Multi-Beam Strategy for Neutrino-Nucleus Cross-Sections

BY

Rainer W. Schiel

Submitted to the graduate degree program in Physics
and the Graduate Faculty of the University of Kansas
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy.

Committee:

Dr. John P. Ralston, Chairperson

Dr. Hume A. Feldman

Dr. David Lerner

Dr. Danny Marfatia

Dr. Douglas W. McKay

Date defended: *July 15, 2008*

The Dissertation Committee for Rainer W. Schiel certifies
that this is the approved version of the following dissertation:

Multi-Beam Strategy for Neutrino-Nucleus Cross-Sections

Committee:

Dr. John P. Ralston, Chairperson

Dr. Hume A. Feldman

Dr. David Lerner

Dr. Danny Marfatia

Dr. Douglas W. McKay

Date approved: *July 15, 2008*

Abstract

Neutrino oscillation experiments rely on the knowledge of neutrino-nucleus cross-sections. Generally, just one scattering process is used to model these cross-sections. However, it is not sufficient to describe the cross-sections by only one scattering process. In the region of momentum transfers $Q^2 \lessapprox 10^5$ MeV, there are two dominant processes, charged-current quasi-elastic scattering and charged-current whole-nucleus scattering. Both of these processes must be accounted for in the analysis.

Determining the neutrino cross-sections experimentally is difficult. In most experiments, only the scattering angle and energy of the charged lepton in the final state are known, although neither the recoiled target nor the energy of the incoming neutrino are measured.

The Multi-Beam Strategy presented in this dissertation is a novel data-based analysis tool. It can incorporate several nuclear processes in the analysis and simultaneously reduce the model-dependence of the analysis.

Contents

Acceptance Page **ii**

Abstract **iii**

1 Introduction and Motivation **1**
1.1 The Standard Model 1
1.2 Standard Model vs. Experiment 2
1.3 Neutrino Oscillations 3
1.4 Overview 4
1.5 Oscillation Theory 5
1.6 Experimental Status 8

2 Neutrino Cross-Sections **10**
2.1 General Description 10
2.2 Neutrino-Nucleus Scattering Processes 14
2.3 Charged-Current Quasi-Elastic Scattering 16
2.4 Charged-Current Whole-Nucleus Scattering 21

3 Kinematics **26**
3.1 Kinematics with known E_ν 26
3.2 General Kinematics 29
3.3 Simplified Models 31
3.4 Detailed Models 34

4 The Multi-Beam Strategy **36**
4.1 Principle 37
4.2 Implementation 41

5 Neutrino Oscillation Experiments **45**
5.1 The LSND Experiment 45
5.2 MiniBooNE Experiment 46

5.3 MiniBooNE Analysis 50
5.4 MiniBooNE Results 50
5.5 Consequences 53

6 Analysis and Results **56**
6.1 CCQE only 57
6.2 CCQE and CCWN combined 60
6.3 Multi-Beam Strategy 63
6.4 MiniBooNE ν_e excess 64

7 Conclusions **69**

A The C++ Code **71**
A.1 main.cpp 71
A.2 boonies.h 85

Bibliography **99**

Chapter 1

Introduction and Motivation

1.1 The Standard Model

In the current era, physicists have arrived at what is called the Standard Model of particle physics. The Standard Model takes into account three of the four fundamental forces: the electromagnetic force, the weak force, and the strong force. Only the fourth force, gravitation, is not a part of the Standard Model.

In the Standard Model, the forces are transmitted by gauge bosons: photons for the electromagnetic force, W and Z bosons for the weak force, and gluons for the strong force. The Standard Model also contains Higgs bosons, which have not yet been observed experimentally.

In addition to the gauge bosons, the Standard Model includes fermions. The fermions are divided into quarks and leptons. Quarks interact via all the three forces in the Standard Model and come in six "flavors": down and up, strange and charm, bottom and top. Leptons are divided into charged leptons and neutrinos, and come in three families: the electron and electron neutrino, the muon and muon neutrino, and the tau and tau neutrino. The charged leptons have one unit of elementary charge and interact via the electromagnetic

force as well as the weak force. The neutrinos, on the other hand, carry no electric charge and interact only via the weak force. And while the masses of the charged leptons range from 511 keV for the electron to 1777 MeV for the tau, the neutrinos are massless in the Standard Model.

1.2 Standard Model vs. Experiment

The predictions of the Standard Model have been tested in many experiments, overall with great success: the predictions for the muon magnetic moment anomaly, for example, agree with experimental data within just a few parts per billion (see, e.g. Ref. [1]).

Neutrino physics, which is discussed in this dissertation, is far less accurate than the above example: as we will see in Section 1.6, only the order of magnitude is known for some of the values. This lack of extremely accurate experimental data can be attributed to the fact that neutrinos interact only via the weak interaction, which is very weak – hence the name. Due to the small cross-sections associated with the weak interaction, there are two requirements for a successful neutrino experiment: a large flux and a large detector volume. Fulfilling these two requirements, however, makes it experimentally difficult to achieve very accurate data.

Aside from experimental accuracy, physicists have been puzzled by the largest neutrino facility in our local vicinity, the Sun. The Sun produces more than 10^{38} neutrinos every second, which yields a flux of more than 10^{14} neutrinos per square-meter per second at the location of the Earth. However,

experiments detected fewer neutrinos than predicted by the best solar models at the time (see, e.g. the discussion in Ref. [2]). This led to the theory of neutrino oscillations, which are a sign for "physics beyond the Standard Model".

1.3 Neutrino Oscillations

Neutrino oscillations can explain the deficiency in the observed solar neutrino flux as follows. On their way from the Sun to the Earth, electron neutrinos converted into muon or tau neutrinos. Since the detector measured only electron neutrinos, this could explain the experimental data.

Strong evidence for the neutrino oscillation theory as an explanation for the Solar neutrino problem came in 2002 from the Sudbury Neutrino Observatory, SNO [3]. SNO measured both electron neutrinos from the Sun, and neutrinos from the Sun of all families combined. It was then clearly shown that by adding the muon and tau neutrinos, the total neutrino flux from the Sun agreed well with the predictions from the solar models.

Among the other experiments that strongly indicated neutrino oscillations are Super-Kamiokande [4], KamLAND [5] and K2K [6]. Refs. [3, 4, 5] and [6] have well more than 500 citations each, indicating the great interest in this field.

1.4 Overview

In this dissertation, we discuss the neutrino-nucleus cross-sections, with focus on the kinematic region of momentum transfer $Q^2 \lessapprox 10^5$ MeV^2, our *kinematic region of interest*. A good understanding of these cross-sections is essential for the success of neutrino oscillation experiments. However, we have noticed that the experimental groups use insufficient models to describe the neutrino-nucleus cross-sections. Therefore, we present a data-driven method to reduce the model-dependence of neutrino oscillation experiments. This method, called "Multi-Beam Strategy" can greatly improve the neutrino experiments.

Our intention is to show a proof of principle for the Multi-Beam Strategy. Doing a detailed re-evaluation of experimental data requires the knowledge of experimental details, such as detector acceptance, and lies beyond the scope of this dissertation.

We proceed as follows. In Sections 1.5 and 1.6, we complete the introduction by presenting the theoretical framework of neutrino oscillations and providing the current experimental status of the oscillation parameters. In Chapter 2, we discuss the neutrino-nucleus cross-sections and the models that are used to describe these cross-sections. Chapter 3 then provides a detailed analysis of the kinematics of the neutrino-nucleus scattering processes. The kinematics are a crucial element in the Multi-Beam Strategy. We present the Multi-Beam Strategy in Chapter 4, using the models and kinematics developed in Chapters 2 and 3, respectively. After discussing the principle of the Multi-Beam Strategy in Section 4.1, we describe how we implemented it in

C++ code in Section 4.2. The C++ code itself can be found in the Appendix. In Chapter 5, we present the LSND and MiniBooNE experiments, which have a significant number of events in the kinematic region of interest and can greatly benefit from the Multi-Beam Strategy. In Chapter 6, we then show the proof of principle for the Multi-Beam Strategy by re-analyzing data from the MiniBooNE experiment. Finally, Chapter 7 will contain some concluding remarks.

1.5 Oscillation Theory

To start the discussion of the oscillation theory, let us first consider only two families of neutrinos, the electron and muon neutrinos. Including the tau neutrinos later on will be straightforward. For oscillations to happen, two requirements have to be fulfilled: First, the neutrinos in the flavor basis (that is, their different families when undergoing weak interactions) must be a mixture of the neutrinos in the mass basis. We will denote the flavor eigenstates by ν_e and ν_μ for the electron and muon neutrino, respectively, and the two mass eigenstates by ν_1 and ν_2. Then we have neutrino mixing as follows, with a mixing angle θ:

$$\begin{pmatrix} \nu_1 \\ \nu_2 \end{pmatrix}_{\text{mass}} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} \nu_e \\ \nu_\mu \end{pmatrix}_{\text{flavor}} . \tag{1.1}$$

Second, the mass eigenstates need to have different masses, in order for the neutrino oscillations to occur. Any non-zero neutrino mass is in contradiction

to the Standard Model, where the neutrinos have no mass, and is therefore one sign of "physics beyond the Standard Model".

Neutrino oscillations work as follows. A neutrino is produced in a weak interaction in a flavor eigenstate. It then propagates through space as a mixture of mass eigenstates. Since the mass eigenstates have different masses, they get out of phase as they propagate. Finally, the neutrino is detected in a weak interaction, therefore in its flavor eigenstate. But since the mass eigenstates got out of phase, the detection flavor can be different from the production flavor: a neutrino oscillation occurred.

As an example, let us start with an electron neutrino, $\nu = (1,0)_{\text{flavor}}$ in the flavor basis, or $\nu = (\cos\theta, -\sin\theta)_{\text{mass}}$ in the mass basis. Neutrinos propagate as waves proportional to $e^{i(Et-\mathbf{p}\cdot\mathbf{x})}$, which simplifies to $e^{i(Et-px)}$ if we assume propagation in the x-direction. Here, the energy E is related to the momentum p by $E^2 - p^2 = m^2$. Since neutrino masses turn out to be small compared to their energies, this equation can be approximated by $p = E - m^2/2E$. We find for the propagating neutrino:

$$\begin{aligned} \nu &= \begin{pmatrix} \cos\theta \; e^{i(Et-(E-m_1^2/2E)x)} \\ -\sin\theta \; e^{i(Et-(E-m_2^2/2E)x)} \end{pmatrix}_{\text{mass}} \\ &= e^{i(Et-(E-m_1^2/2E)x)} \begin{pmatrix} \cos\theta \\ -\sin\theta \; e^{i(m_2^2-m_1^2)x/2E} \end{pmatrix}_{\text{mass}} . \end{aligned} \tag{1.2}$$

Disregarding the overall phase and writing $\Delta m^2 = m_2^2 - m_1^2$, we can transform

back into the flavor basis:

$$\nu = \begin{pmatrix} \cos^2\theta + \sin^2\theta \; e^{i\Delta m^2 x/2E} \\ \cos\theta\sin\theta(1 - \; e^{i\Delta m^2 x/2E}) \end{pmatrix}_{\text{flavor}} . \tag{1.3}$$

Squaring the ν_μ component of the neutrino vector in Eq. (1.3) gives us the probability that an electron neutrino oscillates into a muon neutrino:

$$\begin{aligned} P(\nu_e \to \nu_\mu) &= \left|\cos\theta\sin\theta(1 - \; e^{i\Delta m^2 x/2E})\right|^2 \\ &= \frac{1}{2}\sin^2 2\theta\left(1 - \cos\left(\frac{\Delta m^2 x}{2E}\right)\right). \end{aligned} \tag{1.4}$$

The probability that an electron neutrino stays an electron neutrino is, of course, $P(\nu_e \to \nu_e) = 1 - P(\nu_e \to \nu_\mu)$.

The generalization to three neutrino flavors and masses is, as mentioned earlier, straightforward. The mixing matrix is called Maki-Nakagawa-Sakata matrix, or short MNS matrix [7]. Instead of only one mixing angle, we now have three mixing angles and one complex phase. Two further complex phases are not important for neutrino oscillations and will not be shown here. A common parameterization of the mixing matrix, U, is (see, e.g. Ref. [8]):

$$U = \begin{pmatrix} 1 & 0 & 0 \\ 0 & c_{23} & s_{23} \\ 0 & -s_{23} & c_{23} \end{pmatrix} \begin{pmatrix} c_{13} & 0 & s_{13}e^{-i\delta} \\ 0 & 1 & 0 \\ -s_{13}e^{i\delta} & 0 & c_{13} \end{pmatrix} \begin{pmatrix} c_{12} & s_{12} & 0 \\ -s_{12} & c_{12} & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{1.5}$$

where $c_{23} = \cos\theta_{23}$, $s_{23} = \sin\theta_{23}$, and so on.

1.6 Experimental Status

The mixing parameters introduced above have to be measured experimentally, especially since there is no convincing theory that predicts the mixing parameters. Also, there are two mass squared differences to be determined (since the third mass squared difference is just the difference between the other two). Experiments use several different neutrino sources. Solar neutrinos are produced in fusion processes in the Sun. Atmospheric neutrinos come mostly from the decay of charged pions and kaons in cosmic ray showers. Reactor neutrinos are produced in the radioactive decays in nuclear reactors. Finally, accelerator neutrinos are produced when a high energy particle beam from an accelerator gets dumped in a target, hadronizes, and the charged pions and kaons decay to neutrinos.

Combining results from all these experiments, the currently best values are, according to Ref. [9]:

$$\begin{aligned}
\sin^2 2\theta_{12} &= 0.86^{+0.03}_{-0.04} \\
\sin^2 2\theta_{23} &> 0.92 \\
\sin^2 2\theta_{13} &< 0.19 \\
\Delta m^2_{21} &= (8.0 \pm 0.3)\ 10^{-5}\ \mathrm{eV}^2 \\
\Delta m^2_{32} &= \pm(1.9 \text{ to } 3.0)\ 10^{-3}\ \mathrm{eV}^2.
\end{aligned}$$

Please note that these experimental values have two potential weaknesses. First, the inital neutrino flux has to be known. The results will therefore de-

pend on how good the models for the specific production processes are, such as the Solar models, or the models for cosmic ray evolution. Second, the final neutrino detection needs to be known. The number of detected neutrinos is directly proportional to the neutrino cross-section, thus a mistake in the cross-sections will directly translate into a mistake in neutrino oscillation parameters.

A good knowledge of the neutrino cross-sections, with as little model dependence as possible, is therefore invaluable for neutrino experiments. As a prerequisite for the Multi-Beam Strategy, which reduces the model dependence, we review these neutrino cross-sections in the next Chapter.

Chapter 2

Neutrino Cross-Sections

As we have argued in Section 1.6, a good understanding of neutrino cross-sections is crucial for neutrino oscillation experiments. In this chapter, we discuss the neutrino-nucleus cross-sections. After presenting the most general description, we study two special processes, the charged-current quasi-elastic scattering (CCQE) and the charged-current whole-nucleus scattering (CCWN). We have chosen these two processes over many other models because current experiments happen to favor them, making them central whether or not they are credible. Moreover, the model details are not essential for the Multi-Beam Strategy.

2.1 General Description

In this general description, we treat the scattering of neutrinos on a nuclear target, here called T. The discussion can be easily extended to anti-neutrinos. The reaction on the neutrino side is easy: the neutrino exchanges a W boson and becomes a charged lepton. On the target side, however, things are not so easy: we know that the target interacts with the W, and that the target final

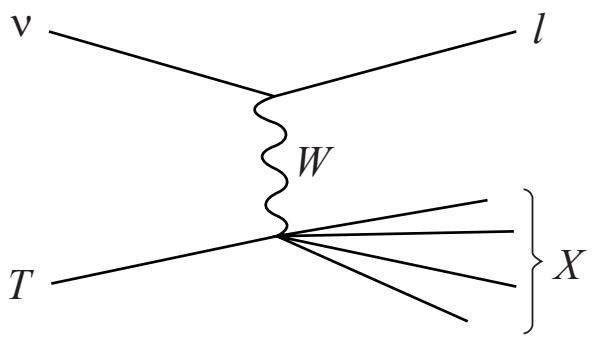


Figure 2.1: Feynman diagram for the general process of a neutrino ν undergoing charged-current scattering off a target T.

state will gain one unit of electric charge. We will also assume that the target is at rest initially, which is accurate neglecting some negligible Brownian motion. But there are many different final states possible, so we will call the target final state X. A Feynman diagram of the process can be seen in Fig. 2.1.

To denote the variables, let us use ν^μ, l^μ, T^μ and X^μ for the four-vectors of the neutrino, charged lepton, target and target final state, respectively. The four-momentum transfer is then $q^\mu = \nu^\mu - l^\mu = X^\mu - T^\mu$, and $Q^2 = -q^2 = -q^\mu q_\mu$, as usual. And the invariant mass of the target final state is $m_X = \sqrt{X^\mu X_\mu}$.

