionic elementary particles of the Standard Model

LEPTONS (spin= 1/2)			**QUARKS (spin= 1/2)**		
Flavor	Mass (GeV/c^2)	Electric charge	Flavor	Approx. Mass (GeV/c^2)	Electric charge
$\boldsymbol{\nu_e}$ electron neutrino	$< 1 \times 10^{-8}$	0	$\boldsymbol{u}$ up	0.003	2/3
$\boldsymbol{e}$ electron	0.000511	-1	$\boldsymbol{d}$ down	0.006	-1/3
$\boldsymbol{\nu_\mu}$ muon neutrino	< 0.0002	0	$\boldsymbol{c}$ charm	1.3	2/3
$\boldsymbol{\mu}$ muon	0.106	-1	$\boldsymbol{s}$ strange	0.1	-1/3
$\boldsymbol{\nu_\tau}$ tau neutrino	< 0.02	0	$\boldsymbol{t}$ top	175	2/3
$\boldsymbol{\tau}$ tau	1.7771	-1	$\boldsymbol{b}$ bottom	4.3	-1/3

produces its associated charged lepton in the final state,

$$\nu_\mu(\overline{\nu}_\mu) + N \rightarrow \mu^-(\mu^+) + X, \tag{1.1}$$

where N is the nucleon and X denotes the outgoing hadrons.

Figure 1.2 shows the scattering of an incoming high-energy neutrino with four-momentum k, off a hadron target with four-momentum P, and mass M. In the process, a charged W boson with four-momentum q is exchanged. An outgoing muon with four-momentum k' and a hadronic shower are produced in the final state. The measured quantities in the laboratory frame are the muon momentum p_μ, the angle θ_μ of the outgoing muon, and the energy E_{had} of the outgoing shower of hadrons. The kinematic quantities of DIS are measured in the laboratory frame in terms of experimental quantities as

$$k = (E_\nu, 0, 0, E_\nu), \tag{1.2}$$

$$k' = (E_\mu, p_\mu \sin\theta_\mu \cos\phi_\mu, p_\mu \sin\theta_\mu \sin\phi_\mu, p_\mu \cos\theta_\mu), \tag{1.3}$$

$$P = (M, 0, 0, 0). \tag{1.4}$$

We want to express these quantities in terms of the following Lorentz-invariant variables:

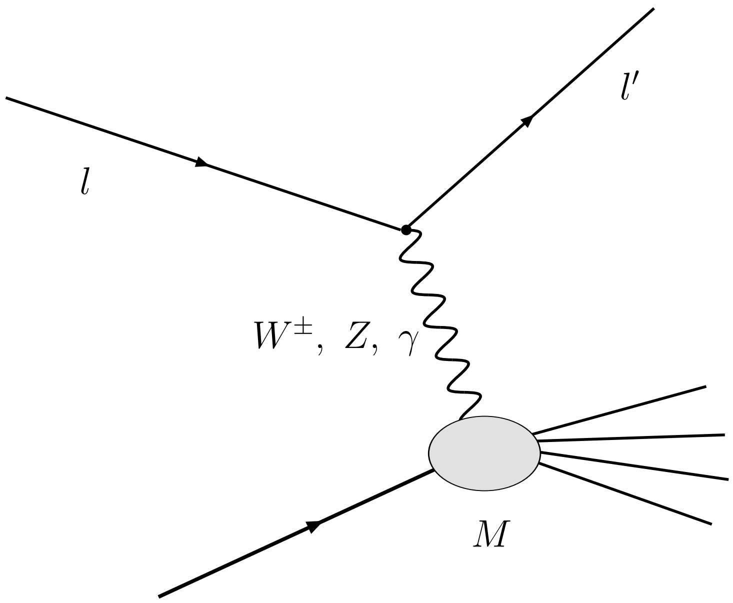


Figure 1.1: Deep inelastic scattering of a lepton off the quark constituents of a nucleon producing a lepton and a shower of hadrons in the final state.

- $\boldsymbol{Q^2}$, the negative four-momentum squared of the exchanged boson that defines the energy scale of the interaction

$$Q^2 = -q^2 = 2(E_\mu E_\nu - \vec{k} \cdot \vec{k}') - m_\mu^2; \tag{1.5}$$

- $\boldsymbol{\nu}$, the energy transfered from the leptonic to the hadronic system

$$\nu = \frac{q \cdot P}{M} = E_\nu - E_\mu = E_{had} - M; \tag{1.6}$$

- **inelasticity** $\boldsymbol{y}$, the fraction of the lepton's energy lost in the nucleon rest frame

$$y = \frac{q \cdot P}{k \cdot P} = \frac{E_{had}}{E_\nu}, \tag{1.7}$$

  which is related to the lepton scattering angle θ^* in the center-of-mass frame,

$$y \approx 1 - \frac{1}{2}(1 + \cos\theta^*). \tag{1.8}$$

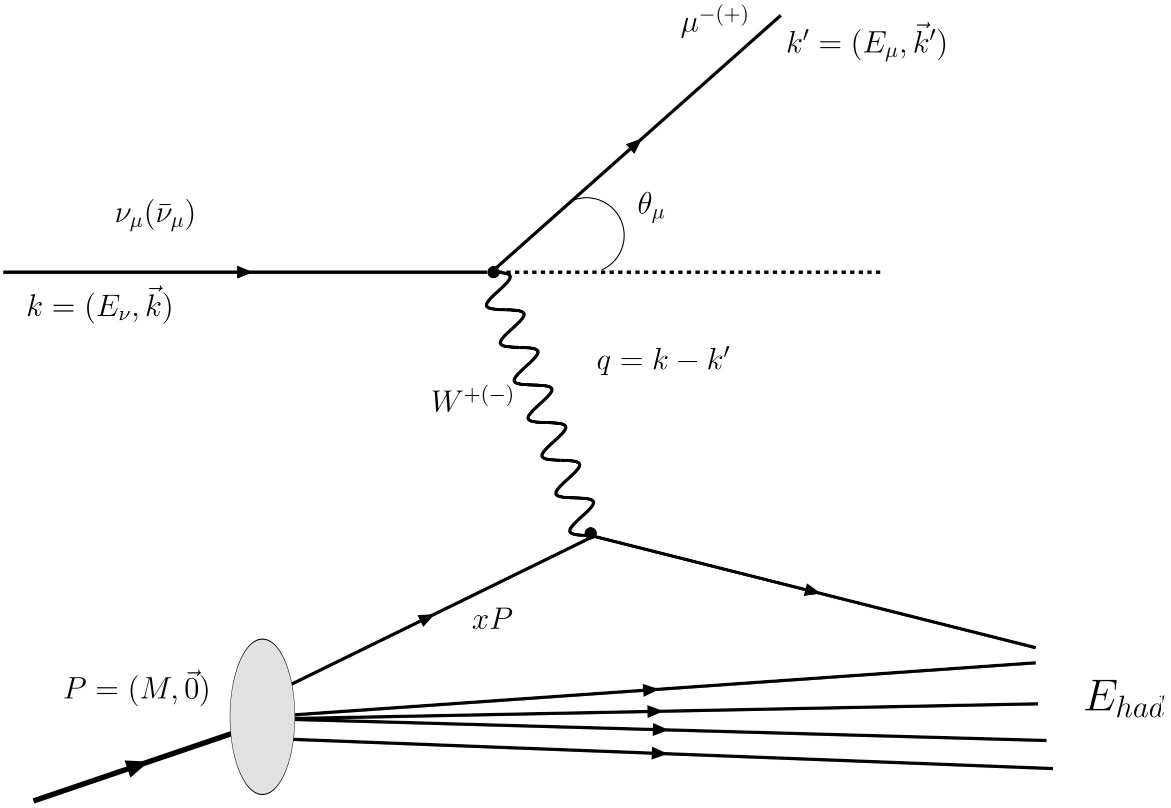


Figure 1.2: Deep Inelastic Charged Neutrino-Proton Scattering.

- **Bjorken scaling variable $\boldsymbol{x}$**, the fractional momentum of the struck parton inside the nucleon [2]

$$x = \frac{-q^2}{2P \cdot q} = \frac{Q^2}{2M\nu}; \tag{1.9}$$

- $\boldsymbol{W^2}$, the invariant mass of the hadronic system

$$W^2 = (P+q)^2 = M^2 + 2M\nu - Q^2; \tag{1.10}$$

- $\boldsymbol{s}$, the center-of-mass energy squared of the lepton-nucleon system

$$s = (k+P)^2 = \frac{Q^2}{xy} + M^2. \tag{1.11}$$

The process in Figure 1.2 is called deep ($Q^2 \gg M^2$) inelastic ($W^2 \gg M^2$) scattering.

The cross section can be written using Fermi's golden rule

$$d\sigma = \frac{1}{flux} \times |\mathcal{M}|^2 \times d\Gamma, \tag{1.12}$$

where $\mathcal{M}$ is the invariant scattering amplitude, $d\Gamma$ is the phase space of the outgoing particles and $flux$ is the incident-particle flux. The matrix element $\mathcal{M}$ can be expressed in terms of the leptonic and hadronic currents:

$$\mathcal{M} = \frac{G_F}{\sqrt{2}} \frac{1}{1 + \frac{Q^2}{M_W^2}} \underbrace{\overline{u}_\mu \gamma_\alpha (1 - \gamma_5) u_\nu}_{leptonic} \underbrace{< X | J_h^\alpha | h >}_{hadronic}, \tag{1.13}$$

where G_F is the weak coupling constant, M_W is the mass of the W boson, and $\overline{u}_\mu$, u_ν are the lepton spinors. The $\gamma_\alpha(1 - \gamma_5)u_\nu$ is the V-A ("vector minus axial vector") interaction, where γ_5 is one of the Dirac gamma matrices:

$$\gamma^0 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \tag{1.14}$$

$$\gamma^i = \begin{pmatrix} 0 & \sigma^i \\ -\sigma^i & 0 \end{pmatrix}, i = 1, .., 3; \tag{1.15}$$

$$\gamma^5 = i\gamma^0\gamma^1\gamma^2\gamma^3 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \tag{1.16}$$

and σ^i are the Pauli spin matrices. The differential cross section can be factorized into leptonic $L_{\alpha\beta}$ and hadronic $H^{\alpha\beta}$ components as

$$\frac{d^2\sigma_{CC}^{\nu N}}{dxdy} \propto L_{\alpha\beta} H^{\alpha\beta}. \tag{1.17}$$

The leptonic tensor is given by

$$L_{\alpha\beta} = [\overline{u}(k)\gamma_\alpha(1 - \gamma_5)u(k')][\overline{u}(k')\gamma_\beta(1 - \gamma_5)u(k)]. \tag{1.18}$$

A general form for the hadronic tensor can be constructed from the available four-vectors (p^μ and q^μ) and the isotropic tensors ($g^{\alpha\beta}$ and $\epsilon^{\alpha\beta\gamma\delta}$):

$$\begin{aligned} H^{\alpha\beta} &= -g^{\alpha\beta}W_1 + \frac{p^\alpha p^\beta}{M^2}W_2 - \frac{i\epsilon^{\alpha\beta\gamma\delta}p_\gamma q_\delta}{2M^2}W_3 + \qquad (1.19) \\ &+ \frac{q^\alpha q^\beta}{M^2}W_4 + \frac{p^\alpha q^\beta + p^\beta q^\alpha}{M^2}W_5 + i\frac{p^\alpha q^\beta - p^\beta q^\alpha}{2M^2}W_6. \qquad (1.20) \end{aligned}$$

The W_i are called structure functions which depend only on scalars q^2 and $p \cdot q$.

The time reversal invariance of QCD implies that $W_6 = 0$. The hadronic current is conserved, $\partial_\nu J_h^\nu = 0$, hence

$$q_\mu \cdot H^{\mu\nu} = 0 \text{ and } H^{\mu\nu} \cdot q_\nu = 0. \qquad (1.21)$$

As a consequence,

$$\begin{aligned} W_5 &= -\frac{p \cdot q}{q^2}W_2, \qquad (1.22) \\ W_4 &= \left(\frac{p \cdot q}{q^2}\right)^2 W_2 + \frac{M^2}{q^2}W_1. \qquad (1.23) \end{aligned}$$

Evaluating the contraction of leptonic and hadronic tensors, the W_4 and the W_5 terms are proportional to the lepton mass and can be neglected because they are small relative to other scales. This leaves us with three remaining structure functions W_1, W_2 and W_3. The latter is the parity-violating structure function, accessible only by neutrino scattering.

In deep inelastic scattering, the nucleon is resolvable for $Q, \nu \gg M$, and the structure functions depend on the Lorentz scalar variables ν and Q^2. In the DIS regime ($Q, \nu \gg M$), large Q^2 resolves point-like constituents inside the proton. In the $Q^2, \nu \to \infty$ limit, the functions $W_i(Q^2, \nu)$ depend (approximately) only on the ratio $x = Q^2/2M\nu$. This feature of structure functions is known as Bjorken scaling [2]. However, experimental observations show small Q^2 dependences of the structure functions (scaling violations).

The dimensionless structure functions are defined as follows:

$$\begin{aligned} 2xF_1 &\equiv 2xMW_1, \qquad (1.24) \\ F_2 &\equiv \nu W_2, \qquad (1.25) \\ xF_3 &\equiv \nu x W_3. \qquad (1.26) \end{aligned}$$

Thus, the general form for differential neutrino-nucleon DIS cross-section can be written in terms of three independent structure functions $2xF_1(x,Q^2)$, $F_2(x,Q^2)$ and $xF_3(x,Q^2)$ as

$$\frac{d^2\sigma^{\nu(\overline{\nu})N}}{dxdy} = \frac{G_F^2 M E_\nu}{\pi(1+Q^2/M_W^2)^2}\left[\frac{y^2}{2}2xF_1^{\nu(\overline{\nu})N}+\left(1-y-\frac{Mxy}{2E_\nu}\right)F_2^{\nu(\overline{\nu})N}\pm y\left(1-\frac{y}{2}\right)xF_3^{\nu(\overline{\nu})N}\right]. \quad (1.27)$$

The differential cross section is often expressed in terms of the longitudinal structure function $R_L(x,Q^2)$ defined as the ratio of the cross-sections for longitudinally (σ_L) polarized to transversely (σ_T) polarized W bosons. The interpretation of R_L can be inferred by expressing the structure functions in terms of cross-sections for longitudinally (σ_L) and transversely (σ_T) polarized W boson:

$$F_1 = \frac{MK}{\pi G_F\sqrt{2}}\sigma_T, \quad (1.28)$$

$$F_2 = \frac{\nu K}{\pi G_F\sqrt{2}}\frac{Q^2}{Q^2+\nu^2}(\sigma_T+\sigma_L), \quad (1.29)$$

where K denotes the flux of virtual bosons, and $\nu = Q^2/2Mx$. Thus, R_L relates F_2 to $2xF_1$ as

$$R_L = \frac{\sigma_L}{\sigma_T} = \frac{F_2(x,Q^2)(1+4M^2x^2/Q^2)-2xF_1(x,Q^2)}{2xF_1(x,Q^2)}. \quad (1.30)$$

In this analysis we will use the neutrino-nucleon differential cross-section expressed in terms of R_L, F_2, and xF_3 as

$$\frac{d^2\sigma^{\nu(\overline{\nu})N}}{dxdy} = \frac{G_F^2 M E_\nu}{\pi(1+Q^2/M_W^2)^2}\left[\left(\frac{y^2+\frac{4M^2x^2y^2}{Q^2}}{2+2R_L^{\nu(\overline{\nu})N}}+1-y-\frac{Mxy}{2E_\nu}\right)F_2^{\nu(\overline{\nu})N}\pm y\left(1-\frac{y}{2}\right)xF_3^{\nu(\overline{\nu})N}\right]. \quad (1.31)$$

1.2.2 Structure Functions in The Quark Parton Model

Early evidence for Bjorken scaling in electron DIS scattering experiments [1] led to the formulation of the parton model (Bjorken and Pashos, 1969; Feynman, 1972). In this model, the hadron is viewed as an assembly of non-interacting point-like particles named partons, which are interpreted as quarks and gluons. This model is valid in the "infinite momentum frame" in which the characteristic time scale for the strong interactions among partons inside the nucleon is much longer than the collision time. In this frame, the neutrino-nucleon scattering can be viewed as elastic scattering between neutrinos and point-like partons inside the nucleon which carry a fraction x of nucleon's momentum.

Structure functions are described in terms of momentum-weighted parton distribution functions (PDFs) which describe the probability that the interacting parton carries a fraction x of the hadron's longitudinal momentum.

Elastic scattering of two point particles depends on the center-of-mass energy and the net angular momentum of the system. Neutrinos and anti-neutrinos are spin-$\frac{1}{2}$ fermions, hence the total angular momentum from the scattering of two spin-$\frac{1}{2}$ particles may be either 0 or ± 1. The V-A nature of the weak charged current operator selects only left-handed particles and right-handed anti-particles. Therefore, there are only four possible helicity configurations for scattering of two spin-$\frac{1}{2}$ particles as illustrated in Figure 1.3. Neutrinos are left-handed (helicity $= -1$), hence their spin is aligned in the opposite direction as their momentum, while antineutrinos are right-handed (helicity $= +1$) with their spin aligned in the same direction as their momentum. As a result of angular momentum conservation, the spin-zero contributions to the total charged current cross sections are given by νq ($\overline{\nu q}$) interactions, while the spin-one contributions are given by $\nu\bar{q}$ ($\overline{\nu}q$) interactions. As shown in Figure 1.3, the spin-zero configuration has no angular dependence (i.e. no y dependence), while the spin-one has $(1-y)^2$ dependence.

From the above helicity arguments, the neutrino differential cross sections for elastic

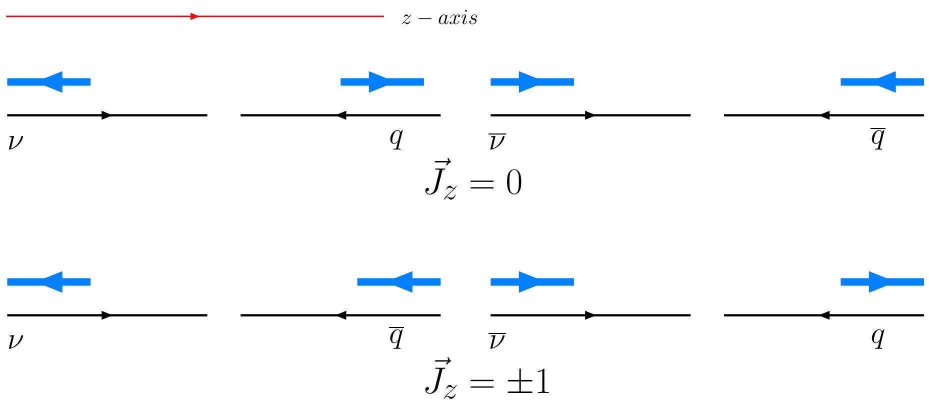


Figure 1.3: Possible helicity configurations for neutrinos and anti-neutrinos scattering from quarks and antiquarks. The top rows depict the configurations for $\vec{J}_z = 0$, without preferred scattering direction. The bottom rows shows the configurations for $\vec{J}_z = \pm 1$.

scattering of neutrinos with spin-$\frac{1}{2}$ partons can be derived as

$$\frac{d^2\sigma^{\nu}}{dxdy} = \frac{2G_F^2 ME}{\pi}[xq(x) + (1-y)^2 x\overline{q}(x)], \tag{1.32}$$

$$\frac{d^2\sigma^{\overline{\nu}}}{dxdy} = \frac{2G_F^2 ME}{\pi}[x\overline{q}(x) + (1-y)^2 xq(x)]. \tag{1.33}$$

We can relate neutrino structure functions to the parton densities from Eqs. (1.27), (1.32) and (1.33),

$$2xF_1^{\nu(\overline{\nu})}(x,Q^2) = 2[xq^{\nu(\overline{\nu})}(x) + x\overline{q}^{\nu(\overline{\nu})}(x)], \tag{1.34}$$

$$xF_3^{\nu(\overline{\nu})}(x,Q^2) = 2[xq^{\nu(\overline{\nu})}(x) - x\overline{q}^{\nu(\overline{\nu})}(x)]. \tag{1.35}$$

For purely spin-$\frac{1}{2}$ constituents, the Callan-Gross relation [3] holds,

$$2xF_1 = F_2, \tag{1.36}$$

as confirmed by early experiments at SLAC [1].

Neutrino charged current scattering resolves the flavor of nucleon quarks due to the charge conservation at the quark vertex. Thus, neutrinos can scatter only from d, s, $\overline{u}$, $\overline{c}$

quarks, while antineutrinos scatter only from $\overline{d}$, $\overline{s}$, u, c quarks. For a proton target, the quark densities contributing to the structure functions are

$$\begin{aligned} q^{\nu p}(x) = d^p(x) + s^p(x), \quad q^{\overline{\nu} p}(x) = u^p(x) + c^p(x), \\ \overline{q}^{\nu p}(x) = \overline{u}^p(x) + \overline{c}^p(x), \quad \overline{q}^{\overline{\nu} p}(x) = \overline{d}^p(x) + \overline{s}^p(x). \end{aligned} \tag{1.37}$$

For a neutron target, the quark densities are

$$\begin{aligned} q^{\nu n}(x) = d^n(x) + s^n(x), \quad q^{\overline{\nu} n}(x) = u^n(x) + c^n(x), \\ \overline{q}^{\nu n}(x) = \overline{u}^n(x) + \overline{c}^n(x), \quad \overline{q}^{\overline{\nu} n}(x) = \overline{d}^n(x) + \overline{s}^n(x). \end{aligned} \tag{1.38}$$

The quark content of a proton and a neutron is related through isospin symmetry

$$\begin{aligned} u^p(x) = d^n(x) \equiv u(x), \quad \overline{u}^p(x) = \overline{d}^n(x) \equiv \overline{u}(x), \\ d^p(x) = u^n(x) \equiv d(x), \quad \overline{d}^p(x) = \overline{u}^n(x) \equiv \overline{d}(x), \\ s^p(x) = s^n(x) \equiv s(x), \quad \overline{s}^p(x) = \overline{s}^n(x) \equiv \overline{s}(x), \\ c^p(x) = c^n(x) \equiv c(x), \quad \overline{c}^p(x) = \overline{c}^n(x) \equiv \overline{c}(x). \end{aligned} \tag{1.39}$$

Therefore, for an isoscalar target, which has equal number of protons and neutrons, we have

$$\begin{aligned} q^{\nu N}(x) = \frac{1}{2}\left[u(x) + d(x) + 2s(x)\right], \quad q^{\overline{\nu} N}(x) = \tfrac{1}{2}\left[u(x) + d(x) + 2c(x)\right], \\ \overline{q}^{\nu N}(x) = \frac{1}{2}\left[\overline{u}(x) + \overline{d}(x) + 2\overline{c}(x)\right], \quad \overline{q}^{\overline{\nu} N}(x) = \tfrac{1}{2}\left[\overline{u}(x) + \overline{d}(x) + 2\overline{s}(x)\right]. \end{aligned} \tag{1.40}$$

The neutrino-nucleon structure functions $2xF_1^{\nu N}$ and $xF_3^{\nu N}$ for an isoscalar target are obtained by using Eqs. (1.34), (1.35),

$$2xF_1^{\nu N} = xu(x) + xd(x) + 2xs(x) + x\overline{u}(x) + x\overline{d}(x) + 2x\overline{c}(x), \tag{1.41}$$

$$xF_3^{\nu N} = xu(x) + xd(x) + 2xs(x) - x\overline{u}(x) - x\overline{d}(x) - 2x\overline{c}(x). \tag{1.42}$$

For antineutrinos, only the positively charged quarks contribute

$$2xF_1^{\overline{\nu} N} = xu(x) + xd(x) + 2xs(x) + x\overline{u}(x) + x\overline{d}(x) + 2x\overline{c}(x), \tag{1.43}$$

$$xF_3^{\overline{\nu} N} = xu(x) + xd(x) - 2xs(x) - x\overline{u}(x) - x\overline{d}(x) + 2x\overline{c}(x). \tag{1.44}$$

The quarks can be separated into valence quarks [66], which carry charge and sea quarks and antiquarks which carry no net charges. The quark and antiquark distributions are given by

$$xq(x) = xq_v(x) + xq_s(x), \tag{1.45}$$

$$x\overline{q}(x) = x\overline{q}_s(x), \tag{1.46}$$

$$xq_s(x) = x\overline{q}_s(x), \tag{1.47}$$

where q_v denotes the valence quark distribution, q_s is the sea quark distribution, and $\overline{q}$ the antiquark distribution. To characterize structure functions, we note that $2xF_1^{\nu(\overline{\nu})N}$ is related to the sum of quark and antiquark distributions in the nucleon. The average xF_3^{ave} structure function,

$$xF_3^{ave} = \frac{xF_3^{\nu N} + xF_3^{\overline{\nu} N}}{2} = xu_v + xd_v, \tag{1.48}$$

is sensitive to valence-quark distributions. The difference ΔxF_3,

$$\Delta xF_3 = xF_3^{\nu N} - xF_3^{\overline{\nu} N} = 4(xs - xc), \tag{1.49}$$

is sensitive to the strange and charm distributions.

For charged-leptons the scattering is mediated by virtual photons. In this case the structure function xF_3 is not present. The electromagnetic charged-lepton structure functions are sensitive to quark charges e_i,

$$2xF_1^{lN} = \sum_i e_i^2 \left(xq_i^l + x\overline{q}_i^l \right). \tag{1.50}$$

1.3 QUANTUM CHROMODYNAMICS

The quark parton model does not consider dynamics between quarks. The theory that describes strong interactions among colored quarks and gluons is called Quantum Chromodynamics (QCD) [Fritzsch et al., 1973; Gross and Wilczek, 1973; Weinberg 1973]. The theory of QCD was born as a result of assembling many ideas and experimental results. In terms of the group symmetries, QCD is the $SU(3)$ component of the $SU(3) \times SU(2) \times U(1)$ Standard Model of the fundamental particles.

The QCD Lagrangian density that describes the interactions of quarks and gluons is given, up to gauge-fixing terms, by

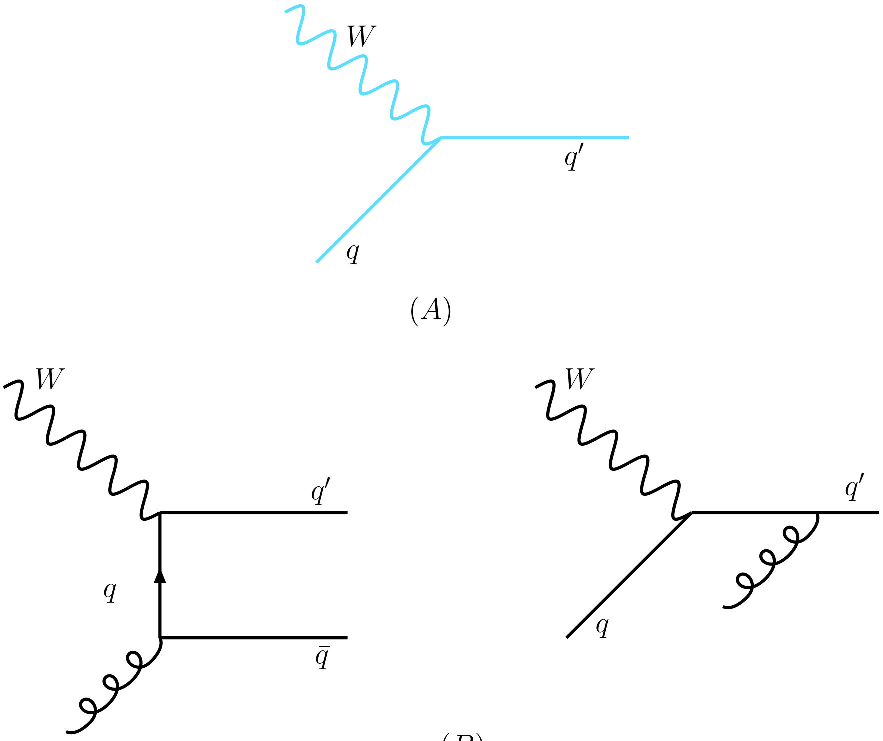
$$\begin{aligned} \mathrm{L}_{QCD} &= -\frac{1}{4}F^{(a)}_{\mu\nu}F^{(a)\mu\nu} + i\sum_q \overline{\psi^i_q}\gamma^u(D_\mu)_{ij}\psi^j_q \\ &- \sum_q m_q\overline{\psi^i_q}\psi_{qi}, \\ F^{(a)}_{\mu\nu} &= \partial_\mu A^a_\nu - \partial_\nu A^a_\mu - g_s f_{abc}A^b_\mu A^c_\nu, \\ (D_\mu)_{ij} &= \delta_{ij}\partial_\mu + ig_s\sum_a \frac{\lambda^a_{i,j}}{2}A^a_\mu, \end{aligned} \tag{1.51}$$

where $\psi^i_q(x)$ are the four-component Dirac spinors associated with each quark field of color i, $(i = 1, .., 3)$, and flavor q, $A^a_\mu(x)$ are gluon fields (of which there are eight), g_s is the QCD coupling constant, and f_{abc} are the structure constants of the SU(3) algebra.

Quarks have never been observed free. The confinement of quarks is related to the strength of the long distances interaction. The key feature of the theory of QCD is asymptotic freedom, (Gross and Wilczek, 1973; Politzer 1973), which is related to the weakness of the short distances processes.

1.3.1 Perturbative QCD and α_S

In perturbative QCD (pQCD), calculations are performed by expanding terms in a perturbation series in the coupling strength α_S. This is only valid when α_S is small (which happens at high Q^2, as shown in Figure 1.6 and discussed in Section 1.3.1.2). The calculation of a scattering cross section in perturbative QCD reduces to summing over the amplitudes of all possible intermediate states. Each graph is a symbolic representation for a term in the perturbative calculation. Leading Order (LO) term corresponds to the Quark Parton Model and is considered to be of order α_S^0 in the perturbative expansion. LO Feynman diagrams have no gluon vertices as shown in Figure 1.4 (A). Next-to-Leading Order (NLO) diagrams add quark-gluon interactions to this pictures. NLO graphs have one gluon vertex and correspond to terms of order α_S^1 in the perturbative expansion. NLO Feynman diagrams for hard scattering are illustrated in Figure 1.4 (B). Following this idea, higher order terms (NNLO, NNNLO, etc) would correspond to addition of more gluon vertices (two, three, etc).



(A)

(B)

Figure 1.4: (A) Feynman diagrams for LO DIS process, and (B) NLO DIS process with gluon or quark initiated processes.

1.3.1.1 Renormalization

The four-momentum is conserved at each vertex. However, including higher order diagrams (i.e. loop diagrams), the momentum circulating in the loop is not constrained. The integration over all momentum space for a loop diagram leads to logarithmic divergences when momentum goes to infinity, as

$$\int d^4k \frac{1}{k^2[(k+p)^2-m^2]} \sim \int d\Omega_3 \int dk k^3 \frac{1}{k^2 \cdot k^2} \sim \int \frac{dk}{k}. \tag{1.52}$$

This type of divergences, called ultraviolet (UV) divergences, is removed by a procedure called “renormalization”. This procedure introduces an arbitrary parameter μ with dimension of mass, representing the point at which the subtraction of the UV divergence is performed.

1.3.1.2 Running of α_S

Any physical observable $\mathcal{R}$ must be independent of the choice for μ, therefore we impose the following condition:

$$\mu^2 \frac{dR}{d\mu^2} = \left(\mu^2 \frac{\partial}{\partial \mu^2} + \mu^2 \frac{\partial \alpha_s}{\partial \mu^2} \frac{\partial}{\partial \alpha_S} \right) = 0. \tag{1.53}$$

Hence, any explicit dependence of the physical observable on the renormalization scale must be canceled by a proper renormalization scale dependence of α_S. The strong coupling α_S is determined by renormalization group equation given by

$$Q^2 \frac{\partial \alpha_S}{\partial Q^2} = \beta(\alpha_S) = -\alpha_S \sum_{n=0}^{\infty} \beta_n \left(\frac{\alpha_S}{4\pi} \right)^{(n+1)} = -\frac{\beta_0}{4\pi} \alpha_S^2(Q^2) - \frac{\beta_1}{16\pi^2} \alpha_S^3 + O(\alpha_S^4), \tag{1.54}$$

where the coefficients β_0 and β_1 depend on the number of active quark flavors n_f and scale Q^2 as

$$\beta_0 = 11 - \frac{2}{3} n_f, \tag{1.55}$$

$$\beta_1 = 102 - \frac{38}{3} n_f. \tag{1.56}$$

Expansion of the β-function is carried out to two loops, which corresponds to a NLO analysis.

The solution to renormalization group Eq. (1.54) provides the scale dependence of the strong coupling α_S, i.e. the "running" of α_S. Perturbative QCD predicts the scale dependence of the strong coupling, but α_S at a specific scale is obtained from experiment. Therefore, α_S at a reference scale is a fundamental parameter of the theory of QCD. Measuring the strong coupling from various experiments at different characteristic energy scales is an important test of QCD.

There are different physics processes from where α_S can be determined, such as DIS, e^-e^+ annihilations, hadron collisions, or quarkonia decays. The corresponding Feynman graphs for these processes are shown in Figure 1.5. Figure 1.6 shows the experimental measurements of α_S over a large range in Q^2. The decrease in $\alpha_S(Q^2)$ with increasing Q (i.e. "running of α_S") is clearly observed. The standard convention of the reference scale is chosen to be the Z mass. The value of α_S at different scales can be determined from the following relation:

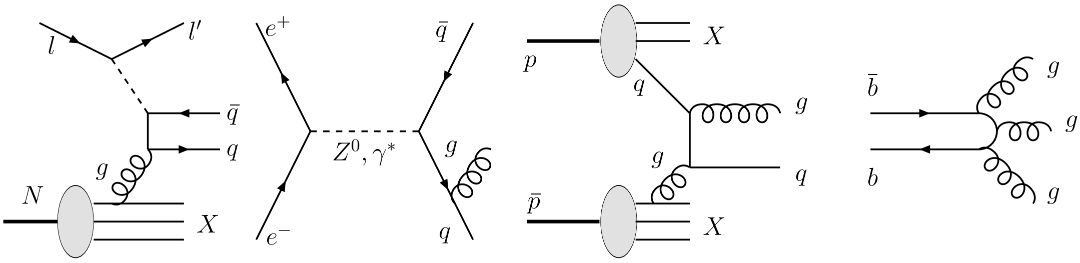


Figure 1.5: Feynman diagrams for processes used to measure α_S. The first graph corresponds to DIS process, the second to e^+e^- annihilation, the third to hadron collisions, and last one to quarkonia decays

$$\log \frac{Q^2}{M_Z^2} = \int_{\alpha_S(M_Z)}^{\alpha_S(Q)} \frac{d\alpha_S}{\beta(\alpha_S)}, \tag{1.57}$$

where M_Z is the mass of the Z boson.

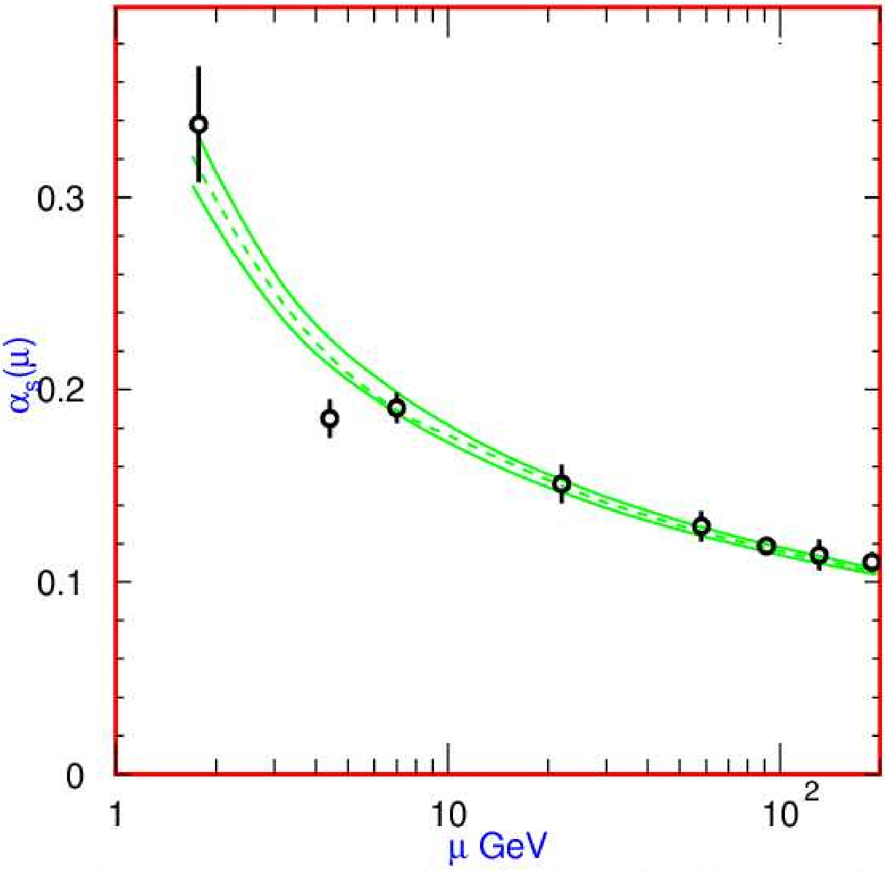


Figure 1.6: Running of the strong coupling α_S with scale $\mu \equiv Q$. The lines show the world average and the $\pm 1\sigma$. The measurements correspond, in increasing order of μ, to: τ width, Y decays, DIS, e^+e^- event shapes at 22 and 58 GeV, Z^0 width and e^+e^- event shapes shapes at 135 and 189 GeV (taken from PDG 2005).

1.3.1.3 Λ_{QCD} parameter

Alternatively, one can introduce a dimensional parameter, Λ_{QCD}, to provide a parametrization of the scale dependence of the strong coupling α_S. Following the convention from Ref.[4], Λ_{QCD} is defined by writing the solution of the renormalization group equation at a NLO as:

$$\alpha_S(Q^2) = \frac{4\pi}{\beta_0 \ln \frac{Q^2}{\Lambda^2_{QCD}}} \left[1 - \frac{\beta_1 \ln \left(\ln \frac{Q^2}{\Lambda^2_{QCD}} \right)}{\beta_0^2 \ln \frac{Q^2}{\Lambda^2_{QCD}}} \right]. \tag{1.58}$$

Λ_{QCD} represents the scale at which perturbative QCD becomes strongly coupled, i.e. the scale for which the coupling α_S is large and the perturbative QCD theory breaks down. This scale is comparable with the masses of the light hadrons (≈ 0.5 GeV). In other words, Λ_{QCD} determines the boundary between quasi-free state of interacting quarks and gluons (weak coupling) and the state where hadrons are formed (strong coupling). In this analysis we determine Λ_{QCD} from the NLO fits to the scaling violations of the nucleon structure functions, which is then related to the strong coupling α_S through Eq. (1.58).

1.3.1.4 Quark Thresholds

When quark mass effects are taken into account, n_f becomes a discontinuous function of the scale: below the charm threshold, three active flavor quarks are considered, while above, n_f depends on whether the scale is above the mass threshold for heavy quarks, i.e. $n_f = 4$ for scale between the charm and bottom threshold, $n_f = 5$ above the bottom quark threshold, etc.

The matching prescription is imposed by the condition that α_S must be a continuous function for all momenta, including the region of crossing a quark threshold. Thus, if Eq. (1.58) is to remain valid with α_S remaining continuous in Q^2, but n_f discontinuous in Q^2, then the effective value of Λ_{QCD} will depend on the number of active quark flavors. The matching conditions are given by

$$\alpha_S^{n_f=4}(m_b^2) = \alpha_S^{n_f=5}(m_b^2), \tag{1.59}$$

$$\alpha_S^{n_f=3}(m_c^2) = \alpha_S^{n_f=4}(m_c^2). \tag{1.60}$$

For the scale range probed by the fixed-target DIS experiments, it is most appropriate to quote Λ_{QCD} for four active quark flavors. One can make modifications to account for fewer or greater number of active flavors using Eqs. (1.58-1.56). Also, Λ_{QCD} depends on the renormalization scheme and on the order to which the calculation is carried. Therefore, when quoting a value for Λ_{QCD} one must specify the associated n_f, order and renormalization scheme explicitly.

1.3.1.5 Factorization

Perturbative calculations are performed in context of the factorization theorem which states that structure functions can be written in a factorized form with a soft (non-perturbative) and a hard (calculable, process-dependent) component [5],

$$F^{WN} = \sum_i f_N^a \otimes \omega^{Wa} + O\left(\frac{\Lambda^2}{Q^2}\right), \tag{1.61}$$

with the sum carried over all interacting quarks and gluons. The symbol $\otimes$ stands for the convolution between the parton distribution functions, f_N^a, and the hard scattering coefficients, ω^{Wa}. It is defined as

$$g \otimes h = \int_x^1 \frac{dy}{y} g(u) h\left(\frac{x}{y}\right). \tag{1.62}$$

Figure 1.7 illustrates a handbag diagram of the factorization theorem. The factorization scale, μ_F, defines the separation between soft (non-calculable) and hard (calculable using Feynman rules, ultraviolet dominated) physics. For processes with scales $\mu_F > Q$, the physics is non-perturbative QCD and contained in the PDF, f_N^a, while for $\mu_F < Q$ the physics is perturbative QCD (calculable) contained in the hard scattering coefficients ω^{Wa}.

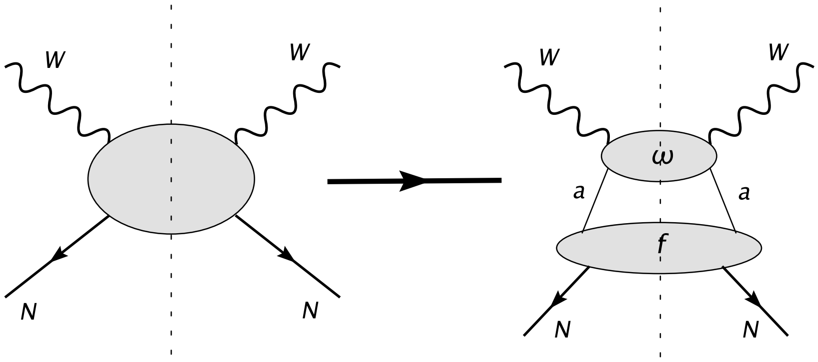


Figure 1.7: Illustration of the factorization theorem for hadronic structure functions. Process on the left $N(P) + W(q) \rightarrow X(Px)$ and the factorized process on the right $N(P) \rightarrow a(k_1)$ (represented by PDF, f_N^a) with the successive hard scattering interaction $a(k_1) + W(q)$ (represented by ω^a).

1.3.2 QCD Theory Models

There are various approaches of dividing the structure functions into calculable processes and PDFs. This analysis is presenting two NLO QCD models used in determination of α_S. The $\overline{MS}$ scheme provides a massless calculation of the structure functions and the $ACOT$ scheme fully accounts for quark masses. These models are described separately in the following subsections.

1.3.2.1 Massless $\overline{MS}$ Scheme

This is a model that deals with two scales: Λ_{QCD}, which describes the scale of a hadronic bound-state, and Q^2, the defining scale of large virtuality. This calculation is carried out under the assumption of massless quarks. The $\overline{MS}$ factorization scheme uses dimensional regularization to remove UV divergences [6]. The structure functions are obtained to NLO from the convolution of the hard scattering coefficients (process dependent) with the parametrized

PDFs (process independent). The quark distributions are conveniently divide into non-singlet, q^{NS}, and singlet, q^S, quark distributions:

$$\begin{aligned} q^{NS} &= \sum_i (q_i - \overline{q}_i) = u_v + d_v, & (1.63)\\ q^S &= \sum_i (q_i + \overline{q}_i). & (1.64) \end{aligned}$$

The structure functions at NLO can be written as follows:

$$F_3(x,Q^2) = \int_x^1 \frac{dy}{y}\left[C_3^q\left(\frac{x}{y},Q^2\right) q^{NS}(x,Q^2)\right], \tag{1.65}$$

$$\frac{F_2(x,Q^2)}{x} = \int_x^1 \frac{dy}{y}\left[C_2^q\left(\frac{x}{y},Q^2\right) q^S(x,Q^2) + C_2^g\left(\frac{x}{y},Q^2\right) G(x,Q^2)\right], \tag{1.66}$$

where the C_i are the coefficient functions which depend on the factorization and renormalization schemes. The $\overline{MS}$ coefficient functions are calculated at NLO in Ref. [7].

To leading order (α_S^0) the coefficients of quark terms are proportional to $\delta(1-\frac{x}{y})$, while the coefficients of gluon induced terms vanish. Therefore, the structure functions in the LO can be written simply as,

$$xF_3(x,Q^2) = xq^{NS}(x,Q^2), \tag{1.67}$$

$$F_2(x,Q^2) = xq^S(x,Q^2). \tag{1.68}$$

1.3.2.2 ACOT Scheme

In heavy quark production, the presence of the heavy quark mass, M_H, introduces a third scale. The heavy quarks will require a different organization of the perturbation series for an efficient separation of the short-distance and long-distance parts depending on the relative magnitude of M_H and the characteristic energy scale of the process. The factorization theorem has been extended by Collins [8] to incorporate quark masses as,

$$F = \sum_a \int \frac{dy}{y} f_N^a(x, \mu_F) \otimes \omega_{h/a} \left(\frac{x}{y}, \frac{Q}{\mu_F}, \frac{M_H}{\mu_F}, \alpha_S(\mu_R) \right) + O\left(\frac{\Lambda_{QCD}}{Q} \right), \tag{1.69}$$

where $\omega_{h/a}$ represents the coefficient functions and f_N^a is the parton density. The index a is over all quark flavors. There are different approaches to the treatment of heavy quark production. These include the Fixed Flavor Number (FFN) and Variable Flavor Number (VFN) schemes [9]. Both schemes are equivalent if calculations are carried out to all orders in α_S.

- **FFN scheme**

  Heavy quark contributions are explicitly included in the hard cross sections. This is intended for the factorization scale μ_F in the range of M_H (for $M_H \sim \mu_F$). In this case, heavy-quark production can be calculated from hard processes initiated by light quarks and gluons, and effects of the charm mass are included in the perturbative coefficient functions. This procedure gives problems when $\mu_F \gg M_H$ because the coefficient functions at higher order in α_S contain large logarithms that spoil the perturbative series [10].

- **VFN scheme**

  PDFs for heavy quarks are introduced and the number of active flavors changes by one unit when a heavy quark threshold is crossed. In other words, when $M_H \gg \mu_F$ this heavy quark should be excluded from the hadron (decoupling) and when $M_H \ll \mu_F$ it should be included. In this scheme the large logarithms which were encountered in the FFN scheme are re-summed via heavy quark PDFs, hence the convergence of the perturbative series is preserved [11].

The Aivazis-Collins-Olness-Tung ($ACOT$) scheme [12] belongs to the group of VFN factorization schemes that uses the renormalization method of Collins-Wilczek-Zee (CWZ) [8]. This scheme involves a mixture of the $\overline{MS}$ scheme for light partons (and for heavy partons if $\mu_F > M_H$) and the zero-momentum subtraction renormalization scheme for graphs with heavy quark lines (if $\mu_F < M_H$). The NLO $ACOT$ scheme is graphically presented in Figure 1.8. The first graph corresponds to the heavy quark excitation subprocess, $Wq \to q$ (order α_S^0 quark-scattering); the second to the heavy quark creation subprocess, $Wg \to qq$ (order α_S^1 gluon-fusion); and the third to the subtraction term.

The $ACOT$ scheme is designed to deal with finite quark masses, but neutrino structure functions have both light and heavy quark contributions,

$$F_i^{\nu}(x, Q^2) = F_{i,l}^{\nu}(x, Q^2) + F_{i,c}^{\nu}(x, Q^2), \tag{1.70}$$

where l stands for light quark flavor, and c for charm quark flavor. Therefore, this analysis uses a variant of the $ACOT$ scheme which still fully accounts for heavy quark production, but is numerically more stable when dealing with light quarks because the light quark masses are set to zero. The results in the limit $M_H \to 0$ agree with the massless $\overline{MS}$ case, as it should (see Appendix G.3.3 for more details).

1.3.3 Evolution and QCD Sum Rules

The Q^2 dependence of quark momentum densities is described by the coupled, integro-differential Dokshitzer-Gribov-Lipatov-Altarelli-Parisi (DGLAP) evolution equations [13]. For the non-singlet case the evolution doesn't depend on the gluon distribution, while the

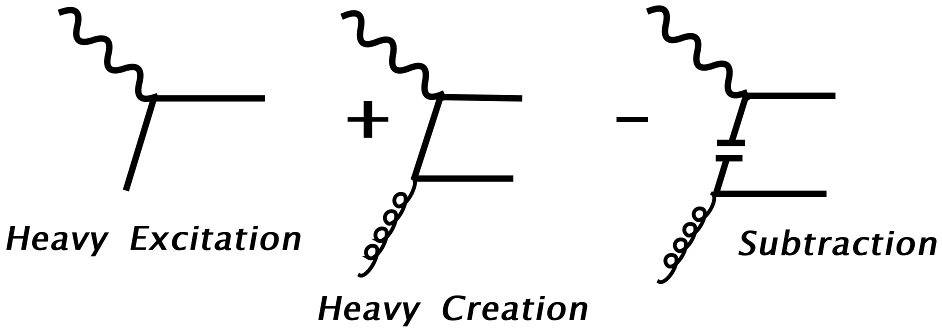


Figure 1.8: Schematic representation of *ACOT* scheme at NLO.

singlet distribution co-evolves with the gluon distribution, as can be seen from the following relations:

$$\frac{\partial q^{NS}}{\partial \ln Q^2} = \frac{\alpha_S(Q^2)}{2\pi} P_{qq} \otimes q^{NS}; \tag{1.71}$$

$$\frac{\partial}{\partial \ln Q^2}\begin{pmatrix} q^S \\ g \end{pmatrix} = \frac{\alpha_S(Q^2)}{2\pi}\begin{pmatrix} P_{qq} & 2n_f P_{qg} \\ P_{gq} & P_{gg} \end{pmatrix} \otimes \begin{pmatrix} q^S \\ g \end{pmatrix}, \tag{1.72}$$

where the symbol $\otimes$ denotes the convolution operator. These evolution equations can be expressed in terms of

$$t = \ln \frac{Q^2}{\Lambda^2}; \quad s = \ln \frac{\ln \frac{Q^2}{\Lambda^2}}{\ln \frac{Q_0^2}{\Lambda^2}} = \ln \frac{t}{t_0}; \tag{1.73}$$

as

$$\frac{dq^{NS}(x,s)}{ds} = t\frac{\alpha_S(t)}{2\pi}\int_x^1 \frac{dy}{y} q^{NS}(y,s) P_{qq}^{NS}\left(\frac{x}{y}\right), \tag{1.74}$$

$$\frac{dq^S(x,s)}{ds} = t\frac{\alpha_S(s)}{2\pi}\int_x^1 \frac{dy}{y}\left(q^S(y,s)P_{qq}^S\left(\frac{x}{y}\right) + G(y,s)P_{qG}\left(\frac{x}{y}\right)\right), \tag{1.75}$$

$$\frac{dG(x,s)}{ds} = t\frac{\alpha_S(s)}{2\pi}\int_x^1 \frac{dy}{y}\left(q^S(y,s)P_{Gq}^S\left(\frac{x}{y}\right) + G(y,s)P_{GG}\left(\frac{x}{y}\right)\right). \tag{1.76}$$

The splitting functions $P_{ij}(\frac{x}{y})$ give the probability of parton j with momentum fraction y be resolved as parton i with momentum fraction $x < y$. They are calculated perturbatively to a given order in α_S. LO and NLO splitting function diagrams are shown in Figs.1.9, 1.10. Evolution equations describe the physical picture in which valence quarks are surrounded by a cloud of virtual particles which are continuously emitted and absorbed. The quarks are also emitting and absorbing virtual particles of their own, corresponding in these equations to the branching probability densities. This picture explains why the structure of the hadron appears to change as it is seen at different distance scales. Thus, at low Q^2 there are fewer partons and their PDFs are skewed to high momentum fractions. At high Q^2, the momentum is shared through the parton branchings, and hence the low x region is filled with gluons and sea quarks which have a high probability to undergo $g \to gg$, $g \to q\bar{q}$ branchings. QCD doesn't predict the PDFs at any scale, rather it predicts how PDFs evolve with the scale through the evolution equations.

As a remark, in this analysis a massless evolution of the PDFs has been used regardless of the above selected theory model In fact, it has been shown in Ref. [14] that in a consistently formulated perturbative QCD as long as proper matching between the hard scattering and PDFs is observed, the final result is the same whether massless or massive evolution is used.

The PDFs are constrained through the following conservation laws, referred to as QCD sum rules:

- **Fermion sum rule:** In the simple parton model the fermion sum rule is the integral over the valence quark distributions with an expectation of 3 for proton(neutron),

$$\int_0^1 dx u_v(x) = 2; \quad \int_0^1 dx d_v(x) = 1. \tag{1.77}$$

  Since the number of bosons is not conserved, gluon distributions do no have such constraint.

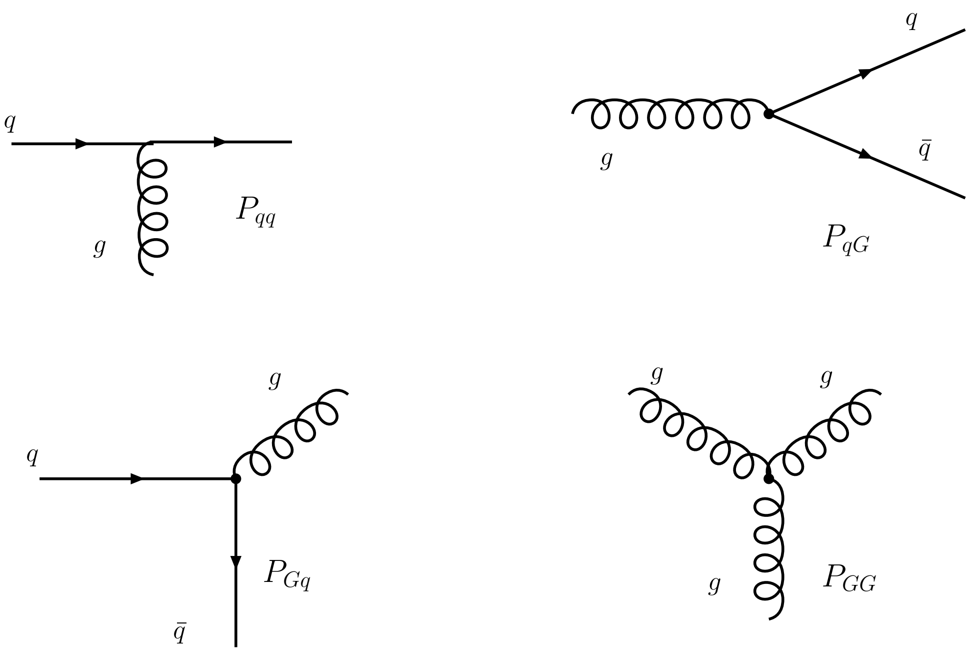


Figure 1.9: Leading Order splitting functions diagrams.

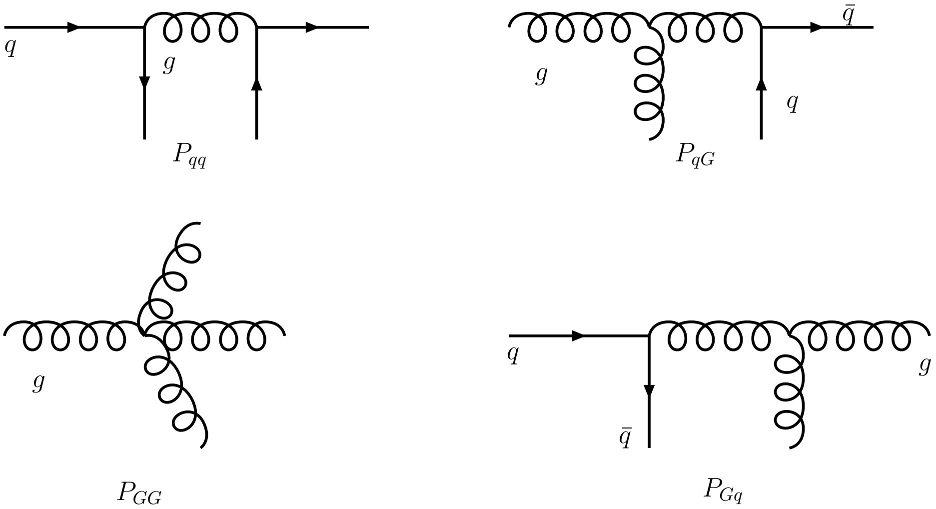


Figure 1.10: Examples of Next to Leading Order splitting functions diagrams.

- **Momentum sum rule:** The momentum sum rule states that the total momentum of the proton carried by quarks and gluons is 1,

$$\int_0^1 (xq_S + xG)dx = 1. \tag{1.78}$$

This rule is used to constrain the gluon normalization parameter.

1.3.4 Non-Perturbative QCD Effects

At low Q^2, the strong coupling constant becomes large and the perturbative calculations fail. In this non-perturbative region the assumption of scattering from massless, point-like, quarks is no longer valid. Also, the resolving power is not large enough to probe a single quark scattering. To obtain high interaction rates, neutrino experiments use heavy targets. Therefore, nuclear effects must be considered as well. These non-perturbative effects are discussed in the following subsections.

1.3.4.1 Target Mass

At low Q^2 and high x, one can not neglect the effects of the target mass. The meaning of x as the fraction of the nucleon's momentum carried by the struck quark is not suitable when $Q^2 \approx M^2$. A “target mass” correction must be applied to account for these effects.

The target mass correction (TMC) to the structure functions have been first determined by Georgi and Politzer in 1976 [15]. More recently, new derivations were performed [16] which lead to the following relations:

$$\begin{aligned}
F_1^{TM}(x,Q^2) &= \frac{x}{\xi}\frac{F_1^{(0)}(\xi,Q^2)}{k} + \frac{M_p^2x^2}{Q^2k^2}\int_\xi^1 \frac{F_2^{(0)}(u,Q^2)}{u^2}du \\
&+ \frac{2M_p^4x^3}{Q^4k^3}\int_\xi^1 du\int_u^1 \frac{F_2^{(0)}(v,Q^2)}{v^2}dv; && (1.79) \\
F_2^{TM}(x,Q^2) &= \frac{x^2}{\xi^2}\frac{F_2^{(0)}(\xi,Q^2)}{k^3} + \frac{6M_p^2x^3}{Q^2k^4}\int_\xi^1 \frac{F_2^{(0)}(u,Q^2)}{u^2}du \\
&+ \frac{12M_p^4x^4}{Q^4k^5}\int_\xi^1 du\int_u^1 \frac{F_2^{(0)}(v,Q^2)}{v^2}dv; && (1.80) \\
F_3^{TM}(x,Q^2) &= \frac{x}{\xi}\frac{F_3^{(0)}(\xi,Q^2)}{k^2} + \frac{2M_p^2x^2}{Q^2k^3}\int_\xi^1 \frac{F_3^{(0)}(u,Q^2)}{u}du, && (1.81)
\end{aligned}$$

with ξ and k defined as

$$k = \sqrt{1+\frac{4x^2M_p^2}{Q^2}}, \quad \xi = \frac{2x}{1+k}. \tag{1.82}$$

Sometimes the longitudinal structure function, F_L is used instead of F_2 or F_1 structure functions. F_L is defined as the combination between the absorption of transversely (F_1) and longitudinally ($F_2 - 2xF_1$) polarized virtual bosons as follows:

$$F_L(x,Q^2) = \left(1+\frac{4M^2x^2}{Q^2}\right)F_2(x,Q^2) - 2xF_1(x,Q^2). \tag{1.83}$$

One can then compute the target mass corrected longitudinal structure function:

$$\begin{aligned}
F_L^{\mathrm{TMC}}(x,Q^2) &= k^2F_2^{\mathrm{TMC}}(x,Q^2) - 2xF_1^{\mathrm{TMC}}(x,Q^2) \\
&= \frac{x^2}{\xi^2k}[F_2^{(0)}(\xi) - 2\xi F_1^{(0)}(\xi)] + \frac{4M_p^2x^3}{Q^2k^2}\int_\xi^1 \frac{F_2^{(0)}(u,Q^2)}{u^2}du \\
&+ \frac{8M_p^4x^4}{Q^4k^3}\int_\xi^1 du\int_u^1 \frac{F_2^{(0)}(v,Q^2)}{v^2}dv; \\
&= \frac{x^2}{\xi^2k}F_L^{(0)}(\xi) + \frac{4M_p^2x^3}{Q^2k^2}\int_\xi^1 \frac{F_2^{(0)}(u,Q^2)}{u^2}du \\
&+ \frac{8M_p^4x^4}{Q^4k^3}\int_\xi^1 du\int_u^1 \frac{F_2^{(0)}(v,Q^2)}{v^2}dv. && (1.84)
\end{aligned}$$

Here are some clarifying remarks on these equations:

- The target mass corrected structure functions are functions of (x,Q^2) and not (ξ,Q^2).

- The relations are general, holding to all orders in α_S.
- Quark masses effects are taken into account in $F_i^{(0)}$.
- $F_L^{TMC}(x, Q^2)$ is non-zero and thus violates the Callan-Gross relation, as it should. The leading term ($F_L^{(0)}$) will be non-zero for finite quark masses, and the sub-leading terms that involve integrals are non-zero for finite hadron mass M_p.

The target mass effects are large at high x and low Q^2. Figure 1.11 shows the size of the the TMC for both $F_2(x, Q^2)$ and $F_3(x, Q^2)$ structure functions as function of x. The size of the corrections increases with decreasing scale, as shown in Figure 1.11 by various colors for $Q^2 = 4, 8, 16, 32, 64, 128$ GeV^2.

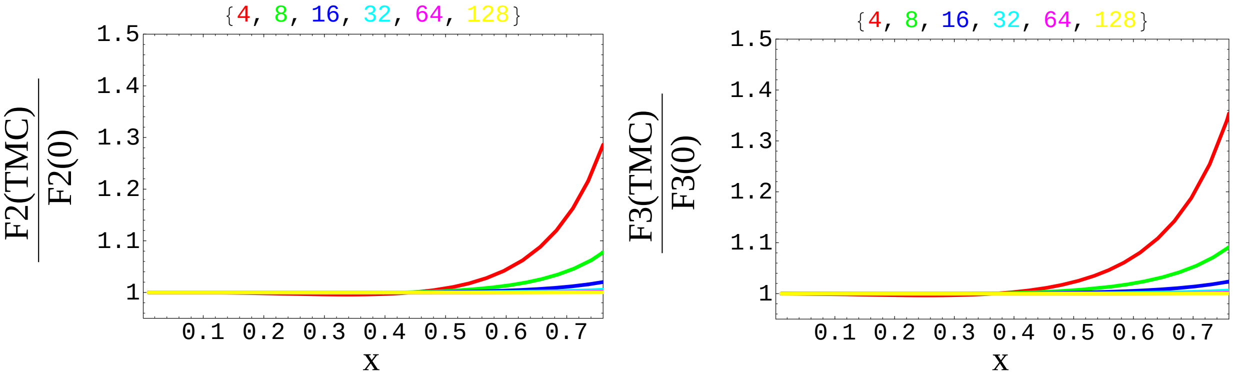


Figure 1.11: The size of the target mass correction for F_2 and F_3 $\left(\frac{TM_{ON}}{TM_{OFF}}\right)$. Red, green, blue, cyan, magenta, and yellow colors correspond to 4, 8, 16, 32, 64, and 128 GeV^2, respectively

1.3.4.2 Higher-Twist

The operators in pQCD are classified according to their twist, τ, defined as the difference between the free field dimension d and the spin j of the operator. The leading operators are these of order twist-2 which do not consider inter-quarks interactions. Twist $\tau > 2$ terms (higher-twists) are suppressed by increasing powers of $(1/Q^2)^n$ with respect to the leading twist. The higher-twist terms include interactions with other quarks, as shown in

Figure 1.12. Since target mass effects also involve powers of $1/Q^2$, they are referred to as "kinematic higher-twist" effects. The scattering involving a conglomerate of quarks processes are referred to as "dynamical higher-twist". These effects are important at low Q^2 and high x.

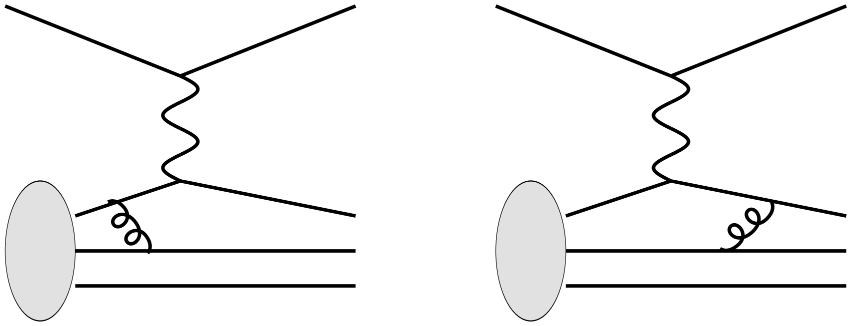


Figure 1.12: Examples of Higher-Twist QCD diagrams.

The higher-twist contributions are calculated using phenomenological approaches. However, our analysis uses data in the kinematic range where these effects are negligible. Therefore only kinematical higher-twist effects are studied (target mass effects). No additional higher-twist correction is applied.

1.3.4.3 Nuclear Target

It has been observed in the charged lepton scattering experiments (SLAC [17], NMC [18], EMC [19]) that the nuclear environment in which a struck quark is embedded modifies the x distribution of the quarks inside the nucleon. Figure 1.13 shows the size of nuclear target effects as function of x. Although limited data exists for nuclear effects on neutrino scattering, the standard correction for these effects is obtained from a fit to charged-lepton scattering on nuclear targets. This correction is large at low and high x, but is independent of Q^2. Hence, it does not affect the extraction of Λ_{QCD} from QCD fits to Q^2 dependence of structure functions.

There are new theoretical treatments that consider a Q^2 dependent nuclear target cor-

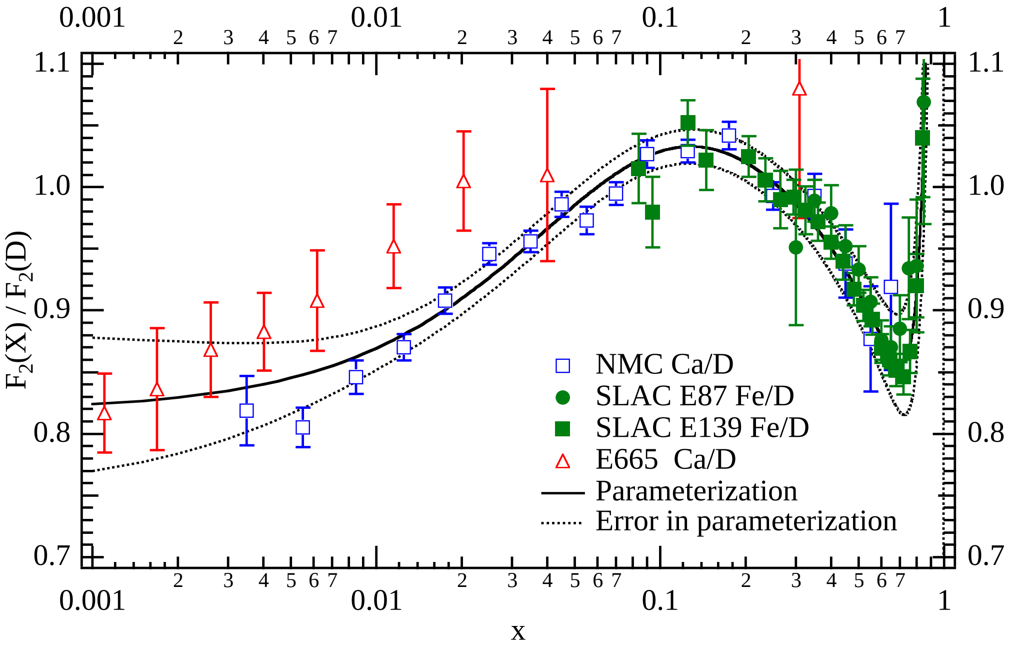


Figure 1.13: The size of nuclear effects as function of x for charged lepton data from Ref. [17, 18, 19]. Plot taken from Ref. [50].

rections at low x [20, 21]. Recent results from NuTeV hint that neutrino experiments might favor smaller nuclear effects than the charged lepton experiments [22] at high x.

This thesis does not take into account Q^2 dependent nuclear corrections. We can interpret our extracted PDFs from QCD fits as "effective nuclear PDFs", which have the nuclear effects absorbed into them.

2.0 NUTEV EXPERIMENT

In the previous Chapter, we outlined the theoretical background of deep inelastic neutrino scattering. In order to examine a neutrino interaction, it is necessary to provide a neutrino beam and a target with a detector to record the events. NuTeV is a precision neutrino-iron DIS experiment which collected separate high statistics neutrino and antineutrino events in 1996-97 at Fermilab. The unique design of NuTeV lead to the achievement of the following physics goals:

- A precise measurement of the electro-weak mixing angle [23];
- A search for neutral heavy leptons and other new processes[24, 27];
- A search for $\nu_\mu \rightarrow \nu_e$ oscillations [25];
- A measurement of the strange quark sea [26];
- A measurement of the neutrino-iron differential cross sections [22].

The final NuTeV result is a measurement of the strong coupling constant from QCD scaling violations of structure functions, which is the focus of this thesis. These results depend on a detailed understanding of the response of the target-calorimeter. We present in this Chapter the experimental setup of NuTeV including the detector and the neutrino beam. The calibration of the detector and data reconstruction are also presented in this Chapter.

2.1 NUTEV DETECTOR

Neutrinos interact only weakly. Hence, to increase the statistical precision of neutrino measurements a large dense target is often used. The NuTeV detector is composed of an iron target calorimeter followed immediately downstream by an iron toroidal muon spectrometer. This 1000 ton detector is illustrated in Figure 2.1. The two components are described in the next sections.

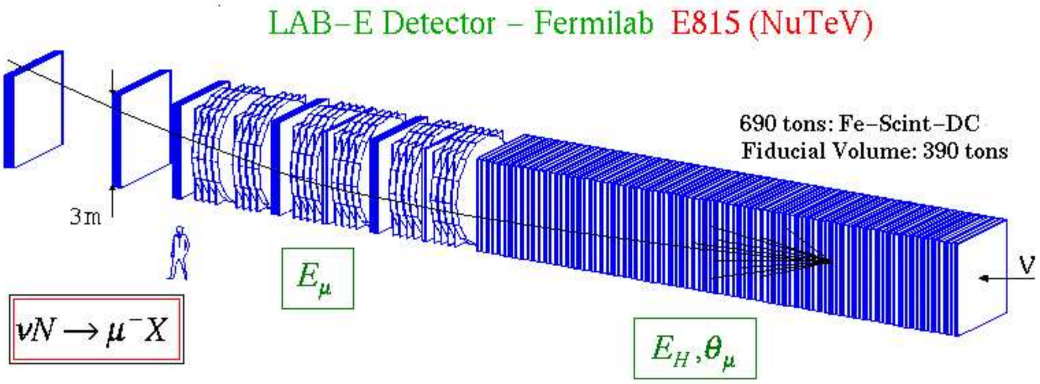


Figure 2.1: NuTeV detector

2.1.1 Target Calorimeter

The target calorimeter is 18 m long and weighs 690 ton. It is composed of a passive part of 168 plates of steel (3 m (H) × 3 m (W) × 5.1 cm (L)), and an active part of 84 scintillation counters (3 m (H) × 3 m (W) × 2.5 cm (L)) and 42 drift chambers. One calorimeter unit consists of two scintillation counters, each sandwiched between two steel plates, and a drift chamber placed every four steel plates, i.e. 10.35 cm of steel between scintillation counters and 20.7 cm of steel between drift chambers (see Figure 2.2). The composition of one unit of NuTeV target calorimeter in length, interaction length (λ_I) and radiation length (λ_{rad}) is given in Table 2.1 [28]. This unit is repeated 42 times to make up the whole calorimeter.

Table 2.1: Composition in radiation length and interaction length of one unit of the NuTeV calorimeter.

Component	Length (cm)	Radiation Length λ_{rad}	Interaction Length λ_I
4 Steel Plates	20.7	11.75	1.24
2 Scint. Counter	13.0	0.51	0.16
1 Drift Chamber	3.7	0.17	0.03
Total	37.4	12.43	1.43

Neutrino interactions take place in the steel plates (high Z material) producing hadron shower and muons sampled by scintillation counters and drift chambers (low Z material).

- **Scintillation counters:** are used to measure the energy deposited by charged particles in the calorimeter. The 84 liquid scintillation counters are numbered starting from the most upstream counter (84) to the most downstream counter (1). Each counter is instrumented with four photomultiplier tubes (PMT) located at each corner of the counter. A schematic view of a scintillation counter is shown in Figure 2.3. One scintillation counter is a 3 m×3 m and 2.5 cm thick Lucite box filled with 65 gallons of Bicron 517 L liquid scintillator oil. Lucite ribs do not scintillate, therefore they are spaced about 2.5-5 cm apart and staggered to avoid formation of dead zones. Two plastic water-filled bags are positioned on each side of the counter to balance the pressure of the oil.

  The mechanism of a scintillation counter is as follows: a charged current particle passing through the scintillator excites primary flours in the oil which emit photons as ultraviolet light. This light is absorbed by secondary flours in the oil, radiating visible blue light. The blue light is transmitted to the edges of the counter through total internal

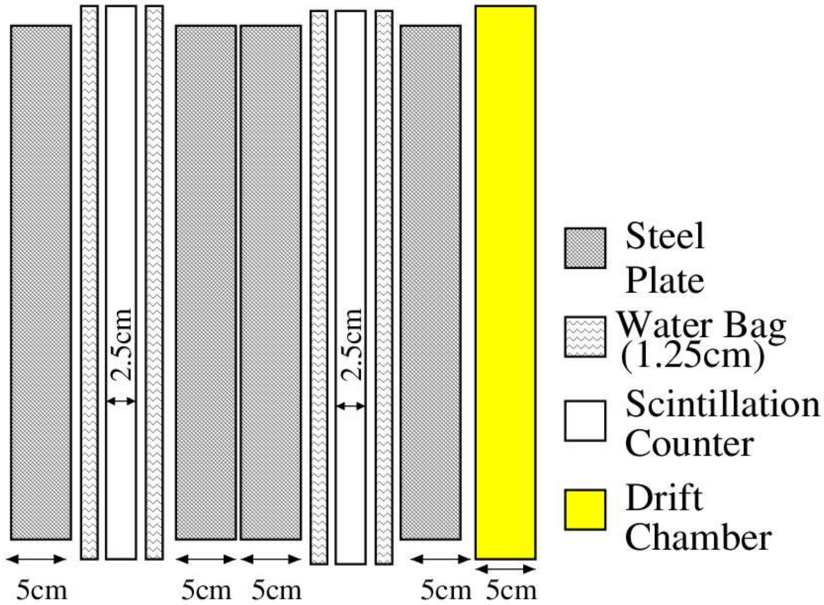


Figure 2.2: A sketch of one target unit consisted from steel plates, water bags, scintillation counters and one drift chamber

reflection where it is absorbed by eight half-inch thick wavelength-shifter bars (WLS) which shift blue light to green light of longer wavelength. This green light is piped to the phototubes mounted at the corner of each counter. The 10-stage Hamamatsu R2154 phototubes with green-extended photocathodes operate with a potential drop of 1400 V across the cathode to anode. The photocathode has about 20% probabibility of emitting an electron via the photoelectric effect. A single electron is amplified by a factor of 10^6 by a series of dynodes. A typical signal for a muon passing through the center of the counter is about 30 photoelectrons, and higher near the edge.

- **Drift chambers:** are used to track the transverse positions of the muons inside the calorimeter target. Each drift chamber consists of two orthogonally oriented planes with 24 horizontal and 24 vertical cells each filled with an equal mixture of argon and ethane

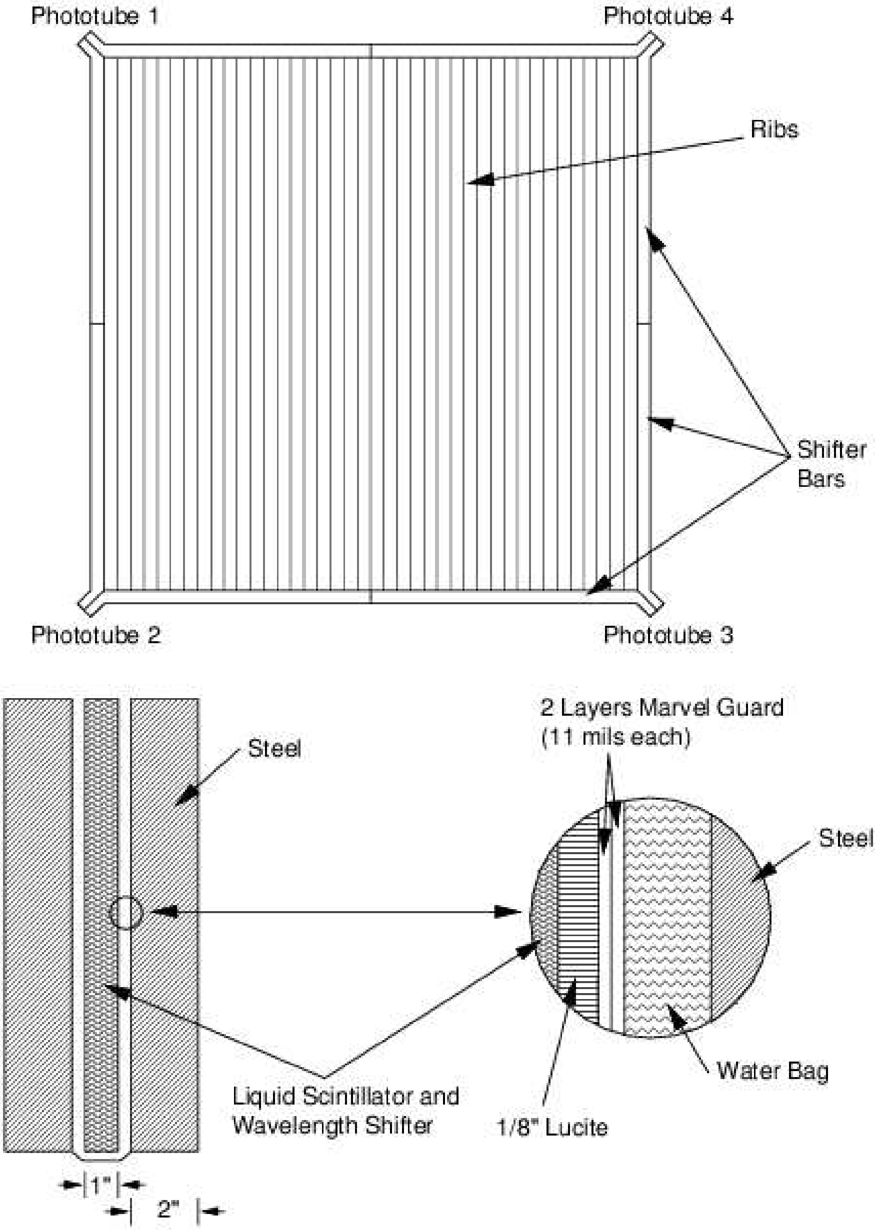


Figure 2.3: An illustration of a NuTeV scintillation counter.

gas. Each cell is 5 inches wide and contains three wires at center. The sketch of a drift chamber is shown in Figure 2.4. The drift chamber cells are defined by parallel aluminum I-beams which support G10 cover plates. For each cell the G10 cover is clad with 19 copper cathode strips set at a voltage to maintain a uniform electric fields of 690 V/cm. The central field wire within each cell is held at +350 V. On each side of the field wire there are two sense wires held at +1750 V. The I-beams operate at −4500 V. Hence, for a constant drift velocity of 52.4 μm/s the drift time for electrons to cross 2.5 in inside a cell is 1.2 μs.

