THESIS

SIMULATION OF RADIOLOGICAL BACKGROUND DATA FOR BENCHMARKING STATISTICAL ALGORITHMS TO ENHANCE CURRENT RADIOLOGICAL DETECTION CAPABILITIES

Submitted by

Michael A. LaBrake

Department of Environmental and Radiological Health Sciences

In partial fulfillment of the requirements

For the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Fall 2017

Master's Committee

Advisor: Alexander Brandl
Co-Advisor: Thomas Johnson

Sandra Biedron

ABSTRACT

SIMULATION OF RADIOLOGICAL BACKGROUND DATA FOR BENCHMARKING STATISTICAL ALGORITHMS TO ENHANCE CURRENT RADIOLOGICAL DETECTION CAPABILITIES

Reliable, trustworthy, and sensitive methods for detection of nuclear material are essential, particularly when the material is a weak or shielded source. Since weak or shielded sources often require longer measurement times to distinguish between the source distribution and the background distribution, a statistical approach is being developed that will utilize prior information obtained by measurement systems, such as portal monitors, which collect data on a continuous basis. The hypothesis is that patterns can be identified in sequences of repeated count rate measurements and used in conjunction with classical statistics to identify and locate a source. By measuring background distributions and establishing standard data for specific locations, it is possible to use the probability of observing individual measurement results in successive measurement intervals at or above the decision threshold, y^*, and use this information to help pinpoint a weak or shielded source. Because radioactive decay and detection of radioactive material are stochastic events, pseudo-random numbers are used in conjunction with a mathematical method that enables simulation of various background distributions. The related y^* values are calculated for each distribution and repeated measurements at or above y^* are counted and compared to those expected for the given distribution. Four distributions were investigated: the triangular, sinusoidal, normal, and Poisson distributions. For each distribution, large random number samples were generated to confirm the expected probabilities for various sequences of values at or above the decision threshold y^*. All investigated sequences found the 95% confidence interval for the expected number of sequences greater than y^* to include the observed number of sequences greater than y^*.

TABLE OF CONTENTS

Introduction

Background

Detection of nuclear materials is an important and necessary scientific as well as societal endeavor in modern society. With many nations around the world developing an interest in nuclear technologies, both for benign purposes and militaristic ones, measuring radioactive materials is more important now than ever. Further, the possibility of non-state actors acquiring means to initiate a nuclear attack is more credible now than in the past [1].

Since the materials used in nuclear applications are typically alpha-emitting radionuclides, detection of clandestine use and deployment of those materials can prove difficult. Other obstacles in the positive detection of radioactive materials include external shielding of the material in containers, self-shielding by the radioactive material, cosmic background, and similar emissions in natural background. The obvious approach is to develop more sensitive technologies capable of measuring the low-energy and statistically infrequent decays associated with those radioactive materials. The simplest method of increasing event detection is to increase the size of the detector; however, it has been suggested that merely increasing detector size does not improve signal-to-noise ratios for distant sources [2]. Although the effect of shielding a source is less pronounced than increased distance, the idea remains the same. By increasing the distance to or shielding a source, the efficiency of a detector is reduced, and a logical conclusion is to increase the detector's size to collect more radioactive emissions in a similar time interval. Since the physical size of a detection system might be limited by the engineering or economic constraints, a more refined statistical approach is being investigated to enable measurement of low-activity or shielded nuclear sources using current detector technologies.

Radioactive decay is, by nature, a stochastic process. Each radioactive nuclide, called a radionuclide, has an associated half-life, after which half of the atoms originally present have decayed into another element. A detector measures a snapshot in time of the decay process. A device can be used

to count the number of radioactive decays over a short time interval, which can be related to the intensity or amount of radioactive material of the source, or even used to determine the exact radionuclide present in some circumstances. A detector works by measuring charge as a result of the particle or wave that enters the sensitive region of the detector. Different kinds of detectors can measure different types of radiation, which include alpha particles, beta particles, and gamma rays. Alpha and gamma emissions are monoenergetic events, which means the particle or gamma ray is always emitted with a unique initial energy. Beta decay is the release of an electron or positron accompanied by a neutrino, and the energy of the beta particle varies on a spectrum, which means the sum of the beta and neutrino energies will always be the same for every emission, but the amount of energy each possesses individually is subject to the stochastic sharing of energy and momentum between the two particles resulting from the decay.

Detection of Radionuclides

When a nucleus decays, a detector can be used to capture the energy of the emitted particle. By measuring the amount of energy deposited in the detector, and knowing the type of particle the detector is designed to measure, the radionuclide from which the particle originated can be determined. Due to the different designs required, a single detector is usually poor at detecting all types of radiation, although a suite of detectors may be used to detect several types of radiation. This can prove to be a benefit since certain particles will not be expected to enter a given detector. For example, ionization chambers are designed to measure primarily gamma radiation, but can be sensitive to beta and alpha radiation as well. By designing the chamber casing to be sufficiently thick, all alpha particles can be blocked entirely. This means that any event measured in the detector cannot be from an alpha particle. In a similar fashion, a beta window can be designed for the unit. When the window is open, beta particles pass more easily through the detector wall at that location, and the detector will reflect the increase in energy deposition. By closing the window, beta particles can be mostly stopped before entering the chamber, resulting in the majority of readings coming from gamma rays. Thus if an unknown radionuclide is measured with an ionization chamber with the beta window closed but events are still observed, the radionuclide must be emitting gamma radiation.

Radionuclides which decay by gamma emission emit monoenergetic photons, with the energy of the gamma being characteristic of the radionuclide releasing the gamma ray. However, gamma rays are not emitted by all radionuclides, nor are they a completely independent event. Instead, radionuclides usually emit alpha or beta particles prior to the release of a gamma ray. Sometimes, a nuclear reaction occurs when a nucleus captures another particle (neutron or gamma typically) and subsequently emits another particle of the same or different type. When an alpha particle is released, the atomic number of the element decreases by two, and the mass number decreases by four. This is because an alpha particle is comprised of two protons and two neutrons; an alpha particle is essentially a helium ion. In some instances, after the release of the alpha particle, the resulting nucleus is left in an excited state. This can be thought of as excess energy leftover in the nucleus that is simply released as photons as the nucleus reaches the non-excited or ground state. Similarly, the nucleus can be left in an excited state after the emission of a beta particle. Releasing a beta minus transforms a neutron into a proton, since the net charge on the neutron becomes more positive. Conversely, the release of a beta plus, or positron, transforms a proton into a neutron. As mentioned before, beta particles have a range of energies on release, but the new nucleus after transformation may still be left in an excited state, reaching the ground state by emission of a gamma ray.

Owing to their monoenergetic nature, alpha particles and gamma rays are generally more useful in determining the radionuclide of origin than a beta particle. However, since alpha particles are easily blocked, they are not a useful means to evaluate the nature of a radionuclide in a situation such as a checkpoint or port of entry where a detector is used to scan passing vehicles and determine whether they are carrying potentially hazardous radioactive materials. Alpha particles tend to exhibit energies on the order of several mega-electron volts (MeV; 1 MeV = 1.60218×10^{-13} J), and many radionuclides commonly release alpha particles on the order of 4 or 5 MeV. Two alpha particles of similar energies but not exactly the same can produce peaks on a spectrum that cannot be resolved; that is, the difference between the energies of the two alphas is so small that it cannot be measured on a typical alpha detector. For example, ^{239}Pu and ^{240}Pu both emit alpha particles at about 5.1 MeV and differ in energy by only

about 12 keV, and the resolution of a typical alpha detector is on the same order of magnitude or larger [3]. A similar phenomenon can occur with gamma emission; if the two particles emitted possess similar energy, and a detector with poor resolution is used, it is impossible to know which is present without prior knowledge or some other indicator, such as chemical properties or appearance. For instance, observation of their chemical behavior (adding a compound suspected to cause a reaction) or simply noticing a difference in color or state at the temperature observed (liquid versus solid) can provide additional insight as to the identity of the radionuclides present.

Fortunately, many of the radionuclides of concern emit a gamma ray in addition to an alpha particle. A measurement of an unknown substance is taken, and the measuring instrument is an ion chamber or sodium iodide detector, intended to measure gamma radiation. Both ^{239}Pu and ^{240}Pu emit gamma rays; however, the probability of this occurring is extremely low at 3.14×10^{-4} per decay for ^{239}Pu and 4.47×10^{-4} per decay for ^{240}Pu for the most probable gamma emissions for each element [4]. Still, the energies of the gammas are sufficiently different that if they were measured, ^{239}Pu and ^{240}Pu could be distinguished and their presence confirmed or rejected. To illustrate, the energy for the most probable gamma from ^{239}Pu is 1.298×10^{-2} MeV, compared to the most probable gamma for ^{240}Pu, which is 4.524×10^{-2} MeV [4]. Note that in the case of these two radionuclides the gamma energies are low enough that detection with the kind of equipment typically used at a portal monitor (ion chamber or sodium iodide detector) may prove difficult or impossible, especially over such a short time interval. The amount of material and time of measurement are both important factors; however, for certain radionuclides (Pu included) the mass number is high enough that any low-energy gamma rays will be self-shielded, which occurs when the mass of a radionuclide is high enough that emissions by the innermost atoms in the volume containing the radionuclide deposit all their energy before exiting. Thus more material will not necessarily result in a higher number of measurable gamma rays under all circumstances; greater surface area of a source would decrease self-shielding, for example. For certain gamma-emitting radionuclides, specifically ones with sufficiently high energy, more mass in general may increase the probability of detection. This is a result of the stochastic nature of radiation and radiation measurements; larger sample

sizes improve the chance that an observer will witness a low-probability event. In the case of many heavy radionuclides, if the gamma emitted possesses low energy, time of measurement is often the most important factor, since longer measuring times will result in better signal-to-noise ratios. However, long measuring times may be prohibitive due to costs and feasibility of setting up such measurement systems.

A similar consideration would apply for radionuclides that decay by beta emission. Since a typical beta can be easily blocked by low-density materials such as aluminum or wood, if one desired to detect the presence of beta particles through such a material, measurement of associated gammas again might be more easily accomplished. As an example, while ^{137}Cs itself emits a beta particle and decays to ^{137m}Ba, it is the ^{137m}Ba which emits the characteristic gamma ray which commonly is associated with ^{137}Cs. If an appreciable amount of ^{137}Cs were on a truck en-route to some target location, and the truck were to pass through a portal monitor intended to detect radioactive materials, the beta emission would be of little use since designing external shielding would be simple enough. However, without adequate shielding for the gamma, the ^{137}Cs could be detected and the truck investigated further.

Hypothesis

Two radionuclides of concern are ^{239}Pu and ^{235}U which are both fissile materials that, when present in sufficient quantities, can be used to create a weapon capable of inflicting damage over a large area. Other materials of concern include ^{137}Cs. Pu-239 and ^{235}U would likely be in a form intended for a fission device, while the ^{137}Cs would likely be used in a conventional bomb to create a dirty bomb and spread radioactive material. In this instance, if a vehicle carrying radioactive materials were to pass through a portal monitor, exact knowledge of the material is irrelevant. The important fact is that any radioactive material is present at all. However, if the vehicle possesses adequate external shielding, little radioactivity may be detected during a short measurement. Radiation portal monitor systems have been designed and deployed with measuring times as low as 150 ms [15]. With adequate shielding, no alpha or beta particles will enter the detector. Due to their charge, alpha and beta particles will deposit all their energy if sufficient external shielding material is present; i.e., they are completely shielded. However, gamma particles have no charge, and thus shielding relies on direct interactions of gammas with the

atoms in the external shielding material. For a gamma to lose its energy, it must interact directly with an atom, thus imparting some of its energy into the target atom. If very few gammas are released in a short measuring time, it is unlikely that any will be detected. However, if enough material were present, the same argument from earlier applies: that the probability of observing an event in a given timeframe increases with the number of times that event occurs in that timeframe. Thus if a substantial amount of radioactive material were present, it is more probable that some gammas will escape the shielding without interacting with any atoms. If sufficient gammas are released, a change in background may be noted; if enough excess gammas are present and are measurable by detecting additional interactions in a detector, further investigation may be warranted.

The hypothesis is that patterns and sequences can be identified in repeated count rate measurements and used in conjunction with classical statistics to identify and locate a source. The basis for this thesis is to supply a proof of concept of an algorithm intended to discriminate low signals from electronic noise and natural background [6]. Although some literature suggests that algorithm development is hindered by access to data and common performance metrics [7], the particular algorithm examined does not necessarily require access to classified data [7]. The proposed method requires only a long-term background radiation measurement to establish a baseline standard for assessment of the decision threshold, and subsequent shorter background measurements [6], neither of which poses a problem for laboratories already equipped with detection equipment. This thesis aims to provide support for an algorithm that examines the probabilities associated with observing events which exceed a threshold; if natural background is expected to generate a baseline count rate, then presence of a source may alter the background rate enough to suggest a shielded or weak source is present.

While the number of events counted in a gamma detector for gamma rays that escape the external shielding may still be relatively small, detection should be quantifiable when compared against a known or expected background count rate. If a particular measurement is conducted under conditions of adequate shielding and low measurement time, the probabilities of detecting such gamma rays are small. If extensive background radiation data are collected, and a vehicle carrying a certain amount of

radioactive material parks outside a checkpoint, the background with the truck can be compared to the expected background with no source. Using classical statistics, a decision threshold[1], denoted by the symbol y^*, can be established. The decision threshold accounts for the acceptable rate of false positive measurements, in units of counts per time, which means that some percentage of measurements will fall above the established decision threshold in the absence of a radioactive source [17]. Then two data distributions may be described; one which is typically due to background and one which is due to a source [16]. In the health physics literature, the accepted rate of false positive measurement, denoted α, often is 5% [16]. This false positive rate, defined as the number of events (count rates) above y^*, suggests even when no source is present, 5% of the time a given measurement will exceed the decision threshold. If a source is present, then the count rate may exceed the threshold the majority of the time. This depends on the false-negative rate, the exception being when the count rate falls below the false-negative threshold, which is statistically possible despite presence of a source. Finally, if a shielded or weak source is present, the threshold may not be exceeded notably or may not be exceeded at all. However, if this is the case, then the count rate can be compared to background to check for anomalies in the data. Because radioactive decay is a completely stochastic process, the previous decays have no bearing on whether another decay does or does not occur. This means that each decay is an independent event, and, consequently, counts taken over a time interval are also independent events; thus the probabilities of observing a specified count rate in sequential measurements can simply be multiplied to determine the likelihood of seeing the specified number of events occur sequentially. For example, if y^* is some count rate r per time t, then the fixed probability of seeing two measurements at or above the decision threshold is given by multiplying the probabilities of two independent events:

$$\alpha^2 = 0.05^2 = 0.025 \tag{1}$$

The same applies for three, four, and n measurement intervals, being 0.05^3, 0.05^4, and 0.05^n, respectively. Thus if prior knowledge of the distribution of background radiation is utilized, a statistical

[1] In the older literature, the decision threshold is called the critical level L_C. In MARLAP [5], this value was also called y_C.

approach can be used to decide whether a given sequence of count rates is probable under the null hypothesis that no source is present in the truck. If a radioactive material detector at a portal monitor were utilized on a parked truck and two successive measurement intervals were taken, the probability of those two measurements existing above the decision threshold is low and may warrant further investigation if both exceed y^*. The specific number of sequential count rates exceeding y^* would be determined for each portal monitor site, based on the sensitivity and type of equipment present. Background radiation and source distributions are shown relative to the y^* location in Figure 1.