The amplitude for this process is

$$|\mathcal{M}(\nu T \to lX)| = \frac{g_w}{2\sqrt{2}} \left[\bar{u}(l)\gamma^\mu(1-\gamma_5)u(\nu)\right] \frac{g_{\mu\nu} - \frac{q_\mu q_\nu}{m_W^2}}{q^2 - m_W^2} \frac{g_w}{2\sqrt{2}} \left\langle X\right| J^\nu \left|T\right\rangle , \quad (2.1)$$

where g_w is the weak coupling constant, and m_W the mass of the W boson. The matrix element $\langle X| J^\nu |T\rangle$ is undetermined yet. Since $m_W \gg q^2$, we can

neglect terms of the order $\mathcal{O}(q^2/m_W^2)$, and the amplitude simplifies to

$$|\mathcal{M}(\nu T \to lX)| = \frac{G_F}{\sqrt{2}} \left[\bar{u}(l)\gamma^\mu(1-\gamma_5)u(\nu)\right] \langle X| J_\mu |T\rangle . \tag{2.2}$$

Here, we have used the Fermi constant, $G_F = \sqrt{2}g_w^2/8m_W^2$. As usual, the cross-section is given by

$$\mathrm{d}\sigma = \frac{\left\langle |\mathcal{M}|^2 \right\rangle \mathrm{d}ps_n}{2\lambda^{1/2}(s, m_\nu^2, m_T^2)(2\pi)^{3n-4}}. \tag{2.3}$$

Here, $\mathrm{d}ps_n$ stands for the n-body phase-space differential, and λ is given by $\lambda(a,b,c) = a^2 + b^2 + c^2 - 2ab - 2ac - 2bc$. The neutrino mass is so small that we can neglect it, and we find in the lab frame: $\lambda^{1/2}(s, m_\nu^2, m_T^2) = 2E_\nu m_T$. We can write the phase-space differential as

$$\mathrm{d}ps_n = \frac{\mathrm{d}^3\mathbf{p}_l}{2E_l}\mathrm{d}ps_m, \tag{2.4}$$

where $m = n - 1$ is the number of bodies in the final state X. Then, the cross-section is

$$\begin{aligned} E_l\frac{\mathrm{d}\sigma}{\mathrm{d}^3\mathbf{p}_l} &= \frac{G_F^2}{64\pi^2 E_\nu m_T} \left(\left[\bar{u}(l)\gamma^\mu(1-\gamma_5)u(\nu)\right]\left[\bar{u}(\nu)\gamma^\nu(1-\gamma_5)u(l)\right] \times \right. \\ &\quad \left. n_{\text{spins}}^{-1} \langle X| J_\mu |C\rangle \langle X| J_\nu |C\rangle^* \mathrm{d}ps_m (2\pi)^{3-3m}\right), \end{aligned} \tag{2.5}$$

where n_{spins} is the number of spins of the target. At this point, it is convenient to define the lepton tensor $L^{\mu\nu}$ and the hadron tensor $W_{\mu\nu}$ (see, e.g. [10]). The

lepton tensor is

$$\begin{aligned} L^{\mu\nu} &= \frac{1}{4}\left[\bar{u}(l)\gamma^{\mu}(1-\gamma_5)u(\nu)\right]\left[\bar{u}(\nu)\gamma^{\nu}(1-\gamma_5)u(l)\right] \\ &= \frac{1}{4}\mathrm{Tr}\left[(\not{p}_l + m_l)\gamma^{\mu}(1-\gamma_5)\not{\nu}\gamma^{\nu}(1-\gamma_5)\right] \\ &= 2\left(\nu^{\mu}l^{\nu} + l^{\mu}\nu^{\nu} - g^{\mu\nu}\nu\cdot l + i\varepsilon^{\mu\nu\lambda\rho}\nu_{\lambda}l_{\rho}\right), \end{aligned} \tag{2.6}$$

and the hadron tensor is given by

$$W_{\mu\nu} = \frac{1}{n_{\mathrm{spins}}}\sum_{\mathrm{spins}}\frac{\mathrm{d}ps_m}{2m_T}\left\langle X\right|J_{\mu}\left|T\right\rangle\left\langle X\right|J_{\nu}\left|T\right\rangle^{*}(2\pi)^{3-3m}. \tag{2.7}$$

With these definitions, we can write the cross-section as

$$E_l\frac{\mathrm{d}\sigma}{\mathrm{d}^3\mathbf{p}_l} = \frac{G_F^2}{8\pi^2 E_{\nu}}L^{\mu\nu}W_{\mu\nu}. \tag{2.8}$$

The big unknown in this equation is the hadron tensor. If we have no information about the matrix elements $\left\langle X\right|J_{\mu}\left|T\right\rangle$, then we can write $W_{\mu\nu}$ in terms of the six Lorentz invariants $g_{\mu\nu}$, $q_{\mu}q_{\nu}$, $T_{\mu}T_{\nu}$, $q_{\mu}T_{\nu}$, $T_{\mu}q_{\nu}$ and $\varepsilon_{\mu\nu\lambda\rho}q^{\lambda}T^{\rho}$. Neglecting terms of order $\mathcal{O}(m_l^2)$, $W_{\mu\nu}$ can be written in terms of only three structure functions, W_1, W_2, and W_3:

$$W_{\mu\nu} = -g_{\mu\nu}W_1 + \frac{T_{\mu}T_{\nu}}{m_T^2}W_2 - \frac{i\varepsilon_{\mu\nu\lambda\rho}T^{\lambda}q^{\rho}}{2m_T^2}W_3. \tag{2.9}$$

However, no matter what the number of structure functions is, they have to be measured! Measurements are difficult in neutrino experiments, as will

be shown in detail in Chapter 5. It has therefore become standard practice in neutrino physics to use nuclear models as inputs, and then measure and adjust maybe one or two parameters.

Many neutrino scattering processes have only one body in the target final state, so we will have a closer look at these processes. For this case, it is convenient to define the one-body hadron tensor $\widetilde{W}_{\mu\nu}$:

$$\widetilde{W}_{\mu\nu} = \frac{1}{n_{\text{spins}}} \sum_{\text{spins}} \langle X | J_\mu | T \rangle \langle X | J_\nu | T \rangle^* . \tag{2.10}$$

The cross-section is then

$$E_l \frac{\mathrm{d}\sigma}{\mathrm{d}^3\mathbf{p}_l} = \frac{G_F^2}{8\pi^2 E_\nu} \frac{\mathrm{d}ps_1}{2m_T} L^{\mu\nu} \widetilde{W}_{\mu\nu}, \tag{2.11}$$

or, after some algebra,

$$\frac{\mathrm{d}\sigma}{\mathrm{d}q^2} = \frac{G_F^2}{32\pi E_\nu^2 m_T^2} L^{\mu\nu} \widetilde{W}_{\mu\nu}. \tag{2.12}$$

Before we apply the equations that we have just developed to specific processes, we give an overview over the different processes in neutrino-nucleus scattering.

2.2 Neutrino-Nucleus Scattering Processes

The most important processes are charged-current whole-nucleus scattering, charged-current quasi-elastic scattering, one-pion production and deep inelas-

tic scattering. CCWN and CCQE are described in Sections 2.4 and 2.3, respectively. One-pion production includes all processes in which the final state consists of the charged lepton, the recoiled nuclear target and one additional pion. And deep inelastic scattering summarizes the processes in which the neutrino scatters off single quarks rather than nucleons or nuclei.

The contributions to the total neutrino-nucleus cross section from these different processes is shown in Fig. 2.2. The figure from Ref. [11] neglects, however, the whole-nucleus scattering at low neutrino energies. Quasi-elastic scattering provides the single largest contribution to the cross-section estimated in Fig. 2.2 in the kinematic region of interest, and therefore has to be included. One-pion production gets significant at higher neutrino energies and momentum transfers, and should be included for analyses that are sensitive to the higher energy events, like the measurement of the axial mass.

Here, however, we are concerned with the lower momentum transfer events and choose to not include one-pion-production processes. In favor of this decision, it can also be argued that the charged, or neutral, pion would decay to a muon and ν_μ, or two photons, respectively, which would be detected and disqualify this event as CCQE neutrino event. To which degree it is possible to distinguish a one-pion production event from a CCQE event depends on details of the experiment, and lies beyond the scope of this dissertation.

Finally, at even higher neutrino energies, deep inelastic scattering takes over. However, the influence of DIS on the kinematic region of interest is negligible.

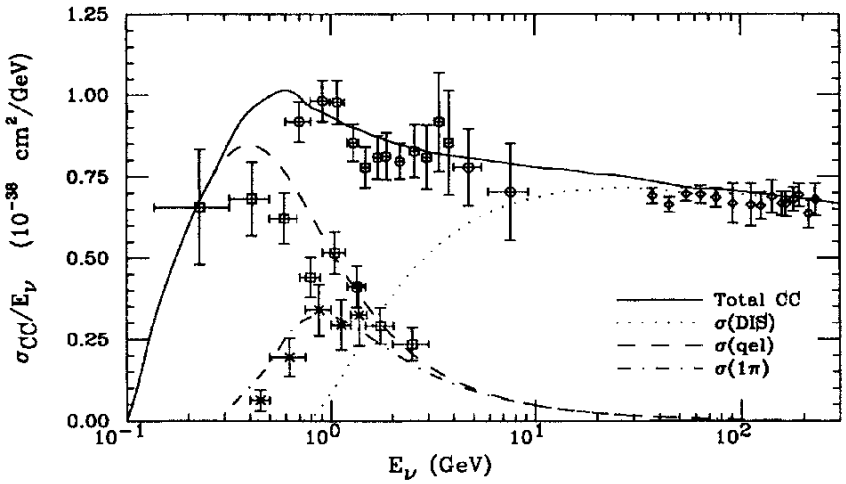


Figure 2.2: Charged-current ν_μ cross-sections as a function of neutrino energy, from Ref. [11]. The cross-sections shown are for the inclusive process (Total CC), quasi-elastic scattering (qel), one-pion production (1π) and deep inelastic scattering (DIS).

In the following, we will discuss in more detail the two processes that are important for the kinematic region of interest, $Q^2 \lessapprox 10^5$ MeV^2: charged-current quasi-elastic scattering and charged-current whole-nucleus scattering.

2.3 Charged-Current Quasi-Elastic Scattering

Charged-current quasi-elastic scattering describes the scattering in which the W boson (from the $\nu \to l$ side of the reaction) interacts with a single neutron in the nucleus. This neutron then becomes a proton and gets knocked out of the nucleus.

This process is usually treated in the impulse approximation, meaning that the remainder of the nucleus does not participate in the reaction, except for providing a binding energy for that one nucleon. The impulse approximation

is not ideal for nuclear reactions, since the nucleons in the nucleus are all very close together and interact strongly. However, an inclusion of nucleon-nucleon interactions inside the nucleus would make the problem much more complicated. Therefore, we will accept the impulse approximation for this discussion.

The model for the nuclear part that we will choose is the relativistic Fermi gas model (RFG) [12]. This choice has two motivations: First, it is the model chosen by the several current neutrino experiments, and we will therefore not differ in this point of their analysis. Second, the RFG is still simple enough not to require a whole nuclear model machinery behind the calculations. Shell model calculations, as another possible example, would require input, such as the exact nuclear potentials to be used, from external experts and it would require a lot more time to get the models up and running. Since the purpose of this dissertation is to show a proof of principle, rather than provide accurate numerical results, it is acceptable to use the RFG, even though it is more than 35 years old and has its limitations. Our approach (Chapter 4) has been designed so that more detailed calculations can always be done later when the level of experimental detail needed for these calculations is at hand.

In the initial state, the relativistic Fermi gas model assumes the nucleons to be evenly distributed in momentum space, up to a certain Fermi momentum. For ^{12}C, a popular nuclear target for accelerator neutrino experiments, the nominal Fermi momentum is $p_F = 220$ MeV. Also, the nucleons are bound by a binding energy of $E_b = -34$ MeV [13].

For the interaction, the neutrons are now treated as free particles except for the binding energy. To calculate the cross-section, we make use of the formalism developed in Section 2.1. Since the initial state is a neutron, the number of spins in Eq. (2.10) will be $n_{\text{spins}} = 2$. Neutrons and protons have been studied in electron scattering experiments, so we actually have some good information on the matrix element $\langle p| J_\mu |n\rangle$. Neglecting the pseudoscalar form factor, the matrix element can be written as [14]

$$\langle p| J_\mu |n\rangle = \cos\theta_C \bar{u}(p)\Gamma_\mu u(n), \tag{2.13}$$

where θ_C is the Cabbibo angle and the vertex is given by

$$\Gamma_\mu = \gamma_\mu F_V^1(q^2) + \frac{i\sigma_{\mu\nu}q^\nu \xi F_V^2(q^2)}{2m_n} + \gamma_\mu\gamma_5 F_A(q^2). \tag{2.14}$$

We use this matrix element in Eq. (2.10) and the one-body hadron tensor in Eq. (2.12) to get the differential cross section. After some algebra, we find

$$\frac{\mathrm{d}\sigma}{\mathrm{d}q^2} = \frac{m_n^2 G_F^2 \cos^2\theta_C}{8\pi E_\nu^2}\left(A(q^2) - B(q^2)\frac{s-u}{m_n^2} + C(q^2)\frac{(s-u)^2}{m_n^4}\right), \tag{2.15}$$

where m_n is the neutron mass, G_F the Fermi constant, E_ν the neutrino energy, s and u the usual Mandelstam variables with $(s-u) = 4E_\nu m_n + q^2 - m_l^2$, and $A(q^2)$, $B(q^2)$ and $C(q^2)$ are given by

$$A(q^2) = \frac{m_l^2 - q^2}{4m_n^2}\left[\left(4 - \frac{q^2}{m_n^2}\right)|F_A|^2 - \left(4 + \frac{q^2}{m_n^2}\right)|F_V^1|^2 - \frac{q^2}{m_n^2}|\xi F_V^2|^2\right.$$

$$\times \left(1+\frac{q^2}{4m_n^2}\right) - \frac{4q^2 F_V^1 \xi F_V^2}{m_n^2} - \frac{m_l^2}{m_n^2}((F_V^1+\xi F_V^2)^2 + |F_A|^2\Bigg],$$

$$B(q^2) = \frac{q^2}{m_n^2}((F_V^1+\xi F_V^2)F_A),$$

$$C(q^2) = \frac{1}{4}\left(|F_A|^2 + |F_V^1|^2 - \frac{q^2}{4m_n^2}|\xi F_V^2|^2\right). \quad (2.16)$$

The form factors F_V^1 and F_V^2 can be expressed in terms of the electric and magnetic form factors of the proton and neutron (the so-called Sachs form factors), G_E^p, G_E^n, G_M^p, and G_M^n, respectively:

$$F_V^1(q^2) = \left(1-\frac{q^2}{4m_n^2}\right)^{-1}\left[G_E^p(q^2) - G_E^n(q^2) - \frac{q^2}{4m_n^2}(G_M^p(q^2) - G_M^n(q^2))\right]$$

$$\xi F_V^2(q^2) = \left(1-\frac{q^2}{4m_n^2}\right)^{-1}\left[G_M^p(q^2) - G_M^n(q^2) - G_E^p(q^2) + G_E^n(q^2)\right]. \quad (2.17)$$

The Sachs form factors are well measured from electron scattering experiments. We will use the parameterization of these form factors shown in Ref. [15]. The form factors are parameterized by

$$G(Q^2) = \frac{\sum_{k=0} a_k (Q^2/4m_n^2)^k}{1+\sum_{k=1} b_k (Q^2/4m_n^2)^k}, \quad (2.18)$$

where the values for a_k and b_k are shown in Table 2.1.

The axial form factor F_A is approximated as a dipole,

$$F_A(q^2) = -\frac{g_A}{(1-q^2/m_A^2)^2}, \quad (2.19)$$

with $g_A = 1.2720$. The axial mass, m_A, is treated as a free parameter and will

Form Factor	a_0	a_1	a_2	b_1	b_2	b_3	b_4
G_E^p	1	-0.0578		11.1	13.6	33.0	
G_M^p	1	0.150		11.1	19.6	7.54	
G_E^n	0	1.25	1.30	-9.86	305.	-758.	802
G_M^n	1	1.81		14.1	20.7	68.7	

Table 2.1: Values for the parameterization of the form factors, Eq. (2.18), from Ref. [15].

be fit to data.

Now the final state has to be considered. Since nucleons are fermions, they have to obey the Pauli exclusion principle. This leads to what is referred to as Pauli blocking: if the scattered nucleon has a momentum less than the Fermi momentum, then it would be in a state that is already occupied by a nucleon, which is forbidden according to the Pauli exclusion principle. In that case, the reaction would simply not happen at all. Mathematically, in terms of the energy of the scattered nucleon, E', this requirement is:

$$E' > \sqrt{m_p^2 + p_F^2}. \tag{2.20}$$

At this point in the model, an *ad hoc* Pauli blocking parameter κ is introduced (see, e.g. Ref. [16]). It modifies Eq. (2.20) to

$$E' > \kappa\sqrt{m_p^2 + p_F^2}. \tag{2.21}$$

For $\kappa > 1$, scattered nucleons that barely made it out before will now be blocked. There is not much theoretical motivation to the Pauli blocking pa-

rameter other than patching up a basic model with parameters to make agreement with experimental data better.

2.4 Charged-Current Whole-Nucleus Scattering

A contribution to the neutrino-nucleus cross-section that has been neglected (or overlooked) both by many experiments as well as by Lipari *et al.* is the charged-current whole-nucleus scattering (CCWN). This is the reaction where the W converts the target nucleus into a different nucleus. In the popular case of charged-current reactions of neutrinos on ^{12}C, the reaction would be

$$\nu +^{12} C \rightarrow l^{-} +^{12} N. \tag{2.22}$$

If one considers only the exclusive reaction with the ground state of nitrogen, $^{12}N_{\text{g.s.}}$ in the final state, then the cross-section would indeed be very small, and neglecting it compared to CCQE would be justified.

If one considers, however, the inclusive reaction, with all excited states of nitrogen, $^{12}N^{*}$, allowed in the final state, then the CCWN cross-section is actually comparable to the CCQE cross-section, at neutrino energies of 300 MeV, for example. Cross-section results from model calculations and data, illustrating these claims, are shown in Table 2.2.

The state-of-the-art way to calculate these reactions is with Shell Model calculations, or calculations closely related to Shell Model calculations, such as the Random Phase Approximation. The gist of Shell Model calculations is

σ/cm^2		Model used	Ref.
ν_e	ν_μ		
CCWN, exlusive to $^{12}N_{\text{g.s.}}$			
$8\ 10^{-41}$		Continuum Random Phase Approximation	[17]
$8\ 10^{-41}$	$6\ 10^{-41}$	RPA, one-body transitions densities, elementary particle treatment [a]	[18]
$8\ 10^{-41}$	$8\ 10^{-41}$	Form factors from experiment [b]	[19]
$9\ 10^{-41}$		Model-invariant, directly from data [c]	[20]
CCWN, inclusive to $^{12}N^*$			
$2\ 10^{-38}$		Continuum Random Phase Approximation	[17]
$1\ 10^{-39}$	$1\ 10^{-39}$	Particle-hole model	[21]
CCQE			
	$2\ 10^{-39}$	From data; scattering is mostly quasi-elastic in this region	[11]
$2\ 10^{-38}$	$2\ 10^{-38}$	Quasi-elastic, local density approximation, includes Pauli blocking, Fermi motion, strong interaction renormalization; the results are within a factor of two compared to Fermi gas models	[22]

Table 2.2: Neutrino cross-sections for neutrinos of energy 300 MeV, except for [a] 250 MeV, [b] $>$ 125 MeV for ν_e and $>$ 150 MeV for ν_μ, [c] 100 MeV. While σ for exclusive CCWN scattering is about two orders of magnitude below σ for CCQE, σ for inclusive CCWN scattering is comparable to σ for CCQE.

that they calculate the wavefunctions of the nucleons inside the nucleus in the different shells – much like the orbitals for electrons in an atom. A nuclear reaction is then calculated from the overlap of the initial wavefunction and the final wavefunction (see, e.g. Ref. [23]). Unlike the case with electrons around atoms, the nuclear potential inside the nucleus is not known a priori: the potential has to be found by fitting it to data.

