FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

BACHELOR THESIS

Martin Vavřík

Simulation and Reconstruction of Charged Particle Trajectories in an Atypic Time Projection Chamber

Institute of Particle and Nuclear Physics

Supervisor of the bachelor thesis: Mgr. Tomáš Sýkora, Ph.D.
Study programme: Physics
Study branch: Physics

Prague 2023

I declare that I carried out this bachelor thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In .... date ....

Author's signature

Dedication.

Title: Simulation and Reconstruction of Charged Particle Trajectories in an Atypic Time Projection Chamber

Author: Martin Vavřík

Institute: Institute of Particle and Nuclear Physics

Supervisor: Mgr. Tomáš Sýkora, Ph.D., Institute of Particle and Nuclear Physics

Abstract: Abstract.

Keywords: key words

Contents

Introduction

Time Projection Chamber (TPC) is a type of gaseous detector that detects charged particle trajectories by measuring the position and drift time of ions created in the gas (details are given in section 1). The energy of such particles can be determined thanks to the curvature of their trajectory in the magnetic field.

The goal of this thesis is to develop an algorithm for the reconstruction of charged particle trajectory and energy in an atypic TPC (with orthogonal electric and magnetic fields, i.e. Orthogonal Fields TPC (OFTPC)) used in the X17 project in Institute of Experimental and Applied Physics, Czech Technical University in Prague (IEAP CTU). Furthermore, we present the results of testing this algorithm with different samples of simulated data. In the future, we also wish to test this algorithm by measuring real particles with known energy distribution. In order to achieve this, we use the Garfield++ toolkit [1] in combination with the ROOT framework [2]. We run some of our more demanding simulations on MetaCentrum.

The X17 project in IEAP CTU aims to reproduce measurements of anomalous behavior in the distribution of angular correlation of pairs produced by the Internal Pair Formation (IPF) mechanism during the decay of certain excited nuclei (^{8}Be, ^{12}C and ^{4}He) observed by the ATOMKI group in Hungary.

Add citations MetaCentrum, X17 project, VdG, ATOMKI papers. Maybe also TPC, IPF, ...

0.1 ATOMKI Measurements

Short summary of results of measurements in ATOMKI.

0.2 X17 IEAP CTU

Short summary of our goals, maybe mention the grant.

0.2.1 Our Detector

Short description of our detector. Why we use atypic TPC. Gas mixture used in the detector (70/30) and its effect.

0.2.2 Magnetic Field Simulation

Magnetic field simulations in Maxwell. Some figures. When working with magnetic field outside the regular grid trilinear interpolation is used.

Trilinear Interpolation

Trilinear interpolation is a generalization of linear interpolation in 3D. It can be used to interpolate a function whose values are known on a regular grid. We

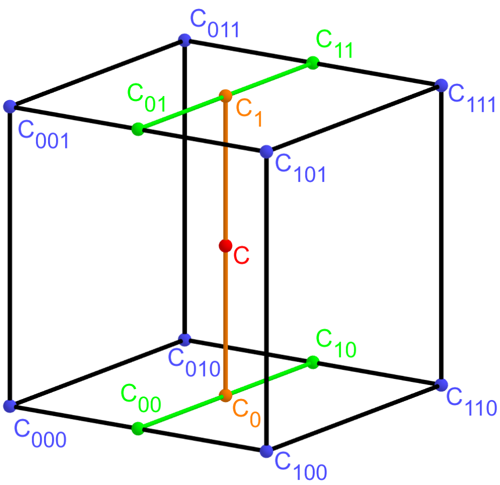

Figure 1: Visualization of trilinear interpolation.

use this simple method for magnetic field, later it is also used in section 3.2.1 to interpolate the Ionization Electron Map. In both cases, we use a cubic grid.

Let us consider a cube (a cell of our regular grid) with an edge of length a containing the point $C = (x, y, z)$ where we want to interpolate a function $f\colon \mathbb{R}^3 \to X$. We know the values of this function on the vertices of this cube $C_{ijk} = (x_0 + ia, y_0 + ja, z_0 + ka)$, where $i, j, k \in \{0, 1\}$. We also define the points $C_{ij} = (x, y_0 + ia, z_0 + ja)$ and $C_i = (x, y, z_0 + ia)$. Then the interpolated value $\widehat{f}(C)$ can be calculated as follows:

$$x_d = \frac{x - x_0}{a}, \; y_d = \frac{y - y_0}{a}, \; z_d = \frac{z - z_0}{a}, \tag{1}$$

$$\widehat{f}(C_{ij}) = (1 - x_d) f(C_{0ij}) + x_d f(C_{1ij}), \tag{2}$$

$$\widehat{f}(C_i) = (1 - y_d) \widehat{f}(C_{0i}) \;\; + y_d \widehat{f}(C_{1i}), \tag{3}$$

$$\widehat{f}(C) = (1 - z_d) \widehat{f}(C_0) \;\; + z_d \widehat{f}(C_1). \tag{4}$$

Maybe a citation here, although I am not sure it is necessary since it could be considered common knowledge.

0.2.3 Coordinate System

Description of the coordinate system used in this thesis (+ figure). Introduce the detector/readout space.