A charged particle passing through a chamber cell ionizes gas molecules along its path. The electrons drift in an electric field toward the anode (sense) wire and accelerate producing further ionization. A signal pulse from collected electrons is registered on the sense wire. The particle's position in one dimension is determined by comparing the time of the electron in the uniform field (T^0) to the known time of the event (T). The position in the $X-Y$ plane of the charged particle is determined by combining hit information from two paired orthogonal cells as follows:

$$\begin{aligned} x &= x^0 + v_D(T - T^0_x), \\ y &= y^0 + v_D(T - T^0_y), \end{aligned} \tag{2.1}$$

where x^0, y^0 denote the locations of sense wires, and v_D is the drift velocity of free electrons.

2.1.2 Muon Spectrometer

The muon spectrometer is located immediately downstream of the target calorimeter. It consists of three large toroidal magnets. Each magnet is composed of eight steel washers with an outer radius of 1.8 m and an inner radius of 0.12 m. The toroids are magnetized by four copper coils which carry a DC current of 1250 A. Figure 2.5 shows a sketch of a cross sectional view of the toroid.

To track the passing muons, magnets are instrumented with five sets of single-wire drift chamber arrays. Three sets are located in the gaps between the toroids, and the remaining

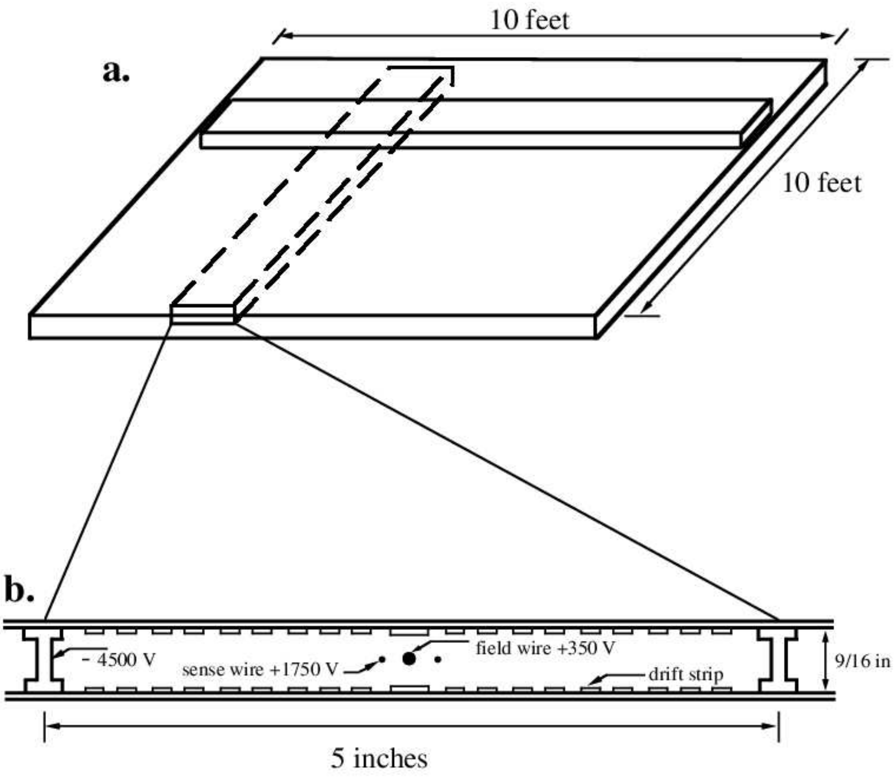


Figure 2.4: An illustration of a NuTeV drift chamber section. There are two orthogonally-oriented planes per station, consisting of three-wire cells.

two are downstream of the last toroid (blue carts). The blue carts are used to measure the high momentum muons. Figure 2.6 shows a side view of the muon spectrometer with toroidal magnets and drift chambers.

The toroid magnetic field varies between 1.9 T near the hole and 1.5 T near the edge. A muon passing through the whole length of the spectrometer receives a transverse momentum kick of 2.4 GeV. The polarity of the magnetic field is set to always focus on the primary muon, i.e. for neutrino mode μ^- are bent towards the center, for anti-neutrino mode μ^+ are focused. This enables NuTeV to better reconstruct muon momentum.

Early in the run, one of the coils of the second toroid (marked with asterisk in Figure 2.5)

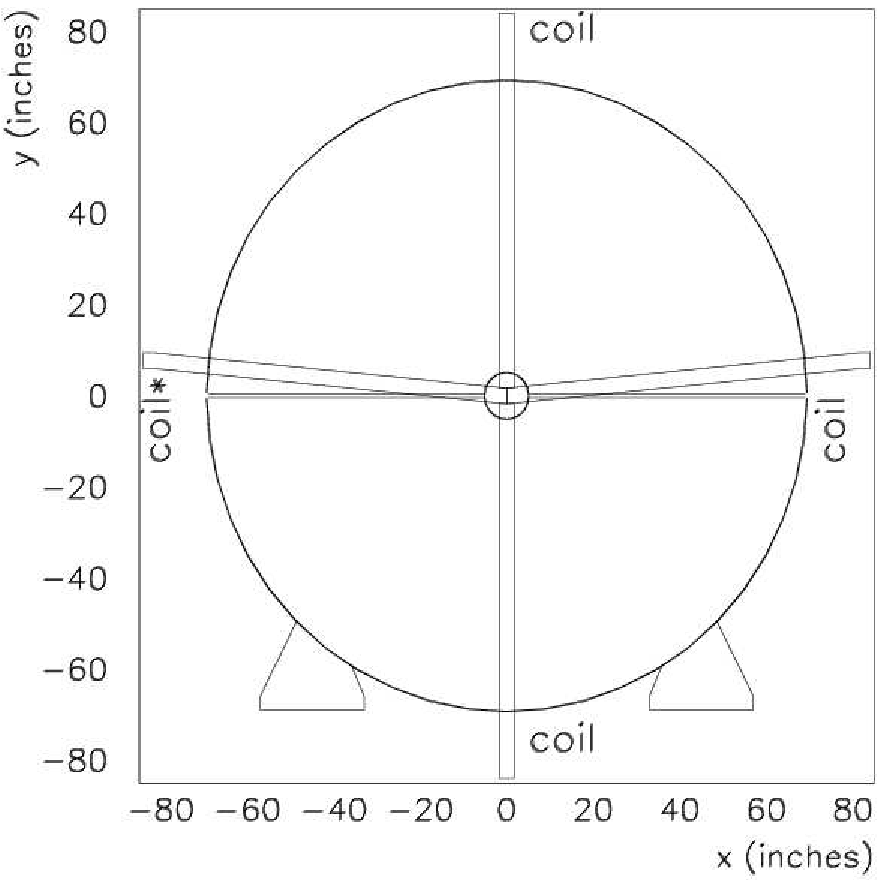


Figure 2.5: Cross section view of a NuTeV toroid magnet with four coils.

shorted to ground, and had to be disconnected. The effect of the burnt-out coil on the magnetic field of the toroid spectrometer is shown in Figure 2.7. This asymmetry in the magnetic field was included in NuTeV analysis software [30].

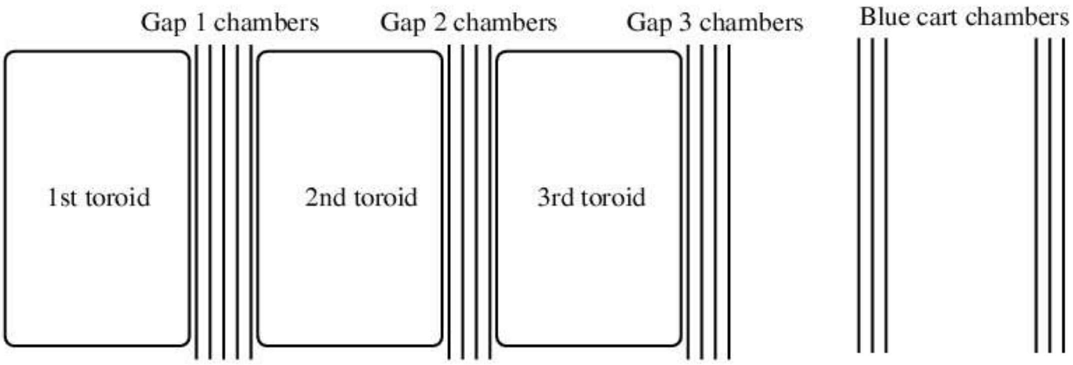


Figure 2.6: Side view of the NuTeV spectrometer with three toroid magnets and five sets of drift chambers.

2.2 NUTEV BEAMLINE

Neutrinos emerge from decay of the secondary pions and kaons produced in the interaction of 800 GeV protons with a low-Z BeO target. The layout of the neutrino beamline is sketched in Figure 2.8. High energy protons from the Tevatron were obtained through the following steps:

- A dense plasma of hydrogen interacts with a cesium cathode producing H^- ions.
- A Cockcroft-Walton electro-statically accelerates the ions to 750 KeV.
- Ions are injected into a 79 m long linear accelerator (Linac) reaching energies of 400 MeV.
- H^- are passed through a carbon foil which strips the electrons resulting in a proton beam which is injected at a rate of 15 Hz to the Booster.
- The Booster, a 70 m radius synchrotron, accelerates protons to 8 GeV.
- The proton beam is sent to the Main Ring, an 1 km radius synchrotron which accelerates protons to 150 GeV before being injected into the Tevatron.

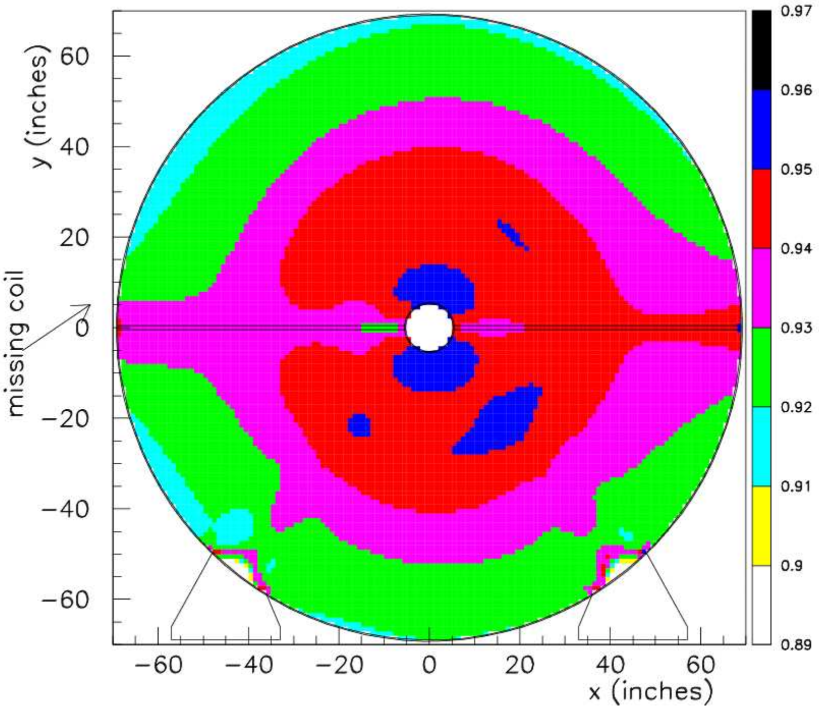


Figure 2.7: The effect of the burnt-out coil on the magnetic field of the toroid spectrometer from an ANSYS simulation. The ratio of 3 coil field to 4 coil field is shown. The burnt coil is on the left.

- Tevatron accelerates protons to 800 GeV. The beam is then directed to various experiments through a switch-yard.

The protons are extracted in one minute cycle of fast and slow spills. Figure 2.9 shows the structure of the beam timing. The fast spills consist of five 0.5 ms pings separated by 5 ms. The fast spills allow NuTeV to minimize the contamination from cosmic rays.

The fast pings are directed into a 33 cm long BeO target producing a secondary beam of mesons. This beam enters into NuTeV's Sign Selected Quadrupole Train (SSQT) [31]. The SSQT is one of two key features of the NuTeV experiment. It provides separate high purity ν and $\overline{\nu}$ beams through a series of magnets which directs the desired charge of secondaries

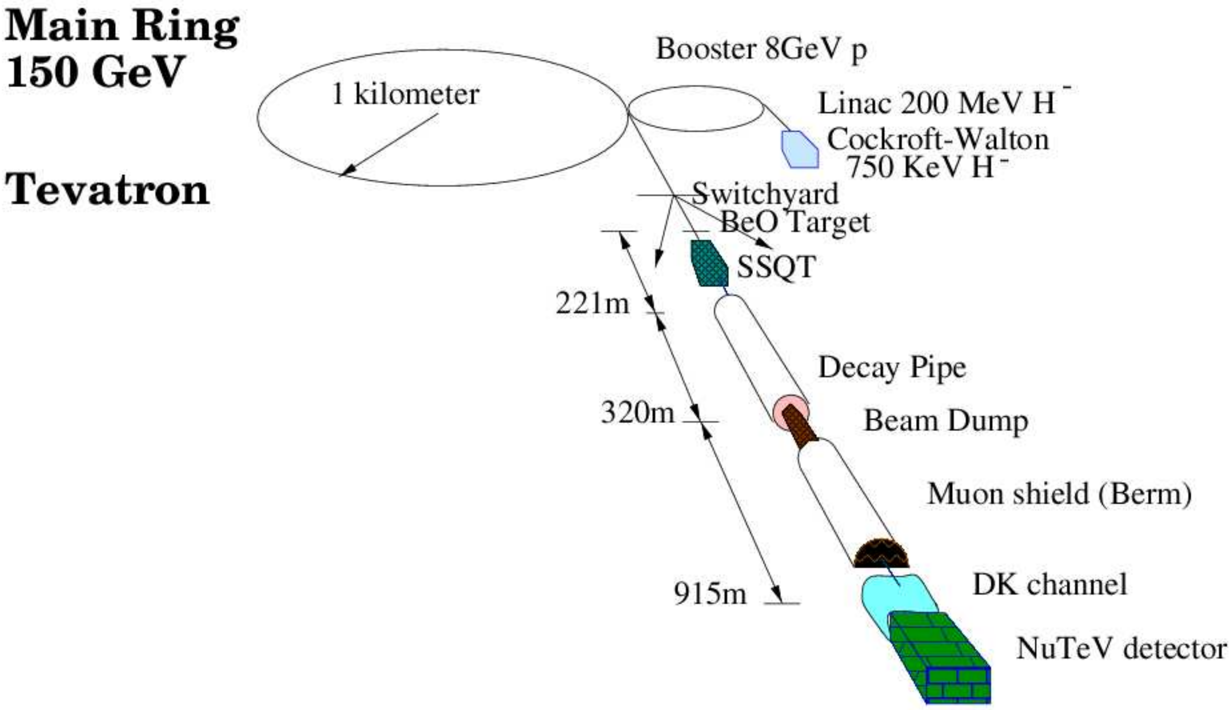


Figure 2.8: Schematic view of the Fermilab Neutrino beamline

("right-sign" mesons) into a decay space. The schematic view of the SSQT is shown in Figure 2.10. Undesired charged ("wrong-sign") particles are bent away to hit the beam dumps using the first dipole. Neutral particles go forward and are aimed away from NuTeV's downstream detector. Desired charged ("right-sign") mesons are directed at a 6 mrad angle towards the NuTeV detector. On their way, a few percent decay in-flight in a 300 m decay region as

$$\pi^{+(-)} \rightarrow \mu^{+(-)} + \nu_\mu(\overline{\nu}_\mu), \tag{2.2}$$

$$K^{+(-)} \rightarrow \mu^{+(-)} + \nu_\mu(\overline{\nu}_\mu), \tag{2.3}$$

producing muons and neutrinos (dominant processes which produce $\nu_\mu(\overline{\nu}_\mu)$s). Following the decay region is 900 m of earth and steel in which the un-wanted products of the decay (i.e.

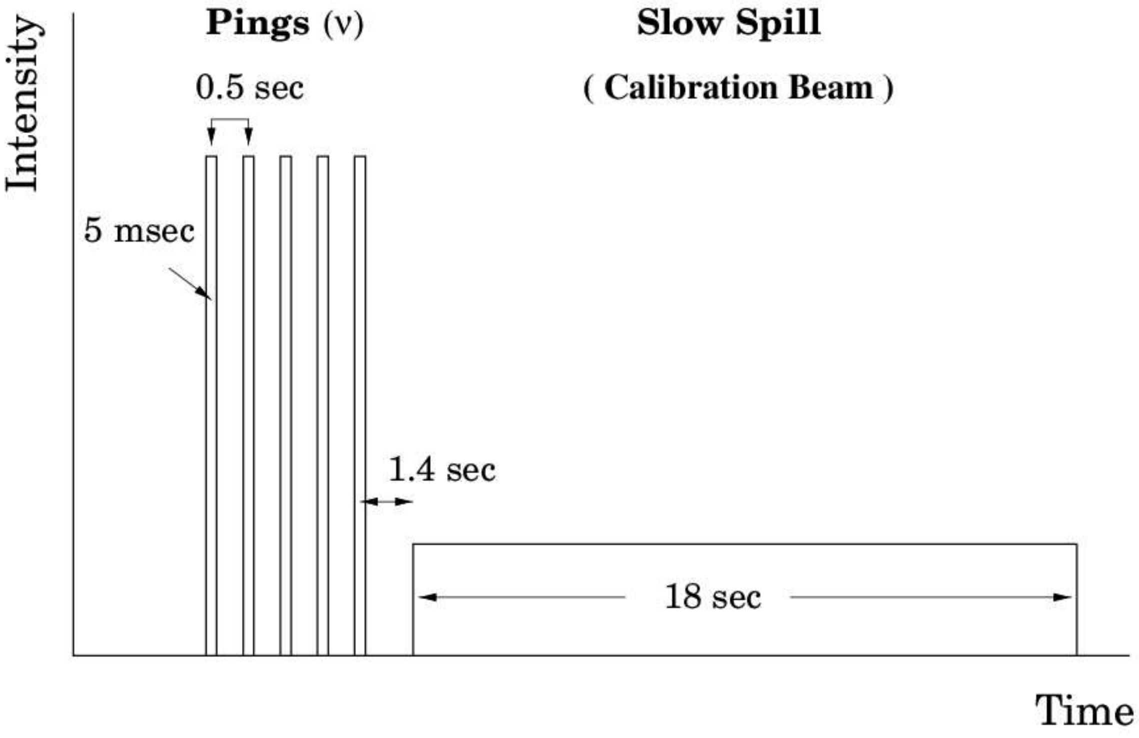


Figure 2.9: Structure of the beam timing. One cycle of 60.1 s is divided into fast and slow spills. Neutrinos are generated in five fast spills, the slow spill is dedicated to the calibration beam.

muons) are stopped. Therefore, NuTeV produces high purity neutrino or antineutrino beams. The contamination of antineutrinos in the neutrino beam (neutrino mode) is 0.03%, and neutrinos in the antineutrino beam is 0.4%. To reduce the electron-neutrino contamination of the final beam, the primary proton beam is not aimed in the direction of the NuTeV detector. Therefore, the dominant source of $\nu_e(\overline{\nu}_e)$s in the beam are from decay processes such as

$$K^{+(-)} \rightarrow \pi^0 + e^{+(-)} + \nu_e(\overline{\nu}_e). \tag{2.4}$$

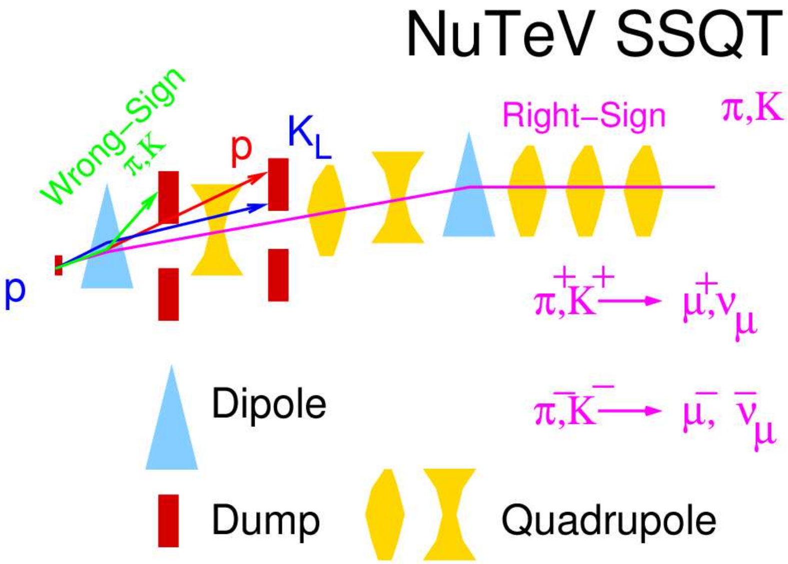


Figure 2.10: A sketch of NuTeV'Sign Selected Beam. A series of magnets select the desired charge of the secondaries.

The electron-neutrino contribution to the beam is 1.7% in neutrino mode and 1.6% in antineutrino mode. To predict various sources of neutrinos, a detailed beam Monte Carlo was used [23, 25].

Figure 2.11 shows flux spectra for neutrinos and antineutrinos in the NuTeV detector as function of neutrino energy. The first peak corresponds to neutrinos produced from pion decays, the second peak to neutrinos from kaon decays. The contaminations from wrong-sign and electron neutrinos also are shown.

2.3 DATA ACQUISITION AND EVENT TRIGGERING

A Data Acquisition System (DAQ) selects neutrino events, converts raw electronic signals into recordable data, and writes neutrino events to storage. These events are analyzed and

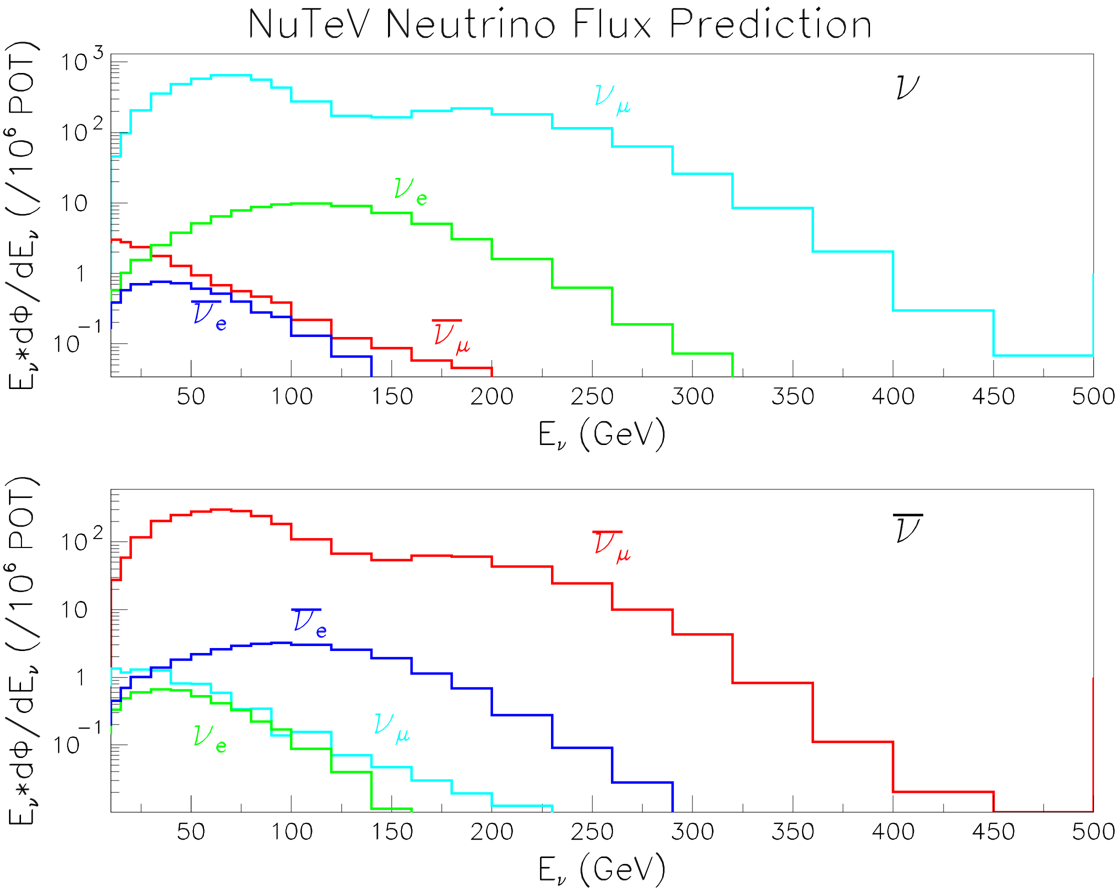


Figure 2.11: Flux prediction of neutrino beam for neutrino (top) and antineutrino (bottom) modes.

converted to physical quantities such as muon energy, muon angle, and hadronic shower energy.

2.3.1 Readout Electronics

Raw electronic signals from the phototubes of scintillation counters are in form of pulse heights (PH). This information is used in the logic of the trigger system. The phototube signal is digitized by a readout system of Analog-to-Digital Converters (ADC) into ADC counts. Scintillation counters are registering pulses from wide energy ranges of muons and hadron showers, i.e. a typical muon deposits about 0.2 GeV per counter. The readout system is divided into ADC channels of different sensitivities as:

- **LOW:** corresponds to un-amplified signal from a single phototube (84 counters × 4 PMTs LOW ADC counts);
- **COMBINATION LOW (COMBLO):** is the sum of un-amplified signals from all 4 PMTs in a counter (84 COMBLO ADC counts);
- **HIGH:** is the 10× amplified COMBLO signal (10 × COMBLO ADC counts);
- **SUPERLOW:** is the sum of attenuated signals from 8 PMTs from different counters each separated by 10 counters.

A typical muon passing through the center of the counter will produce about 2 LOW, or 8 COMBLO, or 80 HIGH, or 0.2 SUPERLOW ADC counts. Figure 2.12 shows the readout for a single counter. The signal from an energy deposition which saturates the HIGHs can be measured by LOWs. If a strong shower saturates the LOWs, then the SUPERLOWs are used. The signal from a muon is too small to be measured by LOWs, hence HIGHs are used. COMBLOs are used to form triggers.

The raw signals from phototubes are duplicated through a fan-out device. The signal is discriminated and its timing is recorded in a time-to-digital converter (TDC). The discriminated signal ("bits") are constructed as follows:

- **SBIT:** Amplifies the signal from COMBLOs by 100 times, then discriminates it with a 150 mV threshold. The SBITs are a single charged particle indicator from a single counter.
- **TBIT:** Amplifies the signal from COMBLOs by 100 times, then discriminates it with a 450 mV threshold. The TBITs is a multiple charged particle indicator from a counter (should not fire for a single muon).

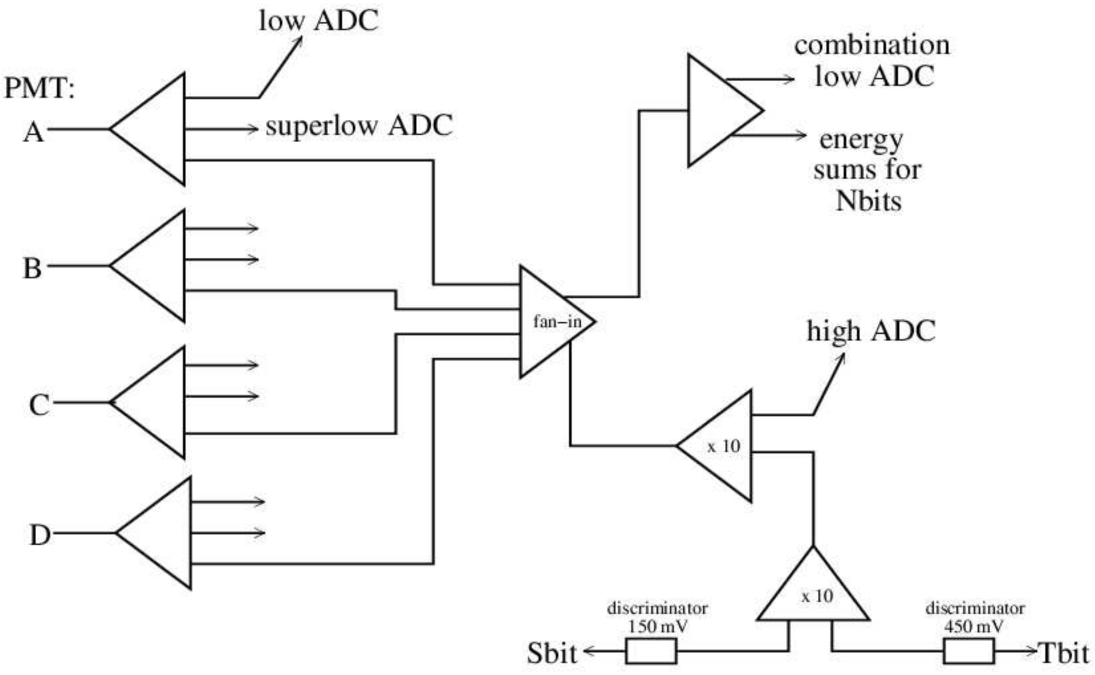


Figure 2.12: Configuration for the readout electronics of a single counter scintillation counter.

- **NBIT:** Sums un-amplified signals counters from COMBLOs of 8 consecutive counters, then discriminates it with a 55 mV threshold.

2.3.2 Event Triggers

Twelve neutrino event triggers are formed from different combination of the digitized signals. The triggers are designed to select a particular event signature in the detector and to distinguish neutrino events from background and noise.

- **TRIGGER 1** is designed to identify charged-current events which penetrate into toroid. The trigger is fired for one of these two scenarios:

  (A) There are hits at least in two of the last four counters and in two toroid gaps.

  (B) There are hits at least in two of the last four counters, in two upstream counters

(9-12), and in the first toroid gap only.

Trigger 1 was used in the selection criteria for the charged-current cross section analysis.

- **TRIGGER 2** is designed to identify neutral-current events which deposit only hadronic showers. The trigger is fired if 8 consecutive counters have at least 5 GeV.
- **TRIGGER 3** is used to select charge-current events in which a muon either stops in the target, or exits out the side. This trigger requires 16 counters to pass the SBIT threshold and at least 4 GeV total energy deposited in any 8 consecutive counters.
- **TRIGGER 4** is used to measure the TRIGGER 1 inefficiency.
- **TRIGGER 5** is used for calibration beam running. No detection requirements.
- **TRIGGER 6** is used to select the muons produced upstream of the detector. A hit in each target carts and hits in the same quadrant of each toroid gap are required. It is used for drift chambers alignment and counter calibration.
- **TRIGGER 8** is designed to select cosmic ray muons. It requires hits in at last 40 semi-consecutive counters and in first toroid gap.
- **TRIGGER 9** is designed to select possible neutral heavy lepton candidates which deposit energy only in the most upstream section of the detector..
- **TRIGGER 10** is used to measure the in-spill pedestal.
- **TRIGGER 11** defines events which shower in the toroid.
- **TRIGGER 12** is designed to monitor the detector with no activity.

A veto system, composed of a wall of scintillation counters, is situated upstream of the detector to remove charged particles produced upstream of the detector. The first four triggers require not to have a veto signal. Table 2.2 shows number of neutrino events which fulfilled each trigger requirement (an event can satisfy more than one trigger).

Table 2.2: Number of neutrino and antineutrino events for each trigger

Trigger	Name	ν events	$\overline{\nu}$ events
1	Charged-Current	2 574 783	914 886
2	Neutral-Current	5 445 024	2 804 981
3	Penetrating Muon	4 306 217	3 065 412
5	Test Beam	17 000 000	
6	Straight Through Muon	1 712 591	735 222
9	Neutral Heavy Lepton	1 334 675	1 616 637
10	Straight Through Muon	189 885	183 794
11	Toroid Pedestal	785 040	153 461

2.4 EVENT RECONSTRUCTION

The goal of the event reconstruction is to convert raw signal data into relevant physical quantities such as hadronic energy (E_{had}), muon energy (E_μ), and muon angle (θ_μ) at the vertex. ADC information is converted into pulse heights for each counter to determine the z-position of the interaction from the location of the first hit counter. The profile of energy deposition determines the length of the shower region which is used to find E_{had}. The track information is used to determine θ_μ. The muon's path in the toroid can be fitted to find the muon momentum. This is described in detail bellow.

2.4.1 Pulse Height

In a charged current neutrino interaction, a neutrino strikes a constituent quark inside a nucleon via a W boson. The struck quark undergoes strong interactions, and the nucleon is fragmented into a number of hadrons, mostly pions, but also kaons, protons, neutrons and other baryons. The energy of the hadronic shower is totally absorbed by the calorimeter, but only about 3% of it is detected by scintillation counters (the rest is deposited in the steel). High energy particles deposit the same amount of energy in the calorimeter counters through ionization, referred to as minimum-ionizing-energy (MIE). The energy deposition is measured in units of minimum-ionizing-particles (MIPs), i.e. the amount of energy measured in a single scintillator divided by the amount of MIE. The definition of one MIP used in NuTeV is the truncated mean energy deposited by a 77 GeV muon in one counter. The response of each counter is subject to variation in gain over time due to scintillation oil aging, PMT gain change and electronics response. The muon energy loss is well understood, therefore it is used as a calibration tool. It is described by the Landau distribution [32] characterized by a sharp peak due to the ionization losses of the muon, followed by a long tail due to the catastrophic energy losses such as e^+e^- pair production and bremsstrahlung (see Figure 2.13). NuTeV has an excellent calibration source to track counter gains: straight-through muons produced upstream of the NuTeV detector which share the same time gate with actual neutrino beam and produce a similar illumination of the detector (TRIGGER 6 muons). This sample is used to monitor the position and time dependencies of each counter gain.

The energy deposited in a single counter i, $E_{cntr}(i)$, registers electronic signals as the ADC pulse heights, $PH(i)$, which is modified by a number of factors as

$$E_{cntr}(\text{MIPs}) = \frac{h(i)G(i,t)PH(i)}{Map(i,V_x,V_y,t)}. \tag{2.5}$$

$Map(i,V_x,V_y,t)$, $G(i,t)$ and $h(i)$ are the counter map correction (V_x, V_y are the transverse vertex of the event), the counter gain and the relative hadron gain function, respectively, which are discussed below. The energy deposited by a charge particle in a single counter i is

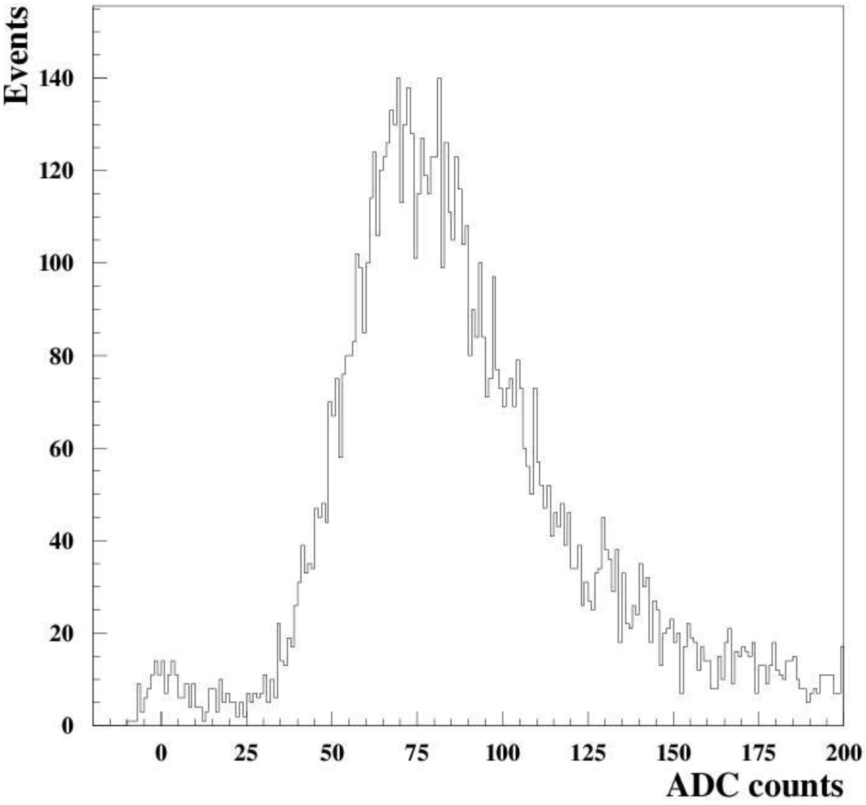


Figure 2.13: A typical energy deposition of muons traversing one scintillation counter in units of ADC counts. Plot taken from Reference [28].

measured from the pulse height, $PH(i)$, which is registered in either LOW or HIGH channels: if the number of ADC counts in the LOW channel is less than 35 the HIGH channel is used (muons); if one of the LOW channels is saturated (more than 1900 ADC counts), the SUPERLOW channel is used. For example, a muon passing through the center of the counter deposits about 30 photoelectrons, which corresponds to 80 ADC counts in the HIGH channel. The pedestals (detector response when no trigger was fired) are removed from the pulse heights.

The counter gain factor, $G(i,t)$, converts the pulse height response in ADC counts into MIPs. The counter response to muons is calculated from the truncated mean of the muon distributions [28]. This mean energy loss defines the conversion from pulse height (registered in high ADC counts) to MIPs for each counter.

The relative response of each counter varies in the geometry and over time. The relative hadron gain function, $h(i)$, takes these variations and non-uniformities into account (i.e. variations in steel thickness or water bags).

The response of each counter depends on the location of the traversing muon. The position dependence of the muon response for each counter is determined by dividing the counter into a grid and measure the energy response in each section of the grid, as shown in Figure 2.14. This Figure shows that a muon passing near the center of the counter produces fewer ADC counts than a muon passing near the corners where the PMTs are located. The pulse heights are corrected for these variations. The position-dependent correction $Map(i, V_x, V_y, t)$ is determined by the ratio of the counter response at the (x, y) position to the counter response at the center $(0, 0)$. The time variations in the counter are measured at the center of each counter where the event population is the highest.

2.4.2 Raw Event Variables

Figure 2.15 presents a typical event display of a charged current neutrino interaction in both views (X and Y). In the figure, a neutrino entering from the left interacts inside the target producing a hadronic shower and a muon detected by the target scintillators and drift chambers. The **RUN** and **EVENT** information are recorded in the header of each event, as well as which **TRIGGER** fired for that event. The red lines correspond to the pulse heights of the scintillation counters: the hadron shower is characterized by the spikes, and the muon by a nearly constant signal as it traverses the detector. The corrected pulse-height can be used to identify the interaction region along the z axis of the detector. The beginning of the interaction is marked by several large pulse heights that define the position of the interactions. The variable **PLACE** is defined as the most upstream counter of two

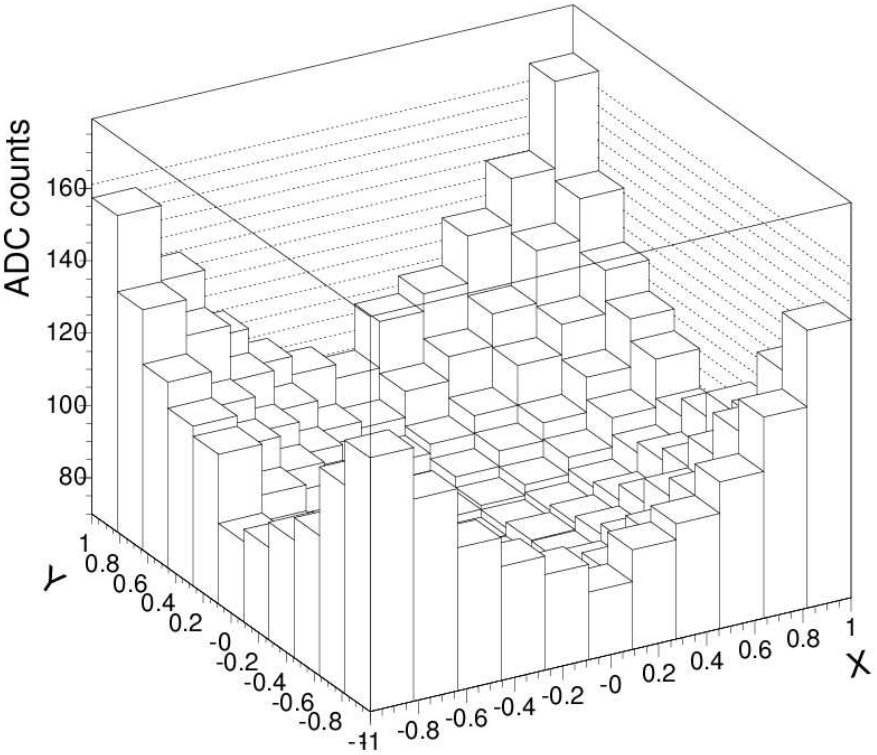


Figure 2.14: Average counter response to muons as a function of the position. The coordinates are normalized to the half width of the counter (1.5m) on both axes.

consecutive counters with at least 4 MIPs. The variable that defines the end of the shower is **SHEND**. It corresponds to the first of three consecutive counters with less than 4 MIPs. An illustration of the event length determination for charged current neutrino is shown in Figure 2.16. Using these definitions, the reconstruction of data can be separated into three regions which helps determine the physical quantities such as E_{had}, E_{μ}, and θ_{μ}:

1. Shower Region: extends between PLACE and SHEND-5. This is where hadronic energy is determined.
2. Target Muon Track: follows the shower region from SHEND-6 to counter 1. It is used to determine muon energy loss in the target.

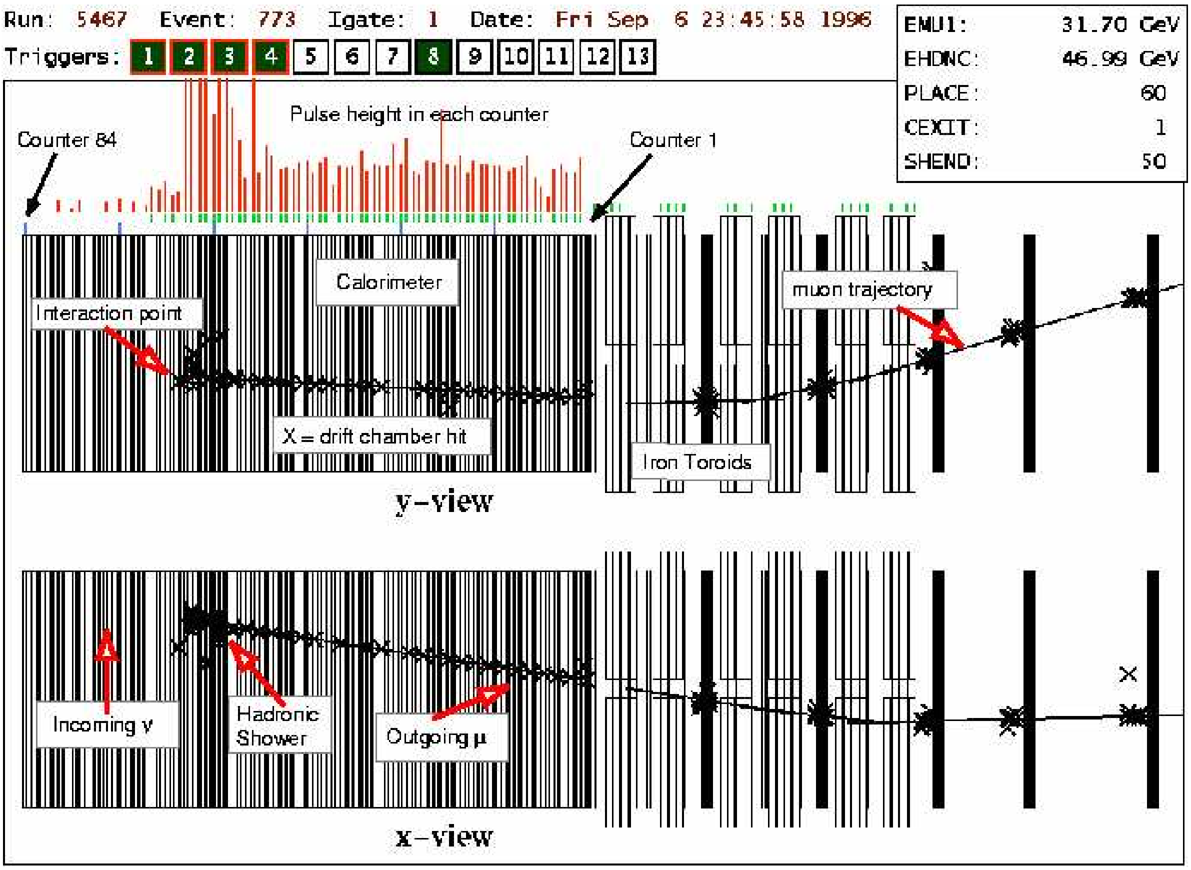


Figure 2.15: A ν_μ event display for a typical charged current neutrino interaction.

3. Toroid Spectrometer: muon energy is determined by fitting the muon track

Next, we present the reconstruction of the hadronic energy, muon energy, and muon angle.

- **Reconstruction of the hadronic energy:**

  Hadronic energy is determined by a simple sum of the counter pulse heights within the shower region with energy deposited by the primary muon subtracted,

$$E_{had}(\text{GeV}) = C_\pi \sum_{i=SHEND-5}^{PLACE} \left(E_{cntr}(i) - E_\mu^{loss}(i)\right), \tag{2.6}$$

  where C_π is the hadron calibration constant that converts MIPs to GeV and it is discussed in Sect. 2.5. Muon energy in the shower region is not directly measured. Instead, a

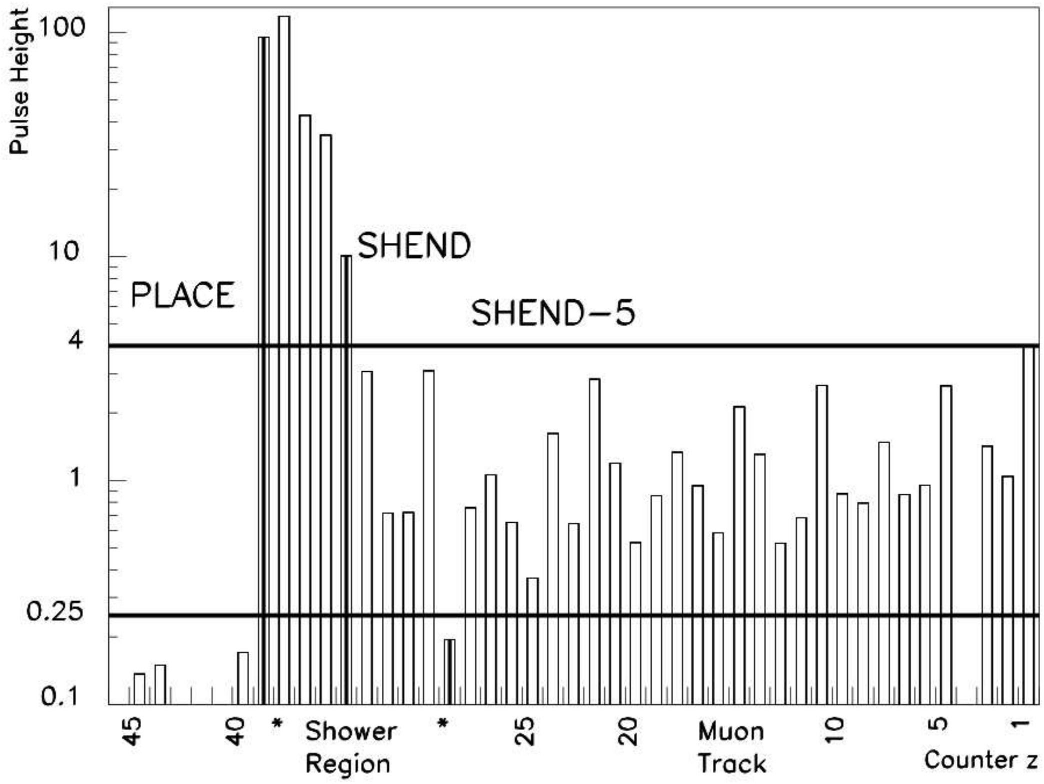


Figure 2.16: Pulse height of an event (in MIPs) versus counter number. Muon track and shower regions are shown.

parametrization of the average energy loss for a minimum ionizing particle (in MIPs) is used:

$$E_{\mu}^{loss}(i) = 0.9315 + 0.02359 \log E_{\mu}^{est}, \tag{2.7}$$

where E_{μ}^{est} is the estimated energy loss of the muon (in GeV).

- **Reconstruction of the muon angle:**

  The muon angle at the vertex is determined by fitting the muon track (obtained from the recorded muon hits in the drift chambers) near the vertex. Because no magnetic field is present in the target, the muon track in the target is straight. In the track fitting procedure, a line segment is chosen in the calorimeter to fit the hits in the most downstream drift chambers. This straight-line segment is extrapolated upstream to PLACE, but due to the hadron shower the hits immediately after PLACE are not used for tracking. The number of drift chambers nearest the vertex position that are omitted from the tracking procedure depends on the hadron energy: higher energy hadron showers traverse more drift chambers. The multiple Coulomb scattering (MCS) effects complicate the muon track reconstruction. A particle with momentum P_μ traversing a distance L in a material with radiation length λ_{rad} will suffer a mean Coulomb scattering deflection

$$\sigma_\theta^{MCS} = \frac{0.015}{P_\mu}\sqrt{\frac{L}{\lambda_{rad}}}. \tag{2.8}$$

Therefore the muon angle measurement is correlated with muon momentum, and the fitting procedure is iterated.

- **Reconstruction of the muon energy:**

  The energy of the muon is reconstructed in three parts: muon deposited within the hadron shower, E_μ^{shower}, muon deposited in the calorimeter beyond hadron shower, E_μ^{target} (i.e. dE/dx), and muon energy measured at the front face of the steel in the toroid spectrometer, E_μ^{FF}:

  $$E_\mu(\text{GeV}) = E_\mu^{shower} + E_\mu^{target} + E_\mu^{FF}. \tag{2.9}$$

  The muon energy loss in the shower region cannot be determined due to absence of position information in the shower region. Therefore, a parametrization of the most probable dE/dx loss per counter (in GeV) is used,

  $$E_\mu^{shower} = \sum_{i=SHEND-5}^{PLACE} (0.1668 + 0.00037 E_\mu^{est}(i))/\cos\theta_\mu, \tag{2.10}$$

where $E_\mu^{est}(i)$ is the estimated muon energy and θ_μ is the muon angle. Muon energy loss beyond the shower region is determined from the sum of corrected PHs in each counter beyond the shower region as,

$$E_\mu^{target}(\mathrm{GeV}) = \sum_{i=1}^{SHEND-6} E_\mu^{cntr}(i), \tag{2.11}$$

where

$$E_\mu^{cntr}(i) = \begin{cases} C_\mu(E_\mu) E_{cntr}(i), & \text{for small PHs} \\ C_\mu(E_\mu) + C_e \left(E_{cntr}(i) - 1\right), & \text{for large PHs} \end{cases} \tag{2.12}$$

For small pulse heights ($<$ 5 MIPs), the muon energy loss is assumed to be from ionization processes with a muon energy dependent conversion factor, $C_\mu(E_\mu)$. For high pulse heights ($>$ 5 MIPs), the muon energy loss is attributed to catastrophic processes (i.e. pair production, Bremsstrahlung) which produce an electromagnetic cascade, hence, requiring an electromagnetic calibration factor, C_e. The calibration factors, $C_\mu(E_\mu)$ and C_e, which convert MIPs into GeVs are discussed in Sect. 2.5.

The muon momentum at the front face of the toroid spectrometer is obtained from the bend of the muon track in the magnetic field of the toroid. In a constant magnetic field in vacuum, the bend angle is given by relation

$$\theta_{bend} = \frac{0.3 B L_\mu}{P_\mu}, \tag{2.13}$$

where P_μ is the muon momentum and L_μ is the length of the track in the magnetic field. The algorithm for muon momentum determination is as follows: a projected track is obtained from the measured muon angle in the target and by using a model of the magnetic field; the hits in the toroid chambers are fit to this projected track. The MCS effects and muon energy loss in the steel are included in the track fitting procedure.

2.5 CALIBRATION

The measurements from previous neutrino experiments were dominated by the experimental uncertainties due to energy scales (muon and hadronic) [33], [34]. There are two key features that enable NuTeV to be the most precise neutrino experiment to date. One is the SSQT which was presented in Sect.2.2, and the other feature is NuTeV's calibration beam which considerably improves its knowledge of the energy scales, presented next. Here we outline some characteristics of the NuTeV calibration beam, the reader is referred to Ref. [28] for more details on the calibration and response measurements.

2.5.1 In Situ Calibration Beam

The NuTeV detector was continuously calibrated with a charged particle beam. Calibration beam running was concurrent with neutrino data taking. The calibration data stream was separated from the neutrino data by 1.4 s within the same accelerator cycle. The slow spill in Figure 2.9 corresponds to calibration beam running, while the fast spill to neutrino running. Protons during the slow spill were passed through an aluminum block producing secondaries which were focused and selected for momentum through a series of collimators and dipoles. The selected particles (i.e. electrons, muons, and hadrons over a wide range of energy from 4.8 to 190 GeV) were then passed through a low mass spectrometer with a long lever arm. The schematic view of this spectrometer and calibration beam is shown in Figure 2.17. The spectrometer was instrumented with drift chambers, dipole magnets, a Čerenkov detector and an array of transition radiation detectors (TRDs) used for particle identification. The drift chambers are used to track the charged particles, the dipole magnets are adjusted so that the particles of the correct momentum are directed to NuTeV detector. The Čerenkov detector and the TRD are used to discriminate among electrons, pions, kaons and anti-protons when needed, otherwise they are rolled out of the beamline and replaced by

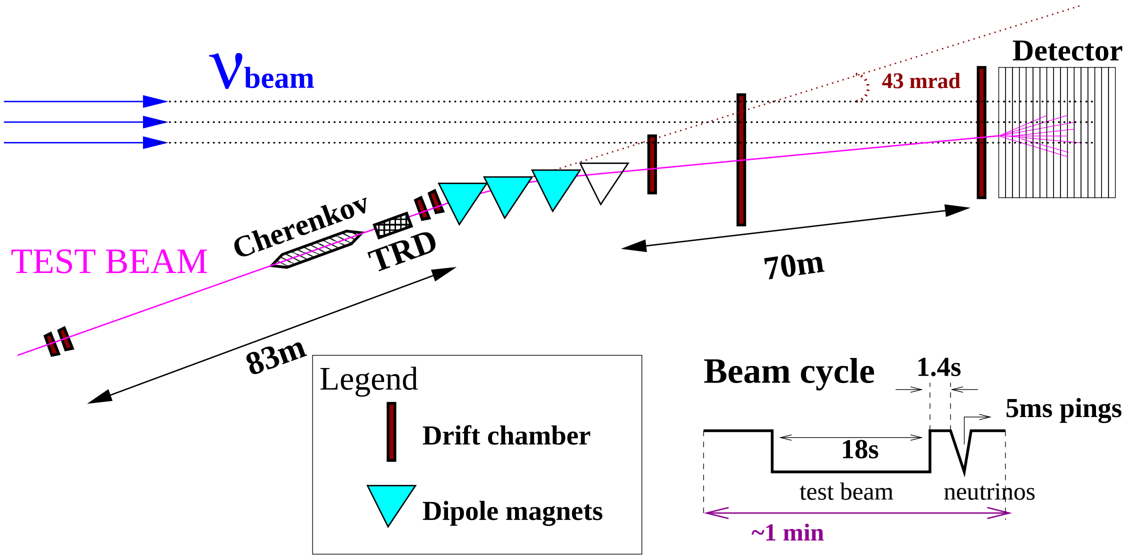


Figure 2.17: A schematic view of the NuTeV Calibration Beam directed at 43 mrad angle and the low mass spectrometer with a long lever arm.

a vacuum pipe to reduce the multiple scattering. Figure 2.18 shows a clean identification for pions, kaons and anti-protons within the 50 GeV hadron beam by the Čerenkov counter using N_2 gas at 160 torr pressure. The last dipole magnet of the calibration beam spectrometer could be steered to map the response of the over the front face of the detector.

The spectrometer measured energy with a resolution of 0.3% in an event-by-event basis. This was achieved by using precisely calibrated dipole magnets with $\int Bdl$ known to better than 0.1% in the region traversed by the beam (i.e. "good field" region). A calibration beam trigger, which consists of two scintillation paddles, was used to maintain the 0.1% tolerance of the $\int Bdl$. Also, the long distances of 83.3 m between the most upstream chamber and momentum analyzing magnets, and of 69.2 m between the most downstream chamber and magnets allowed a relative chamber alignment uncertainty of 1 mm. This can be translated into 0.1% uncertainty in the beam momentum.

The absolute hadron energy scale of the calorimeter is determined by measuring its

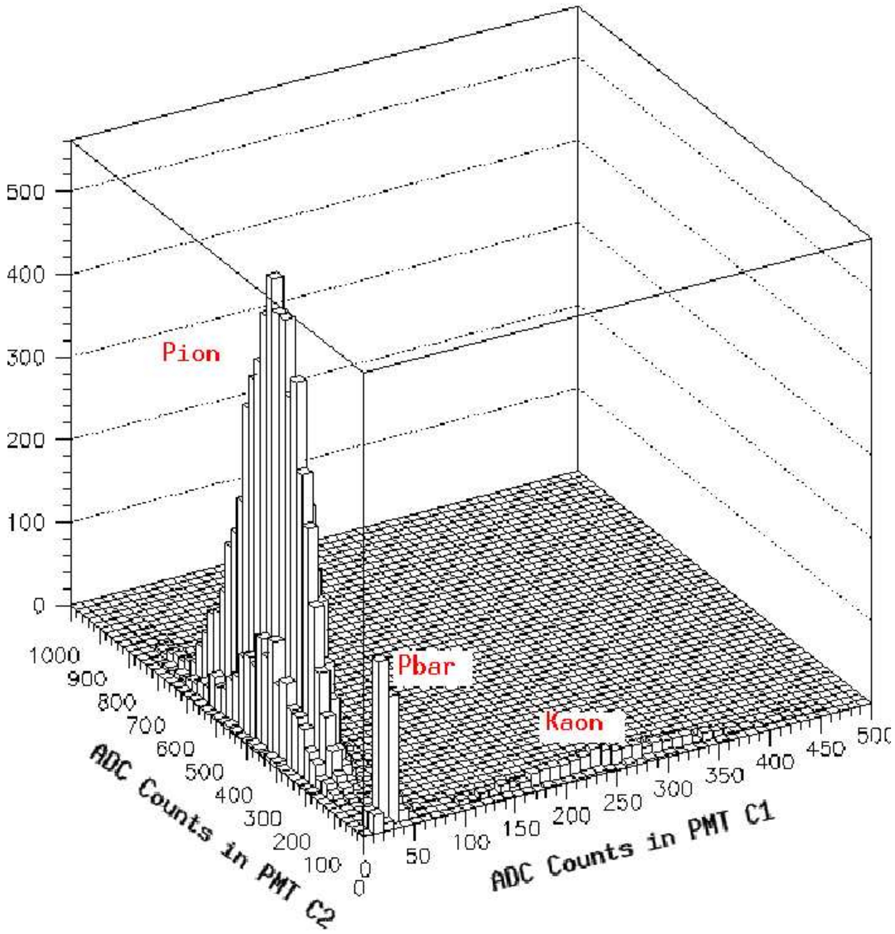


Figure 2.18: Particle identification among π, K, $\overline{p}$ at 50 GeV hadron beam shown by PMT signals (C1 and C2, placed on opposite directions) of the Čerenkov counter with 160 Torr N_2 gas. Plot taken from Ref. [28].

response to single pion interactions. The calibration beam composition of hadrons has been studied in details. Table 2.3 summarizes particle composition in the hadron calibration beam for various energies. Electrons were removed from the low energy samples through cuts based on Čerenkov particle ID system. The response of the calibration beam was corrected for the contamination of pions from kaons and antiprotons using clean samples of each particle type

Table 2.3: Particle composition in hadron calibration beam for various energies.

P (GeV)	Electron fraction (%)	Hadron ($\pi^-, K^-, \overline{p}$) fraction (%)		correction to E_{had}%
5	92	8		-0.6 ± 0.2
15	66	34	(π^- : 95.6; $\overline{p} + K^-$: 4.1)	-0.22
20	< 1	>99	(π^- : 95.5; $\overline{p} + K^-$: 4.5)	-0.15
30	< 0.25	>99.75	(π^- : 94.9; $\overline{p} + K^-$: 5.1)	-0.14
50	0	>100	(π^- : 93.9; $\overline{p}$: 3.0; K^- : 3.1)	-0.017
75	0	>100	(π^- : 91.7; $\overline{p}$: 3.2; K^- : 5.1)	-0.010
120	0	>100	(π^- : 91; $\overline{p}$: 2.8; K^- : 6.2)	-0.002

given by the Čerenkov counter information. It has been found that the showers from kaons agree with those from pions, while those from antiprotons give a higher response due to the $p\bar{p}$ annihilation processes. The correction factors for these effects on hadronic energy are estimated and are presented in Table 2.3 (see the last column). This effect is negligible for energies greater than 50 GeV (i.e. has less than 0.03% effect).

2.5.2 Calorimeter Calibration

The reconstructed energy by calorimeter is proportional to the number of shower particles produced by the incident hadron. Therefore, the resolution function of the calorimeter for hadrons can be characterized by a Poisson distribution. The electromagnetic component of the shower causes the response to look Poisson-like at low energies and Gaussian at high energies as shown in Figure 2.19.

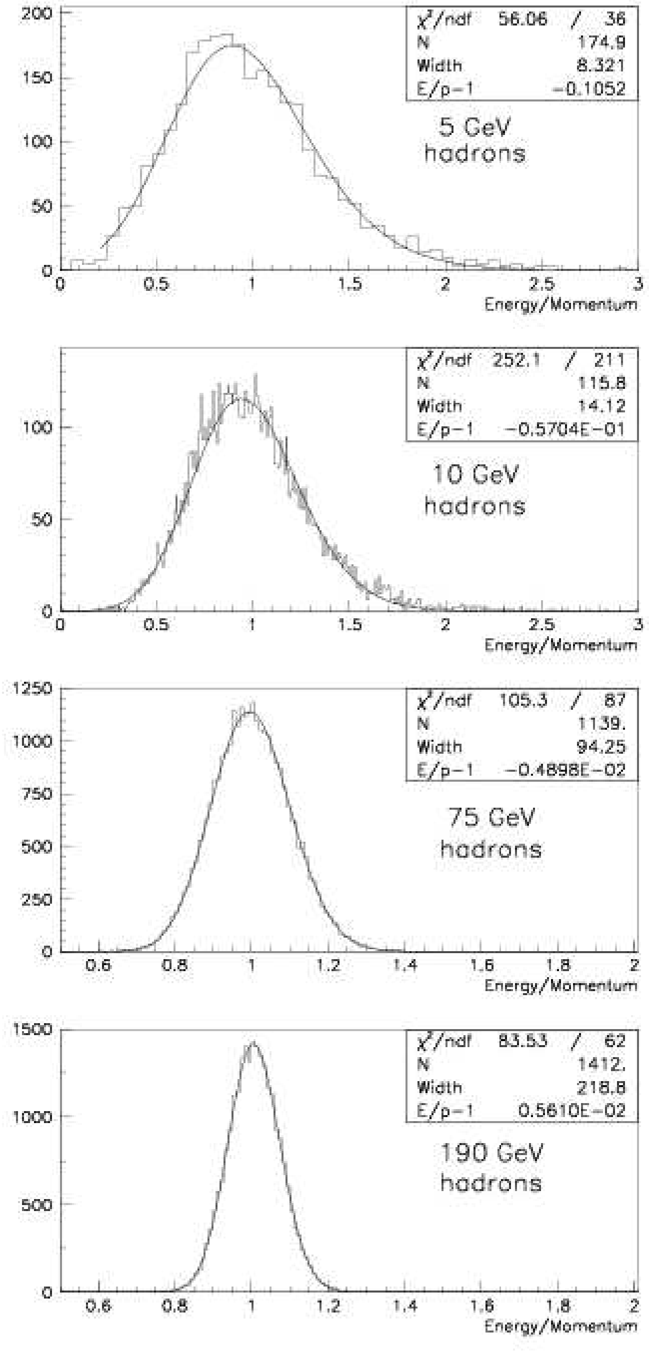


Figure 2.19: Poisson fits to the ratio of measured hadron calorimeter energy to reconstructed calibration beam momentum distributions for 5 GeV, 10 GeV, 75 GeV, 190 GeV hadrons.

The figure shows the calorimeter energy E divided by the reconstructed calibration beam momentum p. The hadron conversion factor C_π converts the pulse height MIP response of the detector to GeV. It is determined by fitting the hadron energy dependencies of the average E/p distributions for hadrons ranging from 10 to 190 GeV, as shown in Figure 2.20.

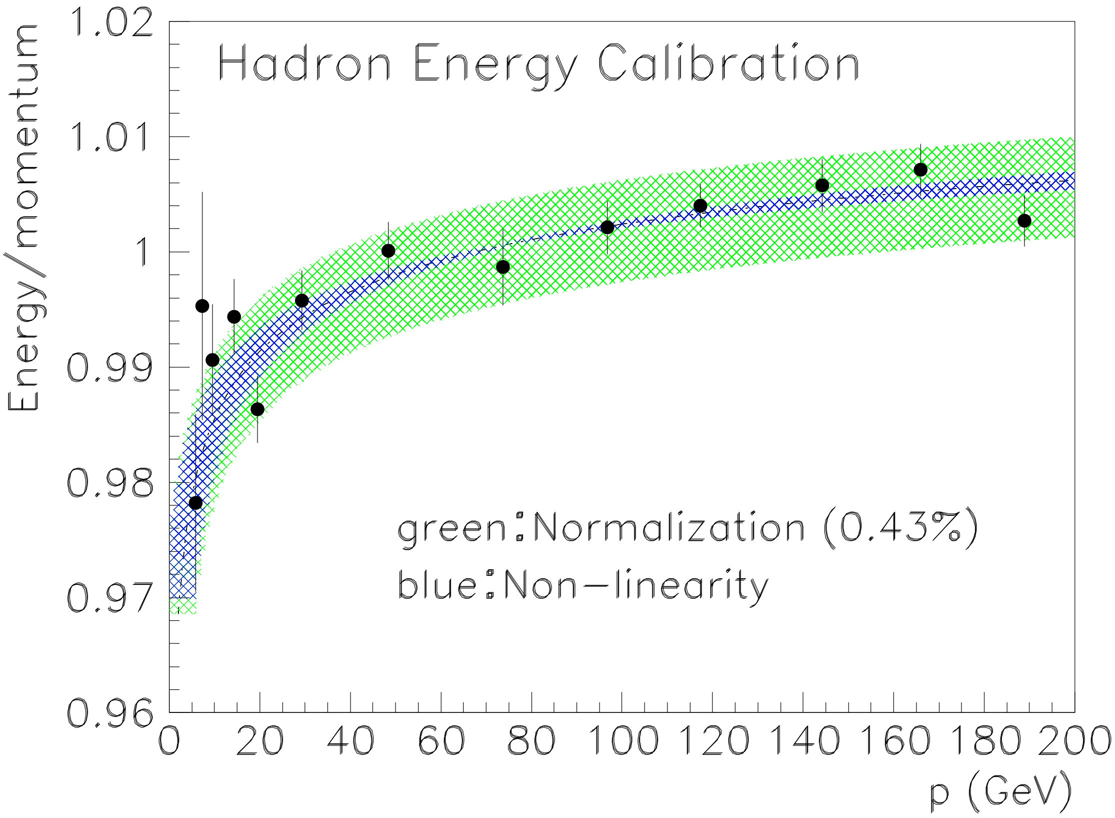


Figure 2.20: Ratio of hadron energy response over the calibration beam momentum and the comparison with the fit for the hadron non-linearity (the blue line - narrow band) of the calorimeter between 10 -190 GeV. The green band (large band) reflects the overall uncertainty of 0.43% in the hadron energy scale.

The hadron showers have nearly the same probability of producing neutral pions π^0 as π^+, π^-. The π^0 particle decays into two photons that initiate electromagnetic cascade. This causes a non-linear response of the detector to hadron energy (i.e. there is an electromagnetic component in the shower). This non-linear response to hadrons in the calorimeter is taken into account. Thus, the calibration for a realistic hadronic shower that contains both the "pure" hadrons and electrons can be expressed as

$$C_\pi = e f_{\pi^0}(E_{had}) + h[1 - f_{\pi^0}(E_{had})], \tag{2.14}$$

where h, e are the calibration constants for a "pure" hadronic shower, electromagnetic shower, respectively; f_{π^0} is the fraction of π^0 produced in the shower (i.e., the fraction of electromagnetic energy in a hadronic shower). The hadron energy response is fitted assuming an energy dependent parametrization for f_{π^0} given by Groom's parametrization (Ref. [35]),

$$f_{\pi^0}(E_{had}) = 1 - \left(\frac{E_{had}}{A}\right)^B,$$

with $A = 0.96$ GeV and $B = -.0184$. It yields a value of $e/h = 1.079 \pm 0.011$ for the ratio of electron to hadron response. The non-linearity in the hadron energy response is found to be about 3%. The green band (large band) in Figure 2.20 reflects the overall uncertainty of 0.43% in the hadron energy scale. For $C_\pi = 0.212$ GeV/MIPs the ratio of the mean energy response in the detector to the reconstructed hadron calibration beam momentum at 75 GeV is 1.000 ± 0.001.

The energy resolution as function of energy is determined through a fit to Poisson-width using a standard form:

$$\frac{\sigma(E)}{E} = A \oplus \frac{B}{\sqrt{E}} \oplus \frac{C}{E}, \tag{2.15}$$

where A is a constant term associated with the calibration uncertainties, B is the stochastic term associated with sampling fluctuations, and C is the noise due to pedestal fluctuations (this term was removed later from the fit because data showed no evidence for noise). Figure 2.21 shows the fit to Poisson widths of E/p distributions as function of energy. The energy

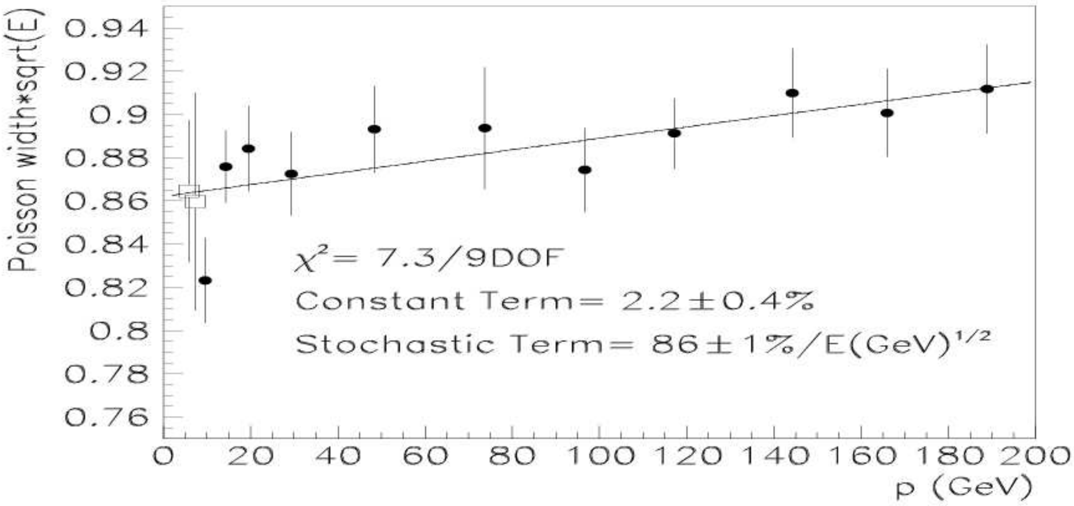


Figure 2.21: Measured hadron resolution from fit to the width of Poisson-like distributions.

resolution of the calorimeter was found to be

$$\frac{\sigma(E)}{E} = 0.022 \pm 0.001 \oplus \frac{0.86 \pm 0.01}{\sqrt{E}} \tag{2.16}$$

To study any possible difference between hadron showers produced by neutrino interactions and showers produced by single charged pions a GEANT-based Monte Carlo with LEPTO interaction generator was used Differences were found to be small. The overall energy scale is consistent with zero to better than 0.1% [28].

The electromagnetic conversion factor C_e was determined from the calorimeter response to electrons using the electron calibration beam. Figure 2.22 shows the Poisson fits to the ratios of calorimeter electron energy to the calibration electron beam for 30 and 170 GeV. It was obtained a value of $C_e = 0.195$ GeV/MIP for electron calibration constant. The electron resolution is determined to be

$$\frac{\sigma(E)}{E} = 0.042 \pm 0.002 \oplus \frac{0.499 \pm 0.008}{\sqrt{E}}. \tag{2.17}$$

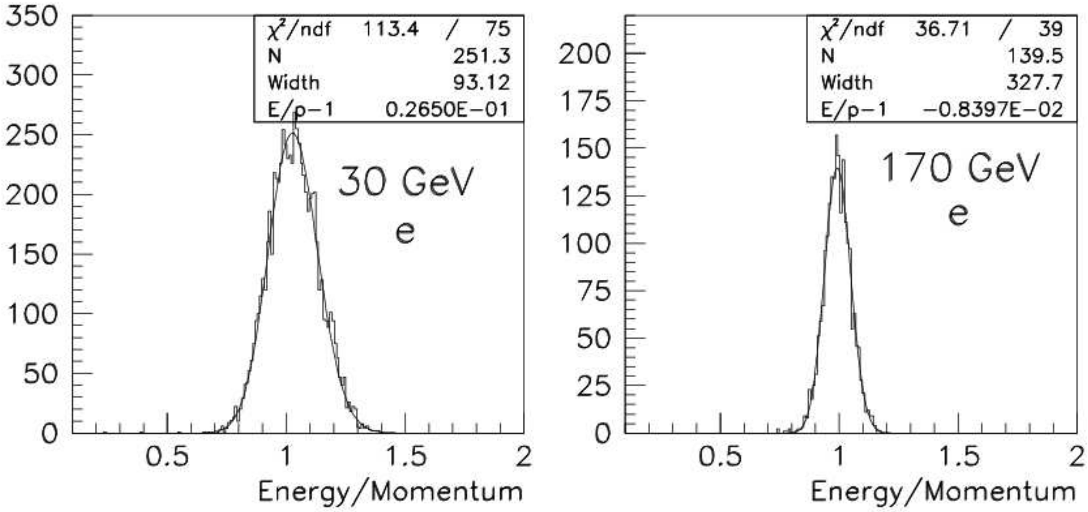


Figure 2.22: Poisson fits to the ratio of measured electron calorimeter energy to reconstructed calibration beam momentum distributions for 30 and 170 GeV electrons.