While Shell Model calculations are state-of-the-art, there is no great advantage to get a Shell Model code up and running. First, the uncertainties of the calculations are still too large, and different calculations do not agree. Second, many calculations are "proprietary ", making it difficult to judge the accuracy and validity of the results. Finally, Shell Model calculations use an impulse approximation that neglects interactions from the event, thus the results cannot be exact. Nevertheless, we designed the Multi-Beam Strategy (Chapter 4) so that, in principle, contributions from Shell Model calculations can be incorporated.

Let us return to the formalism developed in Section 2.1. Due to the sparse amount of data, we must limit ourselves to a single structure function. Gauge invariantly, we can write for the nuclear matrix element

$$\langle X|\, J_\mu \,|T\rangle \propto T_\mu - q_\mu \frac{T \cdot q}{q^2}. \tag{2.23}$$

The matrix element in Eq. (2.23) depends on the size of the nucleus, thus we include a form factor

$$F(q^2) = e^{bq^2}, \tag{2.24}$$

with $b = r^2/6$, for the radius of the nucleus, r. The radius is usually approximated by $r = r_0 A^{1/3}$, where $r_0 = 1.25$ fm, and A is the atomic mass number. In the case of ^{12}C, we nominally have $r = 2.86$ fm. However, we will leave r as a free parameter to account for deviations from this approximation.

Finally, we introduce a multiplicative free parameter, A, and get

$$\langle X| J_\mu |T\rangle = AF(q^2)\left(T_\mu - q_\mu \frac{T\cdot q}{q^2}\right). \tag{2.25}$$

We plug this result into Eq. (2.10) and find the one-body hadron tensor:

$$\widetilde{W}_{\mu\nu} = A^2 F(q^2)^2 \left(T_\mu - q_\mu \frac{T\cdot q}{q^2}\right)\left(T_\nu - q_\nu \frac{T\cdot q}{q^2}\right). \tag{2.26}$$

Using this one-body hadron tensor in Eq. (2.12), we get, after some algebra:

$$\frac{\mathrm{d}\sigma}{\mathrm{d}q^2} = \frac{G_F^2}{32\pi E_\nu^2 m_T^2} A^2 F^2(q^2)\left(4m_T^2 E_\nu^2 + \left[q^2 - m_l^2\right]\left[2E_\nu m_T + m_T^2 - \frac{m_l^2}{4}\right]\right). \tag{2.27}$$

A plot of the differential cross-section for a ^{12}C target for different values of the neutrino energy is shown in Fig. 2.3.

In the next Chapter we review the kinematics of our kinematic region of interest. This will complete the background needed for the Multi-Beam Strategy, which follows in Chapter 4.

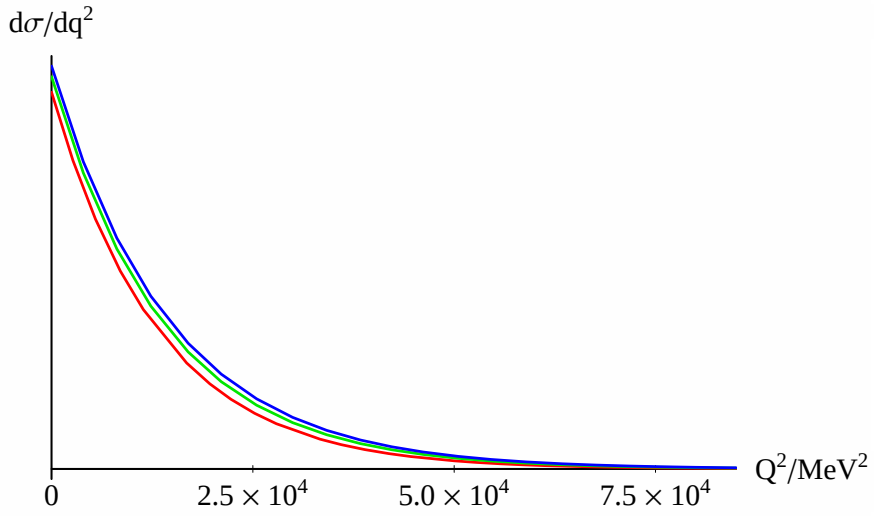


Figure 2.3: CCWN differential cross-section $\mathrm{d}\sigma/\mathrm{d}q^2$ for muon neutrinos scattering off a ^{12}C target as a function of Q^2 for different neutrino energies: 200 MeV (red), 300 MeV (green) and 800 MeV (blue).

Chapter 3

Kinematics

In the previous Chapter we have seen some of the processes that occur in neutrino-nucleus scattering. It is not an easy task to experimentally distinguish one process from the other. One way to do this is to measure the recoiled target, but that is experimentally very difficult in the kinematic region of interest. More about the experimental difficulty of measuring the recoiled target can be found in Section 5.5. And even if the recoiled target could be measured, it does not necessarily point back to one specific process, as we will see shortly.

The next best option to measuring the recoil target is to have the full kinematic information about the incoming neutrino and the outgoing charged lepton. This case will be discussed in section 3.1. The rest of the chapter then discusses the kinematics of the even more difficult case – the reality – in which not even the neutrino energy is known.

3.1 Kinematics with known E_ν

If the energy of the incoming neutrino were known, and we measured the energy of the outgoing lepton and its scattering angle, we would know the

complete kinematics on the lepton side. In particular, we could calculate the momentum transfer $q^\mu = \nu^\mu - l^\mu = (\nu, \mathbf{q})$, where ν is the energy transfer.

If the reaction were elastic scattering off a target initially at rest, the case would be easy: the recoil target four-vector would be $(m_T + \nu, \mathbf{q})$, and its invariant mass still has to be m_T. From this, we find the well-known equation

$$m_T = \frac{Q^2}{2\nu}, \tag{3.1}$$

where $Q^2 = -q^2 = -q^\mu q_\mu$.

The processes discussed in Chapter 2, however, are not elastic scattering off a target initially at rest. In charged-current whole-nucleus scattering, the target (e.g. ^{12}C) is initially at rest, yet the recoil target (e.g. $^{12}N^*$) has a greater mass than the target. If the mass of the recoil target is $m_X = m_T + \varepsilon$, where m_T is once again the target mass, then we find the following relationship:

$$\nu = \varepsilon + \frac{\varepsilon^2}{2m_T} + \frac{Q^2}{2m_T}. \tag{3.2}$$

Knowing ν and Q^2 therefore does not uniquely determine the target mass. In the $Q^2 - \nu$-plot, Fig. 3.1, this results in a band rather than one single line.

For charged-current quasi-elastic scattering, things are even more difficult: not only is there a binding energy, E_b, but the nucleons can have an initial momentum, p_n. This yields the following equation to be solved:

$$2E_nE_b + E_b^2 + 2E_n\nu + 2E_b\nu - 2p_n\sqrt{\nu^2 + Q^2}\cos\zeta = Q^2. \tag{3.3}$$

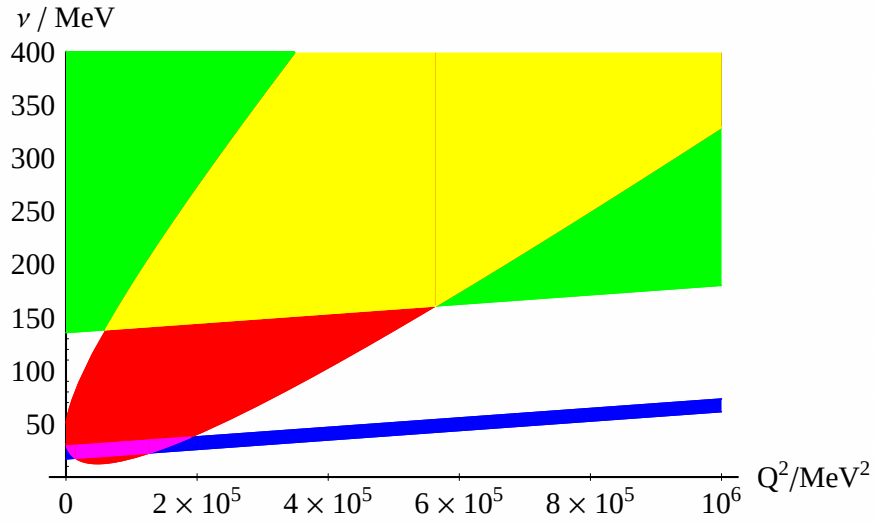


Figure 3.1: $Q^2 - \nu$-plot for CCWN (blue), CCQE (red) and one-pion-production (green). The regions where CCWN and CCQE, and CCQE and one-pion-production overlap are shown in purple and yellow, respectively.

Here, $E_n = \sqrt{m_n^2 + p_n^2}$ is the energy of the initial nucleon, and ζ is the angle between $\mathbf{p}_n$ and $\mathbf{q}$. In the $Q^2 - \nu$ plot, Fig. 3.1, this results in a large area, for all the different values that $\mathbf{p}_n$ can take.

The overlap of the CCWN and CCQE areas in Fig. 3.1 covers almost the whole CCWN band in the kinematic region of interest. In the overlap region, it is impossible to distinguish CCWN from CCQE processes based solely on information from the lepton side. However, even if the recoil target could be detected, it would not necessarily be possible to distinguish between CCWN and CCQE: the main decay channel of $^{12}N^*$ is by proton emission [24]. So whether the detected proton was knocked out in a CCQE reaction, or was a decay proton from an excited nitrogen nucleus, cannot be determined. In fact,

we must acknowledge that in this kinematic region, both models describe the same physical process. This is in no way a flaw of the models, but it makes us aware that they are just *models*.

Finally, we discuss the lowest lying of the other possible processes in Fig. 3.1, the one-pion-production. Kinematically, it is given by the target nucleus of mass m_T initially at rest. The lower bound for the final state invariant mass is $m_T + m_\pi$, where m_π is the mass of the pion. This just means that the recoil target and the produced pion do not move with respect to one another. This lower bound is given, similarly to Eq. 3.2, by:

$$\nu = m_\pi + \frac{m_\pi^2}{2m_T} + \frac{Q^2}{2m_T}. \tag{3.4}$$

There is no upper bound for the kinematics, but it is obvious that other reactions will take over at higher energy transfers.

The overlap of the CCQE and one-pion-production channels in Fig. 3.1 shows that it is impossible to distinguish between these two processes just from knowing the lepton side. However, if the target final state was measured, it would be no problem to distinguish between a proton and a pion.

3.2 General Kinematics

In neutrino experiments, the amount of data per event is actually very small: the direction of the neutrino beam is know, and the active ingredient in the target is known. Besides some negligible Brownian motion, the target is ini-

tially at rest. Then, the angle of the charged lepton with respect to the neutrino beam and the charged lepton's energy are measured. A priori, nothing is known about the recoiled target. For now, we will just assume that its invariant mass is m_X.

Without loss of generality, we can then let the neutrino momentum be in positive x-direction, and the scattered charged lepton move in the $x-y$-plane. The four-vector equation then is

$$\begin{pmatrix} E_\nu \\ E_\nu \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} m_T \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} E_l \\ p_l \cos\theta \\ p_l \sin\theta \\ 0 \end{pmatrix} + \begin{pmatrix} E_X \\ E_\nu - p_l \cos\theta \\ -p_l \sin\theta \\ 0 \end{pmatrix}, \tag{3.5}$$

where we have already written the three-momentum of the recoil target to fulfill the equality. With $E_X = \sqrt{m_X^2 + (E_\nu - p_l \cos\theta)^2 + (p_l \sin\theta)^2}$, we can then solve for the energy of the incoming neutrino:

$$E_\nu = \frac{m_X^2 - m_T^2 + 2m_T E_l - m_l^2}{2(m_T - E_l + p_l \cos\theta)}. \tag{3.6}$$

The big unknown in this equation is m_X, and we need to go back to our specific nuclear models to determine it.

3.3 Simplified Models

Let us first discuss the charged-current quasi-elastic scattering. For simplicity, we will assume that the neutron that gets struck by the W is initially at rest. If we define $n_R^\mu = (m_n, 0, 0, 0)$ for one nucleon at rest, with nucleon mass m_n (we neglect the small difference between proton and neutron mass), then the initial state is of course $T^\mu = A_T n_R^\mu$, where A_T is the mass number of the target nucleus. In a reaction with momentum transfer q^μ, we then find $X^\mu = A_T n_R^\mu + q^\mu$. So m_X is given by

$$m_X^2 = X^\mu X_\mu = A_T^2 m_n^2 + 2A_T n_R \cdot q + q^2 = m_T^2 + 2A_T n_R \cdot q + q^2. \tag{3.7}$$

We also know that the scattered proton will be on-shell, so

$$m_n^2 = (n_R^\mu + q^\mu)(n_{R\mu} + q_\mu) = m_n^2 + 2n_R \cdot q + q^2. \tag{3.8}$$

Combining these equations, we find

$$m_X = \sqrt{m_T^2 - (A_T - 1)Q^2}. \tag{3.9}$$

An easier way to approach this problem is to just think of this process as scattering off a free nucleon. The other $A - 1$ nucleons do not participate in the reaction anyway. For this we can recycle Eq. 3.6, with the substitutions $m_T \to m_n + E_b$ and $m_X \to m_n$. Here, we have even taken care of the binding

energy. The result is

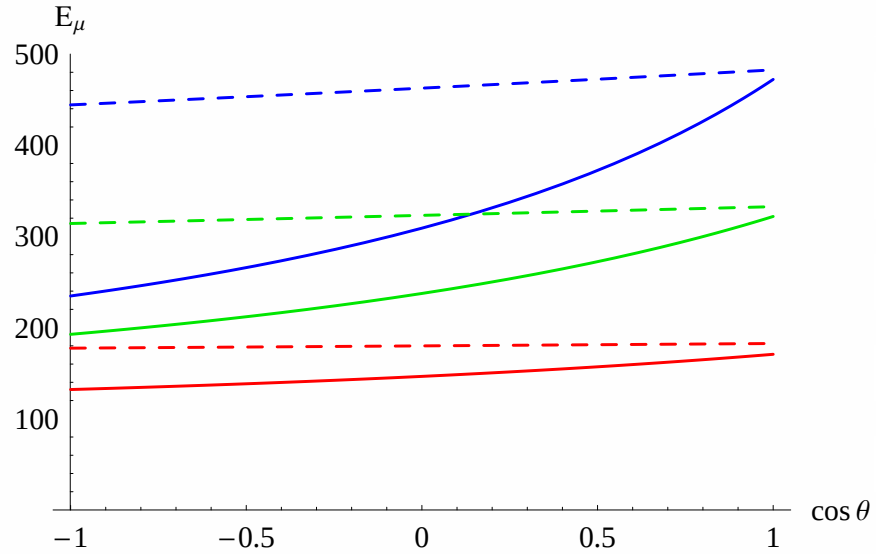
$$E_\nu = \frac{-2m_n E_b - E_b^2 + 2m_n E_l + 2E_b E_l - m_l^2}{2(m_n + E_b - E_l + p_l \cos\theta)}. \tag{3.10}$$

The treatment of charged-current whole-nucleus scattering is even easier: in the initial state, we have a nuclear target with mass m_T, and in the final state, we have the recoiled target with mass $m_X = m_T + \varepsilon$, where ε is the energy of the respective nuclear level above the target mass. For our popular example, $^{12}N_{\text{g.s.}}$, $\varepsilon = 17.338$ MeV [25]. We can therefore find E_ν from

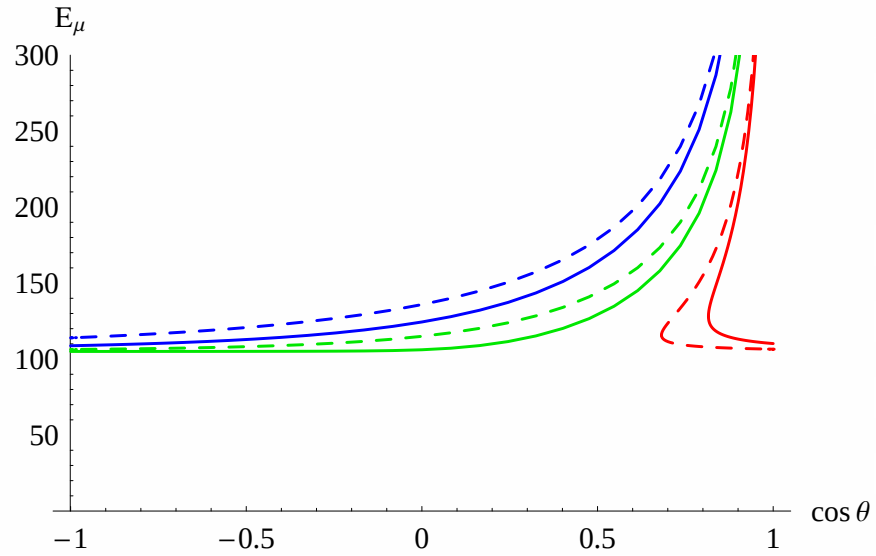
$$E_\nu = \frac{2m_T\varepsilon + \varepsilon^2 + 2m_T E_l - m_l^2}{2(m_T - E_l + p_l \cos\theta)}. \tag{3.11}$$

We note that the kinematics for CCWN differ from those for CCQE. The different form of E_ν, depending on the underlying process, is shown in Fig. 3.2(a). We have chosen E_l and $\cos\theta$ as the axes, since these are the quantities measured in the experiment, and do not depend on the nuclear model used. The different forms of E_ν imply different forms of Q^2, which are shown in Fig. 3.2(b).

For neutrino experiments, this means the following: given a certain charged lepton energy and angle, it is not possible to determine the energy of the neutrino that caused this event. It is also not possible to determine the momentum transfer in a particular reaction. This means that neither flux nor cross-section for any events are known. "Reconstructing" E_ν and Q^2, assuming that the reaction was CCQE, does not do the CCWN events justice: not only is the



(a) Lines of constant neutrino energy: 200 MeV (red), 350 MeV (green), 500 MeV (blue).



(b) Lines of constant Q^2: $(105\ \text{MeV})^2$ (red), $(140\ \text{MeV})^2$ (green), $(176\ \text{MeV})^2$ (blue).

Figure 3.2: $\cos\theta$-E_μ-plots for CCQE (solid lines) and CCWN (dashed).

cross-section different, but the reconstructed values point back to a different neutrino flux and momentum transfer.

3.4 Detailed Models

Before we proceed, let us fill in the model details that we have neglected in the above discussion. For the CCQE kinematics, we assumed that the nucleon was initially at rest. However, the Fermi gas model tells us that the nucleons have a momentum up to p_F inside the nucleus. If we take this into account, the mathematical description of the kinematics gets lengthy, and is not shown here. The lines from Fig. 3.2(a) will now become areas, and are shown in Fig. 3.3.