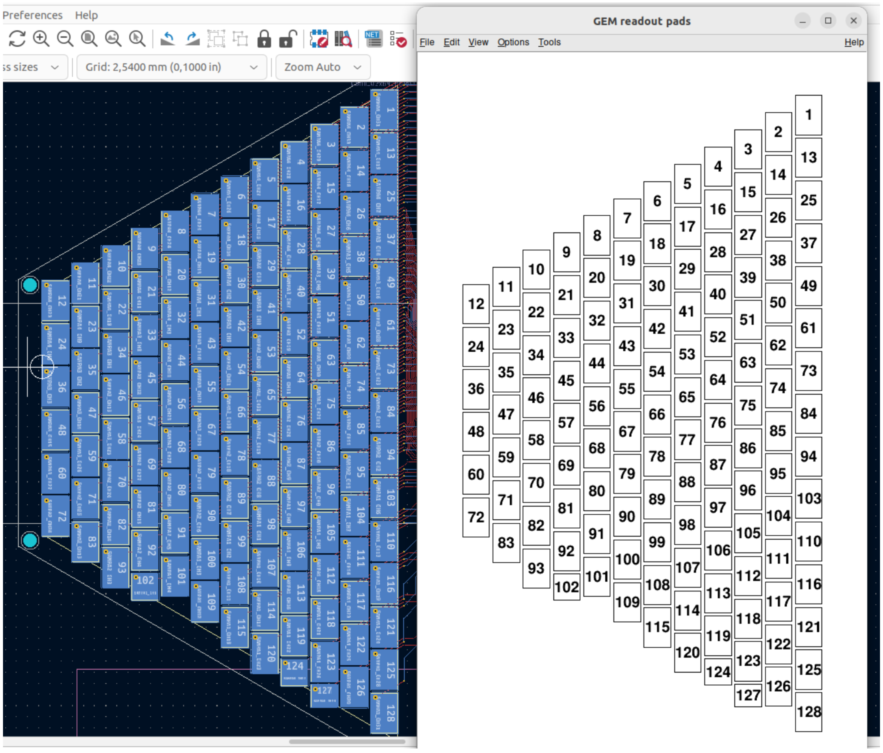

Figure 2: Pad layout of the TPC. Swap for better image.

1. Time Projection Chamber

Description of TPC, working principle, standard vs our field layout.

2. Track Simulation

In order to develop and test the reconstruction algorithm, electron and positron tracks are simulated inside our detector with different initial parameters. Three approaches are used to simulate tracks for different purposes.

The **Microscopic Simulation** uses the Garfield++ toolkit [1]. Within this toolkit, the High Energy Electro-Dynamics (HEED) program [3] is used to simulate the primary particle and the class *AvalancheMicroscopic* to simulate the drift of secondary electrons created by ionization in the gas. This is the most precise and time-consuming simulation used, our current goal is to be able to successfully reconstruct its results and determine our best-case energy resolution.

The **Runge-Kutta Simulation** uses the 4th order Runge-Kutta numerical integration (add citation for Runge-Kutta) to simulate the trajectory of the primary particle in the electromagnetic field inside the detector. It is relatively fast since it does not simulate the secondary particles. It is used as a part of our reconstruction algorithm as well as for testing of some parts of the reconstruction.

The **Fast Simulation with Ionization Electron Map** is planned for the future, it will use the HEED program [3] to simulate the primary particle and the Ionization Electron Map (see section 3.2) to simulate the drift of secondary electrons. It should be significantly faster than the Microscopic Simulation but offer comparable precision since it will rely on an already simulated drift map.

All of these simulations require the knowledge of the electromagnetic field inside the detector. Uniform electric field 400 $\mathrm{V{\cdot}cm^{-1}}$ is assumed. The magnetic field was simulated in Maxwell (add citation? details? own subsection with figures? more details in section 0.2?).

Single track in positive x direction or initial parameters randomization. Importance of gas composition, used gas compositions.

2.1 Microscopic Simulation

The microscopic simulation is the most detailed simulation used in this work. We use the Garfield++ toolkit [1] for this purpose.

The electron transport properties are simulated using the program Magboltz (Add citation.). Two different gas mixtures were used, 90% Ar + 10% CO_2 and 70% Ar + 30% CO_2. The second mixture is planned to be used in our detector. The temperature is set to 20 °C, the pressure is atmospheric.

The primary track is simulated using the program HEED [3], which is an implementation of the photo-absorption ionization model. From this program, we get the parameters of ionizing collisions. HEED can also be used to simulate the transport of delta electrons, we do not account for these in the current simulation but plan to include them in future. The photons created in the atomic relaxation cascade are also not simulated.

Finally, we use the microscopic tracking provided by the class *AvalancheMicroscopic* to simulate the drift of the ionization electrons. Each electron is followed from collision to collision (using the collision rates calculated by Magboltz) using the equation of motion.

First simulated track in the z direction. Figures.

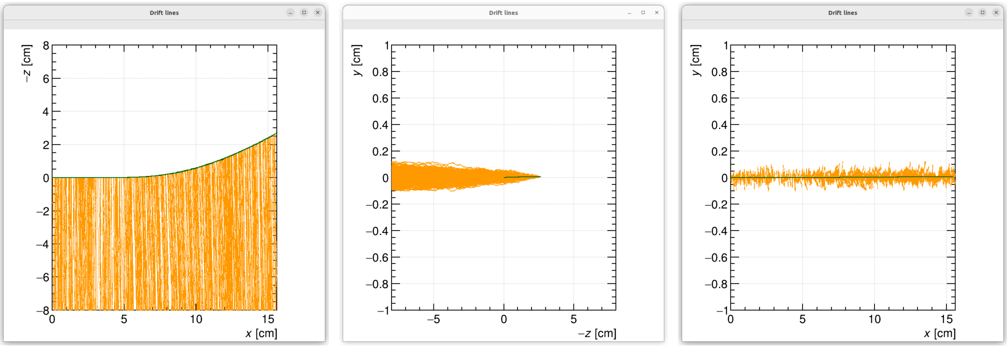

Figure 2.1: Example of a simulated electron track in 70 % argon and 30 % CO_2 atmosphere (on the left). Swap for better images, better zoom. Explain drift lines, primary particle.