The muon conversion factor C_μ from MIPs to GeV was determined from muon energy loss in the calorimeter which is attributed to ionization loss. The parametrization for C_μ is determined from fitting muon pulse height distribution as

$$C_\mu(E_\mu) = \begin{cases} 0.1419 + 0.004357 \log E_\mu, & E_\mu < 60 \text{ GeV} \\ 0.1419 + 0.004357 \log(60) + 0.00006822(E_\mu - 60), & E_\mu \geq 60 \text{ GeV}. \end{cases} \qquad (2.18)$$

2.5.3 Toroid Response Calibration

The muon energy calibration was performed using the calibration beam of muons at 50 and 100 GeV scanning the entire face of the toroid. There were 45 calibration beam runs used in the muon energy calibration. For each test beam muon run a fit to the average ratio of the muon momentum measured by the calibration beam spectrometer to the muon momentum measured by NuTeV detector is determined (i.e. P_μ^{TB}/P_μ^{DET}). Figure 2.23 shows the fitted mean location at the front face of the toroid for muon calibration beam runs. The

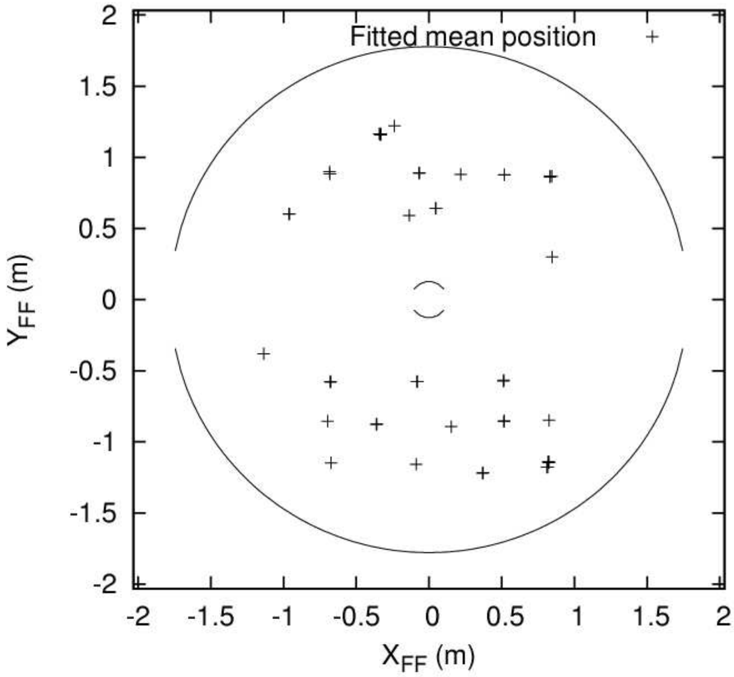


Figure 2.23: Location at the front face of the toroid of the fitted mean position of the ratio of the calibration beam muon momentum to the measured momentum by NuTeV detector for all 45 muon runs.

absolute muon energy scale uncertainty is given by the variance of these fitted means and is determined to be 0.7% [22].

The resolution for the toroid spectrometer is limited by the MCS effects in the steel toroids. A typical sigma for an angular deflection due to MCS is proportional to inverse muon momentum (see Eq. (2.8)). The uncertainty due to chamber resolution is constant. Therefore, the momentum resolution function is given by

$$\frac{\Delta(1/P_\mu)}{1/P_\mu} = \frac{P_\mu^{TB} - P_\mu^{DET}}{P_\mu^{DET}}, \tag{2.19}$$

where P_μ^{TB} is the muon momentum measured by the test beam spectrometer, and P_μ^{DET} is muon momentum distribution measured by the NuTeV detector. The resolution function of the muon spectrometer is fitted to the following parametrization:

$$\frac{\Delta(1/P_\mu)}{1/P_\mu} = exp\left[-\frac{1}{2}\frac{x^2}{\sigma^2_{lead}(P_\mu)}\right] + R_{tail}(P_\mu)exp\left[-\frac{1}{2}\frac{(x - x_{tail}(P_\mu))^2}{\sigma^2_{tail}(P_\mu)}\right], \qquad (2.20)$$

where $\sigma_{lead}(P_\mu)$ is the width of the leading Gaussian contribution due to MCS, and $R_{tail}(P_\mu)$, $\sigma_{tail}(P_\mu)$ and $x_{tail}(P_\mu)$ are the height, the width and the offset of the asymmetric tail of the resolution function. The width of the leading Gaussian has a small energy dependence and is parametrized as

$$\sigma_{lead}(P_\mu) = \sqrt{A^2 + (BP_\mu)^2}. \qquad (2.21)$$

The tail parameters instead are parametrized with linear functions. The response of the detector to muon calibration beam at 100 GeV is fitted using the above parametrization of the resolution function and is shown in Figure 2.24. The fit parameters for a muon momentum at 100 GeV are given in Table 2.4 (see Ref. [22]).

2.5.4 Muon Angle Calibration

Muon angle at the vertex is determined by fitting the muon track near the vertex. The number of hits near the vertex is large due to the hadron shower which limits the tracking ability. Therefore, the resolution of the muon angle is limited by hadron energy and by uncertainty due to MCS. The muon angle measurement is correlated with muon momentum, therefore the procedure is iterated using GEANT hit level simulation [36] of the detector to parametrize the angular resolution function (see Sect. 3.2.3).

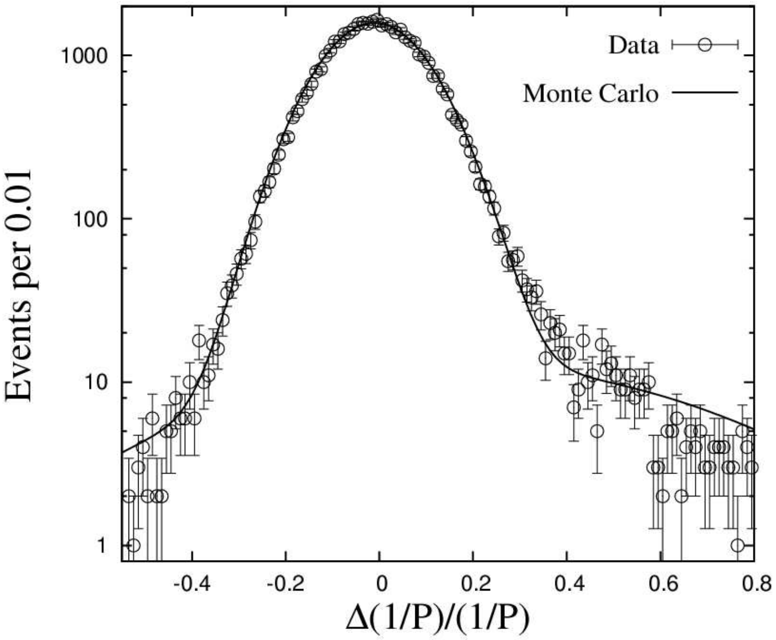


Figure 2.24: Fit to the detector response to muon calibration beam of 100 GeV. Plot taken from Ref. [22]

Table 2.4: Parameters of the muon energy resolution function for a muon momentum entering the toroid at 100 GeV.

$\sigma_{lead}(100)$	width of leading Gaussian	0.11
$\sigma_{tail}(100)$	width of the asymmetric tail function	0.47
$x_{tail}(100)$	offset of the asymmetric tail	0.18
$R_{tail}(100)$	normalization coefficient	0.079

3.0 EXTRACTION OF THE DIFFERENTIAL CROSS SECTION

The strong coupling constant is determined from fits to the extracted neutrino-nucleon structure functions. However, the quantity which is directly measured by the experiment is the neutrino-nucleon differential cross section on iron.

Measured physical quantities are smeared by detector resolution effects and are also limited by detector acceptance. In order to account for these effects a Monte Carlo simulation is used which is meant to convert smeared measured distributions into true physical distributions. The simulation employs a complete description of the detector and a cross section model which is iteratively fitted to differential cross-section data. In this Chapter we summarize the mechanism of extraction of the differential cross section as it is schematically shown in Figure 3.1 (detailed description can be found in Ref. [22]). The reminder of this chapter presents the extraction of the structure functions which are used in determination of the strong coupling constant.

3.1 DATA SELECTION

Once data is reconstructed as described in the previous Chapter, the events are selected for the structure function analysis to ensure good efficiency and small acceptance corrections. The selection criteria is divided into fiducial-volume cuts and geometric cuts.

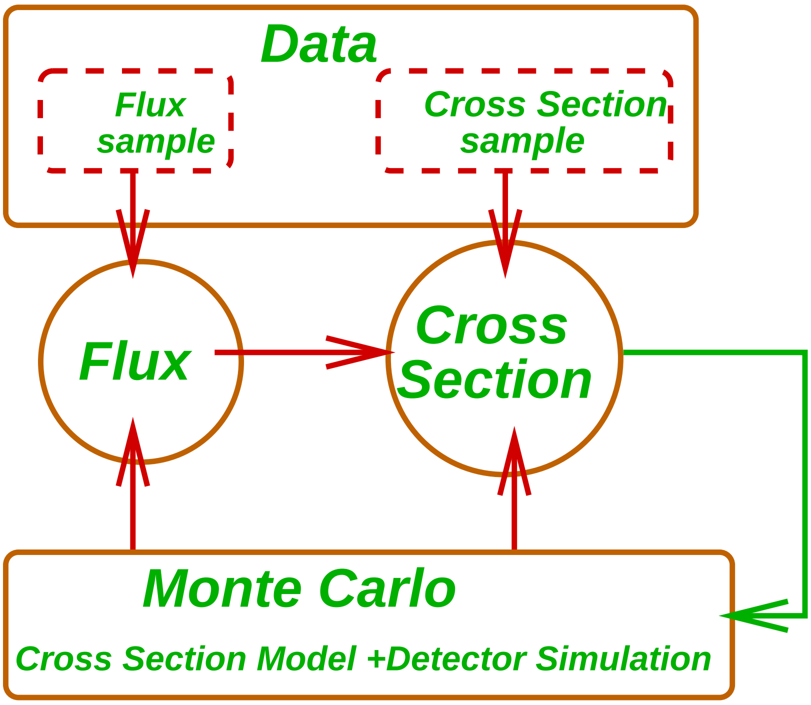


Figure 3.1: A diagram of the cross-section extraction mechanism.

3.1.1 Fiducial-Volume Cuts

The fiducial volume cuts assure that the event takes place within the active area of the detector. These are:

- **BADRUNS:** removes runs or events for which there are known detector hardware failures.
- **IGATE:** selects neutrino run of the accelerator cycle (fast gate).
- **NSTIME:** selects single interaction events (muon track time required to be within 36 ns of the hadron shower).
- **$20 < \text{PLACE} < 80$:** assures good longitudinal containment of the hadron shower in the calorimeter.
- **$|V_x| < 50$ in, $|V_y| < 50$ in:** assures good transverse containment of hadron shower in

the calorimeter.

3.1.2 Geometric Cuts

Geometric cuts assure good acceptance of the detector and quality of the muon track. The following cuts are considered:

- **TRIGGER 1:** selects the charged current events with muon track in the calorimeter and hits in the toroid drift chambers.
- **TARGET TRACK:** requires hits in the target drift chambers for a good identification of a muon track. This is important for an accurate muon angle determination.
- **TOROID CONTAINMENT:** the extrapolated muon track from hits in the target to toroid is required to have a radius at the front face of the toroid $\mathrm{R_{FF}} < 60$ in, and at T2 the radius to be $\mathrm{R_{T_2}} < 55$ in. This cut assures good muon track in the toroid.
- **TOROID TRACK:** requires hits in the toroid drift chambers for a good muon track reconstruction in the toroid.
- **HOLE CUT:** requires $\theta_\mu > 0.007$ rad and $\mathrm{R_{FF}} > 6$ in to assure muon not passing through the hole in the center of the spectrometer where magnetic field is not well modeled.
- **80%:** of toroid muon track within the steel of the toroid is required.
- **GOOD FIT:** requires a fitted muon track in the toroid, with number of degrees of freedom corresponding to numbers of drift chambers with hits.
- **TWO GAPS:** requires hits in the both first and second gaps drift chambers to ensure a good determination of muon momentum.

3.1.3 Data Samples

Data is divided into nearly two independent data samples used for cross-section analysis (cross section sample) and flux extraction (flux sample) as shown in Figure 3.1.

3.1.3.1 Flux sample

This sample corresponds to data of low hadronic energy used to determine the relative neutrino flux (described in Sect. 3.3). It requires in addition to fiducial volume and geometric cuts the following cuts:

- $\mathbf{E_{had} < 20}$ **GeV:** to allow the use of "fixed-ν_0" method to extract relative flux as discussed in Sect. 3.3.
- $\theta_\mu <$ **150 mrad:** to ensure good acceptance of the muon track in the toroid.
- $\mathbf{600 > E_\mu > 15}$ GeV and $\mathbf{E_\mu^{FF} > 3}$ GeV: Very high energy muons have straight tracks in muon spectrometer resulting in poor momentum resolution. Low energy muons are poorly reconstructed due to multiple Coulomb scattering.

Total number of events passing the flux cuts was found to be 4.3×10^5 for neutrinos and 2.1×10^5 for antineutrinos [22].

3.1.3.2 Cross section sample

This sample is used to extract the DIS neutrino differential cross section on Iron. In addition to fiducial volume and geometric cuts the following cuts are required:

- $\theta_\mu <$ **150 mrad,** $\mathbf{600 > E_\mu > 15}$ **GeV and** $\mathbf{E_\mu^{FF} > 3}$ **GeV** (the same as for the flux sample).
- $\mathbf{600 > E_{had} > 10}$ **GeV** – The low hadron energy cut minimizes the correlation with the flux sample, and removes quasi-elastic and resonances effects.

- $\mathbf{Q^2 > 1}$ GeV^2 –This cut minimizes the non-perturbative contributions, and reduces the contributions from "quasi-elastic" events.
- $\mathbf{360 > E_\nu > 30}$ GeV – This cut ensures that the flux is well understood and the muon momentum is well reconstructed.

There were 8.7×10^5 neutrino and 2.4×10^5 antineutrino events passing the cuts for cross section analysis [22]. Table 3.1 summarizes the number of events in each data sample after all cuts were applied.

Table 3.1: Number of events in each data sample.

Mode	Cross Section Sample	Flux Sample
ν	866,000	426,803
$\overline{\nu}$	240,000	210,795

3.2 MONTE CARLO SIMULATION

The only purpose of a Monte Carlo Simulation in the extraction of neutrino differential cross section analysis is to correct data samples for acceptance and smearing effects. For this, the NuTeV analysis uses a detailed Monte Carlo event simulation program called NUMONTE. The NUMONTE program provides a full tracking for high energy muons, but relies on the parametrization of the detector response from calibration beam data for other final state particles (e.g. hadron showers). Therefore, this Monte Carlo is fast, as it can simulate several million neutrino events in a matter of hours. Since cosmic rays and beam muons are removed from data through a stringent set of cuts, there is no need to model them.

The program consists of three main building blocks: cross section model, event generation of a neutrino interaction, and detector model to simulate the response of the detector for each generated event. Each of these blocks are discussed briefly in the following Sections.

3.2.1 Cross-Section Model

A neutrino cross-section model is used in extraction of the neutrino flux from the low-hadron energy data (described in Sect.3.3), in determination of the detector acceptance and resolution effects, and in physics corrections such as radiative corrections, isoscalarity, propagator corrections, charm production thresholds. Therefore, NUMONTE requires as input a model for neutrino cross-section which is iteratively fit to our data. The neutrino cross-section model in NUMONTE is based on the standard deep inelastic formalism which was described in Section 1.2. We assume the V-A structure of the neutrino cross section, and the charged-current neutrino-nucleon DIS cross-section is expressed in terms of three structure functions as in Eq. (1.27). The NuTeV analysis uses an enhanced LO QCD A. J. Buras - K. F.J. Gaemers model [38] with added terms to allow deviations from a strict LO calculation. This model is intended to be a simple phenomenological fit to our data (in order to reduce the theory dependencies which would otherwise contribute large uncertainties) and should not depend on specifics of QCD.

In an enhanced LO model, the structure functions can be expressed in terms of PDFs as

$$\begin{aligned}
2xF_1(x,Q^2) &= xu_v + xd_v + 2xu_s + 2xd_s + 2xs, \\
F_2(x,Q^2) &= 2xF_1(x,Q^2)\left(\frac{1+R_L(x,Q^2)}{1+4M^2x^2/Q^2}\right), \\
xF_3(x,Q^2) &= xu_v + xd_v,
\end{aligned}$$

where the PDFs are for an iron target. Thus, the cross section model can be parametrized in terms of these PDFs.

3.2.1.1 Parametrization

The PDF parametrization can be separated into valence quark (u_v, d_v), light quark sea ($\overline{u}$, $\overline{d}$), strange sea distributions (s, $\overline{s}$), charm sea (c, $\overline{c}$) (neglected in this model), and gluons. The parametrization is performed as function of Bjorken-x, at a starting value of $Q_0^2 = 12.6$ GeV^2 and then evolved to any Q^2 using functional forms similar to QCD, but without imposing exact QCD calculations.

- **valence distributions**:

  The parametrization is characterized with a functional form as

$$\begin{aligned} xu_v(x, Q_0^2) &= u_v^{tot}\left[x^{E_1}(1-x)^{E_2} + AV_2 x^{E_3}(1-x)^{E_4}\right], & (3.1)\\ xd_v(x, Q_0^2) &= d_v^{tot} xu_v(x, Q_0^2)(1-x), & (3.2)\end{aligned}$$

  where exponents $E_{1,\ldots,4}$ and parameter AV_2 describe u and d valence distribution, where the d valence distribution is connected to the u valence distribution by the $(1-x)$ factor (neutrino scattering is relatively insensitive to the difference between u_v and d_v). Parameters u_v^{tot} and d_v^{tot} are the overall normalizations for the valence distributions. The evolution of the valence distributions is performed according to a simple logarithmic function (proposed by Buras and Gaemers, [38]):

$$\begin{aligned} E_1 &= E_{10} + sE_{11},\\ E_2 &= E_{20} + sE_{21},\\ E_3 &= E_{30} + sE_{21},\\ E_4 &= E_{40} + sE_{21}, & (3.3)\end{aligned}$$

  where,

$$s = \log \frac{\log\left(\frac{Q^2}{\Lambda_0^2}\right)}{\log\left(\frac{Q_0^2}{\Lambda_0^2}\right)}. \qquad (3.4)$$

The exponents $E_{10,\ldots,40}$ correspond to parametrization of the valence distributions independent of Q^2. The parameter A_0^2 takes the place of Λ_{QCD} in QCD. Valence normalization parameters are constrained by the corrected Fermion Sum Rule (GLS) in powers of α_S as

$$\int_0^1 F_3(x,Q^2)dx \;=\; 3\left(1-\frac{\alpha_S(Q^2)}{\pi}\right) = 3\left(1-\frac{A_1}{\log\frac{Q^2}{A_0^2}}-\frac{A_2}{\left(\log\frac{Q^2}{A_0^2}\right)^2}\right), \qquad (3.5)$$

and by quark charge constraint as

$$\left(+\frac{2}{3}\right)\int_0^1 u_v(x,Q^2)dx+\left(-\frac{1}{3}\right)\int_0^1 d_v(x,Q^2)dx = 1-\frac{A_1}{\log\frac{Q^2}{A_0^2}}-\frac{A_2}{\left(\log\frac{Q^2}{A_0^2}\right)^2}. \qquad (3.6)$$

Note that although the parameters are constraint by QCD-like functional forms, their values are allowed to float (in order not to bias our results).

- **light sea distributions**:

The light sea quark distributions, $\overline{u}$, $\overline{d}$, are assumed to have the same behavior in x and Q^2. Their parametrization is given by

$$\begin{aligned} x\overline{u}(x,Q^2) &= x\overline{d}(x,Q^2) = \frac{1}{2(k+2)}xS(x,Q^2) \\ &= \frac{1}{2(k+2)}\left(AS(1-x)^{ES}+AS_2(1-x)^{ES_2}\right), \qquad (3.7) \end{aligned}$$

where k relates the normalization of the strange quark sea to the non-strange quark sea. The light quark sea distributions decrease rapidly with x, therefore can be be determined from the first two moments as

$$\begin{aligned} S_2 &= \int_0^1 xSdx = \frac{3}{4}D_{22}+\frac{1}{4}D_{12}, \\ S_3 &= \int_0^1 x^2Sdx = \frac{3}{4}D_{23}+\frac{1}{4}D_{13}, \qquad (3.8) \\ & \qquad (3.9) \end{aligned}$$

where D_{ij} are completely specified by LO QCD [38]. The parameters AS_2, ES_2 are evolved in Q^2 according to

$$
\begin{aligned}
AS_2 &= AS_{20} + AS_{21} \log Q^2, \\
ES_2 &= ES_{20} + ES_{21} \log Q^2,
\end{aligned}
\tag{3.10}
$$

and ES, AS depend on second and third moments of the light sea distributions.

- **strange sea distributions:**

  The s and $\overline{s}$ quark distributions are important in neutrino scattering because when a neutrino strikes a s or $\overline{s}$ quark it produces a heavy charm quark, which decays into a high-energy muon (dimuon event). Therefore, the strange sea is constrained by the LO fit to CCFR/NuTeV dimuon differential cross section data [39] [1]. This model assumes strange sea asymmetry, but it allows a different shape from the light quark distributions:

$$
\begin{aligned}
xs(x,Q^2) &= x\overline{s}(x,Q^2) \\
&= \frac{k}{2(k+2)}\frac{AS}{ES+1}(ES+1+\alpha)(1-x)^{ES+\alpha},
\end{aligned}
\tag{3.11}
$$

  where α describes the relative shapes between the strange and non-strange sea (i.e., light quark sea).

- **Charm sea distribution**: is neglected in this model.

- **Gluon distribution**: enters in the model only through missing quarks terms.

There are 19 free PDF fit parameters:

AV_2, A_0, A_1, A_2, E_{10}, E_{20}, E_{21}, E_{30}, E_{40}, S_2, S_3, AS_{20}, AS_{21}, ES_{20}, ES_{21}, A_1, A_2, G_0.

---

[1] New NLO NuTeV measurement of the strange sea from dimuon data was performed, see Ref. [26]

3.2.1.2 Physical Corrections

This parametrization of the cross section model needs to be adjusted to account for various physical effects not included in the LO model. The following corrections were added:

- **d/u Correction**. PDF parametrization assumes up and down seas the same. However, an asymmetry has been observed by Drell-Yan data [40]. The form of this asymmetry was parametrized as following:

$$f(\overline{d}/\overline{u}) = \frac{1}{MAX(1 - x(2.7 - 0.14\log Q^2 - 1.9x), 0/1)}. \tag{3.12}$$

Thus, the light sea quark distributions corrected for this asymmetry can be written as

$$\begin{aligned} \overline{u'} &= \overline{u}\frac{\overline{u}+\overline{d}}{\overline{u}+f(\overline{d}/\overline{u})\overline{d}}, \\ \overline{d'} &= \overline{d}\frac{\overline{u}+\overline{d}}{\overline{u}+f(\overline{d}/\overline{u})\overline{d}}f(\overline{d}/\overline{u}), \end{aligned} \tag{3.13}$$

with the constraint that $\overline{u}+\overline{d} = \overline{u'}+\overline{d'}$.

Valence distributions (d_v and u_v) also are corrected using parametrization of F_2^d/F_2^p NMC data [18], so that

$$\delta(d/u) = 0.12079 - 1.3303x + 4.9829x^2 - 8.4465x^3 + 5.7324x^4, \tag{3.14}$$

which lead to the following correction for valence distributions:

$$\begin{aligned} u'_v &= \frac{u_v}{1+\delta(d/u)\frac{u_v}{u_v+d_v}}, \\ d'_v &= \frac{d_v + u_v\delta(d/u)}{1+\delta(d/u)\frac{u_v}{u_v+d_v}}. \end{aligned} \tag{3.15}$$

- **Higher-Twist Correction.** This correction is used to account for non-perturbative QCD effects which are important at high-x and low Q^2 region. It is determined from fits to F_2 charged lepton data (proton and deuterium) [43, 45, 44] for $x > 0.4$ using an empirical model [42], which rescales x to include a Q^2 dependence such as,

$$x \rightarrow \xi_{HT} = x \frac{Q^2 + B_{HT}}{Q^2 + x A_{HT}}, \tag{3.16}$$

where A_{HT}, B_{HT} enter as fit parameters in the cross section model. Before the fit is performed, charged lepton data needs to be converted to neutrino-iron data. The following corrections have to be applied:

1. Correction for the electric charge assignment to quarks for electromagnetic interaction versus weak interaction. This correction is referred to as the "5/18th" rule:

$$F_2^{ld}(x, Q^2) = F_2^{\nu d}(x, Q^2) \frac{5}{18} \left(1 - \frac{3}{5} \frac{xs + x\overline{s}}{xd_v + xu_v + 2xd_s + 2xu_s + 2xs} \right), \tag{3.17}$$

   where F_2^{ld} is the deuteron structure function F_2 for charged lepton data and $F_2^{\nu d}$ is for neutrino data. This relation is valid at all orders of perturbation theory if it is defined in the DIS factorization scheme. Therefore, the CTEQ4D scheme (where D stands for DIS scheme) is used for the PDFs.

2. Correction for the difference in light versus heavy nuclear targets. This correction is given by a parametrization to the charged lepton data (described in Sect.1.3.4.3) of the form

$$f(x) = 1.10 - 0.36x - 0.28e^{-21.9x} + 2.77x^{14.4}. \tag{3.18}$$

The charged lepton data is included in the PDF fit in order to constrain better the fit at the edge of our data acceptance. The relative normalizations among each data sets are allowed to float. The χ^2 for each data set is calculated separately and then added to the fit of the NuTeV data to extract the parameters of the cross-section model. Figure 3.2 shows the effect of including the higher twist parametrization in the cross-section model (before and after) compared to SLAC and BCDMS $F_2(x, Q^2)$ data [43, 44] at high x and low Q^2. The solid curve in the Figure corresponds to our model.

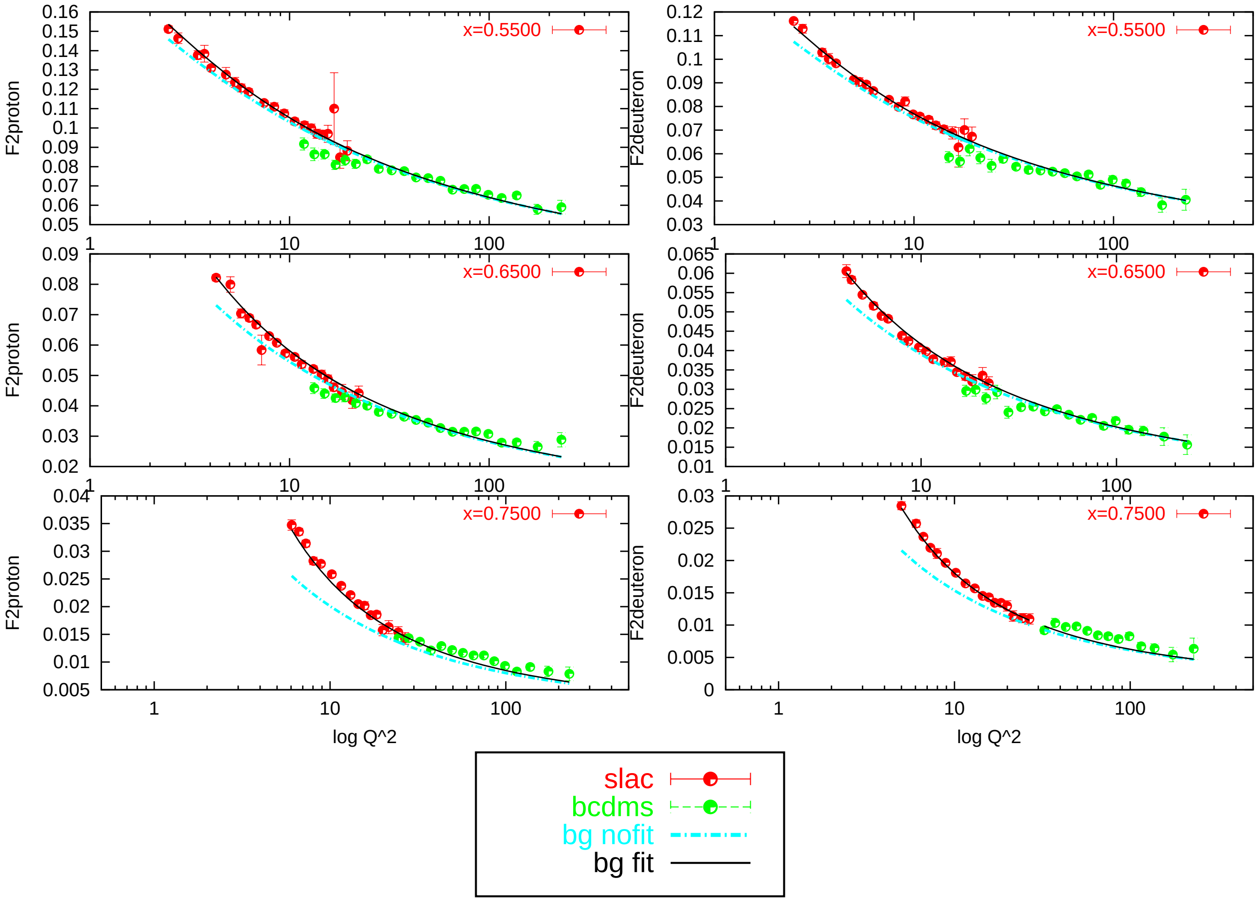


Figure 3.2: The effects of applying higher-twist effects on NuTeV cross section model compared on SLAC and BCDMS $F_2(x, Q^2)$ proton and deuteron data.

- **Low-Q^2 extrapolation.** At low x and Q^2 the Buras-Gaemers parametrization is not well behaved. Therefore, the Q^2 shape of GRV94LO PDFs [41] is used to extrapolate this model down to $Q^2 = 0.23$ GeV^2 with matching performed at $Q^2 = 1.35$ GeV^2. The extrapolation is used in order to model well the edge at low x and Q^2 of our data range.

- **Longitudinal Structure Function R_L.** It is introduced in the Monte Carlo Simulation in order to account for gluon effects in the cross section model. As a consequence, the Callan-Gross relation 1.36 is violated, such that

$$F_2(x, Q^2) = \frac{1 + R_L(x, Q^2)}{1 + \frac{4M^2x^2}{Q^2}} 2xF_1(x, Q^2). \tag{3.19}$$

For the structure function R_L an empirical parametrization of the world's available data on R_L is used [46]:

$$R_L(x, Q^2) = \frac{0.0635}{\log \frac{Q^2}{0.04}} \Theta(x, Q^2) + \frac{0.5747}{Q^2} - \frac{0.3534}{Q^4 + 0.09}, \tag{3.20}$$

where $\Theta(x, Q^2)$ is

$$\Theta(x, Q^2) = 1 + 12 \left(\frac{Q^2}{1 + Q^2} \right) \left(\frac{0.125^2}{x^2 + 0.125^2} \right). \tag{3.21}$$

- **Non-Isoscalar Target corrections.** The NuTeV target is made of iron which has an excess of neutrons compared to protons (non-isoscalar target) by

$$\frac{N - Z}{A} = 0.0567, \tag{3.22}$$

where $N,\ Z,\ A$ are the number of neutrons, protons, nucleons, respectively. This correction is taken into account in NUMONTE by computing separately the differential cross section for protons and neutrons. Then, a weighted average is computed as

$$\frac{d^2\sigma^{\nu(\overline{\nu})N}}{dxdy} = \frac{Z}{A} \frac{d^2\sigma^{\nu(\overline{\nu})p}}{dxdy} + \frac{N}{A} \frac{d^2\sigma^{\nu(\overline{\nu})n}}{dxdy}. \tag{3.23}$$

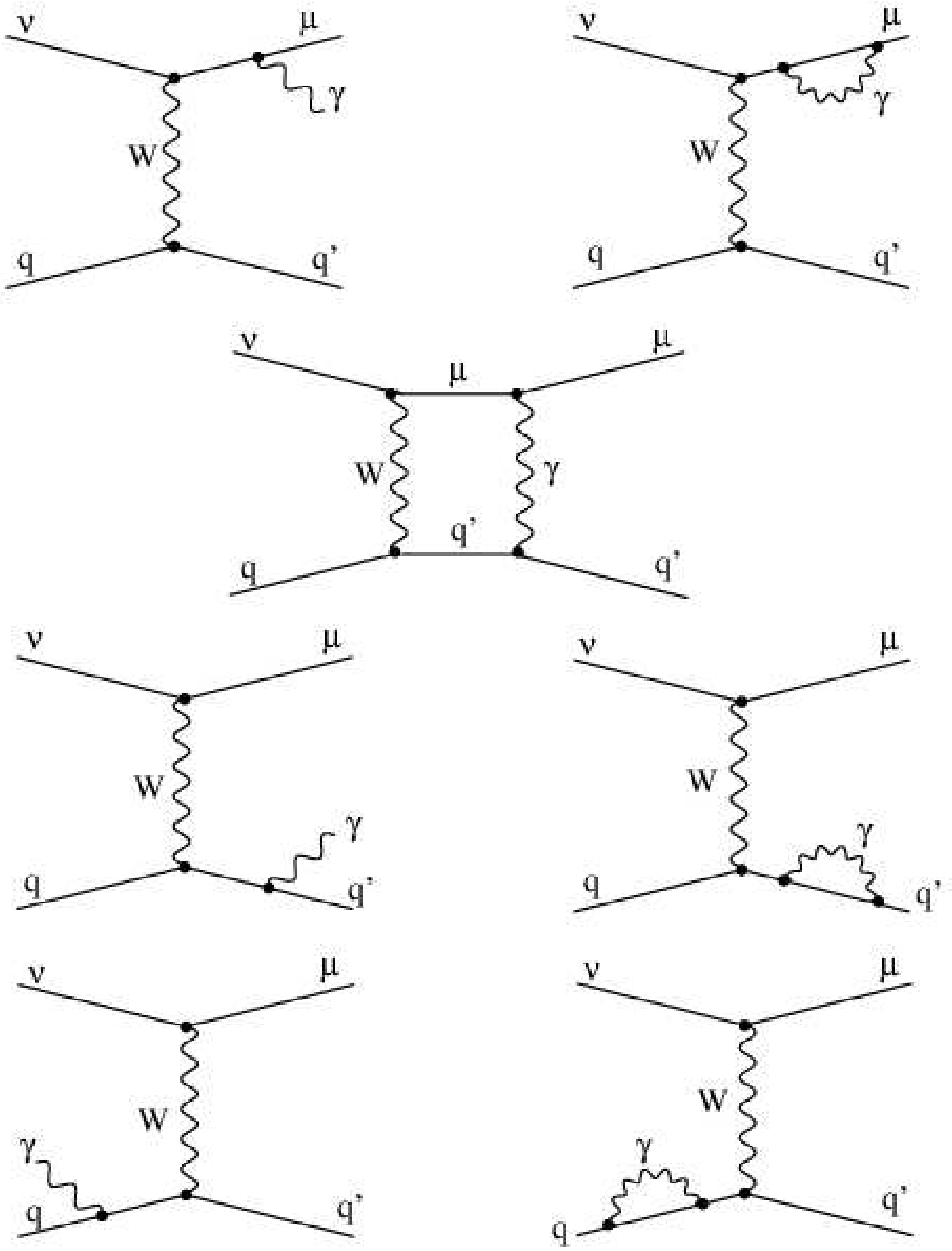


Figure 3.3: Feynman diagrams for QED radiative corrections for charged-current neutrino scattering.

- **Radiative Corrections** account for radiation of real and virtual photon by the charged particles. Figure 3.3 shows some of the Feynman diagrams for QED radiative corrections for CC neutrino scattering. The correction is computed by Bardin [47] within the framework of Quark Parton Model and stored in a table in $x, \ , y, \ E$ bins. This correction is applied to bare cross section as

$$\frac{d^2\sigma^{\nu(\overline{\nu})}}{dxdy} \longrightarrow \left(\frac{\left(\frac{d^2\sigma^{\nu(\overline{\nu})}}{dxdy}\right)_{QED-1-loop}}{\left(\frac{d^2\sigma^{\nu(\overline{\nu})}}{dxdy}\right)_{bare}} \right) \left(\frac{d^2\sigma^{\nu(\overline{\nu})}}{dxdy}\right)_{bare}. \tag{3.24}$$

- **Propagator Term.** The correction for the massive mediating W boson of a neutrino interaction is taken into account as follows:

$$\frac{d^2\sigma^{\nu(\overline{\nu})}}{dxdy} \longrightarrow \frac{1}{\left(1+\frac{Q^2}{M_W^2}\right)^2} \frac{d^2\sigma^{\nu(\overline{\nu})}}{dxdy}. \tag{3.25}$$

- **Charm Production Threshold.** About 10% of total charged current events produce heavy charm quarks when a neutrino strikes off a d or s quark. The model uses a slow rescaling method to account for charm production in neutrino scattering [15, 49]. This method rescales Bjorken-x taking into account charm mass m_c such as

$$x \longrightarrow \xi = x\left(1+\frac{m_c^2}{Q^2}\right). \tag{3.26}$$

The heavy quark thresholds are accounted in the model by applying the following modifications to the structure functions:

$$\begin{aligned} 2xF_1(x,Q^2) &\longrightarrow \frac{x}{\xi}2\xi F_1(\xi,Q^2) \\ xF_3(x,Q^2) &\longrightarrow \frac{x}{\xi}\xi F_3(\xi) \\ F_2(x,Q^2) &= \frac{1+R_L(\xi,Q^2)}{1+\frac{4M^2\xi^2}{Q^2}}2\xi F_1(\xi,Q^2). \end{aligned} \tag{3.27}$$

These adjustments depend on the charm mass parameter, its value is obtained from the weighted average of leading-order measurements, $m_c = 1.4 \pm 0.18$ GeV [39, 48]. This value is used through out this thesis.

3.2.1.3 Cross Section Fit Results

The PDFs are determined from the fit to the extracted cross section tables (described in Sect. 3.4) using the above cross section model. The χ^2 of the fit is constructed as follows:

$$\chi^2 = \sum_{i=1}^{N^{data}} \frac{(X_i^{DATA} - X_i^{model})^2}{\sigma_i^2}, \tag{3.28}$$

where X_i^{DATA} are the extracted NuTeV cross section data points and X_i^{model} corresponds to the cross section model parametrized through 21 parameters (including the higher twist parameters). The charged lepton data is included in this fit in order to constrain better cross section model as described in Sect. 3.2.1.2:

$$\chi^2 = \chi^2_{NuTeV} + \chi^2_{SLAC} + \chi^2_{NMC} + \chi^2_{BCDMS}. \tag{3.29}$$

The fit is performed for the kinematic region of $0.001 < x < 0.7$, $30 < E_\nu < 360$ (GeV), and $Q^2 > 1.35$ (GeV^2). Note that the fit is performed iteratively and the final fit results are presented in the Table 3.2 [22]. The iteration procedure is outlined in Sect.3.4.

3.2.2 Neutrino Event Generation

The probability of having a neutrino interaction in the detector is proportional to the flux of neutrino passing through the detector times the cross section of that interaction:

$$\mathrm{P}(E_\nu, V_x, V_y, x, y) \propto \Phi(E_\nu, V_x, V_y) \frac{d^2\sigma(E_\nu, x, y)}{dxdy}. \tag{3.30}$$

NUMONTE generates charged current neutrino events for a given cross section model (described above) and a neutrino flux (determined from low hadron energy data sample, as discussed in Sect. 3.3).

The event generation proceeds through the following steps [2]:

---

[2] NUMONTE assigns to the "TRUE" generated variables a "G" ending to their names to distinguish them from smeared variables.

Table 3.2: Fit results for NuTeV cross section model after final iteration.

Parameters	Value ± Error
A_0	0.583 ± 0.017
A_1	0.295 ± 0.013
A_2	0.17 ± 0.03
AV_2	637 ± 75
E_{10}	0.5333 ± 0.0025
E_{11}	-0.028 ± 0.011
E_{20}	2.61 ± 0.015
E_{21}	1.31 ± 0.045
E_{30}	4.56 ± 0.14
E_{40}	12.5 ± 0.35
S_2	0.1625 ± 0.0013
S_3	0.01589 ± 0.0004
G_3	0.031 ± 0.003
AS_{20}	1.06 ± 0.11
AS_{21}	1.76 ± 0.25
ES_{20}	185 ± 20
ES_{21}	8.4 ± 8.0
A_{HT}	1.187 ± 0.035
B_{HT}	0.3332 ± 0.02

- The neutrino type (neutrino or antineutrino mode) and toroid polarity is selected.

- Neutrino energy ($ENUG$) and vertex coordinates transverse to the beam axis (VXG,

VYG) are randomly selected from measured flux distribution (as discussed in Sect. 3.3).

- $PLACE$, the longitudinal interaction position, is randomly thrown from a uniform distribution.
- The Bjorken-XG and inelasticity YG are thrown according to the cross section model.
- The other kinematic variables like muon momentum ($EMUG$), hadronic energy ($EHADG$), muon angle ($THETAMUG$) can be calculated from the above generated distributions as in Eq. (1.5-1.11).
- The hadron calibration beam is used to determine the dependence of the shower distribution on hadron energy. This library of hadron showers is used to generate the longitudinal location of the shower end ($SHEND$). Thus, the hadron shower length, defined as $PLACE - SHEND + 5$ (as it is defined for data), is generated.
- The tracking of the muon through the target is performed by stepping through each counter unit and computing energy loss (simulating MCS for each step). The muon energy loss ($DEDXG$) is generated in two parts: restricted (due to ionization) and catastrophic losses (due to Bremsstrahlung, pair production, delta ray production). Range tables are used for small energy losses (< 1 GeV). Therefore, the generated muon energy at the front face of the toroid ($EMUFFG$) is obtained by subtracting $DEDXG$ from the generated muon energy at the vertex ($EMUG$), $EMUFFG = EMUG - DEDXG$.

3.2.3 Detector Model

To simulate the detector response, NUMONTE smears the generated kinematic variables using either a parametrization from calibration beam data or a simulated response by GEANT hit-level Monte Carlo [36]. The same procedure as in data is used to reconstruct the variables. Bellow we sketch the procedure to smear the main generated kinematic quantities used in NuTeV analysis (see Ref. [22] for more details).

- **Muon Energy smearing and reconstruction:**

  The total muon energy is reconstructed in three pieces: muon energy in the toroid, in the target calorimeter beyond the shower region, and within the shower region.

  The generated toroid muon energy is smeared by sampling the toroid resolution function, given by

  $$R = \frac{1/P_\mu^{FF}(meas) - 1/P_\mu^{FF}(gen)}{1/P_\mu^{FF}(gen)}, \tag{3.31}$$

  where $P_\mu^{FF}(meas)$ and $P_\mu^{FF}(gen)$ are the smeared and generated muon momenta in the toroid. The muon momentum resolution function is displayed in Figure 2.24.

  The restricted energy loss is smeared with a Gauss distribution of width 0.045 GeV. The catastrophic energy loss is instead smeared using calibration data for electrons with a Gaussian smearing function of width $0.5\sqrt{E_{catastrophic}}$.

  NUMONTE simulates the pulse height(in MIP) for each counter using the conversion factors determined from the calibration beams (see Sect. 2.5). The restricted energy loss is converted to MIPs using C_μ conversion factor, while the catastrophic energy loss uses the electromagnetic conversion factor, C_e. The simulated pulse height in each counter is given by the sum of catastrophic and restricted energies loss per each counter in MIPs. Therefore, the reconstructed muon energy in the target beyond the shower region is determined by summing up the pulseheights from SHEND-6 to 1 and converting them into GeV using Eq. (2.18). NUMONTE uses the same parametrization of the muon energy loss in the shower region as it is used for the real data.

- **Hadron Energy Smearing:**

  The generated hadron energy is corrected for the observed non-linearity in the NuTeV calorimeter, then it is smeared by sampling the Poisson-like resolution function given by Eq. (2.15). The contribution of the muon energy deposited in the shower region was simulated by adding to the smeared hadron energy the simulated pulse-heights in the counters within the shower region. The muon energy loss, which is parametrized according to the restricted energy loss, is subtracted from the hadronic energy.

- **Muon Angle Smearing:**

  As described in Sect. 2.5.4, angular resolution is dominated by MCS effects in the target and is parametrized using GEANT hit-level simulation.

$$\Delta\theta_\mu = (96.8 + 0.87PLACE + 0.24EHADG)\frac{1}{P_\mu^G}, \tag{3.32}$$

  where $P_\mu^G = \sqrt{(EMUG)^2 - m_\mu^2}$ is the generated muon momentum.

From the smeared generated variables presented above one ca obtain the remaining kinematic quantities using Eqs. (1.5-1.11).

Figures 3.4 and 3.5 show data distributions for muon energy, hadron energy, and muon angle at the vertex compared to Monte Carlo simulation(for neutrino and antineutrino modes). NUMONTE models real data very well, at the level of 2%.

3.3 NEUTRINO FLUX EXTRACTION

Obviously, the neutrino flux cannot be determined directly, while monitoring the secondary meson beam for this experiment was too intense to allow a precise measurement of the neutrino flux. Therefore, NuTeV measures the neutrino flux from the number of charged current interactions in the detector using a low hadronic data sample. This method is referred to as "fixed-ν_0" method [50, 33] (summarized below) and it actually determines the relative flux in energy bins. The overall normalization is determined using the world average of absolute cross section measurements.

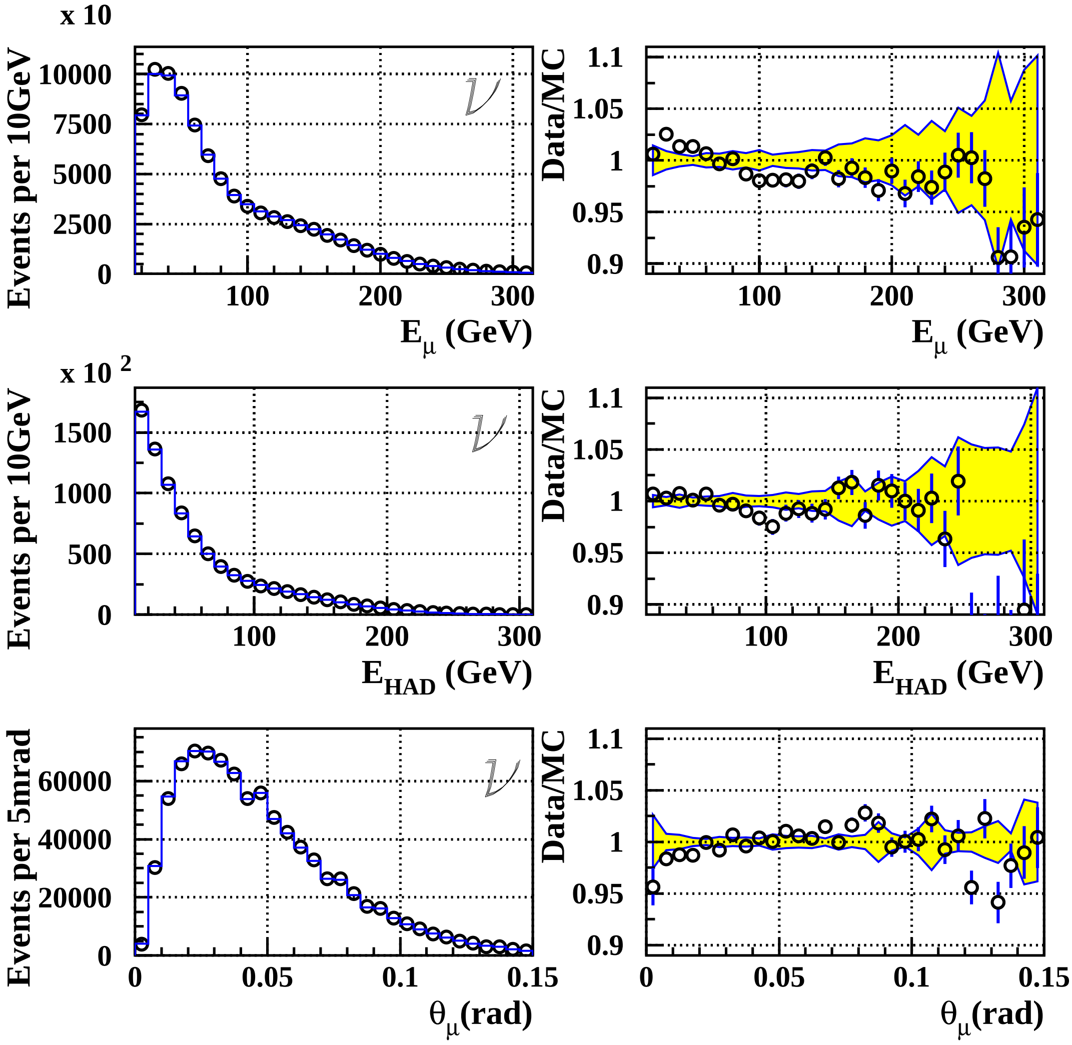


Figure 3.4: Data – Monte Carlo comparison for muon energy, hadron energy and muon angle (neutrino mode). Systematic uncertainty is shown with yellow band. Plots are courtesy of M. Tzanov.

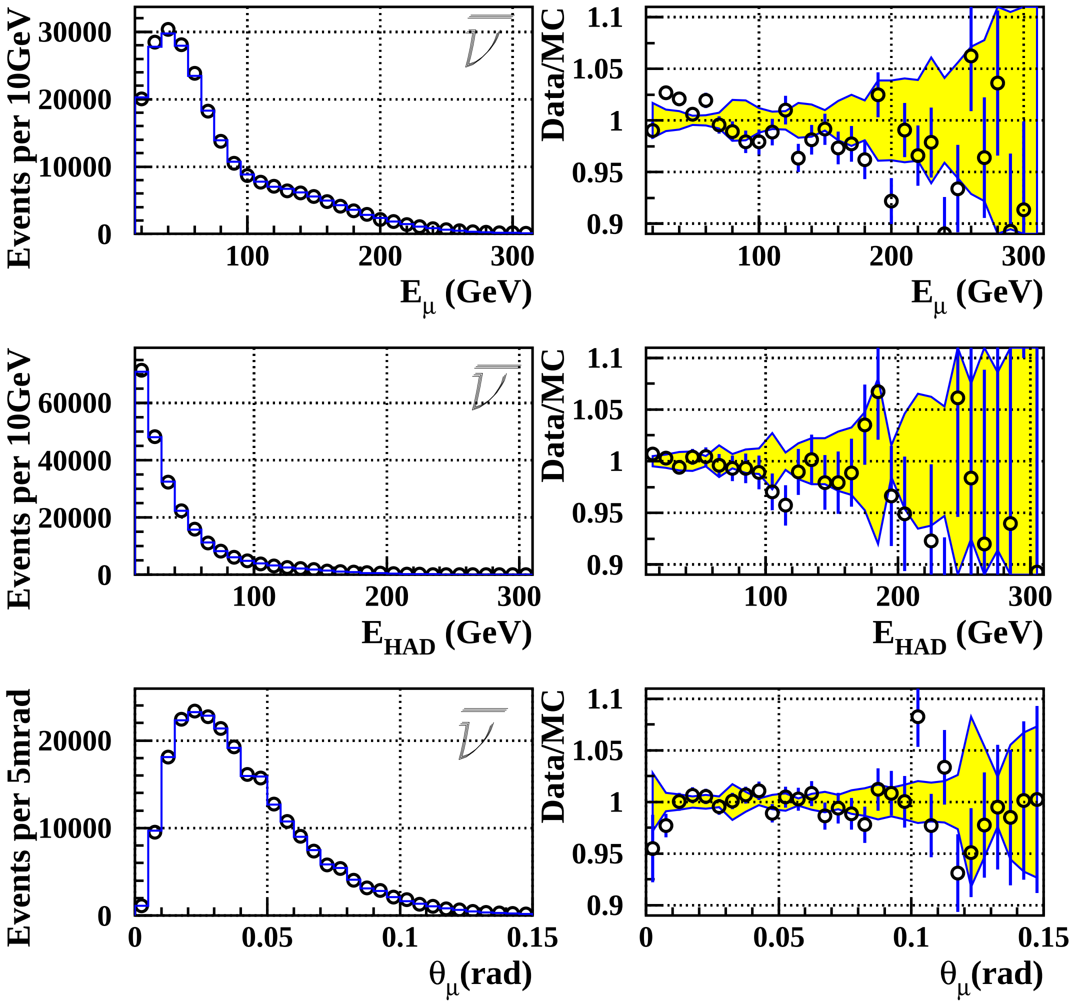


Figure 3.5: Data – Monte Carlo comparison for muon energy, hadron energy and muon angle (antineutrino mode). Systematic uncertainty is shown with yellow band. Plots are courtesy of M. Tzanov.

3.3.1 Fixed E_{had} Method

The relative flux is determined from flux data sample of low hadronic energy (sample defined in Sect.3.1.3.1). DIS cross section from Eq. (1.31) can be re-written in terms of $\nu \equiv E_{had}$ instead of inelasticity y using $\nu = E_\nu y = E_{had}$ as:

$$\frac{d^2\sigma^{\nu(\overline{\nu})}}{dxdy} = \frac{G^2 M}{\pi}\left[\left(1 - \frac{\nu}{E_\nu} - \frac{Mx\nu}{2E_\nu^2} + \frac{\nu^2}{2E_\nu^2}\frac{1+\frac{2M}{\nu}}{1+R_L}F_2 \pm \frac{\nu}{E_\nu}\left(1 - \frac{\nu}{2E_\nu}\right)xF_3\right)\right], \tag{3.33}$$

where the propagator term was dropped out. To ease the notations, we define the term that includes the longitudinal structure function R_L as

$$\tilde{R} \equiv \frac{1+\frac{2Mx}{\nu}}{1+R_L} - \frac{Mx}{\nu} - 1. \tag{3.34}$$

Integration over x for fixed ν and E_ν gives:

$$\begin{aligned}\frac{d\sigma^{\nu(\overline{\nu})}}{d\nu} &= \underbrace{\frac{G^2M}{\pi}\int F_2 dx}_{A} - \frac{\nu}{E_\nu}\underbrace{\frac{G^2M}{\pi}\left(\int F_2 dx \mp \int xF_3 dx\right)}_{B} \\ &+ \frac{\nu^2}{2E_\nu^2}\underbrace{\frac{G^2M}{\pi}\left(\int F_2 dx \pm \int xF_3 dx - \int \tilde{R}F_2 dx\right)}_{C}.\end{aligned} \tag{3.35}$$

In the limit $\nu \longrightarrow 0$ the differential number of events is proportional to the relative flux.

$$\frac{dN}{d\nu} = A\Phi(E_\nu)\left(1 + \frac{B}{A}\frac{\nu}{E_\nu} - \frac{C}{A}\frac{\nu^2}{2E_\nu^2}\right) \longrightarrow \Phi(E_\nu)A. \tag{3.36}$$

Substituting the coefficient C in terms of A and B the relation can be rearranged as

$$\frac{dN}{d\nu} = A\Phi(E_\nu)\left(1 + \frac{B}{A}\left(\frac{\nu}{E_\nu} - \frac{\nu^2}{2E_\nu^2}\right) + \frac{\nu^2}{2E_\nu^2}\frac{\int F_2\tilde{R}dx}{\int F_2 dx}\right), \tag{3.37}$$

where

$$\frac{B}{A} = -\left(1 + \frac{\int xF_3 dx}{\int F_2 dx}\right). \tag{3.38}$$

Integrating Eq. (3.37) over ν yields the following relation for the relative flux:

$$N(E_\nu)|_{\nu<\nu_0} = \Phi(E_\nu) \int_0^{\nu_0} d\nu A \left[1 + \frac{B}{A}\left(\frac{\nu}{E} - \frac{\nu^2}{2E_\nu^2}\right) + \frac{\nu^2}{2E_\nu^2}\frac{\int F_2\tilde{R}dx}{\int F_2 dx}\right], \tag{3.39}$$

where the integration is performed up to a fixed ν_0 cut chosen to ensure enough statistics in high energy bins ($\nu_0 = 20$ GeV). Hence, the energy dependence of the incident neutrino flux is directly proportional to integrated number of events in energy bins up to some small corrections in powers of $\frac{\nu}{E}$.

3.3.2 Flux Extraction

For the extraction of the flux, the raw number of events from the flux sample (see Sect. 3.1.3.1) has to be corrected for detector acceptance and resolution effects using a Monte Carlo simulation. The events are binned in E_ν, E_{had} bins to form a $\frac{dN(E)}{d\nu}$ distribution for a given E_ν. The relative flux is then determined from:

$$\Phi(E_\nu) \propto \int_0^{\nu_0} \frac{\frac{dN(E)}{d\nu}}{1 + \left\langle\frac{B}{A}\right\rangle\left(\frac{\nu}{E} - \frac{\nu^2}{2E_\nu^2}\right) + \frac{\nu^2}{2E_\nu^2}\frac{\int F_2\tilde{R}dx}{\int F_2 dx}} d\nu, \tag{3.40}$$

where $\frac{\nu}{E_\nu}$ is the correction accounting for energy dependence. The integral term $\frac{\int F_2\tilde{R}dx}{\int F_2 dx}$ from the denominator is calculated using PDFs from our cross section model. The $\frac{B}{A}$ term is obtained by a fit to $\frac{dN}{d\nu}$ data for each energy bin corrected for the radiative effects, charm mass, isoscalar target, and propagator (as described in 3.2.1). These corrections need to be applied because the expression which relates $\frac{B}{A}$ to $\frac{dN}{d\nu}$ assumes a bare DIS differential cross-section with massless quarks.

The fit is performed over $5 < \nu < 20$ GeV range for each neutrino energy bin. The lower bound is chosen to reduce the quasi-elastic and resonance contributions. The average

$\left\langle \frac{B}{A} \right\rangle$ is determined by averaging $\frac{B}{A}$ from fits in all energy bins separately for neutrinos and antineutrinos, obtaining the following values [22]:

$$\begin{aligned} < B/A >^{\nu} &= -0.34 \pm 0.04; \\ < B/A >^{\overline{\nu}} &= -1.68 \pm 0.03. \end{aligned}$$

3.3.3 Flux normalization

Once the relative flux is determined, the total neutrino cross section can be extracted by:

$$\frac{\sigma}{E_{\nu}} = \frac{N^{DATA}}{E_{\nu}\Phi(E_{\nu})}, \tag{3.41}$$

where N^{DATA} is the number of events from a data sample that selects all the events containing a muon, corrected for detector acceptance and resolution effects (see Ref. [22] for more details).

The absolute incident neutrino flux is determined by normalizing the measured total neutrino cross section to the world average neutrino cross-section for an isoscalar target [50]:

$$\frac{\sigma^{\nu}_{world}}{E} = 0.677 \pm 0.014 \times 10^{-38} \text{ cm}^2/\text{GeV}. \tag{3.42}$$

3.4 DIFFERENTIAL CROSS-SECTION EXTRACTION

The differential cross section for inclusive charge current neutrino iron scattering is extracted from the cross section data sample. The events are binned in 12 x bins, 11 y bins, and 7 E_ν bins. The average differential cross section for each kinematic bin can be expressed in terms of number of events and neutrino flux as

$$\left\langle \frac{d^2\sigma}{E_\nu dxdy} \right\rangle_{ijk} = \frac{XSEC^{DATA}_{ijk} \times CORR\_ACC_{ijk}}{\Phi(E^k_\nu)E^k_\nu \Delta x_i \Delta y_j}, \tag{3.43}$$

where i, j, k represents the x, y, E_ν bins and Δx, Δy are the widths of x and y bins chosen according to the event population and detector resolution. $XSEC^{DATA}(x, y, E_\nu)$ is the number of raw data events in each kinematic bin of teh cross section sample defined in Sect. 3.1.3.2. The NUMONTE program is used to correct the binned cross section data sample for detector acceptance and resolution effects. This correction is defined as follows:

$$CORR\_ACC_{ijk} = \frac{GEN^{DXSEC}_{ijk}}{SME^{DXSEC}_{ijk}}, \tag{3.44}$$

where GEN^{DXSEC}_{ijk} and SME^{DXSEC}_{ijk} are the generated and the smeared number of events in each bin of the high statistics Monte Carlo sample (more than 20 times data statistics). Also, a bin centering correction has to be applied to correct for the number of events at the average value of each kinematic bin versus the value at the bin center:

$$CORR\_BC_{ijk} = \frac{\frac{d^2\sigma}{Edxdy}(x^C_i, y^C_i, E^C_k)}{\frac{d^2\sigma}{Edxdy}(x^{ave}_i, y^{ave}_i, E^{ave}_k)}, \tag{3.45}$$

where $x^C_i = (x_{i+1} + x_i)/2$, $y^C_j = (y_{j+1} + y_j)/2$,$E^C_k = (E_{k+1} + E_k)/2$.

3.4.1 Iteration Procedure

Since Monte Carlo needs to be used both in neutrino flux extraction and in differential cross section extraction to account for detector effects, the process has to be iterated. The iteration procedure is as follows:

- Input an initial flux and PDF parameters for the cross section model from CCFR, the predecessor of the NuTeV experiment [51].
- Generate Monte Carlo events using the NUMONTE program. Produce acceptance corrections.
- Correct the flux and cross section samples for acceptance and resolution effects.
- Perform a fit to the differential cross-section and extract a new set of PDF parameters.
- Calculate the radiative corrections using the new PDFs and extract a new flux vertex distribution.
- Repeat this procedure till relative change in the differential cross section averaged over all data points is less that 0.1%. Require at least 20% acceptance for each kinematic bin on last iteration.

For the final iteration the final goodness of the fit per degree of freedom was found to be $\boldsymbol{\chi^2/\mathrm{dof} = 2225/2599}$**,** which includes the systematic uncertainties. There are 1423 number of neutrino differential cross section points and 1195 number of antineutrinos. The final fit results of the cross-section model parameters are shown in Table 3.2. Figures 3.6, 3.7 show an example of the extracted differential cross section distribution for neutrino and antineutrino scattering on iron as function of inelasticity y in various x bins, at 110 GeV and 190 GeV neutrino energy [22].

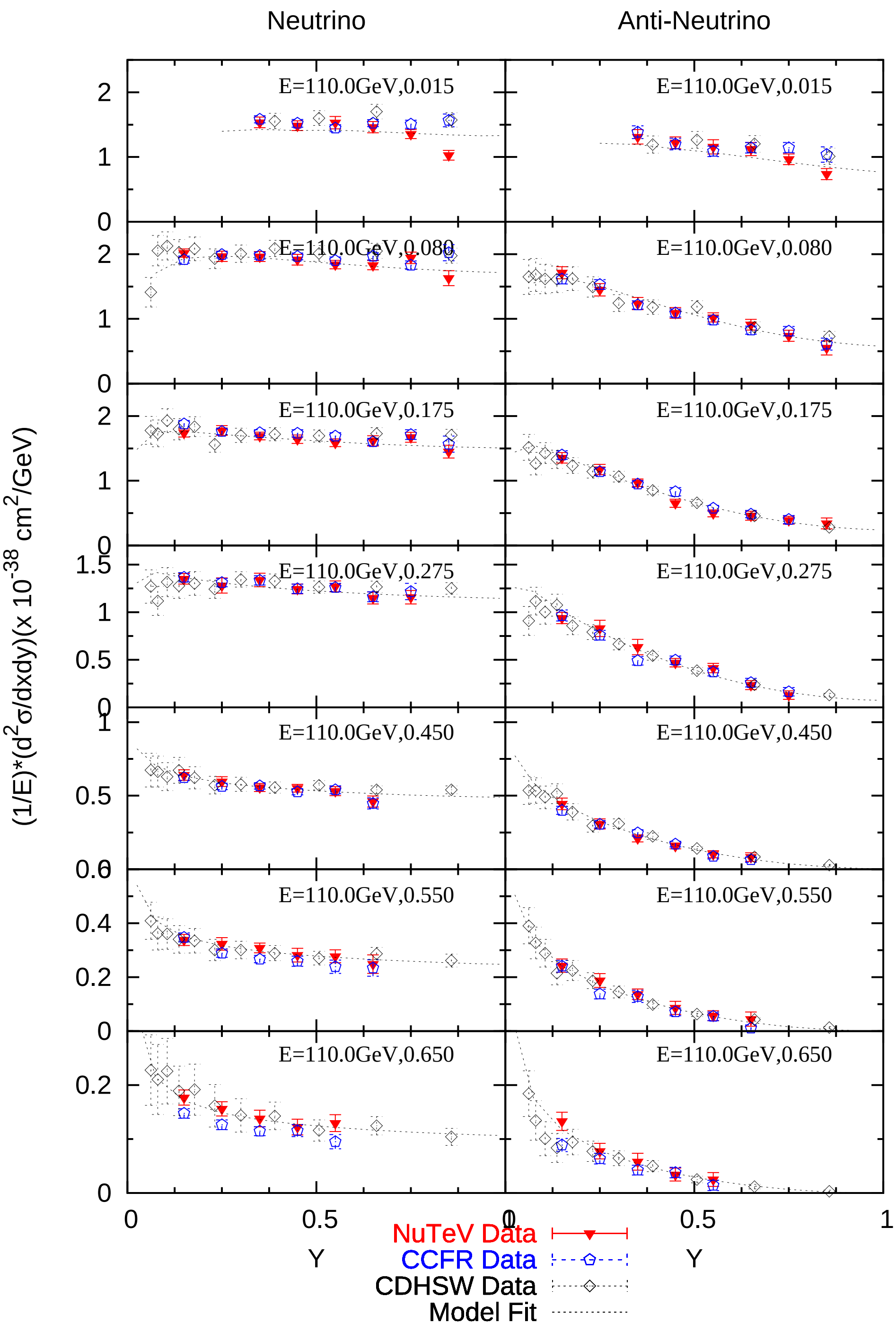


Figure 3.6: Comparison of NuTeV neutrino and antineutrino differential cross section (red triangles) to previous neutrino measurements on iron (CCFR -blue circles, CDHSW -black diamonds) as function of y for various x bins. This is shown for 190 GeV neutrino energy.

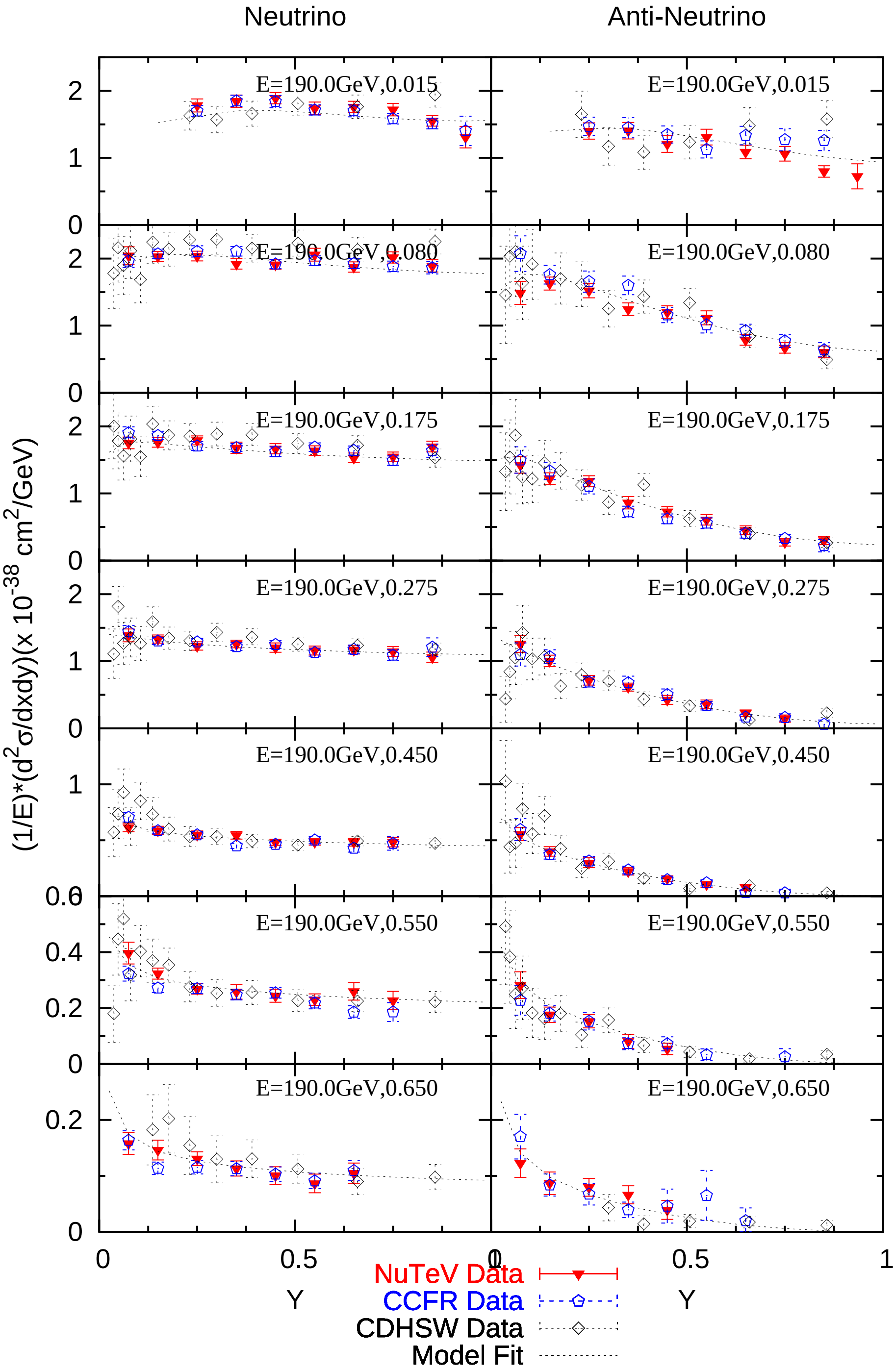


Figure 3.7: Comparison of NuTeV neutrino and antineutrino differential cross section (red triangles) to previous neutrino measurements on iron (CCFR -blue circles, CDHSW -black diamonds) as function of y for various x bins. This is shown for 190 GeV neutrino energy.

3.5 FLUX AND CROSS SECTION SYSTEMATIC UNCERTAINTIES

The following sources of systematic uncertainties are considered in the extraction procedure of the differential cross-section extraction:

- **Muon energy scale (0.7%):** affects both flux and cross-section extraction. To take the effect of this systematic into account, the data for both cross section and flux samples are shifted by $\pm 1\sigma$ and then re-extract the flux and cross-section. The result is compared to the nominal (default) cross-section.
- **Hadron energy scale (0.43%):** affects both flux and cross-section extraction. The effect of this systematic is taken into account by shifting both cross section and flux samples by $\pm 1\sigma$ and re-extract both the differential cross section and flux.
- **Charm mass:** is used in the charm production model to correct the flux sample to extract flux. The uncertainty on m_c is taken to be $\delta m_c = 0.18$ GeV. The effect of the systematic is evaluated by shifting the value by $\pm 1\sigma$ up and down and extract differential cross section for each shift.
- ***B*/*A*:** is the statistical error on the $\frac{dN}{d\nu}$ fit to the flux sample. It only affects the flux measurement. The effect of this systematic is evaluated by shifting $\frac{B}{A}$ by 1σ and re-extract flux. Using the shifted flux, a cross section is extracted and the output is compared to the nominal cross-section.
- **Muon energy smearing:** affects the acceptance correction from used to correct the raw data. Its effect is evaluated by taking the difference between two models: NuTeV smearing function and CCFR smearing function for muon energy smearing.
- **Hadron energy smearing:** affects the acceptance correction used to correct the raw data. The hadron smearing function is determined from a fit to the calibration data. The error from the fit is used to change the hadron energy response by $\pm 1\sigma$.
- **Flux normalization:** the uncertainty in the absolute flux determination is taken to be

2.1% (from the normalization to the world average neutrino cross-section). This is set to be the overall normalization of the experiment.

The dominant systematic errors come from the uncertainties in the muon and hadron energy scales. The uncertainties in the parameters of the cross section model are consistent with statistical fluctuations and have been neglected.

A point-to-point covariance matrix was calculated to account for the correlations between data points, which are large for neighboring bins. The covariance matrix is constructed as follows:

$$M_{\alpha\beta} = \sum_{i=1}^{7} \delta_{i|\alpha}\delta_{i|\beta}, \tag{3.46}$$

where $\delta_{i|\beta}$ corresponds to 1σ shift in data point α due to systematic uncertainty i. The statistical error is added in quadrature to the diagonal elements of the covariance matrix. NuTeV designed a package with a user friendly interface called NuTeVpack. It provides access to its measured neutrino and anti-neutrino differential cross section on Iron together with a full covariance error matrix which accounts for all systematic errors. NuTeVpack has the flexibility of turning ON and OFF any of the considered systematic uncertainties.

3.6 STRUCTURE FUNCTIONS EXTRACTION

Structure functions can be obtained from 1-parameter or 2-parameter fits to the y dependence of the linear combination of measured neutrino and antineutrino differential cross sections. The relation that connects structure functions to differential cross sections is based on the DIS bare differential cross section relation, i.e. Eq. (1.27). Therefore, before performing the fits data needs to to be corrected for non-isoscalarity, radiative, and massive propagator effects as described in Sect. 3.2.1.2. The corrections are applied as multiplicative factors to the extracted differential cross section data

$$C_{bare}(x,y,E) = \frac{\frac{d^2\sigma}{dxdy}(\text{ISO},\ M_W = 0\ \text{GeV, no rad. corr.})}{\frac{d^2\sigma}{dxdy}(\text{Fe},\ M_W = 80.4\ \text{GeV, rad. corr.})}. \tag{3.47}$$

Data is then re-binned from $(E,\ x,\ ,y)$ to express the structure functions as function of x and Q^2. The bin-center correction is given by

$$C_{Q^2} = \frac{\frac{d^2\sigma}{dxdy}(x,y,Q^2_c)}{\frac{d^2\sigma}{dxdy}(x,y,Q^2 = 2MxyE)}. \tag{3.48}$$

The procedures of extracting the nucleon structure functions used in the QCD analysis are presented below.

3.6.1 From 1-Parameter Fit

Nucleon structure function xF_3 is obtained from one-parameter fit to y-dependence of the following combination of the cross sections:

$$\left[\frac{d^2\sigma^\nu}{dxdy} - \frac{d^2\sigma^{\overline{\nu}}}{dxdy}\right]\frac{\pi}{2MG^2E_\nu} = \Delta F_2\left(1 - y - \frac{Mxy}{2E} + \frac{1+\frac{4M^2x^2}{Q^2}}{1+R_L}\frac{y^2}{2}\right) + \left(y - \frac{y^2}{2}\right)xF_3^{avg}, \tag{3.49}$$

$$\begin{aligned} F_{2(3)}^{avg}(x,Q^2) &= \frac{1}{2}\left(F^\nu_{2(3)}(x,Q^2) + F^{\overline{\nu}}_{2(3)}(x,Q^2)\right), && (3.50)\\ \Delta F_2 &= F_2^\nu - F_2^{\overline{\nu}} \approx 0, && (3.51) \end{aligned}$$

ΔF_2 is very small and is neglected, thus no need to input model for extracting xF_3. The resulting data table for the xF_3 structure function is presented in the Appendix B.1. Structure function xF_3 from one-parameter fit is used in the Non-Singlet QCD fit to determine Λ_{QCD} (see Sect. 4.2.2).

3.6.2 From 2-Parameter Fit

This is a new method of extracting the structure functions, and thus of Λ_{QCD}. One also could extract structure functions from 2-parameter fits to the $\nu(\overline{\nu})$ cross sections as follows:

$$\begin{aligned}
\frac{d^2\sigma^\nu}{dxdy} &= \frac{2MG^2E_\nu}{\pi}\left[\left(1-y-\frac{Mxy}{2E}+\frac{1+\frac{4M^2x^2}{Q^2}}{1+R_L}\frac{y^2}{2}\right)\left(F_2^{avg}+\frac{\Delta F_2}{2}\right)\right. \\
&+ \left. y\left(1-\frac{y}{2}\right)\left(xF_3^{avg}+\frac{\Delta xF_3}{2}\right)\right], \qquad (3.52) \\
\frac{d^2\sigma^{\overline{\nu}}}{dxdy} &= \frac{2MG^2E_\nu}{\pi}\left[\left(1-y-\frac{Mxy}{2E}+\frac{1+\frac{4M^2x^2}{Q^2}}{1+R_L}\frac{y^2}{2}\right)\left(F_2^{avg}-\frac{\Delta F_2}{2}\right)\right. \\
&+ \left. y\left(1-\frac{y}{2}\right)\left(xF_3^{avg}-\frac{\Delta xF_3}{2}\right)\right], \qquad (3.53)
\end{aligned}$$

$$\begin{aligned}
F_2^{avg}(x,Q^2) &= \frac{1}{2}\left(F_2^\nu(x,Q^2)+F_2^{\overline{\nu}}(x,Q^2)\right), \qquad (3.54) \\
xF_3^{avg}(x,Q^2) &= \frac{1}{2}\left(xF_3^\nu(x,Q^2)+xF_3^{\overline{\nu}}(x,Q^2)\right), \qquad (3.55) \\
\Delta xF_3 &= xF_3^\nu - xF_3^{\overline{\nu}}, \qquad (3.56) \\
\Delta F_2 &= F_2^\nu - F_2^{\overline{\nu}}. \qquad (3.57)
\end{aligned}$$

For this case, input models for ΔxF_3 and R_L are needed. For ΔxF_3 we use a NLO model from TRVFS with MRST99 PDF set (see Ref.[69]). Note that for $x > 0.1$, ΔxF_3 is negligible. For R_L a parametrization of world data is used (see Eq. (3.20)) [46]. The simultaneous extraction of F_2 and xF_3 uses the full covariance error matrix of the differential cross sections described in Sect. 3.5. A separate χ^2 is constructed for neutrino and antineutrino fits by minimizing the above relations. The resulting data tables for the F_2 abd xF_3 structure functions are presented in the Appendix B.2. These structure functions are used in the Combined QCD fits to determine Λ_{QCD} (see Section 4.2.3).

3.6.3 Structure Functions Systematic Uncertainties

The sources for the systematic uncertainties in extracting the structure functions originate from the extraction of the differential cross sections presented in Section 3.5. For the structure functions from two-parameter fits there are additional systematic errors due to the input models for ΔxF_3 and R_L. The systematic uncertainty due to ΔxF_3 model is evaluated by taking the difference between structure functions extracted with two different NLO models of ΔxF_3: TRVFS [69] and ACOTFFS [12]. The systematic uncertainty due to R_L parametrization is evaluated by shifting its value by ±15% (which corresponds to 1 sigma uncertainty in the parametrization) and re-extracting the structure functions for each shift.

4.0 Λ_{QCD} FROM NLO QCD FITS

Once the structure functions have been obtained, we can proceed to QCD analysis. In this Chapter we present the methods of extracting the fundamental parameter of QCD, Λ_{QCD} and the fit results. The Λ_{QCD} parameter is determined from two types of NLO QCD fits. The "non-singlet" fit uses only $xF_3(x,Q^2)$, while the "combined" fit uses both $F_2(x,Q^2)$ and $xF_3(x,Q^2)$ structure functions and their correlation. The non-singlet fit is independent of any assumption about gluon distribution. The combined fit has the advantage of improved statistical precision for Λ_{QCD}.