Similarly, the plots for the CCWN process shown in Fig. 3.2 show only the transition to the ground state of the recoiled target. However, we argued earlier that not only the ground state contributes to CCWN scattering, but also excited states of the recoiled target. The mathematical description is easy, since only the value of the excitation energy ε in Eq. (3.11) has to be updated. The resulting plots are also shown in Fig. 3.3.

In this and the previous Chapter, we have seen that CCWN and CCQE are completely different processes. The practice of many experiments to keep one and neglect the other is therefore not justified. In the next Chapter, we will suggest a better way to proceed.

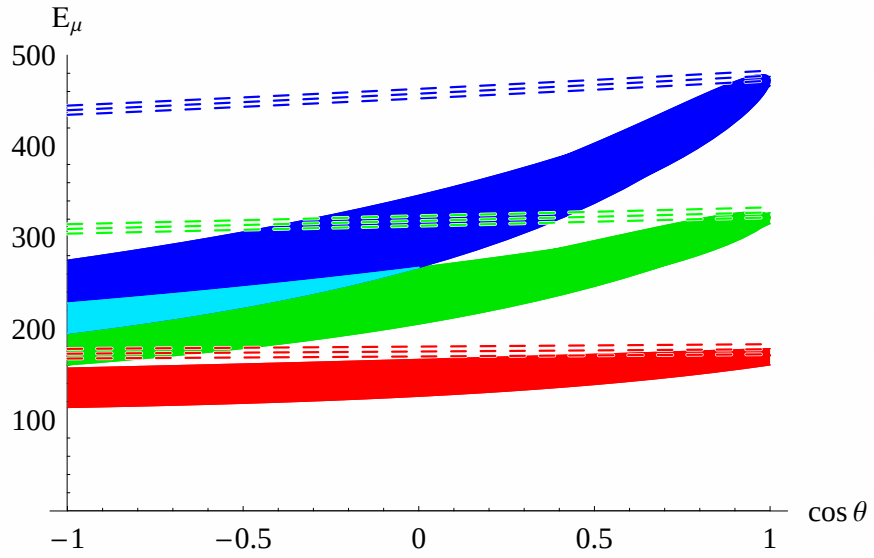


Figure 3.3: $\cos\theta$-E_μ-plots for CCQE (solid areas) and CCWN (dashed). Shown are regions of constant neutrino energy: 200 MeV (red), 350 MeV (green), 500 MeV (blue). The overlap between the 350 MeV area and 500 MeV area for CCQE is shown in turquoise. For CCQE, $p_F = 110$ MeV was used instead of $p_F = 220$ MeV to keep the figure from being too cluttered.

Chapter 4

The Multi-Beam Strategy

In Chapters 2 and 3, we have seen how different the cross-sections and kinematics of CCWN and CCQE processes are. In order to create better experiments, it is therefore essential to take both the CCWN and CCQE processes into account. In this Chapter we present a method that can take both processes into account using standard technology. We call this method *Multi-Beam Strategy*. The Multi-Beam Strategy also has the desirable feature of reducing model-dependence.

After introducing the principle of the Multi-Beam Strategy, we will describe the $C++$ software that we wrote for our analysis. The $C++$ source code itself can be found in the Appendix. In Chapter 6, we will show our results, fitting the models to experimental data. More precisely, we have chosen the LSND and MiniBooNE experiments to show the proof of principle for the Multi-Beam Strategy. Chapter 5 will provide the necessary experimental background to help understand our analysis.

However, we urge the reader to take our results with a grain of salt: while we know the neutrino flux and model the cross-sections, we do not have enough

information to take care of the MiniBooNE detector acceptance. It is to be assumed that the acceptance is greater at higher energies of the charged lepton, but the details are unknown to us. So while our analysis and results provide a proof of principle for the Multi-Beam Strategy, the results may not represent the final numerical value. Finding the exact numerical results, however, could be easily done with information known to the MiniBooNE collaboration, since they already have all the machinery necessary to describe detector acceptance and other experimental details.

4.1 Principle

As will be discussed in more detail in Chapter 5, there are two major experimental difficulties for neutrino experiments: the inability to measure the recoil target and the unavailability of monochromatic neutrino beams. While we do not deal with the former here, the Multi-Beam Strategy will provide a solution for the latter problem. The primary requirement is two (or more) neutrino beams with different neutrino energy spectra. With many accelerator-based neutrino experiments in existence, fulfilling this requirement is not hard.

We will actually make use of the fact that the different processes have different kinematics. To introduce the strategy, we will use the kinematics for the simplified models of Section 3.3. We will then generalize the strategy to the more detailed models, and use it for our analysis.

In the simplified models, the neutrino energy can be uniquely calculated from the charged lepton energy, E_l, and angle, θ, provided that one knows

whether the reaction was CCQE or CCWN. This had been done in Eqs. (3.10) and (3.11), respectively. We will denote these neutrino energies by $E_\nu^{\mathrm{QE}}(E_l,\theta)$ and $E_\nu^{\mathrm{WN}}(E_l,\theta)$, respectively. Having E_ν, we can then find q^2 from $q^2 = m_l^2 - 2E_\nu(E_l - p_l\cos\theta)$. We will denote these q^2 values by $q^2_{\mathrm{QE}}(E_l,\theta)$ and $q^2_{\mathrm{WN}}(E_l,\theta)$, respectively. Then, the neutrino reaction rate R that produces a charged lepton in a small bin around E_l and θ is given by

$$R^Y(E_l,\theta) = \frac{\mathrm{d}\sigma_{\mathrm{QE}}}{\mathrm{d}q^2}(E_l,\theta) f^Y(E_\nu^{\mathrm{QE}}(E_l,\theta)) + \frac{\mathrm{d}\sigma_{\mathrm{WN}}}{\mathrm{d}q^2}(E_l,\theta) f^Y(E_\nu^{\mathrm{WN}}(E_l,\theta)), \quad (4.1)$$

where f is the neutrino flux. The index Y stands for one particular neutrino beam, with a specific neutrino energy spectrum. Of course, the same experiment can be done with another neutrino beam, with a different neutrino energy spectrum, which we will call Z. Just as above, we then get

$$R^Z(E_l,\theta) = \frac{\mathrm{d}\sigma_{\mathrm{QE}}}{\mathrm{d}q^2}(E_l,\theta) f^Z(E_\nu^{\mathrm{QE}}(E_l,\theta)) + \frac{\mathrm{d}\sigma_{\mathrm{WN}}}{\mathrm{d}q^2}(E_l,\theta) f^Z(E_\nu^{\mathrm{WN}}(E_l,\theta)). \quad (4.2)$$

In Eqs. (4.1) and (4.2), the neutrino fluxes f^Y and f^Z are known, and the rates R^Y and R^Z are measured. We then have two equations for the two unknowns $\mathrm{d}\sigma_{\mathrm{QE}}/\mathrm{d}q^2$ and $\mathrm{d}\sigma_{\mathrm{WN}}/\mathrm{d}q^2$. We can solve for these two cross-sections if the two equations are linearly independent; that is, if

$$f^Y(E_\nu^{\mathrm{QE}}) f^Z(E_\nu^{\mathrm{WN}}) - f^Y(E_\nu^{\mathrm{WN}}) f^Z(E_\nu^{\mathrm{QE}}) \neq 0. \quad (4.3)$$

This condition is just the mathematical description for the requirement that

the neutrino fluxes be different. If condition (4.3) is fulfilled, we can solve for the unknowns:

$$\begin{aligned}
\frac{\mathrm{d}\sigma_{\mathrm{QE}}}{\mathrm{d}q^2}(E_l,\theta) &= \frac{R^Y f^Z(E_\nu^{\mathrm{WN}}) - R^Z f^Y(E_\nu^{\mathrm{WN}})}{f^Y(E_\nu^{\mathrm{QE}})f^Z(E_\nu^{\mathrm{WN}}) - f^Y(E_\nu^{\mathrm{WN}})f^Z(E_\nu^{\mathrm{QE}})} \\
\frac{\mathrm{d}\sigma_{\mathrm{WN}}}{\mathrm{d}q^2}(E_l,\theta) &= \frac{R^Z f^Y(E_\nu^{\mathrm{QE}}) - R^Y f^Z(E_\nu^{\mathrm{QE}})}{f^Y(E_\nu^{\mathrm{QE}})f^Z(E_\nu^{\mathrm{WN}}) - f^Y(E_\nu^{\mathrm{WN}})f^Z(E_\nu^{\mathrm{QE}})}.
\end{aligned} \tag{4.4}$$

This simple result is a bit astounding: it is possible to determine the differential cross-section, with the only model dependence being in the kinematics.

The above technique has one problem, though: since the neutrino cross-sections are so small, there will be very few events in each (E_l, θ) bin, unless the bins are large. This will affect the statistics, and finally put large uncertainties on the cross section measurements. In order to avoid these large statistical uncertainties, we will have to make the bins larger, or add up several small bins. That, however, will make it impossible to directly measure the differential cross-section as a function of momentum transfer. We have to re-introduce some model dependence to find the cross-sections. There can – and should – be free parameters in the models, though, that can then be fixed by experimental data. We can parameterize the differential cross-sections by $\mathrm{d}\sigma_{\mathrm{QE}}/\mathrm{d}q^2(E_l, \theta; \alpha_{\mathrm{QE}})$ and $\mathrm{d}\sigma_{\mathrm{WN}}/\mathrm{d}q^2(E_l, \theta; \alpha_{\mathrm{WN}})$, where α_{QE} and α_{WN} are the (multi-dimensional) parameters for the CCQE and CCWN models, respectively. Integrated over a range of (E_l, θ) values Ω, the rate is then given

by

$$\int_{\Omega} \mathrm{d}E_l \mathrm{d}\theta R^{Y,Z}(E_l,\theta) = \int_{\Omega} \mathrm{d}E_l \mathrm{d}\theta \left[\frac{\mathrm{d}\sigma_{\mathrm{QE}}}{\mathrm{d}q^2}(E_l,\theta;\alpha_{\mathrm{QE}}) f^{Y,Z}(E_\nu^{\mathrm{QE}}(E_l,\theta)) + \frac{\mathrm{d}\sigma_{\mathrm{WN}}}{\mathrm{d}q^2}(E_l,\theta;\alpha_{\mathrm{WN}}) f^{Y,Z}(E_\nu^{\mathrm{WN}}(E_l,\theta)) \right]. \quad (4.5)$$

If the total number of parameters (CCQE and CCWN combined) is equal to the number of (E_l,θ) ranges chosen, then the parameters can be uniquely determined. If the number of parameters is less than the number (E_l,θ) ranges, then the parameters can be determined in a best-fit way, leaving extra experimental data to determine the goodness of the fit, and therefore the goodness of the models underlying these calculations.

Thus far, we have considered the simplified models, that had one unique kinematic expression. Now we are ready to move to the more detailed models. While the analytic expressions for the neutrino energy as a function of E_l and θ become very lengthy for the CCQE case, we can in principle write E_ν as a function of E_l, θ and reaction parameters β_{QE} or β_{WN}. For quasi-elastic scattering, β_{QE} will be a three-dimensional parameter, for the three components of momentum that the initial nucleon can have. The case of whole-nucleus scattering is easier, since β_{WN} will be only one-dimensional, for the excitation energy that the recoiled target will have in the final state. To find the rate, we will then have to integrate over the reaction parameters:

$$\int_{\Omega} \mathrm{d}E_l \mathrm{d}\theta R^{Y,Z}(E_l,\theta) =$$

$$\int_\Omega \mathrm{d}E_l \mathrm{d}\theta \left[\int \mathrm{d}\beta_{\mathrm{QE}} \frac{\mathrm{d}\sigma_{\mathrm{QE}}}{\mathrm{d}q^2}(E_l, \theta; \alpha_{\mathrm{QE}}; \beta_{\mathrm{QE}}) f^{Y,Z}(E_\nu^{\mathrm{QE}}(E_l, \theta; \beta_{\mathrm{QE}})) \right.$$
$$\left. + \int \mathrm{d}\beta_{\mathrm{WN}} \frac{\mathrm{d}\sigma_{\mathrm{WN}}}{\mathrm{d}q^2}(E_l, \theta; \alpha_{\mathrm{WN}}; \beta_{\mathrm{WN}}) f^{Y,Z}(E_\nu^{\mathrm{WN}}(E_l, \theta; \beta_{\mathrm{WN}})) \right]. \quad (4.6)$$

In this equation, the differential cross-section depends on β through the q^2-dependence on β.

It should be mentioned that, since the reaction parameters are integrated out, the number of free parameters is not increased by the reaction parameters. The number of free parameters is therefore still determined only by the α_{QE} and α_{WN}.

4.2 Implementation

In this section we present how we have implemented the Multi-Beam Strategy in our $C++$ software, the source code of which can be found in the Appendix. The software is not optimized for ultimate speed, but that is not necessary, since it takes only a few seconds computing time on a personal computer per set of parameters $(\alpha_{\mathrm{QE}}, \alpha_{\mathrm{WN}})$. Even a relatively large range of parameters can be explored this way.

As previously noted, the integrand in Eq. (4.6) becomes very lengthy, and explicitly performing these integrals seemed like an overly hard way to proceed. We have therefore decided to perform the calculations as in an "event-generator": neutrinos are generated weighted by their spectrum. They undergo CCQE and CCWN reactions, and produce charged leptons in the final state. For each of these charged lepton events, the lepton four-vectors and a

production rate are recorded. Later, these events can be binned to whichever E_l, θ ranges are desired.

The spectra for the neutrino beams from the MiniBooNE experiment and the Los Alamos neutrino beam have been parameterized based on the spectra shown in Fig. 5.1 and Ref. [26], respectively. Flux modeling has been done for both the ν_e and ν_μ beams. The event generator then goes through these spectra at certain E_ν intervals. The neutrino flux at each energy will then be a multiplicative factor for the resulting events.

The next step is the nuclear reaction. This is relatively easy for the CCWN case, since the target is assumed to be initially at rest. First, the code checks whether there is enough energy to make the reaction happen. If the center-of-mass energy is greater or equal than the mass of the charged lepton and mass of the final nuclear state, then the reaction is allowed.

Boosting into the center-of-mass system makes the whole scattering process easier to handle. Once in the center-of-mass system, it is straight-forward to find the momentum p of the two products after the reaction:

$$p = \frac{\sqrt{\lambda\left(E_{\text{c.m.}}^2, m_l^2, (m_T + \varepsilon)^2\right)}}{2E_{\text{c.m.}}}, \tag{4.7}$$

where $\lambda(a, b, c) = a^2 + b^2 + c^2 - 2ab - 2ac - 2bc$ and $E_{\text{c.m.}}$ is the center-of-mass energy. Once the momentum is found, a scattering angle is assigned for the scattered lepton. Without loss of generality, the four-vector of the scattered lepton in the center-of-mass system is then $p_l^\mu = \left(\sqrt{m_l^2 + p^2}, p\cos\theta, p\sin\theta, 0\right)$. Several events will be created with $\cos\theta$ covering the whole range from -1 to

+1, and the differential cross-section for the reactions will be applied as

$$\frac{\mathrm{d}\sigma}{\mathrm{d}\cos\theta} = \frac{\mathrm{d}\sigma}{\mathrm{d}q^2}\frac{\mathrm{d}q^2}{\mathrm{d}\cos\theta}. \tag{4.8}$$

The last step is to boost the charged lepton back into the lab frame. The four-momentum of the charged lepton and the rate (flux multiplied by cross-section) are recorded in a file, to be binned and sorted at a later time. This whole process gets repeated for the different possible excitation energies for the recoiled target.

For the CCQE process, the procedure is a little bit more difficult than for the CCWN process, but in principle very similar. The four-vector for the nucleon in the initial state in the lab frame is $p_n^\mu = (E_n, p_n\cos\eta, p_n\sin\eta, 0)$. Here, $E_n = \sqrt{m_n^2 + p_n^2} + E_b$, taking care of the binding energy. The initial momenta range from 0 to the Fermi momentum p_F. After boosting into the center-of-mass frame, the charged lepton momentum will be distributed over the whole sphere: $p_l^\mu = \left(\sqrt{m_l^2 + p^2}, p\cos\theta, p\sin\theta\cos\phi, p\sin\theta\sin\phi\right)$. After assigning the cross-section and boosting back to the center-of-mass frame, the code checks whether the reaction is actually allowed to happen, or whether it is Pauli-blocked. If the reaction was allowed, then the charged lepton four-momentum and the production rate get recorded in the file.

Binning the events is then an easy procedure. The charged lepton four-vectors have been recorded, and are readily available to be binned into whichever bins have been chosen. The corresponding rates then just have to be added up.

Before we apply the Multi-Beam Strategy to real data in Chapter 6, we have a closer look at the neutrino experiments in the next Chapter.

Chapter 5

Neutrino Oscillation Experiments

Current experiments in our kinematic region of interest include the LSND and MiniBooNE experiments. Incidentally, the neutrino spectra of these two experiments are such that they complement one another perfectly for the Multi-Beam Strategy. In this Chapter, we will have a closer look at the experimental details of LSND and MiniBooNE.

5.1 The LSND Experiment

LSND [26] stands for the Liquid Scintillator Neutrino Detector developed for the experiment conducted at the Los Alamos National Laboratory. The experiment attempted to measure the oscillation parameter θ_{12} and the mass squared difference Δm^2_{12} for a relatively short oscillation length.

The experimental set-up was as follows: LSND used the proton beam at the Los Alamos Neutron Science Center, at an energy of 798 MeV. This beam was dumped in a beam stop, creating charged pions. A magnetic field filtered out the π^- and left only the π^+. The latter then decayed, either in flight or at rest. The decay products are ν_μ and μ^+. The muons then decay to positrons,

$\bar{\nu}_\mu$ and ν_e.

Thirty meters downstream, the neutrinos were detected by a detector filled with mineral oil ($^{12}CH_2$) and scintillator. The reaction that was sought was $\bar{\nu}_e + p \rightarrow e^+ + n$. Since there were no $\bar{\nu}_e$ in the original neutrino beam, any detection of $\bar{\nu}_e$ would be a sign of oscillations or other new physics.

And indeed, LSND observed 88 events consistent with a $\bar{\nu}_e$ in excess of the expected background. In order to get this number of events from neutrino oscillations, mass squared differences on the order of $0.2 - 10$ eV^2 would be needed, in strong disagreement with the values from all other neutrino oscillation experiments that we quoted in Section 1.6. This discrepancy between the oscillation parameters that could explain the LSND results and the oscillation parameters from other neutrino oscillation experiments demanded a new experiment that should confirm or reject the results seen in the LSND experiment.

5.2 MiniBooNE Experiment

This new experiment, which is located at the Fermi National Accelerator Laboratory, is called MiniBooNE [13]. MiniBooNE was designed to measure oscillation parameters equivalent to LSND, so a simultaneous scaling of neutrino energies and oscillation length was planned. Multiplying both the energy and the oscillation length by the same factor will yield no change in the argument of the cosine in Eq. (1.4).