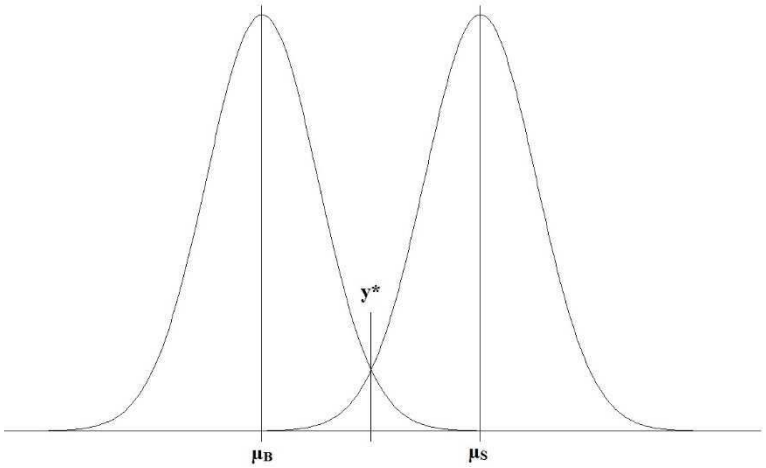

Figure 1: Comparison of underlying distributions for background radiation (mean μ_B) and source (mean μ_S). The location of y^* is also included.

The statistical concept described in the preceding paragraph can be expanded to include non-sequential events as well, or measurements taken in series where one lies above y^* followed by some number of measurements that fall below y^* before another measurement is taken above y^*. For example, suppose a detector measured two count rates above the decision threshold and one count rate below the decision threshold, in any order. A binomial expansion may be used to determine the probability of such an occurrence as follows:

$$P(X=x)=\binom{n}{k} p^k (1-p)^{n-k} \quad (2)$$

Where

n is the number of trials

k is the number of successes

p is the probability of success

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

In the context of Equation 2, a success is defined as observing one of the binary values supplied; that is, the number of times the desired outcome is generated. Substituting the values described above into Equation 2,

$$P(X=x) = \frac{3!}{2!(3-2)!} 0.05^2 \cdot (1-0.05)^{3-2}$$

$$P(X=x) = 0.007125$$

From this calculation, the probability of two measurement intervals providing a result above the y^* decision threshold and one below is about 0.007. This holds true for any order of two successes (i.e., results in a measurement interval exceeding y^*) and one failure (i.e., not exceeding y^*); the order does not matter as the binomial calculation merely computes the probability of seeing k successes in n trials.

If a vehicle were being inspected for radioactive materials, a recording of sequential counts per time might be generated. A program could be written to scan the recording for any measurement interval above the threshold, and then check some specified number of measurement intervals after, noting whether they are above or below the threshold. Some limit on the number of measurement intervals below the decision threshold in sequence would be imposed based on experimental data for that particular portal site. For instance, the program could scan and find the result in the first measurement interval above y^*, and then check the next two measurement intervals and determine their results are below the threshold. The program would abort that portion of the measurements and search for another event above y^*. If the program finds a measurement interval above y^* and scans the next two intervals and reports their results are also at or above y^*, and prior knowledge at the site suggests that the likelihood of measuring three successive measurement intervals at or above y^* is low, the vehicle could be detained and inspected for radioactive materials. A similar approach could be used in conjunction with the

binomial probability. A set of data in multiple measurement intervals would be collected and the number of times y^* is exceeded in a pre-determined sequence length (i.e., number of measurement intervals) would be counted and compared to expected probabilities at that specific site.

A waveform generator was used to create signals that subsequently were fed to a multichannel analyzer (MCA). The MCA records the input signals as simulated radioactive signals, enabling the user to simulate a radiation detector. A variety of distributions that can represent background distributions or source distributions was generated. Once these distributions were created, the probability of seeing successive count rates above y^* was calculated and compared with the number of times the successive events are actually seen in the distributions. The objective was a proof-of-concept of the methodology; if the probabilities are not consistent with expectations, then the method fails and must be adjusted to compensate. Both successive and binomial cases were considered, as described above. In all cases, the investigated sequences found the 95% confidence interval for the expected number of sequences greater than y^* to include the observed number of sequences greater than y^*.

Materials and Methods

Equipment

The equipment used included a Lynx Multichannel Analyzer (MCA) (Canberra, Arvada, Colorado) and a 1102D Arbstudio Waveform Generator (Teledyne LeCroy, Chestnut Ridge, New York). Additional equipment included a Bicron 2M2/2 sodium iodide detector (Bicron Electronics, Torrington, Connecticut) and a computer. The general approach was to determine the waveform the Lynx expected to receive, copy that format to generate a waveform of the same shape from the waveform generator, and finally use a random number generator to create varying pulse heights to simulate a detector output.

The Lynx requires a range of inputs between 5 mV and 10 mV and a gain setting that results in an output signal between 10 mV and 1000 mV. Signals below the 5 mV or above the 10 mV thresholds may cause the Lynx to register a count at an incorrect voltage; care should be taken to ensure the input voltages stay within the acceptable range. The input signal magnitude is extremely important to avoid loss of data. Additionally, signal shaping is paramount since the Lynx expects a specific shape of input and may not recognize waveforms of other shapes. Once the Lynx receives the expected pulse, the Lynx will reshape the pulse and output a new pulse with a different shape which is subsequently counted by the MCA. The Lynx checks the incoming pulses to determine whether a signal has been received or not; if the pulse is within the expected parameters, a count is registered and the Lynx tallies a count. After sufficient pulses have been measured, the Lynx allows for a graphical display of the tallies to generate a spectrum.

To determine the shape of the input pulse the Lynx requires, a sodium iodide detector was connected to the device, and a ^{137}Cs button source was used. The sodium iodide detector was first connected to an oscilloscope to monitor the outgoing pulse shapes from the detector. Subsequently, the Lynx was connected to the oscilloscope, and the reshaped pulses were viewed. At the same time, settings were adjusted in the software at a work station connected to the Lynx to ensure that the spectrum plotted

by the Lynx displayed the expected ^{137}Cs spectrum. If the spectrum was not of the expected shape, either the Lynx was not receiving the correct pulse shape or magnitude, or the settings on the Lynx were incorrect and must be adjusted until the spectrum appears as commonly shown in the literature [8]. Once the spectrum was satisfactory, the oscilloscope was connected to the monitor output on the Lynx to examine the shape after transformation to compare the output of the Lynx once it is connected to the waveform generator to simulate a detector. To properly simulate a signal the Lynx is capable of reading, the shape of the signal is important. The rise time should be on the order of 20 nanoseconds to a few microseconds with a fall time of 50 microseconds or greater. Further, the flat top should be set to be longer than the rise time.

After the requirements of the Lynx were ascertained, the Lynx was connected to the waveform generator in place of a detector. The oscilloscope was connected to the monitor output of the Lynx and the output of the waveform generator directly. Connecting the Lynx and the waveform generator to the oscilloscope enabled comparison of the output signals to those obtained previously from using the sodium iodide detector to ensure the proper signal shapes had been generated.

The waveform generator allows the user to create pulses of varying shapes and magnitudes up to several volts. However, because the default waveforms are not of the shape required by the Lynx, a custom waveform had to be designed. Using the output wave of the sodium iodide detector as a model, a multi-segment wave was assembled to replicate the shape and magnitude of the true sodium iodide detector signal. The wave used four main parts, including an approach, a rise, a flat top, and a fall. The true output shape of the sodium iodide detector is compared with the simulated pulse in Figure 2.

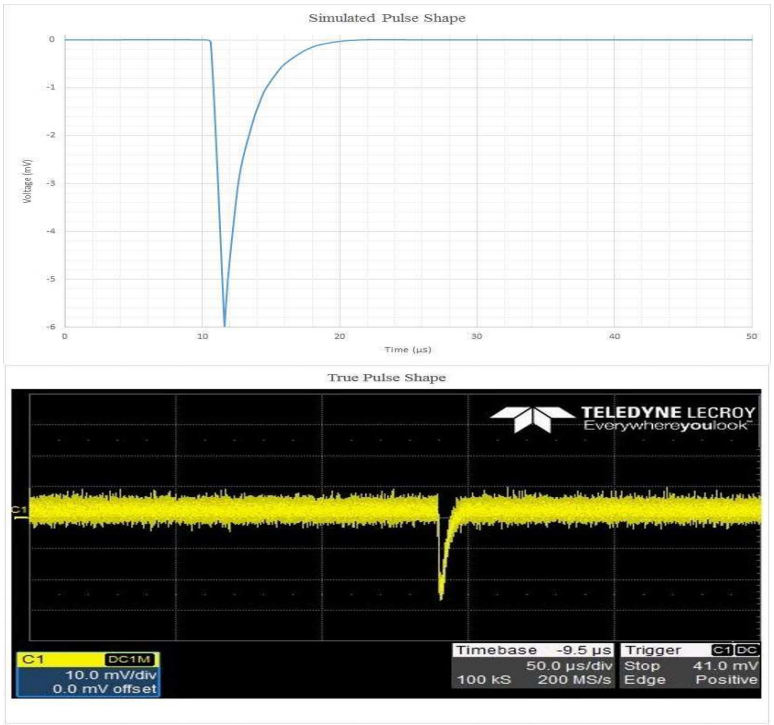

Figure 2: Comparison of pulse shape input to Lynx (top) against a true Pulse generated from a sodium iodide detector with a Cs-137 source (bottom).

The equation types used to generate the pulse displayed in Figure 2 are shown in Table 1. To vary the pulse magnitude, the slope of the rise and the magnitude of the exponential portion had to be varied. However, the form of the equations does not change beyond substituting the magnitude of the required pulse into the equations. Random numbers were the source of the varying magnitudes to simulate a detector measuring radioactive decays.

Table 1: Segments of pulse used to simulate a true pulse from a radioactive source. The approach segment gives a zero output to allow sequential signals to be discerned. The rise segment reaches the desired magnitude (output) of the wave. The flat top remains at the magnitude until an exponential decay returns the pulse to zero magnitude.

Component	Equation type	Duration
Approach	Linear	10 µs
Rise	Linear	600 ns
Flat top	Linear	1 µs
Fall	Exponential	200 µs

Random Number Generation

Perhaps the most important fundamental concept of this thesis is the random numbers used to simulate radiation. Since both radioactive decay and its measurement are stochastic events, each radioactive decay and measurement occur independently of the others. No true computational random

number generator exists, as all coding uses some sort of seed to generate its random numbers, called pseudo-random numbers [11]. However, pseudo-random numbers are acceptable for most applications, including those presented here.

Distributions

The objective of using the waveform generator is to simulate a real radiation detector measuring some source of radiation or background. Since radiation is a stochastic process, the number of counts from one measurement to the next may vary, sometimes substantially. If a number of measurements were taken and the counts plotted on a histogram, the resulting shape would be called the distribution of the source (or background). Thus if a measurement falls in the range of that distribution, within some tolerance, it is considered to be part of the distribution. Random numbers were used in tandem with equations that describe known distributions to create distributions of voltages intended to simulate possible radiation distributions, which in turn were used to create varied pulse heights in the waveform generator. The pulses were sent from the waveform generator to the Lynx to be counted.

Several distributions of voltage were considered. For distributions having a closed-form cumulative distribution function (CDF), a standard approach was used. First, the probability density function (PDF) was determined. From the PDF, integration yields the CDF. Finally, the inverse of the CDF was computed if the CDF was a bijection, which means the function is both one-to-one and onto; i.e., for each input, one and only one corresponding output exists (one-to-one) and over the entire domain of the function (all possible inputs) the CDF has a defined output (onto). The inverse CDF enables transformation of a uniform distribution into any desired distribution, simply by transforming the values in the uniform distribution to a corresponding value in the target distribution. For the functions having a non-exact or difficult CDF, a numerical method was used. Similarly, a discrete distribution was examined, and since the CDF for a discrete distribution is not continuous, another approach had to be found in lieu of the inverse-CDF method.

Uniform Distribution

The simplest distribution to generate is a uniform distribution. This is most readily done by using a pseudo-random number generator, and many computer software routines have this function built-in. The software package R [9] has such a built-in function, and that is used for the basis of the random number generation here. No equation is derived for the generation of the uniform distribution; rather, the uniform distribution served as a starting point for the creation of other, more complex distributions.

Triangular Distribution

The triangular distribution is the next simplest distribution. Often called the least-information distribution, two forms of the triangular distribution are considered. The first form follows the shape of an isosceles triangle, which means the areas to the left and the right of the median are equal. The second form considered is a scalene triangle. Both can be described by the same equation. A graphical representation is shown in Figure 3.

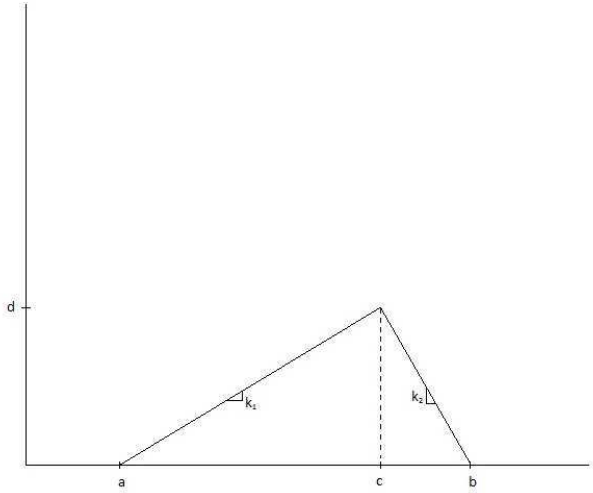

Figure 3: General shape of a triangular distribution, where *a*, *b*, and *c* can be any desired value.

