Problems and Projects for the DIY Geiger Lab

Author: Areg Danagoulian

Dear students, during the one-week long course we worked on a number of topics related to nuclear detection: nuclear physics, statistics, electronics, and hands-on lab work. Here I describe a number of problems that could help you further improve your understanding of the area, as well as develop your own ideas for future projects involving nuclear detection with your Geiger counter. In the end I propose a few projects for you to think about…

Statistics

In class we discussed error propagation. The following problems will help you better understand how to use error propagation in order to determine the uncertainties on the various measurement results.

Problem S.1

The expression for the standard deviation σ_f of function $f(x_1, x_2,..., x_N)$ is $\sigma_f^2 = \sum_{i=1}^{N} (\sigma_i \frac{\partial f}{\partial x_i})^2$, assuming no correlations between $\{x_1, x_2,..., x_N\}$. Here σ_f is the *uncertainty* or *error* on f (i.e. the envelope where the values of f fall with the probability of 68%), while σ_i are the *uncertainties* or *errors* for variables $\{x_1, x_2,..., x_N\}$. The following proofs are very useful to derive and to remember.

1. Prove that when $f = x \pm y$ the uncertainty on f can be determined by adding the errors in quadratures, i.e. $\sigma_f^2 = \sigma_x^2 + \sigma_y^2$
2. Prove that when $f = kx$, where k is a constant, $\sigma_f = k\sigma_x$
3. Prove that when $f = x \cdot y$ OR $f = x/y$ the *relative uncertainties* add up in quadratures, i.e. $[\frac{\sigma_f}{f}]^2 = [\frac{\sigma_x}{x}]^2 + [\frac{\sigma_y}{y}]^2$

Problem S.2

You want to measure the counts/min emitted by a source. However the source is not very strong, and you realize that a significant part of your counts will be from the background (i.e. other sources, such as gammas from the concrete walls and muons from the sky). So you devise an experiment:

a) You first measure the background counts with the source absent, say, for 1 minute, and get counts c_{bg}.
b) Then you add the source, and measure the counts, while realizing that what you measure is the *sum* of source and background counts. You denote this as c_{total}
c) You then subtract the two, and extract $c_{source} = c_{total} - c_{bg}$.

You are happy, however you realize that c_{source} is just an *estimate* of the source intensity. The real number could be somewhat different. You also remember that a professor from MIT told you that when you estimate a number you should ALWAYS also estimate the *uncertainty*, or the *error* on that number.
Questions:

1. Given the errors on total and background counts, σ_{total} and σ_{bg} derive the equation for σ_{source} using your results from Problem S.1.
2. Now let's work with real numbers. Let's say your 1 minute long measurements yielded $c_{total} = 40$ and $c_{bg} = 20$. Determine σ_{total} and σ_{bg} assuming Poisson statistics.
3. Now, calculate c_{source} and σ_{source}.

Problem S.3

In class we discussed that during a Poisson process the waiting time between consecutive hits is exponentially distributed. Remember, the Poisson formula for the probability of detecting x counts, when the expected mean is $\bar{x}$, is

$P(x) = \bar{x}^x \exp(-\bar{x})/x!$.

Now prove that the probability density function of the consecutive counts is distributed as:

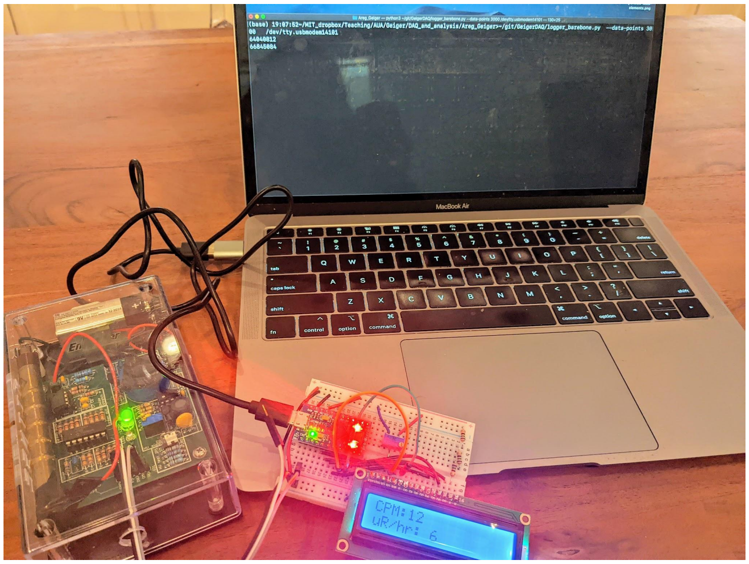
$p(t) = r\exp(-rt)$

Solution at the end of this document.

Data Acquisition(DAQ)

Using Arduino…

There are also much better and more precise ways of doing this.



Areg Hovhannisyan(areg_hovhannisyan@edu.aua.am) has led an effort to use Arduino to read out the TTL signal from the Geiger module. The code for this can be found on https://github.com/ustajan/GeigerDAQ. The code contains:

- Arduino code (tested on Pro Micro and Nano Every) for
  - reading in the TTL
  - determining the counts-per-minute and the dose
  - Displaying those on a 16x2 LCD
  - Transferring it to the computer over the serial protocol
- Python code for reading in data from Arduino
- Python and C++ code for analyzing the data

Here you can watch a brief video describing the setup of the readout, using an Arduino Pro Micro (~$10). The LCD shown in the video is optional.

Statistical Experiments

You can use the simple python code above to register the time (in seconds) relative to the start of the run. For a start, I suggest you change the python code to run for at least 10 hours (overnight). Make sure you have a fresh battery, as this is a long time.
Once you have taken your data you can do a few interesting statistical tests.

Counting statistics

Take your data and split it in time windows such that on average you have only 3-4 counts. Remember, the probability that a particle registers with your detector is small. This means that Poisson statistics is a good model for your count distribution.
Break the 10*3600=36000 second long data into ~10 sec long windows. You should have on average about 3 counts/window. Assignments and questions:

a) histogram all your data.
b) What distribution you do expect? What distribution do you see?
c) What is its mean?
d) What is its standard deviation? Is your $\sigma^2 = \bar{x}$?

Now, change the window such that the average number of counts/window is ~50. Try 120 seconds.

a) What statistical model would you expect in this case? Why?
b) Histogram your data, and see if you can reproduce the expected statistics.

Waiting time

You can also do the following interesting experiment. As you go through your data, for every count calculate the time between that count and the preceding count. This is called the waiting time.
Histogram the waiting time. Questions:

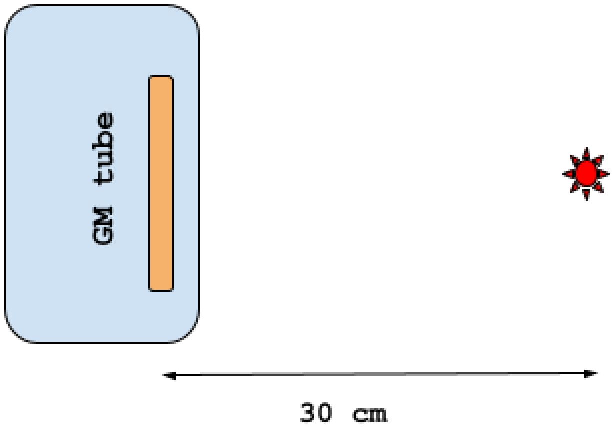
a) What shape do you expect it to have and why?
b) Do you get that shape when you histogram the data? Show your work.
c) What is the most likely waiting time?
d) What is the mean waiting time and how does it compare to the counts/seconds?

Physics Experiments

By now you've understood how statistics works, how to do error analysis, and how to set up your system to acquire data from the box. You can finally start setting up experiments of gamma absorption and gamma detection.

Efficiency measurements

Let's start off by quantifying the efficiency of our detector. What is efficiency? It's the probability that a particle crossing the volume of the detector will register a count. Let's take a calibrated 60Co source, and place it at a distance of 30cm from the GM tube of our detector – perpendicular to the tube.