For MiniBooNE, both the oscillation length and neutrino energy were in-

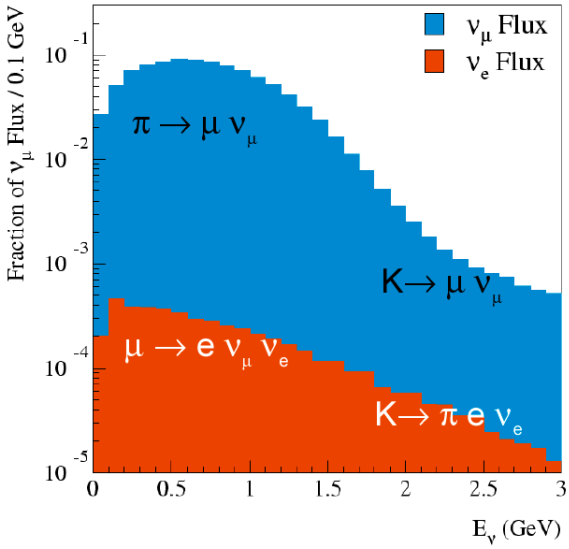


Figure 5.1: MiniBooNE neutrino flux, from Ref. [27]. The ν_μ and ν_e fluxes are shown, together with the dominant production channels.

creased, keeping E/L at about the same value that LSND claimed to measure. In order to produce the neutrinos, the 8 GeV Fermilab Booster proton beam was dumped on an Beryllium target. Charged pions and kaons were produced in that process. While being focused in a horn, a magnetic field separated out the negatively charged mesons. The positive pions and kaons decayed to ν_μ and μ^+, with the latter decaying to e^+, ν_e and $\bar{\nu}_\mu$. Since these decays happen at a forward boost, the flux of the primary ν_μ, which peak at around 700 MeV, is a lot greater than that of the secondary ν_e and $\bar{\nu}_\mu$. This is shown in Fig. 5.1.

The neutrinos then travel 500 m downstream before they hit the detector. The detector looks for the appearance of ν_e in the beam of mostly ν_μ. While there are some ν_e in the original beam, they will be treated as background. The neutrinos are detected via the reaction $\nu_l \rightarrow l^- + W^+$ for either lepton,

$l = e, \mu$. However, no measurements are taken on the target side.

The detector is a spherical tank of 800 t of mineral oil, $^{12}CH_2$, divided into an inner and an outer volume. There are photo multiplier tubes (PMTs) on the outside of the inner volume and around the outer volume. The outer volume just serves as a veto for any events that did not happen inside the inner volume, such as neutrino events happening outside the inner volume, or cosmic rays. As the charged lepton is produced and travels through the mineral oil, it creates an electromagnetic shower. This electromagnetic shower makes Cherenkov radiation, which is detected by the PMTs.

The experimental signatures to distinguish electrons from muons, and therefore electron neutrinos from muon neutrinos, are shown in Fig. 5.2. Muons, being highly penetrating, move through the tank on an essentially straight line. The circles of Cherenkov cones intersecting the photomultiplier tubes get smaller and smaller until they are filled. Electrons, on the other hand, have a much shorter radiation length. Therefore the electromagnetic shower usually starts and ends within the tank, creating only a Cherenkov circle on the PMTs, and not a filled circle. Also, since the electrons are lighter than the muons, they tend to have a zig-zaggy path, making the Cherenkov circle fuzzier than the muon circle.

Backgrounds to electron signals are plentiful. The greatest background comes from neutral pion decay. The π^0 get produced, for example, in neutral current reactions like the one shown in Fig. 5.3. They decay to two photons, each of which starts an electromagnetic shower similar to the one started by

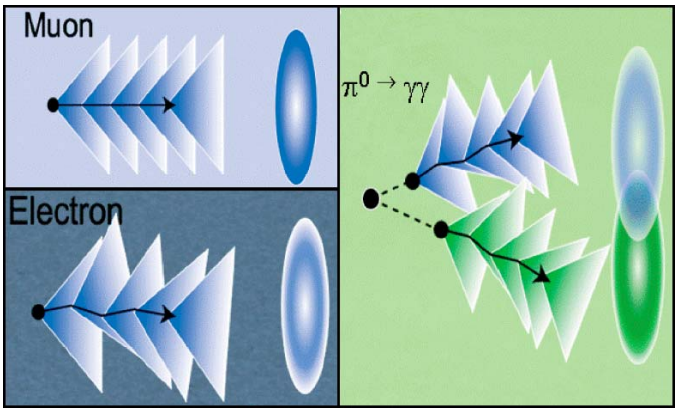


Figure 5.2: Electron, muon and π^0 signatures in the MiniBooNE detector, from Ref. [13].

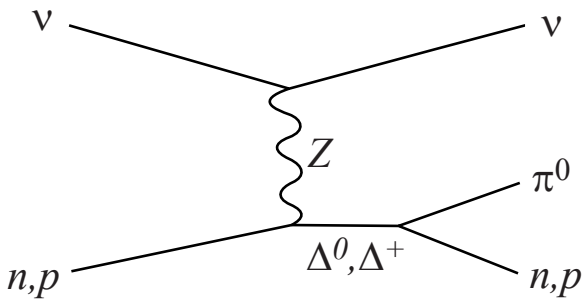


Figure 5.3: Neutral pion production by Δ resonances.

an electron. The $\pi^0 \rightarrow \gamma\gamma$ signal is also shown in Fig. 5.2. If one of the two photons gets absorbed, or goes undetected for whichever reason, only one photon remains, and its electromagnetic shower makes it look like an electron. Other backgrounds are from misidentified muons, neutral current ν_μ - electron scattering, and dirt events that escaped the outer tank veto.

5.3 MiniBooNE Analysis

For the analysis, the MiniBooNE collaboration assumed that the neutrino reaction was charged-current quasi-elastic scattering (CCQE), which we discussed in Section 2.3. To describe the nuclear reactions, MiniBooNE used the relativistic Fermi gas model, with the free parameters m_A (axial mass) and κ (Pauli-blocking parameter).

In order to fix the free parameters, the muon data was used. This is because there were many muon neutrinos in the beam, and a small neutrino oscillation (which is the most that would be expected) would not make a significant difference. The number of ν_μ events was plotted versus reconstructed momentum transfer, Q^2_{rec}. Then the values of the free parameters were varied to find the best agreement (least χ^2) of the data to a Monte Carlo calculation using these parameters. The results of this fit are shown in Fig. 5.4.

The values of the Pauli-blocking parameter and axial mass were then used to determine the expected background of electron neutrino events from the ν_e contamination in the beam.

5.4 MiniBooNE Results

The results from the MiniBooNE experiment were shown and published in 2007. At first, only data for reconstructed ν_e energies of greater or equal than 475 MeV was supposed to be presented, since this was the energy region that the MiniBooNE collaboration had confidence in. However, in the end, data all the way down to 200 MeV was released. The results of ν_e signals as a function

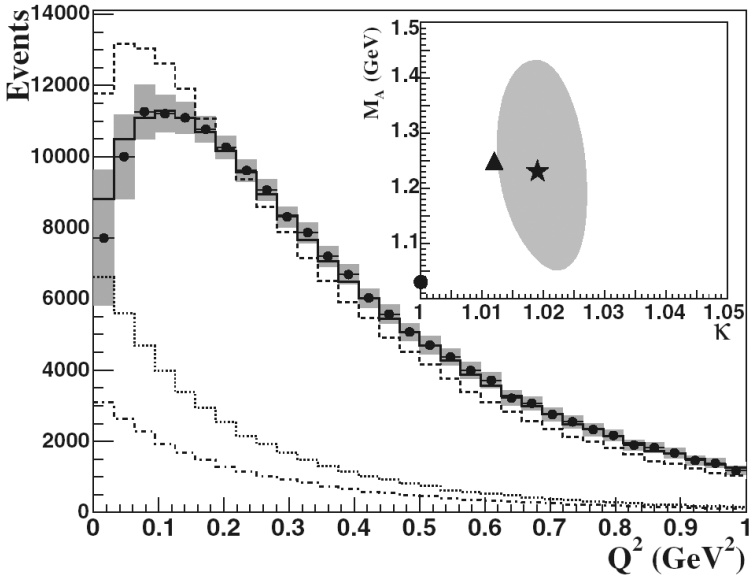


Figure 5.4: ν_μ events as a function of Q^2 in MiniBooNE, from Ref. [16]. The inlay shows the error ellipses for the parameters κ and m_A.

of reconstructed energy are shown in Fig. 5.5.

Above 475 MeV, the data agrees very well with the prediction for no neutrino oscillations (with the accepted neutrino oscillation parameters, almost no oscillations would be expected). This is in contradiction to the LSND experiment, which had seen oscillations with these L/E parameters.

However, below 475 MeV, an excess of ν_e - like events has been seen. Explaining this excess with neutrino oscillations would yield mass squared differences much higher than the accepted values, and with Δm^2_{21} explaining the < 475 MeV data, the ≥ 475 MeV data would turn out wrong.

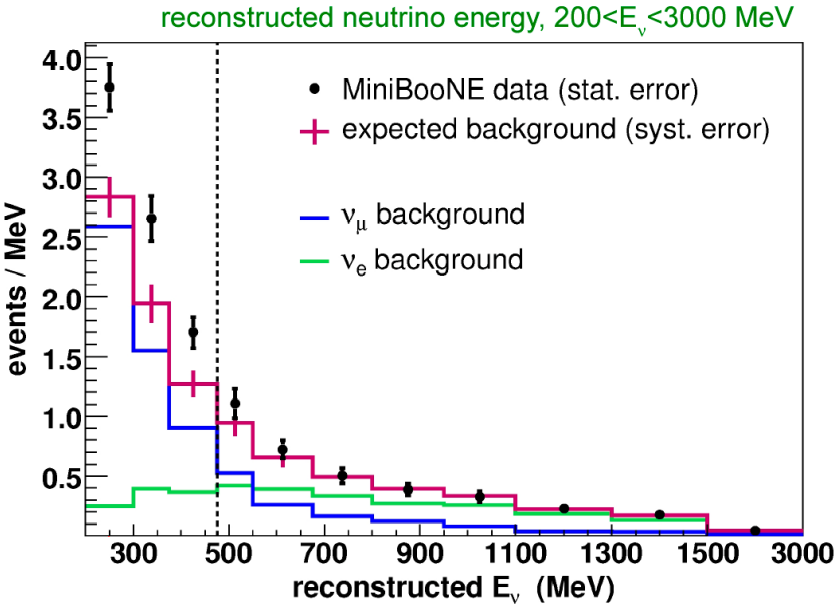


Figure 5.5: MiniBooNE ν_e events as a function of reconstructed neutrino energy, from Ref. [27].

5.5 Consequences

As a consequence, many attempts to explain the MiniBooNE low energy excess have been made. One of them is the idea of the existence of additional neutrinos, the so-called "sterile neutrinos" [28, 29, 30]. While sterile neutrinos do not interact via the weak interaction, they take part in the neutrino mixing. Therefore, e, μ, or τ neutrinos can oscillate to sterile neutrinos. "$3+1$" models, with only one sterile neutrino, have been ruled out beforehand, and "$3+2$" and "$3+3$" models could not explain the low energy excess either.

Other ideas include neutrinos that take short-cuts through extra dimensions [31], Lorentz-violating neutrinos [32, 33], and neutrinos that decay to final states with unparticles [34]. None of these ideas could convincingly explain the low energy excess, though. Usually, the parameters that would explain the MiniBooNE data would yield results in disagreement with other neutrino oscillation experiments.

There was also an attempt to explain the excess by including a process that had been overlooked by MiniBooNE: muon internal bremsstrahlung [35]. The MiniBooNE collaboration, however, replied that this effect would not be sufficient to explain the low energy excess [36].

In addition there is the question whether all the backgrounds have been correctly understood. In order to get the $\pi^0 \to \gamma\gamma$ background under control, a new experiment has been suggested: MicroBooNE [37]. It would use the same Fermilab booster proton beam and Beryllium target, but would use a Liquid Argon time projection chamber (LArTPC) as detector. The LArTPC

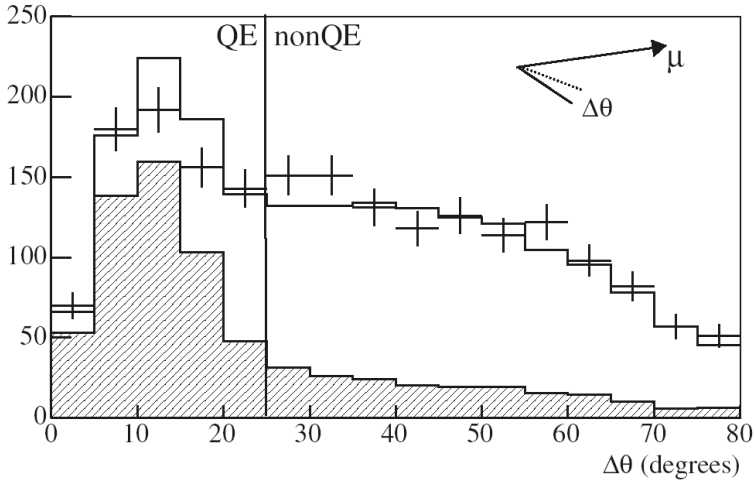


Figure 5.6: Distribution of the angular difference between the calculated and measured trajectory of the recoiled proton, from Ref. [38]. The shaded region is the CCQE prediction, and the cut-off to distinguish CCQE-events from non-CCQE-events is chosen to be 25°.

is claimed to be able to distinguish an electron induced electromagnetic shower from a photon induced shower. This would significantly reduce the background for low neutrino energies, therefore making for a better experiment.

Another experimental improvement could be made by measuring the recoiled target in the final state. Unfortunately, this seems to be technically unfeasible at the moment, at least within certain financial margins. The K2K experiment in Japan [38] is similar to MiniBooNE. K2K operates at somewhat higher neutrino energies, and uses ^{16}O as a target, instead of ^{12}C. Higher energies make it naturally easier to measure the recoil target, but even K2K's results, shown in Fig. 5.6, are not convincing. At even lower energies, the results will be even less significant.

Yet another improvement could be achieved if a monochromatic neutrino source was used: fixing the neutrino energy at a certain value would allow one to use the neutrino energy as an input for the analysis. However, in the current experimental setup, with a continuous neutrino spectrum, the neutrino energy has to be reconstructed from the other data, making it an output. Unfortunately, monochromatic neutrino beams are not readily available, at least not in the energy region at hand. One source of monochromatic electron (anti)neutrinos are nuclear reactions with electron (positron) capture. These are responsible, for example, for the ^7Be and *pep* lines in the solar neutrino spectrum. However, the energy for neutrinos like these is limited to a few MeV. A source of monochromatic muon neutrinos is the decay of monochromatic charged pions. The decay of pions at rest actually provided the 30 MeV neutrino line for the LSND experiment. To achieve higher neutrino energies, the pions would have to be boosted. While the charged pion lifetime should be long enough to allow a beam of charged pions to pass through a Wien velocity filter, for example, and filter out a certain pion energy from a pion energy spectrum, this filtering would greatly reduce the neutrino flux.

Finally, we have argued in Chapters 2 and 3 that both CCWN and CCQE processes ought to be taken into account. However, the MiniBooNE collaboration has neglected the charged-current whole-nucleus scattering. This is therefore an excellent opportunity to apply the Multi-Beam Strategy that we developed in Chapter 4. Our analysis and results are presented in the next Chapter.

Chapter 6

Analysis and Results

Here we re-analyze the MiniBooNE data with the computer code described in Section 4.2 and shown in the Appendix. In order to apply the Multi-Beam Strategy, we need data from a second neutrino beam. The spectrum from the neutrino beam at Los Alamos National Laboratory, which was used for the LSND experiment, is ideally suited to complement the spectrum from the Fermilab Booster beam, which was used for MiniBooNE.

In fact, we could just use the data taken by the LSND experiment and correct for the differences between the LSND and MiniBooNE detectors. However, there are two problems with this approach. First, the LSND collaboration has not published data that is binned such that it can be readily used for our analysis [26]. Second, correcting for the differences between the detectors would require detailed information about the detectors, and lies beyond the task of providing a proof of principle.

Therefore, we will use projected data in our analysis, assuming that the MiniBooNE detector took data in the Los Alamos neutrino beam. We will refer to this as "LosAlaBooNE".

We would also like to remind the reader of the limitations of our analysis that we discussed at the beginning of Chapter 4. Since information about detector acceptance is something only available to collaboration members, the results presented in this Chapter may not be the final numerical results. However, the proof of principle can be beautifully shown, and including detailed detector information in this analysis at a later time will just be a trivial complication.

6.1 CCQE only

The MiniBooNE data that we fit is the Q^2-data for muon neutrinos, shown in Fig. 5.4. This is the data chosen by the MiniBooNE collaboration, and has the advantage that there are plenty of events available, making the statistical errors minimal. The MiniBooNE collaboration has chosen this data to fit the axial mass m_A and the Pauli-blocking parameter κ. The results of this fit are shown in the inlay in Fig. 5.4. In addition to these two parameters, we also have one overall normalization α for the CCQE processes as a free parameter. This is necessary, to compensate for the lack of information about the detector.

The Q^2 in the above data is the "reconstructed Q^2", assuming that the underlying reaction was CCQE with the initial nucleon being at rest. Knowing the charged lepton energy and scattering angle, this can be easily calculated from $Q^2 = 2E_\nu(E_l - p_l \cos\theta) - m_l^2$, with the reconstructed neutrino energy from Eq. (3.10). We find the reconstructed momentum transfer when we bin our simulation data, being well aware that this has little to do with the real

Q^2 in the reaction.

First, we re-fit the MiniBooNE data with CCQE only. For each pair of m_A and κ values, we determine α to minimize χ^2. If we denote the MiniBooNE data vector by d, and our charged-current quasi-elastic simulation by $\mathrm{qe}(m_A, \kappa)$, then χ^2 is

$$\chi^2 = (\alpha\ \mathrm{qe}(m_A, \kappa) - d).w.(\alpha\ \mathrm{qe}(m_A, \kappa) - d), \tag{6.1}$$

where w is the weight-matrix, $w = \mathrm{diag}(1/\sigma_1^2, 1/\sigma_2^2, \ldots)$, and the σ_i are the uncertainties for the ith data-point. Minimizing χ^2 leads to

$$\alpha = \frac{\mathrm{qe}(m_A, \kappa).w.d}{\mathrm{qe}(m_A, \kappa).w.\mathrm{qe}(m_A, \kappa)}. \tag{6.2}$$

Using this value of α, we then find the χ^2 for the given parameters m_A and κ. We show our results for χ^2/dof in Fig. 6.1, where the number of degrees of freedom is 32 (the number of data points) minus the number of free parameters, 3. The best fit was found for $m_A = 1850$ MeV and $\kappa = 1.038$, with $\chi^2_{\mathrm{min}}/\mathrm{dof} = 0.1094$.