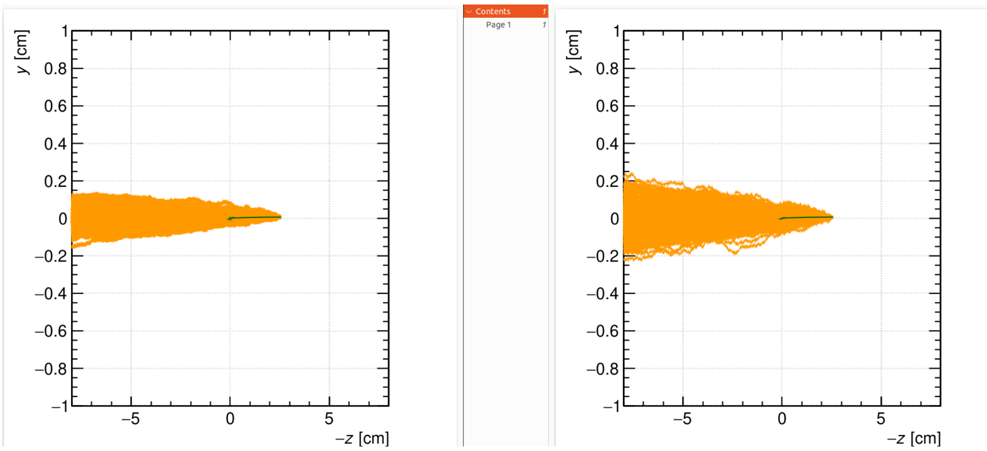

Figure 2.2: Comparison of diffusion in a simulated electron track in 70 % argon, 30 % CO_2 atmosphere and in 90 % argon, 10 % CO_2 atmosphere (on the right). Swap for better image, better zoom. Or put same pictures for both comparisons in one subfigure, etc. Describe better.

2.2 Runge-Kutta Simulation

Trajectory simulation with 4th order Runge-Kutta. Relativistic equation that is numerically integrated by the algorithm.

2.3 Future?: Fast Simulation with the Ionization Electron Map

Primary track simulated in HEED. Readout parameters by interpolating the map. Diffusion from the map for randomization.

3. Track Reconstruction

The first stage of our reconstruction algorithm is the reconstruction of the track of the primary particle (electron or positron). The results of this step are then used to determine the energy of the particle (see section 4).

First Attempts at a track reconstruction were made using the standard approach. Here we assume we know the readout coordinates (x', y', t) exactly (i.e. we neglect the pads and time bins). In standard TPC (with parallel fields) we only need to reconstruct the z coordinate from drift time using the known drift velocity.

Reconstruction with the **Ionization Electron Map** (from now on referred to as *the map*) uses simulation of the drift of the secondary (ionization) electrons in the volume of the detector. This simulation can then be used to interpolate the initial position of the secondary electrons. First attempts neglect the pads.

The **Discrete Reconstruction** is made using the map, instead of reconstructing the exact position of each electron we reconstruct the middle point of each hit pad with time corresponding to the middle of the time bin. The number of electrons in each TPC bin (consisting of the pad and the time bin) is counted and used as a charge in the energy reconstruction.

Reconstruction of one track simulated with microscopic tracking in Garfield++.

3.1 First Attempts

As the first step of the work, we decided to try to reconstruct an electron track with a special set of initial parameters. The origin of the particle is given by the origin of our coordinate system. The initial direction is given by the positive x axis. This means the magnetic field of our detector is perpendicular to the momentum of our particle at all times and we can reduce the problem to two dimensional space. We use a track simulated using the microscopic simulation (see section 2.1) with a kinetic energy of 8 MeV. The gas composition used in this simulation is 90% Ar + 10% CO_2.

In this first approach to the reconstruction of the track, we decided to use the common method used in a standard TPC. This will allow us to explore the significance of the atypical behavior in our OFTPC. At the same time, we consider the readout to be continuous to further simplify the problem. In this approximation we reconstruct the initial position of each ionization electron.

The reconstruction is then defined by the following relations between the coordinates of the detector space and the readout space (see section 0.2.3):

$$x = x', \tag{3.1}$$

$$y = y', \tag{3.2}$$

$$z = v_d t, \tag{3.3}$$

where v_d is the drift velocity of electrons in the given gas mixture. On a phenomenological level, this velocity can be considered a function of the electric field $\boldsymbol{E}$ and the magnetic field $\boldsymbol{B}$:

$$v_d = v_d(\boldsymbol{E}, \boldsymbol{B}). \tag{3.4}$$

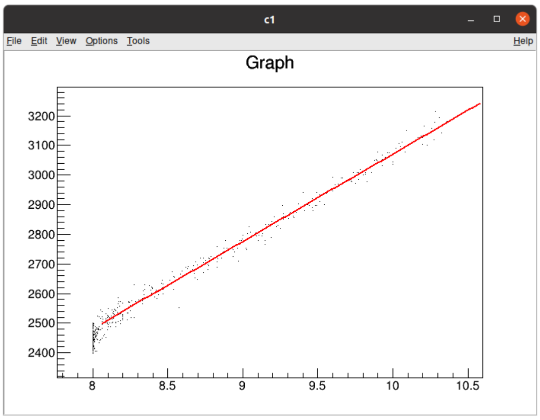

Figure 3.1: Dependence of the drift time on the z coordinate in 90 % argon and 10 % CO_2 atmosphere, fitted with a linear function. The fitted function gives us the average drift velocity in the gas and can be used for rough reconstruction in our TPC. Swap for better image with axis labels, etc. Maybe write the fitted equation.