Acquire data until you have at least 1000 counts. From here now you can determine the efficiency of your GM tube. Remember, you can determine the number of counts via the following formula:

$C_s = It\epsilon A/4\pi R^2$ where I is the intensity of your source (counts/second emitted), t is the measurement time, A is the cross-sectional area of your GM tube (you can find it here), ϵ is the actual efficiency that you are trying to find out, and R is the distance between the source and the tube. ϵ is the probability that if a particle crosses the volume of the detector it will cause a signal.

Now, everything would be very simple, except for one problem: your detector isn't just measuring gammas from the source, it's also measuring gammas from the background! This means that you need to know what your background is. How do you do this? Well, you just remove the source and measure the background alone for exactly time t! Let's call it C_{bg}.

So the counts that can be attributed to the source are $C_s = C_{total} - C_{bg} = It\epsilon A/4\pi R^2$.

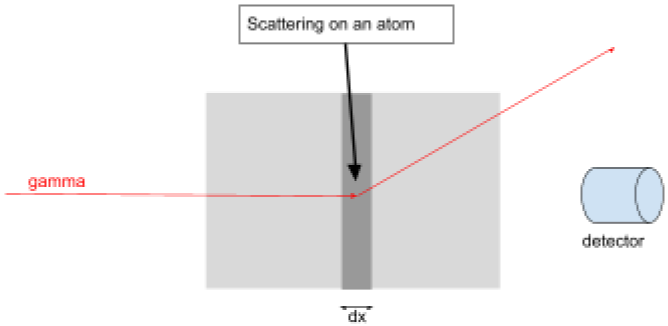
Now find ϵ and determine the uncertainty on it (using your work from Problem S.2).

Attenuation Measurements

What happens to gammas when they interact with the surrounding materials? At the ~MeV energy scale there are three primary processes: photoelectric effect; Compton scattering; pair production (>1.022 MeV).

Cross section

Let's assume that the combined cross section for all these interactions is σ (note, this is different from st. deviation!). Remember, cross section is in the units of area, and it denotes the "effective area" of the scatterer. It typically has the unit of 1 barn = 10^{-24} cm^2. This may sound like a tiny number, however given how many atoms there are, 1 barn ends up being a very large number.



Let's consider the thin sliver of material, dx, as depicted in the image above, where the scattering happens. You want to know the total area covered by all the atoms in the sliver, if one atom has an effective area of σ. If the material density is ρ, and the atomic weight of the material is A, then the total area covered by all the atoms is the total number of atoms X σ. The total number of atoms is volume of the sliver times density times Avogadro's number divided by atomic weight. So the total area is

$$Atomic\ area = \sigma \cdot YZ \cdot dx \cdot \rho N_{Av}/A$$

Where Y and Z are the height and the width of your material. And the probability that a random particle streaming through the material will interact with one atom is just the ratio of this "Atomic area" and the total area, hence:

$$dP = [Atomic\ area]/YZ = dx\,\sigma\rho N_{Av}/A$$

We next want to ask ourselves: if N number of particles are shot at the material, how many of them will interact in the thin sliver dx?

$$\#\ of\ Interacted\ particles = NdP = Ndx\,\sigma\rho N_{Av}/A$$

But remember, the change in N, i.e. dN, is exactly equal to the # of interacted particles, with a minus sign (# of particles is positive, while the change in N is negative). So we get the following first order homogeneous differential equation that governs the number N:

$$-\,dN = N\,dx\,\sigma\rho N_{Av}/A$$

Here we denote $\mu \equiv \sigma N_{Av}/A$, and we call it the mass attenuation coefficient. The new equation takes the following form:

$$-\,dN = N\,dx\,\mu\rho$$

By moving some variables around we get $\frac{dN}{N} = -\,dx\,\mu\rho$, which we integrate for $0 \le x' \le x$ and $N_0 \ge N' \ge N$, and it gives us

$$\frac{N}{N_0} = \exp(-\,\mu\rho x)$$

Note that μ strongly depends on the material *and* the energy of the gamma. You can look up values of μ on the XCOM database at the National Institute of Standards and Technology (NIST).