The very low values of χ^2/dof need to be discussed. They could indicate that the MiniBooNE collaboration has significantly over-estimated their uncertainties, but we want to give them the benefit of the doubt, and assume that the uncertainties are indeed correct. However, we are unable to tell to which degree the uncertainties are correlated. The correlated uncertainties would then have to be treated as such, and the overall χ^2/dof would turn out

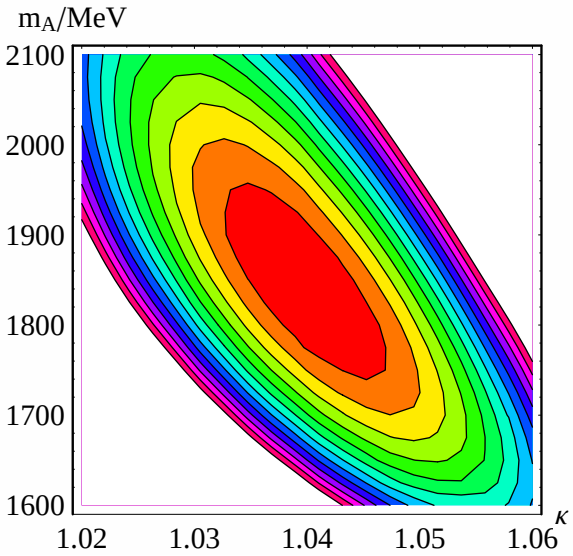


Figure 6.1: χ^2 contour plot for m_A and κ. The contour lines are $\chi^2_{\text{min}}/2$ apart from one another.

greater than found in our analysis.

The values found for m_A and κ differ significantly from the MiniBooNE analysis, which found $m_A = 1230$ MeV and $\kappa = 1.019$. Even though our result for κ differs by only 2% from the MiniBooNE result, this difference has to be considered significant, because the results are very sensitive to even small variations of κ, as can be seen in Fig. 6.1. The discrepancy in m_A and κ exists even though the same model has been used for the MiniBooNE analysis and our analysis. Assuming that the MiniBooNE analysis has been performed correctly, we have to attribute this discrepancy to our lack of information of detector details. In particular, the different acceptance of charged leptons for different energies can explain the difference between our and MiniBooNE's results: it is reasonable to assume that the acceptance is greater at greater

lepton energies, producing less events at lower energies and more events at higher energies. This means that MiniBooNE will detect less events at lower Q^2 and more events at higher Q^2 than our simulation would predict. In order to make up for this, we need to increase the Pauli blocking and the axial mass in our simulation, which is exactly the trend that we see.

6.2 CCQE and CCWN combined

Now let us add the charged-current whole-nucleus scattering to the analysis. This also introduces two new parameters. A^2 acts as an overall factor for CCWN, and b is related to the radius-squared of the nucleus, as shown in Section 2.4. This brings the total number of parameters to 5, and dof $= 32 - 5 = 27$. χ^2 is now

$$\chi^2 = (\alpha \text{ qe} + A^2 \text{ wn} - d).w.(\alpha \text{ qe} + A^2 \text{ wn} - d), \tag{6.3}$$

where wn stands for the CCWN simulation vector, and A^2 has been pulled out of the CCWN cross-section, since it is a linear variable and it is more convenient to handle it this way. Also, the parameter dependence of the simulation vectors is implied.

We can minimize χ^2 with respect to α and A^2:

$$\frac{\partial \chi^2}{\partial \alpha} = 0 = \frac{\partial \chi^2}{\partial A^2}, \tag{6.4}$$

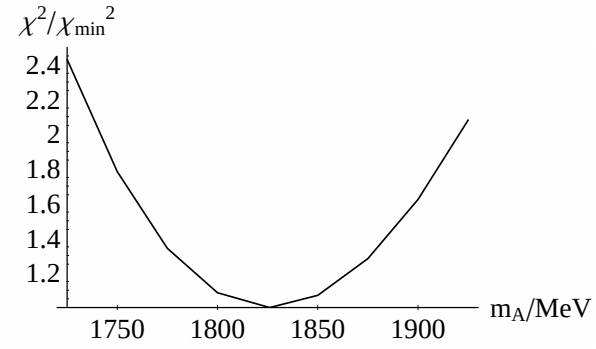
and then solve for the best-fit values of α and A^2:

$$\begin{aligned} \alpha &= \frac{\text{qe}.w.d\ \text{wn}.w.\text{wn} - \text{qe}.w.\text{wn}\ \text{wn}.w.d}{\text{qe}.w.\text{qe}\ \text{wn}.w.\text{wn} - \text{qe}.w.\text{wn}\ \text{wn}.w.\text{qe}} \\ A^2 &= \frac{\text{qe}.w.\text{qe}\ \text{wn}.w.d - \text{qe}.w.d\ \text{wn}.w.\text{qe}}{\text{qe}.w.\text{qe}\ \text{wn}.w.\text{wn} - \text{qe}.w.\text{wn}\ \text{wn}.w.\text{qe}}. \end{aligned} \tag{6.5}$$

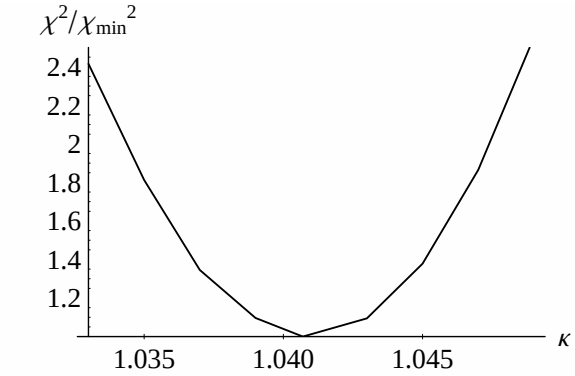
This leaves the three parameters m_A, κ and b to be fit in a parameter search. The least χ^2 was found for $m_A = 1826$ MeV, $\kappa = 1.0407$ and $b = 2.30\ 10^{-5}$ MeV^{-2}. For these values, $\chi^2_{\text{min}}/\text{dof} = 0.1031$, slightly better than without the CCWN contribution.

When we look at the Q^2-dependence of the CCWN contribution, we see that the slight improvement in $\chi^2_{\text{min}}/\text{dof}$ is actually a satisfactory result: compared to the CCQE contribution, the CCWN contribution is only numerically significant in the first two data bins. And since these two data bins have large uncertainties, we did not expect a huge improvement in $\chi^2_{\text{min}}/\text{dof}$ by including CCWN.

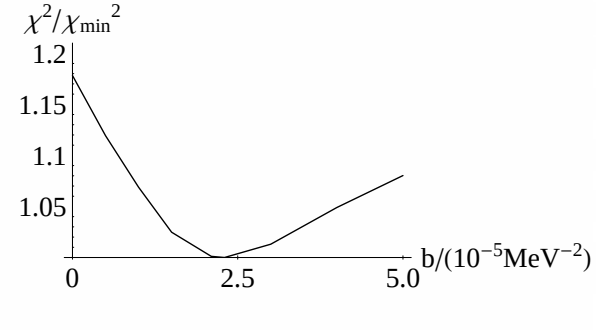
The dependence of χ^2 on each of the parameters m_A, κ, and b, with the other two being fixed, is shown in Fig. 6.2. While the plots for m_A and κ, in Figs. 6.2(a) and 6.2(b), respectively, look as one might have expected, the dependence of χ^2 on b, as shown in Fig. 6.2(c), is relatively small. This behavior is directly related to the relatively small improvement that we gained by including CCWN. χ^2 is already good without charged-current whole-nucleus scattering, and the worst possible χ^2 that could be obtained by varying b is the χ^2 with CCQE only.



(a) χ^2 plot for m_A.



(b) χ^2 plot for κ.



(c) χ^2 plot for b.

Figure 6.2: χ^2 plots for m_A, κ and b, while keeping the other two parameters fixed.

One combination of parameters that we find especially interesting are A^2 and κ. To show the $A^2 - \kappa$-plot, we keep m_A and b fixed at their best-fit values, and adjust α for the least χ^2 at each $A^2 - \kappa$-point. Ideally, we would also adjust m_A and b to yield the least χ^2, but that would greatly increase the computing time, while providing only limited additional insight. The resulting $A^2 - \kappa$-plot is shown in Fig. 6.3. The elongated error ellipses illustrate that there is a parameter degeneracy between A^2 and κ: increasing the Pauli-blocking parameter κ, and simultaneously increasing the CCWN factor yields little change in χ^2. This behavior also makes sense when looking at what the changes in these parameters mean. Increasing κ makes it harder for processes to happen when the final state nucleon has little momentum, which then implies that processes at low Q^2 are suppressed. CCWN, on the other hand, is dominant at low Q^2. Therefore, increasing the contribution of CCWN makes up for the events that were lost by increasing the Pauli-blocking parameter.

6.3 Multi-Beam Strategy

In order to resolve the above mentioned parameter degeneracy, we will use the Multi-Beam Strategy. To "create" the LosAlaBooNE data set, we use the best-fit values from the fit to the MiniBooNE data and run them with the Los Alamos neutrino spectrum. The error bars of the LosAlaBooNE data have been assumed to be equal to the square root of the value of the data points. Using the same procedure as outlined in the previous section, we find the A^2-κ

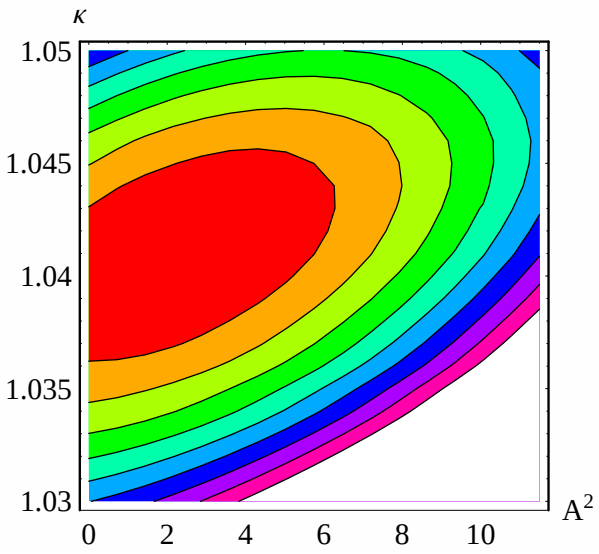


Figure 6.3: χ^2 contour plot for A^2 and κ for MiniBooNE. The contour lines are $\chi^2_{\rm min}/2$ apart from one another.

– χ^2-plot, Fig. 6.4.

This error ellipse shows us that LosAlaBooNE is indeed a perfect companion to MiniBooNE: the error ellipses of these two experiments are almost perpendicular to one another, allowing us to resolve the parameter degeneracy. A combined plot, with added χ^2 values, illustrates this and is shown in Fig. 6.5. By fixing free parameters to experimental values, we also effectively reduce the dependence on the particular models that are involved in the analysis.

6.4 MiniBooNE ν_e excess

Having determined the model parameters with the Multi-Beam Strategy, we now use these model parameters and re-analyze the MiniBooNE neutrino os-

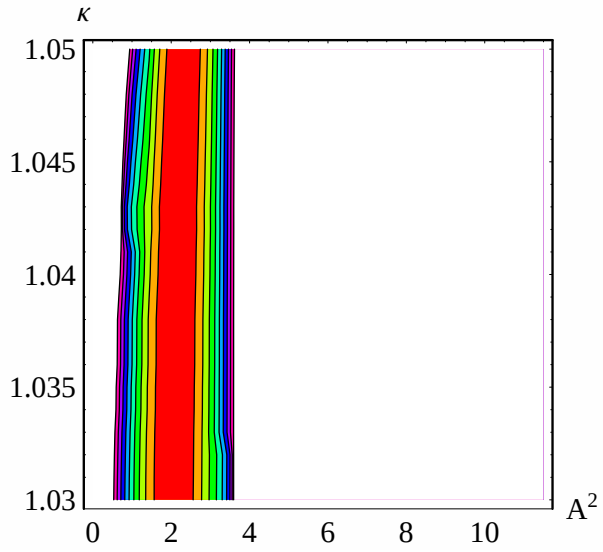


Figure 6.4: χ^2 contour plot for A^2 and κ for LosAlaBooNE. The contour lines are $\chi^2_{\text{min}}/2$ apart from one another.

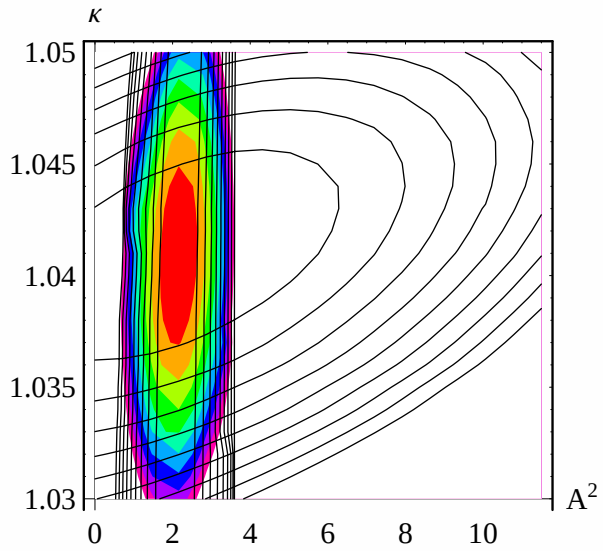


Figure 6.5: χ^2 contour plot for A^2 and κ. The colored areas show the combined χ^2, and the black contour lines are from the individual plots.

cillation data. To do so, the software is set to compute electron neutrinos, with the MiniBooNE electron neutrino flux. We bin the resulting data in reconstructed neutrino energy bins, so that we can compare it to the MiniBooNE results.

Unfortunately, we again face the different detector acceptance at different charged lepton energies. To take this into account, we proceed as follows: We first calculate our predictions for the electron neutrino events in the CCQE only case, with the parameters determined in Section 6.1. From these, we then determine scaling factors for every neutrino energy bin to match the MiniBooNE data. Then, we redo the calculation with CCQE and CCWN combined, with the parameters that we determined in Section 6.2. Once we have this, we apply the scaling factors that we had just determined. In Fig. 5.5, the new results would be barely distinguishable from the original graph: the data in the lowest energy bin is less than 10% greater than without including CCWN, and the difference in the other bins is smaller yet. We repeat this procedure using values for A and κ that are one standard deviation from the best fit, while maximizing A. One standard deviation is still perfectly reasonable – however, only a LosAlaBooNE experiment could tell. The results of this calculation are shown in Fig. 6.6.

We see that neither of the above calculations is able to explain the MiniBooNE low-energy ν_e excess. That rules CCWN out as a possible explanation for the MiniBooNE excess. While it is still possible that other processes or a misunderstanding of the detector caused the excess, this result brings us closer

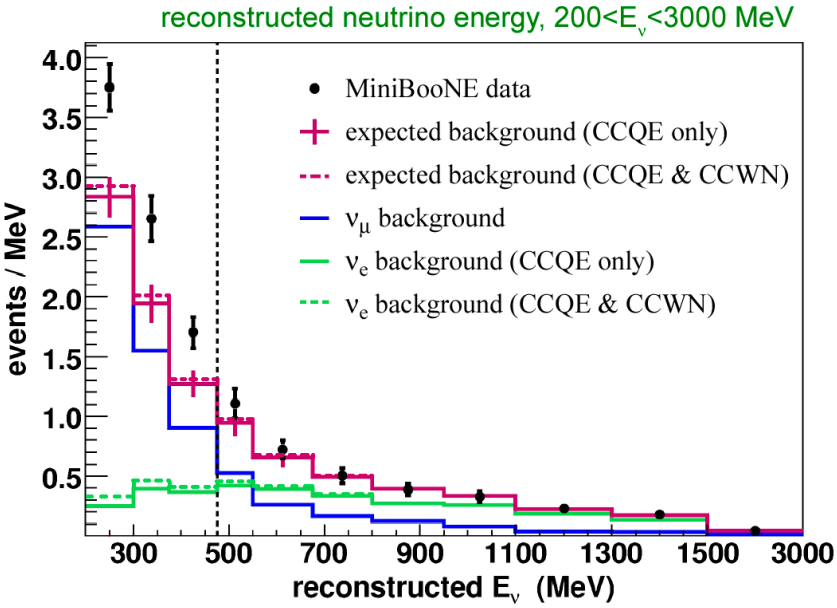


Figure 6.6: MiniBooNE ν_e events as a function of reconstructed neutrino energy; the results when including CCWN are shown as dashed lines.

to claiming that MiniBooNE has actually observed a signal of new physics beyond the standard neutrino oscillation model.

The main reason that the inclusion of CCWN cannot explain the MiniBooNE low energy excess is that the electron neutrino flux is small to begin with, and that electron neutrinos are responsible for only a relatively small fraction of the expected signal at low neutrino energies. So even though the inclusion of charged-current whole-nucleus scattering increases the real electron neutrino events by 30%, this makes only a small difference, since the background from other events that look ν_e-like is so large. A significant amount of these background events is due to misidentified photons, which would no longer be a problem for the proposed MicroBooNE experiment. Thus, in the MicroBooNE experiment, the expected signal would highly depend on the actual neutrino events, and the inclusion of CCWN will be essential to adequately describing the neutrino-nucleus cross-sections.

Chapter 7

Conclusions

A good understanding of neutrino-nucleus cross-sections is essential for the success of neutrino oscillation experiments. However, experimental groups have not yet paid adequate attention to these cross-sections. Instead, the trend seems to be that if one experiment produces data inconsistent with other data, a new experiment will be built. This has happened with LSND and MiniBooNE, and might happen again with MiniBooNE and MicroBooNE.

While we think that new experimental techniques are valuable, this practice of building new experiments without fixing the theoretical foundations is a waste of time and taxpayers' money. New experiments can only do so much without having a solid theoretical foundation. However, when a solid theoretical foundation is added to technically sound experiments, amazing results can happen.

In this dissertation, we have discussed the neutrino-nucleus cross-sections. They are among the most important parts of a neutrino experiment, yet they are one of the weakest links. Due to the lack of precise and correct nuclear models, a data-driven way to obtain the neutrino-nucleus cross-sections is in

order.

The Multi-Beam Strategy, as presented in Chapter 4 and applied in Chapter 6 provides an excellent approach to determining cross-sections from data. It does not require new experimental techniques, and can be easily implemented in the analysis of an experiment.

