

Problem Set for Hadron Collider Physics 2015

Prof. Dr. Karl Jakobs, Dr. Karsten Köneke

Problem Set 2

Please note that this problem set consists of two parts for a total of two weeks. Problem 1-3 are due on Tuesday, May 5th 2015. Whereas problems 4-5 are due on Tuesday, May 12th 2015. Please drop your solutions into the mailbox number 1 on the ground floor of the Gustav-Mie building on the indicated days by 10:10am!

1. Tracking

In order to measure the momentum of charged particles, one attempts to measure several points in space along the trajectory of the charged particle inside a magnetic field. The charged particle is deflected inside the magnetic field due to the Lorentz force. Measurements are performed at several points along the trajectory in order to determine precisely where the charged particle passed through the different layers of the detector.

The reconstruction of the trajectory of an electron with a momentum of 20 GeV/c is performed where the electron momentum is perpendicular to a 5 T homogeneous magnetic field. For this measurement, ten thin layers of a tracking detector are spaced equidistantly over a total length of 5 m, perpendicular to the trajectory of the incoming electron, see Fig. 1.

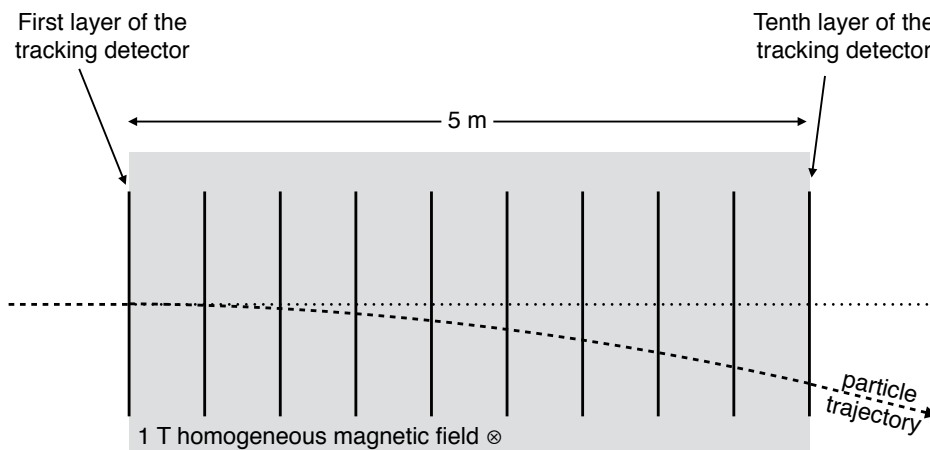


Figure 1: Schematic drawing of a tracking detector with ten active measurement layers inside a homogeneous magnetic field.

(a) Derive for this simple case the equation

$$p_T = 0.3 B R \quad (1)$$

where

- p_T is the component of the momentum perpendicular to the magnetic field in GeV/c,
- B is the strength of the homogeneous magnetic field in Tesla,
- R is the radius of the circular trajectory in meters on which the particle is forced due to the Lorentz force in the homogeneous magnetic field.

Calculate with this formula the radius of curvature R of the charged particle and the angle of deflection. What is the minimum momentum that a charged particle must have in order to reach the last active layer of the tracking chamber? **[1 point]**

- (b) The individual layers of the tracking detector have a spacial resolution of $200 \mu\text{m}$. What is the relative momentum resolution $\sigma(p_T)/p_T$? You can neglect the term describing multiple scattering. (*Hint: You can use the “Reviews” of the “Particle Data Group” (<http://pdg.lbl.gov/>), especially chapter 33.11. Furthermore, you need the law of uncertainty propagation of independent sources of uncertainty $\sigma(y) = \sqrt{\sum_i \left[\left(\frac{\partial y}{\partial x_i} \right)^2 (\sigma(x_i))^2 \right]}$, in order to go from the uncertainty on the curvature to the uncertainty on the momentum.) **[2 points]***

- (c) Assume this tracking chamber is followed by a calorimeter which has a relative energy resolution for charged pions of

$$\frac{\sigma(E)}{E} = \frac{80\%}{\sqrt{E \text{ (GeV)}} \oplus 5\%. \quad (2)$$

Would you rather use the calorimeter or the tracking chamber to measure the energy of a charged pion with a momentum of $3 \text{ GeV}/c$? How about a charged pion with a momentum of $300 \text{ GeV}/c$? Justify your answer. **[1 point]**

2. Electromagnetic showers

The aim of the following exercise is to get a handle on the features of an electromagnetic shower. A very schematic model for the shower is the following. After each radiation length X_0 , the number of particles is doubled (at the first stage, after one radiation length, an incoming electron emits a photon. After another radiation length, the electron will radiate a photon, while the photon produced in the first interaction creates an electron-positron pair, and so on...). Assume that at each stage, the energy is split equally among the particles emitted in one interaction. The multiplication process continues until the energy of the particles reaches the critical energy. Determine:

- The number of particles present in the shower as a function of the shower depth. **[1 point]**
- The dependence of the shower maximum (that is, the position at which the number of particles is maximum) on the incoming primary particle energy and the calorimeter critical energy. **[1 point]**

3. Hadronic showers

The aim of this exercise is to get a handle on why hadronic particles in non-compensating calorimeters have a response that depends in a non-linear way on the incoming particle energy.