The fits are performed using two theoretical models: the massless $\overline{MS}$ scheme (referred to as $\overline{MS}$ model in the rest of this work) [52], and, for the first time, a model that takes into account heavy quark production using a scheme devised by Aivazis-Collins-Olness-Tung (referred to as $ACOT$ model) [12]. Both models have been described in section 1.3.1.

4.1 QCD PROGRAM

The NLO QCD fits are performed using a fitting FORTRAN program [53, 54] that allows the user to choose between $ACOT$ and $\overline{MS}$ models. Also, this program includes a complete treatment of the correlations between $F_2(x,Q^2)$ and $xF_3(x,Q^2)$. The program can be dissected in the following building blocks: the PDF parametrization and evolution, the NLO calculations of the structure functions using either $\overline{MS}$ or $ACOT$ model, and the minimization of χ^2 for either uncorrelated or correlated case. A schematic flow of the program is illustrated in Figure 4.1. Each of these components is discussed in detail in the following Sections.

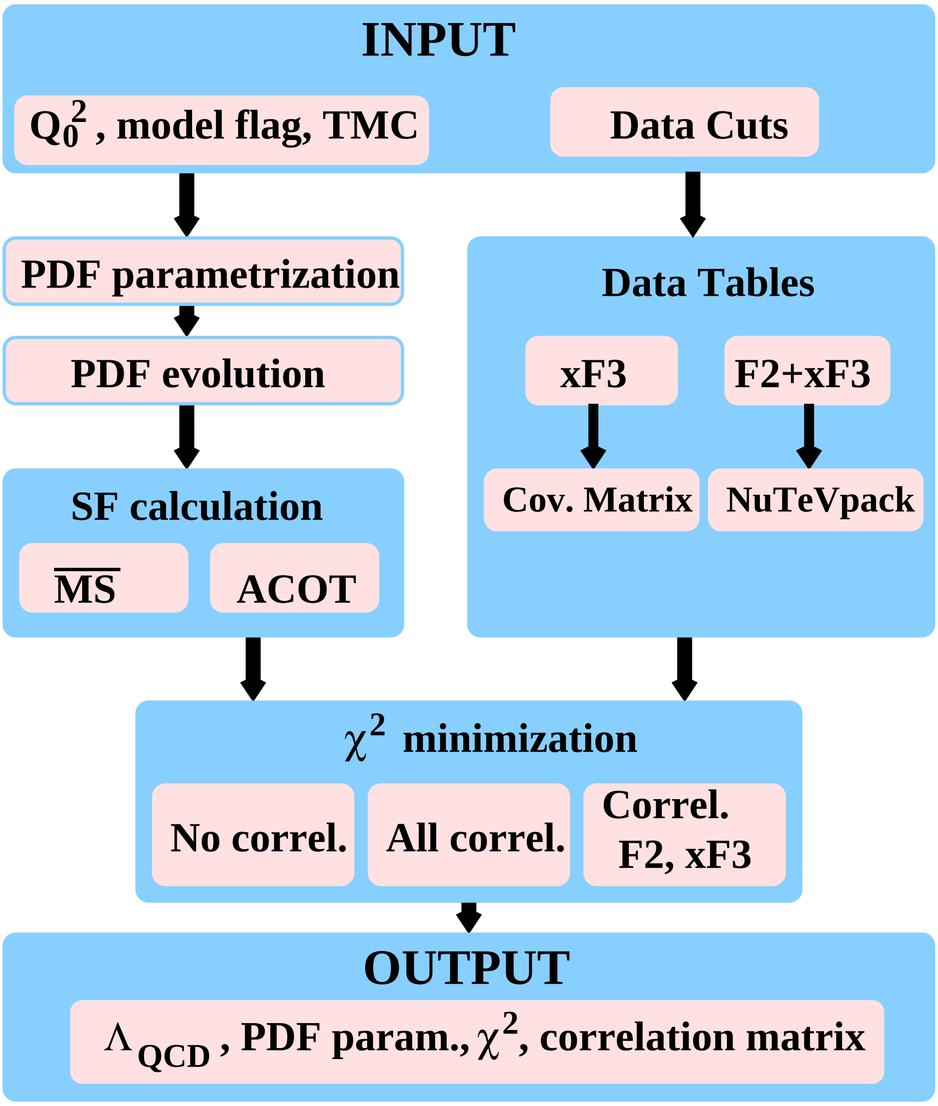


Figure 4.1: A Flow-chart of the NLO QCD fitting code.

4.1.1 PDF Parametrization and Evolution

An initial parametrization of the PDFs at a reference scale Q^2 is a required input to the QCD program. This reference scale is chosen here to be $Q_0^2 = 5$ GeV^2 to minimize the non-perturbative QCD effects. For the non-singlet fit we use the following functional form

$$xq^{NS} = xu_v + xd_v = (A0_{u_v} + A0_{d_v})x^{A1_{u_v}}(1-x)^{A2_{u_v}}, \tag{4.1}$$

where NS refers to non-singlet distribution. For the combined fit, we also need to input a parametrization for singlet and gluon distributions such as

$$\begin{aligned} xq^S &= \underbrace{xu_v + xd_v}_{xq^{NS}} + 2A0_{ud}(1-x)^{A2_{ud}}, \\ xG &= A0_g(1-x)^{A2_g}, \end{aligned} \tag{4.2}$$

where S refers to singlet distribution and G to gluon distribution. A more general parametrization was first tried, then the parameters for which the fit was not sensitive to were removed from the fits after initial studies (this is discussed further in Section 4.3.1).

The PDFs must satisfy the QCD sum rules, which were described in Section 1.3.3. However, NuTeV data covers a limited x range. Therefore, it does not directly constrain the PDFs via sum rules. Consequently, one must choose either to impose the QCD sum rules on the fitting function or to check if the fitted PDFs satisfy the sum rules. This analysis imposes the QCD sum rules on the fitting functions. As a result, the normalization parameters $A0_{u_v}$, $A0_{d_v}$, and $A0_g$ are constrained by these rules. For example, Eq. (1.77) constrains $A0_{u_v}$, $A0_{d_v}$, and Eq. (1.78) constrains $A0_g$. Thus, the only free parameters for the non-singlet fit are $A1_{u_v}$, $A2_{d_v}$, Λ_{QCD}, while for the combined fit these are $A1_{u_v}$, $A2_{d_v}$, $A0_{ud}$, $A2_{ud}$, $A2_g$, Λ_{QCD}. The Λ_{QCD} parameter enters into the fit as a free parameter via evolution Equations (1.74)-(1.76). In addition, there is an overall normalization parameter, "*norm*", which is allowed to float freely.

Once the functional forms for PDFs are defined, the initial PDFs are evaluated at a reference point $Q_0^2 = 5$ GeV^2 with the initial values for the fit parameters ranging between 0.8 and 5, with $\Lambda_{QCD} = 337$ MeV. The fit does not depend on the choice of starting values. Then the parametrized PDFs are evolved following the steps below:

- Apply evolution Equations (1.74), (1.75), (1.76) to the initial PDFs to obtain $\partial q(x, s)/\partial s$ at $Q^2 = Q_0^2$ (i.e. $s = 0$, where $s = \ln[\ln(Q^2/\Lambda^2)/[\ln(Q_0^2/\Lambda^2)]]$).
- Use a predictor-corrector algorithm to estimate the PDFs for incrementally increased s, $q(x, s + \delta s)$ [55].
- Repeat until the whole kinematic range is covered.

The evolved PDFs are then stored in a grid in (x_i, s_i) and the values of the PDFs at any arbitrary (x, s) are obtained via a linear interpolation in x, s from the grid. These PDFs are used to calculate the NLO structure functions.

4.1.2 QCD Models

The NLO QCD models used in this analysis ($\overline{MS}$ and $ACOT$) were presented in Section 4.1.2. The NLO structure functions are obtained from convolution of the parametrized PDFs with the hard scattering coefficients. Since neutrino scattering is sensitive to heavy quark production, the primary result of this measurement uses the $ACOT$ model, while the $\overline{MS}$ model is only used to observe the differences between the massless and finite quark mass treatments. Throughout this analysis the following values for quark masses are used: for bottom quark $m_b = 4.3 \pm 0.2$ GeV (taken from *2005 Review Particle Listing* [4]) and for charm quark $m_c = 1.4 \pm 0.2$ GeV (obtained from a weighted average of LO experimental measurements [39], [48]). The latter one is chosen to be consistent with the extraction method of the differential cross section (see Sect. 3.2.1.2). Also, the CKM[1] matrix elements

---

[1] CKM stands for Cabibbo-Kobayashi-Maskawa. CKM is the matrix that relates the quark mass eigenstates to the weak eigenstates

used in the program are the average values from Ref. [4],

$$V = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} = \begin{pmatrix} 0.97485 & 0.2225 & 0.00365 \\ 0.22250 & 0.9740 & 0.04100 \\ 0.00900 & 0.0405 & 0.99915 \end{pmatrix}. \qquad (4.3)$$

4.1.2.1 Target Mass Correction

The QCD models do not account for target mass effects. Hence, the models should be corrected to fit data which includes these effects. Target mass effects are only important at low Q^2 and high x, as described in Sect. 1.3.4.1. For the *ACOT* model, the corrections are implemented in the QCD program for the structure functions $F_2(x, Q^2)$ and $F_3(x, Q^2)$ using Eqs. (1.80) and (1.81). Since the $\overline{MS}$ model returns in the program $xF_3(x, Q^2)$ instead of $F_3(x, Q^2)$, Eq. (1.81) is adjusted as follows:

$$\begin{aligned} xF_3^{TM} &= x\frac{x}{\xi}\frac{F_3(\xi)}{k^2} + x\frac{2M_p^2x^2}{Q^2k^3}\int_\xi^1 \frac{F_3(u)}{u}du; \\ xF_3^{TM} &= \left(\frac{x}{\xi}\right)^2\frac{\xi F_3(\xi)}{k^2} + \frac{2M_p^2x^3}{Q^2k^3}\int_\xi^1 \frac{uF_3(u)}{u^2}du. \end{aligned} \qquad (4.4)$$

The effect of including the target mass correction on the measurement of Λ_{QCD}, in the NuTeV kinematic range, was studied (see Section 4.3.4) and was found to be negligible.

4.1.3 χ^2 Minimization

In this analysis the minimization of the χ^2 is based on the Marquardt fitting algorithm [56]. It behaves like a gradient search far from the minimum and like a linear least squares near the minimum. The fitting program allows a treatment for the correlations between F_2 and xF_3, as well as for the point-to-point correlations. We discuss here three possibilities.

4.1.3.1 No Correlations Case

This method is used for the non-singlet fit when only statistical errors are taken into account, and also when the weight of each contributing systematic error is evaluated separately. The χ^2 for this case is

$$\chi^2 = \sum_i \frac{1}{\sigma_i} \left(F_i^{DATA} - F^{TH}(x_i) \right)^2 , \tag{4.5}$$

where F_i^{DATA} corresponds to xF_3 data points, σ_i to xF_3 statistical errors, and $F^{TH}(x_i)$ to theoretical xF_3 which depends on the fit parameters associated with PDF parametrization. The index i is over the total number of data points (i.e., $i = 1, ..., 63$). The minimization process uses the linearized function obtained by Taylor expansion of $F^{TH}(x_i)$ to first order as,

$$F^{TH}(x_i) = F'(x_i) + \sum_{j=1}^{m} \left(\frac{\partial F'(x_i)}{a_j} \delta a_j \right) , \tag{4.6}$$

where index j runs over the number of fit parameters. Hence, the χ^2 can be re-written as

$$\chi^2 = \sum_i \left(\frac{1}{\sigma_i^2} \left[F_i^{DATA} - F'(x_i) - \sum_{j=1}^{m} \left(\frac{\partial F'(x_i)}{a_j} \delta a_j \right) \right]^2 \right) . \tag{4.7}$$

The derivative of the χ^2 is computed for each parameter increment, δa_j. A set of equations is then obtained:

$$\frac{\partial \chi^2}{\partial \delta a_k} = -2 \sum_i \left(\frac{1}{\sigma_i^2} \left[F_i^{DATA} - F'(x_i) - \sum_{j=1}^{m} \left(\frac{\partial F'(x_i)}{a_j} \delta a_j \right) \right] \frac{\partial F'(x_i)}{\partial a_k} \right) = 0. \tag{4.8}$$

The results can be expressed in matrix elements form as

$$\beta_k = \delta a_j \; \alpha_{jk}, \tag{4.9}$$

where

$$\begin{aligned} \beta_k &= \sum_i \left(\frac{1}{\sigma_i^2} \left[F_i^{DATA} - F'(x_i) \right] \frac{\partial F'(x_i)}{\partial a_k} \right) , \\ \alpha_{jk} &= \sum_i \left(\frac{1}{\sigma_i^2} \frac{\partial F'(x_i)}{\partial a_j} \frac{\partial F'(x_i)}{\partial a_k} \right) . \end{aligned} \tag{4.10}$$

α_{jk} represents the symmetric curvature matrix.

In Marquardt algorithm the interpolation between gradient search and analytical methods is done by adding to the diagonal terms of the curvature matrix (α_{jk}) a factor $1+\lambda$, as follows:

$$\alpha'_{jk} = \begin{cases} \alpha_{jk}(1+\lambda) & \text{for } j = k; \\ \alpha_{jk} & \text{for } j \neq k. \end{cases} \tag{4.11}$$

The solution for each j parameter increment δa_j is obtained as

$$\delta a_j = \sum_{k=1}^{m} (\beta_k \epsilon'_{jk}), \tag{4.12}$$

where the error matrix $\boldsymbol{\epsilon} = \boldsymbol{\alpha}^{-1}$ is the inverse of the curvature matrix. If λ is very small the solution is similar to the analytical case described above. If λ is very large, then the diagonal terms dominate and the matrix equation degenerates into m separate equations:

$$\beta_j = \lambda \delta a_j \alpha_{jj}. \tag{4.13}$$

The uncertainty in parameter a_j corresponds to $\sigma_{a_j} = \epsilon_{jj}$.

The minimization recipe given by Marquardt [56, 55] is applied in the program as

1. Calculate $\chi^2(a)$ with $\lambda = 0.001$.
2. Compute δa and $\chi^2(a+\delta a)$.
3. If

$$\begin{cases} \chi^2(a+\delta a) > \chi^2(a) & \text{then } \lambda \to 10 \times \lambda; \\ \chi^2(a+\delta a) < \chi^2(a) & \text{then } \lambda \to \frac{\lambda}{10} \text{ and } a \to a' = a+\delta \text{ go to step 1 with } \lambda = \lambda/10. \end{cases}$$

4.1.3.2 Point-to-Point Correlations Case

The minimization method for this case is similar to the uncorrelated case. However, here the point-to-point correlations are taken into account by constructing the structure function covariance error matrix (discussed in Section 4.2.1). The method is only applied for the non-singlet fit because for the combined fit, the correlations between F_2 and xF_3 must also be considered (see Section 4.1.3.3). The general form of the χ^2 that includes the full covariance matrix is

$$\chi^2 = \sum_i \sum_j \left(F_i^{DATA} - F^{TH}(x_i)\right) V_{ij}^{-1} \left(F_j^{DATA} - F^{TH}(x_j)\right), \tag{4.14}$$

where both indices $i,\ j$ run over the total number of data points (i.e. $i, j = 1, ..., 63$) and F_i^{DATA} and F_i^{TH} have the same meaning as for the previous case. In the following derivations it has been assumed that the covariance matrix is symmetric (i.e. $V_{ij}^{-1} = V_{ji}^{-1}$). The fitting function is linearized using a first order Taylor expansion, so that

$$\begin{aligned} \chi^2 \; = \; & \sum_{i,j} \left(F_i^{DATA} - F'(x_i) - \sum_m \frac{\partial F'(x_i)}{\partial a_m} \delta a_m \right) V_{ij}^{-1} \\ & \left(F_j^{DATA} - F'(x_j) - \sum_n \frac{\partial F'(x_j)}{\partial a_n} \delta a_n \right), \end{aligned} \tag{4.15}$$

with indices m and n corresponding to fit parameters. Then the χ^2 is minimized as follows:

$$\frac{\partial \chi^2}{\partial \delta a_k} = 0; \tag{4.16}$$

$$\begin{aligned} \sum_{i,j} \Bigg(& -\frac{\partial F'(x_i)}{\partial a_k} V_{ij}^{-1} \left(F_j - F'(x_j) - \sum_n \frac{\partial F'(x_j)}{\partial a_n} \delta a_k \right) \\ & - \left(F_i - F'(x_i) - \sum_m \frac{\partial F'(x_i)}{\partial a_m} \delta a_k \right) V_{ij}^{-1} \frac{\partial F'(x_j)}{\partial a_k} \Bigg) = 0. \end{aligned} \tag{4.17}$$

We define β_k and α_{kn} similarly to the previous case as,

$$\begin{aligned} \beta_k &= \sum_{i,j} \left(F_i - F'(x_i)\right) V_{ij}^{-1} \frac{\partial F'(x_j)}{\partial a_k}, \\ \alpha_{kn} &= \sum_{i,j} \frac{\partial F'(x_i)}{\partial a_k} V_{ij}^{-1} \frac{\partial F'(x_j)}{\partial a_n}. \end{aligned} \tag{4.18}$$

Then, Eq. (4.17) can be rewritten as

$$-2\beta_k + \sum_n \alpha_{kn}\delta a_n + \sum_m \alpha_{mk}\delta a_m = 0, \tag{4.19}$$

and if we multiply the right hand side by the inverse of $\boldsymbol{\alpha}$, $\boldsymbol{\epsilon} = \boldsymbol{\alpha}^{-1}$ we obtain,

$$2\sum_k \beta_k \epsilon_{kl} = \sum_n \alpha_{kn}\alpha_{kl}^{-1}\delta a_n + \sum_m \alpha_{km}\alpha_{kl}^{-1}\delta a_m. \tag{4.20}$$

For a symmetric covariance error matrix (i.e. $V_{ij}^{-1} = V_{ji}^{-1}$) $\alpha_{kn} = \alpha_{nk}$ and

$$\begin{aligned} 2\sum_k \beta_k \epsilon_{kl} &= \sum_n \delta_{ln}\delta a_n + \sum_m \delta_{lm}\delta a_m \\ 2\sum_k \beta_k \epsilon_{kl} &= 2\delta a_l. \end{aligned} \tag{4.21}$$

Therefore, the solution for the parameter increment δa_l has the same form as one for the uncorrelated case,

$$\delta a_l = \sum_k \beta_k \epsilon_{kl}. \tag{4.22}$$

4.1.3.3 Correlation between F_2 and xF_3 Case

In this case, one takes into account the correlations between F_2 and xF_3 which were obtained by fitting the same set of differential cross section points. This method only applies for the combined fit. The χ^2 for this case is expressed as

$$\chi^2 = \sum_{i=1}\left(\sum_{l=1}^{2}\sum_{m=1}^{2}\left(F_l^{i(DATA)} - F_l^{i(TH)}\right)(V_{lm}^i)^{-1}\left(F_m^{i(DATA)} - F_m^{i(TH)}\right)\right). \tag{4.23}$$

The indices l, m are over the number structure functions F_2 and xF_3, i.e. $l, m = 1$ corresponds to structure function F_2, while $l, m = 2$ to xF_3. The i index is over the total number of points ($i = 1, ..., 132$). The covariance matrix at each x, Q^2 point is given by

$$V_{lm}^i = \begin{pmatrix} \delta F_2^2 & \delta F_2\ \delta xF_3 \\ \delta xF_3\ \delta F_2 & \delta xF_3^2 \end{pmatrix}. \tag{4.24}$$

The diagonal elements contain the total error (statistical and systematic) from differential cross section points (described in Sect. 3.6.3 and in Ref. [22]). Additional systematic errors due to model inputs (R_L and ΔxF_3) in extracting the structure functions (described in Section 3.6.2) are added in quadrature to the diagonal elements because they are independent of the cross section uncertainties. The cross terms, $\delta F_2\ \delta xF_3$, give the correlated errors between F_2 and xF_3.

The minimization of the χ^2 is performed similarly to the previous cases, only that, for this case, β and α are defined as

$$\begin{aligned}
\beta_k &= \sum_{i=1} \left(\sum_{l,m=1}^{2} \left(F_l^i - F'(x_l^i)\right) (V_{lm}^i)^{-1} \frac{\partial F'(x_m^i)}{\partial a_k} \right), \\
\alpha_{kn} &= \sum_{i=1} \left(\sum_{l,m=1}^{2} \frac{\partial F'(x_l^i)}{\partial a_k} (V_{lm}^i)^{-1} \frac{\partial F'(x_m^i)}{\partial a_n} \right).
\end{aligned} \tag{4.25}$$

4.2 Λ_{QCD} FROM NLO QCD FITS TO STRUCTURE FUNCTIONS

Here we discuss the results from non-singlet and combined fits. In order to minimize the non-perturbative QCD effects only data points with $Q^2 > 5$ GeV^2 and $W^2 > 10$ GeV^2 are used in the fits. Our data is in the kinematic range of $0.001 < x < 0.8$ and $5 < Q^2 < 125$ GeV^2. Fixed-target DIS experiments are not sensitive to bottom quark excitation, therefore a Λ_{QCD} for four active flavors is quoted.

4.2.1 Structure Function Covariance Matrix

In order to perform non-singlet fits taking into account the systematic errors, a covariance error matrix for xF_3 is constructed. The systematic uncertainties for the non-singlet fit originate from the extracted differential cross section data points (see Sec. 3.6.1). Seven sources of systematic errors are considered: the muon and hadron energy scales, the smearing models for the muon and hadron energy distributions, and the flux uncertainties which include charm mass and $\frac{B}{A}$ correction uncertainties (see Section 3.5 for details) [22].

To obtain the covariance matrix for xF_3 each systematic is varied by $\pm 1\sigma$ (or by using a model difference in some cases, for example, muon energy smearing models). xF_3 is then re-extracted for each shift by going through the whole processes of fitting the differential cross section with each shifted systematic. As a result, one obtains xF_3 data tables corresponding to each variation of the systematic. Hence, the structure function covariance error matrix is constructed from these shifts as

$$V_{ij} = \sum_{k=1}^{7} \delta_{k|i}\delta_{k|j}. \tag{4.26}$$

The sum is over all considered systematic shifts (i.e. $k = 1, ..., 7$) and $\delta_{k|i}$ corresponds to 1σ shifts in the data point i due to systematic uncertainty k. In other words,

$$\delta_{k|i} = \frac{F_k^+(i) - F_k^-(i)}{2}, \tag{4.27}$$

where $F^+(F^-)$ denotes the extracted structure functions from the fits to the cross sections shifted by $+1\sigma(-1\sigma)$. The statistical errors are added in quadrature to the diagonal elements of the covariance matrix. The xF_3 data table with its contributing systematic uncertainties are appended in Appendix B.1.

For the combined fit the structure functions F_2 and xF_3 were simultaneously extracted from the differential cross section using the differential cross section covariance matrix as described in Sec 3.6.2). As a result, all information about the contributing systematic uncertainties is inglobed into one single error. These data tables are provided in Appendix B.2.

4.2.2 Non-Singlet Fit Results

Table 4.1 shows non-singlet fit results using only statistical errors. The results are presented for both $\overline{MS}$ and $ACOT$ models. We notice that the inclusion of the quark masses into calculations ($ACOT$ model) causes a slight increase in the value for Λ_{QCD}.

Table 4.1: Non-Singlet QCD fit results with statistical errors only.

Parameter	$\overline{MS}$	ACOT
$\Lambda^{(n_f=4)}$(MeV)	442 ± 58	454 ± 57
$A1_{u_v}$	0.73 ± 0.01	0.73 ± 0.01
$A2_{u_v}$	3.54 ± 0.05	3.53 ± 0.05
$A0_{u_v} + A0_{d_v}$	4.79+2.39	4.77+2.38
χ^2/dof	97/59	93/59

Table 4.2 shows fit results using the structure function covariance matrix that takes into account statistical and systematic errors. We observe that Λ_{QCD} increases by $\approx$ 35 units compared to the previous fit, while the goodness of the fit is improved, i.e. χ^2/dof decreases to 77/59 (with a probability of 5%). We also note that the overall error for Λ_{QCD} has changed

Table 4.2: Non-Singlet QCD fit results with all systematic errors.

Parameter	$\overline{MS}$	ACOT
$\Lambda^{(n_f=4)}$(MeV)	476 ± 60	488 ± 59
$A1_{u_v}$	0.73 ± 0.01	0.73 ± 0.01
$A2_{u_v}$	3.47 ± 0.06	3.47 ± 0.06
$A0_{u_v} + A0_{d_v}$	4.74+2.37	4.73+2.36
χ^2/dof	78/59	77/59

little. Hence, there are cancellation effects among the systematic errors. This also might be emphasized by the fact that xF_3 is determined from the difference of the antineutrino to neutrino cross sections. This method of evaluating the Λ_{QCD} and its error does not provides us information of the weight of each systematic. The effects of each contributing systematic is discussed in Section 4.2.5.

Figure 4.2 shows the non-singlet fit results as function of $\log Q^2$ in all x bins using both theory models. The QCD NLO curves fit well the NuTeV data. The two theory models are nearly indistinguishable. Scaling violations are clearly seen in the deviations from flatness of the curves for different x bins.

4.2.3 Combined F_2 and xF_3 Fit Results

The combined fit results using only statistical errors are shown in Table 4.3 for both $\overline{MS}$ and $ACOT$ models. We observe that the inclusion of quark masses increases the value of Λ_{QCD}.

Table 4.4 shows the combined fit results which include systematic uncertainties via differential cross section error matrix. Similarly to the non-singlet case, we observe that Λ_{QCD} increases when we take into account the systematic errors (by $\approx$ 25 units). Also, the χ^2 decreases when F_2 and xF_3 are simultaneously extracted using the differential cross section covariance matrix, because the errors increase by a factor ≈ 3 compared to the case when only statistical errors are used. This increase causes χ^2 to decrease by a factor ≈ 9 when the systematic errors are taken into account ($\chi^2 \propto 1/\sigma^2$). Also, the uncertainty on Λ_{QCD} increases by a factor two when the systematic uncertainties are taken into account. We speculate that F_2 dominates, being determined from the sum of neutrino and anti-neutrino cross sections.

Figures 4.3, 4.4 show the NLO QCD fit results for F_2 and xF_3 as function of $\log Q^2$ in all x bins. Both models fit well NuTeV data (with statistical and systematic uncertainties).

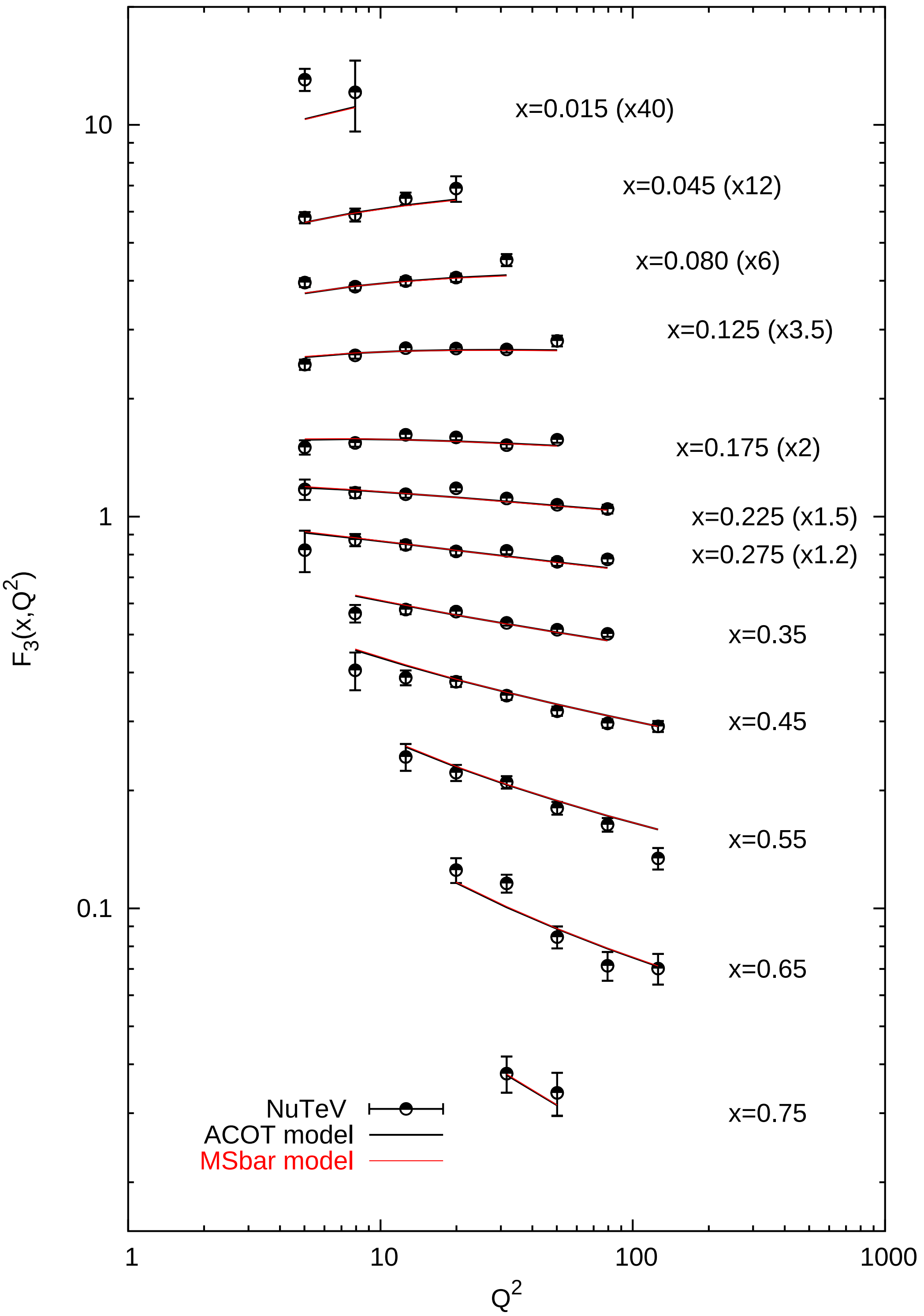


Figure 4.2: xF_3 non-singlet fit using $ACOT$ and $\overline{MS}$ models. Points are NuTeV data, curves show fits. Error bars contain both statistical and systematical uncertainties.

Table 4.3: Combined QCD fit results; only statistical errors.

Parameter	$\overline{MS}$	ACOT
$\Lambda^{(n_f=4)}$(MeV)	410 ± 23	434 ± 22
$A1_{u_v}$	0.74 ± 0.01	0.75± 0.01
$A2_{u_v}$	3.55 ± 0.02	3.56± 0.02
$A0_{u_v} + A0_{d_v}$	4.88+2.44	4.97+2.49
$A0_{ud}$	0.71 ± 0.01	0.71± 0.01
$A2_{ud}$	6.89 ± 0.08	7.06± 0.08
$A0_g$	2.35	2.09
$A2_g$	4.88 ± 0.64	4.23 ± 0.47
χ^2/dof	562/125	548/125

The curves correspond to the theory models which are very similar.

4.2.4 Comparison between Non-Singlet and Combined Fits

In this Section we compare the results between non-singlet and combined fits. We observe that Λ_{QCD} obtained from both non-singlet and combined fits is consistent with each other. For the combined fit the value of Λ_{QCD} and the shape of the gluon distribution are strongly correlated (see Appendix F.1.2 for tables of correlation among fit parameters). The statistical precision of Λ_{QCD} from the combined fit improves by a factor two compared to the non-singlet fit. This is mainly due to the inclusion of F_2 data set in the combined fit. Also, the size of the errors for structure functions used in the combined fit are larger than these used in the non-singlet fit. This explains the difference in the goodness of the fit between these two types of fits. For example, we compare the χ^2 value of non-singlet fit using data file from

Table 4.4: Combined QCD fit results; all systematic errors included.

Parameter	$\overline{MS}$	ACOT
$\Lambda^{(n_f=4)}$(MeV)	433 ± 51	458 ± 41
$A1_{u_v}$	0.71 ± 0.02	0.72± 0.02
$A2_{u_v}$	3.50 ± 0.05	3.49± 0.05
$A0_{u_v} + A0_{d_v}$	4.47+2.23	4.50+2.25
$A0_{ud}$	0.68 ± 0.03	0.67± 0.03
$A2_{ud}$	6.74 ± 0.20	6.83± 0.21
$A0_g$	2.42	2.21
$A2_g$	4.83 ± 1.38	4.30 ± 0.41
χ^2/dof	79/125	76/125

1-parameter fit, which is $\chi^2/dof = 97/59$ (statistical errors only), to $\chi^2/dof = 173/62$ from the non-singlet fit using data file from 2-parameter fit (statistical errors).

4.2.5 Λ_{QCD} Systematic Uncertainties

In this Section we discuss the effect of each contributing experimental systematic uncertainty on Λ_{QCD}. This is determined by taking each differential cross section shifted by $\pm 1\sigma$, re-extracting the structure functions from these shifted cross sections, and determining Λ_{QCD} for each shift. Therefore, we consider the "effective" systematic uncertainty on Λ_{QCD} by

$$\Lambda(sys) = \frac{\Lambda(+) + \Lambda(-)}{2}, \tag{4.28}$$

where $\Lambda(\pm)$ denotes Λ_{QCD} extracted for each cross-section systematic shifted up and down by 1σ. Because most of these errors are of the size of statistical fluctuations we cannot

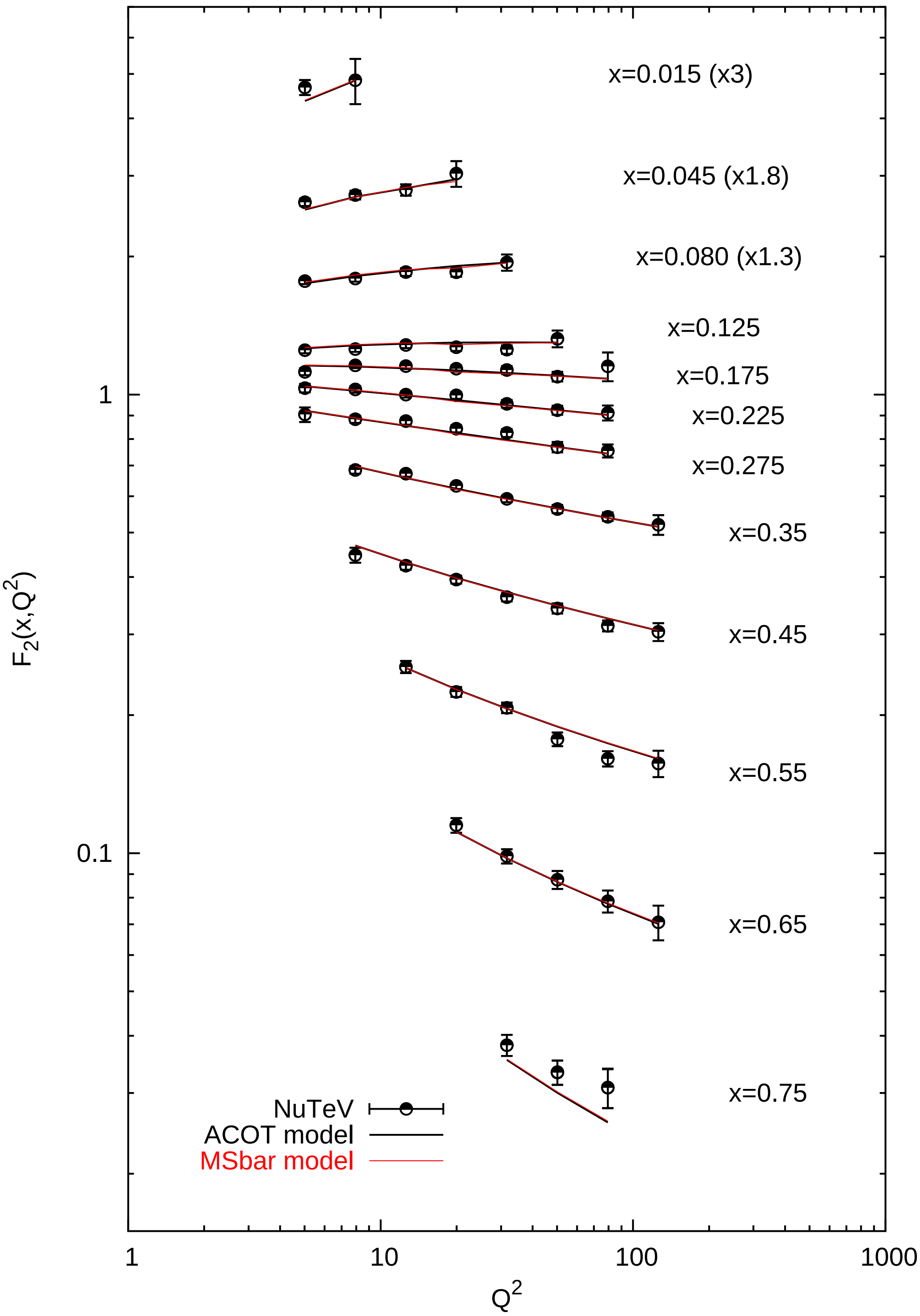


Figure 4.3: F_2 from combined fit using $ACOT$ and $\overline{MS}$ models; both statistical and systematic uncertainties included

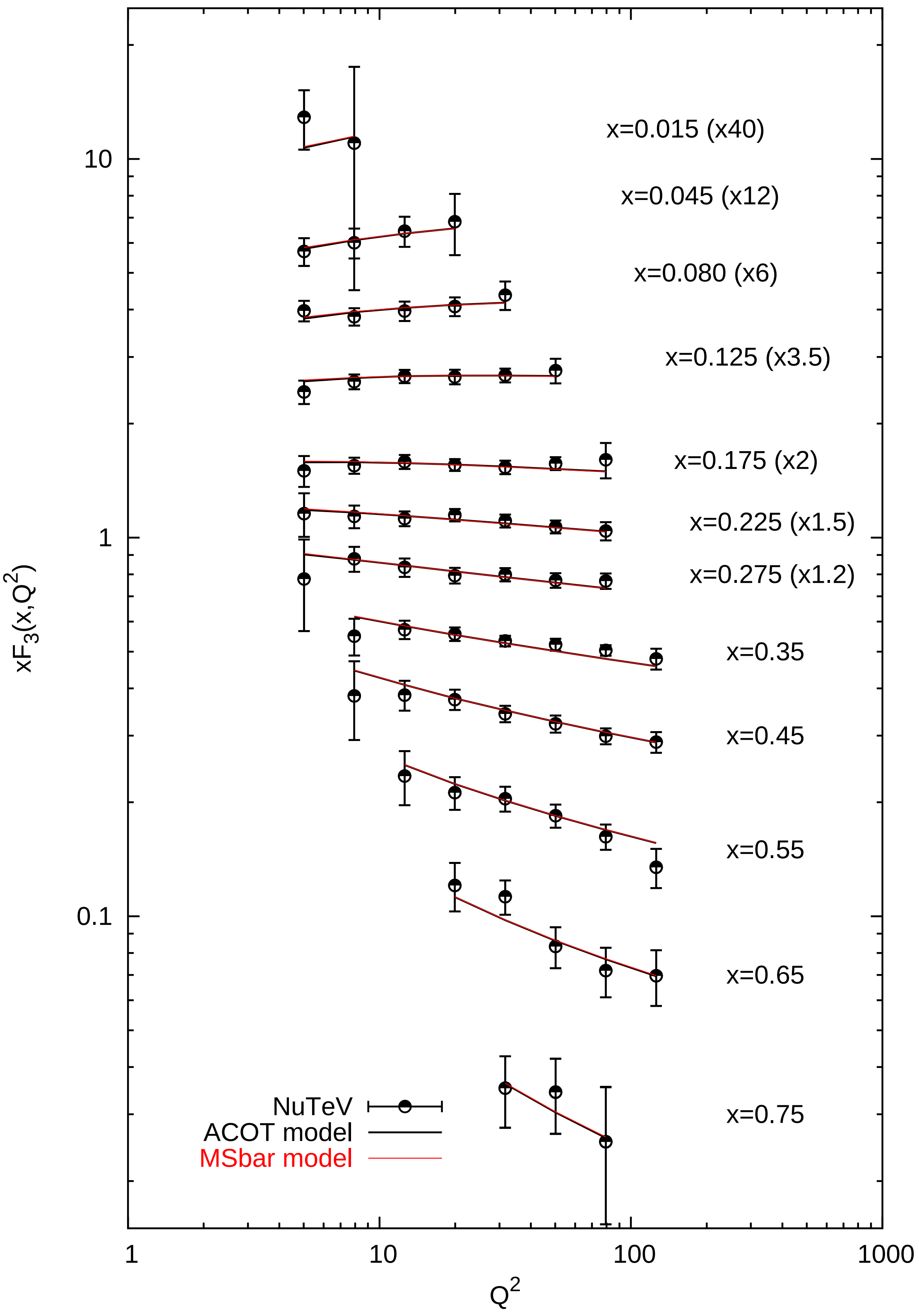


Figure 4.4: xF_3 from combined fit using $ACOT$ and $\overline{MS}$ models; both statistical and systematic uncertainties included

accurately determine their effect. Therefore, we repeat the above procedure for ±2σ shifts of each systematic (for our default model, *ACOT*), whenever is possible. If a 2σ shift gives the same order for the error we assume it is a negligible contribution. The results for non-singlet fit are listed in the Table 4.5. The largest experimental systematic uncertainties are due to the energy scales, and the energy smearing effects in our simulation. Other experiments systematic contribution are negligible.

Table 4.5: Non-Singlet fit results showing the effect of each contributing systematic by varying each $\pm 1\sigma$, $\pm 2\sigma$ .Both massless and massive predictions are shown.

	$\overline{MS}$		ACOT			
$\Lambda_{Central} \pm stat$ (MeV)	442 ± 58		454 ± 57			
χ^2/dof	97/59		93/59			
Systematic	$+1\sigma$	-1σ	$+1\sigma$	-1σ	$+2\sigma$	-2σ
E_μ scale	+67	−53	+76	−38	+130	−92
E_{had} scale	−23	+46	−10	+65	−33	+109
E_μ smear model	−37		−28		−66	
E_{had} smear model	−5	−9	+3	0	−10	−16
m_c	−20	−5	+1	0	−10	0
$\frac{B}{A}$	−10	−7	−6	+4	−6	+3

Results of the systematic error analysis for the combined fit are shown in in Table 4.6. Additional systematic contributions due to the model inputs, ΔxF_3 and R_L (see Sect. 3.6.3) The largest systematic errors are again due to the energy scales and the smearing models and are of the order of statistical precision. We note that the systematic errors due to the flux uncertainties, in particular charm mass (see Section 3.5), has more weight for *ACOT* model than for $\overline{MS}$ model.

Table 4.6: Combined QCD fit results showing the effect of each contributing systematic. Both massless and massive predictions are shown.

	$\overline{MS}$		ACOT			
$\Lambda_{Central}$ (MeV)	410 ± 23		434 ± 22			
χ^2/dof	562/125		548/125			
Systematic	$+1\sigma$	-1σ	$+1\sigma$	-1σ	$+2\sigma$	-2σ
E_μ scale	+15	−33	+21	−8	+38	−5
E_{had} scale	−19	+13	−10	+26	−19	+51
E_μ smear model	−23		−22		−35	
E_{had} smear model	−16	+12	−13	+8	−9	+39
m_c	+1	−15	+28	−15	+18	−12
$\frac{B}{A}$	+14	−16	+2	−11	+39	−20
R_W	+12	−12	+8	−13	+16	−27
ΔxF_3 model	+25		+31		–	

4.2.6 Theoretical Uncertainties

A hard scattering calculation introduces a renormalization scale to remove the ultraviolet divergences (see Sect.1.3.1.1), and a factorization scale that separates the hard from soft physics (see Sect. 1.3.1.5). The scale dependence which arises represents the main theoretical uncertainty in the NLO DIS measurements of Λ_{QCD}. We also evaluate the systematic uncertainties associated with quark masses. These are found to have small effects.

4.2.6.1 Scale Dependence

Factorization and renormalization scales (μ_F, μ_R, respectively) are both usually chosen to be Q, the characteristic energy scale in the process. In this standard convention various logarithms in the calculations vanish. This program is using $\mu_R = \mu_F = Q$ as well, and thus does not include the vanishing logs explicitly in the program. In order to properly account for the scale dependence, these contributions need to be re-introduced in code.

The scale dependencies appear in the general form for structure functions as follows:

$$F(x, Q^2) = \sum_i C_i(\mu_R, \mu_F, Q^2, y) \otimes f_i(z, \mu_F), \tag{4.29}$$

where $C_i(\mu_R, \mu_F, Q^2, y)$ are the coefficient functions that depend both on renormalization and factorization scales, and $f_i(z, \mu_F)$ are the PDFs which depend on the factorization scale. The physical structure functions should not depend on the choices made for μ_F or μ_R. If calculations could be carried to infinite order in the perturbation series, the scale dependencies would cancel out, but for a truncated order calculation, scale dependencies remain. The calculations in this analysis are performed to NLO in α_S, therefore

$$\begin{aligned} \mu_R \frac{\partial F}{\partial \mu_R} &= 0 \text{ up to } \mathrm{O}(\alpha_S^2); & (4.30) \\ \mu_F \frac{\partial F}{\partial \mu_F} &= 0 \text{ up to } \mathrm{O}(\alpha_S^2). & (4.31) \end{aligned}$$

The general form for the NLO structure function, $F(x, Q^2)$ can be written by expanding the hard scattering coefficients up to NLO as follows:

$$F(x, Q^2) = \sum_i \left[C_i^0(\mu_R, \mu_F, Q^2, y) + \alpha_S(\mu_R) C_i^1(\mu_R, \mu_F, Q^2, y)\right] \otimes f_i(z, \mu_F). \tag{4.32}$$

Then,

$$\mu_R \frac{\partial F}{\partial \mu_R} = \sum_i \left[\underbrace{\mu_R \frac{\partial C_i^0}{\partial \mu_R}}_{O(\alpha_s^0)} + \underbrace{\mu_R \frac{\partial \alpha_S}{\partial \mu_R} C_i^1}_{O(\alpha_s^2)} + \underbrace{\alpha_S \frac{\partial C_i^1}{\partial \mu_R}}_{O(\alpha_s^1)} \right] \otimes f_i = 0 \Longrightarrow \alpha_S \frac{\partial C_i^1}{\partial \mu_R} = 0, \tag{4.33}$$

where the C_i^0 has no μ_R dependence. This means that μ_R enters in the NLO term of the coefficient functions, but at this order there are no compensating μ_R terms to cancel it out.

Performing a similar analysis for the factorization scale using Eq. (4.31), one obtains canceling terms in the coefficient functions C_i^1, i.e. $\frac{C_q^0}{2\pi} P_{qq}(P_{qg}) \ln \frac{Q^2}{\mu_F^2}$. In these logarithmic terms, for $\mu_F = Q$ cancellations occur up to $\mathcal{O}(\alpha_S^2)$, but not for an arbitrary choice of μ_F. The method of evaluating the scale dependence is by its nature imprecise due to the subjectivity of defining the reasonable range over which the scale is varied. Currently there is no preferred method to evaluate the theoretical uncertainty due to scale dependence in finite order calculations. To get a range of uncertainty the choice of factorization scale is varied up and down by a factor of two (conform Ref. [4]), while the renormalization scale is kept to $\mu_R = Q$. The results of this study are shown in Table 4.7 for the $\overline{MS}$ model (results should be similar for *ACOT* model).

Table 4.7: Factorization scale dependence for $\overline{MS}$ model.

	xF_3 fit		xF_3 and F_2 fit	
C_i	$\Lambda^{(n_f=4)}$ (MeV)	χ^2/dof	$\Lambda^{(n_f=4)}$ (MeV)	χ^2/dof
1	476 ± 60	78/59	433 ± 51	79/125
2	+74	79/59	+61	75/125
0.5	−113	53/44	−87	47/93

We evaluate a theoretical uncertainty for Λ_{QCD} due to the factorization scale dependence to be $^{+74}_{-113}$ MeV for the non-singlet fit. For the combined fit, the theoretical uncertainty for Λ_{QCD} due to the factorization scale dependence is estimated to be $^{+61}_{-87}$ MeV. Since this is the dominant uncertainty in the measurement one expects that precision will improve with improvement in theoretical predictions.

4.2.6.2 Quark Masses Dependence

The quark mass dependence could arise in the massless model ($\overline{MS}$ model) only at the matching conditions in α_S when a heavy quark threshold is crossed. The variation of the charm and bottom quark masses could introduce small errors. We consider effects due to charm and bottom quark masses separately.

- **Charm mass uncertainty:** For $Q^2 > 5$ GeV^2 we are above the threshold for which we would consider only three number of active flavors, $n_f = 3$. Therefore, there is no charm mass dependence for the $\overline{MS}$ model. For the $ACOT$ model, which explicitly takes into account the quark masses, we expect to see some dependence, especially for the combined fit (F_2 is sensitive to charm production). The effects of varying the quark mass by $\pm 1\sigma$ are shown in Table 4.8. Negligible effects on Λ_{QCD} are observed for non-singlet fit. The effects for combined fit are small. Even if we shift the quark masses by a factor of two the effects are still negligible for non-singlet fit, and small for combined fit.

Table 4.8: The effect of m_c varied by $\pm 1\sigma$ and by a factor two on Λ_{QCD}, for both non-singlet and combined fits.

		xF_3		$F_2 + xF_3$	
Model	m_c (GeV)	Λ (MeV)	χ^2/dof	Λ (MeV)	χ^2/dof
ACOT	1.4	488 ± 59	77/59	458 ± 41	76/125
ACOT	-1σ	-4	77/59	-2	76/125
ACOT	$+1\sigma$	$+2$	77/59	$+10$	76/125
ACOT	0.7	-2	77/59	-6	77/125
ACOT	2.8	$+2$	77/59	$+18$	80/125

- **Bottom mass uncertainty:** Both models should see some dependence on the uncertainty in the bottom quark mass. Table 4.9 shows the result of varying the b-threshold within 1σ of its central value. The effects are negligible for both theory models.

  We could also assess the theoretical uncertainty due to quark mass effects by looking at

Table 4.9: The effect of varying the bottom mass by $\pm 1\sigma$ for non-singlet and combined fits. The results are shown for both theory models.

		xF_3		$F_2 + xF_3$	
Model	m_b (GeV)	Λ (MeV)	χ^2/dof	Λ (MeV)	χ^2/dof
$\overline{MS}$	4.3	476 ± 60	78/59	433 ± 51	79/125
$\overline{MS}$	4.1	+5	78/59	−5	79/125
$\overline{MS}$	4.5	+1	78/59	+6	77/125
ACOT	4.3	488 ± 59	77/59	458 ± 41	76/125
ACOT	4.1	+1	76/59	−17	77/125
ACOT	4.5	+0	77/59	−12	76/125

the difference between *ACOT* and $\overline{MS}$ models. *ACOT* is a model that fully accounts for quark masses, while $\overline{MS}$ model assumes massless quarks. The difference in the value of Λ_{QCD} for non-singlet fit is +12 MeV corresponding to a difference in $\alpha_S(M_Z)$ of +0.0003. For the combined fit, the difference in the value of Λ_{QCD} is of +25 MeV, which corresponds to a positive shift of +0.0011 in $\alpha_S(M_Z)$ value. The effects of the quark mass are small compared to the scale uncertainty.

4.2.7 Relate Λ_{QCD} to α_S

Λ_{QCD} provides a parametrization of the scale dependence of α_S. Using Equation (1.58) we can relate our measurement of Λ_{QCD} to the appropriate α_S, which corresponds to 2-loop beta function expansion. Furthermore, in order to compare the values of α_S from various experiments, we must evolve them to a common scale. The standard choice for this reference scale is the mass of the neutral Z boson, M_Z. We obtained the following values for $\alpha_S(M_Z)$

using *ACOT* model

$$\begin{aligned}\text{NS fit: } \Lambda^{n_f=4}_{ACOT} &= 488 \pm \underbrace{59}_{\text{exp}} \underbrace{{}^{+74}_{-113}}_{th} \text{ MeV} \rightarrow \alpha_S(M_Z) = 0.1260 \pm 0.0028 \underbrace{{}^{+0.0034}_{-0.0050}}_{theory} \quad (4.34)\\ \text{CO fit: } \Lambda^{n_f=4}_{ACOT} &= 458 \pm \underbrace{41}_{\text{exp}} \underbrace{{}^{+61}_{-87}}_{th} \text{ MeV} \rightarrow \alpha_S(M_Z) = 0.1247 \pm 0.0020 \underbrace{{}^{+0.0030}_{-0.0047}}_{theory} \quad (4.35)\end{aligned}$$

where the first line corresponds to results from non-singlet fit, and the second line to results from combined fit.

4.3 RESULT BIAS STUDY

In this section we examine possible effects on the QCD results due to the various choices of PDF parametrization and kinematic cut dependencies. The effects of the target mass correction are also studied here. Various checks of the QCD code were performed which are presented in the Appendix G.

4.3.1 PDF Parametrization

Since it is possible that the choice of the PDF functional form could bias the QCD fit, we consider first a general form. For the non-singlet quark distributions the following general initial parametrization was used:

$$xq^{NS} = xu_v + xd_v = (A0_{u_v} + A0_{d_v})x^{A1_{u_v}}(1-x)^{A2_{u_v}}e^{A3_{u_v}x}(1+x)^{A5_{u_v}}. \quad (4.36)$$

For singlet and gluon distributions the starting general forms are given by

$$\begin{aligned} xq^S &= \underbrace{xu_v + xd_v}_{xq^{NS}} + 2A0_{ud}(1-x)^{A2_{ud}}e^{A3_{ud}x}(1+x)^{A5_{ud}},\\ xG &= A0_g(1-x)^{A2_g}e^{A3_gx}(1+x)^{A5_g}. \end{aligned} \quad (4.37)$$

The fit results using these forms are shown in Tables 4.11, 4.10. We observe that the parameters $A3_{u_v}$, $A3_{ud}$, $A3_g$, $A5_{u_v}$, $A5_{ud}$, and A_g are not well constrained by the fit regardless of what theory model is used. Therefore, a simplified functional forms for the non-singlet, singlet and gluon distributions are adopted, as

$$\begin{aligned} xq^{NS} &= xu_v + xd_v = (A0_{u_v} + A0_{d_v})x^{A1_{u_v}}(1-x)^{A2_{u_v}}, \\ xq^{S} &= \underbrace{xu_v + xd_v}_{xq^{NS}} + 2A0_{ud}(1-x)^{A2_{ud}}, \\ xG &= A0_g(1-x)^{A2_g}. \end{aligned} \tag{4.38}$$

Other PDF functional forms have been tried for both non-singlet and combined fits.

4.3.2 Kinematic Cut Dependence: x

NuTeV data ranges up to $x = 0.75$ where it is possible that Fermi motion effects are important. To understand how these may affect QCD fits, Table 4.12 shows the effect of removing the largest x bin on the measurement of Λ_{QCD} for both non-singlet and combined fits. We observe no significant dependence on including the highest x bin.

4.3.3 Kinematic Cut Dependence: Q^2

As discussed in section 1.3.4, at low Q^2, non-perturbative effects become important. Here we explore the possible effects on Λ_{QCD} due to the choice of minimum Q^2 in the fits. For this study we vary Q^2_{min} to be $5,\ 7,\ 10$ GeV^2 and use in the fit only data points with $Q^2 > Q^2_{min}$. The results of this study are shown in Table 4.13. For the combined fit we see no effect if we increase the Q^2 cut, while for the non-singlet fit we notice a small increase in Λ_{QCD} which is well within the uncertainty. Therefore, for the final results we choose $Q^2 > 5$ GeV^2.

Table 4.10: QCD fit results using $\overline{MS}$ model for the combined fit using a general and a simplified parametrization of the PDFs.

Parameters	General	Simple
$\Lambda^{(n_f=4)}$(MeV)	383 ± 55	433 ± 51
$A0_{u_v} + A0_{d_v}$	4.13+2.06	4.47+2.23
$A1_{u_v}$	0.70 ± 0.11	0.71± 0.02
$A2_{u_v}$	2.53 ±. 0.75	3.50± 0.05
$A3_{u_v}$	-5.55 ± 4.75	-
$A5_{u_v}$	5.33 ± 5.39	-
$A0_{ud}$	0.69 ± 0.09	0.68± 0.03
$A2_{ud}$	11.40 ± 22.90	6.74± 0.20
$A3_{ud}$	11.39 ± 61.77	-
$A5_{ud}$	-6.54 ± 40.96	-
$A0_g$	1.39	2.42
$A2_g$	-4. ± 169	4.83 ± 1.38
$A3_g$	-93 ± 446	-
$A5_g$	96 ± 287	-
χ^2/dof	54/119	79/125

4.3.4 Effects of Target Mass Correction

Data includes the target mass effects, therefore we need to correct the theory predictions for these effects using Eqs. (1.80), (1.81). The target mass correction (TMC) affects the highest x and low Q^2 region. We expect to see that including TMC would decrease the value of Λ_{QCD}. Table 4.15 shows that the effect of taking into account the TMC on Λ_{QCD} extracted

Table 4.11: QCD fit results using *ACOT* model for the combined fit using a general and a simplified parametrization of the PDFs.

Parameters	General	Simple
$\Lambda^{(n_f=4)}$(MeV)	462 ± 53	458 ± 41
$A0_{u_v} + A0_{d_v}$	4.07+2.04	4.50 ± 2.25
$A1_{u_v}$	0.70 ± 0.07	0.72± 0.02
$A2_{u_v}$	2.30 ±. 0.48	3.49± 0.05
$A3_{u_v}$	-6.8 ± 2.9	-
$A5_{u_v}$	6.58 ± 3.3	-
$A0_{ud}$	0.71 ± 0.05	0.67± 0.03
$A2_{ud}$	11.27 ± 5.3	6.83± 0.21
$A3_{ud}$	16.52 ± 12	-
$A5_{ud}$	-13.08 ± 7.8	-
$A0_g$	1.57	2.21
$A2_g$	6.32 ± 0.83	4.30 ± 0.41
$A3_g$	0.12 ± 6.07	-
$A5_g$	4.7 ± 6.09	-
χ^2/dof	53/117	76/125

using both models is very small. In order to see the right trend of the TMC, the size of the correction is artificially increased by introducing a multiplicative constant C_i in the definition of the ξ variable:

$$\xi = \frac{2x}{1+\sqrt{1+\frac{4m_p^2x^2}{Q^2}}} \Rightarrow \xi = \frac{2x}{1+\sqrt{1+\frac{C_i 4m_p^2x^2}{Q^2}}}. \tag{4.39}$$

Table 4.12: The effect of removing the $x = 0.75$ bin in the NLO QCD fits using both massless and massive predictions.

		xF_3 fit		xF_3 and F_2 fit	
Scheme	x_{max}	$\Lambda^{(n_f=4)}$ (MeV)	χ^2/dof	$\Lambda^{(n_f=4)}$ (MeV)	χ^2/dof
$\overline{MS}$	0.75	476 ± 60	78/59	433 ± 51	79/125
$\overline{MS}$	0.65	466 ± 62	76/57	444 ± 60	70/119
ACOT	0.75	488 ± 59	77/59	458 ± 41	76/125
ACOT	0.65	477 ± 60	75/57	435 ± 40	65/119

Table 4.13: The effect of the increasing Q^2 cut on the extracted Λ_{QCD}

			xF_3 fit	xF_3 and F_2 fit	
Scheme	Q^2_{min}	$\Lambda^{(n_f=4)}$ (MeV)	χ^2/dof	$\Lambda^{(n_f=4)}$ (MeV)	χ^2/dof
$\overline{MS}$	5	476 ± 60	78/59	433 ± 51	79/125
$\overline{MS}$	7	522 ± 70	62/52	406 ± 47	63/111
$\overline{MS}$	10	525 ± 82	52/43	434 ± 48	48/93
ACOT	5	488 ± 59	77/59	458 ± 41	76/125
ACOT	7	535 ± 69	60/52	421 ± 53	61/111
ACOT	10	543 ± 82	52/43	449 ± 48	48/93

The study of varying the $C_i = 1.5, 2$ concludes that the increase in the size of the TMC decreases Λ_{QCD} as expected (see Table 4.14).

Table 4.14: Artificial increase of the TMC in order to observe the direction of the effect on Λ_{QCD} using both theoretical models.

		xF_3 fit		xF_3 and F_2 fit	
	C_i	$\Lambda^{(n_f=4)}$ (MeV)	χ^2/dof	$\Lambda^{(n_f=4)}$ (MeV)	χ^2/dof
$\overline{MS}$	1	476 ± 60	78/59	433 ± 51	79/125
	1.5	426	77/59	334	80/125
	2	359	77/59	250	84/125
ACOT	1	488 ± 59	77/59	458 ± 41	76/125
	1.5	438	75/59	387	75/125
	2	373	75/59	326	81/125

Table 4.15: Fit results using both theoretical models with and without Target Mass Correction. Results from both NS and Combined fits are shown with all the correlations errors taken into account.

		xF_3 fit			xF_3 and F_2 fit		
TMC	scheme	$\Lambda^{(n_f=4)}$ (MeV)	χ^2/dof	Prob %	$\Lambda^{(n_f=4)}$ (MeV)	χ^2/dof	Prob %
ON	$\overline{MS}$	476 ± 60	78/59	5.9	439 ± 54	79/125	99.9
OFF	$\overline{MS}$	+31	87/59	1.0	−4	93/125	98.5
ON	*ACOT*	488 ± 59	77/59	5.8	458 ± 41	76/125	99.9
OFF	*ACOT*	+14	86/59	1.2	+2	90/125	99.2

4.4 COMPARISONS OF NUTEV α_S TO OTHER MEASUREMENTS

In this section we compare our measurement of α_S to results from other experiments. All the measurements are evolved to the reference scale, M_Z. Figure 4.5 shows the results of $\alpha_S(M_Z)$ from various experiments [4]. The green band denotes the world weighted average $\alpha_S(M_Z) = 0.1185 \pm 0.0020$, as quoted by *Particle Data Group 2005* (using selected measurements), which excludes the Lattice QCD calculations. If the Lattice QCD result is included in the global fit, then the world average for $\alpha_S(M_Z)$ is decreased to $\alpha_S(M_Z) = 0.1176 \pm 0.0020$. Both NuTeV measurements are shown in red (from combined and non-singlet fits). Both results situate above the world average, but are consistent with the world average value. Our results are in better agreement with experiments of higher characteristic energy scales, such as e^+e^- collisions, which favor a higher average $\alpha_S(M_Z)$ This reduces any potential discrepancies between low and high Q^2 measurements observed by previous low Q^2 average values (described below). In the next sections we summarize the results from various experiments used in this comparison.

4.4.1 α_S from DIS

As discussed in Section 1.2, DIS provided one of the first precise methods of determining α_S from NLO QCD fits to scaling violating structure functions. The combined analysis of SLAC/BCDMS data [44, 67, 68] in a Q^2 range from 0.5 to 260 GeV^2 produced $\alpha_S(M_Z) = 0.113 \pm 0.005$, where all the errors (experimental and theoretical) are included [57]. The small value for strong coupling from DIS measurements caused some speculations in explaining the differences from measurements in e^+e^- processes. A more recent reanalysis of the SLAC, BCDMS and NMC data on F_2 that takes proper account of point-to-point correlations resulted in an increase of $\alpha_S(M_Z)$ to 0.118 ± 0.005 [58].

The most precise previous determination of α_S from fixed-target DIS experiments is

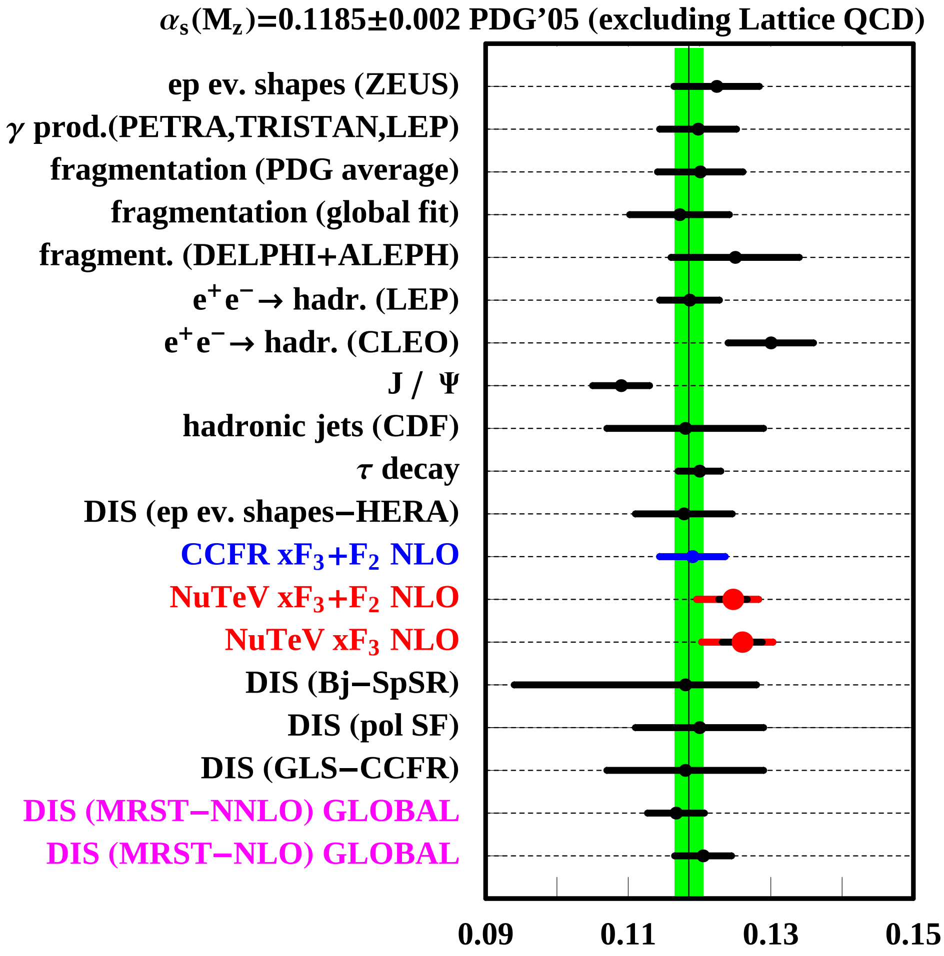


Figure 4.5: Compare NuTeV $\alpha_S(M_{Z^0})$ to other experimental measurements. The values are taken from PDG 2005. The world average excluding Lattice QCD is $\alpha_S(M_Z) = 0.1185 \pm 0.0020$.

provided by the CCFR ν-nucleon scattering experiment shown in blue in Figure 4.5, labeled as "(CCFR xF_3+F_2 NLO)"), which yielded an $\alpha_S(M_Z) = 0.119 \pm 0.002 \pm 0.001(HT) \pm 0.004$ [51]. Some discussions on the CCFR comparison is presented in Section 4.4.1.1.

A global fit, which includes selected scaling violation data, was performed by the MRST04 group [59]. They find $\alpha_S(M_Z) = 0.1205 \pm 0.004$ at NLO (labeled DIS(MRST NLO)), and $\alpha_S(M_Z) = 0.1167 \pm 0.004$ at NNLO (labeled as "DIS(MRST NNLO)"). We are in good agreement with the former NLO DIS result.

Another important measurement of α_S is from a fit to Gross-Llewellyn Smith sum rule obtained by the CCFR experiment [60]. The measured $\alpha_S(M_Z)$ is known here to the NNLO, and corresponds to $\alpha_S(M_Z) = 0.118 \pm 0.002\ (stat) \pm 0.005\ (sys) \pm 0.003\ (th)$, with theoretical error dominated by the higher-twist contribution (labeled as "DIS(GLS-CCFR)").

Polarized lepton-nucleon scattering provides another determination of α_S. The spin-dependent structure functions from SMC [61] and HERMES [62] data yield $\alpha_S(M_Z) = 0.120 \pm 0.009$ (labeled as "DIS(pol SF)"). The strong coupling obtained from the polarized Bjorken sum rule fit gives $\alpha_S(M_Z) = 0.118^{+0.010}_{-0.024}$ (labeled as "DIS(Bj-SpSR)"). The larger error are due to the extrapolation into small x region which is unmeasured. These results are shown in Figure 4.5.

α_S can also be obtained from ep scattering processes at HERA by comparing the jet rates of (1+1) to (2+1) processes. The (1+1) jet state corresponds to the lowest order in α_S scattering process, $e + q \rightarrow e + q$. The (2+1) jet state corresponds to next order in α_S where a gluon is radiated, $e + q \rightarrow e + q + g$. This measurement is performed over a wide range of Q^2, using a NLO QCD calculation. The results from jet production at HERA can be summarized with the weighted average value $\alpha_S(M_Z) = 0.1178 \pm 0.0033\ (exp) \pm 0.006\ (th)$ (labeled as "DIS ep ev.shapes(HERA)") [65].

NuTeV result is as precise as the most precise DIS determination of α_S. Although the central value is higher than previous measurements, it is consistent within experimental errors with other DIS results.

4.4.1.1 Comparison with CCFR Results

The CCFR experiment is the predecessor of the NuTeV experiment. Our measurement is as precise as the CCFR result. NuTeV measurement of α_S is about 2σ higher than the CCFR result. Therefore, it is important to understand the differences in the α_S results. Here we outline the known differences:

- CCFR analysis extracted Λ_{QCD} from NLO QCD fits to structure functions which were corrected for the charm mass production using a LO rescaling prescription; NuTeV data does not remove the charm mass effects.
- NuTeV extracted structure functions by fitting differential cross sections using full point-to-point covariance matrix. The extracted differential cross section differ, in particular at high x. This difference was found to be due to calibration [22].

The CCFR experiment yields a value for Λ_{QCD} (with its corresponding α_S) from combined F_2 and xF_3 fits as

$$\Lambda_{\overline{MS}}^{n_f=4} = 337 \pm 28 \text{ MeV} \quad \longrightarrow \quad \alpha_S(M_Z) = 0.119 \pm 0.002 \pm \underbrace{0.001}_{HT} \pm \underbrace{0.004}_{th}. \tag{4.40}$$

CCFR theoretical uncertainty on Λ_{QCD} is not estimated, but quoted from a QCD analysis of previous DIS measurements, [57]. As for the non-singlet fit, CCFR obtains [50]

$$\Lambda_{\overline{MS}}^{n_f=4} = 381 \pm 53 \text{ MeV}. \tag{4.41}$$

The best way to directly compare the measurements is by using the same QCD program and CCFR data set extracted through a similar mechanism as NuTeV structure function extraction. Unfortunately, only structure function F_2 from CCFR is available [63]. Hence, we can only compare the fit results from F_2. The results from these direct comparisons give -95 MeV difference in Λ_{QCD}, corresponding to 2σ disagreement between CCFR and NuTeV. This implies that the disagreement is purely due to data differences [22].

4.4.2 α_S from Other Processes

We compare NuTeV results to α_S measurements from other QCD processes besides DIS which were included in the Figure 4.5. We quote values of α_S as quoted by PDG 2005 [4]

- α_S from decays of the τ lepton, $\tau \rightarrow \nu_\tau + hadrons$. The strong coupling is determined from the ratio of the hadronic to the electronic branching ratio of the tau lepton, $R_\tau = B_{hadr}/B_e$. The measured value obtained is $\alpha_S(M_Z) = 0.120 \pm 0.003$ (labeled "τ decay").
- α_S obtained from the production of hadronic jets with large transverse momentum. Jet data are fitted over a large range of transverse momenta, assuming a set of structure functions. The CDF experiment obtains $\alpha_S(M_Z) = 0.118 \pm 0.011$ (NLO) from hadronic jet rates and event shapes (labeled "hadronic jets(CDF)" in Figure 4.5).
- α_S from e^+e^- annihilation can be determined from various methods: hadronic event shapes, jet production rates, energy-energy correlation function, scaling violations of fragmentation functions, total hadronic cross section.

  Strong coupling from the scaling violations of the fragmentation functions in the $e^+e^- \rightarrow h + X$ process, is determined similarly to the DIS scaling violations of the structure functions method. The factorization theorem is applied to separate the hard gluon radiation from the hadronization. Experiments like ALEPH, DELPHI and OPAL [64] performed measurements of α_S at NLO, yielding an average $\alpha_S(M_Z) = 0.125 \pm 0.005 \pm 0.008$ (labeled "fragment(DELPHI+ALEPH)" in Figure 4.5.)

  A determination of α_S can be obtained from the total hadronic cross section. This is the first quantity for which NNLO QCD corrections have been calculated, $R_\gamma = \frac{\sigma(e^+e^- \rightarrow hadrons)}{\sigma(e^+e^- \rightarrow \mu^+\mu^-)}$. The PETRA, TRISTAN, LEP experiments obtained $\alpha_S(M_Z) = 0.1198 \pm 0.0054$ (labeled "γ prod.(PETRA,TRISTA,LEP)" in Figure 4.5.)

  The event shape characterizes the structure of a multihadron event by a single number. There are many measurements of α_S based on global event shape variables in the energy range between 22 GeV to around 200 GeV.

- α_S can be obtained from the ratios of partial decay widths at the heavy-quark mass scale, $R_\mu(\Upsilon) = \frac{\Gamma(\Upsilon \to hadrons)}{\Gamma(\Upsilon \to \mu^+\mu^-)}$. The available data for J/ψ and Υ resonances is very precise, therefore the errors are theoretically dominated. The Strong coupling constant is determined to be $\alpha_S(M_Z) = 0.109 \pm 0.004$. The result is shown in Figure 4.5 and is labeled as "J/ψ".
- α_S from Lattice QCD calculations. Mass splitting of heavy quarkonia states can be used to determine $\alpha_S(M_Z) = 0.1170 \pm 0.0012$.

5.0 CONCLUSIONS

This thesis has presented a new measurement of $\Lambda_{QCD}^{n_f=4}$ from NLO QCD fits to Q^2 dependence of the structure functions in the $\nu-$ nucleon DIS process. This is the first determination of Λ_{QCD} which uses a NLO theory model taking into account heavy quark production ($ACOT$ model [12]).

Also, this analysis has presented a new method of evaluating the systematic uncertainties for Λ_{QCD} by constructing a full covariance error matrix which takes into account point-to-point data correlations.

The largest systematic uncertainties in measuring α_S from previous DIS experiments were due to the energy scales. The NuTeV experiment was designed to perform precision QCD measurements by improving considerably its knowledge of energy scales which were determined to a precision of 0.43% for hadrons and 0.7% for muons (almost a factor of two improvement compared to previous neutrino experiment).

We obtained the following results for $\Lambda_{QCD}^{n_f=4}$ with its corresponding $\alpha_S(M_Z)$

$$\begin{aligned}\Lambda_{ACOT}^{NLO,(n=4)} &= 488 \pm 59(stat+sys)^{+74}_{-113}(theory); \alpha_S(M_Z) = 0.1260 \pm 0.0028^{+0.0034}_{0.0050},\\ \Lambda_{ACOT}^{NLO,(n=4)} &= 458 \pm 41(stat+sys)^{+61}_{-87}(theory); \alpha_S(M_Z) = 0.1247 \pm 0.0020^{+0.0030}_{-0.0047},\end{aligned}$$

from non-singlet and combined fits, respectively. The first error corresponds to total experimental uncertainties (statistical and systematic). The second second error is the estimated theoretical uncertainty due to factorization scale dependence. The results from non-singlet and combined fits are consistent with one another.

NuTeV measurement is higher than the world average value, but is consistent within errors. We also observe that this measurement is one of the most precise determination of α_S, and as precise as the most precise measurement of α_S from DIS to date. This is important because NuTeV energy scale situates in the range of low to moderate Q^2 , where there are few precision experiments to test the prediction powers of pQCD. This is the region where α_S is changing most rapidly, thus discrepancies could be more revealing.

The results were compared to other measurements of α_S from DIS, high energy hadron collisions, electron-positron annihilation and heavy quarkonia decay processes. The result is in better agreement with high Q^2 measurements than previous low Q^2 DIS results. Comparing the results extracted from experiments of various characteristic energy scales is an important test of pQCD.

The dominant uncertainty in the α_S measurement is due to the factorization scale dependence. The effects of the quark masses where studied and are negligible compared to the uncertainty associated with the scale dependence. Therefore, any further improvement in the determination of α_S from DIS would require improvement in theoretical calculations.

The largest experimental uncertainties are due to energy scales uncertainties. The size of these uncertainties is of the order of statistical precision. Further improvement in decrease of the experimental uncertainty could be achieved by an increased statistics (especially for non-singlet fits).

The next generation of neutrino scattering experiments, Minerva and Minos near detector, might improve the statistical precision for α_s, and extend measurements to lower Q^2 scale. However, it will have the challenge of dealing with non-perturbative effects that are large at low Q^2, and energy scale uncertainties.

APPENDIX A

THE NUTEV COLLABORATION

T. Adams[4], A. Alton[4], S. Avvakumov[8], L. de Barbaro[5], P. de Barbaro[8], R. H. Bernstein[3], A. Bodek[8], T. Bolton[4], S. Boyd[7], J. Brau[6], D. Buchholz[5], H. Budd[8], L. Bugel[3], J. Conrad[2], R. B. Drucker[6], B. T. Fleming[2], R. Frey[6], J.A. Formaggio[2], olorJ. Goldman[4], M. Goncharov[4], D. A. Harris[8], R. A. Johnson[1], J . H. Kim[2], S. Koutsoliotas[2], M. J. Lamm[3], W. Marsh[3], D. Mason[6], J. McDonald[7], K. S. McFarland[8,3], C. McNulty[2], D. Naples[7], P. Nienaber[3], V. Radescu[7], A. Romosan[2], W. K. Sakumoto[8], H. Schellman[5], M. H. Shaevitz[2], P. Spentzouris[2], E. G. Stern[2], N. Suwonjandee[1], M. Tzanov[7], M. Vakili [1], A. Vaitaitis[2], U. K. Yang[8], J. Yu[3], G. P. Zeller[5] and E. D. Zimmerman[2]

[1]University of Cincinnati, Cincinnati, OH 45221

[2]Columbia University, New York, NY 10027

[3]Fermi National Accelerator Lab., Batavia, IL 60510

[4]Kansas State University, Manhattan, KS 66506

[5]Northwestern University, Evanston, IL 60208

[6]University of Oregon, Eugene, OR 97403

[7]University of Pittsburgh, Pittsburgh, PA 15260

[8]University of Rochester, Rochester, NY 14627

APPENDIX B

STRUCTURE FUNCTIONS TABLES

B.1 DATA TABLE FOR NON-SINGLET FIT

Table B1 displays the xF_3 data file used in the non-singlet fit taking into account all the correlations. This table was also used for constructing the structure function covariance matrix as described in Chapter 3.