To derive the inverse CDF equation, the PDF must be determined. The PDF is just the mathematical description of the function, or an equation that describes the infinite values the continuous random variable can assume. In the case of a triangle, the PDF consists of a stepwise function with two distinct components: the first equation represents the increasing portion of the triangle, and the second

equation represents the decreasing portion. Given a triangle that starts at point a, as shown in Figure 3, with end point b, median location c, and maximum function value d, the slopes can be calculated by

$$k_1=\frac{\text{rise}}{\text{run}}=\frac{d}{c-a} \tag{3}$$

$$k_2=\frac{-d}{b-c} \tag{4}$$

where k_1 and k_2 are the slopes of the increasing portion and decreasing portion of the triangle, respectively. Using the slope-intercept form to determine the equation of a line results in

$$\begin{cases} y_1=k_1x+e_1 & 0<x\leq c \quad (5) \\ y_2=k_2x+e_2 & c<x\leq b \quad (6) \end{cases}$$

where y_1 and y_2 are the function values of the increasing and decreasing portions of the triangle, respectively. For $x=a$, $y_1=0$, and thus the intercept of the function is

$$0=k_1\times a+e_1\rightarrow e_1=-k_1a$$

The same approach is used for finding the intercept of the second portion of the triangle:

$$y_2=k_2x_2+e_2 \rightarrow y_2(x=b)=0 \rightarrow e_2=-k_2b$$

To return the intercept in terms of known points:

$$e_1=\frac{-d\cdot a}{c-a}$$

$$e_2=\frac{-d\cdot a}{c-a}$$

Finally, by normalizing the area under the curve, d can be expressed in terms of b and a, reducing the number of parameters required by one:

$$1=\frac{b-a}{2}\times d\rightarrow d=\frac{2}{b-a}$$

Substituting the corresponding slopes and intercepts into equations 5 and 6 above,

$$\begin{cases} y_1=\frac{2x}{(b-a)(c-a)}-\frac{2a}{(b-a)(c-a)} & 0<x\leq\frac{c-a}{b-a} \\ y_2=\frac{-2x}{(b-a)(b-c)}+\frac{2b}{(b-a)(b-c)} & \frac{c-a}{b-a}<x\leq b \end{cases}$$

Factoring yields

$$\begin{cases} y_1 = \dfrac{2(x-a)}{(b-a)(c-a)} & 0 < x \le \dfrac{c-a}{b-a} \quad (7) \\ y_2 = \dfrac{2(b-x)}{(b-a)(b-c)} & \dfrac{c-a}{b-a} < x \le b \quad (8) \end{cases}$$

Equations 7 and 8 above represent the full PDF for an isosceles triangle (two sides of equal length) and work for the scalene triangle (each side has different length) as well, since all the parameters are general and contain the necessary information to shift the shape of the triangle. Thus the PDFs for both triangular forms are accounted for. However, y_2 is integrated from left to right and would require adding y_1 in its entirety to complete the CDF; therefore, the limits of integration will be switched and the expression will be multiplied by -1 to switch the direction of integration but maintain the positive area. The resulting expression will enable simple calculation of the total PDF by subtracting it from 1. Next the CDF was calculated by integrating the PDF:

$$\begin{cases} CDF_{y_1} = \displaystyle\int \frac{2(x-a)}{(b-a)(c-a)}\,dx & 0 < x \le \dfrac{c-a}{b-a} \\ \mathrm{CDF}_{y_2} = 1 - \displaystyle\int \frac{-2(b-x)}{(b-a)(b-c)}\,dx & \dfrac{c-a}{b-a} < x \le b \end{cases}$$

Completing the integration yields

$$\begin{cases} CDF_{y_1} = \dfrac{(x-a)^2}{(b-a)(c-a)} & 0 < x \le \dfrac{c-a}{b-a} \quad (9) \\ CDF_{y_2} = 1 - \dfrac{(b-x)^2}{(b-a)(b-c)} & \dfrac{c-a}{b-a} < x \le b \quad (10) \end{cases}$$

Equations 9 and 10 above are the final form of the CDF. From the form presented in Equations 9 and 10, the inverse CDF may be calculated. This will be done algebraically by switching the left hand side of the equation with the variable x in the right hand side of the equation, then solving for the original left hand side variable, as follows:

First, switch CDF_y with the variable x in Equations 9 and 10:

(11)

$$\begin{cases} x = \dfrac{\left(CDF_{y_1}\text{-}a\right)^2}{(b\text{-}a)(c\text{-}a)} & 0<x\leq\dfrac{c\text{-}a}{b\text{-}a} \\ x = 1\text{-}\dfrac{\left(b\text{-}CDF_{y_2}\right)^2}{(b\text{-}a)(b\text{-}c)} & \dfrac{c\text{-}a}{b\text{-}a}<x\leq b \end{cases} \tag{12}$$

To solve Equation 11, simply rearrange algebraically, take the square root of both sides, and move *a*, as follows:

$$x = \frac{\left(CDF_{y_1}\text{-}a\right)^2}{(b\text{-}a)(c\text{-}a)} \rightarrow x(b\text{-}a)(c\text{-}a) = \left(CDF_{y_1}\text{-}a\right)^2 \rightarrow CDF_{y_1}^{-1} = a \pm \sqrt{x(b\text{-}a)(c\text{-}a)}$$

To solve Equation 12, a similar approach is used. The only difference in the process is the presence of the 1, which is subtracted from both sides. The procedure follows:

$$x = 1\text{-}\frac{\left(b\text{-}CDF_{y_2}\right)^2}{(b\text{-}a)(b\text{-}c)} \rightarrow (x\text{-}1)(b\text{-}a)(c\text{-}a) = \text{-}\left(b\text{-}CDF_{y_2}\right)^2$$

Multiplying both sides by -1 and taking the square root results in the inverse CDF:

$$CDF_{y_2}^{-1} = b \pm \sqrt{(1\text{-}x)(b\text{-}a)(c\text{-}a)}$$

Note that there are actually four solutions provided due to the quadratic nature. Only two of these are correct. The selection of the correct two is detailed in Appendix A.

The complete inverse CDF is

$$\begin{cases} CDF_{y_1}^{-1} = \mathrm{a} + \sqrt{x(b\text{-}a)(c\text{-}a)} & 0 < x \leq \dfrac{c\text{-}a}{b\text{-}a} \quad (13) \\ CDF_{y_2}^{-1} = \mathrm{b} - \sqrt{(1\text{-}x)(b\text{-}a)(c\text{-}a)} & \dfrac{c\text{-}a}{b\text{-}a} < x \leq 1 \quad (14) \end{cases}$$

Application of Equations 13 and 14 above transforms the input number to a corresponding number from a triangular distribution, described by *a*, *b*, and *c*. The input number must lie on the interval (0,1). The factor $\frac{c-a}{b-a}$ simply sets the location of the bound at which to change from Equation 13 to Equation 14, set as a ratio of the base of the left half of the triangle to the right half. This ratio returns the location of the value in the uniform distribution that corresponds to the location of *c* in the triangular distribution. Using the default random number generator in R will generate numbers with varying

precision, from the tenths place to several decimals. To enable collection of meaningful data, the precision must be controlled to a level that allows small samples to see potential repetition of numbers; otherwise, frequencies of each value (after binning) will be low enough that the distribution may look more uniform than triangular. Each value from the uniform distribution is mapped to one and only one corresponding value in the target distribution. Thus a small change in the uniform distribution may result in a drastically different value in the transformed distribution. If the desired sample is small and the precision is high, each bin in the target distribution may contain only one or two values. To help prevent this and see more meaningful shapes in small data sets, first a vector of random, uniformly-distributed numbers will be generated in R on the interval [1,*prec*], where *prec* was the precision or number of decimal places desired; i.e., if one desires information to the .001 precision level, then *prec*=1000 would be used. A vector of n numbers between 1 and 1000 would be generated, which would subsequently be normalized by dividing each entry by the value used in *prec*, or 1000 in the scenario described. The result is a vector of numbers on the interval (0,1) as required by the transformation above, with precision to the thousandths' place. Higher precision increases the number of possible random numbers generated, which subsequently means a larger sample size will be required to see the shape of the desired distribution begin to form.

The inverse CDF method was used twice: once for generating numbers on the interval (5,10) for use in the waveform generator, since this interval is required for input, and a second time for transforming the uniform (0,1) distribution to a (0,2) triangular distribution with area equal to 1. The second case is done to generate large sample sizes to check that the fraction of values above y^* is the expected 0.05; since these are on the order of 10^6, they were not used directly with the waveform generator but rather as a check to determine if the method supplies values above y^* 5% of the time.

Sinusoidal Distribution

The next inverse CDF to be derived is that of the sine function, shown in Figure 4.

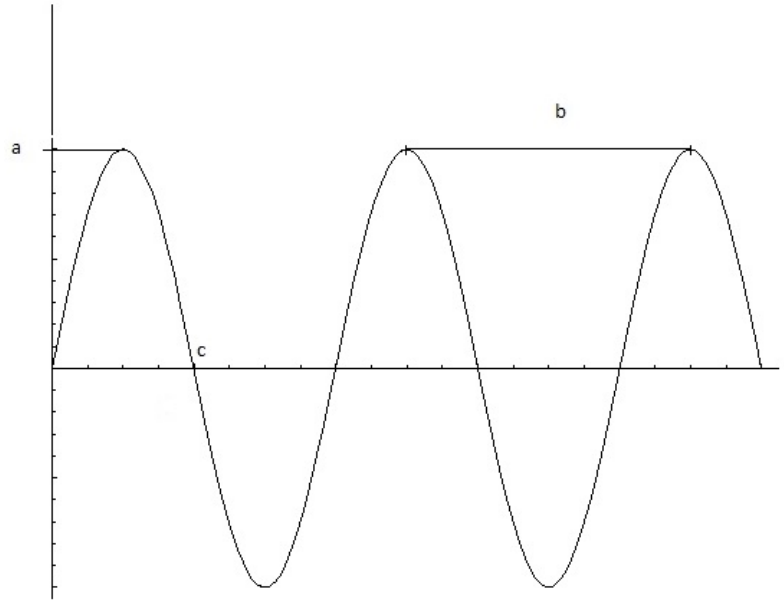

Figure 4: General shape of a sinusoidal distribution.

The same approach as for the triangular distribution is used: begin with the PDF of the sine function, integrate to find the CDF, switch the variables in the CDF, and solve to find the inverse CDF. Beginning with the PDF for a sine function:

$$y=a\cdot\sin(bx) \tag{15}$$

where a is the amplitude of the sine function, $\frac{2\pi}{b}$ is the period, and x is the input of the PDF. The definition of a CDF requires the integral of the PDF to be equal to 1; to ensure an area of 1, Equation 15 is integrated and solved for the upper limit:

$$1=\int_0^c a\cdot\sin(bx)\,dx \tag{16}$$

Solving Equation 16 for c gives

$$1=\frac{a}{b}\cdot(-\cos(bc)+\cos(0))\rightarrow c=\frac{\cos^{-1}\left(1-\frac{b}{a}\right)}{b} \tag{17}$$

Equation 17 describes the upper limit for the integration of Equation 16 to ensure the area under the PDF is always 1. While Equation 17 does not provide information about what values a and b should

take, it should be noted that the area under a full period of a sine function will integrate to zero. Thus if a sinusoidal shape is desired with an area of 1, half the period must be used. Doing so results in use of $c = \frac{2\pi}{2b} = \frac{\pi}{b}$. Using this value for c in Equation 17 results in $b = 2a$. Use of these parameters generates a half sinusoidal function with x-intercepts at $x = 0$ and at $x = c$ and adjusts the height and period to maintain the area requirement. If amplitude (a), interval width (c), or period ($\frac{2\pi}{b}$) are known, the other factors can be adjusted to allow generation of a normalized sinusoidal distribution. Any value for c may be used up to half the period of the sine function, but the relationship between a and b described above will no longer be valid. To find a new valid ratio for a and b, simply limit the argument of the arccosine function in Equation 17 to be $-1 \leq \frac{b}{a} \leq 1$. Then select either a or b to fix either b or a, respectively. Conversely, values for a or b may be chosen and c may be directly computed with Equation 17. Any of these methods should result in an area of 1 under a sine-like curve.

For purposes of this thesis, a half-sine wave is used. A general CDF is derived from Equation 15 by first solving for a in the definite integral form, using c as the general form of the upper limit:

$$y = \int a \cdot \sin(bx) dx$$

$$y = -\frac{a}{b} \cdot \cos(bx) \tag{18}$$

As in the triangular case, Equation 18 is variable-swapped and solved for y to derive the inverse CDF:

$$x = -\frac{a}{b} \cdot \cos(by)$$

$$CDF^{-1} = \frac{\cos^{-1}(-\frac{b}{a}x)}{b} \tag{19}$$

Equation 19 represents the complete CDF for a sine function that crosses the x-axis at any location, defined by the parameters b and a. However, since cosine is an even function and the ratio of b to a is always 2,

$$CDF^{-1} = \frac{\cos^{-1}(-2x)}{b} \tag{20}$$

Equation 20 represents the simplest expression for transforming a uniform variable on (0,1) to a sinusoidal distribution. However, note that the domain of the inverse cosine function is [-1,1]. Therefore using an interval of (0,1) will generate only half of the distribution. The full interval [-1,1] must be considered. The factor of 2 means that the largest absolute value of the interval, 1, will cause an undefined result:

$$CDF^{-1} = \frac{\cos^{-1}(2 \times 1)}{b} = \text{undefined}$$

To get around this, numbers were generated on [-.5,.5]. Following the same procedure described earlier, to control the precision of the random numbers, a uniform distribution is generated on [-*n*,*n*]. Subsequently, all numbers are divided by 2*n*, which scales each number to a value on [-.5,.5]. Multiplying the largest absolute value yields

$$CDF^{-1} = \frac{\cos^{-1}(2\times 0.5)}{b} = 0$$

Thus the resulting distribution is defined across the full domain of a sinusoidal distribution on [0,1].

For use in the waveform generator, the interval must be set between 5 mV and 10 mV. The previous derivation always begins at zero, which suffices for checking repeated number generation. However, a horizontal shift is required and a period scale to force the interval to lie on (5,10). Beginning with Equation 15,

$$y = a\cdot\sin(b(x-c)) \tag{21}$$

Setting Equation 21 equal to 1 and integrating from x=5 to x=10, using the substitution $a=\frac{b}{2}$ results in

$$1=\int_5^{10} \frac{b}{2}\cdot\sin(b(x-5))dx \tag{22}$$

Solving Equation 22 for b yields

$$b=\frac{\cos^{-1}(-1)}{5}=\frac{\pi}{5} \tag{23}$$

Equation 23 results in the requisite value of b to generate a sinusoidal function between 5 and 10 with an area of 1. The full PDF is

$$y = \frac{\pi}{10} \cdot \sin(\frac{\pi}{5}(x\text{-}5)) \tag{24}$$

Integrating Equation 24 yields the CDF:

$$CDF = \frac{\pi}{10} \cdot \int \sin\left(\frac{\pi}{5}(x\text{-}5)\right) dx = \frac{\frac{-\pi}{10}\left(\cos\left(\frac{\pi}{5}(x\text{-}5)\right)\right)}{\frac{\pi}{5}} = \frac{1}{2}\cos(\frac{\pi x}{5}) \tag{25}$$

In solving sine and cosine functions, repeated solutions are usually dropped for simplicity or physical applications. Thus Equation 25 produces a sinusoidal distribution with area of 1, but does not have the correct width and location. Since starting at 5 and ending at 10 is desired, Equation 25 becomes

$$CDF = \frac{1}{2}\cos(\frac{\pi(x\text{-}5)}{5}) \tag{26}$$

Switching the variables in Equation 26 and solving for the inverse results in

$$x = \frac{1}{2}\cos(\frac{\pi(CDF^{-1}\text{-}5)}{5})$$

Where

$$CDF^{-1} = \frac{5\cos^{-1}(2x)}{\pi}\text{-}5 \tag{27}$$

As before, Equation 27 uses an inverse cosine, and so the argument must lie on the interval [-1,1]. Once again the sample is taken from [*-n,n]* and scaled by dividing by 2*n* to ensure the full sinusoidal function is generated between 5 and 10.

Normal Distribution

The next distribution to consider is the normal distribution. While no closed form CDF exists for the normal distribution, good approximations exist. The simplest approach to generating a normal distribution is to use the built-in normal distribution function in R, which requires input of a mean and a standard deviation. The function then uses an approximation to convert a (0,1) uniform distribution to a normal distribution with the specified mean and standard deviation ($X \sim N(\mu,\sigma)$). The resulting distribution can be used for analysis of y^*, discussed later in this section. However, the resulting distribution is not ideal for use in generating normally distributed numbers for use in the waveform generator at large

sample sizes. The reasons are discussed in more detail in the discussion section of this thesis. The general shape of the normal distribution is shown in Figure 5.

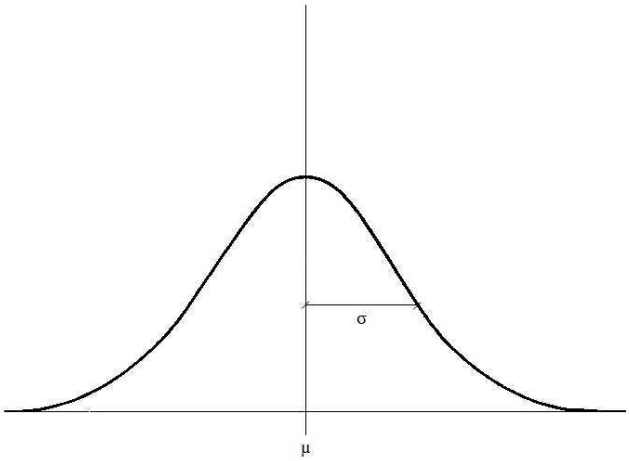

Figure 5: General shape of the normal distribution with mean μ and standard deviation σ.