Ideas for projects involving the detector

Once you have a working Geiger counter instrumented with the DAQ, there are a number of interesting projects that you can work on.

Project 0: instrument a small drone / UAV with a Geiger Counter and perform a radiological search

Imagine there is a radioactive source in the field. How would you find that source? This search problem is a very important one, and is at the forefront of much engineering research. Just to give you an idea of the impact, imagine a nuclear accident, where radioactive material is scattered around the environment, and you are tasked with mapping out the main hot-spots of radiation. This would allow for civilians to avoid those, and clean up crews to focus on their removal. Given Armenia's dependence on nuclear power at Metsamor, this is a very relevant problem for your country.

These days there are many small, commercial UAVs that can be programmed and operated, e.g. with an on-board arduino. You could attach a smaller version of your Geiger counter to the UAV, and feed it information of the observed count rates, thus allowing the drone to fly in a raster pattern over a field and map out the radiation.
You could also approach a more difficult problem, and write a search algorithm which reads the signal and sends instructions to the drone for modifying its search pattern to focus on a particular hot-spot.

Project 1: instrument a bluetooth chip on your Geiger counter

This will allow your counter to send data to your phone. You could then write an app which reads in data, determines your GPS coordinate, and uses those to map out the radiation in Yerevan as you are spending the day walking around the city. What if there are some hot-spots of radiation around your daily commute?   This way you can find that out.

Project 2: instrument a Raspberry Pi with a web server to publish the real-time radiation data

Wouldn't it be great if we had radiological stations that could, in real time, always show the level of radiation, temperature, humidity etc? You could take the PS signal from your Geiger Counter to one of the pins on RasPi's GPIO, along with other (cheap) sensors' data, and have those plotted on a server in real time. That way people could remotely see the environmental conditions around your neighborhood.

Project 3: send your Geiger counter to outer space!!