Table B1: xF_3 structure function table from 1-parameter fit. Each column corresponds to systematic uncertainties. This table was used for non-singlet fit.

x	Q^2	xF_3	stat	E_μ scl	E_h scl	m_c	$\frac{B}{A}$	E_μ sm	E_{had} sm	$\frac{B}{A}$ model
0.0150	1.2589	0.1799	0.0146	-0.0039	0.0006	-0.0003	-0.0001	0.0002	0.0020	0.0032
0.0150	1.9952	0.2182	0.0157	-0.0077	-0.0005	0.0016	-0.0002	0.0038	0.0019	0.0002
0.0150	3.1622	0.2634	0.0177	-0.0029	0.0005	-0.0016	-0.0005	0.0008	-0.0010	0.0014
0.0150	5.0118	0.3262	0.0212	-0.0069	-0.0006	0.0028	-0.0008	0.0142	0.0017	0.0129
0.0150	7.9432	0.3026	0.0621	-0.0288	0.0127	-0.0001	-0.0010	0.0097	0.0000	0.0065
0.0450	1.2589	0.3755	0.0346	0.0000	-0.0013	-0.0075	-0.0011	0.0107	0.0103	0.0038
0.0450	1.9952	0.4007	0.0211	-0.0167	-0.0021	-0.0022	-0.0002	0.0114	0.0001	0.0078
0.0450	3.1622	0.4655	0.0176	0.0001	-0.0030	-0.0033	-0.0001	0.0024	0.0053	0.0044
0.0450	5.0118	0.4829	0.0160	-0.0109	0.0017	-0.0009	-0.0005	0.0033	-0.0014	0.0082
0.0450	7.9432	0.4908	0.0185	-0.0097	-0.0006	0.0008	-0.0010	0.0042	0.0012	0.0034
0.0450	12.5890	0.5400	0.0197	-0.0076	-0.0012	0.0037	-0.0015	0.0023	-0.0004	0.0030
0.0450	19.9520	0.5729	0.0429	-0.0361	0.0284	0.0186	-0.0018	0.0070	0.0137	0.0171
0.0800	1.9952	0.5362	0.0337	-0.0116	-0.0111	-0.0010	-0.0010	0.0035	0.0060	0.0014
0.0800	3.1622	0.6031	0.0229	-0.0061	-0.0109	-0.0094	0.0000	0.0249	0.0017	0.0035
0.0800	5.0118	0.6596	0.0173	-0.0058	0.0013	-0.0032	0.0000	0.0083	0.0006	0.0086
0.0800	7.9432	0.6441	0.0142	-0.0117	-0.0044	-0.0040	-0.0005	0.0104	0.0028	0.0079
0.0800	12.5890	0.6651	0.0165	-0.0059	0.0059	-0.0003	-0.0013	0.0092	-0.0051	0.0027
0.0800	19.9520	0.6790	0.0164	-0.0086	0.0026	-0.0004	-0.0019	0.0162	0.0014	0.0100

Continues next page

Continued

x	Q^2	xF_3	stat	E_μ scl	E_h scl	m_c	$\frac{B}{A}$	E_μ sm	E_{had} sm	$\frac{B}{A}$ model
0.0800	31.6220	0.7531	0.0266	-0.0235	0.0141	0.0086	-0.0024	0.0148	-0.0010	0.0116
0.1250	3.1622	0.6410	0.0321	-0.0021	-0.0041	-0.0132	-0.0010	0.0001	0.0068	0.0103
0.1250	5.0118	0.6980	0.0213	-0.0103	0.0023	-0.0161	0.0001	0.0046	0.0045	0.0015
0.1250	7.9432	0.7373	0.0145	-0.0064	0.0026	0.0022	0.0001	0.0036	0.0011	0.0093
0.1250	12.5890	0.7687	0.0135	-0.0136	0.0016	-0.0018	-0.0008	0.0053	0.0021	0.0032
0.1250	19.9520	0.7677	0.0148	-0.0137	0.0066	0.0014	-0.0016	0.0097	0.0031	0.0012
0.1250	31.6220	0.7634	0.0143	-0.0093	0.0024	0.0056	-0.0022	0.0021	0.0010	0.0037
0.1250	50.1180	0.8024	0.0256	-0.0082	0.0204	0.0144	-0.0026	0.0019	0.0017	0.0067
0.1750	3.1622	0.6167	0.0857	-0.0021	0.0007	-0.0269	-0.0010	0.0115	0.0396	0.0149
0.1750	5.0118	0.7514	0.0314	-0.0008	-0.0068	-0.0152	-0.0001	0.0036	-0.0021	0.0007
0.1750	7.9432	0.7710	0.0170	-0.0054	0.0043	-0.0009	0.0003	0.0005	0.0043	0.0019
0.1750	12.5890	0.8087	0.0156	-0.0051	0.0010	-0.0009	-0.0004	0.0039	0.0015	0.0015
0.1750	19.9520	0.7962	0.0129	-0.0079	0.0034	0.0003	-0.0009	0.0070	-0.0035	0.0024
0.1750	31.6220	0.7608	0.0145	-0.0087	-0.0017	0.0006	-0.0019	0.0034	-0.0040	0.0037
0.1750	50.1180	0.7853	0.0139	-0.0149	0.0040	0.0017	-0.0024	0.0027	0.0016	0.0001
0.2250	5.0118	0.7818	0.0467	-0.0126	0.0034	-0.0265	-0.0005	0.0282	0.0073	0.0034
0.2250	7.9432	0.7667	0.0239	0.0032	-0.0029	-0.0047	0.0003	0.0038	-0.0011	0.0012
0.2250	12.5890	0.7600	0.0156	0.0039	-0.0046	-0.0023	0.0002	0.0015	0.0055	0.0001
0.2250	19.9520	0.7871	0.0136	-0.0004	0.0034	-0.0030	-0.0007	0.0164	-0.0016	0.0120
0.2250	31.6220	0.7413	0.0137	-0.0039	0.0057	0.0003	-0.0015	0.0032	0.0010	0.0002
0.2250	50.1180	0.7149	0.0131	-0.0085	0.0008	0.0013	-0.0020	0.0039	-0.0009	0.0066
0.2250	79.4320	0.6968	0.0178	-0.0095	0.0100	0.0100	-0.0023	0.0073	0.0026	0.0075
0.2750	5.0118	0.6842	0.0829	0.0124	-0.0130	-0.0106	-0.0006	0.0770	-0.0009	0.0742
0.2750	7.9432	0.7264	0.0261	-0.0053	0.0008	-0.0056	0.0003	0.0014	0.0056	0.0129
0.2750	12.5890	0.7057	0.0183	0.0051	-0.0066	-0.0003	0.0001	0.0010	0.0020	0.0058
0.2750	19.9520	0.6791	0.0151	-0.0074	0.0040	-0.0008	-0.0003	0.0020	0.0020	0.0001
0.2750	31.6220	0.6812	0.0132	0.0021	0.0036	0.0035	-0.0011	0.0097	0.0006	0.0038
0.2750	50.1180	0.6387	0.0141	0.0037	-0.0030	0.0008	-0.0017	0.0037	0.0014	0.0015
0.2750	79.4320	0.6480	0.0142	-0.0160	0.0066	0.0037	-0.0021	0.0078	-0.0008	0.0027
0.3500	7.9432	0.5658	0.0294	0.0120	-0.0026	-0.0047	-0.0002	0.0040	0.0025	0.0052
0.3500	12.5890	0.5789	0.0154	0.0080	0.0011	-0.0005	0.0002	0.0017	0.0008	0.0060
0.3500	19.9520	0.5721	0.0112	-0.0002	-0.0002	-0.0045	0.0000	0.0074	0.0016	0.0007
0.3500	31.6220	0.5355	0.0087	0.0012	-0.0022	0.0003	-0.0006	0.0012	0.0003	0.0003
0.3500	50.1180	0.5144	0.0100	-0.0001	0.0007	-0.0021	-0.0012	0.0044	-0.0010	0.0001
0.3500	79.4320	0.5018	0.0082	-0.0003	0.0003	0.0023	-0.0015	0.0009	0.0017	0.0000
0.4500	7.9432	0.4050	0.0446	0.0190	-0.0153	-0.0176	-0.0003	0.0236	0.0000	0.0360
0.4500	12.5890	0.3878	0.0169	0.0142	-0.0106	-0.0009	0.0002	0.0053	-0.0008	0.0014
0.4500	19.9520	0.3787	0.0113	0.0120	-0.0032	0.0006	0.0000	0.0015	0.0009	0.0047
0.4500	31.6220	0.3489	0.0086	0.0088	-0.0039	-0.0011	-0.0003	0.0038	0.0016	0.0036
0.4500	50.1180	0.3187	0.0086	0.0117	-0.0023	0.0010	-0.0006	0.0009	-0.0002	0.0005
0.4500	79.4320	0.2965	0.0075	0.0019	-0.0026	-0.0008	-0.0009	0.0016	0.0008	0.0013
0.4500	125.8900	0.2913	0.0093	0.0089	0.0067	0.0008	-0.0009	0.0022	-0.0011	0.0031
0.5500	12.5890	0.2435	0.0192	-0.0012	-0.0046	-0.0047	-0.0001	0.0029	-0.0005	0.0054
0.5500	19.9520	0.2219	0.0105	0.0074	0.0015	-0.0021	0.0000	0.0028	-0.0001	0.0024
0.5500	31.6220	0.2097	0.0077	-0.0003	-0.0021	0.0009	-0.0002	0.0031	0.0010	0.0029
0.5500	50.1180	0.1801	0.0068	0.0078	-0.0011	-0.0001	-0.0003	0.0017	-0.0017	0.0001
0.5500	79.4320	0.1635	0.0065	0.0010	0.0015	-0.0026	-0.0004	0.0029	-0.0002	0.0037
0.5500	125.8900	0.1341	0.0085	0.0054	0.0001	0.0021	-0.0005	0.0043	0.0017	0.0032
0.6500	12.5890	0.1250	0.0177	0.0079	0.0047	0.0037	0.0002	0.0055	0.0048	0.0003
0.6500	19.9520	0.1252	0.0091	0.0065	-0.0047	0.0003	0.0001	0.0001	0.0016	0.0031
0.6500	31.6220	0.1158	0.0061	0.0058	-0.0013	0.0004	0.0000	0.0018	-0.0012	0.0013
0.6500	50.1180	0.0844	0.0055	0.0041	-0.0005	-0.0005	-0.0001	0.0014	0.0015	0.0001
0.6500	79.4320	0.0713	0.0060	0.0016	-0.0010	0.0015	-0.0002	0.0006	0.0013	0.0004
0.6500	125.8900	0.0702	0.0063	0.0058	0.0009	-0.0016	-0.0002	0.0017	0.0017	0.0037
0.7500	19.9520	0.0501	0.0070	0.0024	0.0006	-0.0023	0.0000	0.0002	0.0004	0.0011
0.7500	31.6220	0.0379	0.0040	0.0044	-0.0019	-0.0005	0.0000	0.0005	-0.0006	0.0011
0.7500	50.1180	0.0338	0.0042	0.0022	0.0012	-0.0016	-0.0001	0.0004	0.0003	0.0009

B.2 DATA TABLE FOR COMBINED FIT

Table B2 displays the correlated F_2 and xF_3 data file used in the combined fit taking into account all the correlations.

Table B2: F_2 and xF_3 structure function from simultaneous fit to differential cross-section using the covariance matrix taking into account point-to-point correlations.

x	Q^2	F2	F2err	F2Rsys	F2msys	xF3	XF3err	xF3Rsys	XF3msys	F2xF3corr	xbin	Q2bin
0.015	1.259	1.1550000	0.0249800	-0.0020000	0.0090000	0.1821000	0.0389500	0.0022000	-0.0002500	-0.1148000	2	4
0.015	1.995	1.3300000	0.0307700	0.0029999	0.0130000	0.2170000	0.0417500	0.0013000	-0.0007500	-0.1582000	2	5
0.015	3.162	1.4730000	0.0397300	-0.0010000	0.0180000	0.2669000	0.0470000	0.0010000	-0.0005500	-0.2252000	2	6
0.015	5.012	1.5580000	0.0591100	0.0080000	0.0270000	0.3220000	0.0575200	0.0006000	-0.0002000	-0.2393000	2	7
0.015	7.943	1.6150000	0.1822000	0.0240000	0.0320000	0.2751000	0.1626000	0.0005000	0.0005000	-0.3664000	2	8
0.045	1.259	1.0930001	0.0246100	-0.0030000	0.0015000	0.3758000	0.0866300	0.0045000	-0.0002000	-0.0426200	3	4
0.045	1.995	1.2280000	0.0232800	0.0000000	0.0040000	0.4073000	0.0526400	0.0025000	-0.0008000	0.0078580	3	5
0.045	3.162	1.3430001	0.0243200	0.0050000	0.0060000	0.4627000	0.0443200	0.0013000	-0.0007500	-0.0010020	3	6
0.045	5.012	1.4590000	0.0274700	0.0120000	0.0095000	0.4747000	0.0398900	-0.0001000	-0.0016500	0.0096090	3	7
0.045	7.943	1.5120000	0.0333000	0.0180000	0.0105000	0.5004000	0.0454700	-0.0004000	-0.0015000	-0.0933900	3	8
0.045	12.589	1.5510000	0.0446700	0.0270000	0.0160000	0.5374000	0.0491000	0.0000000	-0.0005000	-0.1220000	3	9
0.045	19.952	1.6840000	0.1085000	0.0290000	0.0215001	0.5691000	0.1049000	0.0001000	-0.0001500	-0.1893000	3	10
0.080	1.259	1.0670000	0.0670500	-0.0020000	0.0000000	0.4083000	0.4714000	-0.0075000	0.0000000	-0.1067000	4	4
0.080	1.995	1.2180001	0.0222000	-0.0010000	0.0015000	0.5397000	0.0810800	-0.0003000	-0.0003500	0.0002854	4	5
0.080	3.162	1.3020000	0.0220500	0.0009999	0.0025000	0.6069000	0.0550300	0.0001000	-0.0007500	0.0525000	4	6
0.080	5.012	1.3600000	0.0210800	0.0070000	0.0045000	0.6616000	0.0411400	-0.0005000	-0.0012500	0.0835200	4	7
0.080	7.943	1.3770000	0.0217100	0.0110000	0.0065000	0.6383000	0.0337400	-0.0011000	-0.0015500	0.1116000	4	8
0.080	12.589	1.4230000	0.0273800	0.0160000	0.0070000	0.6608000	0.0387700	-0.0012000	-0.0015500	0.0468200	4	9
0.080	19.952	1.4220001	0.0325600	0.0189999	0.0100001	0.6794000	0.0387400	-0.0006000	-0.0007500	0.0158000	4	10
0.080	31.622	1.4930000	0.0613700	0.0210000	0.0130000	0.7281000	0.0630800	-0.0001000	-0.0003000	-0.0001793	4	11
0.125	1.995	1.1350000	0.0631500	-0.0010000	0.0000000	0.7274000	0.4465000	-0.0185000	-0.0000500	-0.0676600	5	5
0.125	3.162	1.2340000	0.0197700	-0.0009999	0.0010000	0.6374000	0.0740200	-0.0046999	-0.0005000	0.0164800	5	6
0.125	5.012	1.2490000	0.0192600	0.0020001	0.0020000	0.6928000	0.0492200	-0.0022000	-0.0009000	0.1099000	5	7
0.125	7.943	1.2560000	0.0171300	0.0050000	0.0030000	0.7372000	0.0331900	-0.0019000	-0.0013000	0.1605000	5	8
0.125	12.589	1.2819999	0.0195700	0.0080000	0.0040000	0.7616000	0.0309000	-0.0018000	-0.0015500	0.2111000	5	9
0.125	19.952	1.2680000	0.0232600	0.0100000	0.0045000	0.7594000	0.0335400	-0.0014000	-0.0012000	0.1628000	5	10
0.125	31.622	1.2530000	0.0267100	0.0130000	0.0060000	0.7665000	0.0321900	-0.0006000	-0.0006000	0.1593000	5	11
0.125	50.118	1.3230000	0.0553400	0.0130000	0.0075000	0.7888000	0.0586500	-0.0002000	-0.0002000	0.1438000	5	12
0.175	3.162	1.1250000	0.0366500	0.0000000	0.0005000	0.5863000	0.1922000	0.0000000	-0.0001500	-0.0296100	6	6
0.175	5.012	1.1210001	0.0192700	0.0000000	0.0005000	0.7508000	0.0702800	0.0000000	-0.0004500	0.0716300	6	7
0.175	7.943	1.1580000	0.0163600	0.0000000	0.0015000	0.7746000	0.0376400	0.0000000	-0.0009500	0.1854000	6	8
0.175	12.589	1.1540000	0.0174800	0.0000000	0.0020000	0.7929000	0.0339100	0.0000000	-0.0011000	0.2218000	6	9
0.175	19.952	1.1390001	0.0183100	0.0000000	0.0030000	0.7779000	0.0280100	0.0000000	-0.0011000	0.2842000	6	10
0.175	31.622	1.1310000	0.0224600	0.0000000	0.0030000	0.7667000	0.0314500	0.0000000	-0.0008500	0.2304000	6	11
0.175	50.118	1.0940000	0.0266200	0.0000000	0.0045000	0.7843000	0.0308400	0.0000000	-0.0003500	0.3463000	6	12
0.175	79.432	1.1520000	0.0830000	0.0000000	0.0050000	0.8030000	0.0861200	0.0000000	0.0000000	0.3541000	6	13
0.225	5.012	1.0330000	0.0221300	0.0000000	0.0000000	0.7713000	0.1014000	0.0000000	-0.0003000	0.0535300	7	7
0.225	7.943	1.0260000	0.0173400	0.0000000	0.0010000	0.7582000	0.0523300	0.0000000	-0.0004500	0.1415000	7	8
0.225	12.589	1.0000000	0.0154600	0.0000000	0.0010000	0.7483000	0.0334600	0.0000000	-0.0008500	0.2449000	7	9
0.225	19.952	0.9956000	0.0170600	0.0000000	0.0016000	0.7646000	0.0288100	0.0000000	-0.0009000	0.3485000	7	10
0.225	31.622	0.9545000	0.0185100	0.0000000	0.0016500	0.7380000	0.0286700	0.0000000	-0.0007500	0.3163000	7	11
0.225	50.118	0.9246000	0.0211400	0.0000000	0.0020500	0.7118000	0.0278000	0.0000000	-0.0004500	0.3358000	7	12
0.225	79.432	0.9121000	0.0345700	0.0000000	0.0027500	0.6938000	0.0385200	0.0000000	-0.0001500	0.4089000	7	13

Continues next page

Continued

x	Q^2	F2	F2err	F2Rsys	F2msys	xF3	XF3err	xF3Rsys	XF3msys	F2xF3corr	xbin	Q2bin
0.275	5.012	0.9046000	0.0331400	0.0000000	0.0002500	0.6478000	0.1758000	0.0000000	-0.0001500	0.0266200	8	7
0.275	7.943	0.8831000	0.0163400	0.0000000	0.0005500	0.7323000	0.0555700	0.0000000	-0.0004500	0.1529000	8	8
0.275	12.589	0.8751000	0.0155300	0.0000000	0.0007500	0.6952000	0.0387100	0.0000000	-0.0005000	0.2369000	8	9
0.275	19.952	0.8424000	0.0154300	0.0000000	0.0008000	0.6615000	0.0315400	0.0000000	-0.0005500	0.2618000	8	10
0.275	31.622	0.8244000	0.0162400	0.0000000	0.0010500	0.6654000	0.0269100	0.0000000	-0.0006000	0.3419000	8	11
0.275	50.118	0.7680000	0.0198900	0.0000000	0.0011500	0.6425000	0.0285100	0.0000000	-0.0003500	0.3719000	8	12
0.275	79.432	0.7539000	0.0249300	0.0000000	0.0015000	0.6400000	0.0303300	0.0000000	-0.0001000	0.4588000	8	13
0.350	7.943	0.6850000	0.0130700	0.0000000	0.0002000	0.5490000	0.0610600	0.0000000	-0.0002000	0.0825400	9	8
0.350	12.589	0.6721000	0.0099800	0.0000000	0.0003000	0.5713000	0.0320600	0.0000000	-0.0002000	0.1804000	9	9
0.350	19.952	0.6320000	0.0095910	0.0000000	0.0004000	0.5561000	0.0228700	0.0000000	-0.0003000	0.2740000	9	10
0.350	31.622	0.5922000	0.0095350	0.0000000	0.0005500	0.5332000	0.0172400	0.0000000	-0.0003000	0.3948000	9	11
0.350	50.118	0.5630000	0.0116100	0.0000000	0.0005500	0.5214000	0.0190000	0.0000000	-0.0003000	0.4014000	9	12
0.350	79.432	0.5415000	0.0118600	0.0000000	0.0006000	0.5037000	0.0161600	0.0000000	-0.0002000	0.4878000	9	13
0.350	125.890	0.5202000	0.0257600	0.0000000	0.0008500	0.4785000	0.0301400	0.0000000	-0.0000500	0.6140000	9	14
0.450	7.943	0.4467000	0.0168000	0.0000000	0.0001000	0.3817000	0.0896800	0.0000000	-0.0001000	0.0536800	10	8
0.450	12.589	0.4236000	0.0086880	0.0000000	0.0001000	0.3838000	0.0347700	0.0000000	-0.0001000	0.1406000	10	9
0.450	19.952	0.3950000	0.0079170	0.0000000	0.0001500	0.3735000	0.0230200	0.0000000	-0.0001000	0.2461000	10	10
0.450	31.622	0.3615000	0.0075300	0.0000000	0.0001500	0.3425000	0.0170500	0.0000000	-0.0001000	0.3342000	10	11
0.450	50.118	0.3417000	0.0085740	0.0000000	0.0002000	0.3222000	0.0167200	0.0000000	-0.0001000	0.3376000	10	12
0.450	79.432	0.3130000	0.0085700	0.0000000	0.0002000	0.2990000	0.0143000	0.0000000	-0.0000500	0.3491000	10	13
0.450	125.890	0.3038000	0.0135800	0.0000000	0.0003000	0.2884000	0.0180400	0.0000000	-0.0000500	0.4808000	10	14
0.550	12.589	0.2548000	0.0077040	0.0000000	0.0000500	0.2347000	0.0382700	0.0000000	-0.0000500	0.0767800	11	9
0.550	19.952	0.2248000	0.0056970	0.0000000	0.0000500	0.2121000	0.0209700	0.0000000	-0.0000500	0.1793000	11	10
0.550	31.622	0.2075000	0.0054880	0.0000000	0.0000500	0.2044000	0.0154400	0.0000000	-0.0000500	0.2163000	11	11
0.550	50.118	0.1772000	0.0061200	0.0000000	0.0000500	0.1843000	0.0128400	0.0000000	-0.0000500	0.3785000	11	12
0.550	79.432	0.1607000	0.0061050	0.0000000	0.0001000	0.1622000	0.0124600	0.0000000	-0.0000500	0.2576000	11	13
0.550	125.890	0.1569000	0.0103700	0.0000000	0.0001000	0.1347000	0.0159400	0.0000000	0.0000000	0.2513000	11	14
0.650	12.589	0.1362000	0.0066600	0.0000000	0.0000500	0.1330000	0.0339500	0.0000000	-0.0000500	0.0966300	12	9
0.650	19.952	0.1150000	0.0042000	0.0000000	0.0000500	0.1206000	0.0177200	0.0000000	0.0000000	0.1804000	12	10
0.650	31.622	0.0985200	0.0035160	0.0000000	0.0000200	0.1126000	0.0116800	0.0000000	-0.0000500	0.2456000	12	11
0.650	50.118	0.0875500	0.0039140	0.0000000	0.0000250	0.0832300	0.0103100	0.0000000	-0.0000150	0.2288000	12	12
0.650	79.432	0.0785800	0.0043430	0.0000000	0.0000200	0.0718600	0.0107200	0.0000000	-0.0000100	0.1101000	12	13
0.650	125.890	0.0707100	0.0061910	0.0000000	0.0000250	0.0696300	0.0117300	0.0000000	0.0000050	0.2309000	12	14
0.750	12.589	0.0626300	0.0051450	0.0000000	0.0000150	0.0499400	0.0292900	0.0000000	-0.0000050	0.0304500	13	9
0.750	19.952	0.0495000	0.0026160	0.0000000	0.0000050	0.0479300	0.0134100	0.0000000	-0.0000050	0.1028000	13	10
0.750	31.622	0.0381800	0.0020210	0.0000000	0.0000050	0.0351600	0.0075340	0.0000000	0.0000000	0.0904300	13	11
0.750	50.118	0.0332900	0.0020300	0.0000000	0.0000050	0.0343500	0.0077230	0.0000000	-0.0000050	0.0541000	13	12
0.750	79.432	0.0308300	0.0030310	0.0000000	0.0000050	0.0254000	0.0100100	0.0000000	0.0000050	-0.1176000	13	13

APPENDIX C

QCD SOURCE CODE: PART A

This Appendix contains relevant portions of the QCD program used in this analysis: the main program, the fitter, and the evolution programs. Table C summarizes the purpose of key QCD programs. The QCD program is a result of fusion of two programs:

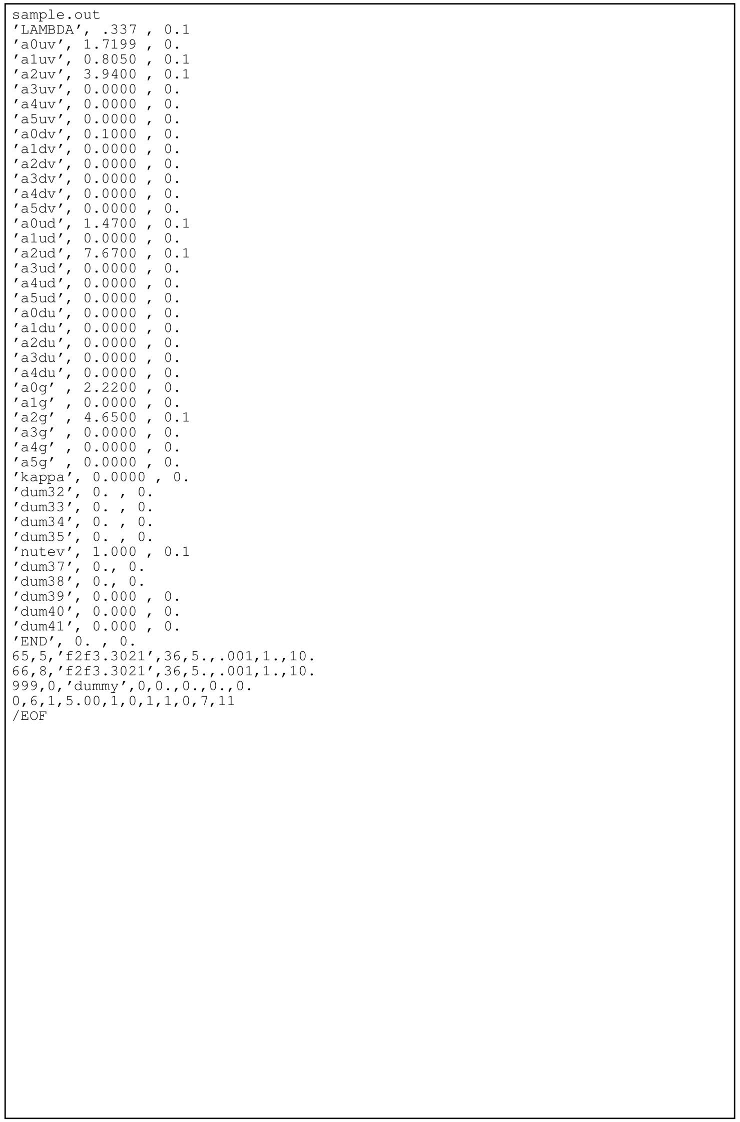
- calculation of structure function taking into account heavy quark treatment [54];
- massless quark evolution, fitting, and calculation of structure functions in a massless $\overline{MS}$ scheme [53].

Table C1: Summary of the QCD program.

Program Name	Purpose	Description
altpfit02.f	Main program	Reads in the input file, stores data points in an array (after cuts), calls minimization, writes the output.
minim.f	Minimization program	Marquardt algorithm; calls **datscn** which calculates χ^2, α, β. It returns χ^2 and errors.
altpar02.f	Param. and evol. PDFs	Parametrizes PDFs at initial Q_0^2. Evolves massless PDFs. QCD sum rules are applied here.
theory02.f	Theory calculations	First call stores the evolved PDFs in a grid. Through a flag choose $\overline{MS}$ model (theory2.f) or $ACOT$ model (theory_fo.f). Target Mass correction is applied here.
alpha_cwz.f	Calculates α_S	Given Λ_{QCD} and Q^2 it returns n_f and α_S. The threshold quark masses are set here.
cc103.f	Returns Structure Functions ($ACOT$)	**fcc123** subroutine loops over all interacting quarks and returns structure functions under every scheme. CKM couplings applied here. The quark masses stored in the common block **xmarray**.
tot104.f	Adds all contributing graphs	The schemes are defined here: sums the contributing diagrams.
unified104.f	Computes each graph	Calculations are made in helicity basis and then transformed to tensor basis.
fredpdf.f	Patch PDFs	This is where the call to the parametrized PDFs is made (**pdfjo** is called here).
alphas70.f	use J. Owens α_S	Adjusted to use J.Owens α_S calculation in **alpha_cwz.f.**
utilities103.f	technical tools	It contains tools needed to perform the calculations in $ACOT$ mode(integrations).

APPENDIX D

QCD PROGRAM: FITTER, EVOLUTION, $\overline{MS}$



```
sample.out
’LAMBDA’, .337 , 0.1
’a0uv’, 1.7199 , 0.
’a1uv’, 0.8050 , 0.1
’a2uv’, 3.9400 , 0.1
’a3uv’, 0.0000 , 0.
’a4uv’, 0.0000 , 0.
’a5uv’, 0.0000 , 0.
’a0dv’, 0.1000 , 0.
’a1dv’, 0.0000 , 0.
’a2dv’, 0.0000 , 0.
’a3dv’, 0.0000 , 0.
’a4dv’, 0.0000 , 0.
’a5dv’, 0.0000 , 0.
’a0ud’, 1.4700 , 0.1
’a1ud’, 0.0000 , 0.
’a2ud’, 7.6700 , 0.1
’a3ud’, 0.0000 , 0.
’a4ud’, 0.0000 , 0.
’a5ud’, 0.0000 , 0.
’a0du’, 0.0000 , 0.
’a1du’, 0.0000 , 0.
’a2du’, 0.0000 , 0.
’a3du’, 0.0000 , 0.
’a4du’, 0.0000 , 0.
’a0g’ , 2.2200 , 0.
’a1g’ , 0.0000 , 0.
’a2g’ , 4.6500 , 0.1
’a3g’ , 0.0000 , 0.
’a4g’ , 0.0000 , 0.
’a5g’ , 0.0000 , 0.
’kappa’, 0.0000 , 0.
’dum32’, 0. , 0.
’dum33’, 0. , 0.
’dum34’, 0. , 0.
’dum35’, 0. , 0.
’nutev’, 1.000 , 0.1
’dum37’, 0., 0.
’dum38’, 0., 0.
’dum39’, 0.000 , 0.
’dum40’, 0.000 , 0.
’dum41’, 0.000 , 0.
’END’, 0. , 0.
65,5,’f2f3.3021’,36,5.,.001,1.,10.
66,8,’f2f3.3021’,36,5.,.001,1.,10.
999,0,’dummy’,0,0.,0.,0.,0.
0,6,1,5.00,1,0,1,1,0,7,11
/EOF
```

Figure D1: Control file

```
      PROGRAM ALTPFIT
      IMPLICIT REAL*8 (A-H,O-Z)
      CHARACTER*10 IPNAME(60),ITITLE(30),IQUIT
     2,LABEL,LABEL2
      Character*20 output
      DIMENSION DATA(6,2500),V(4),NFLAG(30),TX(200),TQ2(200)
      DIMENSION XQ(200),YDAT(200),ERD(200),Q2TH(200),YTH(200)
     2,YOUT(200),CHI(200),THOUT(200),ITOUT(200),itype(30),errn(30)
      DIMENSION Q2CUT(30),XPOS(50),INORM(30)
     2,RSPOS(15),XLO(30),XHI(30),W2MIN(30)
     3,XMLO(15)
      COMMON/GFUNC/CALC(8,40,25)
      COMMON/MINIMB/CURPAR(60),UNCRT(60),ERMIN(40),PWATE(60),IFREP(40),
     * NPARAM,NVAR,IREJ
      COMMON/MINIMC/IPNAME,PLB(60),PUB(60),TOLRNC,IPAR(60),NSTEPS,
     * IPRNT,IERR,IBND,CHISQR
      COMMON/Q2STUF/ Q02,Q2MAX
      COMMON/INPUT/INS,NVL,NMAX,DELTA,S0,IORD
      Common /Ischeme/ Isch, Iset, Iflg, Ihad
      COMMON/voica/ifred, ITMGP
      COMMON/GRPTHY/FLAVOR
      common/cuts/q2cut,xlo,xhi,w2min
      common/expts/ititle,errn,iserr
      COMMON/CONSTANTS/PI,PI2
      COMMON/DYTEST/XSECLO,GLUCOR
      COMMON/FLAGS/ITYPE,NFLAG,INORM
      DATA IQUIT/'END'/
C
C  FITTING PROGRAM FOR DEEP INELASTIC AND DRELL YAN DATA
C  BASED AROUND THE ALTARELLI-PARISI INTEGRATION SUBROUTINE
c  Modified to use CTEQ6 parametrization form
C  ***************************************************************
C  MAXIMUM OF 60 PARAMETERS ALLOWED. OF THESE, A MAXIMUM OF
C  40 CAN BE VARIED (THIS CAN BE EASILY INCREASED). THE FIRST
C  35 PARAMETERS ARE RESERVED FOR THE PARTON DISTRIBUTIONS. THE
C  OTHER 25 ARE FOR NORMALIZATIONS AND OTHER PARAMETERS WHICH DO
C  NOT AFFECT THE PARTON DISTRIBUTIONS. VARIOUS MODS TO INCREASE
C  THE EFFICIENCY RELY ON THIS SPLIT.
C  ***************************************************************
      OPEN(UNIT=7,FILE='input.dat',STATUS='unknown')
      read(7,*) output
      open(unit=6,file=output,status='unknown')
C
C  SET UP VARIOUS CONSTANTS AND INITIALIZE INTEGRATION SUBROUTINES
C
      PI=4.*DATAN(1.D0)
      PI2=PI**2
      NV=6
      CALL WATE4
      CALL WATE8
      CALL WATE16
      CALL WATE32
      CALL WATE8L
      DELTA=.04
      IERR=0
      IDRV=2
      NSTEPS=10
      TOLRNC=.5
      IPRNT=1
      IPLOT=0
C
C  READ IN FIT PARAMETERS
C
      I=0
   32 I=I+1
      READ(7,*) IPNAME(I),CURPAR(I),PWATE(I)
      IPAR(I)=I
      IF(IPNAME(I).EQ.IQUIT) GO TO 31
      GO TO 32
   31 NPARAM=I-1
C
C  READ IN TYPES OF DATA TO BE FIT AND THE CUTS
C
      I=0
```

```
   33 I=I+1
      READ(7,*) ITYPE(I),NFLAG(I),ITITLE(I),INORM(I),
     $Q2CUT(I),XLO(I),XHI(I),W2MIN(I)
C
C  TERMINATE LIST WITH ITYPE = 999
C
      IF(ITYPE(I).EQ.999) GO TO 34
      if(itype(i).le.100)then
         WRITE(6,38) ITITLE(I),Q2CUT(I),XLO(I),XHI(I),W2MIN(I)
 38      FORMAT(1H ,A10,' DIS data with Q2 greater than',F6.2,
     2   ' and x between ',F5.3,' and ',F5.3,' and W2 >= ',F6.2)
      else
         write(6,39) ititle(i),q2cut(i)
 39      format(1h ,a10,'vector boson data with Q2 greater than',
     2   f7.2)
      endif
      goto 33
 34   NFIT=I-1
C
C  READ IN FLAGS FOR THE FITS
C  INS=1....NON-SINGLET ONLY
C     =0....INCLUDES SOME SINGLET TERMS
C  NVL= NUMBER OF NON-SINGLET ARRAYS TO BE EVOLVED (6 in this version)
C  ARRAYS ARE UV,DV, u+, d+, s+, c+,SINGLET,GLUON
C
C  IORD=1....NEXT TO LEADING ORDER TERMS INCLUDED
C      =0....LEADING ORDER TERMS ONLY
C  ISERR=0...NO SYSTEMATIC ERRORS INCLUDED
C       =1...SYSTEMATIC ERRORS ADDED IN QUADRATURE, WHERE AVAILABLE
cv__________ here set the flags needed in ACOT routine
cv         Iset=11                 !** pdf set-> force by hand to use jeff's pdf
cv         Iflg=0                  !** a comment flag
cv         Ihad=1                  !** hadron
cv         ISCH=1                  !** ISCH=1,7 ->full ACOT, shortcut ACOT
cv ifred=1 => use Fred's calculation
cv ifred=0 => use Jeff's calculation
cv itmgp=1 TM ON
cv itmgp=0 TM OFF
c
      READ(7,*) INS,NVL,IORD,Q02,ISERR,iflg,ihad,itmgp,ifred,isch,iset
      WRITE(6,*),'IORD=',IORD,'; Q02=', Q02,'; TM=',itmgp,
     $      '; IFRED=',ifred,'; ISCH=',isch
      IF(ISERR.NE.0) WRITE(6,1004)
 1004 FORMAT(/' SYSTEMATIC ERRORS ADDED IN QUADRATURE')
C
C  NXPOS IS THE NUMBER OF DIFFERENT X VALUES APPEARING IN THE
C  DIS DATA SET. THE VALUES ARE STORED IN THE ARRAY XPOS.
C  THIS LIST DETERMINES WHICH X VALUES WILL BE PRINTED IN THE
C  OUTPUT TABLES.
C
C  ITYPE= 3  NFLAG = 1  is  bcdms hydrogen
C         4        = 2  is  bcdms deuterium
c         9        = 1  is  h1 f2
c        10        = 1  is  zeus f2
c        51        = 1  is  Nmc F2 p
c        52        = 2  is  Nmc F2 d
c        53        = 9  is  NmcF2r  F2d/F2p (x and Q2)
c        54        = 9  is  NmcF2rX F2d/F2p (x and <Q2>)
c
c        65        = 5  is  ccfr or nutev f2
c        66        = 8  is  ccfr or nutev xf3
c
c       106        = 1  is  e605 dimuon on cu target
c       108        = 1  is  e866 dimuon xf pp
c       ---        = 2  is  e866 dimuon pp or pn
c       125        = 3  is  NA51 (pp-pn)/(pp+pn) dimuon ratio
c       133        = 4  is  e866f pd/2pp dimuon asymmetry
c       126        = 5  is  CDF W lepton asymmetry
C
      call fildat(nfit,npts,mpts,data,q2max)
      call rdxpos(xpos,nxpos,rspos,nrspos,npts,mpts,data,ndis)
c
      IF(IORD) 42,42,43
   42 WRITE(6,44)
```

Figure D2: Main Program; pages 1-2.

```
   44 FORMAT(///,' THIS IS A LEADING ORDER FIT')
      GO TO 45
   43 WRITE(6,46)
   46 FORMAT(///,' THIS IS A NEXT TO LEADING ORDER FIT')
   45 CONTINUE
      CLOSE(7)
      WRITE(6,71) Q02
   71 FORMAT(/,' Q02=',F8.2)
c
c  prepare for fitting
c
      FIT=0.
      DO 66 IF=1,NPARAM
   66 FIT=FIT+PWATE(IF)
      if(fit.eq.0.d0)then
         V(1)=DATA(3,1)
         V(2)=DATA(4,1)
         V(4)=DATA(6,1)
         DUM=THEORY(1,1,V,CURPAR)
         WRITE(6,74)
 74      FORMAT(/,' INPUT PARAMETERS')
         DO 72 L=1,NPARAM
 72      WRITE(6,73) IPNAME(L),CURPAR(L)
 73      FORMAT(1H ,A10,5X,E15.4)
      else
         CALL MINIM(NV,NPTS,MPTS,DATA,IDRV)
      endif
C
C  PRINT OUTPUT NOW.
C  JT=1....PROTON TARGET F2 (MUON OR ELECTRON)
C    =2....ISOSCALAR TARGET F2 (MUON OR ELECTRON)
C    =3....F2 NEUTRINO PROTON
C    =4....F2 NEUTRINO NEUTRON
C    =5....ISOSCALAR TARGET F2 NEUTRINO
C    =6....XF3 NEUTRINO PROTON
C    =7....XF3 NEUTRINO NEUTRON
C    =8....ISOSCALAR TARGET XF3 NEUTRINO
C    =9....F2 NEUTRON/PROTON RATIO
C
      CHITOT=0.
      NEXP=0
      CHITEMP=0.
c
c  print out remaining dis results
c
      DO 50 JT=1,8
      DO 20 J=1,NXPOS
      NEXP=0
      CHITEMP=0.
      DO 21 K=1,NDIS
      IT=DATA(6,K)
      IF(JT.NE.NFLAG(IT)) GOTO 21
      IF(DATA(3,K).NE.XPOS(J)) GO TO 21
      NEXP=NEXP+1
      IT=DATA(6,K)
      IN=INORM(IT)
      FAC=CURPAR(IN)
      XQ(NEXP)=DATA(4,K)
      YDAT(NEXP)=DATA(1,K)*FAC
      ERD(NEXP)=DATA(2,K)*FAC
      ITOUT(NEXP)=DATA(6,K)
      V(1)=DATA(3,K)
      V(2)=DATA(4,K)
      V(4)=IT
      THOUT(NEXP)=FAC*THEORY(1,2,V,CURPAR)
      CHI(NEXP)=((YDAT(NEXP)-THOUT(NEXP))/ERD(NEXP))**2
      CHITEMP=CHITEMP+CHI(NEXP)
      IS=-1
      IF(THOUT(NEXP).GE.YDAT(NEXP)) IS=1
      CHI(NEXP)=IS*CHI(NEXP)
   21 CONTINUE
      IF(NEXP) 23,23,24
   24 CONTINUE
      WRITE(6,22) XPOS(J)
```

```
   22 FORMAT(/,'#', 10X,' X=',F4.3)
      WRITE(6,25)
   25 FORMAT('#',5X,' Q2',5X,'THEORY',6X,'DATA',5X,'ERROR',2X,
     2'CHI SQUARE',3X,'ITYPE')
      WRITE(6,26) (XQ(K),THOUT(K),YDAT(K),ERD(K),CHI(K),
     2ITITLE(ITOUT(K)),K=1,NEXP)
   26 FORMAT(5F10.4,5X,a10)
      WRITE(6,27) CHITEMP,NEXP
   27 FORMAT('#',' CHI SQUARE=',F8.2,' FOR ',I4,' POINTS')
   23 CONTINUE
      CHITOT=CHITOT+CHITEMP
   20 CONTINUE
   50 CONTINUE
c
c  iout=1...m^3 dsig/dm/dy
c  iout=2...m^3 dsig/dm/dxf
c  iout=3...(pp-pn)/(pp+pn)
c  iout=4...pd/2pp
c  iout=5...W asymmetry
c
      ND=NDIS+1
      DO 201 KT=1,5
      DO 203 KRS=1,NRSPOS
      NEXP=0
      CHITEMP=0.
      DO 202 K=ND,NPTS
      IEXP=DATA(6,K)
      IT=ITYPE(IEXP)
      nfl=nflag(iexp)
      IF(nfl.NE.KT) GOTO 202
      IOUT=KT
      TST=RSPOS(KRS)**2
      IF(abs(DATA(3,K)-TST).gt..1) GO TO 202
      NEXP=NEXP+1
      XQ(NEXP)=DATA(4,K)/sqrt(data(3,k))
      YDAT(NEXP)=DATA(1,K)
      ERD(NEXP)=DATA(2,K)
      ITOUT(NEXP)=DATA(6,K)
      YOUT(NEXP)=DATA(5,K)
      V(1)=DATA(3,K)
      V(2)=DATA(4,K)
      V(3)=DATA(5,K)
      V(4)=DATA(6,K)
      THOUT(NEXP)=THEORY(1,2,V,CURPAR)
      CHI(NEXP)=((YDAT(NEXP)-THOUT(NEXP))/ERD(NEXP))**2
      CHITEMP=CHITEMP+CHI(NEXP)
      IS=-1
      IF(THOUT(NEXP).GE.YDAT(NEXP)) IS=1
      CHI(NEXP)=IS*CHI(NEXP)
  202 CONTINUE
      IF(NEXP) 222,222,223
  223 CONTINUE
      WRITE(6,224) RSPOS(KRS)
  224 FORMAT(///,10X,' SQRT(S)=',F8.2)
      GO TO (231,232,231,232,231),IOUT
  231 CONTINUE
      index=v(4)
      if(itype(index).eq.108.or.itype(index).eq.110)then
         write(6,244)
      else
         WRITE(6,241)
      endif
  241 FORMAT(///,' SQRT(TAU)',5X,'Y',6X,'THEORY',6X,'DATA',
     25X,'ERROR',2X,'CHI SQUARE',3X,'ITYPE')
      WRITE(6,242) (XQ(N),YOUT(N),THOUT(N),YDAT(N),ERD(N),CHI(N),
     2ITITLE(ITOUT(N)),N=1,NEXP)
  242 FORMAT(6F10.4,5X,a10)
      GO TO 234
  232 CONTINUE
      WRITE(6,244)
  244 FORMAT(///,' SQRT(TAU)',5X,'XF',5X,'THEORY',6X,'DATA',5X,'ERROR',
     22X,'CHI SQUARE',3X,'ITYPE')
      WRITE(6,245) (YOUT(N),THOUT(N),YDAT(N),ERD(N),CHI(N),
     2ITITLE(ITOUT(N)),N=1,NEXP)
```

Figure D3: Main Program; pages 3-4.

```
 245  FORMAT(6F10.4,5X,a10)
  234 CONTINUE
      WRITE(6,27) CHITEMP,NEXP
      CHITOT=CHITOT+CHITEMP
  222 CONTINUE
  203 CONTINUE
  201 CONTINUE
  204 CONTINUE
      WRITE(6,2001) CHITOT,NPTS
   10 CONTINUE
 2003 FORMAT('  LAMBDA =',F10.3)
 2001 FORMAT(' TOTAL CHI-SQUARE =',F10.3,' FOR ',I4,' POINTS')
 1000 FORMAT(I10,5F10.0,10X,A10)
 1001 FORMAT(' ITYPE=',I2,' X=',F5.3,' Q2=',F7.2,' DATA=',
     $ F7.4,' ERROR=',F7.4,' LABEL=',A10)
 1002 FORMAT(' THERE ARE',I5,' DATA POINTS ENTERED')
 5002 FORMAT(F6.2,8E12.4)
 5001 CONTINUE
 5000 CONTINUE
 5003 FORMAT(//,' Q2=',F6.2)
 5004 FORMAT(/,2X,' X',7X,'XG',10X,'XU',10X,'XD',10X,'XS',
     210X,'XUB',9X,'XDB',9X,'XC ',9X,'XB')
C
      STOP
      END

      subroutine fildat(nfit,npts,mpts,data,q2max)
      implicit real*8 (a-h,o-z)
      character*80 title(4)
      character*10 ititle(30),fln,tmp
      character*20 file1
      dimension data(6,2500), ifit(30),q2cut(30),xlo(30),xhi(30),
     $w2min(30),errn(30),dum(11)
      common/cuts/q2cut,xlo,xhi,w2min
      common/flags/itype(30),nflag(30),inorm(30)
      common/expts/ititle,errn,iserr
      data fln/'./data/'/
      common/correl/EMAT(2,2,200)
      dimension sys(7,2500)
      dimension DMAT(2,2)
      common/datvoica/datnew(2,200),errnew(2,200)
      do inp=1,2
         do jnp=1,2
            DMAT(inp,jnp)=0.
         enddo
      enddo
      npts=0
      q2max=0.d0
      do 100 j=1,nfit
      mpts=0
      call trmstr(fln,len1)
      tmp=ititle(j)
      call trmstr(tmp,len2)
      file1=fln(1:len1)//tmp(1:len2)
      len=len1+len2
      open(unit=8,file=file1(1:len),status='old')
      do 99 l=1,3
 99   read(8,*) title(l)
      read(8,*,err=98) errn(j),ncorr
      goto 98
 97   ncorr=0.
 98   continue
      read(8,*) title(4)
 200  continue
      read(8,*,end=96) (dum(jk),jk=1,11),ak0,ak1
      if(itype(j).le.100)then
         x=dum(1)
         q2=dum(2)
         dat1=dum(3)
         dat2=dum(7)
         if(iserr.eq.0)then
            err1=dum(4)
            err2=dum(8)
         else
```

```
            err1=dum(4)
            err2=dum(8)
         endif
c correlation
         corr=dum(11)*err1*err2
         if(x.lt.xlo(j))goto 200
         if(x.gt.xhi(j))goto 200
         if(q2.lt.q2cut(j))goto 200
         w2=q2*(1.d0/x-1.d0)+.88d0
         if(w2.lt.w2min(j))goto 200
         npts=npts+1
         if (j.eq.2) then
            mpts=mpts+1
            datnew(1,mpts)=dat1
            datnew(2,mpts)=dat2
            errnew(1,mpts)=err1
            errnew(2,mpts)=err2
            dmat(1,1)=err1**2 + dum(5)**2 +dum(6)**2
            dmat(2,2)=err2**2 + dum(9)**2+dum(10)**2
            dmat(1,2)=corr
            dmat(2,1)=corr
C invert the matrix
            CALL MINVMAT(DMAT,2,DET)
            do ii=1,2
               do jj=1,2
                  EMAT(ii,jj,mpts)=DMAT(ii,jj)
                  write(35,*),ii,jj, EMAT(ii,jj,mpts)
               enddo
            enddo
         endif
         if(itype(j).eq.65)then
            dat=dum(3)
            err=dum(4)
         elseif(itype(j).eq.66)then
            dat=dum(7)
            err=dum(8)
         endif
         data(1,npts)=dat
         data(2,npts)=err
         data(3,npts)=x
         data(4,npts)=q2
         data(5,npts)=w2
         data(6,npts)=j
      endif
      if(q2.gt.q2max)q2max=q2
      goto 200
 96   continue
      close(8)
      print*,'number of points', npts,mpts
 100  continue
      return
      end

      SUBROUTINE MINVMAT(ARRAY,NORDER,DET)
      implicit real*8 (a-h,o-z)
      DIMENSION ARRAY(2,2),IK(2),JK(2)
C
C  MATRIX INVERSION ROUTINE FROM BEVINGTON, PAGE 302.
C
      DET=1.
      DO 100 K=1,NORDER
      AMAX=0.D0
   21 DO 30 I=K,NORDER
      DO 30 J=K,NORDER
      IF(DABS(AMAX).GT.DABS(ARRAY(I,J))) GO TO 30
   24 AMAX=ARRAY(I,J)
      IK(K)=I
      JK(K)=J
   30 CONTINUE
      IF(AMAX.NE.0.D0) GO TO 41
   32 DET=0.
      GO TO 140
   41 I=IK(K)
      IF(I-K) 21,51,43
```

Figure D4: Main Program; pages 5-6.

```fortran
  43 DO 50 J=1,NORDER
     SAVE=ARRAY(K,J)
     ARRAY(K,J)=ARRAY(I,J)
  50 ARRAY(I,J)=-SAVE
  51 J=JK(K)
     IF(J-K) 21,61,53
  53 DO 60 I=1,NORDER
     SAVE=ARRAY(I,K)
     ARRAY(I,K)=ARRAY(I,J)
  60 ARRAY(I,J)=-SAVE
  61 DO 70 I=1,NORDER
     IF(I.EQ.K) GO TO 70
  63 ARRAY(I,K)=-ARRAY(I,K)/AMAX
  70 CONTINUE
     DO 80 I=1,NORDER
     DO 80 J=1,NORDER
     IF(I.EQ.K.OR.J.EQ.K) GO TO 80
  75 ARRAY(I,J)=ARRAY(I,J)+ARRAY(I,K)*ARRAY(K,J)
  80 CONTINUE
     DO 90 J=1,NORDER
     IF(J.EQ.K) GO TO 90
  83 ARRAY(K,J)=ARRAY(K,J)/AMAX
  90 CONTINUE
     ARRAY(K,K)=1./AMAX
 100 DET=DET*AMAX
     DO 130 L=1,NORDER
     K=NORDER-L+1
     J=IK(K)
     IF(J.LE.K) GO TO 111
 105 DO 110 I=1,NORDER
     SAVE=ARRAY(I,K)
     ARRAY(I,K)=-ARRAY(I,J)
 110 ARRAY(I,J)=SAVE
 111 I=JK(K)
     IF(I.LE.K) GO TO 130
 113 DO 120 J=1,NORDER
     SAVE=ARRAY(K,J)
     ARRAY(K,J)=-ARRAY(I,J)
 120 ARRAY(I,J)=SAVE
 130 CONTINUE
 140 RETURN
     END


     subroutine rdxpos(xpos,nxpos,rspos,nrspos,npts,mpts,data,ndis)
     implicit real*8 (a-h,o-z)
     dimension data(6,2500),xpos(50),rspos(15)
     common/flags/itype(30),nflag(30),inorm(30)
     nxpos=0
     nrspos=0
     ndis=0
     do 100 j=1,npts
     jt=data(6,j)
     it=itype(jt)
     if(it.le.100)then
        x=data(3,j)
        ndis=ndis+1
        if(nxpos.eq.0)then
           nxpos=1
           xpos(nxpos)=x
        else
           do 90 l=1,nxpos
           if(x.eq.xpos(l))goto 91
90         continue
           nxpos=nxpos+1
           if(nxpos.gt.npts)return
           xpos(nxpos)=x
91         continue
        endif
     else
        rs=sqrt(data(3,j))
        if(nrspos.eq.0)then
           nrspos=1
           rspos(nrspos)=rs
```

```fortran
        else
           do 80 l=1,nrspos
           if(rs.eq.rspos(l))goto 81
80         continue
           nrspos=nrspos+1
           rspos(nrspos)=rs
81         continue
        endif
     endif
100  continue
     return
     end
```

Figure D5: Main Program; pages 7-8.

```
      SUBROUTINE MINIM(NX,NNPTS,MMPTS,DATA,IDRV)
      implicit real*8(a-h,o-z)
      CHARACTER*10 IPNAME(60)
      REAL*8 MODALF
      DIMENSION ALPHA(40,40),MODALF(40,40),BETA(40),COV(40,40)
      DIMENSION OLDPAR(60),DATA(1,1)
      COMMON/ERRORS/COV
      COMMON/AMAT/ALPHA,BETA
      COMMON/MINIMC/IPNAME,PLB(60),PUB(60),TOLRNC,IPAR(60),NSTEPS,
     1IPRNT,IERR,IBND,CHISQR
      COMMON/MINIMB/CURPAR(60),UNCRT(60),ERMIN(40),PWATE(60),IFREP(40),
     1NPARAM,NVAR,IREJ
      DATA FROZ/4HFIX /,BLNK/4H    /,FNOT/4H NOT/
C
C  MINIMIZATION ROUTINE USING THE MARQUARDT ALGORITHM
C  FROM BEVINGTON, PAGE 235. SEE PROGRAM CURFIT.
C  VERSION CREATED BY D.W. DUKE CIRCA 1978.
C  COMMENTS AND SOME MODS ADDED BY J.F. OWENS
C
C INPUT AND OUTPUT UNIT NUMBERS
      MI=5
      MO=6
      FLAMDA=0.001
C COUNT NUMBER OF PARAMETERS VARIED
      NVAR=0
      DO 340 I=1,NPARAM
      IF(PWATE(I).EQ.0.0) GO TO 340
      NVAR=NVAR+1
      IFREP(NVAR)=I
C IFREP IS THE SEQUENCE OF PARAMETER NUMBERS FOR THOSE BEING VARIED
  340 CONTINUE
C CALCULATE CHI SQUARE
      CALL DATSCN(1,CHISQR,NX,NNPTS,MMPTS,DATA)
      IF(IERR.EQ.1) GO TO 1
      IF(IPRNT.EQ.0) GO TO 1
      WRITE(MO,1111)(IP,IPNAME(IP),CURPAR(IP),PWATE(IP),IP=1,NPARAM)
 1111 FORMAT(5X,I5,A10,’ VALUE=’,G14.5,’ WEIGHT=’,G14.5)
      WRITE(MO,1305) TOLRNC,NSTEPS
      WRITE(MO,1300) CHISQR
      WRITE(MO,1310)
    1 DO 100 NOSTEP=1,NSTEPS
      IF(FLAMDA.GT.0.001) FLAMDA=0.001
      IF(FLAMDA.LT.1.E-10) FLAMDA=1.E-10
C SAVE OLD PARAMETER VALUES
      OLDCHI=CHISQR
      DO 5 I=1,NPARAM
    5 OLDPAR(I)=CURPAR(I)
C EVALUATE ALPHA AND BETA
      CALL DATSCN(IDRV,DUM,NX,NNPTS,MMPTS,DATA)
      NTRIES=0
   10 DO 20 I=1,NVAR
      DO 15 J=1,NVAR
   15 MODALF(I,J)=ALPHA(I,J)/ SQRT(ALPHA(I,I)*ALPHA(J,J))
   20 MODALF(I,I)=1.+FLAMDA
C INVERT MODALF
      CALL MINV(MODALF,NVAR,DET)
      NTRIES=NTRIES+1
      NISTEP=NOSTEP
C UPDATE PARAMETERS
      DO 25 I=1,NVAR
      K=IFREP(I)
      DO 25 J=1,NVAR
   25 CURPAR(K)=CURPAR(K)+MODALF(I,J)*BETA(J)/ SQRT(ALPHA(I,I)*
     1ALPHA(J,J))
   32 CALL DATSCN(1,CHISQR,NX,NNPTS,MMPTS,DATA)
      IF(CHISQR.LT.OLDCHI) GO TO 40
      IF(NTRIES.GT.6) GO TO 110
      FLAMDA=10.*FLAMDA
      DO 35 I=1,NPARAM
   35 CURPAR(I)=OLDPAR(I)
      GO TO 10
C CHECK FOR CONVERGENCE
   40 IF((OLDCHI-CHISQR).LT.TOLRNC) GO TO 110
C CHECK NUMBER OF STEPS
```

```
      IF(NOSTEP.EQ.NSTEPS) GO TO 110
      IF(IPRNT.EQ.0) GO TO 100
      ERR=BLNK
      IF(PWATE(1).EQ.0.0) ERR=FROZ
      IF(IPRNT.NE.0) WRITE(MO,1320)NISTEP,CHISQR
      DO 60 I=1,NPARAM
      ERR=BLNK
      IF(PWATE(I).EQ.0.0) ERR=FROZ
      IF(IPRNT.NE.0) WRITE(MO,1330) IPAR(I),IPNAME(I),CURPAR(I),ERR
   60 CONTINUE
  100 FLAMDA=FLAMDA/10.0
  110 IF(IERR.EQ.1) RETURN
C EVALUATE ERRORS AND PRINT OUT RESULTS OF FIT
      CALL DATSCN(IDRV,DUM,NX,NNPTS,MMPTS,DATA)
      DO 120 I=1,NVAR
      DO 120 J=1,I
      MODALF(I,J)=ALPHA(I,J)/ SQRT(ALPHA(I,I)*ALPHA(J,J))
  120 MODALF(J,I)=MODALF(I,J)
      CALL MINV(MODALF,NVAR,DET)
      DO 130 I=1,NVAR
      AMOD=MODALF(I,I)/ALPHA(I,I)
      IMOD=1
      IF(AMOD.LT.0.0) IMOD=-1
  130 UNCRT(IFREP(I))=IMOD*SQRT(IMOD*AMOD)
      CHIPNT=CHISQR/(NNPTS-IREJ-NVAR)
      IF(IPRNT.EQ.0) GO TO 170
      WRITE(MO,1320)NISTEP,CHISQR
      WRITE(MO,1335)(IPAR(I),IPNAME(I),CURPAR(I),UNCRT(I),I=1,NPARAM)
      WRITE(MO,1340) NISTEP
      TOL=BLNK
      IF((OLDCHI-CHISQR).GT.TOLRNC) TOL=FNOT
      WRITE(MO,1350) TOL
      IF(IREJ.NE.0) WRITE(MO,1360) IREJ
  160 WRITE(MO,1345)CHIPNT
      DO 180 I=1,NVAR
      DO 180 J=1,NVAR
  180 MODALF(I,J)=MODALF(I,J)/SQRT(ALPHA(I,I)*ALPHA(J,J))
      DO 182 I=1,NVAR
      DO 181 J=1,NVAR
      COV(I,J)=MODALF(I,J)
  181 ALPHA(I,J)=MODALF(I,J)/SQRT(MODALF(I,I)*MODALF(J,J))
  182 WRITE(MO,1380)(ALPHA(I,J),J=1,NVAR)
      WRITE(MO,1370)
  170 CONTINUE
      RETURN
 1300 FORMAT(28H0THE INITIAL CHI SQUARED IS ,1PE11.4)
 1305 FORMAT(1H0,17HTHE TOLERANCE IS ,F9.4/32H THE MAXIMUM NUMBER OF STE
     1PS IS ,I3)
 1310 FORMAT(33H0  STEP  CHI SQUARED    PARAMETER,6X,5HVALUE,9X,5HERROR)
 1320 FORMAT(1H ,60(1H-)/3H   ,I3,3X,1PE11.4)
 1330 FORMAT(22X,I2,2X,A8,2X,1PE11.4,7X,A4)
 1335 FORMAT(22X,I2,2X,A8,2X,1PE11.4,3X,1PE11.4)
 1340 FORMAT(35H0 MINIMIZATION TERMINATED WITH STEP,I3)
 1345 FORMAT(39H0 CHI SQUARED PER DEGREE OF FREEDOM IS ,1PE10.3)
 1350 FORMAT(22H  TOLERANCE CRITERION ,A4,10H SATISFIED)
 1360 FORMAT(12H0 THERE WERE,I3,40H DATA POINTS WITH ZERO OR NEGATIVE ER
     1ROR)
 1370 FORMAT(/1H0,16X,32H***   ***   ***  ***   ***   ***)
 1380 FORMAT(1X,20F6.3)
      END
      SUBROUTINE DATSCN(MODE,CHISQR,NX,NNPTS,MMPTS,DATA)
      implicit real*8 (a-h,o-z)
      DIMENSION ALPHA(40,40),BETA(40)
      COMMON/MINIMB/CURPAR(60),UNCRT(60),ERMIN(40),PWATE(60),IFREP(40),
     1NPARAM,NVAR,IREJ
      COMMON/AMAT/ALPHA,BETA
      DIMENSION DATA(NX,NNPTS),V(10),DRV(2,40)

cv___start
cv   added for the covariance matrix+looping
      common/tharray/theoryarray(2,100)
      dimension fvalarr(mmpts),fval2arr(2,40,mmpts)
      COMMON/correl/EMAT(2,2,200)
      DIMENSION DRVM(40),VM(10),fval(2)
```

Figure D6: Minimization Program; pages 1-2.

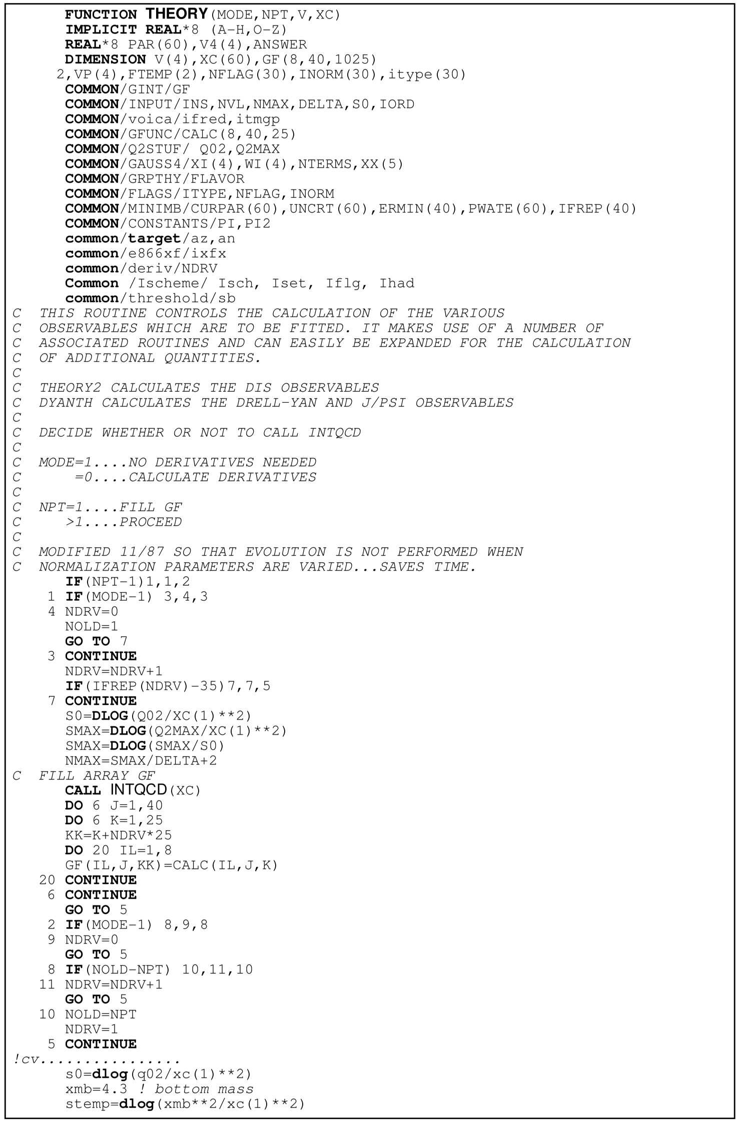
```fortran
      COMMON/datvoica/datnew(2,200),errnew(2,200)

!initialization
      do ipt=1,mmpts
         theoryarray(1,ipt)=0.0
         fvalarr(ipt)=0.0
         theoryarray(2,ipt)=0.0
         fvalarr(ipt)=0.0
      enddo
!storing theory into array
      do ipt=1,nnpts
         v(1)=data(3,ipt)
         v(2)=data(4,ipt)
         v(3)=data(5,ipt)
         v(4)=data(6,ipt)
         if (ipt.le.mmpts) then
            jpt=ipt
            if (v(4).eq.1) then
               theoryarray(1,jpt)=THEORY(1,jPT,V,CURPAR)
            else
               print*,’bad bad bad 1’
            endif
         else
            jpt=ipt-mmpts
            if (v(4).eq.2) then
               theoryarray(2,jpt)=THEORY(1,jPT,V,CURPAR)
            else
               print*,’bad bad bad 2’
            endif
         endif
      enddo


      NN=NX-2

C CALCULATE CHI SQUARE
      IF(MODE.NE.1) GO TO 100
      CHISQR=0.0
      DO 10 NPT=1,mmPTS

         do i=1,2
            do j=1,2
               CHISQR=CHISQR+(DATnew(i,NPT)-theoryarray(i,npt))*
     $              EMAT(i,j,npt)*(DATnew(j,nPT)-theoryarray(j,npt))
               write(36,*),chisqr,i,j,npt
            enddo
         enddo

   10 CONTINUE
      print*,’chisqr’,chisqr

      RETURN
! end chisquare calculation

  100 IREJ=0  ! this is here in order to remove points with negative errors

! store theory array for the case when the CURPAR varries

      DO NPT=1,nnPTS
         DO II=1,NN
            V(II)=DATA(II+2,NPT)
         enddo
         if (npt.le.mmpts) then
            mpt=npt
            if (v(4).eq.1) then
               DO I=1,NVAR
                  K=IFREP(I)
                  SV=CURPAR(K)
                  CURPAR(K)=SV+PWATE(K)
                  fval2arr(1,k,mpt)=THEORY(MODE,mPT,V,CURPAR)
                  CURPAR(K)=SV
               enddo
            else
               print*,’BAAAAAD 1’
```

```fortran
            endif

         else
            mpt=npt-mmpts
            if (v(4).eq.2) then
               DO  I=1,NVAR
                  K=IFREP(I)
                  SV=CURPAR(K)
                  CURPAR(K)=SV+PWATE(K)
                  fval2arr(2,k,mpt)=THEORY(MODE,mPT,V,CURPAR)
                  CURPAR(K)=SV
               enddo
            else
               print*,’BAAAAAD 2’
            endif
         endif

      enddo

C CALCULATE ALPHA AND BETA
!initialization
      DO I=1,NVAR
         BETA(I)=0.0
         do ii=1,2
            DRV(ii,I)=0.0
         enddo
         DO J=1,NVAR
            ALPHA(I,J)=0.0
         enddo
      enddo

!start looping over NPT and MPT
      DO NPT=1,MMPTS
         do ii=1,2
            do jj=1,2

               DO I=1,NVAR
                  K=IFREP(I)
                  SV=CURPAR(K)
                  CURPAR(K)=SV+PWATE(K)
C CALCULATE DERIVATIVE
                  IF(MODE.EQ.3) GO TO 125
C IF MODE EQUALS 3 USER SUPPLIES DERIVATIVES IN ARRAY DRV
C OTHERWISE, PROGRAM CALCULATES DERIVATIVES USING FINITE DIFFERENCE
                  DRV(ii,K)=(fval2arr(ii,k,NPT)-theoryarray(ii,npt))/
     $                 PWATE(K)
                  DRV(jj,K)=(fval2arr(jj,k,NPT)-theoryarray(jj,npt))/
     $                 PWATE(K)
                  CURPAR(K)=SV
               enddo
! beta and alpha
 125           DO I=1,NVAR
                  K=IFREP(I)
                  BETA(I)=BETA(I)+(DATnew(ii,NPT)-theoryarray(ii,npt))*
     $                 EMAT(ii,jj,nPT)*DRV(jj,K)
                  DO J=1,I
                     L=IFREP(J)
                     ALPHA(I,J)=ALPHA(I,J)+DRV(ii,K)*EMAT(ii,jj,NPT)*
     $                    DRV(jj,L)
                     ALPHA(J,I)=ALPHA(I,J)
                  enddo
               enddo
               write(76,*),drv(jj,1),npt,v(1),v(2)

            enddo
         enddo
      enddo
 50   CONTINUE                  ! end npt loop

      RETURN
      END

      SUBROUTINE MINV(ARRAY,NORDER,DET)
      implicit real*8 (a-h,o-z)
```

Figure D7: Minimization Program; pages 3-4.

```
      DIMENSION ARRAY(40,40),IK(40),JK(40)
C
C  MATRIX INVERSION ROUTINE FROM BEVINGTON, PAGE 302.
C
      DET=1.
      DO 100 K=1,NORDER
      AMAX=0.D0
   21 DO 30 I=K,NORDER
      DO 30 J=K,NORDER
      IF(DABS(AMAX).GT.DABS(ARRAY(I,J))) GO TO 30
   24 AMAX=ARRAY(I,J)
      IK(K)=I
      JK(K)=J
   30 CONTINUE
      IF(AMAX.NE.0.D0) GO TO 41
   32 DET=0.
      GO TO 140
   41 I=IK(K)
      IF(I-K) 21,51,43
   43 DO 50 J=1,NORDER
      SAVE=ARRAY(K,J)
      ARRAY(K,J)=ARRAY(I,J)
   50 ARRAY(I,J)=-SAVE
   51 J=JK(K)
      IF(J-K) 21,61,53
   53 DO 60 I=1,NORDER
      SAVE=ARRAY(I,K)
      ARRAY(I,K)=ARRAY(I,J)
   60 ARRAY(I,J)=-SAVE
   61 DO 70 I=1,NORDER
      IF(I.EQ.K) GO TO 70
   63 ARRAY(I,K)=-ARRAY(I,K)/AMAX
   70 CONTINUE
      DO 80 I=1,NORDER
      DO 80 J=1,NORDER
      IF(I.EQ.K.OR.J.EQ.K) GO TO 80
   75 ARRAY(I,J)=ARRAY(I,J)+ARRAY(I,K)*ARRAY(K,J)
   80 CONTINUE
      DO 90 J=1,NORDER
      IF(J.EQ.K) GO TO 90
   83 ARRAY(K,J)=ARRAY(K,J)/AMAX
   90 CONTINUE
      ARRAY(K,K)=1./AMAX
  100 DET=DET*AMAX
      DO 130 L=1,NORDER
      K=NORDER-L+1
      J=IK(K)
      IF(J.LE.K) GO TO 111
  105 DO 110 I=1,NORDER
      SAVE=ARRAY(I,K)
      ARRAY(I,K)=-ARRAY(I,J)
  110 ARRAY(I,J)=SAVE
  111 I=JK(K)
      IF(I.LE.K) GO TO 130
  113 DO 120 J=1,NORDER
      SAVE=ARRAY(K,J)
      ARRAY(K,J)=-ARRAY(I,J)
  120 ARRAY(I,J)=SAVE
  130 CONTINUE
  140 RETURN
      END
```

Figure D8: Minimization Program; pages 5-6.



```
      FUNCTION THEORY(MODE,NPT,V,XC)
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 PAR(60),V4(4),ANSWER
      DIMENSION V(4),XC(60),GF(8,40,1025)
     2,VP(4),FTEMP(2),NFLAG(30),INORM(30),itype(30)
      COMMON/GINT/GF
      COMMON/INPUT/INS,NVL,NMAX,DELTA,S0,IORD
      COMMON/voica/ifred,itmgp
      COMMON/GFUNC/CALC(8,40,25)
      COMMON/Q2STUF/ Q02,Q2MAX
      COMMON/GAUSS4/XI(4),WI(4),NTERMS,XX(5)
      COMMON/GRPTHY/FLAVOR
      COMMON/FLAGS/ITYPE,NFLAG,INORM
      COMMON/MINIMB/CURPAR(60),UNCRT(60),ERMIN(40),PWATE(60),IFREP(40)
      COMMON/CONSTANTS/PI,PI2
      common/target/az,an
      common/e866xf/ixfx
      common/deriv/NDRV
      Common /Ischeme/ Isch, Iset, Iflg, Ihad
      common/threshold/sb
C  THIS ROUTINE CONTROLS THE CALCULATION OF THE VARIOUS
C  OBSERVABLES WHICH ARE TO BE FITTED. IT MAKES USE OF A NUMBER OF
C  ASSOCIATED ROUTINES AND CAN EASILY BE EXPANDED FOR THE CALCULATION
C  OF ADDITIONAL QUANTITIES.
C
C  THEORY2 CALCULATES THE DIS OBSERVABLES
C  DYANTH CALCULATES THE DRELL-YAN AND J/PSI OBSERVABLES
C
C  DECIDE WHETHER OR NOT TO CALL INTQCD
C
C  MODE=1....NO DERIVATIVES NEEDED
C      =0....CALCULATE DERIVATIVES
C
C  NPT=1....FILL GF
C     >1....PROCEED
C
C  MODIFIED 11/87 SO THAT EVOLUTION IS NOT PERFORMED WHEN
C  NORMALIZATION PARAMETERS ARE VARIED...SAVES TIME.
      IF(NPT-1)1,1,2
    1 IF(MODE-1) 3,4,3
    4 NDRV=0
      NOLD=1
      GO TO 7
    3 CONTINUE
      NDRV=NDRV+1
      IF(IFREP(NDRV)-35)7,7,5
    7 CONTINUE
      S0=DLOG(Q02/XC(1)**2)
      SMAX=DLOG(Q2MAX/XC(1)**2)
      SMAX=DLOG(SMAX/S0)
      NMAX=SMAX/DELTA+2
C  FILL ARRAY GF
      CALL INTQCD(XC)
      DO 6 J=1,40
      DO 6 K=1,25
      KK=K+NDRV*25
      DO 20 IL=1,8
      GF(IL,J,KK)=CALC(IL,J,K)
   20 CONTINUE
    6 CONTINUE
      GO TO 5
    2 IF(MODE-1) 8,9,8
    9 NDRV=0
      GO TO 5
    8 IF(NOLD-NPT) 10,11,10
   11 NDRV=NDRV+1
      GO TO 5
   10 NOLD=NPT
      NDRV=1
    5 CONTINUE
!cv................
      s0=dlog(q02/xc(1)**2)
      xmb=4.3 ! bottom mass
      stemp=dlog(xmb**2/xc(1)**2)
```

```
      sb=dlog(stemp/s0)
!cv............
      index=V(4)
      it=itype(index)
      nfl=nflag(index)
      IF(IT.LT.100) GO TO 33
C  AS CURRENTLY SET UP ITYPE < 100 CORRESPONDS TO DEEP
C  INELASTIC SCATTERING
   33 CONTINUE
C  DIS STRUCTURE FUNCTIONS CALCULATED HERE
      xdef=v(1)
      q2def=v(2)
      csi=2.*xdef/(1.+sqrt(1d0+(4d0*0.938d0**2*xdef**2/q2def)))
      if (ifred.eq.0) then ! Jeff's calculation
         if (ITMGP.eq.1) then   !TMC ON
            CALL THEORY2(NDRV,V,XC,ANS1_def)
            v(1)=csi
            CALL THEORY2(NDRV,V,XC,ANS1_csi)
!calculate the integral terms of the TMC (using Gauss quadrature)
            if (nflag(index).eq.8) then
               A1=0.
               DO Itm=1,NTERMS
                  XIP=.5*(1.-csi)*XI(Itm)+.5*(1.+csi)
                  V(1)=XIP
                  CALL THEORY2(NDRV,V,XC,ans_prime)
                  A1=A1+.5*(1-csi)*WI(Itm)*ans_prime/XIP**2
               ENDDO            ! I LOOP
               v(1)=xdef
               ANS1=(xdef/csi)**2/(1.+
     $              (4.*0.938**2*xdef**2/q2def))*ANS1_csi+
     $              2.*0.938**2*xdef**3*(1/sqrt(1.+
     $              (4.*0.938**2*xdef**2/q2def)))**3/
     $              q2def*a1
               elseif (nflag(Index).eq.5) then
               A1=0.
               A2=0.
               DO Itm=1,NTERMS
                  XIP=.5*(1.-csi)*XI(Itm)+.5*(1.+csi)
                  V(1)=XIP
                  CALL THEORY2(NDRV,V,XC,ans_prime)
                  A1=A1+.5*(1-csi)*WI(Itm)*ans_prime/XIP**2
                  A2=A2+.5*(1-csi)*WI(Itm)*(XIP-csi)*ans_prime/XIP**2
               ENDDO            ! I LOOP
               v(1)=xdef
               ANS1=(Xdef/csi)**2/(sqrt(1.+
     $              (4.*0.938**2*xdef**2/q2def)))**3*ANS1_csi+
     $              6.*0.938**2*xdef**3./Q2def/(sqrt(1.+
     $              (4.*0.938**2*xdef**2/q2def)))**4*A1
     $            +12.*0.938**4*xdef**4/Q2def**2/(sqrt(1.+
     $              (4.*0.938**2*xdef**2/q2def)))**5*A2
            endif
         else
            CALL THEORY2(NDRV,V,XC,ANS1)
         endif
         V(1)=xdef
      else
         if (ITMGP.eq.1) then !TMC ON
            call THEORY_FO(NDRV,V,XC,ANS2_def)
            v(1)=csi
            call THEORY_FO(NDRV,V,XC,ANS2_csi)
            if (nflag(index).eq.8) then
!calculate the integral terms of the TMC (using Gauss quadrature)
               A1=0.
               DO Itm=1,NTERMS
                  XIP=.5*(1.-csi)*XI(Itm)+.5*(1.+csi)
                  V(1)=XIP
                  CALL THEORY_FO(NDRV,V,XC,ans2_prime)
                  A1=A1+.5*(1-csi)*WI(Itm)*ans2_prime/XIP
               ENDDO
               v(1)=xdef
               ANS2=(xdef/csi)/(1.+
     $              (4.*0.938**2*xdef**2/q2def))*ANS2_csi+
     $              2.*0.938**2*xdef**2*(1/sqrt(1.+
     $              (4.*0.938**2*xdef**2/q2def)))**3/
```