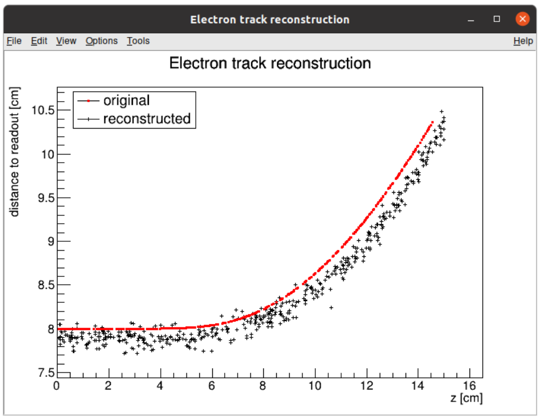

Figure 3.2: First attempt at a track reconstruction using only the drift velocity. This approach works well in a standard TPC (ideally cite some source?). 90 % argon and 10 % CO_2 atmosphere. Swap for better image, correct coordinates.

Taken from Garfield user manual. The Garfield++ toolkit uses this fact to accelerate their drift simulation with non-microscopic approaches. Since we assume uniform electric field in our detector and we want to neglect the effect of our unusual magnetic field, we consider the drift velocity to be constant in this scenario. We then approximate this velocity by fitting the dependence $z(t)$ taken from the simulated ionization electrons.This is in one of the provisional figures. Also this description is not completely accurate, in reality we fit t1:8-y0 with a1*x+a0 and then invert this and use 8-y0 = b1*t1+b0 (old coordinates), b1=1/a1 functions as the drift velocity. Maybe also define this 8-z variable as an alternative to z in the section 0.2.3 and then use it when correcting this.

Later, in a commit after this, I plot some residues (provisional figure) which could be useful but for some reason they are residuals from a spline fit of the track?! Probably redo this without the spline fit, just explore the difference in individual points.

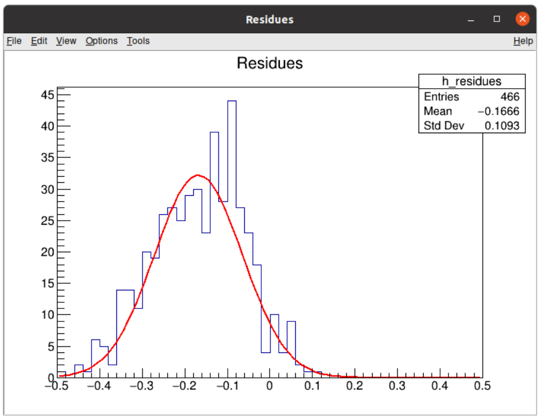

Figure 3.3: First attempt at a track reconstruction using only the drift velocity, residues. Swap for better image, correct coordinates. What's causing the shift? Explain details.

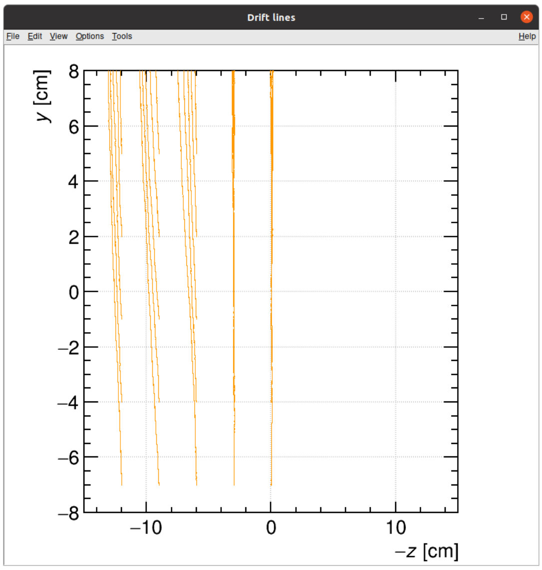

Figure 3.4: Example of map generation. Swap for better image, correct coordinates.

3.2 Ionization Electron Map

Explanation of the map. Simulated on MetaCentrum, workload distribution between multiple jobs. More electrons at one location to get statistics. Two methods of reconstruction using this map. Comparison of 90/10 and 70/30 maps.

3.2.1 Gradient Descent Search

Gradient descent search of a point in the original space that gets mapped to the given point of the readout space (trilinear interpolation).

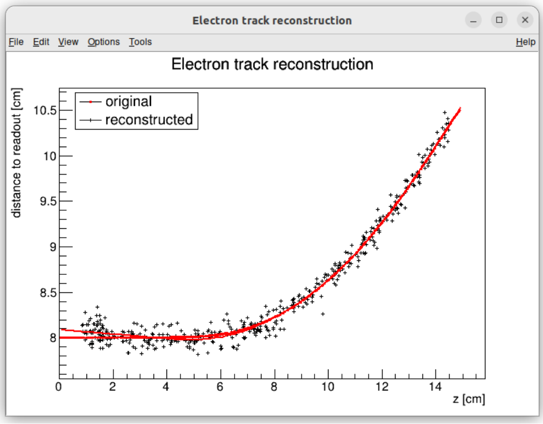

Figure 3.5: Example reconstruction with the map. Swap for better image, correct coordinates.

3.2.2 Interpolating in the Inverse Grid

Interpolating between known points in the readout space. Gaussian elimination, multivariate polynomial.

3.3 Discrete Reconstruction

Reconstruction with pads and time bins. Maybe testing different pads.

