Lab 1: Photon Counting with a Photomultiplier Tube

1 Introduction

1.1 Goals

In this lab, you will investigate properties of light using a photomultiplier tube (PMT). You will assess the quantitative measurements (and limitations) in detecting optical light. You will measure the brightness of a source of light in terms of photons detected within a specified duration of time. You will assess how precisely that brightness can be determined and what uncertainties are intrinsic to counting photons, notably Poisson errors.

1.2 Schedule

2 week - Report due Tuesday, September 11

1.3 Recommended Reading

- Taylor Chapters 1, 2 & 3, especially about Poisson and Gaussian statistics.
- The PMT web page: http://ugastro.berkeley.edu/optical/photon

2 Gathering Data from the PMT

Make a directory (*mkdir*) for the data in this experiment and change directories (*cd*) to it:

```
% mkdir pmtdata
% cd pmtdata
```

Operating from this directory, the output from the PMT will be stored in that same directory. Start IDL. The third step actually records and saves the number of photons recorded by the PMT. Begin recording data from the PMT by typing the following (example) command at the IDL prompt:

```
IDL> result=savephotons(nsamples=100,dt=1,filename='c.100.1.dat')
```

(If an error message occurs, "error making connection", you may establish the connection to the PMT hardware by typing in a linux window (not IDL):

```
% telnet
% open 10.32.92.5
% use "ugastro" (w/o quotes) as login name
% enter the password here
```

```
$ startnet
$ exit
```

The password is the namesake of the lab with a "4" for "a": Look on the main whiteboard in the lab room. It starts "Cud4..." .)

The command, *savephotons*, causes the PMT to make multiple observations of the light source. The number of observations is "nsamples" (i.e. 100 observations in the example). The value of *dt* is the "integration time", the duration of each of the observations, expressed in millisec (ms). The *filename* indicates the name of the file where the data are stored. (Note that the word "data" is plural.) The filename itself should contain the information about the data, notably the number of observations (100) and the integration time (1 ms). The output is "result" which will equal 1 if the data were saved successfully and will equal 0 if there was a problem saving the data. You should get a message that says:

```
Please wait, Acquiring Data
Data saved in file:   c.100.1.dat
```

The file *c.100.1.dat* contains one column of 100 numbers, each representing the number of photons that were detected during each integration time (1 ms). At this time interval of 1 ms, you should get a few counts (or more, but not more than 20) per observation from the Light-Emitting Diode (LED) on and the brightness knob turned to about 11 O'Clock.

Play around with the integration time (i.e., try longer integration times of 2 ms and 10 ms) and convince yourself that the integration time of the observation (from the resulting counts detected) is proportional to the *dt* you specified. Remember to save each new *dt* as a separate data file.

2.1 Reading the PMT Data into IDL

You now need to read into IDL the counts that are in the output file. View your output file using EMACS and notice the format. It is a single column of numbers. There is an IDL program that reads a file into computer memory and labels it with an IDL variable name. In this example the IDL variable is an array of numbers (measured counts) and is called "count_array":

```
IDL> readcol, 'c.100.1.dat', count_array
```

Notice that the file name must be enclosed in single quotes. Any IDL variable between single quotes are called *strings*, i.e. text. Once you have the counts in an IDL array you can plot all of the counts using the IDL command "plot" (see the IDL handout within the lab website for help with IDL commands). Label the axes. Experimental data points should be plotted as single points, not connected with a line. (Theoretical curves are usually displayed as lines.)

3 Statistics

Inside your IDL program, calculate the mean and standard deviation of the number of counts that you measured in the set of 100 integrations (each integration being 1 ms). The IDL functions *total* and *stddev* may be useful. Make a histogram of the 100 counts. You may want to use the IDL command *plothist* to make your histogram.

```
plothist,count_array,bin = whatever size you want
```

A histogram shows the number of occurrences, N, with which a certain number of counts occurred, within some range (bin). Include one histogram in your write-up, noting the mean and standard deviation.

Repeat this experiment six times and compare the results of your experiments. (You only need the one plotted histogram, not all of them.) For each of the six trials, calculate the mean and standard deviation of the six count rates you just measured.

Questions: Do you always get exactly the same mean number of counts per observation? Why not - Explain? Do you get approximately the same standard deviation in each of the six sets of 100 observations?

3.1 Increasing the Number of Observations

Now repeat with more integrations, i.e., increase the number of integrations by a factor of 10, to 1000. But keep the integration time the same as before.

```
IDL> result=savephotons(nsamples=1000,dt=1,filename='c.1000.1.dat')
```

Again, take six sets of data and repeat the exercise of calculating the mean and standard deviation for each longer set of data. Report your results (mean and standard deviation) in a table.

What do you notice about the mean count rate and the standard deviation for these sequences? In what way, if any, is the ensemble of measurements different when you take 1000 observations compared to 100 observations? Explain with careful words, in a few sentences.

3.2 Longer Integration Times: Effects on the Mean & Standard Deviation

Let's get to the root of where these variations come from. Let's show that there is a relation between the number of photons counted and the standard deviation of them. Take a sequence of data with increasingly long integration times, e.g.,

```
 nsamples=100 dt=1 filename=c.100.1.dat
 nsamples=100 dt=2 filename=c.100.2.dat
 nsamples=100 dt=4 filename=c.100.4.dat
.
.
```

.

```
nsamples=100 dt=256 filename=c.100.256.dat
```

Calculate the mean number of counts and the standard deviation for each of the 9 sequences. Suppose "Nbar" and "s" are the arrays containing the means and standard deviations. Plot σ vs. Nbar:

```
IDL> plot, Nbar, s, psym = 5
```

to make a plot of the standard deviation, s, as a function of the mean number of counts, $Nbar$.