Figure D9: Theory $\overline{MS}$ Program; pages 1-2.

```
     $              q2def*a1
            ANS2=xdef*ANS2
         elseif(nflag(index).eq.5) then
            A1=0.
            A2=0.
            DO Itm=1,NTERMS
               XIP=.5*(1.-csi)*XI(Itm)+.5*(1.+csi)
               V(1)=XIP
               CALL THEORY_FO(NDRV,V,XC,ans2_prime)
               A1=A1+.5*(1-csi)*WI(Itm)*ans2_prime/XIP**2
               A2=A2+.5*(1-csi)*WI(Itm)*(XIP-csi)*ans2_prime/XIP**2
            ENDDO            ! I LOOP
            v(1)=xdef
            ANS2=(Xdef/csi)**2/(sqrt(1.+
     $              (4.*0.938**2*xdef**2/q2def)))**3*ANS2_csi+
     $              6.*0.938**2*xdef**3./Q2def/(sqrt(1.+
     $              (4.*0.938**2*xdef**2/q2def)))**4*A1
     X              +12.*0.938**4*xdef**4/Q2def**2/(sqrt(1.+
     $              (4.*0.938**2*xdef**2/q2def)))**5*A2
         endif
      else
 987     continue
         v(1)=xdef
         call THEORY_FO(NDRV,V,XC,ANS2)
         if (nflag(index).eq.5) then
            ANS2=ANS2
         elseif (nflag(index).eq.8) then
            ANS2=xdef*ANS2
         endif
      endif
      endif
      if (ifred.eq.1) then
         ANS=ANS2
      else
         ANS=ANS1
      endif
      IN=INORM(index)
      THEORY=ANS/XC(IN)
      RETURN
      END
cv__________________ construct the interface to call Fred's THEORY
      SUBROUTINE THEORY_FO(NDRV,V,XC,THEORY)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION V(4),XC(60),FTEMP(2)
      COMMON/GAUS16/XI(16),WI(16),NTERMS,XX(17)
      COMMON/GRPTHY/FLAVOR
      COMMON/INPUT/INS,NVL,NMAX,DELTA,S0,IORD
      COMMON/Q2STUF/Q02,Q2MAX
      common/flags/itype(30),nflag(30),inorm(30)
      COMMON/CONSTANTS/PI,PI2
      COMMON/lambda/xlamfred(60)
cv____ here call fred's calculations of SF's
cv____ call to pdf's is made in fredpdf.f
      DIMENSION F123n(3),F123nb(3)
      DO i=1,60
         xlamfred(i)=XC(i)
      enddo
      X=V(1)
      Q2=V(2)
      jt=V(4)
      it=itype(jt)
      IFL=8
      IF(INS) 25,25,26
   25 IFL=NFLAG(JT)
   26 CONTINUE
      Q=sqrt(Q2)
cv_________ Fcc123(icharge, mode, x,q,xmu,F123(i))
cv......... icharge = 1(-1) -->W+(W-) boson
cv......... mode = 1 --> total SF
cv......... xmu = set to q (mu scale)
cv......... F123-->F123(i)-->F123(1) = F1; F123(2)=F2; F123(3)=F3
cv_________ calling twice because DATA=(nu+nubar)/2
      CALL Fcc123(1,1, x, Q,Q, F123n)
      CALL Fcc123(-1,1, x, Q,Q, F123nb)
```

```
cv________ here select F3 or F2
      if (ifl.eq.8) then ! f3
         THEORY=(F123n(3)+F123nb(3))/2
      elseif (ifl.eq.5) then !f2
         THEORY=(F123n(2)+F123nb(2))/2
      endif
      RETURN
      END
      SUBROUTINE THEORY2(NDRV,V,XC,THEORY)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION V(4),XC(60),FTEMP(2)
      COMMON/GAUS16/XI(16),WI(16),NTERMS,XX(17)
      COMMON/GRPTHY/FLAVOR
      COMMON/INPUT/INS,NVL,NMAX,DELTA,S0,IORD
      COMMON/Q2STUF/Q02,Q2MAX
      common/flags/itype(30),nflag(30),inorm(30)
      COMMON/CONSTANTS/PI,PI2
C  CALCULATES DEEP INELASTIC OBSERVABLES
C  FETCH IS USED TO UNFOLD THE PROPER QUARK DISTRIBUTIONS
C  FROM THOSE WHICH ARE EVOLVED BY INTQCD.
C  FOR THE HIGHER ORDER CALCULATION THE CONVOLUTION WITH
C  THE COEFFICIENT FUNCTIONS IS HANDLED HERE. THE CONVENTIONS
C  FOR THE GLUON COEFFICIENT FUNCTION ARE EXPLAINED BELOW.
      X=V(1)
      Q2=V(2)
      jt=V(4)
      it=itype(jt)
      IFL=8
      IF(INS) 25,25,26
   25 IFL=NFLAG(JT)
   26 CONTINUE
      S0=DLOG(Q02/XC(1)**2)
      S=DLOG(DLOG(Q2/XC(1)**2)/S0)
      CALL FETCH(IFL,NDRV,X,S,FTEMP,xc(1))
      THEORY=FTEMP(1)
      IF(IORD.EQ.0) RETURN
      IF3=0
      IF(NFLAG(JT).LE.2.OR.NFLAG(JT).EQ.9) FAC=10./9.
      IF(NFLAG(JT).EQ.6.OR.NFLAG(JT).EQ.7.OR.
     2NFLAG(JT).EQ.8) IF3=1
      CF=4./3.
      AL1=DLOG(1.-X)
      al=alpha_s(iord+1,q2,xc(1),neff)/(4.*pi)
      flavor=neff
      FAC=FLAVOR
      FX=THEORY
      THEORY=FX+FX*AL*CF*(-9.-2.*PI2/3.+AL1*(-3.+2.*AL1))
      DO 23 I=1,NTERMS
      Y=0.5*(1.-X)*XI(I)+0.5*(1.+X)
      XY=X/Y
      AL1=DLOG(1.-Y)
      CALL FETCH(IFL,NDRV,XY,S,FTEMP,xc(1))
      C22=CF*(6.+4.*Y-2.*(1.+Y*Y)/(1.-Y)*DLOG(Y)-2.*(1.+Y)*AL1
     2-IF3*2.*(1.+Y))
      C23=CF*(-3.+4.*AL1)/(1.-Y)
      CG2=2.*FAC*(-1.+8.*Y*(1.-Y)+(1.-2.*Y+2.*Y*Y)*DLOG(1./Y-1.))
C     THE ABOVE GLUON COEFFICIENT FUNCTION CORRESPONDS TO THE
C     CONVENTIONS OF FLORATOS,HERROD,WADA,ETC. THE FOLLOWING
C     EXPRESSION CORRESPONDS TO THE CONVENTION OF ALTARELLI,
C     ELLIS,AND MARTINELLI.
C     CG2=2.*FAC*(6.*Y*(1.-Y)+(1.-2.*Y+2.*Y*Y)*DLOG(1./Y-1.))
      THEORY=THEORY+.5*(1.-X)*WI(I)*AL*(C22*FTEMP(1)+C23*(FTEMP(1)-FX))
      IF(IF3.EQ.1.OR.INS.EQ.1) GOTO 23
   24 CONTINUE
      THEORY=THEORY+.5*(1.-X)*WI(I)*AL*CG2*FTEMP(2)
   23 CONTINUE
   21 CONTINUE
      RETURN
      END
      SUBROUTINE FETCH(IFL,NDRV,X,S,FTEMP,alam)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FTEMP(2)
      COMMON/INPUT/INS,NVL,NMAX,DELTA,S0,IORD
      COMMON/GRPTHY/FLAVOR
```

Figure D10: Theory $\overline{MS}$ Program; pages 3-4.

```fortran
C  UNFOLDS QUARK AND GLUON DISTRIBUTIONS FOR USE IN THEORY2.
      IF(IORD.NE.0.AND.IFL.NE.8) CALL GINTERP(8,NDRV,X,S,FTEMP(2))
      IF(IFL.EQ.5) THEN
         CALL GINTERP(7,NDRV,X,S,FTEMP(1))
         RETURN
      ELSE IF(IFL.EQ.8) THEN
         CALL GINTERP(1,NDRV,X,S,TMPUV)
         CALL GINTERP(2,NDRV,X,S,TMPDV)
         FTEMP(1)=TMPUV+TMPDV
         RETURN
      ELSE
         call pdfjo(ndrv,x,s,u,d,ub,db,sb,cb,bb,glue)
         upubar=u+ub
         dpdbar=d+db
         spsbar=2.*sb
         cpcbar=2.*cb
         bpbbar=2.*bb
      ENDIF
      GOTO (10,20) IFL
   10 FTEMP(1)=(4.*(upubar+cpcbar)+dpdbar+spsbar+bpbbar)/9.
      RETURN
   20 FTEMP(1)=(5.*(upubar+dpdbar)+2.*spsbar+8.*cpcbar+2.*bpbbar)/18.
      RETURN
C  IF NEEDED, HERE IS WHERE NEUTRON AND PROTON NEUTRINO
C  OBSERVABLES CAN BE ADDED IN (IFL=3,4,6, OR 7)
      END
      SUBROUTINE PDFJO(NDRV,X,S,U,D,UB,DB,SB,CB,BB,GLUE)
      IMPLICIT REAL*8 (A-H,O-Z)
      common/threshold/sbbar
      CALL GINTERP(1,NDRV,X,S,UV)
      CALL GINTERP(2,NDRV,X,S,DV)
      CALL GINTERP(3,NDRV,X,S,UPLUS)
      CALL GINTERP(4,NDRV,X,S,DPLUS)
      CALL GINTERP(5,NDRV,X,S,SPLUS)
      call ginterp(6,ndrv,x,s,cplus)
      CALL GINTERP(7,NDRV,X,S,SING)
      CALL GINTERP(8,NDRV,X,S,GLUE)
      flavor=5.
      if(s.lt.sbbar)flavor=4.
      if(flavor.eq.4.)then
         bb=0.
      else if(flavor.eq.5.)then
         bplus=-(uplus+dplus+splus+cplus)
         bb=.5*(bplus+sing/flavor)
      endif
      cb=.5*(cplus+sing/flavor)
      sb=.5*(splus+sing/flavor)
      db=.5*(dplus-dv+sing/flavor)
      ub=.5*(uplus-uv+sing/flavor)
      u=uv+ub
      d=dv+db
      RETURN
      END
      SUBROUTINE GINTERP(I,NDRV,X,S,ANS)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION F1(25),F2(25)
      COMMON/GAUS16/XI(16),WI(16),NTERMS,XX(17)
      COMMON/INPUT/INS,NVL,NMAX,DELTA,S0,IORD
      COMMON/MINIMB/CURPAR(60),UNCRT(60),ERMIN(40),PWATE(60),IFREP(40)
      COMMON/GINT/GF(8,40,1025)
      common/threshold/sb
C  THIS ROUTINE INTERPOLATES AS NEEDED IN THE ARRAY GF TO OBTAIN THE
C  EVOLVED DISTRIBUTIONS AT THE REQUIRED Q2 AND X VALUES. NOTE THAT ONLY
C  40 PARAMETERS CAN BE VARIED IN A GIVEN FIT. OF THESE, AT MOST 35 CAN
C  BE PARAMETERS RELATED TO THE PARTON DISTRIBUTIONS. THIS IS GOVERNED BY THE
C  THIRD DIMENSION OF GF (900 ABOVE). THIS IS 25+25*(NO. OF VARIED PARTON
C  PARAMETERS).
      IS=S/DELTA+1
      isign=1
c  Adjust so that interpolation is not done across the b threshold
      if(s.lt.sb.and.(s+delta).gt.sb)isign=-1
      IS1=IS+isign
      if((is.gt.40).or.(is1.gt.40)) print*,'ALARM!!!!! overflow',is,is1
      if(s.gt.sb.and.(delta*(is-1)).lt.sb)then
```

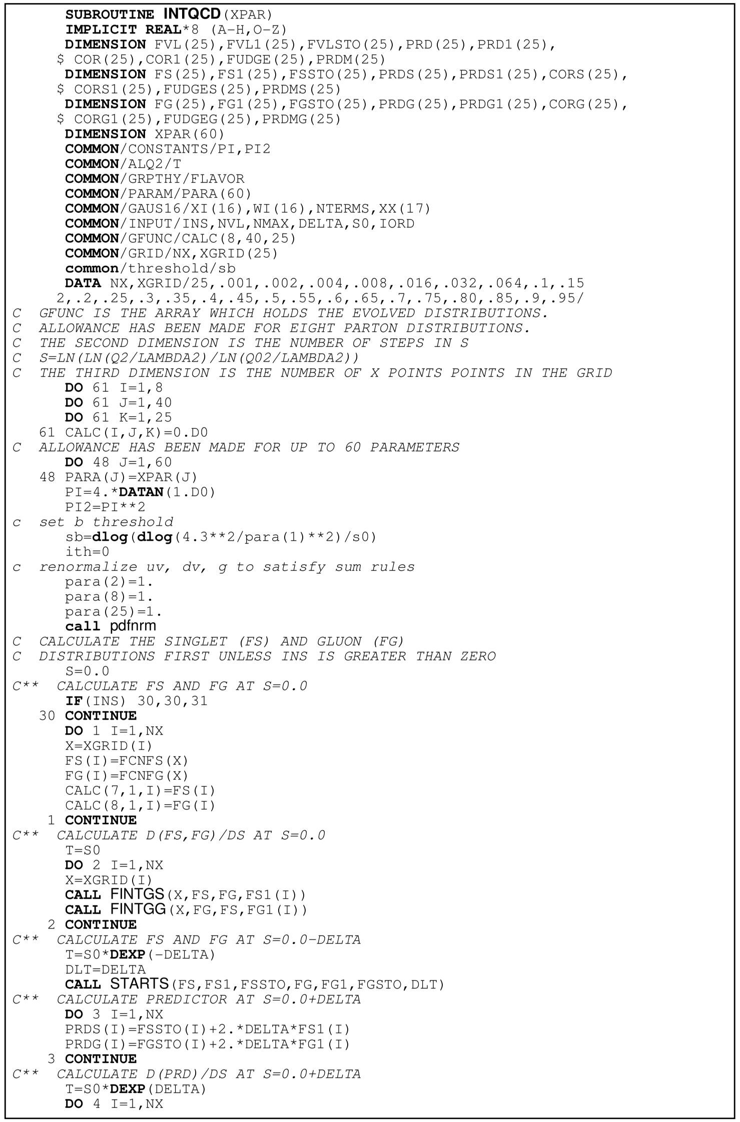
```fortran
         is=is+1
         is1=is+1
      endif
      NDRV2=NDRV
      IF(IFREP(NDRV).GT.35) NDRV2=0
      DO 1 L=1,25
      KL=L+25*NDRV2
      F1(L)=GF(I,IS,KL)
      F2(L)=GF(I,IS1,KL)
    1 CONTINUE
      A1=GETFV(X,F1)
      A2=GETFV(X,F2)
      S1=(IS-1)*DELTA
      S2=S1+DELTA*isign
      ANS=A1*(S-S2)/(S1-S2)+A2*(S-S1)/(S2-S1)
      RETURN
      END
      SUBROUTINE WATE4
      implicit real*8 (a-h,o-z)
      COMMON/GAUSS4/XI(4),WI(4),NTERMS,XX(5)
      NTERMS=4
C   4 POINT GAUSSIAN QUADRATURE
      XI(1)=-.8611363116
      XI(2)=-.3399810436
      XI(3)= .3399810436
      XI(4)= .8611363116
      WI(1)=.3478548451
      WI(2)=.6521451549
      WI(3)=.6521451549
      WI(4)=.3478548451
      DO 1 J=1,4
    1 XX(J)=.5*(XI(J)+1.)
      XX(5)=1.
      RETURN
      END
      FUNCTION BETA(X1,X2)
      IMPLICIT REAL*8 (A-H,O-Z)
C  EULER BETA FUNCTION SUBROUTINE
      CALL GAMMA(X1,G1,IER)
      CALL GAMMA(X2,G2,IER)
      X3=X1+X2
      CALL GAMMA(X3,G3,IER)
      BETA=G1*G2/G3
      RETURN
      END
```

Figure D11: Theory $\overline{MS}$ Program; pages 5-6.

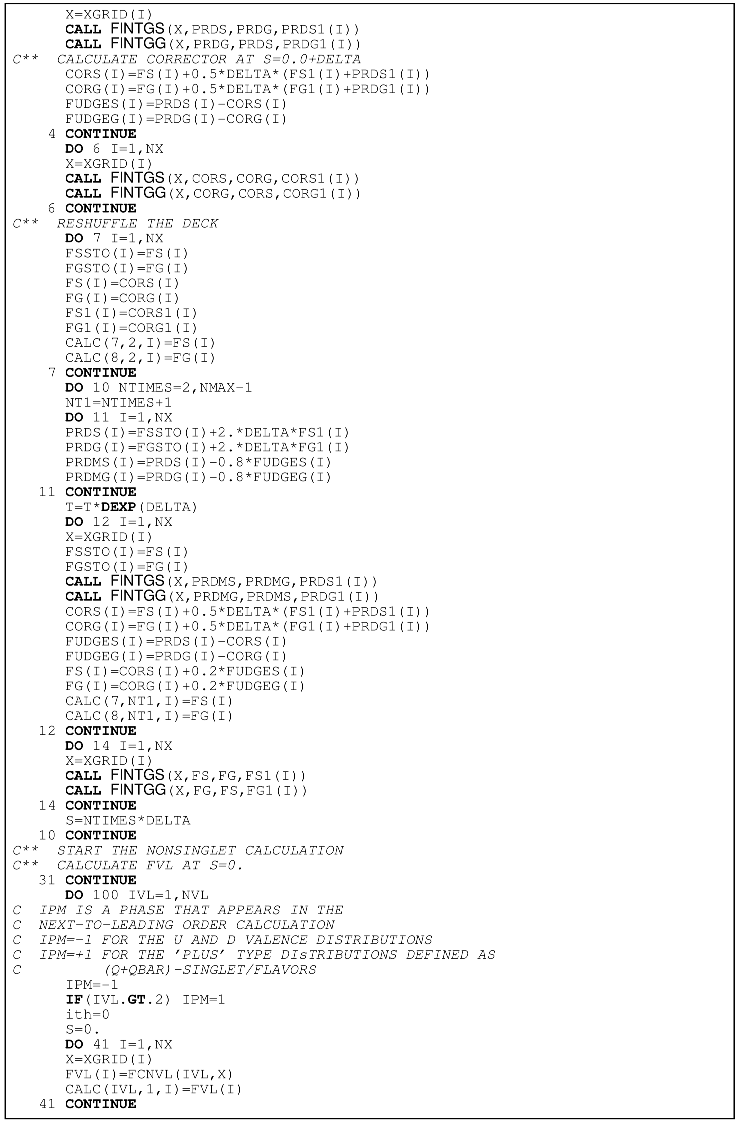
```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C  FUNCTION TO RETURN STRONG COUPLING ALPHA_S=G^2/4/PI         C
C  ---->  WITH INTERPOLATION ACROSS THRESHOLDS  <----          C
C  INPUTS:                                                     C
C         IALPHAS                                              C
C                 = 0: CONSTANT (0.2)                          C
C                 = 1: ONE LOOP                                C
C                 = 2: TWO LOOP                                C
C         MU2                                                  C
C                 = RENORMALIZATION SCALE SQUARED              C
C         LQCD4                                                C
C                 = QCD SCALE FOR NF=4                         C
C  OUTPUT:                                                     C
C         DOUBLE PRECISION ALPHAS                              C
C         NF = EFFECTIVE NUMBER OF FLAVORS                     C
c                                                              c
c  modified 1/18/06 by J.F. Owens to use Lambda(nf=4) with     c
c  the same matching conditions                                c
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      FUNCTION ALPHA_S(IALPHAS,XMU2,XLQCD4,NEFF)
cv      IMPLICIT DOUBLE PRECISION ( A-H,O-Z )
      IMPLICIT real*8 ( A-H,O-Z )
      ALPHA_S = 0.2
      NEFF=4
      IF( IALPHAS.EQ.0 )RETURN
      ALPHA_S = ALFAS4(XMU2,XLQCD4,IALPHAS,NEFF)
c      IF ((XMU2.gt.10.).and.(XMU2.lt.35.)) print*,'alf',alpha_S,xmu2,
c     $     neff
      RETURN
      END
C.------------------------------------------------------------.C
C. FUNCTION TO RETURN ALPHA_S GIVEN LAMBDA_QCD^4 AND SCALE Q2 .C
C. AND ILOOP=1 (ONE LOOP) OR ILOOP=2 (TWO LOOP)               .C
C. SEE G. ALTARELLI ET AL. NUCL. PHYS. B308 (1988) 724.       .C
C. EQNS (7)->(13)                                             .C
C.------------------------------------------------------------.C
      FUNCTION ALFAS4(Q2,ALQCD4,ILOOP,NEFF)
cv      IMPLICIT DOUBLE PRECISION(A-Z)
      IMPLICIT real*8(A-Z)
      INTEGER ILOOP,NEFF
      A  = 1.00
c      MB = 4.750
      mb=4.3
c       mb=4.5
c      MC = 1.50
      mc=1.4
c      mc=1.3
      AMB=A*MB
      AMC=A*MC
      MU=SQRT(Q2)
      ALQCD42=ALQCD4*ALQCD4
      IF      (MU.GT.AMC.and.mu.lt.amb) THEN
         ALFAS4 = ASLOOP(Q2,ALQCD42,4,ILOOP)
C. MUST RETURN NF=4 FOR RENORMALIZATION GROUP TO WORK
         NEFF=4
      ELSE IF (MU.GT.AMb) THEN
         AMB2=AMB*AMB
         ALFAS4 = 1 / ( 1 / ASLOOP(Q2  ,ALQCD42,5,ILOOP) +
     &                  1 / ASLOOP(AMB2,ALQCD42,4,ILOOP) -
     &                  1 / ASLOOP(AMB2,ALQCD42,5,ILOOP) )
C. MUST RETURN NF=5 FOR RENORMALIZATION GROUP TO WORK
         NEFF=5
      ELSE
         AMB2=AMB*AMB
         AMC2=AMC*AMC
         ALFAS4 = 1 / ( 1 / ASLOOP(Q2  ,ALQCD42,3,ILOOP) +
     &                  1 / ASLOOP(AMC2,ALQCD42,4,ILOOP) -
     &                  1 / ASLOOP(AMC2,ALQCD42,3,ILOOP) )
C. MUST RETURN NF=3 FOR RENORMALIZATION GROUP TO WORK
         NEFF=3
      ENDIF
      RETURN
      END
C.--------------------------------------------------------------.C
```

```
C. FUNCTION TO RETURN ALPHAS GIVEN MU2,LQCD2,NF AND NORDER      .C
C.------------------------------------------------------------.C
      FUNCTION ASLOOP(XMU2,XLQCD2,NF,NORDER)
cv      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT real*8 (A-H,O-Z)
      PARAMETER (PI=3.14159)
      IF     ( NORDER.EQ.1 ) THEN
         B1=(33.-2.*NF)/(12.*PI)
         T=XMU2/XLQCD2
         ASLOOP=1/(B1*LOG(T))
      ELSEIF( NORDER.EQ.2 ) THEN
         B1=(33.-2.*NF)/(12.*PI)
         B2=(153.-19.*NF)/(2.*PI*(33.-2.*NF))
         T=XMU2/XLQCD2
         F1=B1*LOG(T)
         ASLOOP=(1.-B2*LOG(LOG(T))/F1)/F1
      ELSE
         WRITE(*,*) 'ERROR IN ASLOOP: NORDER = ',NORDER
         STOP
      ENDIF
      RETURN
      END
```

Figure D12: Alpha_S Program.



```
      SUBROUTINE INTQCD(XPAR)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FVL(25),FVL1(25),FVLSTO(25),PRD(25),PRD1(25),
     $ COR(25),COR1(25),FUDGE(25),PRDM(25)
      DIMENSION FS(25),FS1(25),FSSTO(25),PRDS(25),PRDS1(25),CORS(25),
     $ CORS1(25),FUDGES(25),PRDMS(25)
      DIMENSION FG(25),FG1(25),FGSTO(25),PRDG(25),PRDG1(25),CORG(25),
     $ CORG1(25),FUDGEG(25),PRDMG(25)
      DIMENSION XPAR(60)
      COMMON/CONSTANTS/PI,PI2
      COMMON/ALQ2/T
      COMMON/GRPTHY/FLAVOR
      COMMON/PARAM/PARA(60)
      COMMON/GAUS16/XI(16),WI(16),NTERMS,XX(17)
      COMMON/INPUT/INS,NVL,NMAX,DELTA,S0,IORD
      COMMON/GFUNC/CALC(8,40,25)
      COMMON/GRID/NX,XGRID(25)
      common/threshold/sb
      DATA NX,XGRID/25,.001,.002,.004,.008,.016,.032,.064,.1,.15
     2,.2,.25,.3,.35,.4,.45,.5,.55,.6,.65,.7,.75,.80,.85,.9,.95/
C  GFUNC IS THE ARRAY WHICH HOLDS THE EVOLVED DISTRIBUTIONS.
C  ALLOWANCE HAS BEEN MADE FOR EIGHT PARTON DISTRIBUTIONS.
C  THE SECOND DIMENSION IS THE NUMBER OF STEPS IN S
C  S=LN(LN(Q2/LAMBDA2)/LN(Q02/LAMBDA2))
C  THE THIRD DIMENSION IS THE NUMBER OF X POINTS POINTS IN THE GRID
      DO 61 I=1,8
      DO 61 J=1,40
      DO 61 K=1,25
   61 CALC(I,J,K)=0.D0
C  ALLOWANCE HAS BEEN MADE FOR UP TO 60 PARAMETERS
      DO 48 J=1,60
   48 PARA(J)=XPAR(J)
      PI=4.*DATAN(1.D0)
      PI2=PI**2
c  set b threshold
      sb=dlog(dlog(4.3**2/para(1)**2)/s0)
      ith=0
c  renormalize uv, dv, g to satisfy sum rules
      para(2)=1.
      para(8)=1.
      para(25)=1.
      call pdfnrm
C  CALCULATE THE SINGLET (FS) AND GLUON (FG)
C  DISTRIBUTIONS FIRST UNLESS INS IS GREATER THAN ZERO
      S=0.0
C**  CALCULATE FS AND FG AT S=0.0
      IF(INS) 30,30,31
   30 CONTINUE
      DO 1 I=1,NX
      X=XGRID(I)
      FS(I)=FCNFS(X)
      FG(I)=FCNFG(X)
      CALC(7,1,I)=FS(I)
      CALC(8,1,I)=FG(I)
    1 CONTINUE
C**  CALCULATE D(FS,FG)/DS AT S=0.0
      T=S0
      DO 2 I=1,NX
      X=XGRID(I)
      CALL FINTGS(X,FS,FG,FS1(I))
      CALL FINTGG(X,FG,FS,FG1(I))
    2 CONTINUE
C**  CALCULATE FS AND FG AT S=0.0-DELTA
      T=S0*DEXP(-DELTA)
      DLT=DELTA
      CALL STARTS(FS,FS1,FSSTO,FG,FG1,FGSTO,DLT)
C**  CALCULATE PREDICTOR AT S=0.0+DELTA
      DO 3 I=1,NX
      PRDS(I)=FSSTO(I)+2.*DELTA*FS1(I)
      PRDG(I)=FGSTO(I)+2.*DELTA*FG1(I)
    3 CONTINUE
C**  CALCULATE D(PRD)/DS AT S=0.0+DELTA
      T=S0*DEXP(DELTA)
      DO 4 I=1,NX
```



```
      X=XGRID(I)
      CALL FINTGS(X,PRDS,PRDG,PRDS1(I))
      CALL FINTGG(X,PRDG,PRDS,PRDG1(I))
C**  CALCULATE CORRECTOR AT S=0.0+DELTA
      CORS(I)=FS(I)+0.5*DELTA*(FS1(I)+PRDS1(I))
      CORG(I)=FG(I)+0.5*DELTA*(FG1(I)+PRDG1(I))
      FUDGES(I)=PRDS(I)-CORS(I)
      FUDGEG(I)=PRDG(I)-CORG(I)
    4 CONTINUE
      DO 6 I=1,NX
      X=XGRID(I)
      CALL FINTGS(X,CORS,CORG,CORS1(I))
      CALL FINTGG(X,CORG,CORS,CORG1(I))
    6 CONTINUE
C**  RESHUFFLE THE DECK
      DO 7 I=1,NX
      FSSTO(I)=FS(I)
      FGSTO(I)=FG(I)
      FS(I)=CORS(I)
      FG(I)=CORG(I)
      FS1(I)=CORS1(I)
      FG1(I)=CORG1(I)
      CALC(7,2,I)=FS(I)
      CALC(8,2,I)=FG(I)
    7 CONTINUE
      DO 10 NTIMES=2,NMAX-1
      NT1=NTIMES+1
      DO 11 I=1,NX
      PRDS(I)=FSSTO(I)+2.*DELTA*FS1(I)
      PRDG(I)=FGSTO(I)+2.*DELTA*FG1(I)
      PRDMS(I)=PRDS(I)-0.8*FUDGES(I)
      PRDMG(I)=PRDG(I)-0.8*FUDGEG(I)
   11 CONTINUE
      T=T*DEXP(DELTA)
      DO 12 I=1,NX
      X=XGRID(I)
      FSSTO(I)=FS(I)
      FGSTO(I)=FG(I)
      CALL FINTGS(X,PRDMS,PRDMG,PRDS1(I))
      CALL FINTGG(X,PRDMG,PRDMS,PRDG1(I))
      CORS(I)=FS(I)+0.5*DELTA*(FS1(I)+PRDS1(I))
      CORG(I)=FG(I)+0.5*DELTA*(FG1(I)+PRDG1(I))
      FUDGES(I)=PRDS(I)-CORS(I)
      FUDGEG(I)=PRDG(I)-CORG(I)
      FS(I)=CORS(I)+0.2*FUDGES(I)
      FG(I)=CORG(I)+0.2*FUDGEG(I)
      CALC(7,NT1,I)=FS(I)
      CALC(8,NT1,I)=FG(I)
   12 CONTINUE
      DO 14 I=1,NX
      X=XGRID(I)
      CALL FINTGS(X,FS,FG,FS1(I))
      CALL FINTGG(X,FG,FS,FG1(I))
   14 CONTINUE
      S=NTIMES*DELTA
   10 CONTINUE
C**  START THE NONSINGLET CALCULATION
C**  CALCULATE FVL AT S=0.
   31 CONTINUE
      DO 100 IVL=1,NVL
C  IPM IS A PHASE THAT APPEARS IN THE
C  NEXT-TO-LEADING ORDER CALCULATION
C  IPM=-1 FOR THE U AND D VALENCE DISTRIBUTIONS
C  IPM=+1 FOR THE 'PLUS' TYPE DIsTRIBUTIONS DEFINED AS
C          (Q+QBAR)-SINGLET/FLAVORS
      IPM=-1
      IF(IVL.GT.2) IPM=1
      ith=0
      S=0.
      DO 41 I=1,NX
      X=XGRID(I)
      FVL(I)=FCNVL(IVL,X)
      CALC(IVL,1,I)=FVL(I)
   41 CONTINUE
```

Figure D13: Evolution Code; pages 1-2.

```
C**  CALCULATE D(FVL)/DS AT S=0.
      T=S0
      DO 42 I=1,NX
      X=XGRID(I)
      CALL FINTGV(X,IPM,FVL,FVL1(I))
   42 CONTINUE
C**  CALCULATE FVL AT S=0.-DELTA
      T=S0*DEXP(-DELTA)
      DLT=DELTA
      CALL STARTV(FVLSTO,DLT,FVL,FVL1,IPM)
C**  CALCULATE PREDICTOR AT S=0.+DELTA
      DO 43 I=1,NX
      PRD(I)=FVLSTO(I)+2.*DELTA*FVL1(I)
   43 CONTINUE
C**  CALCULATE D(PRD)/DS AT S=0.+DELTA
      T=S0*DEXP(DELTA)
      DO 44 I=1,NX
      X=XGRID(I)
      CALL FINTGV(X,IPM,PRD,PRD1(I))
      COR(I)=FVL(I)+0.5*DELTA*(FVL1(I)+PRD1(I))
      FUDGE(I)=PRD(I)-COR(I)
   44 CONTINUE
      DO 45 I=1,NX
      X=XGRID(I)
      CALL FINTGV(X,IPM,COR,COR1(I))
   45 CONTINUE
      DO 46 I=1,NX
      FVLSTO(I)=FVL(I)
      FVL(I)=COR(I)
      FVL1(I)=COR1(I)
      CALC(IVL,2,I)=FVL(I)
   46 CONTINUE
C**  INITIALIZATION COMPLETE...ITERATE UNTIL S=SMAX
      DO 47 NTIMES=2,NMAX-1
      NT1=NTIMES+1
      DO 51 I=1,NX
      X=XGRID(I)
      PRD(I)=FVLSTO(I)+2.*DELTA*FVL1(I)
      PRDM(I)=PRD(I)-0.8*FUDGE(I)
   51 CONTINUE
      T=T*DEXP(DELTA)
      DO 52 I=1,NX
      X=XGRID(I)
      FVLSTO(I)=FVL(I)
      CALL FINTGV(X,IPM,PRDM,PRD1(I))
      COR(I)=FVL(I)+0.5*DELTA*(FVL1(I)+PRD1(I))
      FUDGE(I)=PRD(I)-COR(I)
      FVL(I)=COR(I)+0.2*FUDGE(I)
   52 CONTINUE
c  adjust for crossing b threshold
      S=NTIMES*DELTA
      if(s.ge.sb.and.ith.eq.0.and.ivl.gt.2)then
        ith=1
        do 71 i=1,nx
        fvl(i)=fvl(i)+1./20.*calc(7,nt1,i)
        fvlsto(i)=fvlsto(i)+1./20.*calc(7,nt1-1,i)
 71   continue
      endif
      DO 53 I=1,NX
      X=XGRID(I)
      CALL FINTGV(X,IPM,FVL,FVL1(I))
      CALC(IVL,NT1,I)=FVL(I)
   53 CONTINUE
   47 CONTINUE
  100 CONTINUE
      DO 49 J=1,60
   49 XPAR(J)=PARA(J)
      RETURN
      END
      subroutine pdfnrm
      implicit real*8 (a-h,o-z)
      common/gaus32/xi(32),wi(32),nterms,xx(33)
      common/param/xp(60)
c  quantum number sum rule for uv, dv normalization
```

```
      auv=0.
      adv=0.
      do 100 l=1,nterms
      z=.5*(1.+xi(l))
      x=z**3
c  integral of uv from x= 0-1 without overall coefficient
      auv=auv+.5*wi(l)*fuv(x)/x/xp(2)*3.*z**2
c  integral of dv from x= 0-1 without overall coefficient
      adv=adv+.5*wi(l)*fdv(x)/x/xp(8)*3.*z**2
 100  continue
c  renormalize uv, dv to satisfy sum rules
      xp(2)=2./auv
      xp(8)=1./adv
c  momentum sum rule for gluon normalization
      as=0.
      ag=0.
      do 200 l=1,nterms
      z=.5*(1.+xi(l))
      x=z**3
c  integral of x(singlet)
      as=as+.5*wi(l)*fcnfs(x)*3.*z**2
c  integral of xgluon without overall coefficient
      ag=ag+.5*wi(l)*fcnfg(x)/xp(25)*3.*z**2
 200  continue
c  renormalize gluon distribution
      xp(25)=(1.-as)/ag
      return
      end
      FUNCTION FCNVL(IVL,X)
      IMPLICIT REAL*8 (A-H,O-Z)
C  CALCULATES THE INPUT QUARK DISTRIBUTIONS AT Q0**2
C  THE SINGLET AND GLUON DISTRIBUTIONS ARE HANDLED IN
C  SEPARATE ROUTINES (FCNFS AND FCNFG)
      COMMON/PARAM/XP(60)
      COMMON/GRID/NX,XGRID(25)
      COMMON/GRPTHY/FLAVOR
C  IVL = 1  XUV
C  IVL = 2  XDV
c  IVL = 3  XU+
C  IVL = 4  XD+
C  IVL = 5  XS+
C  IVL = 6  XC+
C  Revised 6/3/02 for five flavors
c  Parametrization changed to CTEQ6 style
c  Revised 2/27/03 to implement improved treatment of b threshold
      if(ivl.eq.1)then
c  xuv
        fcnvl=fuv(x)
        return
      else if(ivl.eq.2)then
c  xdv
        fcnvl=fdv(x)
        return
      else if(ivl.eq.3)then
c  x(uv+2.*ubar-.25*singlet)
        ubpdb=fubpdb(x)
        dboub=fdboub(x)
        fcnvl=fuv(x)+2.*ubpdb/(1.+dboub)-.25*fcnfs(x)
        return
      else if(ivl.eq.4)then
c  x(dv+2.*dbar-.25*singlet)
        ubpdb=fubpdb(x)
        dboub=fdboub(x)
        fcnvl=fdv(x)+2.*ubpdb/(1.+1./dboub)-.25*fcnfs(x)
        return
      else if(ivl.eq.5)then
c  x(s + sbar) = kappa*(ubar + dbar)
c  x(s+sbar-.25*singlet)
        fcnvl=xp(31)*fubpdb(x)-.25*fcnfs(x)
        return
      else if(ivl.eq.6)then
c  xcplus = -1/4*singlet
        fcnvl=-.25*fcnfs(x)
        return
```

Figure D14: Evolution Code; pages 3-4.

```
      endif
      end
      function fuv(x)
      implicit real*8 (a-h,o-z)
      common/param/xp(60)
      fuv=xp(2)*x**xp(3)*(1.-x)**xp(4)*exp(xp(5)*x)*
     $(1.+exp(xp(6))*x)**xp(7)
      return
      end
      function fdv(x)
      implicit real*8 (a-h,o-z)
      common/param/xp(60)
      fdv=xp(8)*x**xp(3)*(1.-x)**xp(4)*exp(xp(5)*x)*
     $(1.+exp(xp(6))*x)**xp(7)
C MOD for NUTEV Analysis
C Fix dv to have same shape as uv
      return
      end
      function fubpdb(x)
      implicit real*8 (a-h,o-z)
      common/param/xp(60)
      fubpdb=xp(14)*x**xp(15)*(1.-x)**xp(16)*exp(xp(17)*x)*
     $(1.+exp(xp(18))*x)**xp(19)
      return
      end
      function fdboub(x)
      implicit real*8 (a-h,o-z)
      common/param/xp(60)
      fdboub=xp(20)*x**(xp(21)-1.)*(1.-x)**xp(22)
     $+(1.+xp(23)*x)*(1.-x)**xp(24)
      return
      end
      function fcnfs(x)
      implicit real*8 (a-h,o-z)
      common/param/xp(60)
c  singlet at Q0
      fcnfs=fuv(x)+fdv(x)+fubpdb(x)*(2.+xp(31))
      return
      end
      function fcnfg(x)
      implicit real*8 (a-h,o-z)
      common/param/xp(60)
c  gluon at Q0
      fcnfg=xp(25)*x**xp(26)*(1.-x)**xp(27)*exp(xp(28)*x)*
     $(1.+exp(xp(29))*x)**xp(30)
      return
      end
      SUBROUTINE FINTGV(X,IPM,FVL,FINTG)
      IMPLICIT REAL*8 (A-H,O-Z)
C  INTEGRATES VALENCE TERM(S) - SEE NOTES
      DIMENSION FVL(25)
      COMMON/LAGUER/XL(8),WL(8),NTERML
      COMMON/GAUS16/XI(16),WI(16),NTERMS,XX(17)
      COMMON/GRPTHY/ FLAVOR
      COMMON/INPUT/INS,NVL,NMAX,DELTA,S0,IORD
      COMMON/ALQ2/T
      COMMON/CONSTANTS/PI,PI2
      common/param/xc(60)
      xmu2=xc(1)**2*dexp(t)
      al=alpha_s(iord+1,xmu2,xc(1),neff)/(2.*pi)
      flavor=neff
      FX=GETFV(X,FVL)
      FINTG=T*AL*FX*(2.+8./3.*DLOG(1.-X))
      IF(IORD) 3,3,4
    4 C1=16./9.
      C2=2.
      C3=2./3.*FLAVOR
      C4=1.202056903D0
      C5=-5./8.*C4
      A1=C2*2.*(67./9.-PI2/3.)-C3*20./9.
      C=C1*(3/8.-PI2/2.+C4-8.*C5)+C2*(17./12.+11.*PI2/9.
     2-C4+8.*C5)-C3*(1./6.+2.*PI2/9.)
      FINTG=FINTG+T*AL**2*FX*(C+A1*DLOG(1.-X))
    3 CONTINUE
```

```
      DO 1 I=1,NTERMS
      Z=0.5*(1.-X)*XI(I)+0.5*(1.+X)
      FZ=GETFV(X/Z,FVL)
      FINTG=FINTG+0.5*(1.-X)*T*AL*4./3.*((1.+Z*Z)*FZ-2.*FX)/(1.-Z)*WI(I)
      IF (IORD) 1,1,2
    2 ALZ=DLOG(Z)
      AL1=DLOG(1.-Z)
      Z2=1.+Z*Z
      ZM=1.-Z
      ZP=1.+Z
      AZ=C1*(-2.*Z2*ALZ*AL1-3.*ALZ)+C2*Z2*(ALZ*(ALZ+11./3.)+67./9.-PI2/
     23.)-C3*2./3.*Z2*(ALZ+5./3.)
      PA=4.*ZM+2.*ZP*ALZ+2.*Z2/ZP*S2(Z)
      BZ=C1*(-ALZ*(ALZ*ZP/2.+2.*Z)-5.*ZM+IPM*PA)+C2*(2.*ZP*ALZ
     2+40./3.*ZM-IPM*PA)-C3*4./3.*ZM
      FINTG=FINTG+0.5*(1.-X)*T*AL**2*((AZ*FZ-A1*FX)/ZM+BZ*FZ)*WI(I)
    1 CONTINUE
      RETURN
      END
      FUNCTION S2(X)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION FN(51)
      DATA FN/1.644934076,1.588625448,1.545799712,1.507899041,
     X1.473125860,1.440633797,1.409928300,1.380685041,1.352675161,
     X1.325728728,1.299714723,1.274529160,1.250087584,1.226320101,
     X1.203167961,1.180581124,1.158516487,1.136936560,1.115808451,
     X1.095103088,1.074794600,1.054859830,1.035277934,1.016030062,
     X0.997099088,0.978469393,0.960126675,0.942057798,0.924250654,
     X0.906694053,0.889377624,0.872291733,0.855427404,0.838776261,
     X0.822330471,0.806082689,0.790026024,0.774153992,0.758460483,
     X0.742939737,0.727586308,0.712395042,0.697361058,0.682479725,
     X0.667746644,0.653157631,0.638708705,0.624396071,0.610216108,
     X0.596165361,0.582240526/
      COMMON/CONSTANTS/PI,PI2
C  THESE ARE THE VALUES OF F(X)=LI2(1-X) FOR X BETWEEN O AND .50 IN STEPS
C  OF O.O1 TAKEN FROM ABRAMOWITZ AND STEGUN, PG. 1005, TABLE 27.7.
C  S2(X)=INTEGRAL OF LN((1-Z)/Z)/Z FROM X/(1+X) TO 1/(1+X)
C  REWRITE S2(X) IN TERMS OF F(X/(1+X))
C  USE A LINEAR INTERPOLATION TO OBTAIN F(Z), Z=X/(1.+X)
C  IN THE REGION OF X BELOW 0.8
C  NEAR X=1 SMALL ERRORS IN THE INTERPOLATION ARE AMPLIFIED
C  SINCE THE ANSWER IS A SMALL DIFFERENCE BETWEEN MUCH LARGER
C  NUMBERS. USE A TAYLOR SERIES FOR F(X) EXPANDED ABOUT THE
C  NEAREST X VALUE.
      Z=X/(1.+X)
      N=100.*Z+1
      Z1=(N-1)/100.
      Z2=N/100.
      IF(X.LT.0.8) THEN
         F=FN(N)*(Z-Z2)/(Z1-Z2)+FN(N+1)*(Z-Z1)/(Z2-Z1)
      ELSE
         DELT1=DABS(Z-Z1)
         DELT2=DABS(Z-Z2)
         ZT=Z1
         IF(DELT2.LT.DELT1) THEN
            ZT=Z2
            N=N+1
         ENDIF
         OMZT=1.-ZT
         F0=FN(N)
         F1=DLOG(ZT)/OMZT
         F2=(1./ZT+F1)/OMZT
         F3=(1./ZT**2+2.*F2)/OMZT
         F=F0+(Z-ZT)*F1+.5*(Z-ZT)**2*F2+(Z-ZT)**3/6.*F3
      ENDIF
      S2=PI**2/6.+.5*DLOG(X)**2-DLOG(1.+X)**2-2.*F
      RETURN
      END
      SUBROUTINE FINTGG(X,FG,FS,FINTG)
      IMPLICIT REAL*8 (A-H,O-Z)
C  INTEGRATES GLUON TERM - SEE NOTES
      DIMENSION FG(25),FS(25)
      COMMON/GAUS16/XI(16),WI(16),NTERMS,XX(17)
      COMMON/GRPTHY/ FLAVOR
```

Figure D15: Evolution Code; pages 5-6.

```
      COMMON/INPUT/INS,NVL,NMAX,DELTA,S0,IORD
      COMMON/ALQ2/T
      COMMON/CONSTANTS/PI,PI2
      common/param/xc(60)
      xmu2=xc(1)**2*dexp(t)
      al=alpha_s(iord+1,xmu2,xc(1),neff)/(2.*pi)
      flavor=neff
      FX=GETFV(X,FG)
      FINTG=T*AL*FX*((33.-2.*FLAVOR)/6.+6.*DLOG(1.-X))
      CF=4./3.
      IF(IORD) 2,2,3
    3 CONTINUE
      TR=FLAVOR/2.
      CA=3.
      Z3=1.202056903D0
      AL1=DLOG(1.-X)
      PGG1=-CF*TR-4./3.*CA*TR*(1.+5./3.*AL1)+CA*CA/3.*
     2(8.+9.*Z3+(67/3.-PI2)*AL1)
      FINTG=FINTG+T*AL*AL*FX*PGG1
    2 CONTINUE
      DO 1 I=1,NTERMS
      Z=0.5*(1.-X)*XI(I)+0.5*(1.+X)
      FGZ=GETFV(X/Z,FG)
      FSZ=GETFV(X/Z,FS)
      FINTG=FINTG+.5*(1.-X)*T*AL*WI(I)*(CF*FSZ*(Z*Z-2.*Z+2.)/Z+
     26.*(FGZ*Z-FX)/(1.-Z)+6.*FGZ*(1.-Z+Z*Z-Z**3)/Z)
      IF(IORD) 4,4,5
    5 CONTINUE
      AL1=DLOG(1.-Z)
      ALZ=DLOG(Z)
      PGG2=CF*TR*(4./3.*(1./Z-12.+6.*Z+5.*Z*Z)-2.*(3.+5.*Z)*ALZ
     2-2.*(1.+Z)*ALZ*ALZ)+4./3.*CA*TR*(-1./6.*(23./Z-29.+19.*Z-23.*Z*Z)
     3-(1.+Z)*ALZ)+CA*CA*(-1./18.*(25.+109.*Z)-1./3.*(25.-11.*Z+44.*Z*Z)
     4*ALZ+4.*(1.+Z)*ALZ*ALZ-PI2/3.*(1./Z-2.+Z-Z*Z)+(1./Z-2.+Z-Z*Z+
     51./(1.-Z))*ALZ*(ALZ-4.*AL1)-2.*(1./Z+2.+Z+Z*Z-1./(1.+Z))*S2(Z))
      PGG3=(-20./9.*CA*TR+CA*CA/3.*(67./3.-PI2))/(1.-Z)
      PGQ=4./3.*CF*TR*(2./3.*(5./Z-5.+4.*Z)+(2./Z-2.+Z)*AL1)
     2+.5*CF*CF*(5.+7.*Z-(4.+7.*Z)*ALZ+2.*(6./Z-6.+5*Z)*AL1
     3+(2.-Z)*ALZ**2+2.*(2./Z-2+Z)*AL1**2)+CA*CF*(-1./9.*(9./Z+19.
     4+37.*Z+44.*Z*Z)+1./3.*(36.+15.*Z+8.*Z*Z)*ALZ-.5*(2./Z+6.
     5+3.*Z)*ALZ**2-1./3.*(22./Z-22.+17.*Z)*AL1+(2./Z-2.+Z)*(
     62.*ALZ*AL1+PI2/6.-AL1**2)+(2./Z+2.+Z)*S2(Z))
      PGQ=-PGQ
      FINTG=FINTG+.5*(1.-X)*T*AL*AL*WI(I)*(FSZ*PGQ+FGZ*PGG2+PGG3*(
     2FGZ-FX))
    4 CONTINUE
    1 CONTINUE
      RETURN
      END
      SUBROUTINE FINTGS(X,FS,FG,FINTG)
      IMPLICIT REAL*8 (A-H,O-Z)
C  INTEGRATES SINGLET TERM - SEE NOTES
      DIMENSION FS(25),FG(25)
      COMMON/GAUS16/XI(16),WI(16),NTERMS,XX(17)
      COMMON/GRPTHY/ FLAVOR
      COMMON/INPUT/INS,NVL,NMAX,DELTA,S0,IORD
      COMMON/ALQ2/T
      COMMON/CONSTANTS/PI,PI2
      common/param/xc(60)
      xmu2=xc(1)**2*dexp(t)
      al=alpha_s(iord+1,xmu2,xc(1),neff)/(2.*pi)
      flavor=neff
      FX=GETFV(X,FS)
      FINTG=FX*T*AL*(2.+8./3.*DLOG(1.-X))
      CF=4./3.
      IF(IORD) 2,2,3
    3 CONTINUE
      TR=FLAVOR/2.
      CA=3.
      Z3=1.202056903D0
      AL1=DLOG(1.-X)
      PFF1=-2./9.*CF*TR*(3./4.+PI2+10.*AL1)+CF*CF*(3.-
     24.*PI2+48.*Z3)/8.+CA*CF*(17./3.+44./9.*PI2-24.*Z3+
     3(536./9.-8.*PI2/3.)*AL1)/8.
```

```
      FINTG=FINTG+T*AL*AL*FX*PFF1
    2 CONTINUE
      DO 1 I=1,NTERMS
      Z=0.5*(1.-X)*XI(I)+0.5*(1.+X)
      FSZ=GETFV(X/Z,FS)
      FGZ=GETFV(X/Z,FG)
      FINTG=FINTG+.5*(1.-X)*T*AL*WI(I)*(CF*((1.+Z*Z)*FSZ-2.*FX)/(1.-Z)
     2+FLAVOR*((1.-Z)**2+Z*Z)*FGZ)
      IF(IORD) 4,4,5
    5 CONTINUE
      AL1=DLOG(1.-Z)
      ALZ=DLOG(Z)
      PFF2=2./9.*CF*TR*(20./Z-19.+65.*Z-56.*Z*Z+6.*(2.+8.*Z+4.*
     2Z*Z-1./(1.-Z))*ALZ-9.*(1.+Z)*ALZ**2)+CF*CF*(-1.+Z+(2.-3./
     3(1.-Z))*ALZ-.5*(1.+Z)*ALZ**2-2.*(1.+Z*Z)/(1.-Z)*ALZ*AL1+
     42.*(1.+Z*Z)/(1.+Z)*S2(Z))+CA*CF/8.*(4./9.*(17.-151*Z)
     5+4.*(1.+Z*Z)/(1.-Z)*ALZ*(11./3.+ALZ)+4.*(1.+Z)*PI2/3.-8.*
     6(1.+Z*Z)/(1.+Z)*S2(Z))
      PFF3=(-20./9.*CF*TR+(67./9.-PI2/3.)*CA*CF)/(1.-Z)
      PQG=-CF*TR*(-14.+29.*Z-20.*Z*Z-(3.-4.*Z+8.*Z*Z)*ALZ-
     2(1.-2.*Z+4.*Z*Z)*ALZ**2-8.*Z*(1.-Z)*AL1+2.*(1.-2.*Z+2.*Z*Z)*
     3(2.*ALZ*AL1+PI2/3.-AL1**2))-CA*TR*(-2./9.*(20./Z-18.+225.*Z-
     4218.*Z*Z)-2./3.*(3.+24.*Z+44.*Z*Z)*ALZ+(3.+6.*Z+2.*Z*Z)*
     5ALZ**2+8.*Z*(1.-Z)*AL1+(1.-2.*Z+2.*Z*Z)*(2*AL1**2-PI2/3.)
     6-2.*(1.+2.*Z+2.*Z*Z)*S2(Z))
      FINTG=FINTG+.5*(1.-X)*T*AL*AL*WI(I)*(FSZ*PFF2+PFF3*(FSZ-FX)
     2+PQG*FGZ)
    4 CONTINUE
    1 CONTINUE
      RETURN
      END
      FUNCTION GETFV(X,FVL)
      IMPLICIT REAL*8 (A-H,O-Z)
C  LOGARITHMIC INTERPOLATOR - WATCH OUT FOR NEGATIVE
C  FUNCTIONS AND/OR X VALUES OUTSIDE THE RANGE 0 TO 1.
C  NOTE: DIMENSION OF FVL IS OVERWRITTEN BY VALUE USED
C  IN MAIN ROUTINE.
      DIMENSION FVL(25),xx(3),fx(3)
      COMMON/GRID/NX,XGRID(25)
      IREP=-1
      DO 1 I=1,NX
      IF(X.LT.XGRID(I)) GO TO 2
    1 CONTINUE
    2 I=I-1
      IF(I.EQ.0) THEN
         I=I+1
         IREP=IREP+1
      ELSE IF(I.GT.23) THEN
         I=23
      ENDIF
      xx(1)=xgrid(i)
      xx(2)=xgrid(i+1)
      xx(3)=xgrid(i+2)
      fx(1)=fvl(i)
      fx(2)=fvl(i+1)
      fx(3)=fvl(i+2)
      call polint(xx,fx,3,x,ans,dy)
      getfv=ans
      RETURN
      END
      SUBROUTINE STARTV(FVLSTO,DELTA,FVL,FVL1,IPM)
      IMPLICIT REAL*8 (A-H,O-Z)
C  CALCULATES STARTING VALUES OF VALENCE DISTRIBUTIONS
C  AS NEEDED FOR THE PREDICTOR-CORRECTOR ALGORITHM
      DIMENSION FVLSTO(25),FVL(25),FVL1(25),TVLSTO(25)
      COMMON/GRID/NX,XGRID(25)
      DO 1 I=1,NX
    1 TVLSTO(I)=FVL(I)-DELTA*FVL1(I)
      DO 2 I=1,NX
      X=XGRID(I)
      CALL FINTGV(X,IPM,TVLSTO,FVLST1)
      FVLSTO(I)=FVL(I)-0.5*DELTA*(FVL1(I)+FVLST1)
    2 CONTINUE
      RETURN
```

Figure D16: Evolution Code; pages 7-8.

```
      END
      SUBROUTINE STARTS(FS,FS1,FSSTO,FG,FG1,FGSTO,DELTA)
      IMPLICIT REAL*8 (A-H,O-Z)
C  CALULATES SINGLET AND GLUON STARTING VALUES AS NEEDED
C  FOR THE PREDICTOR-CORRECTOR ALGORITHM
      DIMENSION FS(25),FS1(25),FSSTO(25),TSSTO(25),
     $          FG(25),FG1(25),FGSTO(25),TGSTO(25)
      COMMON/GRID/NX,XGRID(25)
      DO 1 I=1,NX
      TSSTO(I)=FS(I)-DELTA*FS1(I)
      TGSTO(I)=FG(I)-DELTA*FG1(I)
    1 CONTINUE
      DO 2 I=1,NX
      X=XGRID(I)
      CALL FINTGS(X,TSSTO,TGSTO,FSSTO1)
      CALL FINTGG(X,TGSTO,TSSTO,FGSTO1)
      FSSTO(I)=FS(I)-0.5*DELTA*(FS1(I)+FSSTO1)
      FGSTO(I)=FG(I)-0.5*DELTA*(FG1(I)+FGSTO1)
    2 CONTINUE
      RETURN
      END
      SUBROUTINE WATE32
      IMPLICIT REAL*8 (A-H,O-Z)
C  32 POINT GAUSSIAN QUADRATURE ROUTINE
      DIMENSION X(16),W(16)
      COMMON/GAUS32/XI(32),WI(32),NTERMS,XX(33)
      NTERMS=32
      X(1)=0.048307665687738316235
      X(2)=0.144471961582796493485
      X(3)=0.239287362252137074545
      X(4)=0.331868602282127649780
      X(5)=0.421351276130635345364
      X(6)=0.506899908932229390024
      X(7)=0.587715757240762329041
      X(8)=0.663044266930215200975
      X(9)=0.732182118740289680387
      X(10)=0.794483795967942406963
      X(11)=0.849367613732569970134
      X(12)=0.896321155766052123965
      X(13)=0.934906075937739689171
      X(14)=0.964762255587506430774
      X(15)=0.985611511545268335400
      X(16)=0.997263861849481563545
      W(1)=0.096540088514727800567
      W(2)=0.095638720079274859419
      W(3)=0.093844399080804565639
      W(4)=0.091173878695763884713
      W(5)=0.087652093004403811143
      W(6)=0.083311924226946755222
      W(7)=0.078193895787070306472
      W(8)=0.072345794108848506225
      W(9)=0.065822222776361846838
      W(10)=0.058684093478535547145
      W(11)=0.050998059262376176196
      W(12)=0.042835898022226680657
      W(13)=0.034273862913021433103
      W(14)=0.025392065309262059456
      W(15)=0.016274394730905670605
      W(16)=0.007018610009470096600
      DO 1 I=1,16
      XI(I)=-X(17-I)
      WI(I)=W(17-I)
      XI(I+16)=X(I)
      WI(I+16)=W(I)
    1 CONTINUE
      DO 2 I=1,32
    2 XX(I)=0.5*(XI(I)+1.)
      XX(33)=1.0
      RETURN
      END
      SUBROUTINE WATE16
      IMPLICIT REAL*8 (A-H,O-Z)
C  16 POINT GAUSSIAN QUADRATURE ROUTINE
      DIMENSION X(8),W(8)
```

```
      COMMON/GAUS16/XI(16),WI(16),NTERMS,XX(17)
      NTERMS=16
      X(1)=0.095012509837637440185D0
      X(2)=0.281603550779258913230D0
      X(3)=0.458016777657227386342D0
      X(4)=0.617876244402643748447D0
      X(5)=0.755404408355003033895D0
      X(6)=0.865631202387831743880D0
      X(7)=0.944575023073232576078D0
      X(8)=0.989400934991649932596D0
      W(1)=0.189450610455068496285D0
      W(2)=0.182603415044923588867D0
      W(3)=0.169156519395002538189D0
      W(4)=0.149595988816576732081D0
      W(5)=0.124628971255533872052D0
      W(6)=0.095158511682492784810D0
      W(7)=0.062253523938647892863D0
      W(8)=0.027152459411754094852D0
      DO 1 I=1,8
      XI(I)=-X(9-I)
      WI(I)=W(9-I)
      XI(I+8)=X(I)
      WI(I+8)=W(I)
    1 CONTINUE
      DO 2 I=1,16
    2 XX(I)=0.5*(XI(I)+1.)
      XX(17)=1.0
      RETURN
      END
      SUBROUTINE WATE8
      IMPLICIT REAL*8 (A-H,O-Z)
C  8 POINT GAUSSIAN QUADRATURE ROUTINE
      COMMON/GAUSS8/XI(8),WI(8),NTERMS,XX(9)
      NTERMS=8
      XI(4)=-0.183434642495650D0
      XI(3)=-0.525532409916329D0
      XI(2)=-0.796666477413627D0
      XI(1)=-0.960289856497536D0
      WI(4)=0.362683783378362D0
      WI(3)=0.313706645877887D0
      WI(2)=0.222381034453374D0
      WI(1)=0.101228536290376D0
      DO 1 I=5,8
      XI(I)=-XI(9-I)
    1 WI(I)=WI(9-I)
      DO 2 I=1,8
    2 XX(I)=0.5*(XI(I)+1.D0)
      XX(9)=1.
      RETURN
      END
      SUBROUTINE WATE8L
      IMPLICIT REAL*8 (A-H,O-Z)
C  8 POINT LAGUERRE INTGRATOR (FOR EXPONENTIALS OR OTHER
C  STEEPLY FALLING FUNCTIONS
      COMMON/LAGUER/XL(8),WL(8),NTERML
      XL(1)=0.1702796
      XL(2)=0.9037018
      XL(3)=2.2510866
      XL(4)=4.2667002
      XL(5)=7.0459054
      XL(6)=10.758516
      XL(7)=15.740679
      XL(8)=22.863132
      WL(1)=0.4377234
      WL(2)=1.0338693
      WL(3)=1.6697098
      WL(4)=2.3769247
      WL(5)=3.2085409
      WL(6)=4.2685755
      WL(7)=5.8180834
      WL(8)=8.9062262
      NTERML=8
      RETURN
      END
```

Figure D17: Evolution Code; page 9-10.

APPENDIX E

QCD SOURCE CODE: PART B

This chapter presents the key programs which use the *ACOT* scheme [54].

```
C2345678901234567890123456789012345678901234567890123456789012345678901234567890
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
       SUBROUTINE Fcc123(icharge,Mode, XBJ, Q,XMU, F123)
C-----------------------------------------------------
C      Ischeme= 0  !*** Massless MS-Bar
C      Ischeme= 1  !*** Full ACOT Scheme
C      Ischeme= 2  !*** FFS
C      Ischeme= 3  !*** Simplified ACOT Scheme
C      Ischeme= 4  !*** Test Full ACOT Scheme (no NLO Q)
C      Ischeme= 5  !*** LO
C      Ischeme= 6  !*** Massless LO
C------------------------------------------------------------------------------
       Implicit Double Precision (A-H, O-Z)
       CHARACTER HEADER*78
       Dimension xout123(3), XXSC(3), XMARRAY(6), CHARGE(6)
       Dimension F123(3)
       PARAMETER (PI=3.14159265359)
       Common /Ischeme/ Isch, Iset, Iflg, Ihad
       Common  / ActInt /  AERR, RERR, iActL, iActU
C------------------------------------------------------------------------------
         INTEGER UPTYPE(3), DNTYPE(3)
         DIMENSION CKM(3,3)

       DATA  (UPTYPE(I), I=1,3,1) /1,4,6/
       DATA  (DNTYPE(I), I=1,3,1) /2,3,5/
       DATA ((CKM(I,J), J=1,3,1), I=1,3,1)    !*** ABS OF CKM MATRIX
     1   /0.97485, 0.2225, 0.00365,
     2    0.2225, 0.974, 0.041,
     3    0.009, 0.0405, 0.99915/ !PDG 2002
C------------------------------------------------------------------------------
C--- FOR CHARGED CURRENT
       DATA GLQ,GRQ,HMASS
     >   /  1.0,0.0,0.0/
       DATA IPRINT /0/  !*** for debugging use 1
C------------------------------------------------------------------------------
C                          U      D      S      C      B      T
       DATA XMARRAY  /    0.1,   0.1,   0.2,   1.4,   4.3, 175.0/
       DATA CHARGE   /   +2.0,  -1.0,  -1.0,  +2.0,  -1.0,  +2.0/
C-----------------------------------------------------------------------
C INITIALIZATION
C-----------------------------------------------------------------------
          DO I=1,3,1
             F123( I)  = 0.0d0
          ENDDO
C-----------------------------------------------------------------------
C CHOOSE W+ or W- SCATTERING
C-----------------------------------------------------------------------
      IUPDOWN  =-1    !*** W- SCATTERING
      IUPDOWN  =+1    !*** W+ SCATTERING
      IUPDOWN = ICHARGE
      UPDOWN=float(ICHARGE)  !*** CHANGE TO FLOAT
C-----------------------------------------------------
C   ***  LOOP OVER PARTON FLAVORS:
C-----------------------------------------------------
      DO IQUARKIN  =1,3,1
      DO IQUARKOUT =1,3,1
C-----------------------------------------------------
      IF(UPDOWN.GE.0)  THEN  !*** THIS IS FOR DOWN -> UP
           IPARTIN = +DNTYPE(IQUARKIN )
           IPARTOUT= +UPTYPE(IQUARKOUT)
           IUP= IQUARKOUT   !***  USED TO PICK UP CORRECT CKM TERM
           IDN= IQUARKIN
      ELSE                 !*** THIS IS FOR UP -> DOWN
           IPARTIN = +UPTYPE(IQUARKIN )
           IPARTOUT= +DNTYPE(IQUARKOUT)
           IUP= IQUARKIN    !***  USED TO PICK UP CORRECT CKM TERM
           IDN= IQUARKOUT
      ENDIF
      COUPLING = CKM(IUP,IDN)**2
C-----------------------------------------------------------------------
C SETUP PARTON MASSES:
C-----------------------------------------------------------------------
      IF(IPRINT.NE.0) WRITE(6,*) 'PARTON',IPARTIN,'=>',IPARTOUT
C-----------------------------------------------------------------------
```

```
C SETUP PARTON MASSES:
C-----------------------------------------------------------------------
       F1M=XMARRAY(ABS(IPARTIN))
       F2M=XMARRAY(ABS(IPARTOUT))
C-----------------------------------------------------
C NOTE: ASSUME TOP IS ZERO (NOT IMPLEMENTED IN ALL PDF SETS)
C-----------------------------------------------------------------------
      if((abs(IPARTIN).NE.6).and.(abs(IPARTOUT).NE.6)) then
            call TOTF(XBJ,Q,F1M,F2M,GLQ,GRQ,IPARTIN,IPARTOUT,
     >           XMU,ISET,Ihad,HMASS,Isch,Iflg,Icharge,Xout123)
         else
            DO I=1,3,1
               Xout123( I)  =0.0
            ENDDO
      endif
C-----------------------------------------------------
C   ***  ADD PARTON CONTRIBUTION TO TOTAL STRUCTURE FUNCTIONS
C-----------------------------------------------------
           DO I=1,3,1
             F123( I)  = F123( I)  + COUPLING * Xout123( I)
           ENDDO
      ENDDO  !***  IQUARKOUT
      ENDDO  !***  IQUARKIN
C-----------------------------------------------------
C   ***  END LOOP OVER PARTON FLAVORS:
C-----------------------------------------------------
      Return
      END
```

Figure E1: Constructing Structure functions using *ACOT* scheme .

```
C2345678901234567890123456789012345678901234567890123456789012345678901234567890
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C-----------------------------------------------------------------------------
      SUBROUTINE TOTF(XBJ,Q,F1M,F2M,GLQ,GRQ,IPARTIN,IPARTOUT,
     >   XMU,ISET,Ihad,HMASS,Ischeme,Iflg,Ichannel,Xout123)
C      12/08/05 FIO. Add new scheme: FULL ACOT BUT WITH ZERO MASS NLO-Q TERMS::C
      02/06/01 FIO. modify to includ F123
C      08/16/02 FIO. modify to include more schemes
C----------------------------------------------------
c          If(Ischeme.eq.0)   !*** MASSLESS MS-BAR
c          If(Ischeme.eq.1)   !*** FULL ACOT
c          If(Ischeme.eq.2)   !*** FFS
c          If(Ischeme.eq.3)   !*** SHORT-CUT TOT (S-ACOT)
c          If(Ischeme.eq.4)   !*** Test ACOT (NO NLO Q)
c          If(Ischeme.eq.5)   !*** LO
c          If(Ischeme.eq.6)   !*** Massless LO
c          If(Ischeme.eq.7)   !*** FULL ACOT BUT WITH ZERO MASS NLO-Q TERMS::
C----------------------------------------------------
       Implicit Double Precision (A-H, O-Z)
       CHARACTER HEADER*78
       Dimension
     >            XMARRAY(6), Xout123(3)
     >           ,XLO123( 3),XLO123a( 3),XLO123b( 3)
     >           ,XLO1230(3),XLO123a0(3),XLO123b0(3)
     >           ,XSUB123( 3),XSUB123a( 3),XSUB123b( 3)
     >           ,XSUB1230(3),XSUB123a0(3),XSUB123b0(3)
     >           ,XSUB123Q(3),XSUB123Qa(3),XSUB123Qb(3)
     >           ,XNLO123(3),XTOT123(3),XDIFF123(3),XDIFF123q(3)
     >           ,XNLO123Qz(3),XNLO123Qza(3),XNLO123Qzb(3)
     >           ,XG123(3),XGR123(3)
     >           ,XHq123( 3),XHq123a( 3),XHq123b( 3)
     >           ,XHq1230(3),XHq123a0(3),XHq123b0(3)
     >           ,XHqSUBQ123(3),XHqSUBQ123a(3),XHqSUBQ123b(3)
     >           ,XGR123Q(3),XTOT1230(3),XTOT123S(3),XRATQG(3)
     >           ,XTOT123X(3)
     >           ,XTOT123y(3),ratio(3),ratio2(3)
       DIMENSION F123(3)
       DATA F123 / +1.d0, +1.d0, +1.d0 /  !*** SIGN OF ANTI-QUARK TERMS
C  ********* SIGN OF ANTI-QUARK TERMS IS AUTOMATIC WITH GL<=>GR
       PARAMETER (PI=3.14159265359)
       DATA IDEBUG /0/
      aqk=0.0                   !*********  Anti-quark Channel
      IF(ICHANNEL.NE.0) aqk=1.0
C ********* ICHANNEL is mainly for debugging to turn off Anti-quark Channel
C ********* There is no physics process where you want to run this way
C============================================================
C   ***  FIRST LOOK AT THE MASSLESS PARTS
C============================================================
C %%% THIS IS THE MASSLESS MS-BAR FORMULA
       imass=0
C----------------------------------------------------
C   ***  COMPUTE STRUCTURE FUNCTIONS: PICK UP T-CHANNEL GRAPHS
C----------------------------------------------------
       CALL  HADLO123(imass,Iflg,Ihad,IPARTIN,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS,XLO123a0)
       CALL HADSUB123(imass,Iflg,Ihad,IPARTIN,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS,XSUB123a0)
       CALL HADNLOQ1230(Iflg,Ihad,XBJ,Q,F1M,F2M,GLQ,GRQ,XMU,
     >       HMASS,+IPARTIN ,ISET,XNLO123Qza)
C----------------------------------------------------
C   ***  GO BACK AND PICK UP U-CHANNEL GRAPHS
C----------------------------------------------------
C  switch +IPARTIN -> -IPARTOUT and F2M <=> F1M and GLQ <=> GRQ
       CALL  HADLO123(imass,Iflg,Ihad,-IPARTOUT,XBJ,Q,F2M,F1M,
     >   GRQ,GLQ,XMU,ISET,HMASS,XLO123b0)
       CALL HADSUB123(imass,Iflg,Ihad,-IPARTOUT,XBJ,Q,F2M,F1M,
     >   GRQ,GLQ,XMU,ISET,HMASS,XSUB123b0)
       CALL HADNLOQ1230(Iflg,Ihad,XBJ,Q,F2M,F1M,GRQ,GLQ,XMU,
     >       HMASS,-IPARTOUT,ISET,XNLO123Qzb)
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C23456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C============================================================
C   ***  NEXT LOOK AT THE NON-MASSLESS PARTS
```

```
C============================================================
C %%% THIS IS THE MASSIVE ACOT FORMULA
       imass=1
C----------------------------------------------------
C   ***  COMPUTE STRUCTURE FUNCTIONS: PICK UP T-CHANNEL GRAPHS
C----------------------------------------------------
       CALL HADLO123(imass,Iflg,Ihad,IPARTIN,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS,XLO123a)
       CALL HADSUB123(imass,Iflg,Ihad,IPARTIN,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS,XSUB123a)
       CALL HADNLOQ123K(Iflg,Ihad, XBJ,Q,F1M,F2M,GLQ,GRQ,XMU,HMASS
     >                   ,IPARTIN,ISET,XHq123a)
       CALL HADSUBQ123(Iflg,Ihad,IPARTIN,XBJ,Q,F1M,F2M,GLQ,GRQ,XMU,
     >                  ISET,HMASS,XHqSUBQ123a)
C----------------------------------------------------
C   ***  GO BACK AND PICK UP U-CHANNEL GRAPHS
C----------------------------------------------------
C  switch +IPARTIN -> -IPARTOUT and F2M <=> F1M and GLQ <=> GRQ
       CALL  HADLO123(imass,Iflg,Ihad,-IPARTOUT,XBJ,Q,F2M,F1M,
     >   GRQ,GLQ,XMU,ISET,HMASS,XLO123b)
       CALL HADSUB123(imass,Iflg,Ihad,-IPARTOUT,XBJ,Q,F2M,F1M,
     >   GRQ,GLQ,XMU,ISET,HMASS,XSUB123b)
       CALL HADNLOQ123K(Iflg,Ihad, XBJ,Q,F2M,F1M,GRQ,GLQ,XMU,HMASS,
     >                    -IPARTout,ISET,XHq123b)
       CALL HADSUBQ123(Iflg,Ihad,-IPARTout,XBJ,Q,F2M,F1M,GRQ,GLQ,XMU,
     >              ISET,HMASS,XHqSUBQ123b)
C============================================================
C   ***  NOW LOOK AT THE OTHER PARTS
C============================================================
C----------------------------------------------------
C   ***  COMPUTE STRUCTURE FUNCTIONS: These have both T and and U channel
C----------------------------------------------------
C %%% THIS IS THE MASSIVE ACOT FORMULA
       CALL HADNLO123(Iflg,Ihad,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS,XNLO123)
C----------------------------------------------------
C   ***  COMPUTE MASSLESS GLUON CONTRIBUTION FOR COMPARISON
C----------------------------------------------------
C %%% THIS IS THE MASSLESS MS-BAR FORMULA
       CALL HADNLOG123z(Iflg,Ihad,XBJ,Q,F1M,F2M,GLQ,GRQ,
     >   XMU,ISET,HMASS,XG123)
C----------------------------------------------------
C   ***  ADD TOGETHER
C   ***  INCLUDE SIGN "F123" FOR ANTI-QUARK TERMS
C----------------------------------------------------
        DO I=1,3,1
           XLO1230(  I) = XLO123a0(  I) + XLO123b0(  I) *AQK*F123(I) !*** MASSLE
SS MS-BAR
           XSUB1230( I) = XSUB123a0( I) + XSUB123b0( I) *AQK*F123(I) !*** MASSLE
SS MS-BAR
           XLO123(   I) = XLO123a(   I) + XLO123b(   I) *AQK*F123(I) !*** FULL A
COT
           XSUB123(  I) = XSUB123a(  I) + XSUB123b(  I) *AQK*F123(I) !*** FULL A
COT
           XNLO123Qz(I) = XNLO123Qza(I) + XNLO123Qzb(I) *AQK*F123(I) !*** MASSLE
SS MS-BAR
           XHq123(    I)= XHq123a(    I)+ XHq123b(    I)*AQK*F123(I) !*** FULL A
COT
           XHqSUBQ123(I)= XHqSUBQ123a(I)+ XHqSUBQ123b(I)*AQK*F123(I) !*** FULL A
COT

        ENDDO
C23456789012345678901234567890123456789012345678901234567890123456789012
C----------------------------------------------------
C   ***  COMPUTE TOTAL STRUCTURE FUNCTIONS
C----------------------------------------------------
       DO N=1,3,1
          XTOT123(N) = XLO123(N) + XNLO123(N) - XSUB123(N) +   !*** FULL ACOT
     >                        1.0* (XHq123(N) - XHqSUBQ123(N) )
          XTOT123S(N)= XLO1230(N)+ XNLO123(N) - XSUB1230(N) +  !*** SHORT-CUT TO
T (S-ACOT)
     >                                XNLO123Qz(N)
          XTOT1230(N)= XLO1230(N)+ XG123(N)    +               !*** MASSLESS MS-
BAR
```

Figure E2: Program which selects *ACOT* scheme; pages 1-2.

```
     >                             XNLO123Qz(N)
          XTOT123X(N) = XLO123(N) + XNLO123(N) - XSUB123(N) +  !*** Test ACOT (-
NLO Q)
     >                        0.0* (XHq123(N) - XHqSUBQ123(N) )
C  NEW: FULL ACOT BUT WITH ZERO MASS NLO-Q TERMS::
          XTOT123y(N) = XLO123(N) + XNLO123(N) - XSUB123(N) +   !*** NEW
     >                         XNLO123Qz(N)
C----------------------------------------------------
C   ***  Set output
C----------------------------------------------------
          Xout123(N) = 0.0d0
          If(Ischeme.eq.0) Xout123(N) =  XTOT1230(N)   !*** MASSLESS MS-BAR
          If(Ischeme.eq.1) Xout123(N) =  XTOT123( N)   !*** FULL ACOT
          If(Ischeme.eq.2) Xout123(N) =  XNLO123(N)    !*** FFS
          If(Ischeme.eq.3) Xout123(N) =  XTOT123S(N)   !*** SHORT-CUT TOT (S-ACO
T)
          If(Ischeme.eq.4) Xout123(N) =  XTOT123X(N)   !*** Test ACOT (NO NLO Q)
          If(Ischeme.eq.5) Xout123(N) =  XLO123(N)     !*** LO
          If(Ischeme.eq.6) Xout123(N) =  XLO1230(N)    !*** Massless LO
          If(Ischeme.eq.7) Xout123(N) =  XTOT123Y(N)   !*** FULL ACOT BUT WITH Z
ERO MASS NLO-Q TERMS::
       ENDDO
       END
```

Figure E3: Program which selects *ACOT* scheme;pages 3-4.