Assume that a hadronic shower is made up of two components: a pure hadronic component and an electromagnetic component. The pure electromagnetic component consists of mainly π^0 s produced in a hadronic decay that immediately decay through $\pi^0 \rightarrow \gamma\gamma$. Non-compensating calorimeters are characterized by the fact that the electromagnetic and hadronic component of the hadronic shower have a different response. This, combined with the fact that the relative importance of the electromagnetic and hadronic components of the shower are energy dependent, makes the calorimeter response to hadrons non-linear.

Let's say that if the incoming pion energy is E , then a fraction f_0 of this will be seen in the form of photons from π^0 decays, and measured with a response 1. The remaining incoming energy will be measured with a response $h < 1$.

- Write down an expression for the pion energy response, as a function of E , f_0 , h . [1 point]
- Write down an expression for the so-called e/π ratio, that is, the ratio of the energy measured in the calorimeter for an electron and a pion of the same incoming energy. [1 point]

We now want to get an insight on the f_0 dependency on the pion energy. We therefore assume that on each interaction, the available energy is equally split among two charged pions and one neutral pion, until a threshold E_0 for pion production is reached.

- Compute the dependency of f_0 on the incoming pion energy. [1 point]
The following formulas might be helpful:

$$\sum_0^n x^k = \frac{1 - x^{n+1}}{1 - x} \quad (3)$$

$$\alpha^{\log \beta} = \beta^{\log \alpha} \quad (4)$$

- Determine the resulting e/π ratio as a function of the incoming particle energy and plot the result using your favorite program, e.g., ROOT, Mathematica, Maple, Wolfram Alpha,... . [1 point]

The following exercises are meant to make you familiar with the `Pythia` event generator and with ROOT (<https://root.cern.ch>). Please attempt at least the setup and running of one example before you come to the exercise group session on Thursday, May 7th, such that technical problems can be solved there.

4. Basics of event generation

Event generators are important tools in particle physics. They are used to simulate the generation of high-energy collisions as taking place in real life collider experiments. We want to get a first look at the basic functionality of those programs. As an example, we will use `Pythia 8`. Download and install the `Pythia` event generator (available at <http://home.thep.lu.se/~torbjorn/Pythia.html>). Follow the instructions given there. You can find the online manual for this event generator at <http://home.thep.lu.se/~torbjorn/pythia82html/Welcome.html>.

- (a) After compilation, try and understand the `main01` program from the examples directory (Type: “make main01” in this directory in order to compile the code). Attach the log of this example number 1. [1 point]

5. Event record and storage

You will study the event record of `Pythia` in more detail and save it for further analysis. The ROOT `ntuple` is a suitable format for this and is widely used in high energy physics. This section is here to introduce you to these two programs.

- (a) Look again at the output of the `main01` program. Compare the event record of the first event. Summarize in words the differences between the *hard process* and the *full event*. **[2 points]**
- (b) Now, have a look at the `main03` program. This is an example how you can use a steering file, called `main03.cmd` in this case, that you can adjust between runs without recompiling the code. It makes it possible to easily turn on/off the simulation steps: parton shower, multiple parton interactions and hadronization. Try out several combinations and check again the event record and compare to your previous findings. What do you observe? **[2 points]**
- (c) In order to not only be restricted to analyze one event by eye we want to store the information in a ROOT file. To do this you have to create an instance of a `TFile` and a `TTree` :

```
TFile myFile("myFile.root", "RECREATE");
TTree myTree("myTree", "myTree");
```

Use the `Branch` method to connect the variable “var” to your the output file:

```
float var;
myTree.Branch("var", &var, "var/F");
```

In the event loop fill the value of the variable into the tree:

```
myTree.Fill();
```

After the event loop save your output:

```
myFile.Write();
```

For each event, you should store 5 vectors (of the same length, i.e., the number of particles), containing the transverse momentum, the pseudorapidity $\eta = -\ln(\tan(\theta/2))$, the ϕ angle, the charge and the PDG identity of each particle produced in the event. The PDG identity is a unique identifier that stores, what type of particle it is (π^\pm , π^0 , K^\pm , and so on). To decode and understand the PDG identity, have a look at its documentation at <http://pdg.lbl.gov/2002/montecarloorpp.pdf>. Send a working version of your source code via email to your exercise group instructor. Please make sure that your comments in the code and your coding style make it easy for others (and yourself in a few weeks time!) to read your code. **[2 points]**