Joint Institute for Nuclear Research

Analysis and interactive visualization of neutrino event topologies registered in the OPERA experiment.

International Remote Student Training at JINR

Instructor

Dr. Sergey Dmitrievsky

Student

Ibrahim Mohamed

Dec, 2023

Contents

Abstract

This research project is dedicated to analysis and visualization of neutrino events recorded in OPERA experiment, which was aimed to direct observation of ν_τ appearance within a pure ν_μ beam. This achievement was accomplished through the detection of short-lived tau leptons produced in ν_τ charged-current (CC) interactions. The work on this project was structured around three core objectives. Firstly, a thoughtful reading of three OPERA papers led to the successful replication of their results. This involved reproducing the decay length distribution of charmed hadrons resulting from the decay of ν_μ particles at the primary vertex, and determining the impact parameters for their daughter particles resulting from decay at the secondary vertex. The second task involved reproducing the track multiplicity distributions and two-dimensional slope distributions of the produced muons. Lastly, with collaborative guidance from the project supervisor, the code for visualizing neutrino events was restored, which made it possible to correctly display all tracks of particles participating in the event relative to one point in space (i.e., the primary vertex of neutrino interaction).

About the Experiment

2.1 OPERA Experiment

The main goal of the Oscillation Project with Emulsion-tRacking Apparatus (OPERA) experiment was to directly detect tau neutrinos produced in $\nu_\mu \to \nu_\tau$ oscillations. The experiment was a collaboration between CERN in Geneva, Switzerland, and the Laboratori Nazionali del Gran Sasso (LNGS) in Gran Sasso, Italy and used the CERN Neutrinos to Gran Sasso (CNGS) neutrino beam. The process started with protons from the Super Proton Synchrotron (SPS) at CERN being fired in pulses at a carbon target to produce pions and kaons. These particles decayed to produce muons and neutrinos. The data used in our analysis was fetched from CERN Open Data Portal. The OPERA detector, was located in Hall C of the Gran Sasso underground laboratory and was built earlier in 2003–2008. The topologies of neutrino events were reconstructed in "bricks" of photographic films (nuclear emulsion) interleaved with lead plates. Each brick weighed 8.3 kg. The two OPERA supermodules contained 150,000 such bricks arranged in parallel walls interspersed with X-Y planes of plastic scintillator strips. Each supermodule is followed by a magnetic spectrometer for momentum and charge identification of penetrating particles. During data collection, a neutrino interaction and its corresponding brick were tagged in real time by the scintillators and spectrometers. These bricks were extracted from the walls asynchronously with respect to the beam for film development, scanning and for the topological and kinematic search of tau decays.

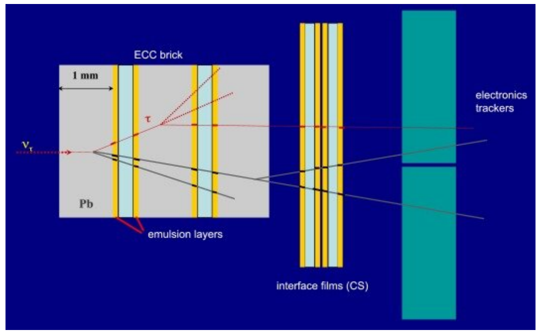

Figure 2.1: The structure of the brick

2.2 Neutrinos

Neutrino oscillation is a quantum mechanical phenomenon in which a neutrino created with a specific lepton family number ("lepton flavor": electron, muon, or tau) can later be measured to have a different lepton family number. The probability of measuring a particular neutrino flavor varies as a sine-like function as the neutrino propagates through space, so to reliably detect oscillations, it is necessary to correctly select the distance between the neutrino source and the detector.

My Work

I have been keeping a track of all the materials, papers, and code that I have worked on in a Drive folder[1], in addition to keeping a local version.

3.1 Task 1

In this task, my work involved reproducing results of decay length and impact parameter distributions through 50 events of ν_μ decay. We call the point at which the decay happens the primary vertex. The decay length of interest is the maximum distance the charmed hardon (the mother particle) takes before decaying again at the secondary vertex. The impact parameter of the daughter particle is the distance between the primary vertex and the daughter's path.