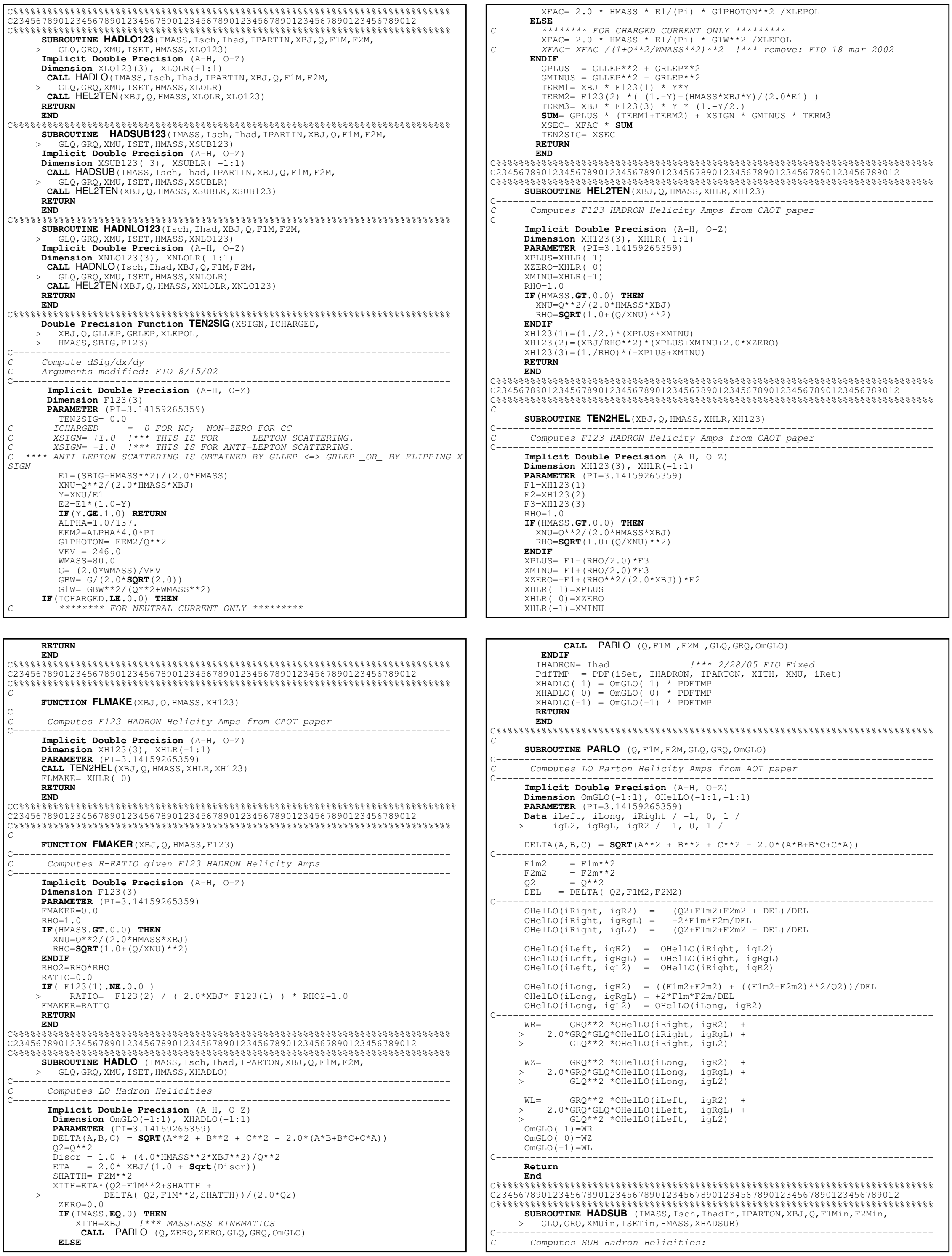
```
C=============================================================================
      FUNCTION PDF(iSet, IHADRONin, IPARTONin, x, q, iRet)
C=============================================================================
C       Front end program for  cteq4 pdf's: Fred Olness  6/20/96
C=============================================================================
      Implicit Double Precision (A-H, O-Z)
      DATA IPARM,IPRINT /2,1/
      SAVE IPARM,IPRINT,IHADRON
      COMMON/Q2STUF/ Q02,Q2MAX
      COMMON/lambda/xlamfred(60)
      COMMON/DERIV/NDRV
C=============================================================================
C=============================================================================
      IHADRON=IHADRONin
      IPARTON=IPARTONin
C=============================================================================
      PDF=0.0D0
      IF((X.GE.1.0).OR.(X.LE.0.0)) RETURN
C=============================================================================
      IF(IPRINT.NE.0)
     >   WRITE(6,*) ' IHADRON: SET TO HADRON=',IHADRON

      If((IHADRON.eq.1).or.(IHADRON.eq.0)) then
      else
        write(6,*) " !*** Only setup for PROTON/NEUTRON "
        STOP
      endif
C=============================================================================
C     IMPLEMENT NEUTRON
      IF(IHADRON.EQ.0) THEN
            IF(IPARTON.EQ. 1)  THEN
                               IPARTON= 2
        ELSEIF(IPARTON.EQ. 2)  THEN
                               IPARTON= 1
        ELSEIF(IPARTON.EQ.-1)  THEN
                               IPARTON=-2
        ELSEIF(IPARTON.EQ.-2)  THEN
                               IPARTON=-1
        ENDIF
      ENDIF
C=============================================================================
      IF(IPARM.EQ.0) THEN
C -------------- THIS OPTIONS USES THE TABLES
        Call SetCtq4(Iset)
        temp=Ctq4Pdf (Iparton, X, Q)
        pdf=temp
        IF(IPRINT.NE.0) WRITE(6,*) ' USING CTEQ4 TABLES. ISET =',ISET
      ELSEIF(IPARM.eq.1) then
C -------------- THIS OPTIONS USES THE PARAMETERIZATION
        temp =  CTQ4PD (ISET, IPARTON, X, Q, IRT2)
        pdf=temp
        IF(IPRINT.NE.0) WRITE(6,*) ' USING CTEQ4 PARM. ISET =',ISET
      ELSEIF(IPARM.eq.2) then

         Q2=Q*Q
         XLAM=xlamfred(1)
         S0=DLOG(Q02/XLam**2)
         S=DLOG(DLOG(Q2/XLam**2)/S0)
         call pdfjo(ndrv,x,s,u,d,ub,db,sb,cb,bb,glue)
         if (IPARTON.eq.1) then
            temp=u
         elseif (IPARTON.eq.2) then
            temp=d
         elseif (IPARTON.eq.3) then
            temp=sb
         elseif (IPARTON.eq.4) then
            temp=cb
         elseif (IPARTON.eq.5) then
            temp=bb
         elseif (IPARTON.eq.6) then
            temp=0
         elseif (IPARTON.eq.-1) then
            temp=ub
         elseif (IPARTON.eq.-2) then
```

```
            temp=db
         elseif (IPARTON.eq.-3) then
            temp=sb
         elseif (IPARTON.eq.-4) then
            temp=cb
         elseif (IPARTON.eq.-5) then
            temp=bb
         elseif (IPARTON.eq.-6) then
            temp=0
         elseif (IPARTON.eq.0) then
            temp=glue
         endif
         pdf=temp/x
      ENDIF
      IPRINT=0
      RETURN
      END
C=============================================================================
C=============================================================================
```

Figure E4: *ACOT* program which calls the parametrized PDFs.

```
       Subroutine test_a()
C ================================================================
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      Data q,iset /91.,1/
1      Write(6,*)  ' ENTER: q,iset'
       Read( 5,*)   q,iset
       alphas=ascteq4(q,iset)
       Write(6,*)   q,iset,alphas
       GOTO 1
       End
C========================================================================
      function alQCD(Q,iset)
C========================================================================
C     *** THIS IS A DUMMY FUNCTION **********
C========================================================================
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/INPUT/INS,NVL,NMAX,DELTA,S0,IORD
      COMMON/lambda/xlamfred(60)
      COMMON/renscale/xmuF,xmuR
      Qin=xmuR
      write(6,*)  ' patch xmuR in alQCD '
      alQCD = alpha_s(iord+1,Qin*Qin,xlamfred(1),neff)
      RETURN
      END
```

Figure E5: *ACOT* program calling α_S.



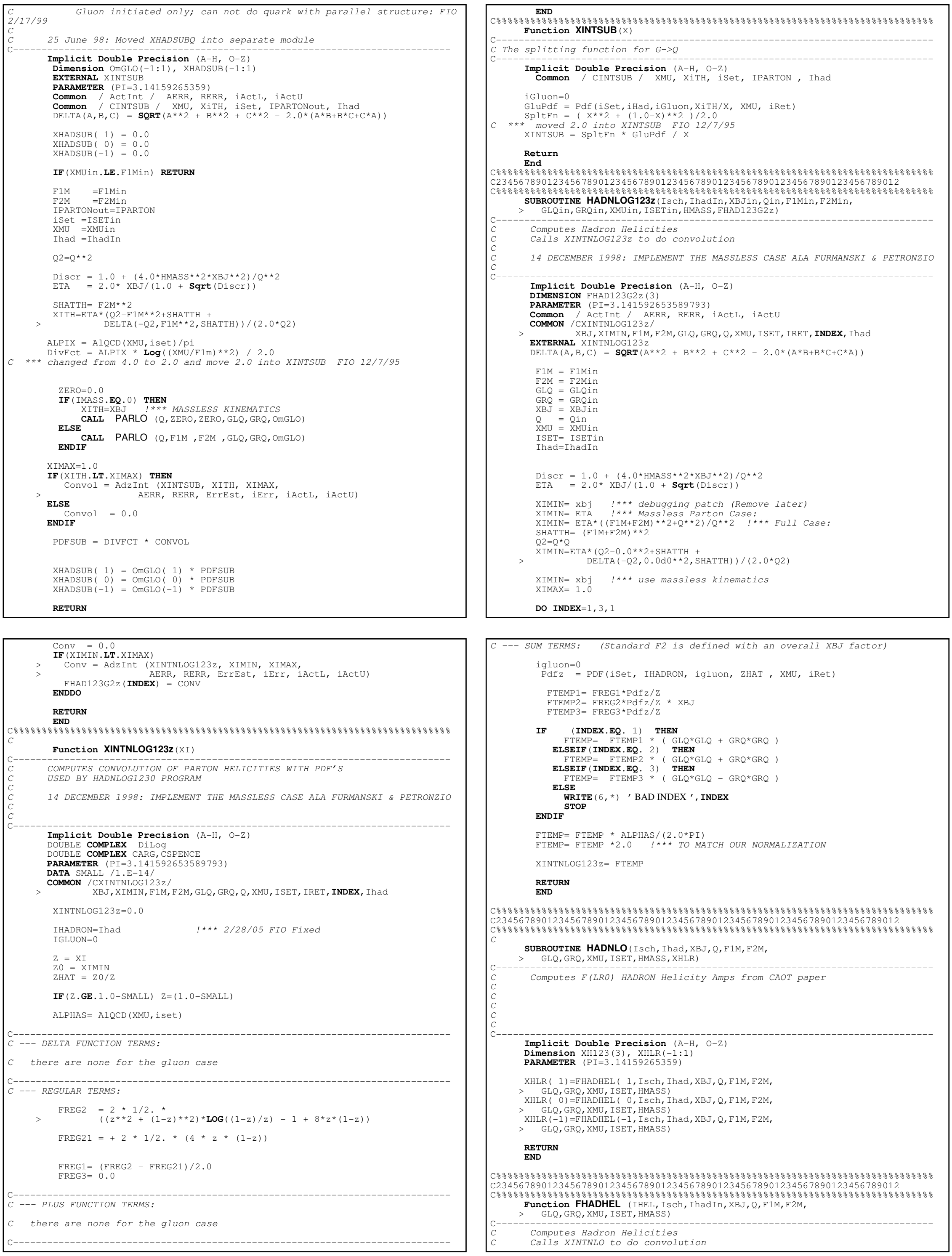
```
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C23456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      SUBROUTINE HADLO123(IMASS,Isch,Ihad,IPARTIN,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS,XLO123)
      Implicit Double Precision (A-H, O-Z)
      Dimension XLO123(3), XLOLR(-1:1)
       CALL HADLO(IMASS,Isch,Ihad,IPARTIN,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS,XLOLR)
       CALL HEL2TEN(XBJ,Q,HMASS,XLOLR,XLO123)
      RETURN
      END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      SUBROUTINE  HADSUB123(IMASS,Isch,Ihad,IPARTIN,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS,XSUB123)
      Implicit Double Precision (A-H, O-Z)
      Dimension XSUB123( 3), XSUBLR( -1:1)
       CALL HADSUB(IMASS,Isch,Ihad,IPARTIN,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS,XSUBLR)
       CALL HEL2TEN(XBJ,Q,HMASS,XSUBLR,XSUB123)
      RETURN
      END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      SUBROUTINE HADNLO123(Isch,Ihad,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS,XNLO123)
      Implicit Double Precision (A-H, O-Z)
      Dimension XNLO123(3), XNLOLR(-1:1)
       CALL HADNLO(Isch,Ihad,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS,XNLOLR)
       CALL HEL2TEN(XBJ,Q,HMASS,XNLOLR,XNLO123)
      RETURN
      END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      Double Precision Function TEN2SIG(XSIGN,ICHARGED,
     >   XBJ,Q,GLLEP,GRLEP,XLEPOL,
     >   HMASS,SBIG,F123)
C-----------------------------------------------------------------------------
C     Compute dSig/dx/dy
C     Arguments modified: FIO 8/15/02
C-----------------------------------------------------------------------------
       Implicit Double Precision (A-H, O-Z)
       Dimension F123(3)
       PARAMETER (PI=3.14159265359)
         TEN2SIG= 0.0
C       ICHARGED     =  0 FOR NC;  NON-ZERO FOR CC
C       XSIGN= +1.0  !*** THIS IS FOR      LEPTON SCATTERING.
C       XSIGN= -1.0  !*** THIS IS FOR ANTI-LEPTON SCATTERING.
C  **** ANTI-LEPTON SCATTERING IS OBTAINED BY GLLEP <=> GRLEP _OR_ BY FLIPPING X
SIGN
         E1=(SBIG-HMASS**2)/(2.0*HMASS)
         XNU=Q**2/(2.0*HMASS*XBJ)
         Y=XNU/E1
         E2=E1*(1.0-Y)
         IF(Y.GE.1.0) RETURN
         ALPHA=1.0/137.
         EEM2=ALPHA*4.0*PI
         G1PHOTON= EEM2/Q**2
         VEV = 246.0
         WMASS=80.0
         G= (2.0*WMASS)/VEV
         GBW= G/(2.0*SQRT(2.0))
         G1W= GBW**2/(Q**2+WMASS**2)
      IF(ICHARGED.LE.0.0) THEN
C        ******** FOR NEUTRAL CURRENT ONLY *********
```

```
         XFAC= 2.0 * HMASS * E1/(Pi) * G1PHOTON**2 /XLEPOL
       ELSE
C        ******** FOR CHARGED CURRENT ONLY *********
         XFAC= 2.0 * HMASS * E1/(Pi) * G1W**2 /XLEPOL
C        XFAC= XFAC /(1+Q**2/WMASS**2)**2  !*** remove: FIO 18 mar 2002
       ENDIF
         GPLUS  = GLLEP**2 + GRLEP**2
         GMINUS = GLLEP**2 - GRLEP**2
         TERM1= XBJ * F123(1) * Y*Y
         TERM2= F123(2) *( (1.-Y)-(HMASS*XBJ*Y)/(2.0*E1) )
         TERM3= XBJ * F123(3) * Y * (1.-Y/2.)
         SUM= GPLUS * (TERM1+TERM2) + XSIGN * GMINUS * TERM3
         XSEC= XFAC * SUM
         TEN2SIG= XSEC
        RETURN
        END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C23456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      SUBROUTINE HEL2TEN(XBJ,Q,HMASS,XHLR,XH123)
C-----------------------------------------------------------------------------
C      Computes F123 HADRON Helicity Amps from CAOT paper
C-----------------------------------------------------------------------------
      Implicit Double Precision (A-H, O-Z)
      Dimension XH123(3), XHLR(-1:1)
      PARAMETER (PI=3.14159265359)
      XPLUS=XHLR( 1)
      XZERO=XHLR( 0)
      XMINU=XHLR(-1)
      RHO=1.0
      IF(HMASS.GT.0.0) THEN
        XNU=Q**2/(2.0*HMASS*XBJ)
        RHO=SQRT(1.0+(Q/XNU)**2)
      ENDIF
      XH123(1)=(1./2.)*(XPLUS+XMINU)
      XH123(2)=(XBJ/RHO**2)*(XPLUS+XMINU+2.0*XZERO)
      XH123(3)=(1./RHO)*(-XPLUS+XMINU)
      RETURN
      END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C23456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE TEN2HEL(XBJ,Q,HMASS,XHLR,XH123)
C-----------------------------------------------------------------------------
C      Computes F123 HADRON Helicity Amps from CAOT paper
C-----------------------------------------------------------------------------
      Implicit Double Precision (A-H, O-Z)
      Dimension XH123(3), XHLR(-1:1)
      PARAMETER (PI=3.14159265359)
      F1=XH123(1)
      F2=XH123(2)
      F3=XH123(3)
      RHO=1.0
      IF(HMASS.GT.0.0) THEN
        XNU=Q**2/(2.0*HMASS*XBJ)
        RHO=SQRT(1.0+(Q/XNU)**2)
      ENDIF
      XPLUS= F1-(RHO/2.0)*F3
      XMINU= F1+(RHO/2.0)*F3
      XZERO=-F1+(RHO**2/(2.0*XBJ))*F2
      XHLR( 1)=XPLUS
      XHLR( 0)=XZERO
      XHLR(-1)=XMINU
```

```
      RETURN
      END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C23456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      FUNCTION FLMAKE(XBJ,Q,HMASS,XH123)
C-----------------------------------------------------------------------------
C      Computes F123 HADRON Helicity Amps from CAOT paper
C-----------------------------------------------------------------------------
      Implicit Double Precision (A-H, O-Z)
      Dimension XH123(3), XHLR(-1:1)
      PARAMETER (PI=3.14159265359)
      CALL TEN2HEL(XBJ,Q,HMASS,XHLR,XH123)
      FLMAKE= XHLR( 0)
      RETURN
      END
CC%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C23456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      FUNCTION FMAKER(XBJ,Q,HMASS,F123)
C-----------------------------------------------------------------------------
C      Computes R-RATIO given F123 HADRON Helicity Amps
C-----------------------------------------------------------------------------
      Implicit Double Precision (A-H, O-Z)
      Dimension F123(3)
      PARAMETER (PI=3.14159265359)
      FMAKER=0.0
      RHO=1.0
      IF(HMASS.GT.0.0) THEN
        XNU=Q**2/(2.0*HMASS*XBJ)
        RHO=SQRT(1.0+(Q/XNU)**2)
      ENDIF
      RHO2=RHO*RHO
      RATIO=0.0
      IF( F123(1).NE.0.0 )
     >     RATIO=  F123(2) / ( 2.0*XBJ* F123(1) ) * RHO2-1.0
      FMAKER=RATIO
      RETURN
      END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C23456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      SUBROUTINE HADLO (IMASS,Isch,Ihad,IPARTON,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS,XHADLO)
C-----------------------------------------------------------------------------
C      Computes LO Hadron Helicities
C-----------------------------------------------------------------------------
       Implicit Double Precision (A-H, O-Z)
        Dimension OmGLO(-1:1), XHADLO(-1:1)
        PARAMETER (PI=3.14159265359)
        DELTA(A,B,C) = SQRT(A**2 + B**2 + C**2 - 2.0*(A*B+B*C+C*A))
        Q2=Q**2
        Discr = 1.0 + (4.0*HMASS**2*XBJ**2)/Q**2
        ETA   = 2.0* XBJ/(1.0 + Sqrt(Discr))
        SHATTH= F2M**2
        XITH=ETA*(Q2-F1M**2+SHATTH +
     >           DELTA(-Q2,F1M**2,SHATTH))/(2.0*Q2)
         ZERO=0.0
         IF(IMASS.EQ.0) THEN
            XITH=XBJ   !*** MASSLESS KINEMATICS
             CALL  PARLO (Q,ZERO,ZERO,GLQ,GRQ,OmGLO)
         ELSE
```

```
             CALL  PARLO (Q,F1M ,F2M ,GLQ,GRQ,OmGLO)
         ENDIF
        IHADRON= Ihad              !*** 2/28/05 FIO Fixed
        PdfTMP  = PDF(iSet, IHADRON, IPARTON, XITH, XMU, iRet)
        XHADLO( 1) = OmGLO( 1) * PDFTMP
        XHADLO( 0) = OmGLO( 0) * PDFTMP
        XHADLO(-1) = OmGLO(-1) * PDFTMP
        RETURN
        END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE PARLO (Q,F1M,F2M,GLQ,GRQ,OmGLO)
C-----------------------------------------------------------------------------
C      Computes LO Parton Helicity Amps from AOT paper
C-----------------------------------------------------------------------------
      Implicit Double Precision (A-H, O-Z)
      Dimension OmGLO(-1:1), OHelLO(-1:1,-1:1)
      PARAMETER (PI=3.14159265359)
      Data iLeft, iLong, iRight / -1, 0, 1 /
     >     igL2, igRgL, igR2 / -1, 0, 1 /

      DELTA(A,B,C) = SQRT(A**2 + B**2 + C**2 - 2.0*(A*B+B*C+C*A))
C-----------------------------------------------------------------------------
      F1m2    = F1m**2
      F2m2    = F2m**2
      Q2      = Q**2
      DEL   = DELTA(-Q2,F1M2,F2M2)
C-----------------------------------------------------------------------------
      OHelLO(iRight, igR2)  =   (Q2+F1m2+F2m2 + DEL)/DEL
      OHelLO(iRight, igRgL) =   -2*F1m*F2m/DEL
      OHelLO(iRight, igL2)  =   (Q2+F1m2+F2m2 - DEL)/DEL

      OHelLO(iLeft, igR2)  =  OHelLO(iRight, igL2)
      OHelLO(iLeft, igRgL) =  OHelLO(iRight, igRgL)
      OHelLO(iLeft, igL2)  =  OHelLO(iRight, igR2)

      OHelLO(iLong, igR2)  = ((F1m2+F2m2) + ((F1m2-F2m2)**2/Q2))/DEL
      OHelLO(iLong, igRgL) = +2*F1m*F2m/DEL
      OHelLO(iLong, igL2)  = OHelLO(iLong, igR2)
C-----------------------------------------------------------------------------
      WR=     GRQ**2 *OHelLO(iRight, igR2)  +
     >    2.0*GRQ*GLQ*OHelLO(iRight, igRgL) +
     >        GLQ**2 *OHelLO(iRight, igL2)

      WZ=     GRQ**2 *OHelLO(iLong,  igR2)  +
     >    2.0*GRQ*GLQ*OHelLO(iLong,  igRgL) +
     >        GLQ**2 *OHelLO(iLong,  igL2)

      WL=     GRQ**2 *OHelLO(iLeft,  igR2)  +
     >    2.0*GRQ*GLQ*OHelLO(iLeft,  igRgL) +
     >        GLQ**2 *OHelLO(iLeft,  igL2)
      OmGLO( 1)=WR
      OmGLO( 0)=WZ
      OmGLO(-1)=WL
C-----------------------------------------------------------------------------
      Return
      End
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C23456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      SUBROUTINE HADSUB (IMASS,Isch,IhadIn,IPARTON,XBJ,Q,F1Min,F2Min,
     >   GLQ,GRQ,XMUin,ISETin,HMASS,XHADSUB)
C-----------------------------------------------------------------------------
C      Computes SUB Hadron Helicities:
```

Figure E6: Program which computes each diagram; pages 1-4.



```
C           Gluon initiated only; can not do quark with parallel structure: FIO
2/17/99
C
C      25 June 98: Moved XHADSUBQ into separate module
C-----------------------------------------------------------------------------
       Implicit Double Precision (A-H, O-Z)
        Dimension OmGLO(-1:1), XHADSUB(-1:1)
        EXTERNAL XINTSUB
        PARAMETER (PI=3.14159265359)
        Common  / ActInt /  AERR, RERR, iActL, iActU
        Common  / CINTSUB /  XMU, XiTH, iSet, IPARTONout, Ihad
        DELTA(A,B,C) = SQRT(A**2 + B**2 + C**2 - 2.0*(A*B+B*C+C*A))

        XHADSUB( 1) = 0.0
        XHADSUB( 0) = 0.0
        XHADSUB(-1) = 0.0

        IF(XMUin.LE.F1Min) RETURN

        F1M    =F1Min
        F2M    =F2Min
        IPARTONout=IPARTON
        iSet =ISETin
        XMU  =XMUin
        Ihad =IhadIn

        Q2=Q**2

        Discr = 1.0 + (4.0*HMASS**2*XBJ**2)/Q**2
        ETA   = 2.0* XBJ/(1.0 + Sqrt(Discr))

        SHATTH= F2M**2
        XITH=ETA*(Q2-F1M**2+SHATTH +
     >           DELTA(-Q2,F1M**2,SHATTH))/(2.0*Q2)

       ALPIX = AlQCD(XMU,iset)/pi
       DivFct = ALPIX * Log((XMU/F1m)**2) / 2.0
C  *** changed from 4.0 to 2.0 and move 2.0 into XINTSUB  FIO 12/7/95


         ZERO=0.0
         IF(IMASS.EQ.0) THEN
             XITH=XBJ   !*** MASSLESS KINEMATICS
             CALL  PARLO (Q,ZERO,ZERO,GLQ,GRQ,OmGLO)
         ELSE
             CALL  PARLO (Q,F1M ,F2M ,GLQ,GRQ,OmGLO)
         ENDIF

       XIMAX=1.0
       IF(XITH.LT.XIMAX) THEN
          Convol = AdzInt (XINTSUB, XITH, XIMAX,
     >                 AERR, RERR, ErrEst, iErr, iActL, iActU)
       ELSE
          Convol  = 0.0
       ENDIF

        PDFSUB = DIVFCT * CONVOL


        XHADSUB( 1) = OmGLO( 1) * PDFSUB
        XHADSUB( 0) = OmGLO( 0) * PDFSUB
        XHADSUB(-1) = OmGLO(-1) * PDFSUB

        RETURN
```

```
        END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      Function XINTSUB(X)
C-----------------------------------------------------------------------------
C The splitting function for G->Q
C-----------------------------------------------------------------------------
      Implicit Double Precision (A-H, O-Z)
        Common  / CINTSUB /  XMU, XiTH, iSet, IPARTON , Ihad

      iGluon=0
      GluPdf = Pdf(iSet,iHad,iGluon,XiTH/X, XMU, iRet)
      SpltFn = ( X**2 + (1.0-X)**2 )/2.0
C  ***  moved 2.0 into XINTSUB  FIO 12/7/95
      XINTSUB = SpltFn * GluPdf / X

      Return
      End
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C234567890123456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      SUBROUTINE HADNLOG123z(Isch,IhadIn,XBJin,Qin,F1Min,F2Min,
     >   GLQin,GRQin,XMUin,ISETin,HMASS,FHAD123G2z)
C-----------------------------------------------------------------------------
C      Computes Hadron Helicities
C      Calls XINTNLOG123z to do convolution
C
C      14 DECEMBER 1998: IMPLEMENT THE MASSLESS CASE ALA FURMANSKI & PETRONZIO
C
C-----------------------------------------------------------------------------
       Implicit Double Precision (A-H, O-Z)
       DIMENSION FHAD123G2z(3)
       PARAMETER (PI=3.141592653589793)
       Common  / ActInt /  AERR, RERR, iActL, iActU
       COMMON /CXINTNLOG123z/
     >         XBJ,XIMIN,F1M,F2M,GLQ,GRQ,Q,XMU,ISET,IRET,INDEX,Ihad
       EXTERNAL XINTNLOG123z
       DELTA(A,B,C) = SQRT(A**2 + B**2 + C**2 - 2.0*(A*B+B*C+C*A))

        F1M = F1Min
        F2M = F2Min
        GLQ = GLQin
        GRQ = GRQin
        XBJ = XBJin
        Q   = Qin
        XMU = XMUin
        ISET= ISETin
        Ihad=IhadIn


        Discr = 1.0 + (4.0*HMASS**2*XBJ**2)/Q**2
        ETA   = 2.0* XBJ/(1.0 + Sqrt(Discr))

        XIMIN= xbj   !*** debugging patch (Remove later)
        XIMIN= ETA   !*** Massless Parton Case:
        XIMIN= ETA*((F1M+F2M)**2+Q**2)/Q**2  !*** Full Case:
        SHATTH= (F1M+F2M)**2
        Q2=Q*Q
        XIMIN=ETA*(Q2-0.0**2+SHATTH +
     >           DELTA(-Q2,0.0d0**2,SHATTH))/(2.0*Q2)

        XIMIN= xbj   !*** use massless kinematics
        XIMAX= 1.0

        DO INDEX=1,3,1
```

```
        Conv  = 0.0
        IF(XIMIN.LT.XIMAX)
     >    Conv = AdzInt (XINTNLOG123z, XIMIN, XIMAX,
     >                  AERR, RERR, ErrEst, iErr, iActL, iActU)
          FHAD123G2z(INDEX) = CONV
        ENDDO

        RETURN
        END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
        Function XINTNLOG123z(XI)
C-----------------------------------------------------------------------------
C      COMPUTES CONVOLUTION OF PARTON HELICITIES WITH PDF'S
C      USED BY HADNLOG1230 PROGRAM
C
C      14 DECEMBER 1998: IMPLEMENT THE MASSLESS CASE ALA FURMANSKI & PETRONZIO
C
C
C-----------------------------------------------------------------------------
       Implicit Double Precision (A-H, O-Z)
       DOUBLE COMPLEX  DiLog
       DOUBLE COMPLEX CARG,CSPENCE
       PARAMETER (PI=3.141592653589793)
       DATA SMALL /1.E-14/
       COMMON /CXINTNLOG123z/
     >         XBJ,XIMIN,F1M,F2M,GLQ,GRQ,Q,XMU,ISET,IRET,INDEX,Ihad

        XINTNLOG123z=0.0

        IHADRON=Ihad             !*** 2/28/05 FIO Fixed
        IGLUON=0

        Z = XI
        Z0 = XIMIN
        ZHAT = Z0/Z

        IF(Z.GE.1.0-SMALL) Z=(1.0-SMALL)

        ALPHAS= AlQCD(XMU,iset)

C-----------------------------------------------------------------------------
C --- DELTA FUNCTION TERMS:

C   there are none for the gluon case

C-----------------------------------------------------------------------------
C --- REGULAR TERMS:

         FREG2  = 2 * 1/2. *
     >         ((z**2 + (1-z)**2)*LOG((1-z)/z) - 1 + 8*z*(1-z))

         FREG21 = + 2 * 1/2. * (4 * z * (1-z))


         FREG1= (FREG2 - FREG21)/2.0
         FREG3= 0.0

C-----------------------------------------------------------------------------
C --- PLUS FUNCTION TERMS:

C   there are none for the gluon case

C-----------------------------------------------------------------------------
```

```
C --- SUM TERMS:   (Standard F2 is defined with an overall XBJ factor)

        igluon=0
         Pdfz  = PDF(iSet, IHADRON, igluon, ZHAT , XMU, iRet)

          FTEMP1= FREG1*Pdfz/Z
          FTEMP2= FREG2*Pdfz/Z * XBJ
          FTEMP3= FREG3*Pdfz/Z

        IF    (INDEX.EQ. 1)  THEN
             FTEMP=  FTEMP1 * ( GLQ*GLQ + GRQ*GRQ )
           ELSEIF(INDEX.EQ. 2)  THEN
             FTEMP=  FTEMP2 * ( GLQ*GLQ + GRQ*GRQ )
           ELSEIF(INDEX.EQ. 3)  THEN
             FTEMP=  FTEMP3 * ( GLQ*GLQ - GRQ*GRQ )
           ELSE
             WRITE(6,*) ' BAD INDEX ',INDEX
             STOP
        ENDIF

        FTEMP= FTEMP * ALPHAS/(2.0*PI)
        FTEMP= FTEMP *2.0   !*** TO MATCH OUR NORMALIZATION

        XINTNLOG123z= FTEMP

        RETURN
        END

C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C234567890123456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE HADNLO(Isch,Ihad,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS,XHLR)
C-----------------------------------------------------------------------------
C      Computes F(LR0) HADRON Helicity Amps from CAOT paper
C
C
C
C
C
C-----------------------------------------------------------------------------
      Implicit Double Precision (A-H, O-Z)
      Dimension XH123(3), XHLR(-1:1)
      PARAMETER (PI=3.14159265359)

      XHLR( 1)=FHADHEL( 1,Isch,Ihad,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS)
      XHLR( 0)=FHADHEL( 0,Isch,Ihad,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS)
      XHLR(-1)=FHADHEL(-1,Isch,Ihad,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS)

      RETURN
      END

C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C234567890123456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      Function FHADHEL (IHEL,Isch,IhadIn,XBJ,Q,F1M,F2M,
     >   GLQ,GRQ,XMU,ISET,HMASS)
C-----------------------------------------------------------------------------
C      Computes Hadron Helicities
C      Calls XINTNLO to do convolution
```

Figure E7: Program which computes each diagram; pages 5-8.

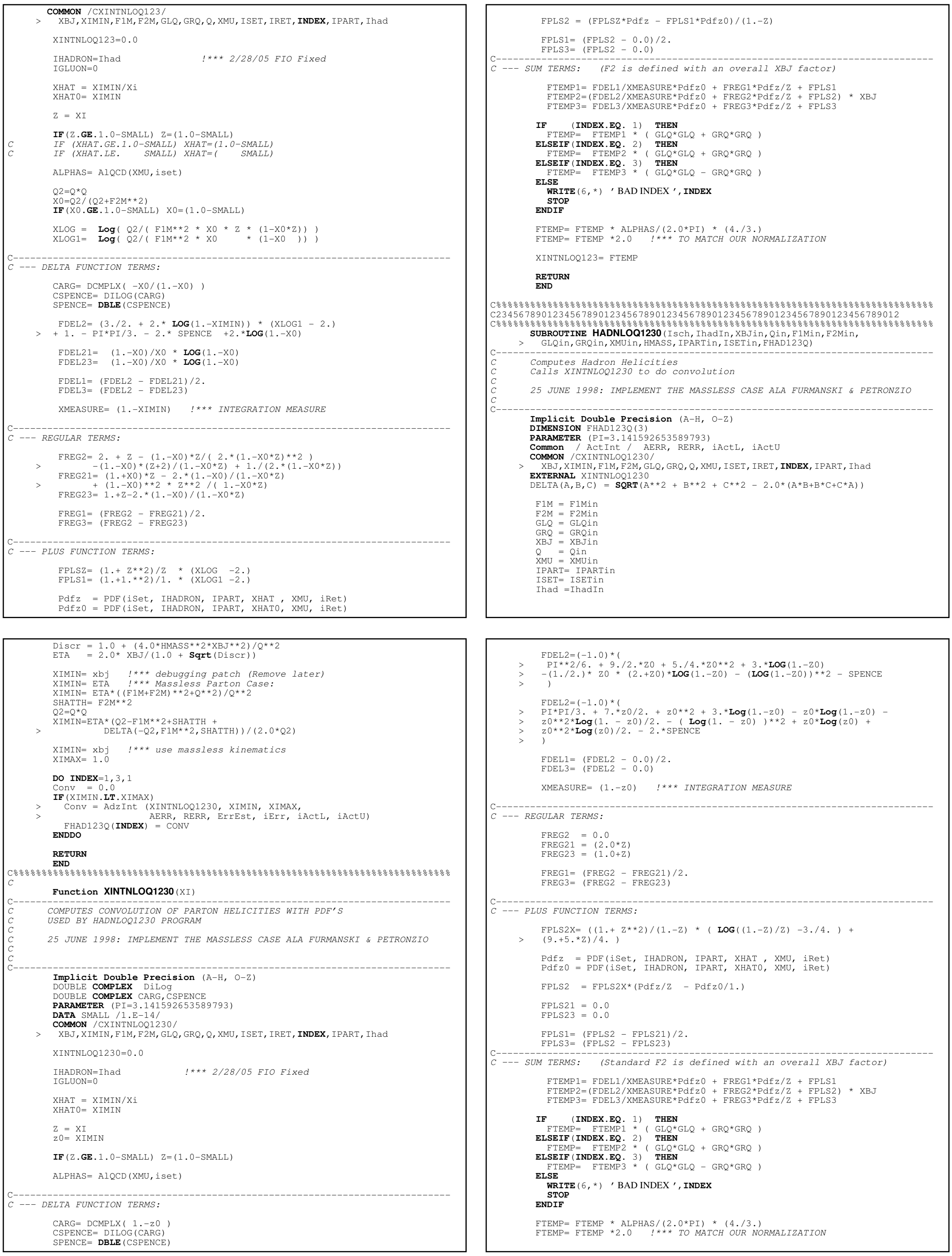
```
C
C
C
C
C-----------------------------------------------------------------------------
      Implicit Double Precision (A-H, O-Z)
       PARAMETER (PI=3.14159265359)
       Common  / ActInt /  AERR, RERR, iActL, iActU
       COMMON /CXINTNLO/ ETA,Qout,F1Mout,F2Mout,GLQout,
     >                GRQout,XMUout,ISETout,IRETout,IHELout,Ihad
       EXTERNAL XINTNLO

        Discr = 1.0 + (4.0*HMASS**2*XBJ**2)/Q**2
        ETA   = 2.0* XBJ/(1.0 + Sqrt(Discr))

        IHELout=IHEL
        Qout   =Q
        F1Mout =F1M
        F2Mout =F2M
        GLQout  =GLQ
        GRQout  =GRQ
        XMUout =XMU
        ISETout=ISET
        Ihad   =Ihadin

        XIMIN= ETA*((F1M+F2M)**2+Q**2)/Q**2
        XIMAX= 1.0

        Conv  = 0.0
        IF(XIMIN.LT.XIMAX)
     >  Conv = AdzInt (XINTNLO, XIMIN, XIMAX,
     >                   AERR, RERR, ErrEst, iErr, iActL, iActU)

        FHADHEL = CONV

        RETURN
        END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
        Function XINTNLO(XI)
C-----------------------------------------------------------------------------
C      COMPUTES CONVOLUTION OF PARTON HELICITIES WITH PDF'S
C      USED BY HADHEL PROGRAM
C
C
C
C
C-----------------------------------------------------------------------------
       Implicit Double Precision (A-H, O-Z)
       Dimension OmGNLO(-1:1)
       PARAMETER (PI=3.14159265359)
       COMMON /CXINTNLO/
     >   ETA,Q,F1M,F2M,GLQ,GRQ,XMU,ISET,IRET,IHEL,Ihad
       COMMON /CXINTNLOTMP/ JSET

        JSET=ISET  !*** THIS IS A PATCH TO PASS ALQCD THE PDF SET: FIO 22SEP99

        IHADRON=1
        IGLUON=0

CCC ******* NOTE, WE USE ETA HERE INSTEAD OF XIMIN BECAUSE THE SLOW-RESCALING
CCC ******* MASS CORRECTION IS BUILT INTO THE MATRIX ELEMENTS AND NEED NOT BE
CCC ******* PUT IN BY HAND
```

```
        XHAT = ETA/XI
        CALL PARNLO (XHAT,Q,F1M,F2M,GLQ,GRQ,XMU,OmGNLO)
        FTEMP = OmGNLO(IHEL)
        PdfTMP  = PDF(iSet, Ihad, IGLUON, XI, XMU, iRet)
        TEMP = FTEMP * PDFTMP / XI
        XINTNLO= TEMP

        RETURN
        END


C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE PARNLO (X,Q,F1M,F2M,GLQ,GRQ,XMU,OmGNLO)
C-----------------------------------------------------------------------------
C      Computes Parton Helicity Amps from CAOT paper
C
C
C
C
C
C-----------------------------------------------------------------------------
      Implicit Double Precision (A-H, O-Z)
      Dimension OmGNLO(-1:1)
      PARAMETER (PI=3.14159265359)
C !*** THIS IS A PATCH TO PASS ALQCD THE PDF SET: FIO 22SEP99
      COMMON /CXINTNLOTMP/ ISET

      DELTA(A,B,C) = SQRT(A**2 + B**2 + C**2 - 2.0*(A*B+B*C+C*A))

      OmGNLO( 1)=0.0
      OmGNLO( 0)=0.0
      OmGNLO(-1)=0.0

      ALPHAS= AlQCD(xMu,iset)

      Q2= Q**2
      F1M2 = F1M ** 2
      F2M2 = F2M ** 2

CCCC   THIS CHECK IS THE SAME AS THE Q CHECK BELOW
      XMAX=Q2/((F1M+F2M)**2+Q2)
      IF((X.GE.1.0).OR.(X.GE.XMAX)) RETURN

CCCC     HMASS=0  !*** TEMP PATCH: THIS IS PARTON LEVEL
CCCC     SMIN=((F1M+F2M+HMASS)**2-HMASS**2)
      SMIN=(F1M+F2M)**2
      QMIN=SQRT( SMIN*X/(1.0-X) )
      IF(Q.LE.QMIN) RETURN

C     Cm-Energy for the hard process for this SHAT=S
      S = Q2 * (1./X - 1.)
      IF(S.LE.SMIN) RETURN
      RS= SQRT(S)

      DEL = DELTA(S, F1M2, F2M2)
      TLOG = + Log(4*F1M2*s/(s+F1M2-F2M2+Del)**2)
      ULOG = + Log(4*F2M2*s/(s-F1M2+F2M2+Del)**2)

      XLAM =  ( Q2 + S)          /(2.0*RS)
      BET =  DEL                /(2.0*RS)
      E1  =  ( F1M2 - F2M2 + S) /(2.0*RS)
      E2  =  (-F1M2 + F2M2 + S) /(2.0*RS)
      EQ  =  (-Q2 + S)          /(2.0*RS)
```

```
      GSPLUS = -((1./2 + e1*(-1. + e1/XLAM)/XLAM)*TLOG) -
     >   (1./2 + e2*(-1. + e2/XLAM)/XLAM)*ULOG
     >   - 2*bet*eq**2/(XLAM**2*rs)

      GXPLUS = -(F1M*F2M*(4*bet*rs + (TLOG + ULOG)*(-F1M2 - F2M2 + s)))/
     >   (4*XLAM**2*s)

      GXPLUS = (2.0)* GXPLUS        !*** ERROR IN ORIGINAL FIO (4/19/93)

      GAPLUS = bet*(-F1M2 + F2M2)/(XLAM**2*rs) -
     >    TLOG*(1./2 + e1*(-1. + e1/XLAM)/XLAM +
     >    F1M2*(-F1M2 + F2M2)/(2*XLAM**2*s)) +
     >    ULOG*(1./2 + e2*(-1. + e2/XLAM)/XLAM -
     >    F2M2*(-F1M2 + F2M2)/(2*XLAM**2*s))

      GSZERO = eq*(-F1M2 + F2M2)*(TLOG - ULOG)/(XLAM**2*rs) +
     >   bet*((-F1M2 + F2M2)**2 +
     >   Q2*(-F1M2 - F2M2 + 2*Q2))/(XLAM**2*Q2*rs) +
     >   (TLOG + ULOG)*(-(eq*(F1M2 + F2M2 - (-F1M2 + F2M2)**2/Q2))/
     >   (2*XLAM**2*rs) + F1M2*F2M2/(XLAM**2*s) +
     >   (F1M2 + F2M2)*(-(-F1M2 + F2M2)**2 + Q2**2 - 2*XLAM**2*s)/
     >   (4*XLAM**2*Q2*s))

      GXZERO = bet*F1M*F2M/(XLAM**2*rs) + F1M*F2M*(TLOG + ULOG)*
     >      (1/Q2 + (-F1M2 - F2M2 + s)/(2*XLAM**2*s))/2

      GXZERO = (2.0)* GXZERO        !*** ERROR IN ORIGINAL FIO (4/19/93)

C-----------------------------------------------------------------------------
C                                                               The Amplitudes
C                                                               --------------
C                                                    The RR helicity amplitude
       ORRGR2 = GSPLUS + GAPLUS
       ORRGLR = GXPLUS
       ORRGL2 = GSPLUS - GAPLUS
C                                             The LONG-LONG helicity amplitude
       OZZGR2 = GSZERO
       OZZGLR = GXZERO
       OZZGL2 = GSZERO
C                                                    The LL helicity amplitude
       OLLGR2 = GSPLUS - GAPLUS
       OLLGLR = GXPLUS
       OLLGL2 = GSPLUS + GAPLUS
C-----------------------------------------------------------------------------
C                                                       Assemble the integrand
C                                                       ----------------------
      XFAC=ALPHAS/PI/2.0
      OmGNLO( 1)=XFAC*(ORRGR2*GRQ**2 +2.0*ORRGLR*GLQ*GRQ +ORRGL2*GLQ**2)
      OmGNLO( 0)=XFAC*(OZZGR2*GRQ**2 +2.0*OZZGLR*GLQ*GRQ +OZZGL2*GLQ**2)
      OmGNLO(-1)=XFAC*(OLLGR2*GRQ**2 +2.0*OLLGLR*GLQ*GRQ +OLLGL2*GLQ**2)
C
      Return
      End
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C23456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      SUBROUTINE HADNLOQ123(Isch,IhadIn,XBJin,Qin,F1Min,F2Min,
     >   GLQin,GRQin,XMUin,HMASS,IPARTin,ISETin,FHAD123Q)
C-----------------------------------------------------------------------------
C      Computes Hadron Helicities
C      Calls XINTNLO to do convolution
C
C      FOR CHARGED CURRENT V-A ONLY, M1 IS PURELY A REGULATOR
```

```
C      ALA GOTTSCHALK, VAN DER BIJ ..., KRAMER ...
C
C-----------------------------------------------------------------------------
       Implicit Double Precision (A-H, O-Z)
       DIMENSION FHAD123Q(3)
       PARAMETER (PI=3.141592653589793)
       Common  / ActInt /  AERR, RERR, iActL, iActU
       COMMON /CXINTNLOQ123/
     >   XBJ,XIMIN,F1M,F2M,GLQ,GRQ,Q,XMU,ISET,IRET,INDEX,IPART,Ihad
       EXTERNAL XINTNLOQ123
       DELTA(A,B,C) = SQRT(A**2 + B**2 + C**2 - 2.0*(A*B+B*C+C*A))

        F1M = F1Min
        F2M = F2Min
        GLQ = GLQin
        GRQ = GRQin
        XBJ = XBJin
        Q   = Qin
        XMU = XMUin
        IPART= IPARTin
        ISET= ISETin
        Ihad= IhadIn


        Discr = 1.0 + (4.0*HMASS**2*XBJ**2)/Q**2
        ETA   = 2.0* XBJ/(1.0 + Sqrt(Discr))

        SHATTH= F2M**2
        Q2=Q*Q

        XIMIN= xbj   !*** debugging patch (Remove later)
        XIMIN= ETA*((F1M+F2M)**2+Q**2)/Q**2
        XIMIN=ETA*(Q2-F1M**2+SHATTH +
     >           DELTA(-Q2,F1M**2,SHATTH))/(2.0*Q2)
        XIMAX= 1.0

        DO INDEX=1,3,1
        Conv  = 0.0
        IF(XIMIN.LT.XIMAX)
     >    Conv = AdzInt (XINTNLOQ123, XIMIN, XIMAX,
     >                   AERR, RERR, ErrEst, iErr, iActL, iActU)
          FHAD123Q(INDEX) = CONV
        ENDDO

        RETURN
        END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
        Function XINTNLOQ123(XI)
C-----------------------------------------------------------------------------
C      COMPUTES CONVOLUTION OF PARTON HELICITIES WITH PDF'S
C      USED BY HADHEL PROGRAM
C
C      FIO 1998
C      ALA GOTTSCHALK, VAN DER BIJ ..., KRAMER ...
C
C
C-----------------------------------------------------------------------------
       Implicit Double Precision (A-H, O-Z)
       DOUBLE COMPLEX  DiLog
       DOUBLE COMPLEX CARG,CSPENCE
       Dimension OmGNLO(-1:1)
       PARAMETER (PI=3.141592653589793)
       DATA SMALL /1.E-14/
```

Figure E8: Program which computes each diagram; pages 9-12.



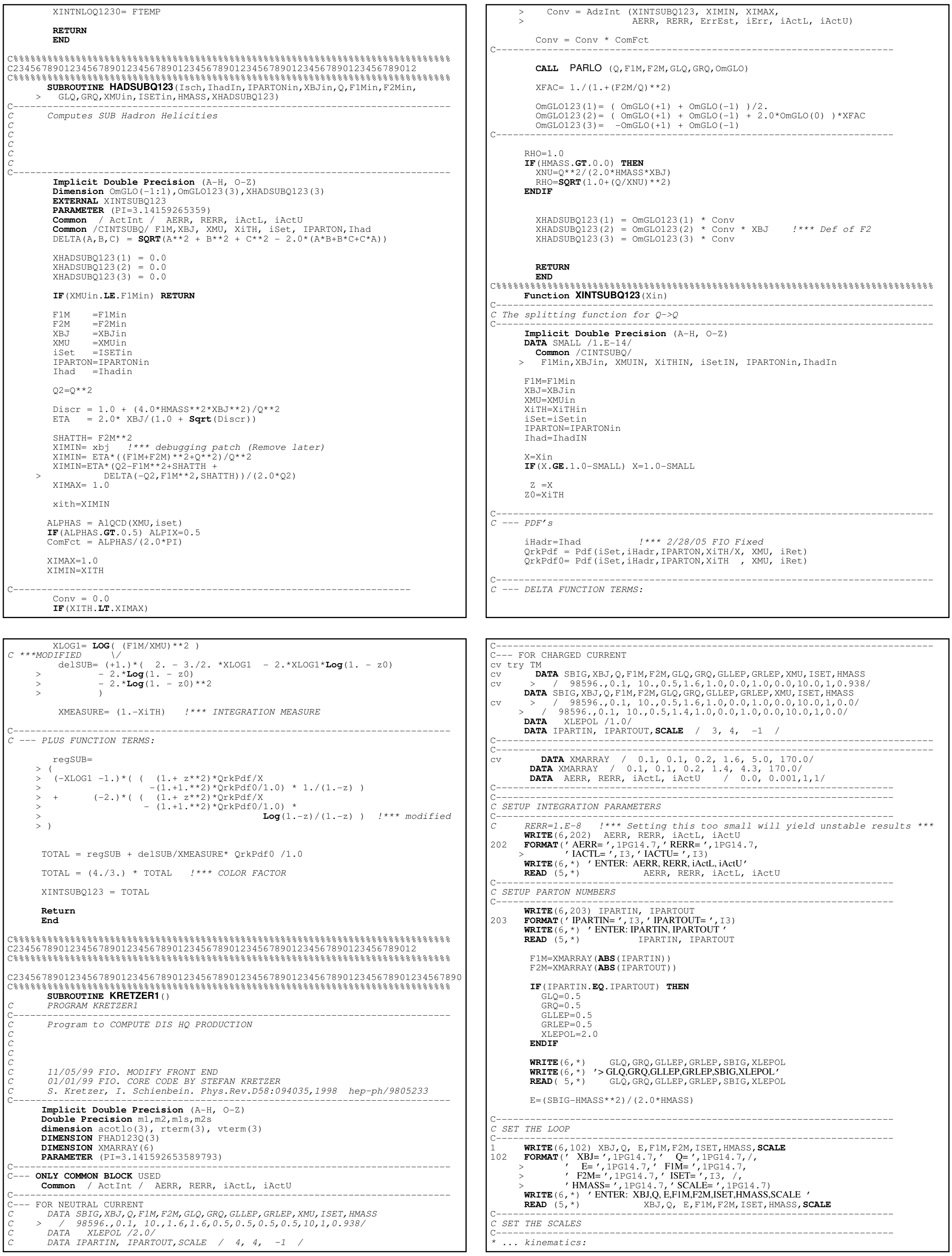
```
       COMMON /CXINTNLOQ123/
     >   XBJ,XIMIN,F1M,F2M,GLQ,GRQ,Q,XMU,ISET,IRET,INDEX,IPART,Ihad

        XINTNLOQ123=0.0

        IHADRON=Ihad              !*** 2/28/05 FIO Fixed
        IGLUON=0

        XHAT = XIMIN/Xi
        XHAT0= XIMIN

        Z = XI

        IF(Z.GE.1.0-SMALL) Z=(1.0-SMALL)
C       IF (XHAT.GE.1.0-SMALL) XHAT=(1.0-SMALL)
C       IF (XHAT.LE.    SMALL) XHAT=(    SMALL)

        ALPHAS= AlQCD(XMU,iset)

        Q2=Q*Q
        X0=Q2/(Q2+F2M**2)
        IF(X0.GE.1.0-SMALL) X0=(1.0-SMALL)

        XLOG =  Log( Q2/( F1M**2 * X0 * Z * (1-X0*Z)) )
        XLOG1=  Log( Q2/( F1M**2 * X0     * (1-X0  )) )

C-----------------------------------------------------------------------------
C --- DELTA FUNCTION TERMS:

        CARG= DCMPLX( -X0/(1.-X0) )
        CSPENCE= DILOG(CARG)
        SPENCE= DBLE(CSPENCE)

         FDEL2= (3./2. + 2.* LOG(1.-XIMIN)) * (XLOG1 - 2.)
     >  + 1. - PI*PI/3. - 2.* SPENCE  +2.*LOG(1.-X0)

         FDEL21=  (1.-X0)/X0 * LOG(1.-X0)
         FDEL23=  (1.-X0)/X0 * LOG(1.-X0)

         FDEL1= (FDEL2 - FDEL21)/2.
         FDEL3= (FDEL2 - FDEL23)

         XMEASURE= (1.-XIMIN)   !*** INTEGRATION MEASURE

C-----------------------------------------------------------------------------
C --- REGULAR TERMS:

         FREG2= 2. + Z - (1.-X0)*Z/( 2.*(1.-X0*Z)**2 )
     >         -(1.-X0)*(Z+2)/(1.-X0*Z) + 1./(2.*(1.-X0*Z))
         FREG21= (1.+X0)*Z - 2.*(1.-X0)/(1.-X0*Z)
     >         + (1.-X0)**2 * Z**2 /( 1.-X0*Z)
         FREG23= 1.+Z-2.*(1.-X0)/(1.-X0*Z)

         FREG1= (FREG2 - FREG21)/2.
         FREG3= (FREG2 - FREG23)

C-----------------------------------------------------------------------------
C --- PLUS FUNCTION TERMS:

         FPLSZ= (1.+ Z**2)/Z  * (XLOG  -2.)
         FPLS1= (1.+1.**2)/1. * (XLOG1 -2.)

         Pdfz  = PDF(iSet, IHADRON, IPART, XHAT , XMU, iRet)
         Pdfz0 = PDF(iSet, IHADRON, IPART, XHAT0, XMU, iRet)
```

```
         FPLS2 = (FPLSZ*Pdfz - FPLS1*Pdfz0)/(1.-Z)

         FPLS1= (FPLS2 - 0.0)/2.
         FPLS3= (FPLS2 - 0.0)
C-----------------------------------------------------------------------------
C --- SUM TERMS:   (F2 is defined with an overall XBJ factor)

          FTEMP1= FDEL1/XMEASURE*Pdfz0 + FREG1*Pdfz/Z + FPLS1
          FTEMP2=(FDEL2/XMEASURE*Pdfz0 + FREG2*Pdfz/Z + FPLS2) * XBJ
          FTEMP3= FDEL3/XMEASURE*Pdfz0 + FREG3*Pdfz/Z + FPLS3

        IF    (INDEX.EQ. 1)  THEN
          FTEMP=  FTEMP1 * ( GLQ*GLQ + GRQ*GRQ )
        ELSEIF(INDEX.EQ. 2)  THEN
          FTEMP=  FTEMP2 * ( GLQ*GLQ + GRQ*GRQ )
        ELSEIF(INDEX.EQ. 3)  THEN
          FTEMP=  FTEMP3 * ( GLQ*GLQ - GRQ*GRQ )
        ELSE
          WRITE(6,*) ' BAD INDEX ',INDEX
          STOP
        ENDIF

        FTEMP= FTEMP * ALPHAS/(2.0*PI) * (4./3.)
        FTEMP= FTEMP *2.0   !*** TO MATCH OUR NORMALIZATION

        XINTNLOQ123= FTEMP

        RETURN
        END

C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C23456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
       SUBROUTINE HADNLOQ1230(Isch,IhadIn,XBJin,Qin,F1Min,F2Min,
     >   GLQin,GRQin,XMUin,HMASS,IPARTin,ISETin,FHAD123Q)
C-----------------------------------------------------------------------------
C      Computes Hadron Helicities
C      Calls XINTNLOQ1230 to do convolution
C
C      25 JUNE 1998: IMPLEMENT THE MASSLESS CASE ALA FURMANSKI & PETRONZIO
C
C-----------------------------------------------------------------------------
       Implicit Double Precision (A-H, O-Z)
       DIMENSION FHAD123Q(3)
       PARAMETER (PI=3.141592653589793)
       Common  / ActInt /  AERR, RERR, iActL, iActU
       COMMON /CXINTNLOQ1230/
     >   XBJ,XIMIN,F1M,F2M,GLQ,GRQ,Q,XMU,ISET,IRET,INDEX,IPART,Ihad
       EXTERNAL XINTNLOQ1230
       DELTA(A,B,C) = SQRT(A**2 + B**2 + C**2 - 2.0*(A*B+B*C+C*A))

        F1M = F1Min
        F2M = F2Min
        GLQ = GLQin
        GRQ = GRQin
        XBJ = XBJin
        Q   = Qin
        XMU = XMUin
        IPART= IPARTin
        ISET= ISETin
        Ihad =IhadIn
```

```
        Discr = 1.0 + (4.0*HMASS**2*XBJ**2)/Q**2
        ETA   = 2.0* XBJ/(1.0 + Sqrt(Discr))

        XIMIN= xbj   !*** debugging patch (Remove later)
        XIMIN= ETA   !*** Massless Parton Case:
        XIMIN= ETA*((F1M+F2M)**2+Q**2)/Q**2
        SHATTH= F2M**2
        Q2=Q*Q
        XIMIN=ETA*(Q2-F1M**2+SHATTH +
     >           DELTA(-Q2,F1M**2,SHATTH))/(2.0*Q2)

        XIMIN= xbj   !*** use massless kinematics
        XIMAX= 1.0

        DO INDEX=1,3,1
        Conv  = 0.0
        IF(XIMIN.LT.XIMAX)
     >    Conv = AdzInt (XINTNLOQ1230, XIMIN, XIMAX,
     >                   AERR, RERR, ErrEst, iErr, iActL, iActU)
          FHAD123Q(INDEX) = CONV
        ENDDO

        RETURN
        END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
        Function XINTNLOQ1230(XI)
C-----------------------------------------------------------------------------
C      COMPUTES CONVOLUTION OF PARTON HELICITIES WITH PDF'S
C      USED BY HADNLOQ1230 PROGRAM
C
C      25 JUNE 1998: IMPLEMENT THE MASSLESS CASE ALA FURMANSKI & PETRONZIO
C
C
C-----------------------------------------------------------------------------
        Implicit Double Precision (A-H, O-Z)
        DOUBLE COMPLEX  DiLog
        DOUBLE COMPLEX CARG,CSPENCE
        PARAMETER (PI=3.141592653589793)
        DATA SMALL /1.E-14/
        COMMON /CXINTNLOQ1230/
     >   XBJ,XIMIN,F1M,F2M,GLQ,GRQ,Q,XMU,ISET,IRET,INDEX,IPART,Ihad

        XINTNLOQ1230=0.0

        IHADRON=Ihad           !*** 2/28/05 FIO Fixed
        IGLUON=0

        XHAT = XIMIN/Xi
        XHAT0= XIMIN

        Z = XI
        z0= XIMIN

        IF(Z.GE.1.0-SMALL) Z=(1.0-SMALL)

        ALPHAS= AlQCD(XMU,iset)

C-----------------------------------------------------------------------------
C --- DELTA FUNCTION TERMS:

        CARG= DCMPLX( 1.-z0 )
        CSPENCE= DILOG(CARG)
        SPENCE= DBLE(CSPENCE)
```

```
         FDEL2=(-1.0)*(
     >    PI**2/6. + 9./2.*Z0 + 5./4.*Z0**2 + 3.*LOG(1.-Z0)
     >   -(1./2.)* Z0 * (2.+Z0)*LOG(1.-Z0) - (LOG(1.-Z0))**2 - SPENCE
     >    )

         FDEL2=(-1.0)*(
     >   PI*PI/3. + 7.*z0/2. + z0**2 + 3.*Log(1.-z0) - z0*Log(1.-z0) -
     >   z0**2*Log(1. - z0)/2. - ( Log(1. - z0) )**2 + z0*Log(z0) +
     >   z0**2*Log(z0)/2. - 2.*SPENCE
     >   )

         FDEL1= (FDEL2 - 0.0)/2.
         FDEL3= (FDEL2 - 0.0)

         XMEASURE= (1.-z0)   !*** INTEGRATION MEASURE

C-----------------------------------------------------------------------------
C --- REGULAR TERMS:

         FREG2  = 0.0
         FREG21 = (2.0*Z)
         FREG23 = (1.0+Z)

         FREG1= (FREG2 - FREG21)/2.
         FREG3= (FREG2 - FREG23)

C-----------------------------------------------------------------------------
C --- PLUS FUNCTION TERMS:

         FPLS2X= ((1.+ Z**2)/(1.-Z) * ( LOG((1.-Z)/Z) -3./4. ) +
     >   (9.+5.*Z)/4. )

         Pdfz  = PDF(iSet, IHADRON, IPART, XHAT , XMU, iRet)
         Pdfz0 = PDF(iSet, IHADRON, IPART, XHAT0, XMU, iRet)

         FPLS2  = FPLS2X*(Pdfz/Z  - Pdfz0/1.)

         FPLS21 = 0.0
         FPLS23 = 0.0

         FPLS1= (FPLS2 - FPLS21)/2.
         FPLS3= (FPLS2 - FPLS23)
C-----------------------------------------------------------------------------
C --- SUM TERMS:   (Standard F2 is defined with an overall XBJ factor)

          FTEMP1= FDEL1/XMEASURE*Pdfz0 + FREG1*Pdfz/Z + FPLS1
          FTEMP2=(FDEL2/XMEASURE*Pdfz0 + FREG2*Pdfz/Z + FPLS2) * XBJ
          FTEMP3= FDEL3/XMEASURE*Pdfz0 + FREG3*Pdfz/Z + FPLS3

        IF    (INDEX.EQ. 1)  THEN
          FTEMP=  FTEMP1 * ( GLQ*GLQ + GRQ*GRQ )
        ELSEIF(INDEX.EQ. 2)  THEN
          FTEMP=  FTEMP2 * ( GLQ*GLQ + GRQ*GRQ )
        ELSEIF(INDEX.EQ. 3)  THEN
          FTEMP=  FTEMP3 * ( GLQ*GLQ - GRQ*GRQ )
        ELSE
          WRITE(6,*) ' BAD INDEX ',INDEX
          STOP
        ENDIF

        FTEMP= FTEMP * ALPHAS/(2.0*PI) * (4./3.)
        FTEMP= FTEMP *2.0   !*** TO MATCH OUR NORMALIZATION
```

Figure E9: Program which computes each diagram; pages 13-16.



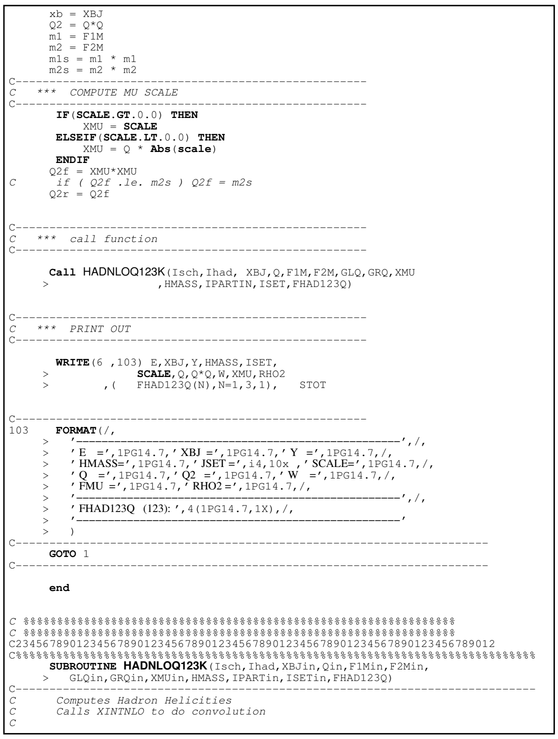
```
        XINTNLOQ1230= FTEMP

        RETURN
        END

C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C23456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
       SUBROUTINE HADSUBQ123(Isch,IhadIn,IPARTONin,XBJin,Q,F1Min,F2Min,
     >   GLQ,GRQ,XMUin,ISETin,HMASS,XHADSUBQ123)
C-----------------------------------------------------------------------------
C      Computes SUB Hadron Helicities
C
C
C
C
C
C-----------------------------------------------------------------------------
        Implicit Double Precision (A-H, O-Z)
        Dimension OmGLO(-1:1),OmGLO123(3),XHADSUBQ123(3)
        EXTERNAL XINTSUBQ123
        PARAMETER (PI=3.14159265359)
        Common  / ActInt /  AERR, RERR, iActL, iActU
        Common /CINTSUBQ/ F1M,XBJ, XMU, XiTH, iSet, IPARTON,Ihad
        DELTA(A,B,C) = SQRT(A**2 + B**2 + C**2 - 2.0*(A*B+B*C+C*A))

        XHADSUBQ123(1) = 0.0
        XHADSUBQ123(2) = 0.0
        XHADSUBQ123(3) = 0.0

        IF(XMUin.LE.F1Min) RETURN

        F1M    =F1Min
        F2M    =F2Min
        XBJ    =XBJin
        XMU    =XMUin
        iSet   =ISETin
        IPARTON=IPARTONin
        Ihad   =Ihadin

        Q2=Q**2

        Discr = 1.0 + (4.0*HMASS**2*XBJ**2)/Q**2
        ETA   = 2.0* XBJ/(1.0 + Sqrt(Discr))

        SHATTH= F2M**2
        XIMIN= xbj   !*** debugging patch (Remove later)
        XIMIN= ETA*((F1M+F2M)**2+Q**2)/Q**2
        XIMIN=ETA*(Q2-F1M**2+SHATTH +
     >           DELTA(-Q2,F1M**2,SHATTH))/(2.0*Q2)
        XIMAX= 1.0

        xith=XIMIN

       ALPHAS = AlQCD(XMU,iset)
       IF(ALPHAS.GT.0.5) ALPIX=0.5
       ComFct = ALPHAS/(2.0*PI)

       XIMAX=1.0
       XIMIN=XITH

C----------------------------------------------------------------------
        Conv = 0.0
        IF(XITH.LT.XIMAX)
```

```
     >    Conv = AdzInt (XINTSUBQ123, XIMIN, XIMAX,
     >                  AERR, RERR, ErrEst, iErr, iActL, iActU)

        Conv = Conv * ComFct
C----------------------------------------------------------------------

        CALL  PARLO (Q,F1M,F2M,GLQ,GRQ,OmGLO)

        XFAC= 1./(1.+(F2M/Q)**2)

        OmGLO123(1)= ( OmGLO(+1) + OmGLO(-1) )/2.
        OmGLO123(2)= ( OmGLO(+1) + OmGLO(-1) + 2.0*OmGLO(0) )*XFAC
        OmGLO123(3)=  -OmGLO(+1) + OmGLO(-1)
C----------------------------------------------------------------------

      RHO=1.0
      IF(HMASS.GT.0.0) THEN
        XNU=Q**2/(2.0*HMASS*XBJ)
        RHO=SQRT(1.0+(Q/XNU)**2)
      ENDIF


        XHADSUBQ123(1) = OmGLO123(1) * Conv
        XHADSUBQ123(2) = OmGLO123(2) * Conv * XBJ    !*** Def of F2
        XHADSUBQ123(3) = OmGLO123(3) * Conv


        RETURN
        END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      Function XINTSUBQ123(Xin)
C-----------------------------------------------------------------------------
C The splitting function for Q->Q
C-----------------------------------------------------------------------------
      Implicit Double Precision (A-H, O-Z)
      DATA SMALL /1.E-14/
        Common /CINTSUBQ/
     >   F1Min,XBJin, XMUIN, XiTHIN, iSetIN, IPARTONin,IhadIn

      F1M=F1Min
      XBJ=XBJin
      XMU=XMUin
      XiTH=XiTHin
      iSet=iSetin
      IPARTON=IPARTONin
      Ihad=IhadIN

      X=Xin
      IF(X.GE.1.0-SMALL) X=1.0-SMALL

       Z =X
      Z0=XiTH

C-----------------------------------------------------------------------------
C --- PDF's

      iHadr=Ihad          !*** 2/28/05 FIO Fixed
      QrkPdf = Pdf(iSet,iHadr,IPARTON,XiTH/X, XMU, iRet)
      QrkPdf0= Pdf(iSet,iHadr,IPARTON,XiTH  , XMU, iRet)

C-----------------------------------------------------------------------------
C --- DELTA FUNCTION TERMS:
```

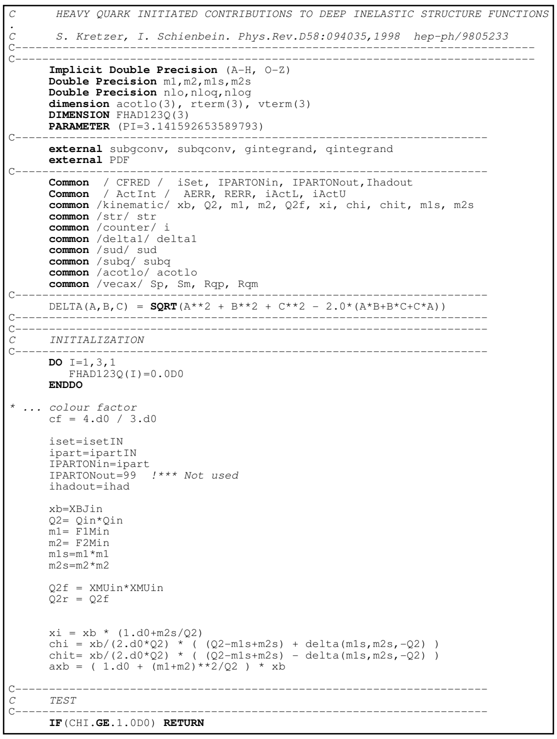
```
        XLOG1= LOG( (F1M/XMU)**2 )
C ***MODIFIED     \/
         delSUB= (+1.)*(  2. - 3./2. *XLOG1  - 2.*XLOG1*Log(1. - z0)
     >          - 2.*Log(1. - z0)
     >          - 2.*Log(1. - z0)**2
     >           )

         XMEASURE= (1.-XiTH)   !*** INTEGRATION MEASURE

C-----------------------------------------------------------------------------
C --- PLUS FUNCTION TERMS:

        regSUB=
     > (
     >  (-XLOG1 -1.)*( (  (1.+ z**2)*QrkPdf/X
     >                  -(1.+1.**2)*QrkPdf0/1.0) * 1./(1.-z) )
     >  +      (-2.)*( (  (1.+ z**2)*QrkPdf/X
     >                 - (1.+1.**2)*QrkPdf0/1.0) *
     >                                     Log(1.-z)/(1.-z) )  !*** modified
     > )


      TOTAL = regSUB + delSUB/XMEASURE* QrkPdf0 /1.0

      TOTAL = (4./3.) * TOTAL   !*** COLOR FACTOR

      XINTSUBQ123 = TOTAL

      Return
      End

C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C23456789012345678901234567890123456789012345678901234567890123456789012
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

C234567890123456789012345678901234567890123456789012345678901234567890123456789
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
       SUBROUTINE KRETZER1()
C      PROGRAM KRETZER1
C-----------------------------------------------------------------------------
C      Program to COMPUTE DIS HQ PRODUCTION
C
C
C
C
C      11/05/99 FIO. MODIFY FRONT END
C      01/01/99 FIO. CORE CODE BY STEFAN KRETZER
C      S. Kretzer, I. Schienbein. Phys.Rev.D58:094035,1998  hep-ph/9805233
C-----------------------------------------------------------------------------
      Implicit Double Precision (A-H, O-Z)
      Double Precision m1,m2,m1s,m2s
      dimension acotlo(3), rterm(3), vterm(3)
      DIMENSION FHAD123Q(3)
      DIMENSION XMARRAY(6)
      PARAMETER (PI=3.141592653589793)
C-----------------------------------------------------------------------------
C--- ONLY COMMON BLOCK USED
      Common  / ActInt /  AERR, RERR, iActL, iActU
C-----------------------------------------------------------------------------
C--- FOR NEUTRAL CURRENT
C      DATA SBIG,XBJ,Q,F1M,F2M,GLQ,GRQ,GLLEP,GRLEP,XMU,ISET,HMASS
C    >   /  98596.,0.1, 10.,1.6,1.6,0.5,0.5,0.5,0.5,10,1,0.938/
C      DATA   XLEPOL /2.0/
C      DATA IPARTIN, IPARTOUT,SCALE  /  4, 4,  -1  /
```

```
C-----------------------------------------------------------------------------
C--- FOR CHARGED CURRENT
cv try TM
cv      DATA SBIG,XBJ,Q,F1M,F2M,GLQ,GRQ,GLLEP,GRLEP,XMU,ISET,HMASS
cv     >   /  98596.,0.1, 10.,0.5,1.6,1.0,0.0,1.0,0.0,10.0,1,0.938/
      DATA SBIG,XBJ,Q,F1M,F2M,GLQ,GRQ,GLLEP,GRLEP,XMU,ISET,HMASS
cv     >   /  98596.,0.1, 10.,0.5,1.6,1.0,0.0,1.0,0.0,10.0,1,0.0/
     >   /  98596.,0.1, 10.,0.5,1.4,1.0,0.0,1.0,0.0,10.0,1,0.0/
      DATA   XLEPOL /1.0/
      DATA IPARTIN, IPARTOUT,SCALE  /  3, 4,  -1  /
C-----------------------------------------------------------------------------
C-----------------------------------------------------------------------------
cv       DATA XMARRAY  /  0.1, 0.1, 0.2, 1.6, 5.0, 170.0/
       DATA XMARRAY  /  0.1, 0.1, 0.2, 1.4, 4.3, 170.0/
       DATA  AERR, RERR, iActL, iActU    /  0.0, 0.001,1,1/
C----------------------------------------------------------------------
C----------------------------------------------------------------------
C SETUP INTEGRATION PARAMETERS
C----------------------------------------------------------------------
C     RERR=1.E-8   !*** Setting this too small will yield unstable results ***
      WRITE(6,202)  AERR, RERR, iActL, iActU
202   FORMAT(' AERR= ',1PG14.7,' RERR= ',1PG14.7,
     >       ' IACTL= ',I3,' IACTU= ',I3)
      WRITE(6,*)  ' ENTER:  AERR, RERR, iActL, iActU'
      READ (5,*)            AERR, RERR, iActL, iActU
C----------------------------------------------------------------------
C SETUP PARTON NUMBERS
C----------------------------------------------------------------------
      WRITE(6,203) IPARTIN, IPARTOUT
203   FORMAT(' IPARTIN= ',I3,' IPARTOUT= ',I3)
      WRITE(6,*)  ' ENTER: IPARTIN, IPARTOUT '
      READ (5,*)           IPARTIN, IPARTOUT

       F1M=XMARRAY(ABS(IPARTIN))
       F2M=XMARRAY(ABS(IPARTOUT))

       IF(IPARTIN.EQ.IPARTOUT) THEN
         GLQ=0.5
         GRQ=0.5
         GLLEP=0.5
         GRLEP=0.5
         XLEPOL=2.0
       ENDIF

       WRITE(6,*)    GLQ,GRQ,GLLEP,GRLEP,SBIG,XLEPOL
       WRITE(6,*)  '> GLQ,GRQ,GLLEP,GRLEP,SBIG,XLEPOL'
       READ( 5,*)    GLQ,GRQ,GLLEP,GRLEP,SBIG,XLEPOL

       E=(SBIG-HMASS**2)/(2.0*HMASS)

C----------------------------------------------------------------------
C SET THE LOOP
C----------------------------------------------------------------------
1     WRITE(6,102) XBJ,Q, E,F1M,F2M,ISET,HMASS,SCALE
102   FORMAT('  XBJ= ',1PG14.7,'   Q= ',1PG14.7,/,
     >       '    E= ',1PG14.7,'  F1M= ',1PG14.7,
     >       '  F2M= ',1PG14.7,' ISET= ',I3, /,
     >       ' HMASS= ',1PG14.7,' SCALE= ',1PG14.7)
      WRITE(6,*)  ' ENTER:  XBJ,Q, E,F1M,F2M,ISET,HMASS,SCALE '
      READ (5,*)            XBJ,Q, E,F1M,F2M,ISET,HMASS,SCALE
C----------------------------------------------------------------------
C SET THE SCALES
C----------------------------------------------------------------------
* ... kinematics:
```

Figure E10: Program which computes each diagram; pages 17-20.

