

Exercises for Statistical Methods in Particle Physics

<http://www.physi.uni-heidelberg.de/~nberger/teaching/ws13/statistics/statistics.php>

Dr. Niklaus Berger (nberger@physi.uni-heidelberg.de)

Dr. Oleg Brandt (obrandt@kip.uni-heidelberg.de)

Exercise 7: simulating a photomultiplier tube

25. November 2013

Hand-in solutions by 14:00, 1. December 2013

Please send your solutions to obrandt@kip.uni-heidelberg.de by 1.12.2013, 14:00, punctually. Make sure that you use *SMIPP:Exercise07* as subject line. Please hand in your solutions as C++ code this time, and put your programme for all of 7.1-7.4 into one single .C file using the provided base template, in order to guarantee some structural similarity of the solutions. If plots are requested, please include print statements to produce pdf files in your code, and provide the plots separately. In contrast to previous problem sheets, please use the histogram names from the template to print the canvas (cf. example in template). Please also write the histograms into a .root file (cf. example in template), which can be practical. Please add comments to your source code explaining the steps. Test macros and programs before sending them off...

The calorimeter is the core instrument of practically any elementary particle physics experiment, which is used to measure the energies of incident particles via the electromagnetic and possibly strong interactions with the detector material. This measurement is typically performed by detecting the scintillation light which is induced by those interactions inside the detector material. The detection is typically performed by photomultiplier tubes (PMT), which use the photoelectric effect to generate primary electrons from the incident scintillation photons, and multiplying them to an electronically measurable signal via a cascade between stages of the PMT (hence the name). For example, the hadronic calorimeter of the ATLAS experiment uses such a detection principle. In this problem sheet, we will simulate the operation of such a device.

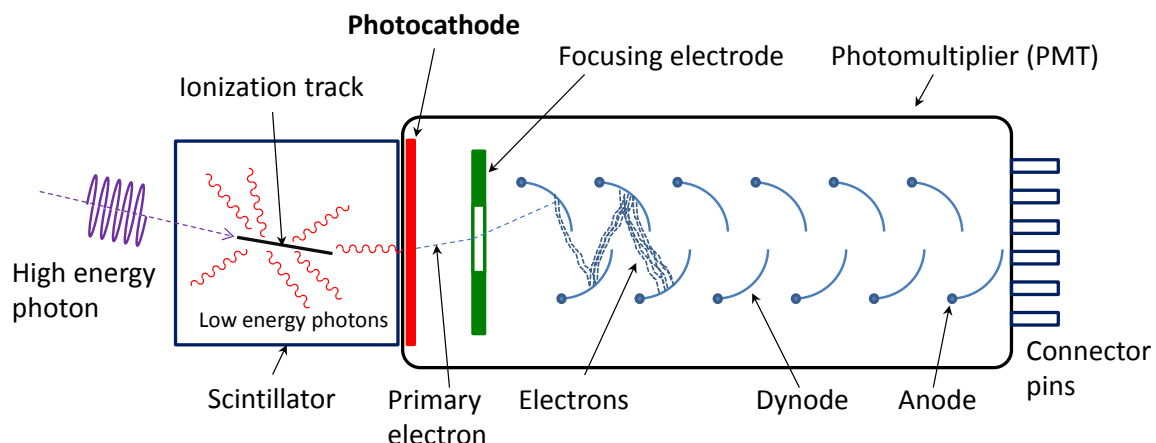


Abbildung 1: Schematic drawing of a PMT alongside an electron cascade.

A schematic drawing of a PMT alongside an electron cascade is given Fig. 1. A photon strikes the photocathode, where there is a certain probability for it to eject an electron (called a photoelectron). The photoelectron is accelerated in an electric field towards an electrode (called dynode). In the collision with the first dynode, the photoelectron can liberate further electrons. These are accelerated towards the second dynode, where more electrons are produced. This continues through a series of stages until the electrons produced at the final dynode are collected. The number of electrons produced at the i -th dynode for each incoming electron can be modeled as a Poisson variable n_i with mean value ν_i , which in general can be different for each stage. Suppose the photomultiplier has N dynodes. The number of electrons n_{out} produced at the final stage for a single incident photoelectron has an expectation value

$$\bar{v}_{\text{out}} = E(n_{\text{out}}) = \prod_{i=1}^N \nu_i. \quad (1)$$

Further information on photomultipliers can be found in [1], [2].

1 One electron in a PMT with six dynodes

Write a Monte Carlo program to determine the distribution of the number of electrons n_{out} at the end of $N = 6$ dynodes produced by a single initial photoelectron.

For this, generate Poisson random numbers with the ROOT function `Random::Poisson(nu)` with $\nu = 3.0$ for each dynode. Run the program to simulate the passage of $M = 10000$ initial photoelectrons, one-by-one, through the PMT. Track the number of photoelectrons being emitted after each dynode stage with a histogram and, in particular, the number emitted from the final stage, n_{out} . Estimate the sample mean and variance using

$$\bar{n}_{\text{out}} = \frac{1}{M} \sum_{i=1}^M n_{\text{out},i} \quad (2)$$

and

$$V(n_{\text{out}}) = \frac{1}{M-1} \sum_{i=1}^M (n_{\text{out},i} - \bar{n}_{\text{out}})^2. \quad (3)$$

Compare the sample mean to the value from Eq. 1. Compare the sample SD $\sigma(n_{\text{out}})$ to the value that one would obtain from a Poisson-distributed variable with mean ν_{out} . Form Eq. 3 in such a way that one loop is sufficient to calculate it. Explain qualitatively why the standard deviation of n_{out} from the variance in Eq. 3 is much larger than in the Poisson case.