To apply the same inverse-CDF approach to the normal distribution, an approximation for the CDF of a normal distribution must be used. While several exist and are easily put to use, most of them are not easily invertible, if they are invertible at all. Thus the selection of the approximation is contingent upon finding one such equation that is an acceptable approximation and simultaneously may be inverted. Consider the PDF of the normal distribution:

$$PDF = \frac{1}{\sqrt{2\sigma^2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \tag{28}$$

where σ is the standard deviation of the distribution and μ is the mean. As before, integration of the PDF would yield the CDF, which would then be inverted. The problem, however, is that integration of the PDF leads to a non-closed form solution, as shown:

$$\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt \tag{29}$$

Equation 29 has no closed-form solution; thus an approximation must be used to apply the inverse-CDF method. One such approximation is shown here [10]:

$$\Phi(x)\approx\frac{1}{2}\{1+\mathrm{sgn}(x)\left[1-e^{-\frac{2}{\pi}x^2}\right]^{\frac{1}{2}}\} \tag{30}$$

where sgn(x) is the sign function that returns only the sign of the variable x. Since the uniform distribution will only return positive numbers on (0,1) or (5,10), the sign(x) will not affect the output of Equation 30. The CDF to be inverted then is

$$\Phi(x)\approx\frac{1}{2}\{1+\left[1-e^{-\frac{2}{\pi}x^2}\right]^{\frac{1}{2}}\} \tag{31}$$

Once again, the independent and dependent variables are switched:

$$x\approx\frac{1}{2}\{1+\left[1-e^{-\frac{2}{\pi}(\Phi(x))^2}\right]^{\frac{1}{2}}\} \tag{32}$$

Using algebra to rearrange Equation 32 results in

$$(2x\text{-}1)^2\text{-}1\approx\text{-}e^{-\frac{2}{\pi}(\Phi(x))^2}$$

Here, an oddity occurs because the natural log of a negative number does not exist in the real number plane. Therefore, -1 must be multiplied to both sides of the equation before taking the natural log. The result will be the natural log of a double-negative, or positive number, which has a real solution:

$$\ln(1\text{-}(2x\text{-}1)^2)\approx\text{-}\frac{2}{\pi}\left(\Phi(x)\right)^2$$

$$\Phi^{-1}(x)\approx\pm\sqrt{-\frac{\pi}{2}\ln(1\text{-}(2x\text{-}1)^2)} \tag{33}$$

Equation 33 represents the CDF, but does not account for mean and standard deviation. From the standard normal PDF, the mean and the standard deviation were adjusted for by subtracting the mean from the x-value and dividing by the standard deviation. Thus to generate a normal distribution with a desired mean and standard deviation,

$$\frac{(\Phi^{-1}(x)\text{-}\mu)}{\sigma}\approx\pm\sqrt{-\frac{\pi}{2}\ln(1\text{-}(2x\text{-}1)^2)}$$

$$\Phi^{-1}(x)\approx\mu\pm\sigma\sqrt{-\frac{\pi}{2}\ln(1\text{-}(2x\text{-}1)^2)} \tag{34}$$

Equation 34 represents the full approximate inverse CDF of the normal distribution and may be used to generate any desired normal distribution with an uncertainty of about 0.003 [10]. Once again, to use this distribution requires input of a uniformly-distributed value on (0,1). As described in the triangular distribution, the method uses a uniformly generated number on the interval (1,*n*), which is divided by *n* before being used in Equation 34 to maintain values on (0,1) with the desired precision. Again, the transform was used twice; first to generate voltages between 5 and 10 mV for use with the waveform generator and a second time to confirm the number of repeated occurrences of values exceeding *y**.

Poisson Distribution

The final distribution considered is a discrete distribution, specifically the Poisson distribution, shown in Figure 6.

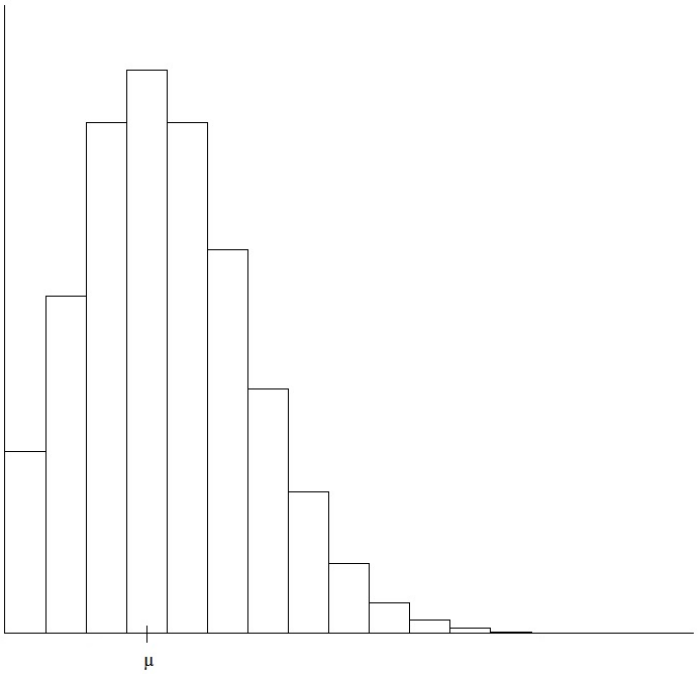

Figure 6: General shape of a Poisson distribution. Note it is defined by a single parameter. For small values of μ, the distribution will be skewed to the right. For large values of μ, the distribution approaches the normal shape.

The Poisson distribution PDF, or rather probability mass function (PMF) in the case of discrete distributions, is an approximation to the binomial distribution. Thus the PMF for the Poisson distribution can be derived from the binomial, as follows:

$$P(X=k)=\binom{n}{k}p^k(1-p)^{n-k} \tag{35}$$

where n is the number of trials, k is the number of successes, and p is the probability of success.

The Poisson distribution is derived as an approximation to the binomial distribution by assigning $np=\lambda$ [13], where λ is the mean of the distribution, then

$$P(X=k)=\frac{n!}{(n-k)!k!}\left(\frac{\lambda}{n}\right)^k\left(1-\frac{\lambda}{n}\right)^{n-k} \tag{36}$$

Using the definition

$$\binom{n}{k}=\frac{n!}{(n-k)!}=n(n-1)(n-2)\cdots(n-k+1) \text{ for } k<n$$

Next, take the limit of Equation 36:

$$P(X=k)=\lim_{n\to\infty}\frac{n!}{(n-k)!k!}\left(\frac{\lambda}{n}\right)^k\left(1-\frac{\lambda}{n}\right)^{n-k}$$

$$P(X=k)=\lim_{n\to\infty}\frac{n(n-1)(n-2)\cdots(n-k+1)}{n^k}\left(\frac{\lambda^k}{k!}\right)\times\lim_{n\to\infty}\left(1-\frac{\lambda}{n}\right)^n\times\lim_{n\to\infty}\left(1-\frac{\lambda}{n}\right)^{-k}$$

$$P(X=k)=\lim_{n\to\infty}\frac{n^k}{n^k}\left(\frac{\lambda^k}{k!}\right)\times\lim_{n\to\infty}\left(1-\frac{\lambda}{n}\right)^n\times\lim_{n\to\infty}\left(1-\frac{\lambda}{n}\right)^{-k}$$

$$P(X=k)=\left(\frac{\lambda^k}{k!}\right)\times e^{-\lambda}\times 1$$

$$P(X=k)=\frac{\lambda^k e^{-\lambda}}{k!}\ \blacksquare \tag{37}$$

Equation 37 represents the full PMF for a Poisson distribution. The Poisson distribution is based on the parameter λ, which is the mean of the function. Herein lies the key difference between the continuous functions and the discrete functions with respect to CDFs. For a discrete function, the inverse CDF does not exist and thus cannot be derived. Instead, an acceptance-rejection criteria script was written R to simulate Poisson events [12]. A random number is generated and compared against a test value of the form $e^{-\lambda}$. An iterative process increases the test value and compares it to the random number. The number of times this comparison occurs until the random number is exceeded is totaled. If the first

generated value is less than $e^{-\lambda}$, a count in the zero bin occurs. This is repeated n times to create a distribution of count rates.

Summations of Equation 37 are taken until the area is approximately 0.95 without exceeding 0.95. A larger sample or mean enables a closer approximation. The CDF to be used is then a modified version of Equation 37:

$$P(X=k)=\sum_{k=0}^{y^*}\frac{\lambda^k e^{-\lambda}}{k!}\leq 0.95 \tag{38}$$

With all the requisite inverse CDFs derived, the next step is to generate distributions of random numbers. First, uniform distributions are used with each of the inverse CDFs above to transform them into triangular (Equations 4 and 5), sinusoidal (Equation 15), normal (Equation 34), and Poisson (Equation 37) distributions. Sample sizes of 10, 20, and 30 are generated for both forms of the triangular distribution, as well as the other three distributions. Histograms are plotted in R in addition to density curves to aid in visualization of the distributions. Each sample is used to create waveforms from each of the distributions, and the counts are monitored through the Lynx. To help recreate the shape of the histogram from R in the Lynx, average values of each bin are used rather than exact values from the number generation, except in the sample size=30 case, where exact numbers are input into the waveform generator. Average values are used since histograms bin data from one end point to another. This means that two values, say 7.4 and 7.6, are called 7.5. However, if the two values are placed in as separate pulses, as are all of the other values from the sample, the resulting spectrum on the Lynx will not mimic the histogram. Using the average value of the bin enables quick comparison between the Lynx output and the histogram.

Determining the Decision Threshold, *y**

After the smaller sample size distributions have been used in the waveform generator, new distributions are generated with large sample size ($n>>100$) to ensure accurate distribution shape. The value corresponding to 0.95 of the area of each distribution is used as the threshold, and the number of

instances of repeated elements is calculated and compared to the expected repetitions. Since each distribution has a different shape, the exact value of y^* varies between distributions. Thus, the next step is to calculate $y^*$for each individual distribution. The y^* may be calculated by setting the integral of each PDF equal to 0.95 and solving for the corresponding upper limit.

First, for the triangular case, two cases must be considered. The first case is the first PDF, Equation 4, which is valid for values of x between a and c. To simplify the calculations, a may be set to zero. In the interest of maintaining generality, however, a will not be set to 0 in this derivation. If y^* occurs at x between a and c, then the triangle is isosceles and skewed right. The second case occurs if the triangle is skewed left, and y^* occurs at x between c and b. Finally, if the triangle is isosceles, the y^* occurs between c and b, so this case is the same as the second case. Thus, the y^* values can be calculated by

$$\begin{cases} 0.95=\int_0^{y^*} \frac{2(x-a)}{(b-a)(c-a)}dx & 0<x\leq\frac{c-a}{b-a} \end{cases} \tag{39}$$

$$\begin{cases} 0.95=\int_0^{y^*} \frac{2(b-x)}{(b-a)(b-c)}dx & \frac{c-a}{b-a}<x\leq b \end{cases} \tag{40}$$

Integrating Equations 39 and 40 results in

$$\begin{cases} y^*=0.1(\sqrt{95}\sqrt{a^2-ab-ac+ab}+10a) & 0<x\leq\frac{c-a}{b-a} \end{cases} \tag{41}$$

$$\begin{cases} y^*=b-\frac{\sqrt{(b-a)(b-c)}}{2\sqrt{5}} & \frac{c-a}{b-a}<x\leq b \end{cases} \tag{42}$$

Equations 39 and 40 are quadratic and have 2 solutions each. The solutions selected for Equations 41 and 42 ensure that the y^* makes sense; i.e., y^* is not negative nor is it above the maximum value of the triangle, b. Note that Equation 41 will only be used in the case that y^* is expected to be below the apex of the triangle.

For the sinusoidal case, only one value of y^* must be calculated. Integrating Equation 15 and setting the left hand side to 0.95,

$$0.95 = \int_0^{y^*} \frac{b}{2} \cdot \sin(bx)\, dx \tag{43}$$

Integrating and solving Equation 43 for y^* results in

$$y^* = \frac{\cos^{-1}(-0.9)}{b} \tag{44}$$

Equation 44 returns the value of y^* for a specified value of b, which also corresponds to the overall shape of the sinusoidal distribution.

Calculating the y^* for the normal distribution uses the approximate form of the CDF, Equation 31. Using the CDF directly will result in the value of y^* that will encompass 0.95 of the area under the normal curve:

$$0.95 = \frac{1}{2}\left\{1 + \left[1 - e^{-\frac{2}{\pi}\left(\frac{y^* - \mu}{\sigma}\right)^2}\right]^{\frac{1}{2}}\right\} \rightarrow y^* = 1.61514 \times \sigma + \mu \tag{45}$$

It is important to note that, since this is an approximation to the normal distribution, the 0.95 area does not occur at the usual 1.645σ, as is the case in the exact normal distribution. Equation 45 allows y^* to be adjusted for specified standard deviation and mean. However, the location of y^* will not correspond exactly to the location for a true normal distribution, because the tails of the approximation are finite. Due to this fact, Equation 45 results in an area approximately equal to 0.95 below y^* for the selected mean and standard deviation. The y^* for the Poisson distribution will be determined iteratively by summing each index of Equation 38 and calculating the cumulative fraction. Once the fraction has reached 0.95 or as close as possible for the particular data set, that value is used as the threshold. Equation 38 is used until the fraction reaches the desired level.

Finally, expected probabilities of various events are calculated. Specifically, the probabilities of seeing 2, 3, 4, and 5 events at y^* or greater in a row are calculated. Since each event is independent of the preceding one, and the total area above y^* corresponds to 0.05, the probabilities are calculated using Equation 1. The calculated probabilities are shown in Table 2.

Table 2: Probabilities of various subsequent, independent events. The probability indicates the likelihood of seeing *n* events occur subsequently in the randomly generated series of numbers from any given distribution.

P(*X*=*n*)	**0.05^n**	**Probability**
P(*X*=2)	0.05^2	0.0025
P(*X*=3)	0.05^3	0.000125
P(*X*=4)	0.05^4	6.25×10^{-6}
P(*X*=5)	0.05^5	3.125×10^{-7}

Additionally, the probability of observing various *n*-choose-*k* probabilities is calculated using the binomial distribution, Equation 35. The expected probabilities are shown in Table 3.

Table 3: Probabilities of various events. The number of trials, n, is the number of subsequent random numbers that are examined. The number of successes, *k*, corresponds to the number of random numbers generated above the *y** value. The probability indicates the likelihood of seeing *k* successes in any *n* subsequent trials.