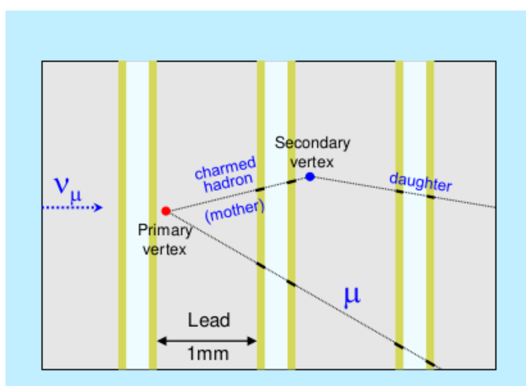

Figure 3.1: The structure of the ν_μ event

3.1.1 Calculations

To find the decay length between the primary vertex at $P_1 = (x_1, y_1, z_1)$ and the secondary vertex at $P_2 = (x_2, y_2, z_2)$, one can use the Pythagoras theorem as follows

[1]Link: `https://drive.google.com/drive/folders/1e4RS10EMUdL52ugW5rtQsA25nvXChiFt?usp=drive_link`

$$d_{decay} = dist(P_1, P_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \tag{3.1}$$

To find the impact parameter (ip), one should find the distance between the the primary vertex at P_1 and the line $\vec{L}(t)$ extended from the line segment between the secondary vertex at P_2 and the daughter particle at P_3. Hence we have

$$\vec{L}(t) = P_2 + (P_3 - P_2)t \tag{3.2}$$

where $t \in R$. Taking the cross product and manipulating the terms, one can write

$$d_{ip}|P_3 - P_2| = |(P_3 - P_2) \times (P_2 - P_1)| \tag{3.3}$$

$$d_{ip} = \frac{|(P_3 - P_2) \times (P_2 - P_1)|}{|P_3 - P_2|} \tag{3.4}$$

3.1.2 Code

Here is an overview of the general structure of my code for this task [2]. It took me a lot of time to produce and test it, because I was learning C++ as I was working on the problem, and because testing the code via Windows Subsystem for Linux (WSL), needs patience for the time it takes and the unintelligiblity of the error messages.

```cpp
#include <TMath.h>
#include <cmath>
#include <iostream>
#include <fstream>
#include <filesystem>
#include <string>
#include <TCanvas.h>
#include <TH1F.h>

std::vector<double> split_string_vectorDoubles(std::string str){…}

void displayVectorDoubles(std::vector<double> numbers){…}

// function to read data
std::vector<std::vector<double>> readFile(std::string path){…}

// functions to compute quantities
double decayLength(std::vector<std::vector<double>> vertices){…}

double impactParameter(std::vector<double> primaryVertex, std::vector<double> daughterTrackLine){…}

std::vector<double> impactParamters(std::vector<std::vector<double>> vertices, std::vector<std::vector<double>> trackLines){…}

// function to read a whole folder
std::vector<std::string> getFilesInFolder(const std::string& folderPath) {…}

// function to group files by ID
std::vector<std::vector<std::string>> groupFilesById(const std::vector<std::string>& filenames) {…}
```

Figure 3.2: General structure of task 1 code, part 1

[2] It can be found in the Drive link mentioned previously in task 1>My Work>main.cpp

```cpp
std::vector<std::vector<std::string>> groupFilesById(const std::vector<std::string>& filenames) {...}

// function that draws histograms
void drawHistogramIPs(const std::vector<double>& data) {...}

void drawHistogramDLs(const std::vector<double>& data) {...}

int main(){
  std::vector<std::string> files = getFilesInFolder("data");
  std::vector<std::vector<string>> filesOfEachId = groupFilesById(files);

  std::vector<double> allImpactParameters;
  std::vector<double> allDecayLengths;
  for(const std::vector<string>& filesOfAnId : filesOfEachId){...}

  // Draw the histograms
  drawHistogramIPs(allImpactParameters);
  drawHistogramDLs(allDecayLengths);

  // Wait for user input before exiting (optional)
  std::cout << "Press enter to exit." << std::endl;
  std::cin.get();
  return 0;
}

```

Figure 3.3: General structure of task 1 code, part 2

3.1.3 Results

I saved my results to histograms and they are shown below, and after them are the results reported in the paper [3]. The results obtained on the distribution of decay lengths are in good agreement with those reported in the paper, however, there is a slight discrepancy (at the level of $\tilde{5}$%) in the distribution of impact parameters. I have discussed that deviation with the professor, and he noted that the bricks gets scanned at different labs and that might cause a slight discrepancy. I also think that the measurement of the daughter tracks is more noisy than that of the secondary vertex, and hence more disagreement is expected.