2 Improving the resolution of single photoelectrons

Ideally, one would like the standard deviation of \bar{n}_{out} to be as small as possible in order to determine as accurately as possible the number of photoelectrons emitted from the cathode (and thus estimate the number of photons entering the detector). In some experimental applications, one would like to have the SD small enough to distinguish between one single and a pair of photoelectrons. Therefore one tries to have a relative resolution, i.e. the ratio of the SD to the mean, of less than unity. One way of achieving this is to increase the mean number of electrons produced at the first dynode. This can be done either by increasing the accelerating electric field, or by using a dynode metal with a low work function, i.e. a high probability for secondary electron emission.

Repeat the simulation from Exercise 7.1 while increasing the mean number of electrons emitted by the first dynode to $\nu_1 = 6.0$. Estimate the ratio of the standard deviation to the mean of

n_{out} for both values of ν_1 . Explain qualitatively why this gives a better resolution than in the case with equal ν_i . Why does this not help much to increase the gain of the dynodes in the later stages of the PMT?

3 Correlation of n_{out} to the number of photoelectrons in the first dynode

Show that the mean of n_{out} is proportional to the number of electrons emitted by the first dynode. Use profile histograms (TProfile) to show this. How many electrons are emitted per electron at the first dynode?

Literatur

- [1] W. R. Leo, *Techniques for Nuclear and Particle Experiments*, Chapter 8, Springer Verlag, Heidelberg.
- [2] K. Kleinknecht, *Detektoren für Teilchenstrahlung*, Chapter 4.1, Teubner, Stuttgart.

Template code

```
// Ex 7.x.x: simulate a PMT
// Statistical Methods in Particle physics WS 2013
//
// Nomen Nominandum
// nomen.nominandum@stud.uni-heidelberg.de

#include <cmath>
#include <iostream>
#include <TRandom3.h>
#include <TFile.h>
#include <TCanvas.h>
#include <TH1D.h>
#include <TProfile.h>

TRandom3 rnd ;

// here go the helper methods to be called from ex_7() at the bottom

// trace one photoelectron through a photomultiplier
//
// returns      number of electrons after last dynode
// ndynodes     number of dynodes
// nus         mean number of emitted el. per incoming el. (per dynode)
// n           number of el. after each dynode
unsigned long pmt(unsigned ndynodes, double nus[], unsigned long n[]) {
    // trace one photoelectron through the PMT
}

// trace electrons through a photomultiplier (M experiments)
//
// nu1         Poisson mean of first dynode
// nurest      Poisson mean of other dynodes
// ndynodes    number of dynodes
```

```

// M          number of experiments
// histos     array of pointers to histograms (if 0, don't try to fill)
//            (for number of electrons after each dynode)
// hprof      pointer to profile histogram of number of el. after first
//            dynode vs number of el. after last dynode
//            (if 0, don't try to fill)
void dynodes6( double nu1, double nurest, unsigned ndynodes
               , unsigned M, TH1D** histos, TProfile* hprof
               ) {
    // set up array of poisson means for each dynode
    double nus[ndynodes] ;
    for( int i=0 ; i!=ndynodes ; ++i ) nus[i] = nurest ;
    nus[0] = nu1 ;
    // array of electrons after each dynode
    unsigned long n[ndynodes] ;

    // do M experiments, fill histograms, calculate mean and variance

    // print mean and variance
    return ;
}

// main program
int ex_7() {
    // open file to store resulting histograms
    TFile file("ex_7.root", "RECREATE") ;
    TCanvas* cnv = new TCanvas( "cnv", "cnv", 300, 300 ) ;
    // first, do exercise 7.1 to 7.3
    // use a for loop to simulate the two values of nu1
    for( int nu1=3 ; nu1<=6 ; nu1+=3 ) {
        // array of histograms, one for each dynode
        TH1D* histos[6] ;
        for( int i=0 ; i!=6 ; ++i ) {
            histos[i] = new TH1D( Form("nu%dn%02d", nu1, i + 1)
                                , Form("number of el. after dynode %i (#nu_{1}=%d)", i + 1, nu1)
                                , 50., -0.5, double(25 + i*1000) - 0.5
                                ) ;
        }
        // profile histogram: number of el. after first dynode vs number of
        // el. after last (for ex. 7.3)
        TProfile* hprof = new TProfile( Form("nu%dn1vsnlast", nu1)
                                       , Form("number of el. after 1st dynode vs. "
                                             "number of el. after last (#nu_{1}=%d)", nu1)
                                       , 20, -0.5, 19.5
                                       ) ;
        // ok, trace 10000 photoelectrons through the PMT, recording the results
        dynodes6( double(nu1), 3., 6, 10000, histos, hprof ) ;
        // print, write and delete histograms
        hprof->Write() ;
        hprof->Draw() ;
        cnv->Print( Form("%s.pdf", hprof->GetName()) ) ; //histo name for printing
        delete hprof ;
        for( int i=0 ; i!=6 ; ++i ) {
            histos[i]->Write() ;
            delete histos[i] ;
        }
    }
}

```

```
}  
// now comes exercise 7.4  
//TH1D *hpmt12 = new TH1D("pmt12", "Number of electrons after 12 dynodes",  
//      100, 0., 5e6) ;  
///// use 10000 photoelectrons entering our 12 dynode PMT  
//for (unsigned iex = 0 ; iex < 10000 ; ++iex)  
//  hpmt12->Fill(dynodes12()) ;  
///// and write the histogram to the output file  
//hpmt12->Write() ;  
//delete hpmt12 ;  
///// close file  
//file.Close() ;  
  
// end main successfully (note: return meaningful error code otherwise)  
return 0 ;  
}
```