Now we can test Poisson Statistics, notably its prediction that the standard deviation, s, of the number of counts is simply the square root of the number of counts, Nbar. To test this, overplot that hypothetical expectation, $\sigma = \sqrt{Nbar}$, as follows:

```
IDL> oplot, Nbar, sqrt(Nbar)
```

What does this tell you about the relation between the mean and the standard deviation of the arrival of photons. Does the prediction of Poisson statistics appear to hold true?

3.3 Poisson Statistics

As you now know, the statistics for events of low probability are called *Poisson statistics*. Most "countable" sets of data fall within this statistical regime. Our PMT data are countable data and thus require Poisson statistical analysis. Plot a histogram for a data sequence that has a very small count rate, around 2–4 counts per sample and lots of samples, e.g., 1000. (You might try using integration times smaller than 1 ms.) The histogram is a plot of the number of integrations (among 1000) that had 0, 1, 2, ... counts, vs. the counts detected. Calculate the mean number of counts, called μ, per integration. Now compare the histogram of the number of counts with the theoretical Poisson distribution,

$$P(x, \mu) = \frac{\mu^x}{x! \cdot e^\mu} \qquad (1)$$

where x is the number of counts, P is the probability that x counts occurred, and μ is the mean number of counts per sample.

In IDL, the command might look something like:

```
IDL> p = mu^x / ( factorial(x) * exp(mu) )
```

Use IDL's OPLOT function to compare the observed histogram to that prediction from the Poisson distribution. The Poisson distribution gives a probability. You now have a measured histogram of counts.

Explain carefully in your report how to choose the correct scaling factor to compare the measured and theoretical distributions. Does the Poisson distribution provide a good description of the data?

3.4 The Regime of Many Counts: The Gaussian

Now take a different set of data obtained with a longer integration time, one that yields more than ∼100 counts per integration instead of only 2 - 4. Aim for several hundred counts per sample.

Plot the histogram again (and put it in your write-up). Describe in a sentence or two what has happened to the shape of the histogram. Calculate the mean and standard deviation, and over-plot (with a solid line) the corresponding Gaussian probability distribution,

$$P(x, \mu, \sigma) = \frac{\exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)}{\sigma\sqrt{2\pi}} \tag{2}$$

In IDL, your command might look something like:

```
IDL> p = exp(-0.5 * ((x-mu)/sigma)^2)/(sigma * sqrt(2 * !pi))
```

Give an explanation of the regime (i.e. range) of counts per observation in which a Gaussian curve is an adequate description of the distribution rather than the Poisson distribution, thereby avoiding the factorial function?

3.5 Standard Deviation of the Mean

The more observations you make the more accurately you can measure the average number of counts per observation. To illustrate this effect, take data with a given number of observations, say 16. Choose a fixed integration time, say 1 ms.

Calculate the mean number of counts per observation in the 16 integrations, and calculate the standard deviation, σ.

Because of statistical fluctuations in the arrival of photons, the mean number of counts per observation is itself uncertain. Its uncertainty is the "uncertainty of the mean," given by:

$\sigma_{\text{Mean}} = \sigma/\sqrt{N_{obs}}$

Here, N_{obs} is the number of observations, 16. This σ_{Mean} is the uncertainty in our knowledge of the average counts per observation. What is the fractional uncertainty in the mean, $\sigma_{\text{mean}}/\text{Mean}$?

How does the σ_{Mean} vary with the number of integrations, i.e. would we benefit by taking more than 16 integrations? Intuition suggests that if we take more than 16 integrations, the uncertainty in the mean will be smaller.

To quantify this effect, repeat this experiment with different number of integrations, namely: 4, 8, 16 (already done), 32, 64, 128, 256, 512, 1024, and 2048. Don't vary the integration time (which you might set at 1 ms, or whatever you prefer).

For each trial with a different number of integrations, calculate the mean, the standard deviation, and the standard deviation of the mean. *Plot each of these three quantities as a function of the number of integrations, N_{integ}.*

Describe carefully in words how these three quantities vary as the number of integrations increases. How does the standard deviation of the mean vary with the number of integrations, i.e., with what functional dependence on N_{integ}? You may want to use the OPLOT function to compare your suspected functional dependence with the data. If you want to improve the accuracy of a measurement of the mean by a factor of two, by what factor do you need to increase the number of observations?

4 Dark Counts

Finally, measure the dark count rate of the PMT in counts/sec. To do this, turn the brightness knob to zero (counterclockwise). Explain in words carefully, to what extent have dark counts contaminated (fractionally) your measured count rates in the experiments above? Give a one paragraph discussion of areas of the experiment in which dark counts did not enter significantly and other areas where dark counts may have significantly affected your results.

5 Appendix: Tips

Since the data are taken through an IDL procedure think about writing a script in IDL to acquire multiple data sets automatically. This will save you lots of time and effort. A script in IDL is simply a text file containing the text that would otherwise be entered at the IDL command prompt with an additional *END* at the end. Once you have a script you can run it in IDL with the following command:

```
IDL> .run myscript.txt
```

where *myscript.txt* is the name of the script. For more advanced users, procedures and functions can be written to automate the data collection process.

Sometimes it is important to plot data in multiple plot windows. IDL allows users to do this plotting by setting the IDL plotting variable

```
IDL> !p.multi = [0,2,3]
```

to present an array of 2x3 plots on a single window (or page), otherwise they will flash by too quickly to see. To get back to normal plotting set

```
IDL> !p.multi=0
```

To create postscipt images of the IDL plots, use the command *psopen* and *psclose* to redirect the plotting to a postscript file rather than the plotting window on your screen.

```
psopen,'filename'
plot,x,y
psclose
```

There are lots of resources available both online and in the lab when questions arise about IDL and this experiment. Just look and/or ask.