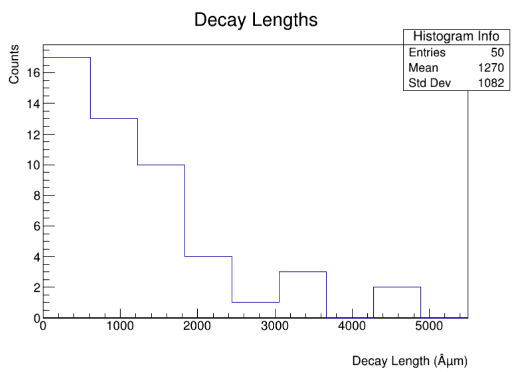

Figure 3.4: My results of the decay length distribution

[3] Agafonova, N. et al. (2014) "Procedure for short-lived particle detection in the OPERA experiment and its application to charm decays," The European physical journal. C, Particles and fields, 74(8). doi: 10.1140/epjc/s10052-014-2986-0.

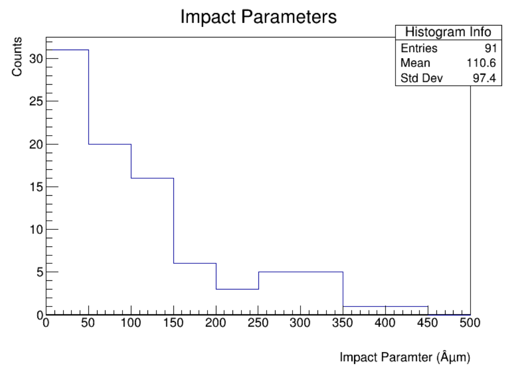

Figure 3.5: My results of the impact parameter distribution

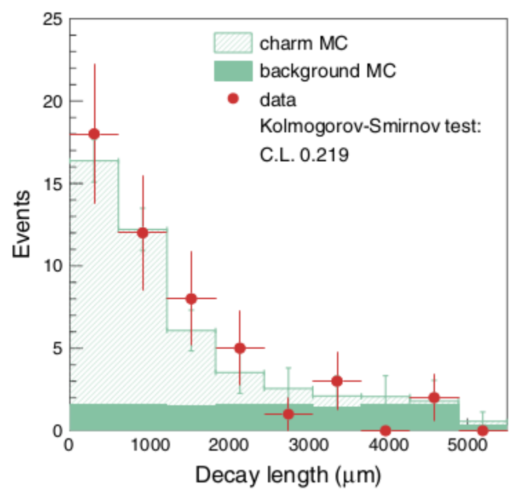

Figure 3.6: Paper's results of the decay length distribution

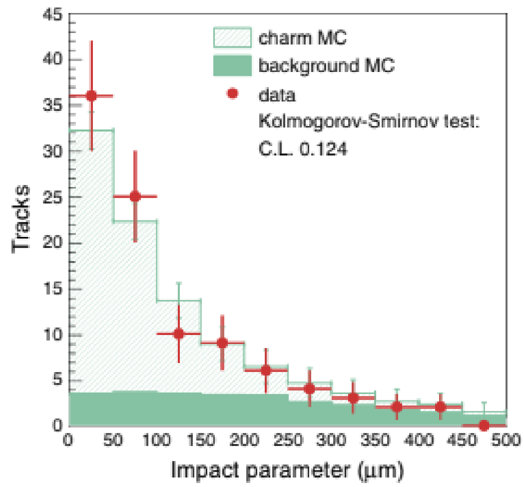

Figure 3.7: Paper's results of the impact parameter distribution

3.2 Task 2

In this task, my work involved reproducing [4] distribution of track multiplicity through 817 events of ν_μ decay, in addition to the two-dimensional distribution of the XZ and YZ slopes, which helps understanding the probability of specific topologies to occur. Track multiplicity of a vertex is the number of charged particle tracks associated with the vertex.