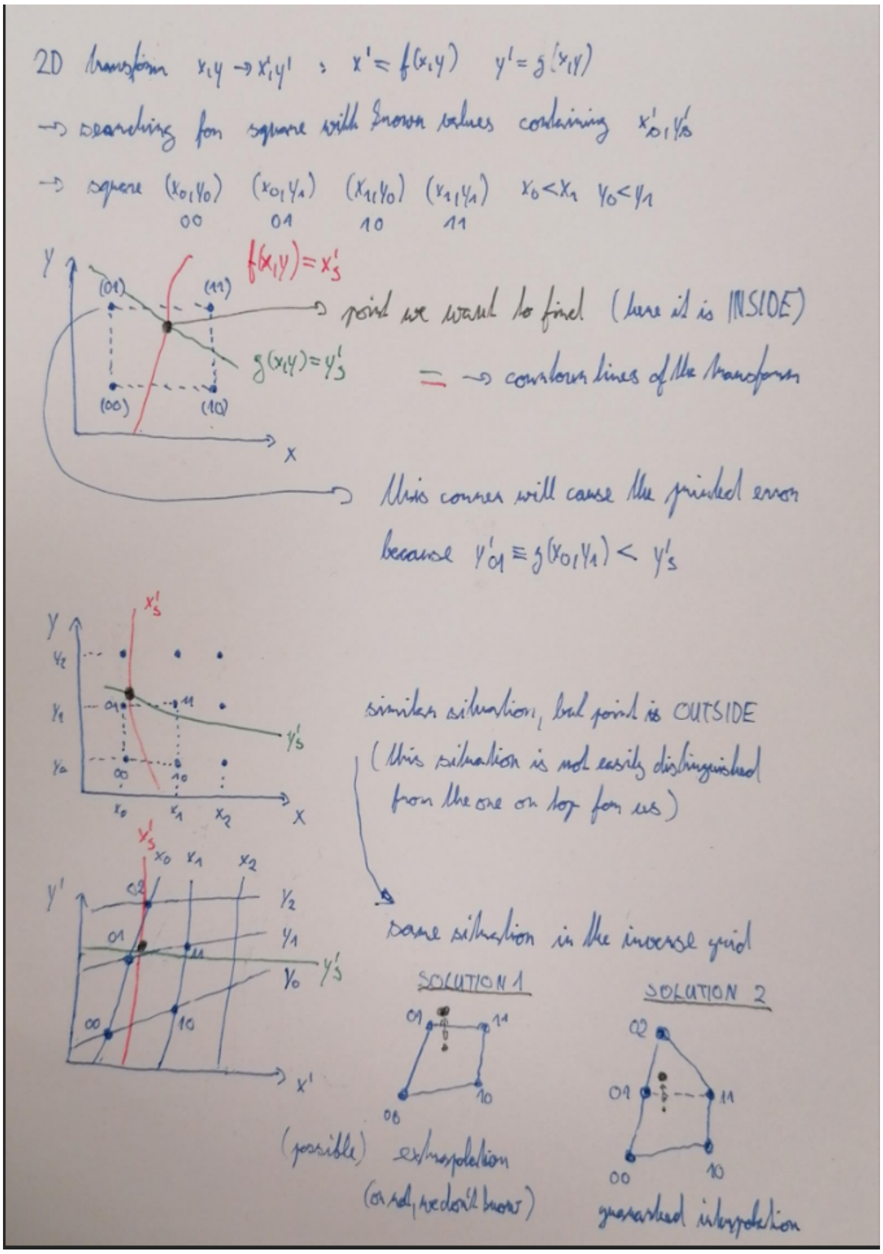

Figure 3.6: Selection of the points for interpolation. Create better images, use the explanation interpolation vs extrapolation strange property. Solution 2 probably does not make much sense.

4. Energy Reconstruction

The second stage of our reconstruction algorithm is the reconstruction of the particle's energy using its reconstructed track (see section 3). We can achieve this by fitting the track and extracting the needed parameters of the trajectory. We have tested three ways of reconstructing the energy. Fitting is done using the MINUIT algorithm implemented in ROOT [2]. Maybe cite some CERN article directly on MINUIT?

The **Cubic Spline Fit** is a rejected attempt at the reconstruction of energy. It uses smoothly connected piecewise cubic polynomials between uniformly spaced nodes. Energy can then be computed using from the fit parameters by computing the radius of curvature in different points of the fitted curve using the known magnitude of the magnetic field perpendicular to the trajectory. This approach was rejected because tuning the fit to have a reasonably stable radius of curvature is unpractical.

The **Circle and Lines Fit** was chosen as an alternative since this corresponds to the shape of a trajectory of a charged particle moving through a finite volume with a homogeneous magnetic field. The energy of the particle can be estimated using the fitted radius and the magnitude of the perpendicular magnetic field in the middle of the TPC.

The **Runge-Kutta Fit** uses the 4th order Runge-Kutta numerical integration described in section 2.2. Initial parameters of the track (including the particle's energy) are optimized so that the integrated trajectory fits to the reconstructed one. This fit can also be performed as a single parameter (energy) fit if we can get the initial position and orientation of the particle on the entrance to the TPC from previous detectors (Timepix 3 (Tpx3) and Multi-Wire Proportional Chamber (MWPC), see section 0.2).