P(*X*=*n*)	**Number of Trials *n***	**Number of Successes *k***	**Probability**
P(*X*=2)	5	2	0.02143
P(*X*=3)	5	3	0.001128
P(*X*=4)	5	4	2.97×10^{-5}
P(*X*=2)	6	2	0.03054
P(*X*=4)	6	4	8.4611×10^{-5}

The probabilities outlined in Tables 2 and 3 are the fraction of events expected to fall in the 0.05 area above *y**, or the false positive range. The number of subsequent events greater than 0.05 is calculated for each distribution and then compared to the expected probabilities in Table 2. The same is done for events in the *n*-choose-*k* manner; the probability for some number of successes *k* is greater than *y** for a given subset of size *n* from each distribution. The number of times this occurs in each distribution is counted and compared to the number of sequences expected from the binomial expansion (Equation 2). To establish a confidence interval, a sampling distribution was generated for each event type (i.e., a sampling distribution was generated for the number of measurements observed twice in a row, for 2 successes in 5 measurements, etc.). The sampling distribution is defined as the distribution of means of a sample statistic. Use of the sampling distribution enables calculation of the true population standard deviation, which is in turn used as the experimental standard deviation of the mean, to calculate a confidence interval. Thus the confidence interval may be defined as follows:

$$CI=E(x)\ \pm\ z_{\frac{\alpha}{2}}\times\sigma_{\bar{x}} \tag{46}$$

where

$E(X)$ is the expected outcome of the scenario

$z_{\frac{\alpha}{2}}$ is the z-score to obtain a specified confidence interval

$\sigma_{\bar{x}}$ is the standard deviation of the sampling distribution of the mean or

experimental standard deviation of the mean

Equation 46 computes a range of values where the true unknown mean exists. To compute the experimental standard deviation of the mean, a sample size of 50 means of each scenario was generated. From the sample, a standard deviation was computed for each expected number of measurements. A 95% confidence interval was generated around the expected outcome and the mean of the sampling distribution of the means was compared to the confidence interval. If the value compared existed in the confidence interval, the difference between the observed value and the expected value was considered to be not statistically significant. Some of the R codes used are included in Appendix B.

Goodness-of-Fit

Finally, to ensure that the distributions generated fit the shape expected, goodness-of-fit tests were performed. The χ^2 Goodness-of-Fit Test enables comparison in a bin-wise fashion, allowing discretized data to be compared to theoretical distributions. Although the distributions with closed-form CDFs are considered continuous, for small sample sizes the χ^2 is sufficient. However, for the normal distribution, a Shapiro-Wilks test was performed in R. Last, the Poisson distribution also used the χ^2 Test, since by definition the Poisson distribution is discrete. Expected values for the triangular and sinusoidal cases were calculated by integrating the corresponding PDFs in small sections and calculating the expected frequency of numbers in those bins. The χ^2 Test uses the expected frequencies as the null probabilities in R and compares the observed number of measurements in corresponding sections of the generated distributions. The threshold chosen is 0.05; thus any *p*-value greater than 0.05 suggested that the observed data were not significantly different from the theoretical distribution.

Results

Triangular Distribution

First, sample sizes of n=10, n=20, and n=30 were generated for an isosceles triangular distribution. Tables 4, 5, and 6 below show the uniform random variable and the mapped value of the inverse transformation. Equations 10 and 11 were used with a=5, b=10, and c=7.5.

Table 4: Uniform values transformed to corresponding triangular values for sample size n=10. Note that the values have been ordered least to greatest intentionally.

Uniform random number	Triangular random number
0.15	6.369306
0.25	6.767767
0.28	6.870829
0.43	7.318405
0.48	7.449490
0.54	7.602084
0.66	7.938447
0.68	8.000000
0.72	8.129171
0.85	8.630694

Table 5: Uniform values transformed to corresponding triangular values for sample size n=20. Note that the values have been ordered least to greatest intentionally.

Uniform random number	Triangular random number	Uniform random number	Triangular random number
0.01	5.353553	0.58	7.708712
0.05	5.790569	0.62	7.820551
0.06	5.866025	0.63	7.849419
0.17	6.457738	0.67	7.968990
0.26	6.802776	0.70	8.063508
0.26	6.802776	0.74	8.197224
0.39	7.207940	0.82	8.500000
0.42	7.291288	0.86	8.677120
0.43	7.318405	0.97	9.387628
0.57	7.681595	1.00	10.000000

Table 6: Uniform values transformed to corresponding triangular values for sample size n=30. Note that the values have been ordered least to greatest intentionally.

Uniform random number	Triangular random number	Uniform random number	Triangular random number
0.03	5.612372	0.43	7.318405
0.09	6.06066	0.44	7.345208
0.12	6.224745	0.47	7.42384
0.18	6.500000	0.48	7.44949
0.19	6.541104	0.49	7.474874
0.21	6.620185	0.5	7.500000
0.25	6.767767	0.53	7.57616
0.26	6.802776	0.54	7.602084
0.27	6.837117	0.56	7.654792
0.3	6.936492	0.56	7.654792
0.34	7.061553	0.62	7.820551
0.34	7.061553	0.7	8.063508
0.36	7.12132	0.82	8.500000
0.37	7.150581	0.83	8.542262
0.39	7.20794	0.87	8.725245

A visual representation of the three distributions is provided in Figure 7, in addition to a distribution of large sample size to illustrate the definitive triangular shape attained. Note they become more triangular as n approaches infinity.

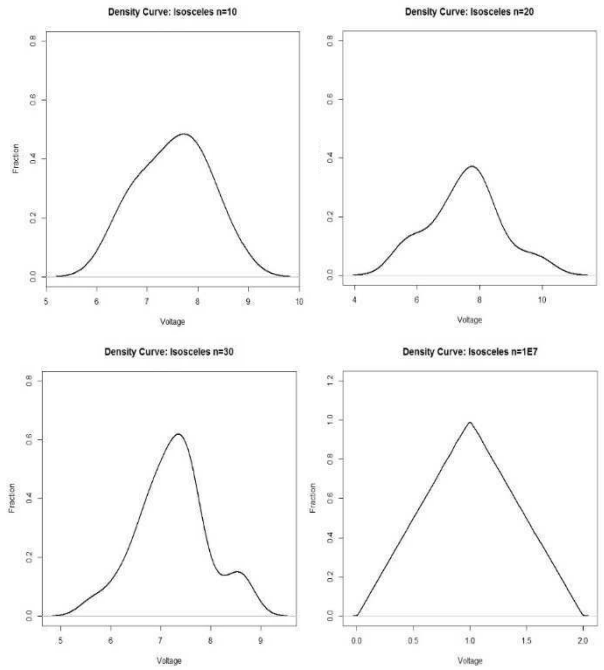

Figure 7: Various density curves of the isosceles triangle.

Next, the expected probabilities greater than y^* are compared with the observed fraction greater than y^* in Table 7 from an isosceles triangular distribution. Using R, 10^7 numbers on (0,1) were generated to a precision of 0.00001. First, the value of y^* was calculated using Equation 42:

$$y^*=b - \frac{\sqrt{(b-a)(b-c)}}{2\sqrt{5}}\approx 1.684$$

The count of 2, 3, 4, and 5 values generated in sequence greater than y^* is calculated. These are shown in Table 7.

Table 7: Probabilities of various subsequent, independent events for an isosceles triangular distribution. The number of times a value greater than or equal to y^* occurred in sequence 2, 3, 4, and 5 times was counted and recorded. Expected probability was calculated by multiplying each expected probability by the sample size 1×10^7. The 95% confidence interval is calculated using Equation 47.

P(X=n)	**0.05^n**	**Expected Sequences, μ_e**	**Observed Sequences, μ_o**	**$\sigma_{\bar{x}}$**	**95% Confidence Interval**
P(X=2)	0.05^2	25000	25018	168	(24670, 25330)
P(X=3)	0.05^3	1250	1253	30	(1190, 1310)
P(X=4)	0.05^4	62.5	63	8	(47, 78)
P(X=5)	0.05^5	3.125	3	2	(0, 7)

Table 8 contains the actual count of sequences greater than y^* in various n-choose-k scenarios.

Table 8: Probabilities of various subsequent, independent events for an isosceles triangular distribution. The number of times a value greater than or equal to y^* occurred in k times in any n-size sequence was observed and recorded. The expected probability was calculated using Equation 2 and multiplying by the same sample size, 10^7. The 95% confidence interval is calculated using Equation 46.

$P\binom{n}{k}$	**Expected Sequences,** μ_e	**Observed Sequences,** μ_o	$\sigma_{\bar{x}}$	**95% Confidence Interval**
$P\binom{5}{2}$	214300	214400	912	(213000,216000)
$P\binom{5}{3}$	11280	11300	172	(10900, 11700)
$P\binom{5}{4}$	297	300	20	(260, 340)
$P\binom{6}{2}$	305400	306000	1200	(303000, 308000)
$P\binom{6}{3}$	21430	21500	257	(20900, 21900)
$P\binom{6}{4}$	846	850	42	(760, 930)
$P\binom{6}{5}$	17.8	18	5	(8, 28)
$P\binom{7}{2}$	406200	406000	1470	(403000, 409000)
$P\binom{7}{3}$	35630	35700	380	(34900, 36400)
$P\binom{7}{4}$	1880	1880	67	(1740, 2000)
$P\binom{7}{5}$	59.2	59	11	(37, 82)
$P\binom{8}{2}$	514600	515000	1720	(511000, 518000)
$P\binom{8}{3}$	54160	54200	515	(53100, 55100)
$P\binom{8}{4}$	3560	3580	100	(3370, 3760)
$P\binom{8}{5}$	150	154	21	(109, 191)
$P\binom{9}{2}$	628500	629000	1960	(625000, 632000)
$P\binom{9}{3}$	77180	77230	655	(75900, 78500)
$P\binom{9}{4}$	6094	6110	135	(5830, 6360)
$P\binom{9}{5}$	321	323	29	(264, 377)
$P\binom{9}{6}$	11.3	11	5	(2, 21)
$P\binom{10}{2}$	746300	746500	2140	(742000, 751000)
$P\binom{10}{3}$	104800	104800	830	(103100, 106400)
$P\binom{10}{4}$	9650	9670	171	(9310, 9980)
$P\binom{10}{5}$	609	613	43	(525, 693)

A visual representation of the isosceles triangular distribution is given in Figure 8, including the location of y^*.

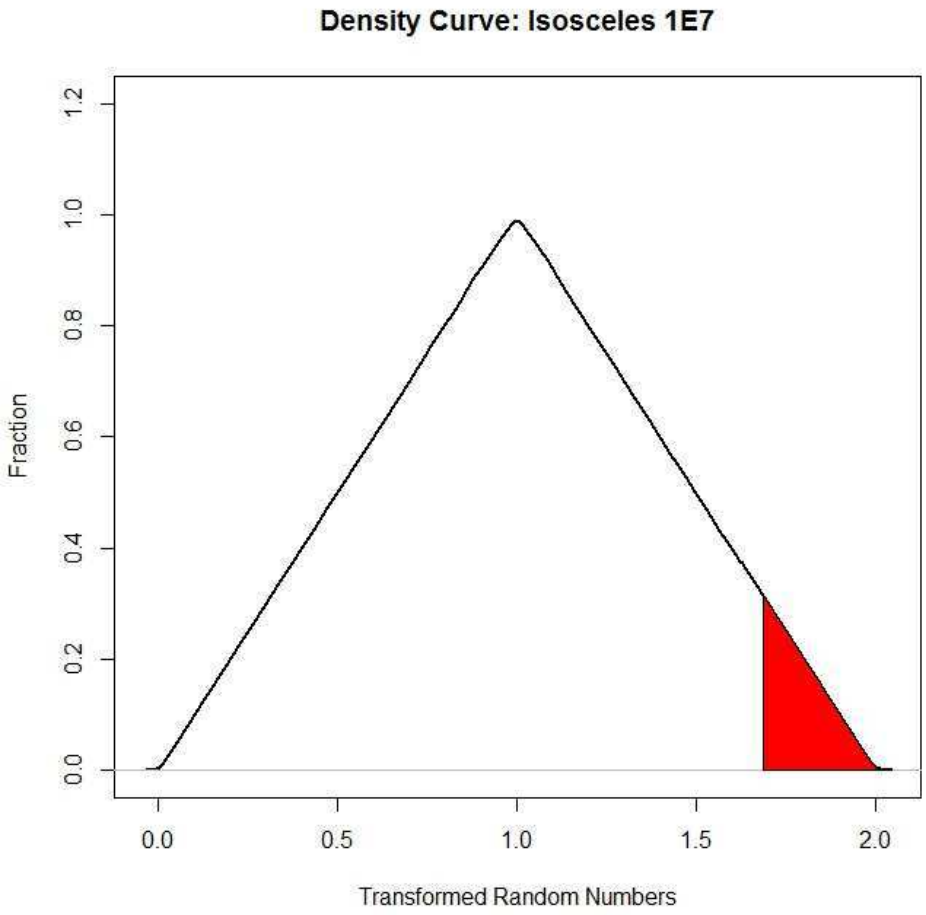

Figure 8: Density curve for 1×10^7 uniform numbers transformed to an isosceles triangular distribution. The shaded area represents the 0.05 area above y^*.

Sinusoidal Distribution

Sample sizes n=10, n=20, n=30, were generated from the sinusoidal transformation. The uniform random variable and the mapped value of the inverse transformation are shown in Tables 9, 10 and 11. Equation 27 is used to transform the uniformly distributed numbers to a sine shape on [5,10].

Table 9: Uniform values transformed to corresponding sinusoidal values for sample size n=10. Note that the values have been ordered least to greatest intentionally.

Uniform random number	Sinusoidal random number	Uniform random number	Sinusoidal random number
0.253	6.655618	-0.088	7.781579
0.232	6.732069	-0.187	8.110070
0.157	6.991654	-0.346	8.716351
-0.022	7.570051	-0.359	8.774711
-0.078	7.749300	-0.433	9.166586

Table 10: Uniform values transformed to corresponding sinusoidal values for sample size n=20. Note that the values have been ordered least to greatest intentionally.

Uniform random number	Sinusoidal random number	Uniform random number	Sinusoidal random number
0.493	5.266629	0.116	7.127365
0.455	5.680407	0.057	7.318168
0.426	5.876948	-0.020	7.563679
0.351	6.261453	-0.047	7.649827
0.336	6.327177	-0.121	7.889017
0.294	6.499577	-0.142	7.958308
0.263	6.618456	-0.178	8.079298
0.243	6.692277	-0.222	8.232205
0.190	6.879620	-0.424	9.110967
0.117	7.124092	-0.451	9.289506

Table 11: Uniform values transformed to corresponding sinusoidal values for sample size n=30. Note that the values have been ordered least to greatest intentionally.

Uniform random number	Sinusoidal random number	Uniform random number	Sinusoidal random number
0.480	5.451672	0.042	7.366152
0.462	5.624499	0.030	7.404450
0.437	5.807588	-0.026	7.582798
0.433	5.833414	-0.050	7.659421
0.423	5.895021	-0.058	7.685036
0.399	6.029457	-0.117	7.875908
0.361	6.216115	-0.146	7.971604
0.246	6.681330	-0.159	8.015056
0.240	6.703183	-0.241	8.300447
0.205	6.827643	-0.274	8.423053
0.191	6.876177	-0.295	8.504361
0.175	6.930908	-0.340	8.690101
0.158	6.988300	-0.420	9.087226
0.146	7.028396	-0.468	9.427509
0.133	7.071489	-0.492	9.714914

A visual representation of the data in Tables 9, 10, and 11 is provided in Figure 9 in addition to a large sample size distribution. As sample size approaches infinity, the density curve appears smoother.