We predict that the future of neutrino experiments will rest on three pillars. First, technically sound and inventive experiments. The proposed MicroBooNE experiment, for example, seems to be a right step in this direction. By reducing backgrounds that were not very well known to begin with, the experimental data can be greatly improved.

Second, improved nuclear models that are actually used for the analysis. While there already exist models better than the relativistic Fermi gas model, for example, these models have to be used in the data analysis. And a continued effort on the theorists' side to develop robust and accurate models will be valuable.

Finally, there needs to be a connection between theory and experiment. It seems highly unlikely that nuclear models will be perfectly accurate and parameter-free in the near future. The values of the parameters will then have to be determined by experiments. Since there is a lack of monochromatic neutrino sources, the Multi-Beam Strategy presented in this dissertation can provide the needed link between theory and experiment.

Appendix A

The C++ Code

A.1 main.cpp

```
#include<fstream>
#include<iostream>
#include<stdlib>
#include<cstdlib>
#include<math.h>
#include"boonies.h"

using namespace std;

double sigmawn( double q2, double s, parameters &params )
// CCWN differential cross-section
{
  double ff = exp( params.wnff( ) * q2 );
  double enu = ( s - sq( massC ) ) / 2 / massC;
  return sq( ff ) * ( 4 + (q2 - sq( params.massLep( ))) * (2 /
          massC / enu + 1 / sq( enu ) - sq( params.massLep( )) /
          4 / sq( enu ) / sq( massC)) );
}

double gep( double t )
// Electric Proton Form Factor
{
  return ( 1. - 0.0578 * t ) / ( 1. + 11.1 * t + 13.6 *
          sq( t ) + 33. * t * sq( t ));
}
```

```
double gmp( double t )
// Magnetic Proton Form Factor
{
  return ( 1. + .15 * t ) / ( 1. + 11.1 * t + 19.6 * sq( t ) +
          7.54 * t * sq( t ) );
}

double gen( double t )
// Electric Neutron Form Factor
{
  return ( 1.25 * t + 1.3 * sq( t ) ) / ( 1 - 9.86 * t + 305 *
          sq( t ) - 758 * t * sq( t ) + 802 * sq( sq( t ) ) );
}

double gmn( double t )
// Magnetic Neutron Form Factor
{
  return ( 1. + 1.81 * t ) / ( 1 + 14.1 * t + 20.7 * sq( t ) +
          68.7 * t * sq( t ) );
}

double f1( double q2 )
// F1 Form Factor
{
  double t = - q2 / 4. / sq( massP );
  return 1. / ( 1. - q2 / 4. / sq( massP ) ) * ( gep( t ) -
          gen( t ) - q2 / 4. / sq( massP ) * ( gmp( t ) -
          gmn( t )));
}

double f2( double q2 )
// F2 Form Factor
{
  double t = - q2 / 4 / sq( massP );
  return 1. / ( 1. - q2 / 4. / sq( massP ) ) * ( gmp( t ) -
          gmn( t ) - gep( t ) + gen( t ));
}
```

```
double fa( double q2, parameters &params )
// Axial Form Factor
{
  return -1.272 / sq( 1. - q2 / params.maxial2( ));
}

double bigA( double q2, parameters &params )
// A in cross section equation
{
  return ( sq( params.massLep( )) - q2) / 4 / sq( massP ) *
          ( ( 4. - q2 / sq( massP ) ) * sq( fa( q2, params ) ) -
          ( 4. + q2 / sq( massP ) ) * sq( f1( q2 )) - q2 /
          sq( massP ) * sq( f2( q2 )) * ( 1. + q2 / 4 /
          sq( massP )) - 4 * q2 * f1( q2 ) * f2( q2 ) /
          sq( massP ) - sq( params.massLep( )) / sq( massP ) *
          ( sq( f1( q2 ) + f2( q2 )) + sq( fa( q2, params ))) );
}

double bigB( double q2, parameters &params )
// B in cross section equation
{
  return q2 / sq( massP ) * (( f1( q2 ) + f2( q2 )) *
          fa( q2, params ));
}

double bigC( double q2, parameters &params )
// C in cross section equation
{
  return .25 * ( sq( fa( q2, params )) + sq( f1( q2 )) - q2 /
          4 / sq( massP ) * sq( f2( q2 )));
}

double sigmaqe( double q2, double s, parameters &params )
// CCQE differential cross-section
{
  double enu2 = sq(( s - sq( massP ))) / 4. / sq( massP );
  double sminusu = 2 * s + q2 - 2 * sq( massP ) -
```

```
          sq( params.massLep( ));
  double dsigdq2 = sq( massP ) / enu2 * ( bigA( q2, params ) -
          bigB( q2, params ) * sminusu / sq( massP ) +
          bigC( q2, params ) * sq( sminusu / sq( massP )));
  return dsigdq2;
}

double minimuspec( double eneu )
// Muon spectrum for MiniBooNE beam
{
  return exp( 2.49 * sin( 1.479 * ( eneu + 375 ) / 1000 ) -
          4.873);
}

double minielespec( double eneu )
// Electron spectrum for MiniBooNE beam
{
  return 5.0e-4 * exp( - eneu / 1655 - sq( eneu ) / 5.1e6 );
}

double lsndmuspec( double eneu )
// Muon spectrum for LosAlaBooNE beam
{
  return ( eneu + 10 ) * exp( - sq( eneu + 10 ) / 10000 ) / 40;
}

double lsndelespec( double eneu )
// Electron spectrum for LosAlaBooNE beam
{
  return ( eneu + 10 ) * exp( - sq( eneu + 10 ) / 10000 ) /
          40000;
}

void wnevents( std::ofstream &outfile, double neue, parameters
      &params, double factor, double excite )
// CCWN scattering event generator
{
  boosty booster;
```

```
  double arg, pcm, cth, q2, events;
  fourvec neu( neue );                // Neutrino
  fourvec tar( massC, 0, 0, 0 );      // Target
  fourvec muon( params.massLep( ), 0, 0, 0 );
  tar.add( neu );                     // 'in'
  // check whether reaction works kinematically
  if( tar.minv( ) >= params.massLep( ) + massC + excite )
  {
    arg = lambda( sq( tar.minv( )), sq( params.massLep( )),
           sq( massC + excite ));
    pcm = sqrt( arg ) / 2 / tar.minv( );
    booster = tar.givebooster( );     // get boost of 'in'
    neu.cmboost( booster );           // boost neutrino to CM
    for( cth = - 1. + 1. / params.cthsteps( ); cth < 1; cth +=
        2. / params.cthsteps( ))
    {
      muon.newvals( params.massLep( ), pcm * cth, pcm *
                  sqrt( 1. - sq( cth )), 0 );
      q2 = sq( params.massLep( )) - 2. * neu.dot( muon );
      events = factor * sigmawn( q2, sq( tar.minv( )), params )
                * 2. * neu.givet( ) * pcm / params.cthsteps( );
      muon.labboost( booster );
      outfile << muon.givet( ) << '\t' << muon.givex( ) << '\t'
              << muon.givey( ) << '\t' << muon.givez( ) << '\t'
              << events << '\n';
    }
  }
}

void qeevents( ofstream &outfile, double neue, parameters
              &params, double factor, double pauli )
// CCQE scattering event generator
{
  boosty booster;
  double arg, pcm, cth, q2, events, pini;
  fourvec neu( neue );
  fourvec tar( massP, 0, 0, 0 );
  fourvec muon( params.massLep( ), 0, 0, 0 );
```

```
fourvec recoil( massP, 0, 0, 0 );
for( long inic = 0; inic < 5; inic++ )
{
  switch( inic )
  {
    case 0: pini =  96.5; break;
    case 1: pini = 146.7; break;
    case 2: pini = 174.3; break;
    case 3: pini = 195.2; break;
    case 4: pini = 212.3; break;
  }
  for( long i = 0; i < params.tarsteps( ); i++ )
  {
    neu.newvals( neue );            // Make neutrino
      // Make target
    tar.newvals( massP, pini * cos( (i + 1 / 2) * pi /
                params.tarsteps( ) ), pini * sin( (i + 1 / 2)
                * pi / params.tarsteps( ) ), 0 );
    tar.bind( eBind );              // Bind target
    tar.add( neu );                 // 'in'
    // Check whether reaction works
    if( tar.minv( ) >= params.massLep( ) + massP )
    {
      arg = lambda( sq( tar.minv( )), sq( params.massLep( )),
                  sq( massP ));
      pcm = sqrt( arg ) / 2 / tar.minv( );
      booster = tar.givebooster( );   // Find boost of 'in'
      neu.cmboost( booster );         // Boost neutrino to CM
      for( cth = - 1. + 1. / params.cthsteps( ); cth < 1; cth
          += 2. / params.cthsteps( ))
      {
        for( long j = 0; j < params.fisteps( ); j++ )
        {
         recoil.newvals( massP, - pcm * cth, - pcm * sqrt( 1.
                        - sq( cth )) * cos( ( j + 1 / 2) * pi
                        / params.fisteps( ) ),  - pcm * sqrt(
                        1. - sq( cth )) * sin( (j + 1 / 2) *
                        pi / params.fisteps( ) ) );
```

```
            recoil.labboost( booster );
            if( recoil.givet( ) > pauli * 963.45 )
            {
              muon.newvals( params.massLep( ), pcm * cth, pcm *
                            sqrt( 1. - sq( cth )) * cos( ( j + 1 /
                            2) * pi / params.fisteps( ) ),  pcm *
                            sqrt( 1. - sq( cth )) * sin( (j + 1 /
                            2) * pi / params.fisteps( ) ) );
              q2 = sq( params.massLep( )) - 2. * neu.dot( muon );
              events = factor * sigmaqe( q2, sq( tar.minv( )),
                         params ) * 2. * neu.givet( ) * pcm /
                         params.cthsteps( ) / 5 /
                         params.fisteps( ) / params.tarsteps( );
              muon.labboost( booster );
              outfile << muon.givet( ) << '\t' << muon.givex( )
                      << '\t' << muon.givey( ) << '\t'
                      << muon.givez( ) << '\t' << events << '\n';
            }
           }
         }
       }
     }
   }
}

void wngen( parameters &params )
// Outer structure for CCWN events
{
  ofstream wnfile( "wnevents.txt" );
  wnfile.precision( 8 );
  for( double excite = eExc; excite < params.maxcite( ); excite
      += params.stepcite( ) )
  {
    switch( params.enedist( ) )
    {
      case 0:
      case 3: wnevents( wnfile, params.enesteps( ), params, 1.,
                        excite );
```

```
                break;
        case 1: for( double energy = 100; energy < 3000; energy +=
                    params.enesteps( ) )
                  wnevents( wnfile, energy, params,
                            minimuspec( energy ), excite );
                break;
        case 2: for( double energy = 100; energy < 3000; energy +=
                    params.enesteps( ) )
                  wnevents( wnfile, energy, params,
                            lsndmuspec( energy ), excite );
                break;
        case 4: for( double energy = 100; energy < 3000; energy +=
                    params.enesteps( ) )
                  wnevents( wnfile, energy, params,
                            minielespec( energy ), excite );
                break;
        case 5: for( double energy = 100; energy < 3000; energy +=
                    params.enesteps( ) )
                  wnevents( wnfile, energy, params,
                            lsndelespec( energy ), excite );
                break;

    }
  }
}

void qegen( parameters &params, double pauli )
// Outer structure for CCQE events
{
  ofstream qefile( "qeevents.txt" );
  qefile.precision( 8 );
  switch( params.enedist( ) )
  {
    case 0:
    case 3: qeevents( qefile, params.enesteps( ), params, 1.,
                      pauli );
            break;
    case 1: for( double energy = 100; energy < 3000; energy +=
```

```
                params.enesteps( ) )
              qeevents( qefile, energy, params,
                        minimuspec( energy ), pauli );
            break;
    case 2: for( double energy = 100; energy < 3000; energy +=
                params.enesteps( ) )
              qeevents( qefile, energy, params,
                        lsndmuspec( energy ), pauli );
              break;
    case 4: for( double energy = 100; energy < 3000; energy +=
                params.enesteps( ) )
              qeevents( qefile, energy, params,
                        minielespec( energy ), pauli );
            break;
    case 5: for( double energy = 100; energy < 3000; energy +=
                params.enesteps( ) )
              qeevents( qefile, energy, params,
                        lsndelespec( energy ), pauli );
            break;

  }
}

void binnyq2( ifstream &infile, ofstream &q2write,
              parameters &params )
// Binning procedure for q2 bins
{
  double e, px, py, pz, p, ev, enurec, q2rec;
  long mybin;
  double bins[32];
  for( long i = 0; i < 32; i++ )
  {
    bins[i] = 0;
  }
  char line[100];
  char* rem;
  while( !infile.getline(line, 100).eof() )
  {
```

```
    e = strtod( line, &rem );
    px = strtod( rem, &rem );
    py = strtod( rem, &rem );
    pz = strtod( rem, &rem );
    ev = strtod( rem, NULL );
    enurec = ((massP + eBind) * e - (2 * massP * eBind +
             sq( eBind ) + sq( params.massLep( ))) / 2) /
             (massP + eBind - e + px);
    q2rec = - sq( params.massLep( )) + 2 * enurec * (e - px);
    mybin = 32. * q2rec / 1.e6;
    if( mybin >= 0 && mybin < 32 )
    {
      bins[mybin] += ev;
    }
  }
  for( long i = 0; i < 31; i++ )
  {
    q2write << bins[i] << "   ";
  }
  q2write << bins[31];
}

void binnye( ifstream &infile, ofstream &ewrite, parameters
            &params )
// Binning procedure for Energy bins
{
  double e, px, py, pz, ev, enurec;
  double bins[11];
  for( long i = 0; i < 11; i++ )
  {
    bins[i] = 0;
  }
  char line[100];
  char* rem;
  while( !infile.getline(line, 100).eof() )
  {
    e = strtod( line, &rem );
    px = strtod( rem, &rem );
```

```
py = strtod( rem, &rem );
pz = strtod( rem, &rem );
ev = strtod( rem, NULL );
enurec = ((massP + eBind) * e - (2 * massP * eBind +
         sq( eBind ) + sq( params.massLep( ))) / 2) /
         (massP + eBind - e + px);
if( enurec >= 200. )
{
  if( enurec < 300 )
  {
    bins[0] += ev;
  }
  else if( enurec < 375 )
  {
    bins[1] += ev;
  }
  else if( enurec < 475 )
  {
    bins[2] += ev;
  }
  else if( enurec < 550 )
  {
    bins[3] += ev;
  }
  else if( enurec < 675 )
  {
    bins[4] += ev;
  }
  else if( enurec < 800 )
  {
    bins[5] += ev;
  }
  else if( enurec < 950 )
  {
    bins[6] += ev;
  }
  else if( enurec < 1100 )
  {
```

```
        bins[7] += ev;
      }
      else if( enurec < 1300 )
      {
        bins[8] += ev;
      }
      else if( enurec < 1500 )
      {
        bins[9] += ev;
      }
      else if( enurec < 3000 )
      {
        bins[10] += ev;
      }
    }
  }
  for( long i = 0; i < 11; i++ )
  {
    ewrite << bins[i] << "   ";
  }
}


void qeloop( parameters &params )
// Loop for CCQE events with varying Pauli blocking parameter
{
  ofstream qeq2write( "pauliq2bins.txt" );
  for( double pauli = params.paulii( ); pauli <
      params.paulif( ); pauli += params.paulis( ) )
  {
    qegen( params, pauli );
    ifstream qeread( "qeevents.txt" );
    binnyq2( qeread, qeq2write, params );
    qeq2write << '\n';
    qeread.close( );
  }
}
```

```
void wnbinq2( parameters &params )
// Outer structure to bin CCWN events
{
  ifstream wnread( "wnevents.txt" );
  ofstream wnq2write( "wnq2bins.txt" );
  binnyq2( wnread, wnq2write, params );
}

void wnbine( parameters &params )
// Outer structure to bin CCWN events
{
  ifstream wnread( "wnevents.txt" );
  ofstream wnewrite( "wnebins.txt" );
  binnye( wnread, wnewrite, params );
}

void qebine( parameters &params )
// Outer structure to bin CCQE events
{
  ifstream qeread( "qeevents.txt" );
  ofstream qeewrite( "qeebins.txt" );
  binnye( qeread, qeewrite, params );
}

void qebinq2( parameters &params )
// Outer structure to bin incoherent events
{
  ifstream qeread( "qeevents.txt" );
  ofstream qeq2write( "qeq2bins.txt" );
  binnyq2( qeread, qeq2write, params );
}

int main( int argc, char* argv[] )
{
  long userinp;
  int argi = 0;
  parameters params;
  for(;;)
```

```
{
  if( argc == 1 )
  {
    cout << "What would you like to do?\n";
    cout << "(1) Generate CCWN events\n";
    cout << "(2) Bin CCWN events by rec. Q2\n";
    cout << "(3) Bin CCWN events by rec. E\n";
    cout << "(4) Generate CCQE events\n";
    cout << "(5) Bin CCQE events by rec. Q2\n";
    cout << "(6) Bin CCQE events by rec. E\n";
    cout << "(7) Run CCQE Pauli blocking loop\n";
    cout << "(9) Edit parameters\n";
    cout << "(0) Exit\n";
    cin >> userinp;
  }
  else
  {
    argi += 1;
    if( argi == argc )
      return 0;
    userinp = strtol( argv[argi], NULL, 0 );
  }
  switch( userinp )
  {
    case 0: return 0;
    case 1: wngen( params );
            break;
    case 2: wnbinq2( params );
            break;
    case 3: wnbine( params );
            break;
    case 4: qegen( params, params.paulii( ) );
            break;
    case 5: qebinq2( params );
            break;
    case 6: qebine( params );
            break;
    case 7: qeloop( params );
```

```
            break;
      case 9: params.editparams( );
            break;
    }
  }
}
```