4.1 Cubic Spline Fit

The first attempt to get an early estimate of the kinetic energy of the particle uses a cubic spline fit. This approach was later rejected in favor of the circle and lines fit described in section 4.2. We use an electron track starting in the origin of

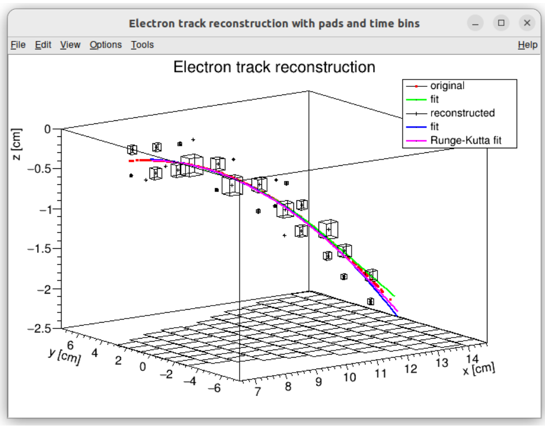

Figure 4.1: Example of a fitted reconstructed track. Swap for better image.

our coordinate system with an initial direction in the positive x axis. The track is simulated microscopically (see section 2.1) with a kinetic energy of 8 MeV in a gas mixture 90% Ar + 10% CO_2 (the same track was used in section 3.1).

In order to calculate the spline, we use the class *TSpline3* from ROOT. This allows us to evaluate the spline using the coordinates (x_n, z_n) of each node and the derivatives d_1, d_2 in the first and the last node. We can fit these parameters of a fixed amount of nodes to the simulated trajectory. We use the IMPROVE algorithm provided by *TMinuit* class in ROOT. This algorithm attempts to find a better local minimum after converging.

After the fit, we want to get an energy estimate. We can calculate it in every point using the radius of curvature of the fitted spline. In ROOT, the part of the spline corresponding to a given node is defined as

$$z(x) = z_n + b\Delta x + c(\Delta x)^2 + d(\Delta x)^3, \tag{4.1}$$

where $\Delta x = x - x_n$ and b, c, d are coefficients. Using this equation, we can derive the radius of curvature:

$$r(x) = \frac{(1 + z'^2(x))^{\frac{3}{2}}}{z''(x)} = \frac{\left(1 + (b + 2c\Delta x + 3d(\Delta x)^2)^2\right)^{\frac{3}{2}}}{2c + 6d\Delta x}. \tag{4.2}$$

From the geometry of the detector, we can assume the magnetic field $\boldsymbol{B}(x, 0, z) = (0, B(x, z), 0)$ for track in the XZ plane. Since the electron is relativistic, the effect of the electric field on its trajectory is negligible. The Lorentz force F_L is then always perpendicular to the momentum of the electron and is therefore equal to the centripetal force F_c:

$$F_L = F_c, \tag{4.3}$$

$$e\boldsymbol{v} \times \boldsymbol{B} = \frac{\gamma m_e v^2}{r}, \tag{4.4}$$

$$ec\beta B = \frac{E_{0e}\beta^2}{r\sqrt{1-\beta^2}}, \tag{4.5}$$

$$\sqrt{1-\beta^2} = \frac{E_{0e}\beta}{ecBr}, \tag{4.6}$$

$$\beta^2(x) = \frac{1}{1 + \left(\frac{E_{0e}}{ecB(x,z(x))r(x)}\right)^2} \tag{4.7}$$

where e is the elementary charge, c is the speed of light in vacuum, m_e is the rest mass of electron, $E_{0e} = m_e c^2$ is the corresponding energy, γ is the Lorentz factor, $\boldsymbol{v}$ is the velocity of the electron and $\beta = \frac{v}{c}$. We can then finally get our estimate of the kinetic energy for given point on the trajectory as follows:

$$E_{\text{kin}}(x) = \left(\frac{1}{\sqrt{1-\beta^2(x)}} - 1\right) E_{0e}. \tag{4.8}$$

We can then average these estimates in multiple points to get one final estimate. Add some figures.

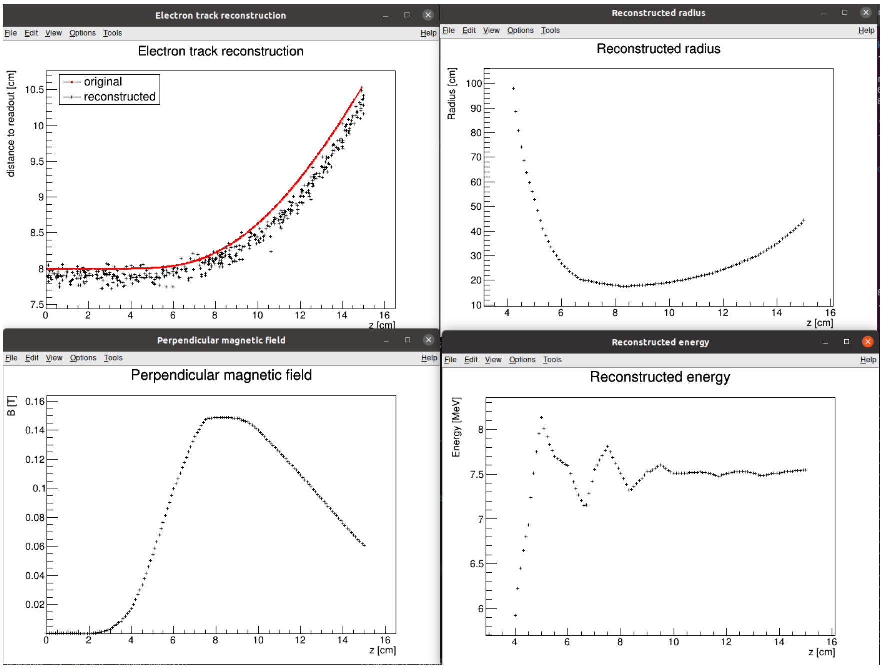

Figure 4.2: First attempt at a track reconstruction using only the drift velocity. Spline energy reconstruction attempt. Swap for better image(s) – subfigure environment., correct coordinates.