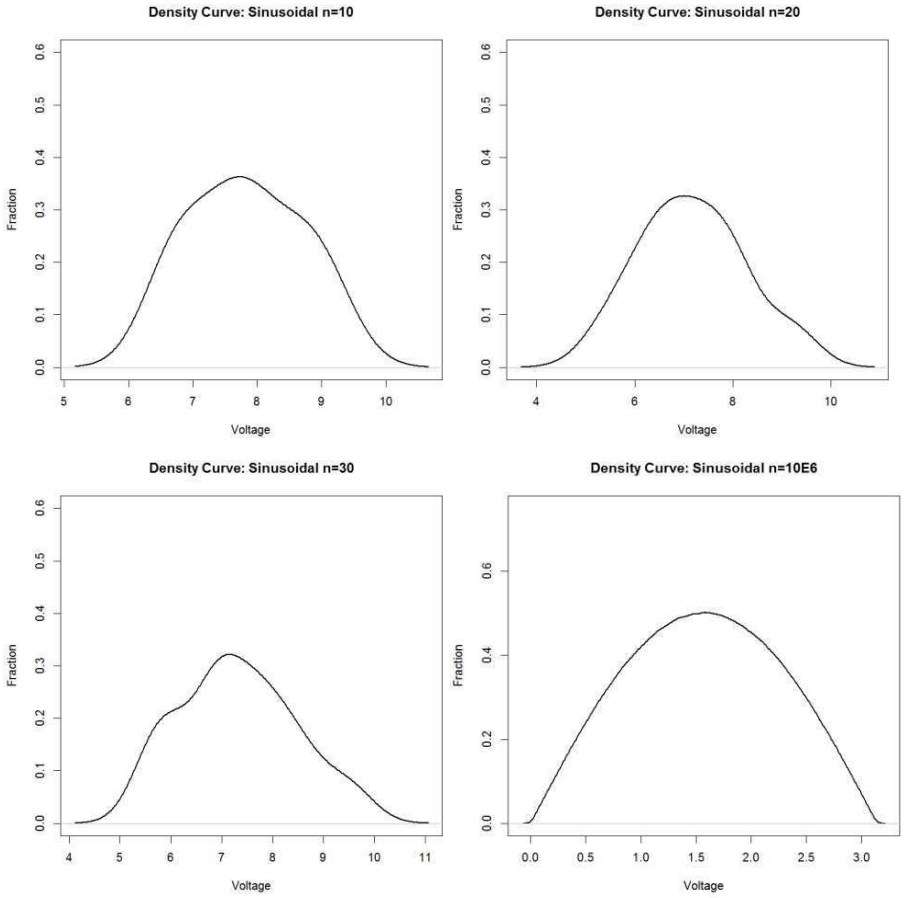

Figure 9: Various density curves of the sinusoidal distribution.

To check for repeated values exceeding the decision threshold, y^* is calculated by Equation 37 with b=1.

$$y^* = \frac{\cos^{-1}(-0.9)}{1} \approx 2.6906$$

Next, the actual probabilities of seeing subsequent groupings of 2, 3, 4, and 5 values greater than y^* were tallied. The actual counts of sequences are reported in Table 12. Once again, 1×10^7 uniform numbers on (0,1) were generated to a precision of 0.00001 and transformed to a sinusoidal distribution. The function used was y=0.5sin(x) on the interval [0,π].

Table 12: Probabilities of various subsequent, independent events for sinusoidal distribution. The number of times a value greater than or equal to y^* occurred in sequence 2, 3, 4, and 5 times was counted and recorded. Expected probability was calculated by multiplying each expected probability by the sample size 1×10^7. The 95% confidence interval is calculated using Equation 47.

P(X=n)	**0.05^n**	**Expected Sequences, μ_e**	**Observed Sequences, μ_o**	**$\sigma_{\bar{x}}$**	**95% Confidence Interval**
P(X=2)	0.05^2	25000	25017	173	(24660, 25340)
P(X=3)	0.05^3	1250	1253	30	(1190, 1310)
P(X=4)	0.05^4	62.5	63	8	(46, 79)
P(X=5)	0.05^5	3.125	3	2	(0, 7)

Table 13 contains the actual count of sequences greater than y^* in various n-choose-k scenarios.

Table 13: Probabilities of various subsequent, independent events for a sinusoidal distribution. The number of times a value greater than or equal to y^* occurred in k times in any n-size sequence was observed and recorded. The expected probability was calculated using Equation 2 and multiplying by the same sample size, 10^7. The 95% confidence interval is calculated using Equation 46.

$\mathbf{P}\binom{n}{k}$	**Expected Sequences, μ_e**	**Observed Sequences, μ_o**	$\sigma_{\bar{x}}$	**95% Confidence Interval**
$P\binom{5}{2}$	214300	214400	934	(213000,216000)
$P\binom{5}{3}$	11280	11300	153	(11000, 11600)
$P\binom{5}{4}$	297	296	23	(253, 341)
$P\binom{6}{2}$	305400	305600	1140	(303000, 308000)
$P\binom{6}{3}$	21430	21500	261	(20900, 21900)
$P\binom{6}{4}$	846	844	40	(768, 924)
$P\binom{6}{5}$	17.8	19	5	(8, 28)
$P\binom{7}{2}$	406200	406400	1380	(403500, 408900)
$P\binom{7}{3}$	35630	35700	371	(34900, 36400)
$P\binom{7}{4}$	1880	1880	59	(1760, 1990)
$P\binom{7}{5}$	59.2	63	12	(36, 83)
$P\binom{8}{2}$	514600	515000	1629	(511000, 518000)
$P\binom{8}{3}$	54160	54250	505	(53200, 55100)
$P\binom{8}{4}$	3560	3550	91	(3380, 3740)
$P\binom{8}{5}$	150	154	21	(105, 195)
$P\binom{9}{2}$	628500	629000	1900	(625000, 632000)
$P\binom{9}{3}$	77180	77300	632	(75900, 78400)
$P\binom{9}{4}$	6094	6090	140	(5820, 6370)
$P\binom{9}{5}$	321	323	36	(250, 392)
$P\binom{9}{6}$	11.3	13	5	(1, 22)
$P\binom{10}{2}$	746300	746500	2180	(742000, 751000)
$P\binom{10}{3}$	104800	104900	765	(103200, 106300)
$P\binom{10}{4}$	9650	9660	200	(9260, 10000)
$P\binom{10}{5}$	609	611	51	(509, 709)

The curve the data in Tables 12 and 13 originate from is shown in Figure 10, including the location of y^*.

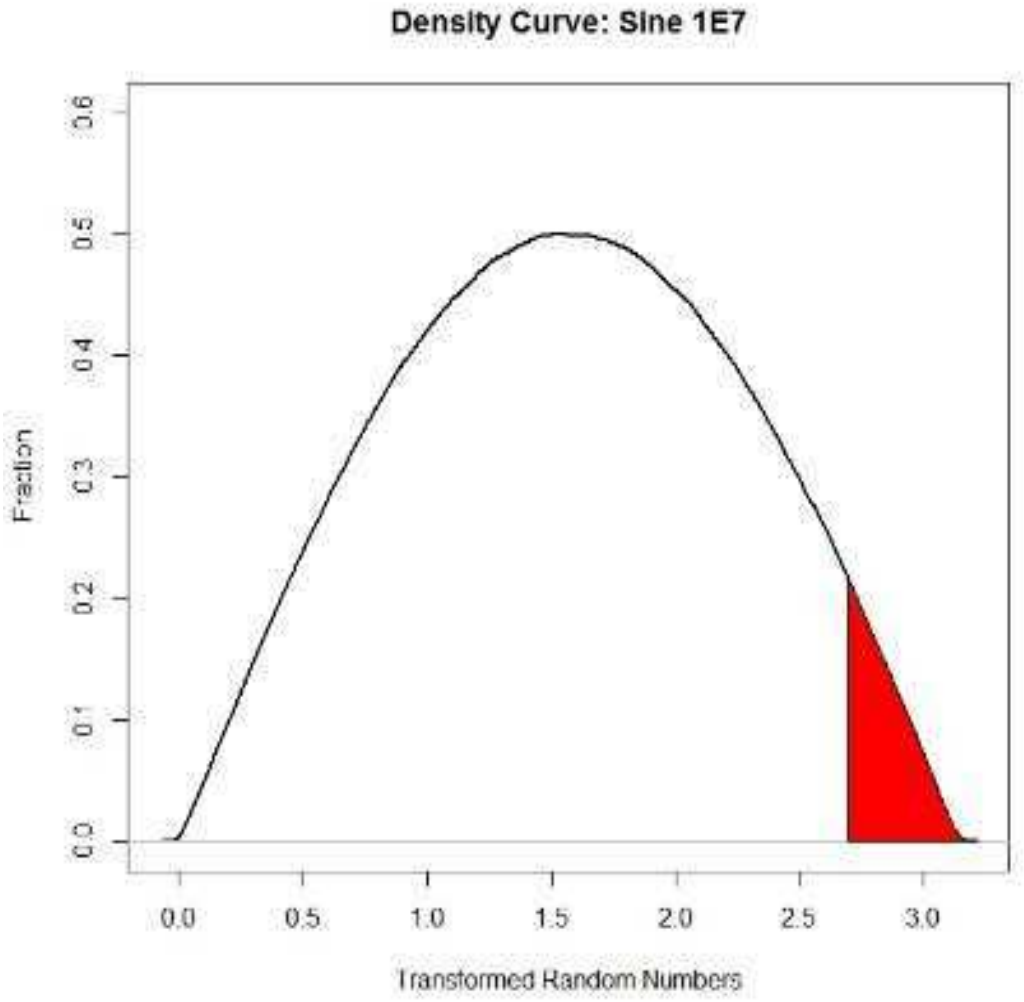

Figure 10: Density curve for 1×10^7 uniform numbers transformed to a sinusoidal distribution. The shaded area represents the 0.05 area above y^*.

Normal Distribution

Sample sizes n=10, n=20, n=30, were generated from the normal transformation. Tables 14, 15, and 16 show the uniform random variable and the mapped value of the inverse transformation. Equation 34 was used to transform the uniformly distributed numbers to a normal shape on [5,10]. Note that the Shapiro-Wilks p-value in Table 14 is roughly half the value of those provided in Tables 15 and 16. This is merely due to statistical variance and small sample size. The 10 randomly generated numbers happen

to be a comparatively poor fit, but another randomly generated 10 numbers may see p-values of similar magnitude.

Table 14: Uniform values transformed to corresponding normal values for sample size n=10. Note that the values have been ordered least to greatest intentionally. The Shapiro-Wilks p-value is also included.

Uniform random number	Normal random number	Shapiro-Wilks p-value
0.0891	8.497015811	0.3364
0.0891	8.497015811	
0.1386	8.308095262	
0.1485	8.276061438	
0.3762	7.736431094	
0.4653	7.565313755	
0.495	7.509400091	
0.495	7.509400091	
0.693	7.122436379	
0.8217	6.812862612	

Table 15: Uniform values transformed to corresponding normal values for sample size n=20. Note that the values have been ordered least to greatest intentionally. The Shapiro-Wilks p-value is also included.

Uniform random number	Normal random number	Shapiro-Wilks p-value
0.0396	8.789883993	0.7157
0.0594	8.650629923	
0.2079	8.107343717	
0.2079	8.107343717	
0.2772	7.942237374	
0.2772	7.942237374	
0.2772	7.942237374	
0.2970	7.898942964	
0.3366	7.815900603	
0.4059	7.678502542	
0.4851	7.528017793	
0.5148	7.472170329	
0.5247	7.453536344	
0.5247	7.453536344	
0.6732	7.163930009	
0.7722	6.942703026	
0.7821	6.918121316	
0.7920	6.892914299	
0.8613	6.692235639	
0.9108	6.50343268	

Table 16: Uniform values transformed to corresponding normal values for sample size n=30. Note that the values have been ordered least to greatest intentionally. The Shapiro-Wilks p-value is also included.

Uniform random number	Normal random number	Shapiro-Wilks p-value
0.0198	9.0027361	0.851
0.0297	8.8816394	
0.0297	8.8816394	
0.0891	8.4970158	
0.0891	8.4970158	
0.1287	8.3416655	
0.1881	8.1599109	
0.1881	8.1599109	
0.2079	8.1073437	
0.2079	8.1073437	
0.2772	7.9422374	
0.2970	7.898943	
0.3069	7.8777759	
0.3366	7.8159006	
0.3564	7.7757942	
0.3861	7.7169876	
0.3960	7.6976828	
0.5346	7.4348749	
0.5445	7.4161749	
0.5445	7.4161749	
0.5445	7.4161749	
0.5841	7.3407587	
0.5940	7.3216906	
0.6831	7.1433146	
0.7029	7.1012724	
0.7524	6.9902074	
0.7920	6.8929143	
0.8514	6.724255	
0.9306	6.4064207	
0.9405	6.3499753	

A visual representation of the data in Tables 14, 15, and 16 is provided in Figure 11 in addition to a large sample size distribution. As sample size approaches infinity, the density curve appears smoother. Note that although the distribution is approximately normal, the tails are not infinite.

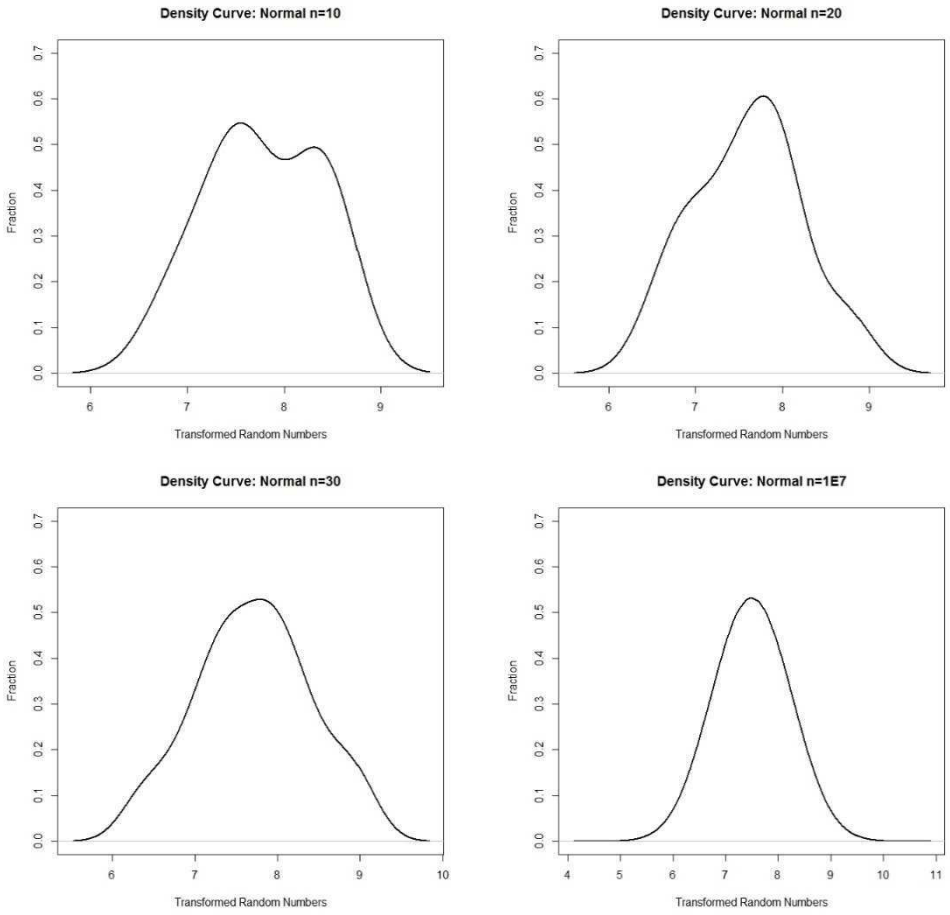

Figure 11: Various density curves of the normal distribution.