A.2 boonies.h

```
using namespace std;

const double massE = 0.511;
const double massMu = 105.;
const double massP = 938.;
const double massC = 12 * massP;
const double eBind = - 34.;
const double eExc = 17.338;
const double pi = 3.1415926;

double sq( double x )
{
  return x * x;
}

struct boosty
// Boost and rotation variables: boost in x, rotation around z,
// around x
{
  double rotz1c, rotz1s, boostxc, boostxs, rotz2c, rotz2s;
};

class fourvec
// Four-vector class
{
  double t, x, y, z;

  public:
  fourvec( double enu );
```

```
  fourvec( double mass, double xx, double yy, double zz );
  void bind( double eb );
  double givet( );
  double givex( );
  double givey( );
  double givez( );
  double minv( );
  double givep( );
  double dot( fourvec ff );
  boosty givebooster( );
  void boostx( double ch, double sh );
  void rotz( double co, double si );
  void labboost( boosty booster );
  void cmboost( boosty booster );
  void add( fourvec ff );
  void newvals( double mass, double xx, double yy, double zz );
  void newvals( double enu );
};

fourvec::fourvec( double enu )
// Create a massless four-vector, momentum in x-direction
{
  t = enu;
  x = enu;
  y = 0;
  z = 0;
}

fourvec::fourvec( double mass, double xx, double yy, double zz )
// Create a four-vector, with given mass and 3-momentum
{
  t = sqrt( sq( mass ) + sq( xx ) + sq( yy ) + sq( zz ) );
  x = xx;
  y = yy;
  z = zz;
}

void fourvec::newvals( double enu )
```

```
// Renew values for massless four-vector
{
  t = enu;
  x = enu;
  y = 0;
  z = 0;
}

void fourvec::newvals( double mass, double xx, double yy,
                       double zz )
// Renew values for four-vector with mass
{
  t = sqrt( sq( mass ) + sq( xx ) + sq( yy ) + sq( zz ) );
  x = xx;
  y = yy;
  z = zz;
}

void fourvec::bind( double eb )
// Apply binding energy to four-vector
{
  t += eb;
}

double fourvec::givet( )
// Return t-component
{
  return t;
}

double fourvec::givex( )
// Return x-component
{
  return x;
}

double fourvec::givey( )
// Return y-component
```

```
{
  return y;
}

double fourvec::givez( )
// Return z-component
{
  return z;
}

void fourvec::boostx( double ch, double sh )
// Boost four-vector along x-axis
{
  double oldt = t;
  double oldx = x;
  t = ch * oldt + sh * oldx;
  x = sh * oldt + ch * oldx;
}

void fourvec::rotz( double co, double si )
// Rotate four-vector around z-axis
{
  double oldx = x;
  double oldy = y;
  x = co * oldx - si * oldy;
  y = si * oldx + co * oldy;
}

void fourvec::labboost( boosty booster )
{
// Boost four-vector from CM frame to lab frame
  this->rotz( booster.rotz1c, booster.rotz1s );
  this->boostx( booster.boostxc, booster.boostxs );
  this->rotz( booster.rotz2c, booster.rotz2s );
}

void fourvec::cmboost( boosty booster )
{
```

```
// Boost four-vector from lab frame to CM frame
  this->rotz( booster.rotz2c, - booster.rotz2s );
  this->boostx( booster.boostxc, - booster.boostxs );
  this->rotz( booster.rotz1c, - booster.rotz1s );
}

void fourvec::add( fourvec ff )
// Add four-vectors
{
  t += ff.givet( );
  x += ff.givex( );
  y += ff.givey( );
  z += ff.givez( );
}

double fourvec::minv( )
// Return invariant mass of a four-vector
{
  double arg = sq( t ) - sq( x ) - sq( y ) - sq( z );
  return sqrt( arg );
}

double fourvec::givep( )
// Return magnitude of 3-momentum of a four-vector
{
  return sqrt( sq( x ) + sq( y ) + sq( z ) );
}

double fourvec::dot( fourvec ff )
// Return the dot product
{
  return t * ff.givet( ) - x * ff.givex( ) - y * ff.givey( ) -
         z * ff.givez( );
}

boosty fourvec::givebooster( )
// Give the full boost and rotation variables that lead to a
// given four-vector
```

```
{
  boosty booster;
  booster.boostxc = t / this->minv( );
  booster.boostxs = this->givep( ) / this->minv( );
  booster.rotz2c = x / this->givep( );
  booster.rotz2s = y / this->givep( );
  double deno = sqrt( sq( booster.boostxc * booster.rotz2c -
               booster.boostxs ) + sq( booster.rotz2s ) );
  booster.rotz1c = (booster.boostxc * booster.rotz2c -
                    booster.boostxs) / deno;
  booster.rotz1s = - booster.rotz2s / deno;
  return booster;
}

class parameters
// Class that handles all the parameters
{
  long icthsteps, ifisteps, itarsteps, ienedist;
  double ienesteps, imaxcite, istepcite, ipaulii, ipaulif;
  double ipaulis, iwnff, imaxial;

  public:
  parameters( );
  void writeparams( );
  void editparams( );
  long cthsteps( );
  long fisteps( );
  long tarsteps( );
  long enedist( );
  double enesteps( );
  double maxcite( );
  double stepcite( );
  double paulii( );
  double paulif( );
  double paulis( );
  double massLep( );
  double wnff( );
  double maxial2( );
```

```
};

parameters::parameters( )
// Initialize parameters by reading from file
{
  ifstream params( "parameters.txt" );
  if( params.fail( ) == 0 )
  {
    char line[100];
    double getty;
    params.getline(line, 100);
    icthsteps = strtol( line, NULL, 0 );
    params.getline(line, 100);
    ifisteps = strtol( line, NULL, 0 );
    params.getline(line, 100);
    itarsteps = strtol( line, NULL, 0 );
    params.getline(line, 100);
    ienedist = strtol( line, NULL, 0 );
    params.getline(line, 100);
    ienesteps = strtod( line, NULL );
    params.getline(line, 100);
    imaxcite = strtod( line, NULL );
    params.getline(line, 100);
    istepcite = strtod( line, NULL );
    params.getline(line, 100);
    ipaulii = strtod( line, NULL );
    params.getline(line, 100);
    ipaulif = strtod( line, NULL );
    params.getline(line, 100);
    ipaulis = strtod( line, NULL );
    params.getline(line, 100);
    iwnff = strtod( line, NULL );
    params.getline(line, 100);
    imaxial = strtod( line, NULL );

  }
  else
  {
```

```
    icthsteps = 32;
    ifisteps = 8;
    itarsteps = 6;
    ienedist = 0;
    ienesteps = 400;
    imaxcite = eExc + 11.;
    istepcite = 2.1;
    ipaulii = 1.;
    ipaulif = 1.5;
    ipaulis = 0.05;
    iwnff = 32.;
    imaxial = 1000.0;
    this->writeparams( );
  }
}

void parameters::writeparams( )
// Write parameters to file
{
  ofstream para( "parameters.txt" );
  para << icthsteps << '\n' << ifisteps << '\n' << itarsteps
       << '\n' << ienedist << '\n' << ienesteps << '\n'
       << imaxcite << '\n' << istepcite << '\n' << ipaulii
       << '\n' << ipaulif << '\n' << ipaulis << '\n' << iwnff
       << '\n' << imaxial;
}

void parameters::editparams( )
// Edit Parameters
{
  long userinp, user2;
  long loopy = 0;
  while( loopy == 0 )
  {
    cout << "Please choose the parameter you want to edit\n";
    cout << "( 1) Cos theta steps ( " << icthsteps << " )\n";
    cout << "( 2) Phi steps ( " << ifisteps << " )\n";
    cout << "( 3) Target steps ( " << itarsteps << " )\n";
```

```
switch( ienedist )
{
  case 0: cout << "( 4) Monochromatic Muon Beam\n";
          cout << "( 5) Energy ( " << ienesteps << " )\n";
          break;
  case 1: cout << "( 4) MiniBooNE Muon Spectrum\n";
          cout << "( 5) Energy steps ( " << ienesteps
               << " )\n";
          break;
  case 2: cout << "( 4) LosAlaBooNE Muon Spectrum\n";
          cout << "( 5) Energy steps ( " << ienesteps
               << " )\n";
          break;
  case 3: cout << "( 4) Monochromatic Electron Beam\n";
          cout << "( 5) Energy ( " << ienesteps << " )\n";
          break;
  case 4: cout << "( 4) MiniBooNE Electron Spectrum\n";
          cout << "( 5) Energy steps ( " << ienesteps
               << " )\n";
          break;
  case 5: cout << "( 4) LosAlaBooNE Electron Spectrum\n";
          cout << "( 5) Energy steps ( " << ienesteps
               << " )\n";
          break;
}
cout << "( 6) Maximal Excitation Energy ( " << imaxcite
     << " )\n";
cout << "( 7) Excitation Energy Step ( " << istepcite
     << " )\n";
cout << "( 8) Pauli Blocking Parameter (initial for loop)( "
     << ipaulii << " )\n";
cout << "( 9) Pauli Blocking Parameter, final ( "
     << ipaulif << " )\n";
cout << "(10) Pauli Blocking Parameter, step ( " << ipaulis
     << " )\n";
cout << "(11) CCWN Form Factor ( " << iwnff << " )\n";
cout << "(12) Axial Mass ( " << imaxial << " )\n";
cout << "( 0) Done\n";
```

```
cin >> userinp;
switch( userinp )
{
  case 0: loopy = 1;
          break;
  case 1: cout << "Currently, Cos theta steps is "
               << icthsteps << '\n';
          cout << "Please enter the new value for Cos theta"
               << " steps: ";
          cin >> icthsteps;
          break;
  case 2: cout << "Currently, Phi steps is " << ifisteps
               << '\n';
          cout << "Please enter the new value for Phi "
               << "steps: ";
          cin >> ifisteps;
          break;
  case 3: cout << "Currently, Target steps is "
               << itarsteps << '\n';
          cout << "Please enter the new value for Target "
               << "steps: ";
          cin >> itarsteps;
          break;
  case 4: cout << "Please choose from the following:\n";
          cout << "(0) Monochromatic Muon Beam\n";
          cout << "(1) MiniBooNE Muon Spectrum\n";
          cout << "(2) LosAlaBooNE Muon Spectrum\n";
          cout << "(3) Monochromatic Electron Beam\n";
          cout << "(4) MiniBooNE Electron Spectrum\n";
          cout << "(5) LosAlaBooNE Electron Spectrum\n";
          cin >> ienedist;
          if( ienedist == 0 || ienedist == 3 )
            ienesteps = 400;
          else
            ienesteps = 25;
          break;
  case 5: if( ienedist == 0 || ienedist == 3 )
          {
```

```
          cout << "Currently, the monochromatic energy "
               << "is " << ienesteps << '\n';
          cout << "Please enter the new value for the "
               << "monochromatic energy: ";
        }
        else
        {
          cout << "Currently, the energy step size is "
               << ienesteps << '\n';
          cout << "Please enter the new value for the "
               << "energy step size: ";
        }
        cin >> ienesteps;
        break;
case 6: cout << "Currently, Maximal Excitation Energy is "
             << imaxcite << '\n';
        cout << "Please enter the new value for Maximal "
             << "Excitation Energy: ";
        cin >> imaxcite;
        break;
case 7: cout << "Currently, Excitation Energy Stepsize"
             << " is " << istepcite << '\n';
        cout << "Please enter the new value for "
             << "Excitation Energy Stepsize\n";
        cout << "(choose it greater than Max Exc Ene if "
             << "you only want the ground state): ";
        cin >> istepcite;
        break;
case 8: cout << "Currently, Pauli Blocking is "
             << ipaulii << '\n';
        cout << "(This is also the inital value for the"
             << " Pauli loop)\n";
        cout << "Please enter the new value for Pauli "
             << "Blocking: ";
        cin >> ipaulii;
        break;
case 9: cout << "Currently, the final value for the Pauli"
             << " Blocking Loop is " << ipaulif << '\n';
```

```
                cout << "Please enter the new value for the final"
                     << " PB parameter: ";
                cin >> ipaulif;
                break;
        case 10: cout << "Currently, the Pauli Blocking Step Size"
                      << " is " << ipaulis << '\n';
                 cout << "Please enter the new value for the PB "
                      << "step size: ";
                 cin >> ipaulis;
                 break;
        case 11: cout << "Currently, the CCWN Form Factor is "
                      << iwnff << '\n';
                 cout << "Please enter the new value for the CCWN"
                      << " form factor: ";
                 cin >> iwnff;
                 break;
        case 12: cout << "Currently, the Axial Mass is "
                      << imaxial << '\n';
                 cout << "Please enter the new value for the"
                      << " axial mass: ";
                 cin >> imaxial;
                 break;
      }
    }
    this->writeparams( );
}

long parameters::cthsteps( )
{
  return icthsteps;
}

long parameters::fisteps( )
{
  return ifisteps;
}

long parameters::tarsteps( )
```

```
{
  return itarsteps;
}

long parameters::enedist( )
{
  return ienedist;
}

double parameters::enesteps( )
{
  return ienesteps;
}

double parameters::maxcite( )
{
  return imaxcite;
}

double parameters::stepcite( )
{
  return istepcite;
}

double parameters::paulii( )
{
  return ipaulii;
}

double parameters::paulif( )
{
  return ipaulif;
}

double parameters::paulis( )
{
  return ipaulis;
}
```

```
double parameters::wnff( )
{
  return 1.0e-6 * iwnff;
}

double parameters::maxial2( )
{
  return sq( imaxial );
}

double parameters::massLep( )
{
  if( ienedist < 3 )
    return massMu;
  else
    return massE;
}

double eofp( double p, double m )
// Calculate energy as function of momentum and mass
{
  return sqrt( sq( p ) + sq( m ) );
}

double lambda( double x, double y, double z )
// The lambda function, l(x, y, z) = x^2 + y^2 + z^2 - 2xy -
// 2xz - 2yz
{
  return sq( x ) + sq( y ) + sq( z ) - 2 * x * y - 2 * x * z -
          2 * y * z;
}
```

Bibliography

[1] F. Jegerlehner, Acta Phys. Polon. B **38**, 3021 (2007) [arXiv:hep-ph/0703125].

[2] J. N. Bahcall and H. A. Bethe, Phys. Rev. Lett. **65**, 2233 (1990).

[3] Q. R. Ahmad *et al.* [SNO Collaboration], Phys. Rev. Lett. **89**, 011301 (2002) [arXiv:nucl-ex/0204008].

[4] Y. Fukuda *et al.* [Super-Kamiokande Collaboration], Phys. Rev. Lett. **81**, 1562 (1998) [arXiv:hep-ex/9807003].

[5] T. Araki *et al.* [KamLAND Collaboration], Phys. Rev. Lett. **94**, 081801 (2005) [arXiv:hep-ex/0406035].

[6] M. H. Ahn *et al.* [K2K Collaboration], Phys. Rev. Lett. **90**, 041801 (2003) [arXiv:hep-ex/0212007].

[7] Z. Maki, M. Nakagawa and S. Sakata, Prog. Theor. Phys. **28**, 870 (1962).

[8] S. Eidelman *et al.* [Particle Data Group], Phys. Lett. B **592**, 1 (2004).

[9] W.-M. Yao *et al.* [Particle Data Group], J. Phys. **G 33**, 1 (2006).

[10] V. D. Barger and R. J. N. Phillips, "*Collider Physics,*" Boulder, CO, USA: *Westview Press (1996) 592 p.*

[11] P. Lipari, M. Lusignoli and F. Sartogo, Phys. Rev. Lett. **74**, 4384 (1995) [arXiv:hep-ph/9411341].

[12] R. A. Smith and E. J. Moniz, Nucl. Phys. B **43**, 605 (1972) [Erratum-ibid. B **101**, 547 (1975)].

[13] A. A. Aguilar-Arevalo *et al.* [The MiniBooNE Collaboration], Phys. Rev. Lett. **98**, 231801 (2007) [arXiv:0704.1500 [hep-ex]].

[14] C. H. Llewellyn Smith, Phys. Rept. **3**, 261 (1972).

[15] R. Bradford, A. Bodek, H. Budd and J. Arrington, Nucl. Phys. Proc. Suppl. **159**, 127 (2006) [arXiv:hep-ex/0602017].

[16] A. A. Aguilar-Arevalo *et al.* [MiniBooNE Collaboration], Phys. Rev. Lett. **100**, 032301 (2008) [arXiv:0706.0926 [hep-ex]].

[17] E. Kolbe, K. Langanke and P. Vogel, Nucl. Phys. A **652**, 91 (1999) [arXiv:nucl-th/9903022].

[18] J. Engel, E. Kolbe, K. Langanke and P. Vogel, Phys. Rev. C **54**, 2740 (1996) [arXiv:nucl-th/9606031].

[19] S. L. Mintz and M. Pourkaviani, Phys. Rev. C **40**, 2458 (1989).

[20] M. Fukugita, Y. Kohyama and K. Kubodera, Phys. Lett. B **212**, 139 (1988).

[21] J. S. O' Connell, T. W. Donnelly and J. D. Walecka, Phys. Rev. C **6**, 719 (1972).

[22] S. K. Singh and E. Oset, Phys. Rev. C **48**, 1246 (1993).

[23] K. L. G. Heyde, "*The Nuclear Shell Model,*" New York, NY, USA: *Springer-Verlag (1994) 438 p.*

[24] E. Kolbe, F. K. Thielemann, K. Langanke and P. Vogel, Phys. Rev. C **52**, 3437 (1995).

[25] G. Audi and A. H. Wapstra, Nucl. Phys. A **595**, 409 (1995).

[26] A. Aguilar *et al.* [LSND Collaboration], Phys. Rev. D **64**, 112007 (2001) [arXiv:hep-ex/0104049].

[27] R. Tayloe, "*Neutrino Oscillation results from MiniBooNE,*" unpublished talk (2007).

[28] M. Maltoni and T. Schwetz, Phys. Rev. D **76**, 093005 (2007) [arXiv:0705.0107 [hep-ph]].

[29] S. Goswami and W. Rodejohann, JHEP **0710**, 073 (2007) [arXiv:0706.1462 [hep-ph]].

[30] G. Drexlin [KATRIN Collaboration], Nucl. Phys. Proc. Suppl. **145**, 263 (2005).

[31] H. Pas, S. Pakvasa and T. J. Weiler, Phys. Rev. D **72**, 095017 (2005) [arXiv:hep-ph/0504096].

[32] T. Katori, V. A. Kostelecky and R. Tayloe, Phys. Rev. D **74**, 105009 (2006) [arXiv:hep-ph/0606154].

[33] A. de Gouvea and Y. Grossman, Phys. Rev. D **74**, 093008 (2006) [arXiv:hep-ph/0602237].

[34] X. Q. Li, Y. Liu and Z. T. Wei, arXiv:0707.2285 [hep-ph].

[35] A. Bodek, arXiv:0709.4004 [hep-ex].

[36] A. A. Aguilar-Arevalo *et al.* [MiniBooNE Collaboration], arXiv:0710.3897 [hep-ex].

[37] H. Chen *et al.* [MicroBooNE Collaboration], "*A Proposal for a New Experiment Using the Booster and NuMI Neutrino Beamlines: MicroBooNE,*" unpublished proposal (2007).

[38] R. Gran *et al.* [K2K Collaboration], Phys. Rev. D **74**, 052002 (2006) [arXiv:hep-ex/0603034].