4.2 Circle and Lines Fit

A simpler alternative for the first estimation of the kinetic energy of the particle is a fit of the trajectory with circular arc with smoothly attached lines. This shape of trajectory corresponds to a movement of a charged particle through homogeneous magnetic field perpendicular to the particle's momentum and limited to a certain volume. In general, the shape of such trajectory in non-perpendicularly oriented field is a spiral. In our case, this component should be negligible since the field is approximately toroidal and the movement of the particles is close to perpendicular to this toroid. At first, we tested a 2D version of this fit, later we adapted it to 3D.

Since our field is not homogeneous, it is not entirely clear what value of magnetic field should be used along with the fitted radius (using equations 4.7 and 4.8) to get the best estimate for the kinetic energy. Since we only use this method to get a first rough estimate that we later refine, an optimal solution of this problem is not required. Instead, we tested two options – taking the value of the field in the middle of the fitted circular arc and taking the average field along it. We haven't really tried to plot this for multiple tracks but these estimates are saved somewhere and could be plotted.

In the 2D case, the fitted function used for the electron track (which bends down, so we need to use the upper part of the circle) described in section 4.1 looks like this: Maybe describe this track that we used at the beginning somewhere earlier (section microscopic simulations → Testing track?) so that it is easier to refer to it in multiple sections. It is not part of the early Github commits so maybe won't be possible to create exact replicas of the images but should be at least very similar.

$$z(x) = \begin{cases} a_1 x + b_1 & x < x_1 \\ z_0 + \sqrt{r^2 - (x - x_0)^2} & x_1 \le x \le x_2 \\ a_2 x + b_2 & x > x_2 \end{cases}, \tag{4.9}$$

where $a_{1,2}$ and $b_{1,2}$ are the parameters of the lines, (x_0, z_0) is the center of the circle, r is its radius and $(x_{1,2}, z_{1,2})$ are the coordinates of the function's nodes. That means we have 9 parameters ($z_{1,2}$ is not used in the function) along with 2 continuity conditions and 2 smoothness conditions. For the fit, we use the coordinates of the nodes and the radius of the circle which gives us 5 independent parameters (only the radius has to be larger than half of the distance between nodes). The continuity conditions (combined with the relations for $z_{1,2}$) are as follows:

$$z_{1,2} = a_{1,2} x_{1,2} + b_{1,2} = z_0 - \sqrt{r^2 - (x_{1,2} - x_0)^2}. \tag{4.10}$$

The smoothness conditions are as follows:

$$a_{1,2} = \frac{x_0 - x_{1,2}}{\sqrt{r^2 - (x_{1,2} - x_0)^2}}. \tag{4.11}$$

Equation 4.10 gives us the values of $b_{1,2}$

$$b_{1,2} = z_{1,2} - a_{1,2} x_{1,2}. \tag{4.12}$$

For the coordinates of the center of the circle, we can use the fact that the center has to lie on the axis of its chord. In other words, there is a value of a parameter t such that, using the parametric equation of the axis

$$\begin{pmatrix} x_0 \\ z_0 \end{pmatrix} = \begin{pmatrix} \frac{x_1+x_2}{2} \\ \frac{z_1+z_2}{2} \end{pmatrix} + t \begin{pmatrix} \frac{z_2-z_1}{2} \\ \frac{x_1-x_2}{2} \end{pmatrix}. \tag{4.13}$$

At the same time, the center has to be in a distance r from the nodes:

$$(x_1 - x_0)^2 + (z_1 - z_0)^2 = r^2, \tag{4.14}$$

$$\left(\frac{x_1 - x_2}{2} + \frac{z_1 - z_2}{2} t\right)^2 + \left(\frac{z_1 - z_2}{2} + \frac{x_2 - x_1}{2} t\right)^2 = r^2, \tag{4.15}$$

$$\left(\left(\frac{x_2 - x_1}{2}\right)^2 + \left(\frac{z_2 - z_1}{2}\right)^2\right) t^2 + \left(\frac{x_2 - x_1}{2}\right)^2 + \left(\frac{z_2 - z_1}{2}\right)^2 - r^2 = 0. \tag{4.16}$$

Since our electron track bends towards negative z and $x_2 > x_1$, we only care about the solution with $t > 0$

$$t = \sqrt{\frac{r^2}{\left(\frac{x_2-x_1}{2}\right)^2 + \left(\frac{z_2-z_1}{2}\right)^2} - 1}, \tag{4.17}$$

$$x_0 = \frac{x_1 + x_2}{2} + \frac{z_2 - z_1}{2} \sqrt{\frac{r^2}{\left(\frac{x_2-x_1}{2}\right)^2 + \left(\frac{z_2-z_1}{2}\right)^2} - 1}, \tag{4.18}$$

$$z_0 = \frac{z_1 + z_2}{2} - \frac{x_2 - x_1}{2} \sqrt{\frac{r^2}{\left(\frac{x_2-x_1}{2}\right)^2 + \left(\frac{z_2-z_1}{2}\right)^2} - 1}. \tag{4.19}$$