To check for repeated values exceeding the decision threshold, first a normal distribution was generated using Equation 45 with μ=7.5 and σ=0.75. Once the distribution was generated, these values for μ and σ were used in Equation 45 to compute y^*. The μ and σ used to generate the distribution are not exactly the same mean and standard deviation that are generated from the transformed data, since Equation 45 is an approximation of the normal CDF, and the sample size is finite. Use of the expected mean and standard deviation yield y^* as follows:

$$y^* = 1.61514 \times 0.75 + 7.5 = 8.711355$$

Next, the actual probabilities of observing subsequent groupings of 2, 3, 4, and 5 values greater than y^* were tallied. The confidence intervals are reported in Table 17. Once again, 1×10^7 uniform numbers on (0,1) were generated to a precision of 0.00001 and transformed to a normal distribution.

Table 17: Probabilities of various subsequent, independent events for a normal distribution. The number of times a value greater than or equal to y^* occurred in sequence 2, 3, 4, and 5 times was counted and recorded. Expected probability was calculated by multiplying each expected probability by the sample size 1×10^7. The 95% confidence interval is calculated using Equation 47.

P(X=n)	**0.05^n**	**Expected Sequences, μ_e**	**Observed Sequences, μ_o**	**$\sigma_{\bar{x}}$**	**95% Confidence Interval**
P(X=2)	0.05^2	25000	25033	169	(24670, 25330)
P(X=3)	0.05^3	1250	1246	36	(1180, 1320)
P(X=4)	0.05^4	62.5	64	10	(43, 82)
P(X=5)	0.05^5	3.125	3	2	(0, 7)

The actual count of sequences greater than y^* in various n-choose-k scenarios are shown in Table 18.

Table 18: Probabilities of various subsequent, independent events for a normal distribution. The number of times a value greater than or equal to y^* occurred in k times in any n-size sequence was observed and recorded. The expected probability was calculated using Equation 2 and multiplying by the same sample size, 1×10^7. The 95% confidence interval is calculated using Equation 46.

$P\binom{n}{k}$	**Expected Sequences, μ_e**	**Observed Sequences, μ_o**	$\sigma_{\bar{x}}$	**95% Confidence Interval**
$P\binom{5}{2}$	214300	214600	805	(213000,216000)
$P\binom{5}{3}$	11280	11300	147	(11000, 11600)
$P\binom{5}{4}$	297	297	25	(248, 346)
$P\binom{6}{2}$	305400	305800	1127	(303000, 308000)
$P\binom{6}{3}$	21430	21400	236	(21000, 21900)
$P\binom{6}{4}$	846	847	49	(750, 942)
$P\binom{6}{5}$	17.8	17	6	(5, 30)
$P\binom{7}{2}$	406200	406500	1500	(403300, 409200)
$P\binom{7}{3}$	35630	35600	331	(35000, 36300)
$P\binom{7}{4}$	1880	1870	79	(1720, 2020)
$P\binom{7}{5}$	59.2	59	12	(35, 84)
$P\binom{8}{2}$	514600	515000	1874	(511000, 519000)
$P\binom{8}{3}$	54160	53280	432	(53300, 55100)
$P\binom{8}{4}$	3560	3570	117	(3330, 3790)
$P\binom{8}{5}$	150	149	19	(112, 188)
$P\binom{9}{2}$	628500	629000	2191	(624000, 633000)
$P\binom{9}{3}$	77180	77200	551	(76100, 78300)
$P\binom{9}{4}$	6094	6100	154	(5790, 6400)
$P\binom{9}{5}$	321	322	32	(259, 383)
$P\binom{9}{6}$	11.3	10	4	(3 20)
$P\binom{10}{2}$	746300	746800	2480	(741000, 751000)
$P\binom{10}{3}$	104800	104800	689	(103400, 106100)
$P\binom{10}{4}$	9650	9660	197	(9260, 10000)
$P\binom{10}{5}$	609	612	51	(509, 709)

The curve the data in Tables 17 and 18 originate from is shown in Figure 12, including the location of y^*.

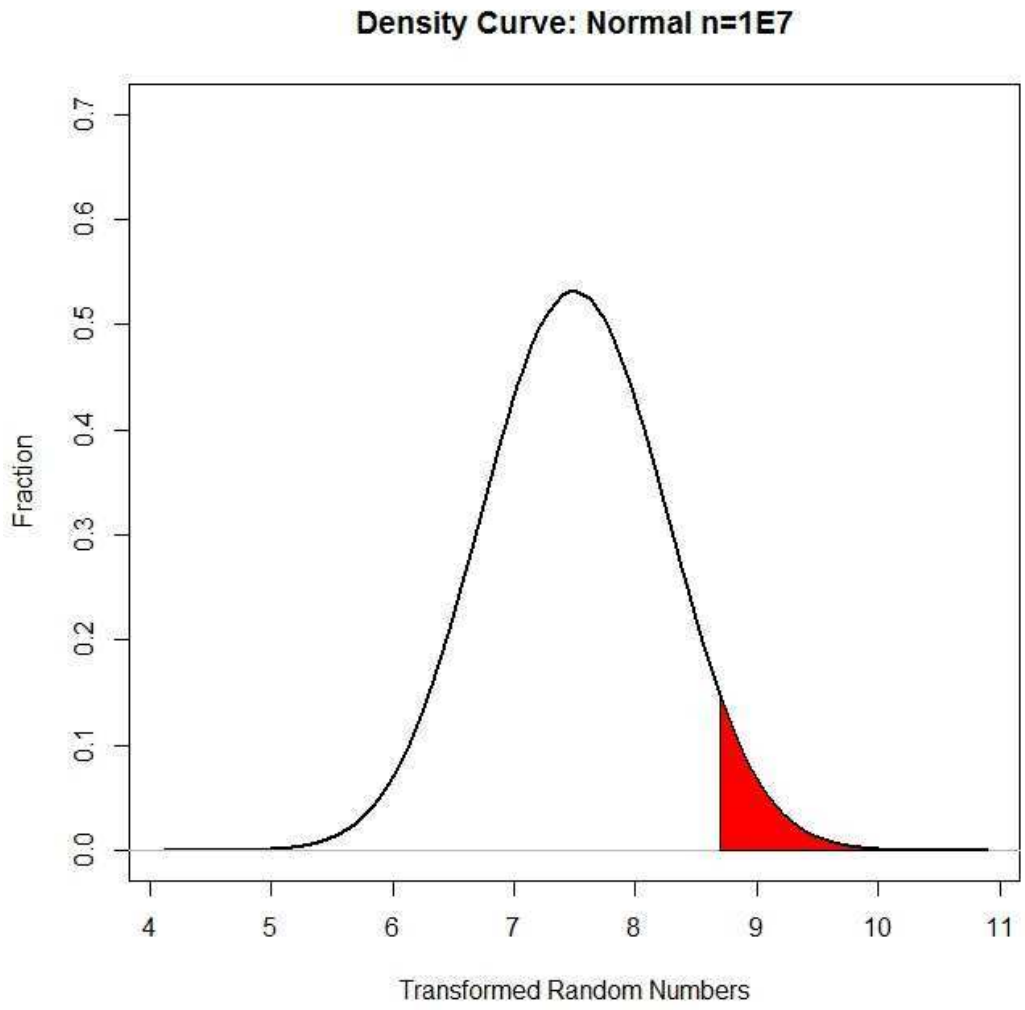

Figure 12: Density curve for 1×10^7 uniform numbers transformed to a normal distribution. The shaded area represents the 0.05 area above y^*.

Poisson Distribution

Sample sizes n=10, n=20, and n=30, were generated from the Poisson equation. Because no CDF exists for the Poisson distribution, no transformation was performed. Instead, the numbers shown in Tables 19, 20, and 21 indicate the number of times the test value was compared to the random number. Once the test value exceeded the random number, the number of iterations required was taken as the generated Poisson random number. Sample sizes are 10, 20, and 30, respectively.

Table 19: Uniform values transformed to corresponding Poisson values for sample size n=10.

Expected Probability, P(X=x) for λ=7.5	**Poisson Random Number, n=10**
P(X=8)=0.1373	8
P(X=5)=0.1094	5
P(X=9)=0.1144	9
P(X=5)=0.1094	5
P(X=3)=0.03889	3
P(X=6)=0.1367	6
P(X=6)=0.1367	6
P(X=6)=0.1367	6
P(X=10)=0.08583	10
P(X=12)=0.03658	12

Table 20: Uniform values transformed to corresponding Poisson values for sample size n=20.

Expected Probability, P(X=x) for λ=7.5	**Poisson Random Number, n=10**
P(X=9)=0.1144	9
P(X=6)=0.1367	6
P(X =8)=0.1373	8
P(X =6)=0.1367	6
P(X=15)=0.005652	15
P(X=7)=0.1465	7
P(X=6)=0.1367	6
P(X=18)=0.0004870	18
P(X=12)=0.03658	12
P(X=7)=0.1465	7
P(X=9)=0.1144	9
P(X=9)=0.1144	9
P(X=11)=0.05852	11
P(X=2)=0.01556	2
P(X=3)=0.03889	3
P(X=7)=0.1465	7
P(X=8)=0.1373	8
P(X=8)=0.1373	8
P(X=5)=0.1094	5
P(X=4)=0.07292	4
P(X=9)=0.1144	15

Table 21: Uniform values transformed to corresponding Poisson values for sample size n=30.

Expected Probability, P(X=x) for λ=7.5	**Poisson Random Number, n=30**
P(X=3)=0.03889	3
P(X=8)=0.1373	8
P(X=12)=0.03658	12
P(X=7)=0.1465	7
P(X=9)=0.1144	9
P(X=12)=0.03658	12
P(X=6)=0.1367	6
P(X=8)=0.1373	8
P(X=9)=0.1144	9
P(X=7)=0.1465	7
P(X=9)=0.1144	9
P(X=4)=0.07292	4
P(X=8)=0.1373	8
P(X=8)=0.1373	8
P(X=7)=0.1465	7
P(X=6)=0.1367	6
P(X=4)=0.07292	4
P(X=7)=0.1465	7
P(X=5)=0.1094	5
P(X=10)=0.08583	10
P(X=8)=0.1373	8
P(X=13)=0.02110	13
P(X=4)=0.07292	4
P(X=5)=0.1094	5
P(X=7)=0.1465	7
P(X=9)=0.1144	9
P(X=12)=0.03658	12
P(X=3)=0.03889	3
P(X=8)=0.1373	8
P(X=10)=0.08583	10

The *p*-values for the χ^2 goodness-of-fit test are included in table 22 for the preceding sample sizes.

Table 22: Uniform values transformed to corresponding Poisson values for sample size n=10, 20 and 30.

Poisson Sample Size n	χ^2 p-value
10	1
20	1
30	1

A visual representation of the data in Tables 19, 20, and 21 is provided in Figure 13 in addition to a large sample size distribution. As sample size approaches infinity, the histograms appear more continuous.

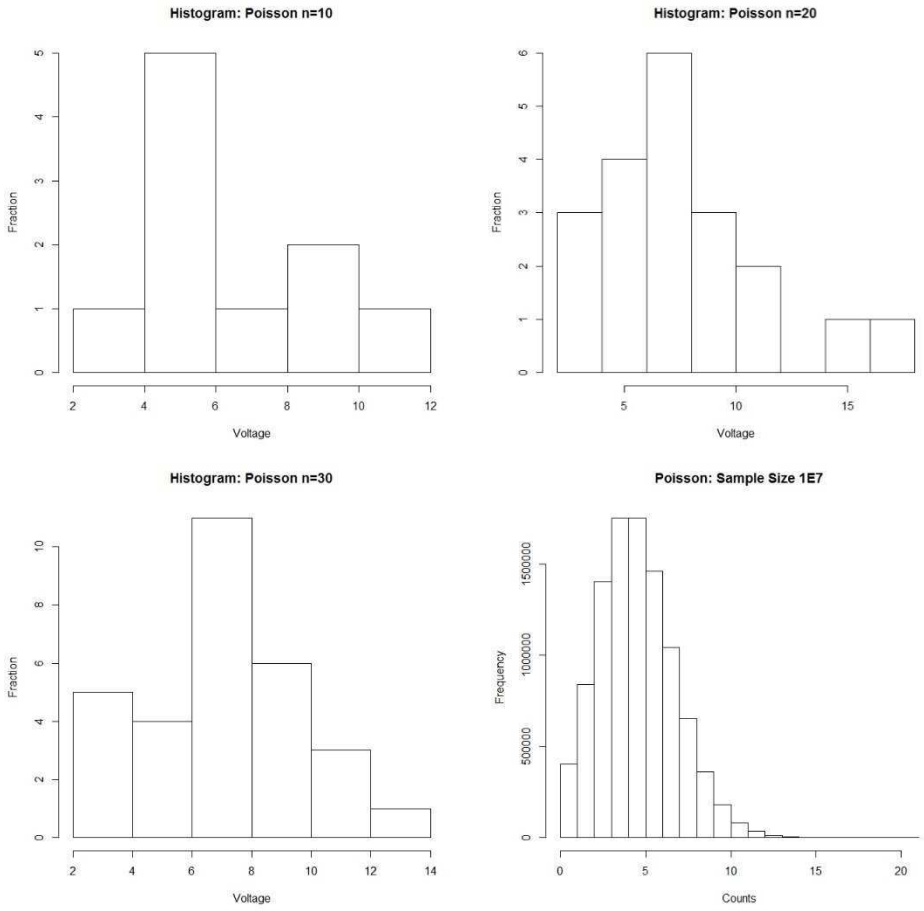

Figure 13: Various density curves of the Poisson distribution.

To check for repeated values exceeding the decision threshold, Equation 38 was used and checked at each iteration to decide whether the 0.95 threshold had been exceeded. The y^* corresponding to the value nearest 0.95 without exceeding was then used. For $\lambda = 5$,

$$P(X=k)=\sum_{k=0}^{y^*}\frac{\lambda^k e^{-\lambda}}{k!}\leq 0.95 \rightarrow y^* = 9$$

To calculate the probabilities of an event occurring greater than the threshold, 0.05 cannot be used in general. Instead, the fraction above the decision threshold must be used, and the tables below reflect this fact. Since the actual y^* was 9, the corresponding fraction greater than y^* was 0.068. Thus 0.068 was used as the basis for expected probabilities in Tables 23 and 24.

Table 23: Probabilities of various subsequent, independent events for Poisson distribution. The number of times a value greater than or equal to y^* occurred in sequence 2, 3, 4, and 5 times was counted and recorded. Expected probability was calculated by multiplying each expected probability by the sample size 1×10^7. The 95% confidence interval is calculated using Equation 47.

P(*X*=*n*)	**0.05^n**	**Expected Sequences, μ_e**	**Observed Sequences, μ_o**	**$\sigma_{\bar{x}}$**	**95% Confidence Interval**
P(*X*=2)	0.068^2	46367	46400	225	(45900, 46800)
P(*X*=3)	0.068^3	3157	3160	56	(3050, 3270)
P(*X*=4)	0.068^4	215	212	16	(184, 246)
P(*X*=5)	0.068^5	15	15	4	(7, 23)

The actual count of sequences greater than y^* in various n-choose-k scenarios are shown in Table 23.

Table 24: Probabilities of various subsequent, independent events for a Poisson distribution. The number of times a value greater than or equal to y^* occurred in k times in any n-size sequence was observed and recorded. The expected probability was calculated using Equation 2 and multiplying by the same sample size, 1×10^7. The 95% confidence interval is calculated using Equation 46.