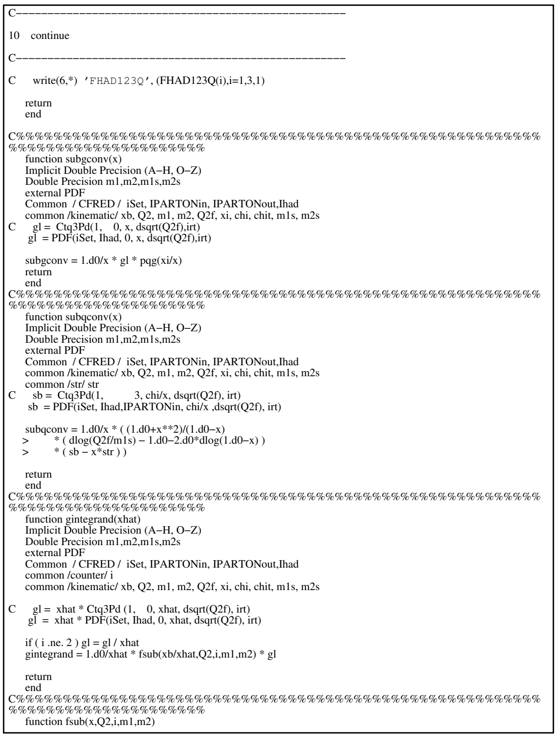


```
      xb = XBJ
      Q2 = Q*Q
      m1 = F1M
      m2 = F2M
      m1s = m1 * m1
      m2s = m2 * m2
C-----------------------------------------------------
C   ***  COMPUTE MU SCALE
C-----------------------------------------------------
       IF(SCALE.GT.0.0) THEN
           XMU = SCALE
       ELSEIF(SCALE.LT.0.0) THEN
           XMU = Q * Abs(scale)
       ENDIF
      Q2f = XMU*XMU
C      if ( Q2f .le. m2s ) Q2f = m2s
      Q2r = Q2f


C-----------------------------------------------------
C   ***  call function
C-----------------------------------------------------

      Call HADNLOQ123K(Isch,Ihad, XBJ,Q,F1M,F2M,GLQ,GRQ,XMU
     >                 ,HMASS,IPARTIN,ISET,FHAD123Q)


C-----------------------------------------------------
C   ***  PRINT OUT
C-----------------------------------------------------

       WRITE(6 ,103) E,XBJ,Y,HMASS,ISET,
     >             SCALE,Q,Q*Q,W,XMU,RHO2
     >        ,(   FHAD123Q(N),N=1,3,1),   STOT


C-----------------------------------------------------
103    FORMAT(/,
     >   '-----------------------------------------------',/,
     >   ' E  =',1PG14.7,' XBJ =',1PG14.7,' Y  =',1PG14.7,/,
     >   ' HMASS=',1PG14.7,' JSET =',i4,10x ,' SCALE=',1PG14.7,/,
     >   ' Q  =',1PG14.7,' Q2  =',1PG14.7,' W  =',1PG14.7,/,
     >   ' FMU =',1PG14.7,' RHO2 =',1PG14.7,/,
     >   '-----------------------------------------------',/,
     >   ' FHAD123Q  (123): ',4(1PG14.7,1X),/,
     >   '-----------------------------------------------'
     >   )
C-----------------------------------------------------------------------
      GOTO 1
C-----------------------------------------------------------------------

      end


C %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C2345678901234567890123456789012345678901234567890123456789012345678901
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      SUBROUTINE HADNLOQ123K(Isch,Ihad,XBJin,Qin,F1Min,F2Min,
     >   GLQin,GRQin,XMUin,HMASS,IPARTin,ISETin,FHAD123Q)
C--------------------------------------------------------------------------
C      Computes Hadron Helicities
C      Calls XINTNLO to do convolution
C
```



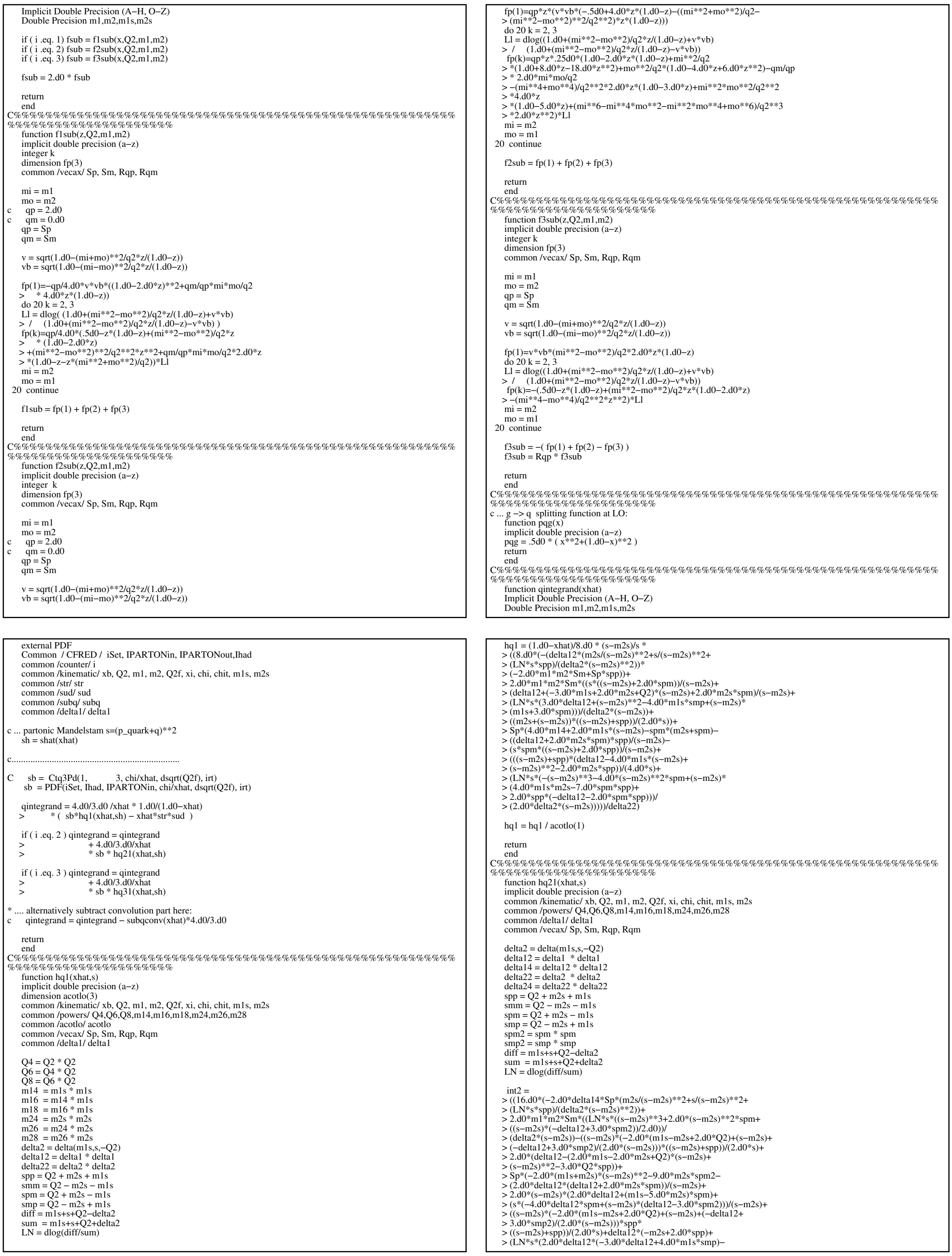
```
C      HEAVY QUARK INITIATED CONTRIBUTIONS TO DEEP INELASTIC STRUCTURE FUNCTIONS
.
C      S. Kretzer, I. Schienbein. Phys.Rev.D58:094035,1998  hep-ph/9805233
C--------------------------------------------------------------------------
C--------------------------------------------------------------------------
      Implicit Double Precision (A-H, O-Z)
      Double Precision m1,m2,m1s,m2s
      Double Precision nlo,nloq,nlog
      dimension acotlo(3), rterm(3), vterm(3)
      DIMENSION FHAD123Q(3)
      PARAMETER (PI=3.141592653589793)
C---------------------------------------------------------------------
      external subgconv, subqconv, gintegrand, qintegrand
      external PDF
C---------------------------------------------------------------------
      Common  / CFRED /  iSet, IPARTONin, IPARTONout,Ihadout
      Common  / ActInt /  AERR, RERR, iActL, iActU
      common /kinematic/ xb, Q2, m1, m2, Q2f, xi, chi, chit, m1s, m2s
      common /str/ str
      common /counter/ i
      common /delta1/ delta1
      common /sud/ sud
      common /subq/ subq
      common /acotlo/ acotlo
      common /vecax/ Sp, Sm, Rqp, Rqm
C---------------------------------------------------------------------
      DELTA(A,B,C) = SQRT(A**2 + B**2 + C**2 - 2.0*(A*B+B*C+C*A))
C---------------------------------------------------------------------
C---------------------------------------------------------------------
C     INITIALIZATION
C---------------------------------------------------------------------
      DO I=1,3,1
         FHAD123Q(I)=0.0D0
      ENDDO

* ... colour factor
      cf = 4.d0 / 3.d0

      iset=isetIN
      ipart=ipartIN
      IPARTONin=ipart
      IPARTONout=99  !*** Not used
      ihadout=ihad

      xb=XBJin
      Q2= Qin*Qin
      m1= F1Min
      m2= F2Min
      m1s=m1*m1
      m2s=m2*m2

      Q2f = XMUin*XMUin
      Q2r = Q2f


      xi = xb * (1.d0+m2s/Q2)
      chi = xb/(2.d0*Q2) * ( (Q2-m1s+m2s) + delta(m1s,m2s,-Q2) )
      chit= xb/(2.d0*Q2) * ( (Q2-m1s+m2s) - delta(m1s,m2s,-Q2) )
      axb = ( 1.d0 + (m1+m2)**2/Q2 ) * xb

C---------------------------------------------------------------------
C     TEST
C---------------------------------------------------------------------
      IF(CHI.GE.1.0D0) RETURN
```

```


*... vector and axial couplings:  General CURRENT PROCESS:
      v1 = GLQin + GRQin
      v2 = GLQin + GRQin
      a1 = GLQin - GRQin
      a2 = GLQin - GRQin

      call couplings(a1,a2,v1,v2,Sp,Sm,Rqp,Rqm)

* ... LO amplitudes including kinematic mass effects:
      acotlo(1) = ( Sp*(Q2+m1s+m2s) - 2.d0*Sm*m1*m2 )
     >          / 2.d0 / delta(m1s,m2s,-Q2)
      acotlo(2) = Sp * xb * delta(m1s,m2s,-Q2)/Q2
      acotlo(3) = 2.d0 * Rqp

C      write(6,*) '(acotlo(i),i=1,3,1),xi,chi,chit,axb',
C     >   (acotlo(i),i=1,3,1),xi,chi,chit,axb


* ... real soft and virtual contributions from NLO quark graphs
      call svnlo(rterm,vterm,Sp,Sm,Rqp,Rqm)

C      str = Ctq3Pd(iset, ipart, chi, dsqrt(Q2f),irt)
       str = PDF(iset, Ihad, ipart, chi, dsqrt(Q2f),irt)

C       write(6,*) str,chi, ipart,iset, dsqrt(Q2f)

C-----------------------------------------------------
* ... for structure function F_i
      do 10 i = 1, 3

      sud = fq(i)

* ... NLO contributions: gluon AND quark initiated
C      nlog = dinteg(gintegrand,axb,1.d0,1.d-3)
       nlog = 0.0
C **** NEED PROTECT THE UPPER ENDPOINT: iActU= 1 OR 2, NOT 0
C      nloq = dinteg(qintegrand,chi,1.d0,1.d-4)
       nloq= 0.0d0
       if(chi.le.1.d0) then
        nloq = AdzInt(qintegrand,chi,1.d0,
     >              AERR, RERR, ErrEst, iErr, iActL, iActU)
          else
          Write(6,*)  " error: integration limits >1 in HADNLOQ123K"
       endif

* ... sum of quark/gluon convolutions and soft/virtual terms
      nlo = nlog + acotlo(i) * ( nloq
     >    + cf * (rterm(i)+vterm(i)) * str
* ... including the Sudakov log from the 1/(1-xi)+ distibution
     >    + cf * sud * str * dlog(1.d0-chi) )

      nloq = nlo - nlog

comment: F_i's normalized to +acotlo(i)*s(chi)+O(alpha_s) for all i
    alphas= alQCD(Sqrt(Q2r),iset)
    nloq  = nloq  * alphas/2.d0/PI

    FHAD123Q(i)=  nloq

C---------------------------------------------------
C *** ADJUST NORMALIZATION CONVENTIONS TO MATCH OURS:
    IF(I.EQ.2)  FHAD123Q(i) =  FHAD123Q(i)/(1.0+M2S/Q2)
```



```
C---------------------------------------------------

10   continue

C---------------------------------------------------

C     write(6,*)  'FHAD123Q', (FHAD123Q(i),i=1,3,1)

    return
    end

C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
    function subgconv(x)
    Implicit Double Precision (A–H, O–Z)
    Double Precision m1,m2,m1s,m2s
    external PDF
    Common  / CFRED /  iSet, IPARTONin, IPARTONout,Ihad
    common /kinematic/ xb, Q2, m1, m2, Q2f, xi, chi, chit, m1s, m2s
C     gl =  Ctq3Pd(1,    0, x, dsqrt(Q2f),irt)
     gl  = PDF(iSet, Ihad, 0, x, dsqrt(Q2f),irt)

    subgconv = 1.d0/x * gl * pqg(xi/x)
    return
    end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
    function subqconv(x)
    Implicit Double Precision (A–H, O–Z)
    Double Precision m1,m2,m1s,m2s
    external PDF
    Common  / CFRED /  iSet, IPARTONin, IPARTONout,Ihad
    common /kinematic/ xb, Q2, m1, m2, Q2f, xi, chi, chit, m1s, m2s
    common /str/ str
C     sb =  Ctq3Pd(1,           3, chi/x, dsqrt(Q2f), irt)
     sb  = PDF(iSet, Ihad,IPARTONin, chi/x ,dsqrt(Q2f), irt)

    subqconv = 1.d0/x * ( (1.d0+x**2)/(1.d0–x)
   >       * ( dlog(Q2f/m1s) – 1.d0–2.d0*dlog(1.d0–x) )
   >       * ( sb – x*str ) )

    return
    end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
    function gintegrand(xhat)
    Implicit Double Precision (A–H, O–Z)
    Double Precision m1,m2,m1s,m2s
    external PDF
    Common  / CFRED /  iSet, IPARTONin, IPARTONout,Ihad
    common /counter/ i
    common /kinematic/ xb, Q2, m1, m2, Q2f, xi, chi, chit, m1s, m2s

C     gl =  xhat * Ctq3Pd (1,    0, xhat, dsqrt(Q2f), irt)
     gl  =  xhat * PDF(iSet, Ihad, 0, xhat, dsqrt(Q2f), irt)

    if ( i .ne. 2 ) gl = gl / xhat
    gintegrand = 1.d0/xhat * fsub(xb/xhat,Q2,i,m1,m2) * gl

    return
    end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
    function fsub(x,Q2,i,m1,m2)
```

Figure E11: Program which computes each diagram; pages 21-24.



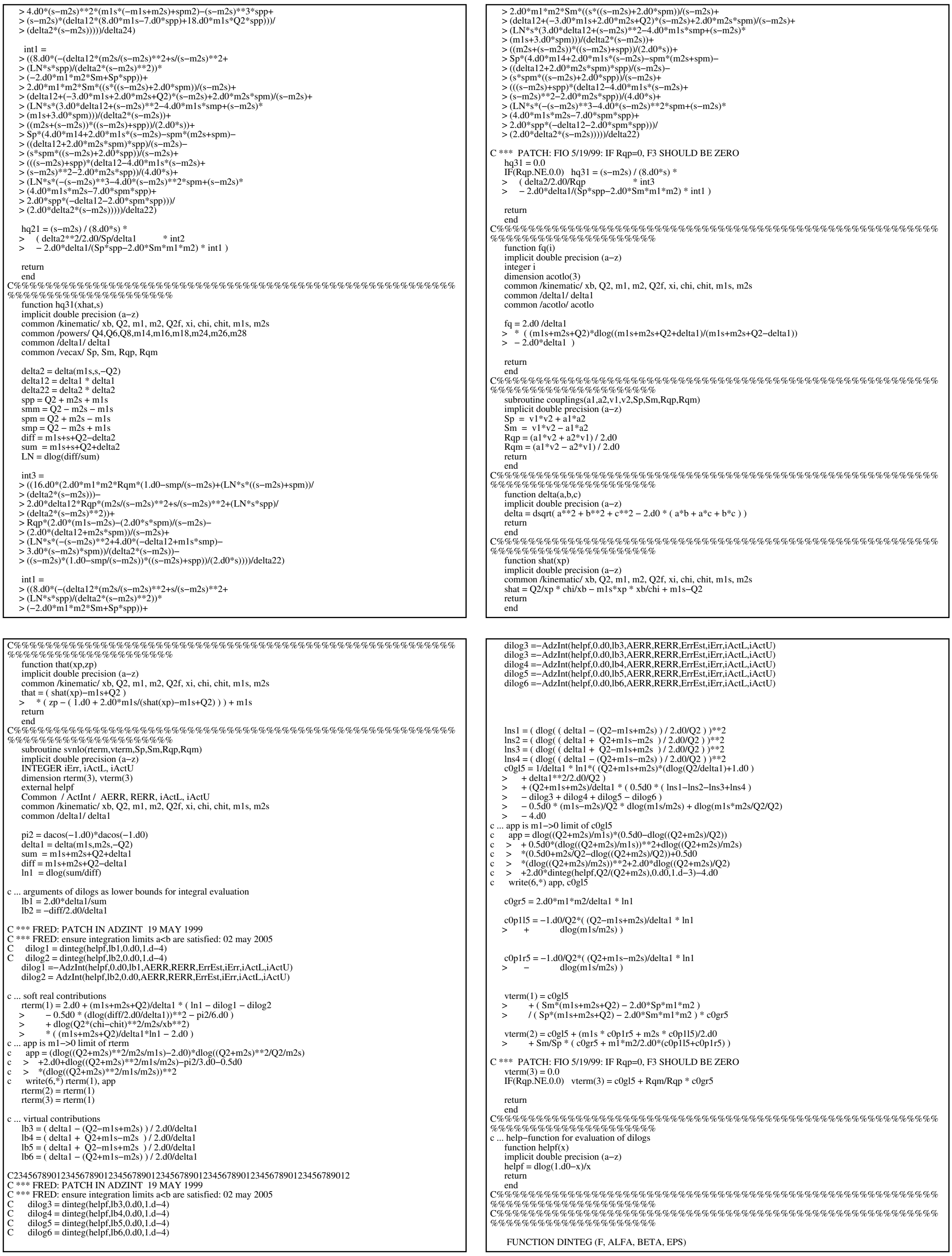
```
      Implicit Double Precision (A−H, O−Z)
      Double Precision m1,m2,m1s,m2s

      if ( i .eq. 1) fsub = f1sub(x,Q2,m1,m2)
      if ( i .eq. 2) fsub = f2sub(x,Q2,m1,m2)
      if ( i .eq. 3) fsub = f3sub(x,Q2,m1,m2)

      fsub = 2.d0 * fsub

      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
      function f1sub(z,Q2,m1,m2)
      implicit double precision (a−z)
      integer k
      dimension fp(3)
      common /vecax/ Sp, Sm, Rqp, Rqm

      mi = m1
      mo = m2
c      qp = 2.d0
c      qm = 0.d0
      qp = Sp
      qm = Sm

      v = sqrt(1.d0−(mi+mo)**2/q2*z/(1.d0−z))
      vb = sqrt(1.d0−(mi−mo)**2/q2*z/(1.d0−z))

      fp(1)=−qp/4.d0*v*vb*((1.d0−2.d0*z)**2+qm/qp*mi*mo/q2
     >     * 4.d0*z*(1.d0−z))
      do 20 k = 2, 3
      Ll = dlog( (1.d0+(mi**2−mo**2)/q2*z/(1.d0−z)+v*vb)
     > /     (1.d0+(mi**2−mo**2)/q2*z/(1.d0−z)−v*vb) )
      fp(k)=qp/4.d0*(.5d0−z*(1.d0−z)+(mi**2−mo**2)/q2*z
     >     * (1.d0−2.d0*z)
     > +(mi**2−mo**2)**2/q2**2*z**2+qm/qp*mi*mo/q2*2.d0*z
     > *(1.d0−z−z*(mi**2+mo**2)/q2))*Ll
      mi = m2
      mo = m1
 20  continue

      f1sub = fp(1) + fp(2) + fp(3)

      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
      function f2sub(z,Q2,m1,m2)
      implicit double precision (a−z)
      integer  k
      dimension fp(3)
      common /vecax/ Sp, Sm, Rqp, Rqm

      mi = m1
      mo = m2
c      qp = 2.d0
c      qm = 0.d0
      qp = Sp
      qm = Sm

      v = sqrt(1.d0−(mi+mo)**2/q2*z/(1.d0−z))
      vb = sqrt(1.d0−(mi−mo)**2/q2*z/(1.d0−z))
```

```
      fp(1)=qp*z*(v*vb*(−.5d0+4.d0*z*(1.d0−z)−((mi**2+mo**2)/q2−
     > (mi**2−mo**2)**2/q2**2)*z*(1.d0−z)))
      do 20 k = 2, 3
      Ll = dlog((1.d0+(mi**2−mo**2)/q2*z/(1.d0−z)+v*vb)
     > /     (1.d0+(mi**2−mo**2)/q2*z/(1.d0−z)−v*vb))
       fp(k)=qp*z*.25d0*(1.d0−2.d0*z*(1.d0−z)+mi**2/q2
     > *(1.d0+8.d0*z−18.d0*z**2)+mo**2/q2*(1.d0−4.d0*z+6.d0*z**2)−qm/qp
     > * 2.d0*mi*mo/q2
     > −(mi**4+mo**4)/q2**2*2.d0*z*(1.d0−3.d0*z)+mi**2*mo**2/q2**2
     > *4.d0*z
     > *(1.d0−5.d0*z)+(mi**6−mi**4*mo**2−mi**2*mo**4+mo**6)/q2**3
     > *2.d0*z**2)*Ll
      mi = m2
      mo = m1
 20  continue

      f2sub = fp(1) + fp(2) + fp(3)

      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
      function f3sub(z,Q2,m1,m2)
      implicit double precision (a−z)
      integer k
      dimension fp(3)
      common /vecax/ Sp, Sm, Rqp, Rqm

      mi = m1
      mo = m2
      qp = Sp
      qm = Sm

      v = sqrt(1.d0−(mi+mo)**2/q2*z/(1.d0−z))
      vb = sqrt(1.d0−(mi−mo)**2/q2*z/(1.d0−z))

      fp(1)=v*vb*(mi**2−mo**2)/q2*2.d0*z*(1.d0−z)
      do 20 k = 2, 3
      Ll = dlog((1.d0+(mi**2−mo**2)/q2*z/(1.d0−z)+v*vb)
     > /     (1.d0+(mi**2−mo**2)/q2*z/(1.d0−z)−v*vb))
       fp(k)=−(.5d0−z*(1.d0−z)+(mi**2−mo**2)/q2*z*(1.d0−2.d0*z)
     > −(mi**4−mo**4)/q2**2*z**2)*Ll
      mi = m2
      mo = m1
 20  continue

      f3sub = −( fp(1) + fp(2) − fp(3) )
      f3sub = Rqp * f3sub

      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
c ... g −> q  splitting function at LO:
      function pqg(x)
      implicit double precision (a−z)
      pqg = .5d0 * ( x**2+(1.d0−x)**2 )
      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
      function qintegrand(xhat)
      Implicit Double Precision (A−H, O−Z)
      Double Precision m1,m2,m1s,m2s
```

```
      external PDF
      Common  / CFRED /  iSet, IPARTONin, IPARTONout,Ihad
      common /counter/ i
      common /kinematic/ xb, Q2, m1, m2, Q2f, xi, chi, chit, m1s, m2s
      common /str/ str
      common /sud/ sud
      common /subq/ subq
      common /delta1/ delta1

c ... partonic Mandelstam s=(p_quark+q)**2
      sh = shat(xhat)

c...................................................................

C      sb =  Ctq3Pd(1,           3, chi/xhat, dsqrt(Q2f), irt)
       sb  = PDF(iSet, Ihad, IPARTONin, chi/xhat, dsqrt(Q2f), irt)

      qintegrand = 4.d0/3.d0 /xhat * 1.d0/(1.d0−xhat)
     >          * (  sb*hq1(xhat,sh) − xhat*str*sud  )

      if ( i .eq. 2 ) qintegrand = qintegrand
     >                      + 4.d0/3.d0/xhat
     >                      * sb * hq21(xhat,sh)

      if ( i .eq. 3 ) qintegrand = qintegrand
     >                      + 4.d0/3.d0/xhat
     >                      * sb * hq31(xhat,sh)

* .... alternatively subtract convolution part here:
c      qintegrand = qintegrand − subqconv(xhat)*4.d0/3.d0

      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
      function hq1(xhat,s)
      implicit double precision (a−z)
      dimension acotlo(3)
      common /kinematic/ xb, Q2, m1, m2, Q2f, xi, chi, chit, m1s, m2s
      common /powers/ Q4,Q6,Q8,m14,m16,m18,m24,m26,m28
      common /acotlo/ acotlo
      common /vecax/ Sp, Sm, Rqp, Rqm
      common /delta1/ delta1

      Q4 = Q2 * Q2
      Q6 = Q4 * Q2
      Q8 = Q6 * Q2
      m14  = m1s * m1s
      m16  = m14 * m1s
      m18  = m16 * m1s
      m24  = m2s * m2s
      m26  = m24 * m2s
      m28  = m26 * m2s
      delta2 = delta(m1s,s,−Q2)
      delta12 = delta1 * delta1
      delta22 = delta2 * delta2
      spp = Q2 + m2s + m1s
      smm = Q2 − m2s − m1s
      spm = Q2 + m2s − m1s
      smp = Q2 − m2s + m1s
      diff = m1s+s+Q2−delta2
      sum  = m1s+s+Q2+delta2
      LN = dlog(diff/sum)
```

```
      hq1 = (1.d0−xhat)/8.d0 * (s−m2s)/s *
     > ((8.d0*(−(delta12*(m2s/(s−m2s)**2+s/(s−m2s)**2+
     > (LN*s*spp)/(delta2*(s−m2s)**2))*
     > (−2.d0*m1*m2*Sm+Sp*spp))+
     > 2.d0*m1*m2*Sm*((s*((s−m2s)+2.d0*spm))/(s−m2s)+
     > (delta12+(−3.d0*m1s+2.d0*m2s+Q2)*(s−m2s)+2.d0*m2s*spm)/(s−m2s)+
     > (LN*s*(3.d0*delta12+(s−m2s)**2−4.d0*m1s*smp+(s−m2s)*
     > (m1s+3.d0*spm)))/(delta2*(s−m2s))+
     > ((m2s+(s−m2s))*((s−m2s)+spp))/(2.d0*s))+
     > Sp*(4.d0*m14+2.d0*m1s*(s−m2s)−spm*(m2s+spm)−
     > ((delta12+2.d0*m2s*spm)*spp)/(s−m2s)−
     > (s*spm*((s−m2s)+2.d0*spp))/(s−m2s)+
     > (((s−m2s)+spp)*(delta12−4.d0*m1s*(s−m2s)+
     > (s−m2s)**2−2.d0*m2s*spp))/(4.d0*s)+
     > (LN*s*(−(s−m2s)**3−4.d0*(s−m2s)**2*spm+(s−m2s)*
     > (4.d0*m1s*m2s−7.d0*spm*spp)+
     > 2.d0*spp*(−delta12−2.d0*spm*spp)))/
     > (2.d0*delta2*(s−m2s))))))/delta22)

      hq1 = hq1 / acotlo(1)

      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
      function hq21(xhat,s)
      implicit double precision (a−z)
      common /kinematic/ xb, Q2, m1, m2, Q2f, xi, chi, chit, m1s, m2s
      common /powers/ Q4,Q6,Q8,m14,m16,m18,m24,m26,m28
      common /delta1/ delta1
      common /vecax/ Sp, Sm, Rqp, Rqm

      delta2 = delta(m1s,s,−Q2)
      delta12 = delta1  * delta1
      delta14 = delta12 * delta12
      delta22 = delta2  * delta2
      delta24 = delta22 * delta22
      spp = Q2 + m2s + m1s
      smm = Q2 − m2s − m1s
      spm = Q2 + m2s − m1s
      smp = Q2 − m2s + m1s
      spm2 = spm * spm
      smp2 = smp * smp
      diff = m1s+s+Q2−delta2
      sum  = m1s+s+Q2+delta2
      LN = dlog(diff/sum)

       int2 =
     > ((16.d0*(−2.d0*delta14*Sp*(m2s/(s−m2s)**2+s/(s−m2s)**2+
     > (LN*s*spp)/(delta2*(s−m2s)**2))+
     > 2.d0*m1*m2*Sm*((LN*s*((s−m2s)**3+2.d0*(s−m2s)**2*spm+
     > ((s−m2s)*(−delta12+3.d0*spm2))/2.d0))/
     > (delta2*(s−m2s))−((s−m2s)*(−2.d0*(m1s−m2s+2.d0*Q2)+(s−m2s)+
     > (−delta12+3.d0*smp2)/(2.d0*(s−m2s)))*((s−m2s)+spp))/(2.d0*s)+
     > 2.d0*(delta12−(2.d0*m1s−2.d0*m2s+Q2)*(s−m2s)+
     > (s−m2s)**2−3.d0*Q2*spp))+
     > Sp*(−2.d0*(m1s+m2s)*(s−m2s)**2−9.d0*m2s*spm2−
     > (2.d0*delta12*(delta12+2.d0*m2s*spm))/(s−m2s)+
     > 2.d0*(s−m2s)*(2.d0*delta12+(m1s−5.d0*m2s)*spm)+
     > (s*(−4.d0*delta12*spm+(s−m2s)*(delta12−3.d0*spm2)))/(s−m2s)+
     > ((s−m2s)*(−2.d0*(m1s−m2s+2.d0*Q2)+(s−m2s)+(−delta12+
     > 3.d0*smp2)/(2.d0*(s−m2s)))*spp*
     > ((s−m2s)+spp))/(2.d0*s)+delta12*(−m2s+2.d0*spp)+
     > (LN*s*(2.d0*delta12*(−3.d0*delta12+4.d0*m1s*smp)−
```

Figure E12: Program which computes each diagram; pages 25-28.



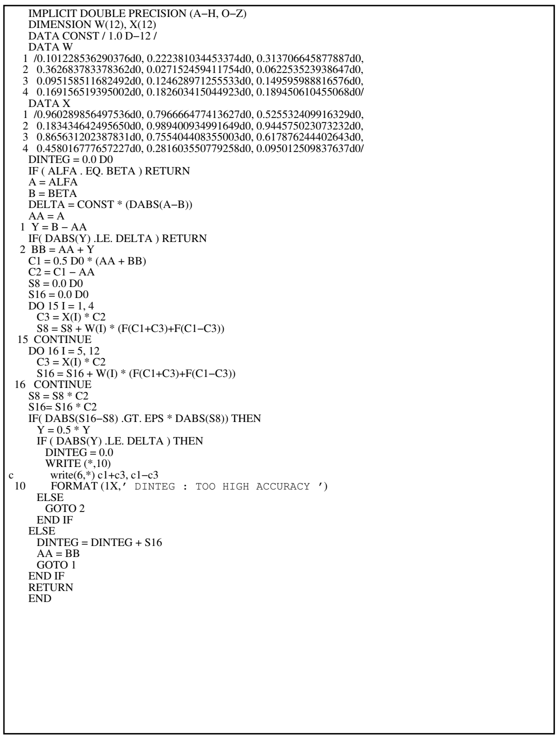
```
     > 4.d0*(s−m2s)**2*(m1s*(−m1s+m2s)+spm2)−(s−m2s)**3*spp+
     > (s−m2s)*(delta12*(8.d0*m1s−7.d0*spp)+18.d0*m1s*Q2*spp)))/
     > (delta2*(s−m2s)))))/delta24)

       int1 =
     > ((8.d0*(−(delta12*(m2s/(s−m2s)**2+s/(s−m2s)**2+
     > (LN*s*spp)/(delta2*(s−m2s)**2))*
     > (−2.d0*m1*m2*Sm+Sp*spp))+
     > 2.d0*m1*m2*Sm*((s*((s−m2s)+2.d0*spm))/(s−m2s)+
     > (delta12+(−3.d0*m1s+2.d0*m2s+Q2)*(s−m2s)+2.d0*m2s*spm)/(s−m2s)+
     > (LN*s*(3.d0*delta12+(s−m2s)**2−4.d0*m1s*smp+(s−m2s)*
     > (m1s+3.d0*spm)))/(delta2*(s−m2s))+
     > ((m2s+(s−m2s))*((s−m2s)+spp))/(2.d0*s))+
     > Sp*(4.d0*m14+2.d0*m1s*(s−m2s)−spm*(m2s+spm)−
     > ((delta12+2.d0*m2s*spm)*spp)/(s−m2s)−
     > (s*spm*((s−m2s)+2.d0*spp))/(s−m2s)+
     > (((s−m2s)+spp)*(delta12−4.d0*m1s*(s−m2s)+
     > (s−m2s)**2−2.d0*m2s*spp))/(4.d0*s)+
     > (LN*s*(−(s−m2s)**3−4.d0*(s−m2s)**2*spm+(s−m2s)*
     > (4.d0*m1s*m2s−7.d0*spm*spp)+
     > 2.d0*spp*(−delta12−2.d0*spm*spp)))/
     > (2.d0*delta2*(s−m2s)))))/delta22)

      hq21 = (s−m2s) / (8.d0*s) *
     >    ( delta2**2/2.d0/Sp/delta1           * int2
     >    − 2.d0*delta1/(Sp*spp−2.d0*Sm*m1*m2) * int1 )

      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
      function hq31(xhat,s)
      implicit double precision (a−z)
      common /kinematic/ xb, Q2, m1, m2, Q2f, xi, chi, chit, m1s, m2s
      common /powers/ Q4,Q6,Q8,m14,m16,m18,m24,m26,m28
      common /delta1/ delta1
      common /vecax/ Sp, Sm, Rqp, Rqm

      delta2 = delta(m1s,s,−Q2)
      delta12 = delta1 * delta1
      delta22 = delta2 * delta2
      spp = Q2 + m2s + m1s
      smm = Q2 − m2s − m1s
      spm = Q2 + m2s − m1s
      smp = Q2 − m2s + m1s
      diff = m1s+s+Q2−delta2
      sum  = m1s+s+Q2+delta2
      LN = dlog(diff/sum)

      int3 =
     > ((16.d0*(2.d0*m1*m2*Rqm*(1.d0−smp/(s−m2s)+(LN*s*((s−m2s)+spm))/
     > (delta2*(s−m2s)))−
     > 2.d0*delta12*Rqp*(m2s/(s−m2s)**2+s/(s−m2s)**2+(LN*s*spp)/
     > (delta2*(s−m2s)**2))+
     > Rqp*(2.d0*(m1s−m2s)−(2.d0*s*spm)/(s−m2s)−
     > (2.d0*(delta12+m2s*spm))/(s−m2s)+
     > (LN*s*(−(s−m2s)**2+4.d0*(−delta12+m1s*smp)−
     > 3.d0*(s−m2s)*spm))/(delta2*(s−m2s))−
     > ((s−m2s)*(1.d0−smp/(s−m2s))*((s−m2s)+spp))/(2.d0*s))))/delta22)

      int1 =
     > ((8.d0*(−(delta12*(m2s/(s−m2s)**2+s/(s−m2s)**2+
     > (LN*s*spp)/(delta2*(s−m2s)**2))*
     > (−2.d0*m1*m2*Sm+Sp*spp))+
```

```
     > 2.d0*m1*m2*Sm*((s*((s−m2s)+2.d0*spm))/(s−m2s)+
     > (delta12+(−3.d0*m1s+2.d0*m2s+Q2)*(s−m2s)+2.d0*m2s*spm)/(s−m2s)+
     > (LN*s*(3.d0*delta12+(s−m2s)**2−4.d0*m1s*smp+(s−m2s)*
     > (m1s+3.d0*spm)))/(delta2*(s−m2s))+
     > ((m2s+(s−m2s))*((s−m2s)+spp))/(2.d0*s))+
     > Sp*(4.d0*m14+2.d0*m1s*(s−m2s)−spm*(m2s+spm)−
     > ((delta12+2.d0*m2s*spm)*spp)/(s−m2s)−
     > (s*spm*((s−m2s)+2.d0*spp))/(s−m2s)+
     > (((s−m2s)+spp)*(delta12−4.d0*m1s*(s−m2s)+
     > (s−m2s)**2−2.d0*m2s*spp))/(4.d0*s)+
     > (LN*s*(−(s−m2s)**3−4.d0*(s−m2s)**2*spm+(s−m2s)*
     > (4.d0*m1s*m2s−7.d0*spm*spp)+
     > 2.d0*spp*(−delta12−2.d0*spm*spp)))/
     > (2.d0*delta2*(s−m2s)))))/delta22)

C ***  PATCH: FIO 5/19/99: IF Rqp=0, F3 SHOULD BE ZERO
      hq31 = 0.0
      IF(Rqp.NE.0.0)   hq31 = (s−m2s) / (8.d0*s) *
     >    ( delta2/2.d0/Rqp                    * int3
     >    − 2.d0*delta1/(Sp*spp−2.d0*Sm*m1*m2) * int1 )

      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
      function fq(i)
      implicit double precision (a−z)
      integer i
      dimension acotlo(3)
      common /kinematic/ xb, Q2, m1, m2, Q2f, xi, chi, chit, m1s, m2s
      common /delta1/ delta1
      common /acotlo/ acotlo

      fq = 2.d0 /delta1
     >  *  ( (m1s+m2s+Q2)*dlog((m1s+m2s+Q2+delta1)/(m1s+m2s+Q2−delta1))
     >  − 2.d0*delta1  )

      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
      subroutine couplings(a1,a2,v1,v2,Sp,Sm,Rqp,Rqm)
      implicit double precision (a−z)
      Sp  =  v1*v2 + a1*a2
      Sm  =  v1*v2 − a1*a2
      Rqp = (a1*v2 + a2*v1) / 2.d0
      Rqm = (a1*v2 − a2*v1) / 2.d0
      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
      function delta(a,b,c)
      implicit double precision (a−z)
      delta = dsqrt( a**2 + b**2 + c**2 − 2.d0 * ( a*b + a*c + b*c ) )
      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
      function shat(xp)
      implicit double precision (a−z)
      common /kinematic/ xb, Q2, m1, m2, Q2f, xi, chi, chit, m1s, m2s
      shat = Q2/xp * chi/xb − m1s*xp * xb/chi + m1s−Q2
      return
      end
```

```
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
      function that(xp,zp)
      implicit double precision (a−z)
      common /kinematic/ xb, Q2, m1, m2, Q2f, xi, chi, chit, m1s, m2s
      that = ( shat(xp)−m1s+Q2 )
     >    * ( zp − ( 1.d0 + 2.d0*m1s/(shat(xp)−m1s+Q2) ) ) + m1s
      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
      subroutine svnlo(rterm,vterm,Sp,Sm,Rqp,Rqm)
      implicit double precision (a−z)
      INTEGER iErr, iActL, iActU
      dimension rterm(3), vterm(3)
      external helpf
      Common  / ActInt /  AERR, RERR, iActL, iActU
      common /kinematic/ xb, Q2, m1, m2, Q2f, xi, chi, chit, m1s, m2s
      common /delta1/ delta1

      pi2 = dacos(−1.d0)*dacos(−1.d0)
      delta1 = delta(m1s,m2s,−Q2)
      sum  = m1s+m2s+Q2+delta1
      diff = m1s+m2s+Q2−delta1
      ln1  = dlog(sum/diff)

c ... arguments of dilogs as lower bounds for integral evaluation
      lb1 = 2.d0*delta1/sum
      lb2 = −diff/2.d0/delta1

C *** FRED: PATCH IN ADZINT  19 MAY 1999
C *** FRED: ensure integration limits a<b are satisfied: 02 may 2005
C     dilog1 = dinteg(helpf,lb1,0.d0,1.d−4)
C     dilog2 = dinteg(helpf,lb2,0.d0,1.d−4)
      dilog1 =−AdzInt(helpf,0.d0,lb1,AERR,RERR,ErrEst,iErr,iActL,iActU)
      dilog2 = AdzInt(helpf,lb2,0.d0,AERR,RERR,ErrEst,iErr,iActL,iActU)

c ... soft real contributions
      rterm(1) = 2.d0 + (m1s+m2s+Q2)/delta1 * ( ln1 − dilog1 − dilog2
     >        − 0.5d0 * (dlog(diff/2.d0/delta1))**2 − pi2/6.d0 )
     >        + dlog(Q2*(chi−chit)**2/m2s/xb**2)
     >        * ( (m1s+m2s+Q2)/delta1*ln1 − 2.d0 )
c ... app is m1−>0 limit of rterm
c     app = (dlog((Q2+m2s)**2/m2s/m1s)−2.d0)*dlog((Q2+m2s)**2/Q2/m2s)
c    >   +2.d0+dlog((Q2+m2s)**2/m1s/m2s)−pi2/3.d0−0.5d0
c    >   *(dlog((Q2+m2s)**2/m1s/m2s))**2
c     write(6,*) rterm(1), app
      rterm(2) = rterm(1)
      rterm(3) = rterm(1)

c ... virtual contributions
      lb3 = ( delta1 − (Q2−m1s+m2s) ) / 2.d0/delta1
      lb4 = ( delta1 +  Q2+m1s−m2s  ) / 2.d0/delta1
      lb5 = ( delta1 +  Q2−m1s+m2s  ) / 2.d0/delta1
      lb6 = ( delta1 − (Q2+m1s−m2s) ) / 2.d0/delta1

C23456789012345678901234567890123456789012345678901234567890123456789012
C *** FRED: PATCH IN ADZINT  19 MAY 1999
C *** FRED: ensure integration limits a<b are satisfied: 02 may 2005
C      dilog3 = dinteg(helpf,lb3,0.d0,1.d−4)
C      dilog4 = dinteg(helpf,lb4,0.d0,1.d−4)
C      dilog5 = dinteg(helpf,lb5,0.d0,1.d−4)
C      dilog6 = dinteg(helpf,lb6,0.d0,1.d−4)
```

```
      dilog3 =−AdzInt(helpf,0.d0,lb3,AERR,RERR,ErrEst,iErr,iActL,iActU)
      dilog3 =−AdzInt(helpf,0.d0,lb3,AERR,RERR,ErrEst,iErr,iActL,iActU)
      dilog4 =−AdzInt(helpf,0.d0,lb4,AERR,RERR,ErrEst,iErr,iActL,iActU)
      dilog5 =−AdzInt(helpf,0.d0,lb5,AERR,RERR,ErrEst,iErr,iActL,iActU)
      dilog6 =−AdzInt(helpf,0.d0,lb6,AERR,RERR,ErrEst,iErr,iActL,iActU)



      lns1 = ( dlog( ( delta1 − (Q2−m1s+m2s) ) / 2.d0/Q2 ) )**2
      lns2 = ( dlog( ( delta1 +  Q2+m1s−m2s  ) / 2.d0/Q2 ) )**2
      lns3 = ( dlog( ( delta1 +  Q2−m1s+m2s  ) / 2.d0/Q2 ) )**2
      lns4 = ( dlog( ( delta1 − (Q2+m1s−m2s) ) / 2.d0/Q2 ) )**2
      c0gl5 = 1/delta1 * ln1*( (Q2+m1s+m2s)*(dlog(Q2/delta1)+1.d0 )
     >     + delta1**2/2.d0/Q2 )
     >     + (Q2+m1s+m2s)/delta1 * ( 0.5d0 * ( lns1−lns2−lns3+lns4 )
     >     − dilog3 + dilog4 + dilog5 − dilog6 )
     >     − 0.5d0 * (m1s−m2s)/Q2 * dlog(m1s/m2s) + dlog(m1s*m2s/Q2/Q2)
     >     − 4.d0
c ... app is m1−>0 limit of c0gl5
c     app = dlog((Q2+m2s)/m1s)*(0.5d0−dlog((Q2+m2s)/Q2))
c    >   + 0.5d0*(dlog((Q2+m2s)/m1s))**2+dlog((Q2+m2s)/m2s)
c    >   *(0.5d0+m2s/Q2−dlog((Q2+m2s)/Q2))+0.5d0
c    >   *(dlog((Q2+m2s)/m2s))**2+2.d0*dlog((Q2+m2s)/Q2)
c    >   +2.d0*dinteg(helpf,Q2/(Q2+m2s),0.d0,1.d−3)−4.d0
c     write(6,*) app, c0gl5

      c0gr5 = 2.d0*m1*m2/delta1 * ln1

      c0p1l5 = −1.d0/Q2*( (Q2−m1s+m2s)/delta1 * ln1
     >     +           dlog(m1s/m2s) )


      c0p1r5 = −1.d0/Q2*( (Q2+m1s−m2s)/delta1 * ln1
     >     −           dlog(m1s/m2s) )


      vterm(1) = c0gl5
     >       + ( Sm*(m1s+m2s+Q2) − 2.d0*Sp*m1*m2 )
     >       / ( Sp*(m1s+m2s+Q2) − 2.d0*Sm*m1*m2 ) * c0gr5

      vterm(2) = c0gl5 + (m1s * c0p1r5 + m2s * c0p1l5)/2.d0
     >       + Sm/Sp * ( c0gr5 + m1*m2/2.d0*(c0p1l5+c0p1r5) )

C ***  PATCH: FIO 5/19/99: IF Rqp=0, F3 SHOULD BE ZERO
      vterm(3) = 0.0
      IF(Rqp.NE.0.0)   vterm(3) = c0gl5 + Rqm/Rqp * c0gr5

      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
c ... help−function for evaluation of dilogs
      function helpf(x)
      implicit double precision (a−z)
      helpf = dlog(1.d0−x)/x
      return
      end
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%

       FUNCTION DINTEG (F, ALFA, BETA, EPS)
```

Figure E13: Program which computes each diagram; pages 29-32.



```
      IMPLICIT DOUBLE PRECISION (A−H, O−Z)
      DIMENSION W(12), X(12)
      DATA CONST / 1.0 D−12 /
      DATA W
     1  /0.101228536290376d0, 0.222381034453374d0, 0.313706645877887d0,
     2   0.362683783378362d0, 0.027152459411754d0, 0.062253523938647d0,
     3   0.095158511682492d0, 0.124628971255533d0, 0.149595988816576d0,
     4   0.169156519395002d0, 0.182603415044923d0, 0.189450610455068d0/
      DATA X
     1  /0.960289856497536d0, 0.796666477413627d0, 0.525532409916329d0,
     2   0.183434642495650d0, 0.989400934991649d0, 0.944575023073232d0,
     3   0.865631202387831d0, 0.755404408355003d0, 0.617876244402643d0,
     4   0.458016777657227d0, 0.281603550779258d0, 0.095012509837637d0/
      DINTEG = 0.0 D0
      IF ( ALFA . EQ. BETA ) RETURN
      A = ALFA
      B = BETA
      DELTA = CONST * (DABS(A−B))
      AA = A
    1  Y = B − AA
      IF( DABS(Y) .LE. DELTA ) RETURN
    2  BB = AA + Y
      C1 = 0.5 D0 * (AA + BB)
      C2 = C1 − AA
      S8 = 0.0 D0
      S16 = 0.0 D0
      DO 15 I = 1, 4
        C3 = X(I) * C2
        S8 = S8 + W(I) * (F(C1+C3)+F(C1−C3))
   15  CONTINUE
      DO 16 I = 5, 12
        C3 = X(I) * C2
        S16 = S16 + W(I) * (F(C1+C3)+F(C1−C3))
  16   CONTINUE
      S8 = S8 * C2
      S16= S16 * C2
      IF( DABS(S16−S8) .GT. EPS * DABS(S8)) THEN
        Y = 0.5 * Y
        IF ( DABS(Y) .LE. DELTA ) THEN
          DINTEG = 0.0
          WRITE (*,10)
c           write(6,*) c1+c3, c1−c3
  10        FORMAT (1X,' DINTEG : TOO HIGH ACCURACY ')
        ELSE
          GOTO 2
        END IF
      ELSE
        DINTEG = DINTEG + S16
        AA = BB
        GOTO 1
      END IF
      RETURN
      END
```

Figure E14: Program which computes each diagram; page 33.

APPENDIX F

FIT RESULTS

F.1 CORRELATION TABLES

F.1.1 Non-Singlet Fit

Table F1 shows the correlations among fit parameters from non-singlet fit results using $ACOT$ model. Table F2 shows the correlations among fit parameters from non-singlet fit results using $\overline{MS}$ model.

Table F1: The correlation matrix for non-singlet fit. $ACOT$ model.

Parameter	Λ_{QCD}	$A1_{u_v}$	$A2_{u_v}$	$norm$
Λ_{QCD}	1.000	-0.285	-0.594	-0.777
$A1_{u_v}$	-0.285	1.000	0.813	0.728
$A2_{u_v}$	-0.594	0.813	1.000	0.717
$norm$	-0.777	0.728	0.717	1.000

Table F2: The correlation matrix for non-singlet fit. $\overline{MS}$ model.

Parameter	Λ_{QCD}	$A1_{u_v}$	$A2_{u_v}$	$norm$
Λ_{QCD}	1.000	-0.283	-0.596	-0.779
$A1_{u_v}$	-0.283	1.000	0.812	0.725
$A2_{u_v}$	-0.596	0.812	1.000	0.717
$norm$	-0.779	0.725	0.717	1.000

F.1.2 Combined Fit

Table F3 shows the correlations among fit parameters from combined fit results using *ACOT* model. Table F4 shows the correlations among fit parameters from combined fit results using $\overline{MS}$ model. All experimental errors taken into account.

Table F3: The correlation matrix for combined fit. *ACOT* model.

Parameter	Λ_{QCD}	$A1_{u_v}$	$A2_{u_v}$	$A0_{ud}$	$A2_{ud}$	$A2_g$	$norm$
Λ_{QCD}	1.000	-0.037	-0.293	-0.238	-0.008	-0.348	-0.436
$A1_{u_v}$	-0.037	1.000	0.888	0.774	0.584	-0.148	0.807
$A2_{u_v}$	-0.293	0.888	1.000	0.629	0.536	-0.049	0.709
$A0_{ud}$	-0.238	0.774	0.629	1.000	0.680	0.242	0.807
$A2_{ud}$	-0.008	0.584	0.536	0.680	1.000	0.122	0.283
$A2_g$	-0.348	-0.148	-0.049	0.242	0.122	1.000	0.016
$norm$	-0.436	0.807	0.709	0.807	0.283	0.016	1.000

Table F4: The correlation matrix for combined fit. $\overline{MS}$ model.

Parameter	Λ_{QCD}	$A1_{u_v}$	$A2_{u_v}$	$A0_{ud}$	$A2_{ud}$	$A2_g$	$norm$
Λ_{QCD}	1.000	0.114	-0.215	-0.449	-0.072	-0.652	-0.426
$A1_{u_v}$	0.114	1.000	0.868	0.573	0.574	-0.246	0.746
$A2_{u_v}$	-0.215	0.868	1.000	0.568	0.563	-0.004	0.699
$A0_{ud}$	-0.449	0.573	0.568	1.000	0.650	0.517	0.797
$A2_{ud}$	-0.072	0.574	0.563	0.650	1.000	0.132	0.337
$A2_g$	-0.652	-0.246	-0.004	0.517	0.132	1.000	0.161
$norm$	-0.426	0.746	0.699	0.797	0.337	0.161	1.000

F.2 NUCLEAR PDFS FROM FIT TO NUTEV DATA

Figures F1, F2 show the shape of the PDF distributions as result of our QCD fits ($ACOT$ and $\overline{MS}$ models).

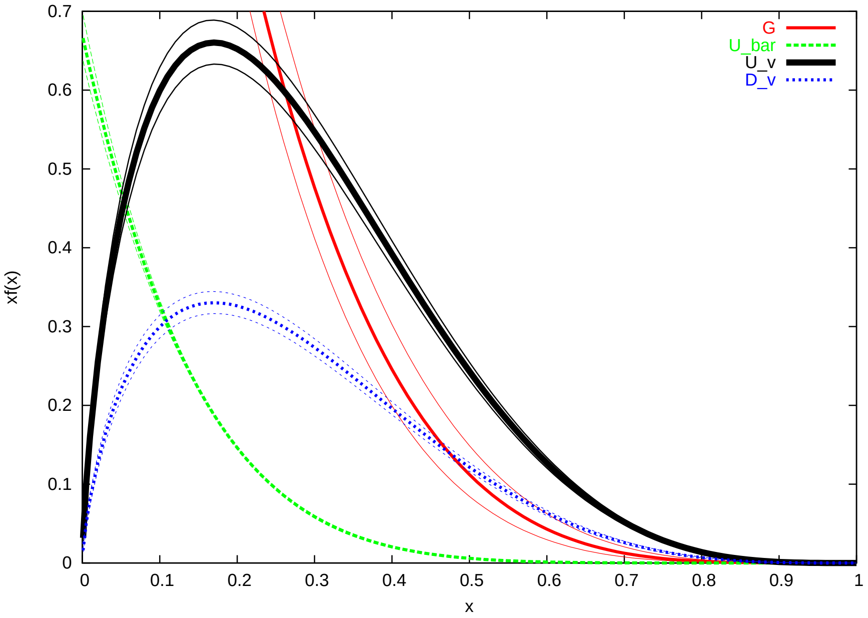


Figure F1: Nuclear PDFs at $Q_0^2 = 5$ GeV^2. ACOT scheme.

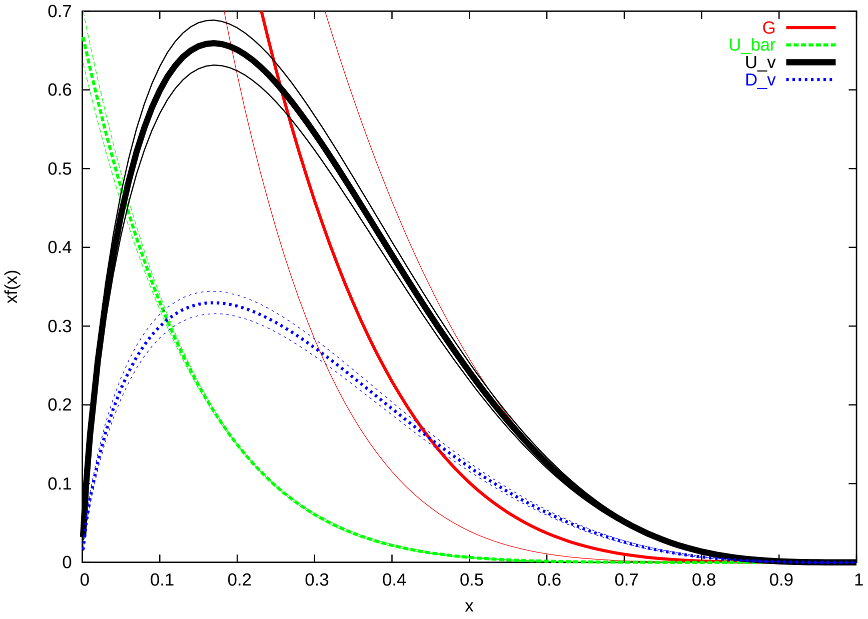


Figure F2: Nuclear PDFs at $Q_0^2 = 5$ GeV^2. $\overline{MS}$ scheme.

APPENDIX G

TESTS OF THE QCD PROGRAM

To ensure consistency of the QCD program checks were performed.

G.1 TEST OF MINIMIZATION USING CCFR DATA

A way to test the minimization of the NuTeV analysis is by comparing the quoted CCFR results in Ref. [50] to the results produced using CCFR data and NuTeV QCD program. The CCFR data used in this study is taken from Durham database, which correspond to the tables used in [50]:

- CCFR xF_3: Phys. Rev. Lett. 79: 1213 (1997) for the Non-Singlet Fit;
- CCFR F_2 and xF_3: Nevis Report 292 (1996) (Bill Seligman Ph.D Thesis).

The output is presented in the table G1. For testing purposes only statistical errors were used. Also, we check the results by fixing the fit parameters to the final CCFR fit results as quoted from Ref. [50] (see Table G2).

We have used the following cuts to be consistent with Ref. [50]:

- $x < 0.7,\ Q^2 > 5\ \ \mathrm{GeV}^2$, $m_b = 4.5$ GeV,$m_c = 1.5$ GeV.

Table G1: Test of minimization using CCFR data

	xF_3 fit		xF_3 and F_2 fit	
	CCFR fitter	NuTeV fitter	CCFR fitter	NuTeV fitter
$\Lambda^{(n_f=4)}$ (MeV)	387±42	364±43	381± 24	392 ± 24
$A_{1u_v} \equiv \eta_1$	0.801±0.010	0.807±0.010	0.804±0.008	0.81 ± 0.01
$A_{2u_v} \equiv \eta_2$	3.89±0.03	3.90± 0.04	3.91±0.02	3.89 ± 0.02
$A_{ou_v} + A_{0d_v} \equiv A_{NS}$	8.57±0.20	6.25+3.12	8.65±0.16	3.06+6.13
$2A_{0u_d} \equiv A_S$			1.52±0.02	$2 \times (0.81 \pm 0.01)$
$A_{2u_d} \equiv \eta_S$			7.82±0.09	7.68± 0.01
$A_{0g} \equiv A_G$			1.94±0.24	2.13
$A_{2g} \equiv \eta_G$			4.44±0.65	4.47 ± 0.70
χ^2/dof	81.4/82	87.56/82	190.6/164	199/165

Table G2: Fit results using CCFR data with parameters fixed to the values quoted from Ref. [50]

	xF_3 fit		xF_3 and F_2 fit	
	CCFR fitter	NuTeV fitter	CCFR fitter	NuTeV fitter
$\Lambda^{(n_f=4)}$ (MeV)	387±42	398±8	381± 24	378 ± 5

Conclusions:

- The results agree within errors.
- It is claimed in Ref. [50] that the QCD constraints would shift the results up by about 15 MeV.

G.1.1 CCFR vs NuTeV

For a more consistent way in comparing the CCFR results to the NuTeV one should compare the CCFR data (unfortunately only the F_2 has been extracted) from:

- UnKi Yang F_2 data: Phys.Rev.Lett.86:2742-2745 (2001) [63]

The extraction of this data set is performed in a similar fashion as the method used by NuTeV.

- compare results using F_2 from UnKi Yang (CCFR) to F_2 from NuTeV. The errors are added in quadrature. See Table G3.

Table G3: Comparison of fit results using CCFR data [63].

	F_2 only	
	UnKi data	NuTeV data
$\Lambda^{(n_f=4)}$ (MeV)	330±64	425 ± 71
$A_{1u_v} \equiv \eta_1$	0.804±0.047	0.762 ± 0.004
$A_{2u_v} \equiv \eta_2$	4.02±0.08	3.67 ± 0.08
$A_{ou_v} + A_{0d_v} \equiv A_{NS}$	6.22+3.11	5.26+2.63
$2A_{0u_d} \equiv A_S$	$2 \times (0.73 \pm 0.07)$	$2 \times (0.72 \pm 0.06)$
$A_{2u_d} \equiv \eta_S$	9.24 ±0.69	8.20 ± 0.68
$A_{0g} \equiv A_G$	2.78	2.58
$A_{2g} \equiv \eta_G$	5.18 ± 1.86	5.00 ± 1.86
χ^2/dof	0.52	0.86

- known differences: CCFR (UnKi) data is from a 2-parameter fit, NuTeV data is from 1-parameter fit. The difference between 1p and 2p fits are known to matter only in the lowest x bins, thus we compare the results for $x > 0.1$ as well (see Table G4).

Table G4: Comparison between fit results using CCFR (UnKi) and NuTeV data for $x > 0.1$

	F_2 only	
	UnKi data	NuTeV data
$\Lambda^{(n_f=4)}$ (MeV)	322±64	454 ± 103
$A_{1u_v} \equiv \eta_1$	0.743±0.086	0.665 ± 0.123
$A_{2u_v} \equiv \eta_2$	3.96±0.09	3.41 ± 0.35
$A_{ou_v} + A_{0d_v} \equiv A_{NS}$	5.22+2.61	3.85+1.92
$2A_{0u_d} \equiv A_S$	$2 \times (0.66 \pm 0.10)$	$2 \times (0.55 \pm 0.10)$
$A_{2u_d} \equiv \eta_S$	9.87 ±2.19	5.33 ± 2.99
$A_{0g} \equiv A_G$	3.50	2.34
$A_{2g} \equiv \eta_G$	6.07 ± 2.68	4.30± 2.98
χ^2/dof	0.55	0.84

- We notice $< 2\sigma$ discrepancy between NuTeV and CCFR Singlet results. This is purely due to the differences in the data between CCFR and NuTeV.

G.2 TEST OF THE FITTER WITH CORRELATIONS

This is easily checked by setting all the correlations to zero and expect to reproduce the uncorrelated results. This has been successfully tested both for non-singlet and combined fits. The comparisons gave identical results as it was expected and thus the check of the correlation has been completed.

G.3 TEST OF THE FUSION

G.3.1 Fake data

1. generate a fake data file with structure functions calculated with a known set of PDFs (i.e. CTEQ4HQ);
2. insert the fake data file in the QCD fit (i.e. use it as real data);
3. compare the output with the input. (Λ_{QCD} quoted by the CTEQ4HQ group, obtained using the same code to calculate structure functions).

The global fits for the CTEQ4HQ, at 4 flavors, in the ACOT were performed for $0.01 < x < 0.7$, $Q2 > 4$ (see Table G5). The NS fit results of the fake data study: The fake data study closes within 1%. Thus, it validates the fitter and the success of the fusion of the packages.

Table G5: Fake Data study using *ACOT* scheme.

pdf	$\Lambda^{(n_f=4)}$ (MeV)
CTEQ4HQ	298
Fake data	301

G.3.2 Comparison between Massless $\overline{MS}$ Models

The very nice feature of this QCD program is its access to different theory models. Therefore, it provides direct comparisons among various calculations. For example, one can compare calculation of J. Owens of the structure functions based on the massless $\overline{MS}$ scheme [53] to calculation of F. Olness using the massless $\overline{MS}$ [54]. The comparisons between these two calculations are shown in Table G6. The calculation of the structure functions agree within 1% providing another good test of the fusion of QCD programs as well as a check of the massless prediction viewed from two different perspectives.

Table G6: Comparisons between two $\overline{MS}$ models for the NS and Combined fits.

Program basis	xF_3 fit $\Lambda^{(n_f=4)}$ (MeV)	xF_3 and F_2 fit $\Lambda^{(n_f=4)}$ (MeV)
Fred Olness	476 ± 60	439 ± 58
Jeff Owens	476 ± 60	433 ± 51

G.3.3 ACOT Technicalities

The factorization theorem is conveniently expressed as a composition of t-channel 2PI amplitudes,

$$F = \sum_{N=1}^{\infty} C(K)^N T + \text{Non factorizable contributions,} \tag{G.1}$$

where C represents a graph for a hard scattering (hard scattering Coefficients), K a graph for a rung (Kernel), T represents a graph that couples to the target (Target). The idea of *ACOT* scheme is to reorganize the infinite sum by introducing a collinear projector operator Z which basically connects the hard scattering terms to the PDFs. Hence, the main point to generalize the factorization theorem to heavy quarks masses is to construct properly this

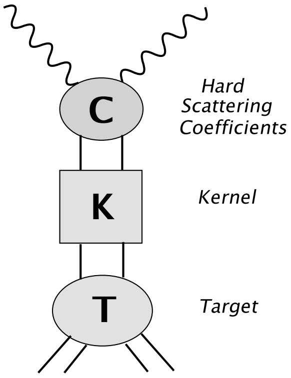


Figure G1: Decomposition of the hadronic structure function in terms of 2PI amplitudes projector operator (full details could be found in [8]).

$$F \approx \underbrace{C\left[\frac{1}{1-(1-Z)K}\right]}_{\text{hard scatt. coeff.(all orders)}} \quad Z\underbrace{\left[\frac{1}{1-K}T\right]}_{\text{pdf's}} + \underbrace{O\left(\frac{\Lambda^2_{QCD}}{Q^2}\right)}_{\text{Non-factorizable contributions}} \quad . \tag{G.2}$$

This relation is valid to all-orders.

G.4 ACOT AND ITS VARIANT

- **FULL** ***ACOT*** **scheme:** Full *ACOT* scheme is designed to work for quarks with finite masses, in particular it is used for the case when only heavy quarks are involved. Figure G2 displays each of the contributing diagrams.

$$HAD\_LO - HAD\_SUB + HAD\_NLO\_Kr - HAD\_SUB\_Q + HAD\_NLO.$$

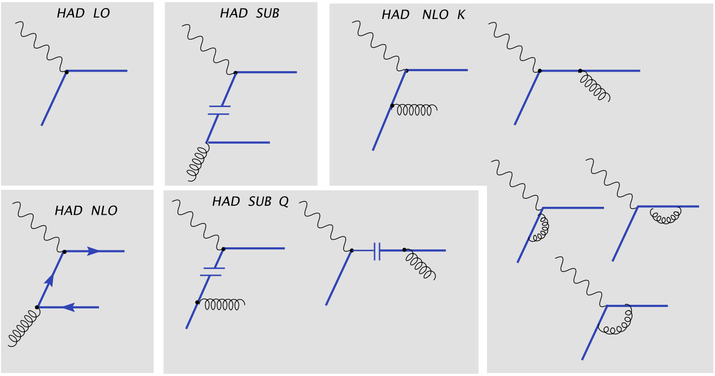


Figure G2: Full *ACOT* scheme with $m_c \neq 0$

- **Shortcut** *ACOT***:** This variant was designed for the structure functions with both light and heavy quarks contributions. Figure G3 displays each of the contributing diagrams. It is numerically more stable when dealing with the light quark parts because proper cancellations then occur.

$$HAD\_LO - HAD\_SUB + HAD\_NLO\_Q(M_H = 0) + HAD\_NLO.$$

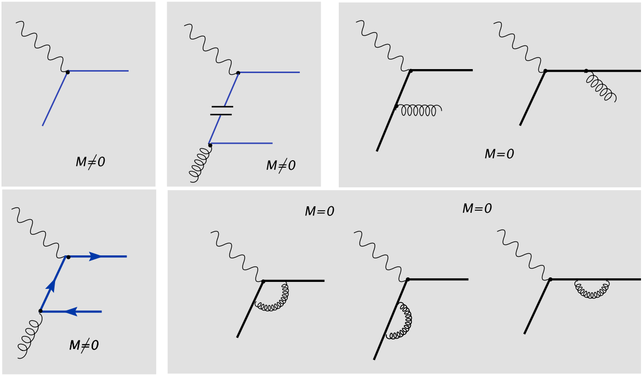


Figure G3: Shortcut ACOT scheme.

BIBLIOGRAPHY

[1] SLAC-MIT collaboration, G. Miller *et al., Phys. Rev.* **D5** 528 (1972).

[2] J. D. Bjorken, *Phys. Rev.* **179,** 1547 (1969).

[3] C. G. Callan and D. Gross, *Phys. Rev. Lett.* **22,** 156 (1969).

[4] *Particle Data Group Listing* (2005).

[5] D. E. Soper, J. C. Collins and G. Sterman, *Factorization of Hard Processes in QCD,* Singapore: World Scientific (1989).

[6] G. Dissertori, I. Knowles and M. Schmelling, *"Quantum Chromodynamics, High Energy Experiments and Theory"* (2003).

[7] W. A. Bardeen, A. J. Buras, D. W. Duke, T. Muta, *Phys. Rev.* **D18** 3998 (1978).

[8] J. C. Collins, *Phys. Rev.* **D58**, 094002 (1998).

[9] Heavy Quark Production and PDF subgroup report; hep-ph/0005112

[10] M. Gluck, E. Hoffmann, and E. Reya, *Z. phys.* **C13**, 119 (1982);
M. Gluck, E. Hoffmann, and E. Reya, *Z. phys.* **C38**, 441 (1988); erratum: ibid., **C39**, 590 (1988); E. Laenen, *Nucl. Phys.* **B392,** 162 (1993)

[11] J. C. Collins, *Phys. Rev.* **D58**, 094002 (1998);
M. A. .G. Aivazis, F. .I. Olness, and W. K. Tung,*Phys. Rev. Lett.***65,**2339 (1990); M. A. G. Aivazis, J. C. Collins, F. I. Olness and W. K. Tung, *Phys. Rev.* **D50**,3102 (1994);
B. .A. Kniehl, M. Kramer, G. Kramer, and M. Spira, *Phys. Lett.* **B356**,539 (1995);
M. .Buza, Y. Matiounine, J. Smith, and W. L. van Neerven, *Eur. Phys. J.* **C1**,301 (1998);
R. S. Thorne and R. G .Roberts, *Phys. Rev* **D57**,6871 (1998);
R. S. Thorne and R. G .Roberts, *Phys. Lett* **B421**,303 (1998);

[12] M. A. G. Aivazis, J. C. Collins, F. I. Olness and W. K. Tung, *Phys. Rev.* **D50**, 3085, 3102 (1994).

[13] Yu. L. Dokshitzer, *Sov.Phys.JETP***46** (1977);
V. N. Gribov and L. N. Lipatov, *Sov.J.Nucl.Phys* **15,** 438 (1972);
L. N. Lipatov, *Sov.J.Nucl.Phys* **20,** 95 (1975);
G. Altarelli and G. Parisi *Nuc.Phys* **B126,** 298 (1977).

[14] F. I. Olness, R. J. Scalise, *Phys. Rev.* **D57(1)** (1998).

[15] H. Georgi and H. D. Politzer, *Phys. Rev.* **D14** 1829 (1976).

[16] S. Kretzer, H. Reno, *Phys. Rev.* **D69** , 034002 (2002).

[17] R. G. Arnold *et al., Phys. Rev. Lett.* **52,** 727 (1984);
J. Gomez *et al., Phys.Rev* **D49,** 4348 (1994).

[18] G. Amaudruz *et al., Nucl. Phys.* **B441,** 3 (1995).

[19] M. .R. Adams *et al., Zeit. Phys* **C67,** 403 (1995).

[20] S. A. Kulagin, R. Petti [hep-ph/0412425].

[21] J. W. Qiu and I. Vitev, *Phys. Lett.* **B587,** 52 (2004).

[22] M. Tzanov *et al.* (NuTeV Collaboration), *Phys. Rev.* **D74** 012008 (2006);
M. Tzanov, Ph.D. Thesis, University of Pittsburgh (2005).

[23] G. P. Zeller, Ph. D. Thesis, Northwestern University (2002).

[24] A. G.Vaitaitis, Ph. D. Thesis, Columbia University, Nevis 298 (2000).

[25] S. Avvakumov, Ph. D. Thesis, University of Rochester (2003).

[26] D. Mason, Ph. D. Thesis, University of Oregon (2005).

[27] J. A. Formaggio, Ph. D. Thesis, Columbia University (2001).

[28] D. A. Harris *et al.* (NuTeV Collaboration), *Nucl. Instrum. Methods* **A 447***,* 377 (2000).

[29] C. W. Fabjan and T. Ludlam, *Ann. Rev. Nucl. Part. Sci.* **32,** 335 (1982).

[30] ANSYS, Finite element analysis software [http://www.ansys.com].

[31] R. H. Bernstein *et al., FERMILAB-TM-1884,* (1994);

J. Yu *et al., FERMILAB-TM-2040* (1998).

[32] L. F. Landau, *J.Phys.* USSR 8:2001 (1994).

[33] J. Conrad, M. H. Shaevitz, and T. Bolton, *Rev. Mod. Phys.* **70** 4 (1998).

[34] G. Sterman *et al., Rev. Mod. Phys.* **67** 157 (1995).

[35] T. A. Gabriel and P. E. Groom, *Nucl. Instr. Meth.* **A338**, 336 (1994).

[36] GEANT Detector Description and Simulation Tool, CERN Program Library Long Writeup W5013.

[37] G. Ingelman, A. Esin, J. Rathsman, *Computer Physics Communications* 101, 108-134 (1997).

[38] A. J. Buras and K. J. F. Gemers, *Nucl. Phys.* **B132,**249 (1978).

[39] S. Rabinowitz *et al.* (CCFR collaboration), *Phys. Rev. Lett.* **70** , 134 (1993).

[40] A. Hawker, *et al. Phys. Rev. Lett.* **80,** 3715 (1998).

[41] M. Gluck *et al.* (GRV collaboration),*Z. Phys.* **C67** 433 (1995).

[42] A. Bodek and U. K. Yang, *Nucl. Phys. Proc. Suppl.* **112** 70 (2002).

[43] L. W. Whitlow *et al., Phys. Lett.* **B282** 475 (1992).

[44] A. C. Benvenuti *et al., Phys. Lett.* **B223** 485 (1989);
A. C. Benvenuti *et al., Phys. Lett.* **B236** 592 (1989).

[45] M. Arneodo *et al., Nucl. Phys.* **B483** 3 (1997).

[46] L. W. Whitlow *et al., Phys. Lett.* **B250,** 193 (1990).

[47] D. Y. Bardin and V. A. Dokuchaeva, *JINR*-E2-86-260 (1986).

[48] P. Villain *et al., Eur. Phys. J.* C11 19 (1999);
P. Astier *et al., Phys. Lett.* **B486** 35 (2000).

[49] R. M. Barnet, *Phys. Rev* **D14,**70 (1976).

[50] W. Seligman, Ph. D. Thesis, Columbia University, Nevis 292 (1997).

[51] W. Seligman *et al., Phys. Rev. Lett.* **79,**1213 (1997);

[52] J. F. Owens, Wu-Ki Tung, *Ann. Rev. Nucl. Part. Sci.* **42:**291-332 (1992).

[53] J. F. Owens, private communication;
A. Devoto, D. W. Duke, J. F. Owens, R. G. Roberts, *Phys. Rev.* **D27**508 (1983).

[54] F. I. Olness, private communication;
M. A. G. Aivazis, J. C. Collins, F. I. Olness and W. K. Tung, *Phys. Rev.* **D50***,* 3085, 3102 (1994).

[55] Numerical Recipes in Fortran 77 [http://library.lanl.gov/numerical/bookfpdf.html].

[56] P. R. Bevington, D. Keith Robinson, *“ Data reduction and error analysis for the physical sciences”* (1992).

[57] M. Virchaux and A. Milsztajn, *Phys. Lett.* **B 274,** 221 (1992).

[58] S. Alekhin, *JHEP* **0302,** 015 (2003).

[59] A. D. Martin, R. G. Roberts, W. J. Stirling, and R. S.Thorne, *Phys. Lett.,* **B604,** 61 (2004).

[60] J. Kim *et al.*(CCFR collaboration),*Phys. Rev. Lett.* **81,** 3595 (1998).

[61] J. Blumlein and H. Bottcher, *Nucl. Phys.* **B636,** 225 (2002).

[62] P. L. Anthony *et al., Phys. Lett.* **B463,** 339 (1999);
P. L. Anthony *et al., Phys. Lett.* **B493,** 19 (2000).

[63] U. K. Yang , Ph. D. Thesis, University of Rochester, UR-1583 (2001).

[64] D. Buskulic *et al.* (ALEPH collaboration), *Phys. Lett.* **B357,** 487 (1995); *ibid.,* erratum *Phys. Lett.* **B364,** 247 (1995);
R. Akers *et al.* (OPAL collaboration), *Z.Phys.* **C68,** 203 (1995);
P. Abreu *et al.* (DELPHI collaboration), *Phys. Lett* **B398,** 194 (1997).

[65] T. Klugg (H1 collaboration) at DIS2005, Madison, Wisconsin (2005);
C. Adloff *et al., Eur. Phys. J.* **C19,** 289 (2001);
ZEUS Collaboration: DESY-05-019 [hep-ex/0502007].

[66] J. Kutti, V. F. Weisskopf, *Phys. Rev.* **D4,** 3418 (1971);
P. V. Landshoff and J. C. Polkinghorne, *Phys. Letters* **34B,** 621 (1971).

[67] A. C. Benvenuti *et al., Phys. Lett.* **B237,** 592 (1990).

[68] L. W. Whitlow, Ph. D. Thesis, SLAC-REPORT-357 (1990).

[69] R. S. Thorne and R. G. Roberts, Phys. Lett. **B 421** 303 (1998);
A. D. Martin *et al.* Eur. Phys. J. **C 18** 117 (2000).

[70] F.Halzen, A.D. Martin *"Quarks and Leptons: An Introductory Course in Modern Particle Physics".*

[71] Ellis, Stirling and Webber, *" QCD and Collider Physics"* (1996).