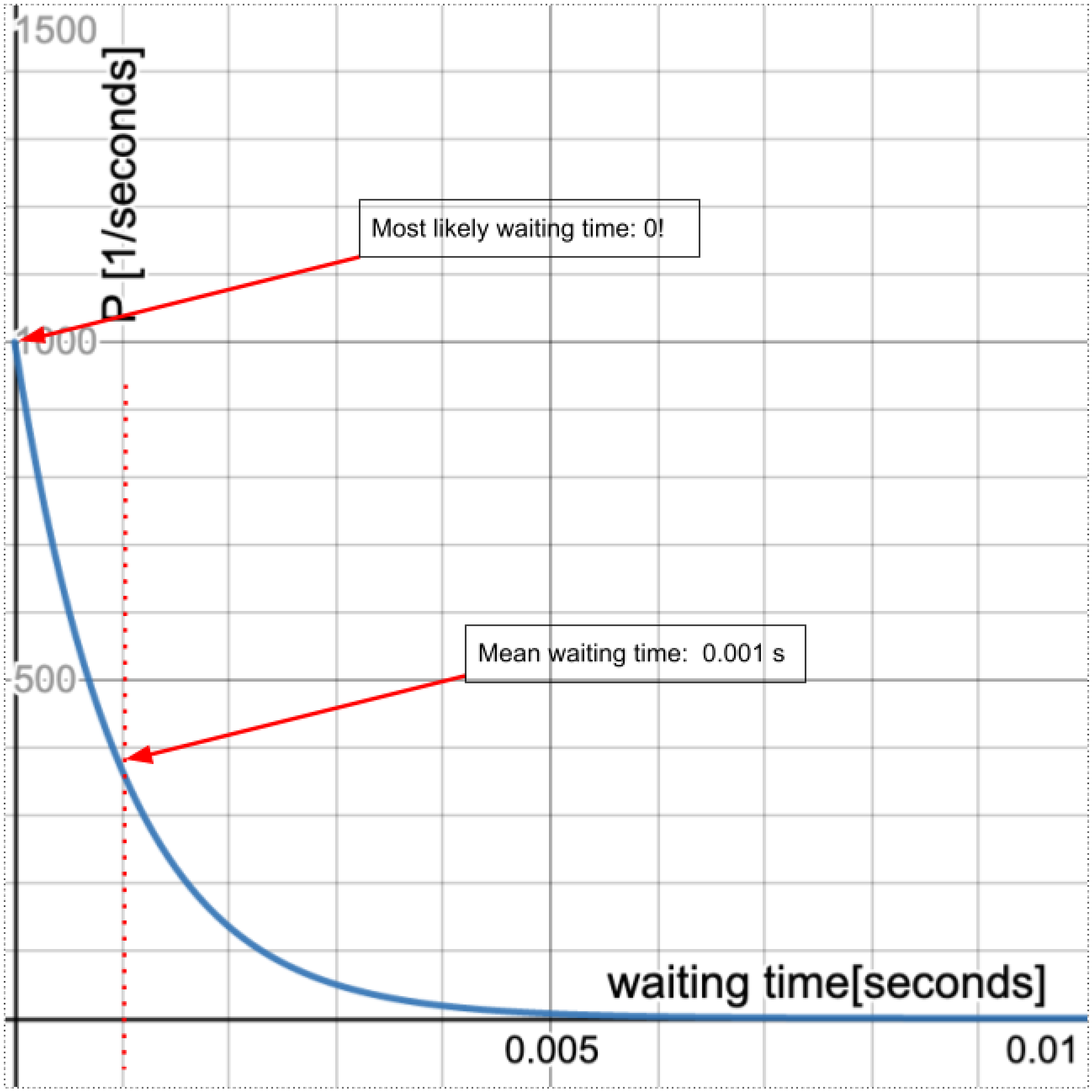
These days weather balloons are quite cheap. And so is cellular transmission. You could instrument your Geiger counter with a device that, using cellular transmission, sends out the data. You could then attach it to a weather balloon and send it to the upper atmosphere. This way in real time you could measure the height of the balloon (using a pressure sensor / altimeter), it's position (using GPS), and the radiation rates. As the balloon rises >1km most of terrestrial radiation goes away (due to absorption by the air), and all you observe from then on is the muons and other cosmic rays. As you rise you see higher and higher rates. What is the dependence between the muon rate and the altitude?
Here's a very nice DIY project that you could pursue:
https://www.instructables.com/How-to-Construct-a-Weather-Balloon/

Other projects

See https://hackaday.com/tag/radiation/ for more fun ideas!

Solutions

SOLUTION to Problem S.3:
If you had a count, what's the probability of having no counts for time t, and then a count in the tiny window dt? This may be confusing, because with the formula above we are used to thinking of x as a discrete variable (one count, two counts, but no 1.324 counts…)...however the time t is continuous. So when it comes to the distribution of waiting time we are working not with probabilities, but rather with *probability density functions*. Let's denote this probability density as $p(t)$, which tells us the probability density of having a 2nd count at time t. The actual probability is then $p(t)dt$. A 2nd count time t away requires two things: no counts for time t, and then a count in the time window dt right after that. The total probability should be the product of these two.
Thus we can write
$p(t)dt = P(x = 0) \cdot rdt$
where r is the rate of counts per second. To be clear, if you know your rate then the number of expected counts within a time window dt is rdt. Similarly, the expectation $\bar{x}$ of counts in the time window t preceding that is $\bar{x} = rt$. Let's plug these, and $x = 0$ (no counts for time t) into the Poisson's equation $P(x) = \bar{x}^{x} \exp(-\bar{x})/x!$. This gives us
$p(t)dt = P(x = 0) \cdot rdt = ((rt)^{0} \exp(-rt)/0!) \cdot rdt = r\exp(-rt)dt$.
The dt's cancel, and we get
$p(t) = r\exp(-rt)$
Below is a plot of the distribution for the value of r=1000 counts/second.


1500
P [1/seconds]
Most likely waiting time: 0!
1000
Mean waiting time: 0.001 s
500
waiting time[seconds]
0.005
0.01