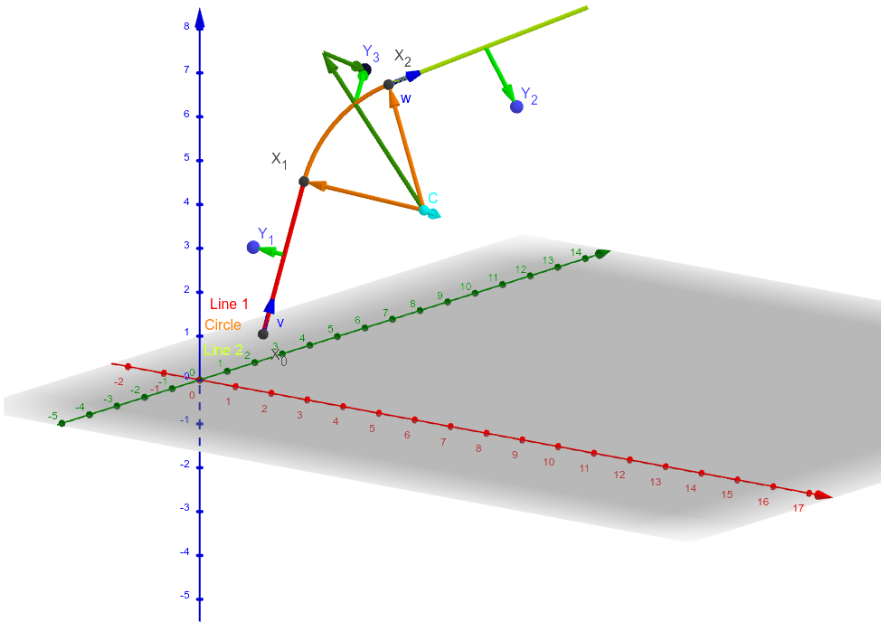

Figure 4.3: Circle and Lines Fit 3D geometry. Swap for better image.

The function defined in the relation 4.9 along with the relations 4.11, 4.12, 4.18 and 4.19 derived using the continuity and smoothness conditions (combined with the relations for $z_{1,2}$) give us the full prescription of our fitted function with parameters $r, x_{1,2}, z_{1,2}$. Some pictures of the fit on the tested track. Results of the fit. Again, the actual fit uses 8-z. Use GeoGebra schematics to generate some picture of 2D geometry.

Energy reconstruction with circle and lines fit. Trilinear interpolation of the magnetic field. Tested on Runge-Kutta sample, future testing with microscopic simulations and map simulation. Preliminary 2D version (done) and complete 3D version. Geometry of the fit with its derivation.

4.3 Runge-Kutta Fit

Single parameter fit with 4th order Runge-Kutta simulated track. Future testing with microscopic simulations and map simulation. Derivation of the geometry (least squares).

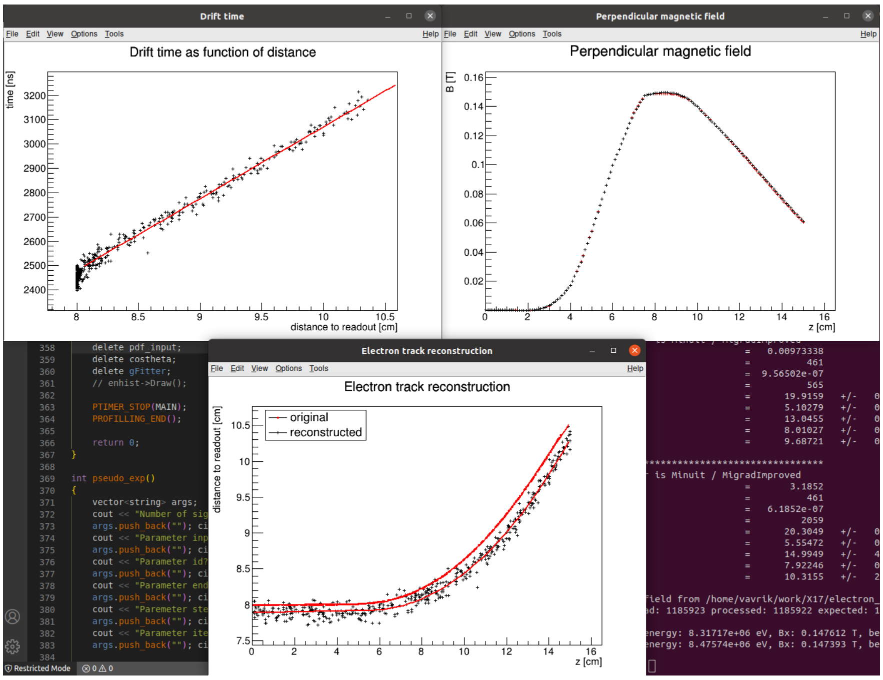

Figure 4.4: First attempt at a track reconstruction using only the drift velocity. Circle and Lines Fit in 2D. Swap for better image, correct coordinates.

Conclusion

Here or at the end of each section. Something about the future of this work?

Bibliography

[1] Garfield++. `https://garfieldpp.web.cern.ch/garfieldpp/`. Accessed: 2023-05-18.

[2] Rene Brun and Fons Rademakers. Root — an object oriented data analysis framework. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 389(1–2):81–86, Apr 1997. Proceedings AIHENP'96 Workshop, Lausanne, Sep. 1996, See also https://root.cern/, Paper published in the Linux Journal, Issue 51, July 1998.

[3] I.B. Smirnov. Modeling of ionization produced by fast charged particles in gases. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 554(1):474–493, 2005.

List of Figures

List of Tables

List of Abbreviations

HEED High Energy Electro-Dynamics

IEAP CTU Institute of Experimental and Applied Physics, Czech Technical University in Prague

IPF Internal Pair Formation

MWPC Multi-Wire Proportional Chamber

OFTPC Orthogonal Fields TPC

TPC Time Projection Chamber

Tpx3 Timepix 3