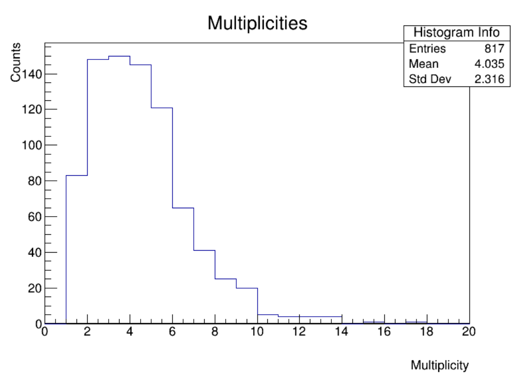

Figure 3.8: My results of the multiplicity distribution

[4]Agafonova, N. et al. (2018) "Study of charged hadron multiplicities in charged-current neutrino–lead interactions in the OPERA detector," The European physical journal. C, Particles and fields, 78(1). doi: 10.1140/epjc/s10052-017-5509-y.

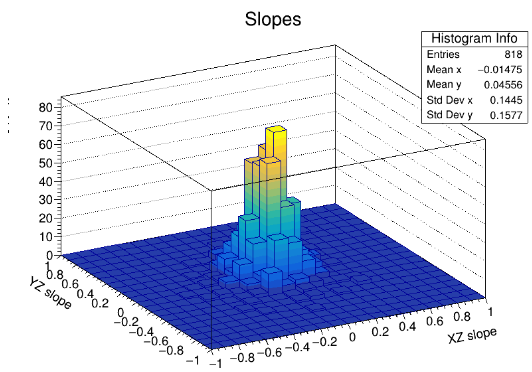

Figure 3.9: My results of the slopes distribution

3.3 Task 3

In this task, we used the THREE.js library in an HTML|CSS|JS project, where we tried to visualize the particle's tracks of the 10 neutrino events that are candidates for the production of a ν_τ particle. Below are a few examples of the interesting event topologies reconstructed in nuclear emulsions, in agreement with those that appear in the paper[5].

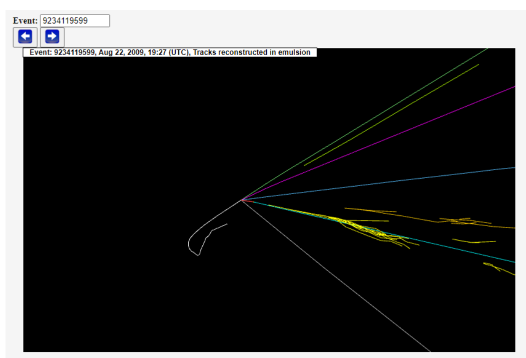

Figure 3.10: The ν_τ candidate event 9234119599

[5] Agafonova, N. et al. (2021) "OPERA tau neutrino charged current interactions," Scientific data, 8(1), pp. 1–18. doi: 10.1038/s41597-021-00991-y.

Figure 3.11: The ν_τ candidate event 11113019758

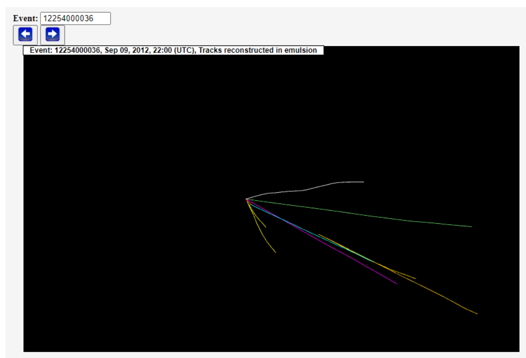

Figure 3.12: The ν_τ candidate event 12254000036

Conclusion and Acknowledgment

In conclusion, the successful execution of this research project on the Analysis and Visualization of Neutrino Event Topologies registered in the OPERA experiment has been both intellectually challenging and rewarding. The encountered technical difficulties, such as the need to develop a substantial amount of code from scratch and the nuanced integration of Linux and Ubuntu in a Windows environment, demanded resourcefulness and perseverance. Furthermore, the transition from a familiar programming landscape of Python and JS to the intricacies of C++ presented a valuable learning opportunity. Reproducing the results of scientific papers regarding decay length distributions, impact parameters, multiplicity and track slopes was a good way to become familiar with what variables are detected by particle detectors, and in particular what parameters physicists are interested in when reconstructing and analyzing the topology of neutrino events using nuclear emulsion.

I extend my sincere gratitude to Professor Sergey Dmitrievsky for his support. I also express my appreciation to the Joint Institute for Nuclear Research (JINR) and the organizers of the INTEREST program for providing a conducive environment for scientific exploration.