$P\binom{n}{k}$	**Expected Sequences, μ_e**	**Observed Sequences, μ_o**	$\sigma_{\bar{x}}$	**95% Confidence Interval**
$P\binom{5}{2}$	375258	375300	995	(373300,377200)
$P\binom{5}{3}$	27420	27500	197	(27000, 27800)
$P\binom{5}{4}$	1002	1000	40	(924, 1080)
$P\binom{6}{2}$	524558	524600	1264	(522000, 527000)
$P\binom{6}{3}$	51105	51200	301	(50500, 51700)
$P\binom{6}{4}$	2801	2800	74	(2660, 2940)
$P\binom{6}{5}$	82	82	9	(64, 100)
$P\binom{7}{2}$	684374	684400	1570	(681300, 687400)
$P\binom{7}{3}$	83344	83400	387	(82600, 84100)
$P\binom{7}{4}$	6090	6100	122	(5800, 6300)
$P\binom{7}{5}$	267	268	20	(229, 305)
$P\binom{8}{2}$	850364	850300	1873	(847000, 854000)
$P\binom{8}{3}$	124271	124300	523	(123000, 125000)
$P\binom{8}{4}$	11350	11400	180	(11000, 11800)
$P\binom{8}{5}$	663	665	34	(598, 729)
$P\binom{9}{2}$	1018876	1018800	2178	(1015000, 1023000)
$P\binom{9}{3}$	173713	174000	690	(172000, 175000)
$P\binom{9}{4}$	19040	19000	246	(18600, 19500)
$P\binom{9}{5}$	1391	1400	59	(1280, 1500)
$P\binom{9}{6}$	68	68	12	(45,91)
$P\binom{10}{2}$	1181963	1186700	2505	(1182000, 1192000)
$P\binom{10}{3}$	231263	231000	886	(229500, 233000)
$P\binom{10}{4}$	29572	29600	342	(28900, 30200)
$P\binom{10}{5}$	2593	2600	85	(2400, 2800)

The curve the data in Tables 23 and 24 originate from is shown in Figure 14, including the location of y^*.

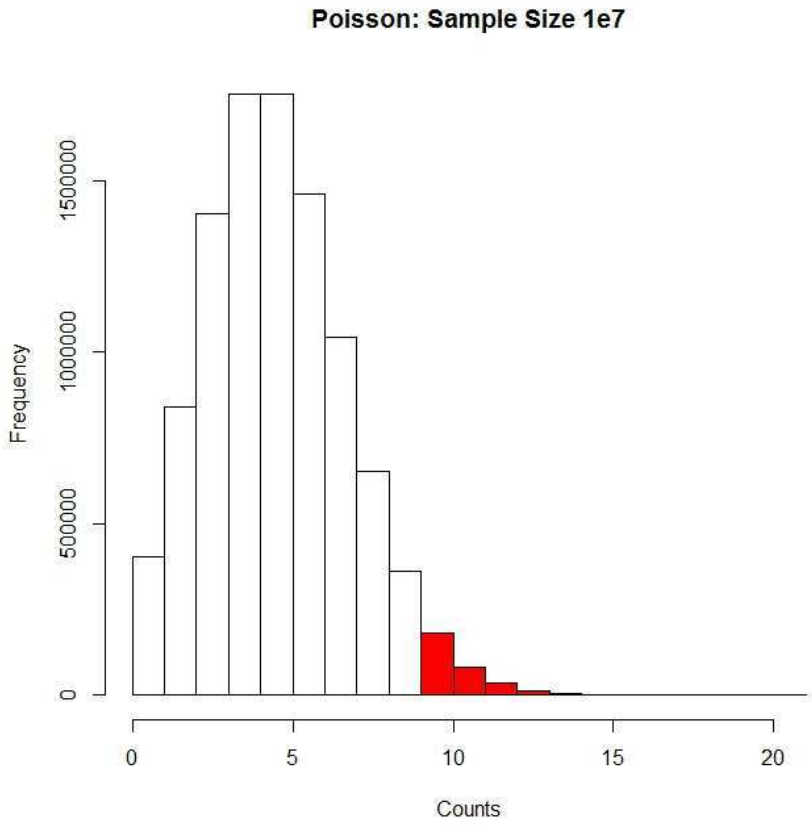

Figure 14: Density curve for 1×10^7 Poisson-generated random numbers. The shaded area represents the ~0.05 area above y^*.

Lynx Output

Input of the random values into the Lynx for n=30 for an isosceles triangle without adjusting the values to be binned in the same manner as R resulted in Figure 15. The gain, polarity, and other options are also shown.

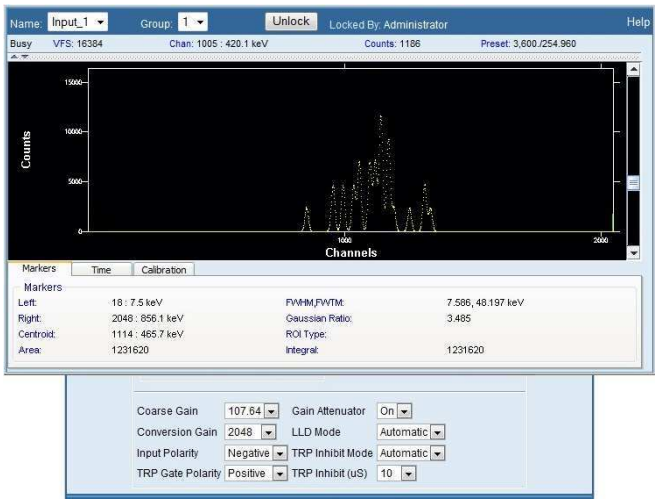

Figure 15: Lynx output for an isosceles triangle with *n*=30random values between 5 mV and 10 mV.

Instead of using the random values generated and displaying them on the Lynx's graphical output, the pulses were also binned to force the output to match the histograms generated by R. The results for an isosceles triangle of *n*=20 output pulses between 5 mV and 10 mV are shown in Figure 16, along with the histogram as plotted in R.

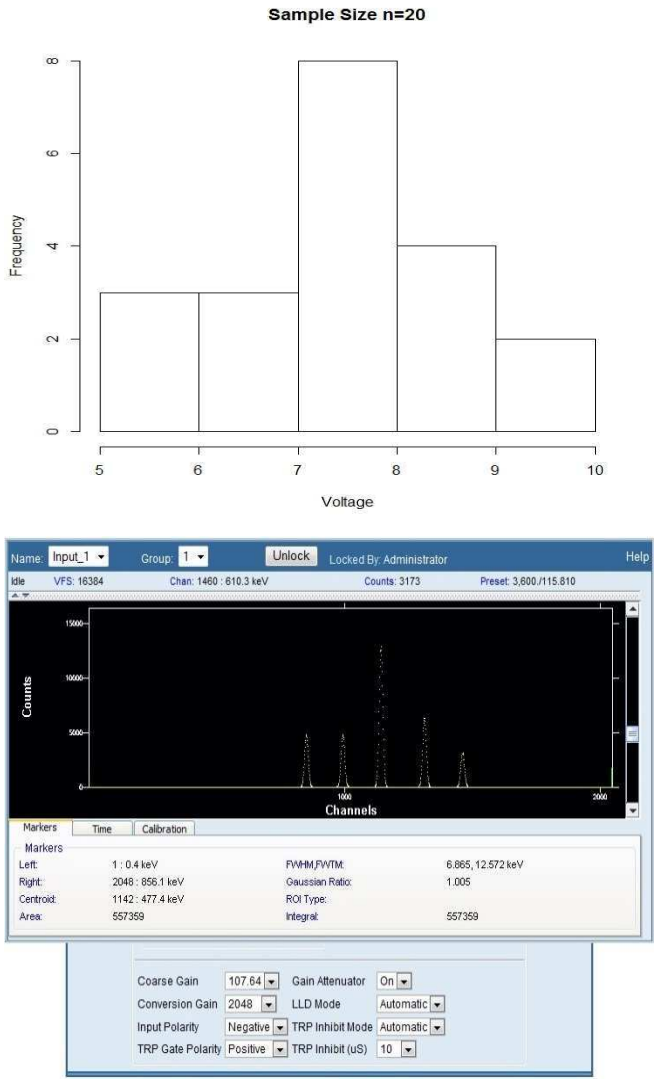

Figure 16: Comparison of Lynx output with R histogram output for *n*=20 for an isosceles triangle with random values between 5 mV and 10 mV.

Discussion

Since radioactive decay is a stochastic process, an atom may or may not decay at any given moment. Thus the count rate over a time interval may differ from another equal time interval count, and a distribution of those count rates may be generated. A frequency histogram would then be plotted with bins of count rates on the abscissa and frequency on the ordinate. Since the waveform generator operates by outputting a specified voltage, the random numbers generated are voltages as opposed to count rates. The concepts remain the same, however, since the analysis is applied to distributions and their associated y^* values. Once a distribution was generated, the number of times repeated values of various permutations occurred was tabulated to determine if classical statistics can predict with sufficient certainty the number of times a random value exceeds a decision threshold, for a particular distribution. Effectively, the idea was that for any distribution with an α of 0.05, a fixed number of doubles, triples, quadruples, quintuples, or n-choose-k scenarios must occur above the corresponding y^*. Confirming this fact enables y^* to be moved to result in different α levels, which ultimately may be used to develop a number of acceptable repeated values under the null hypothesis that no source is present. It is important to note that the repeated values (values exceeding y^* in succession or values exceeding y^* in the n-choose-k manner) are sequences of measurements, more specifically count rates. Each value in the distribution is a count measurement taken over a time interval; thus y^* is itself a decision threshold for count rate. A fixed fraction of measurements (usually 0.05) is expected to exceed y^*.

The decision threshold is specific to the distribution of interest; if the shape of the transformed distribution is not the same or nearly the same as the theoretical distribution, the number of observed events may not fit the predicted events. However, if the distributions are similar, then statistically the location of y^* could be close enough that the number of events observed in the critical region (area greater than y^*) of the derived distribution matches the expected number of events, within some tolerance. Thus y^* alone is not enough to aver that the distribution generated is of a particular form. The p-value for each χ^2 Test for each of the n=10,20, and 30 cases was greater than the 0.05 threshold, and thus resulted in

failure to reject the null hypothesis that the distributions were the same as the corresponding theoretical distributions.

Because the mathematical mapping of the functions is one-to-one, each value on the input side of the inverse CDF corresponds to one and only one output. Therefore, for small sample sizes, if numbers of high precision are generated, the binning may result in a distribution that appears uniform or essentially uniform. This is because a number input at a low precision may generate a considerably different number than a similar number of higher precision but similar in value. To generate more representative distributions for small sample sizes, lower precision should be used. Conversely, to generate numbers that result in well-defined shapes, larger sample size alone is not sufficient; the precision of each number generated must also be increased. Further, each function is also "onto", a mathematical term which means that every possible value in the transformation can be represented, leaving no gaps across the transformed distribution's domain. For a function that is not onto, large samples would result in skewed data or data with portions missing; for small samples, any individual distribution may not be noticeably impacted, but distributions of larger sample sizes would lack important data and result in generated distributions that do not belong to the target distribution.

Large sample size may cause the normal distribution to generate negative numbers, since a hallmark of the normal distribution is its infinite tails. Since the approximation of the normal distribution used in this paper does not have infinite tails, an appropriate standard deviation will help prevent negative numbers if only the shape of the distribution is desired. In practice, while radionuclide count data may be represented as normal, this is because the mean of the dataset can be described as the 0 point, while the z values are left or right of the center of the distribution. However, the actual values of the measurements are never less than zero. In generating a normal distribution, some central value greater than zero must be selected with a corresponding standard deviation that does not result in a significant portion of the data falling below true 0. In generating random numbers, a small mean and/or a large standard deviation may place many values below 0. In addition to the method presented here, another possible means of avoiding this is use of the truncated normal, which enables an essentially normal shape but with definite bounds.

Since the tails of a normal distribution are infinite in theory only, the truncated normal shape may provide a more reliable source of approximation for the normal distribution in count measurement data. However, some bias will be introduced using a bounded distribution to approximate an unbounded distribution. In most cases, the bias will likely be small enough to be ignored, but its presence needs to be stated in those cases wherein the bias may result in a significant deviation from the true value of the parameter being estimated.

To confirm that the observed sequences greater than y^* were in fact statistically the same as those expected, confidence intervals were calculated. Thus for a sample size of 1×10^7, 2.5×10^4 pairs of values greater than y^* are expected. In the case of the triangular distribution, 25018 pairs greater than y^* were observed. Using Equation 46, a confidence interval was generated around the expected value of 25000. The resulting interval included the value 25018, suggesting that there are no statistical grounds on which to dismiss the claim that the generated measurements greater than y^* is statistically identical to the predicted measurements greater than y^*. The same thought process was applied to every scenario to develop 95% confidence intervals. In all examined cases, all sequences generated above y^* were statistically the same as those expected, which confirms the hypothesis that sequences above the decision threshold y^* can be predicted with accuracy. However, it is important to note that for the 95% confidence interval, 5% of the time the interval generated will not contain the true mean. If 5% is unacceptable, a larger sample size can be used if computing power is available. Combined with use of the 99% confidence interval will result in a narrower interval.

Because three of the four distributions, and to some extent the approximation of the normal distribution itself, are not normal, the use of confidence intervals may raise concern. However, because of the Central Limit Theorem (CLT), application of confidence intervals is justifiable [14]. Under the CLT, the distribution of the sample means is normal; thus if repeated distributions were generated and the sequences calculated many times in the same manner as before, plotting the number of counts of each sequence would generate a normal distribution with a mean approximately equal to the true mean of expected counts. Because of this, the Assumption of Normality is valid and confidence intervals are a

reasonable tool to decide whether the number of sequences observed is statistically the same as the number expected. It is important to note that the Assumption of Normality applies to the underlying distribution of the sample statistic of interest; in this case, the sample statistic is the number of observations of a particular form greater than y^*. To ensure accuracy in determining the mean of the sampling distribution and the experimental standard deviation of the mean, a sample size of at least n=30 should be used. That means the distribution generator should be applied at least 30 times to acquire 30 different values for the particular sequence of interest. The standard deviation of the resulting distribution may then be used for the experimental standard deviation of the mean directly. For n<30, the assumption of normality may not hold, and thus the confidence interval may not be reliable.

While the transformation works well for closed-form CDFs, including the normal approximation used in this thesis, it is important to note that the goodness-of-fit tests will likely fail for the normal approximation using large sample sizes. This is because the tails are finite. With small sample sizes, the probability of selecting a value in the tail of a true normal distribution is small, and thus the approximation works well for small sample sizes. However, as the sample size approaches infinity, some random numbers should be sampled from the tails of the distribution. Because the tails are not present in the approximation, the goodness-of-fit will reflect a lack of fit. Finally, as an alternative to the χ^2 Test, the Kolmogorov-Smirnov test may prove useful for testing continuous data when a significantly large sample results in failure of the χ^2 Test, or simply as reinforcement that no reason to reject the null hypothesis exists.

Use of the waveform generator as a source is a preferable method to test hypotheses prior to obtaining a true source. The waveform generator allows signals to be generated in terms of voltage and easily communicates with an MCA such as the Lynx. However, the settings are critical, since a single wrong value can result in a nonsensical signal. Additionally, the correct wave shape must be input to the Lynx to result in a count registering. With the incorrect signal input, or other parameters not properly adjusted, the Lynx may record counts for a time and stop, record signals higher or lower than the intended input, or not recognize any signal is input. Finally, it is time-intensive and potentially equipment-

prohibitive to manually feed large sample sizes into the Lynx. To that end, an automated process using visual basic or similar computer coding may be useful, especially to generate extremely large sample sizes. Due to memory constraints in the waveform generator itself, it may prove useful to feed waveforms as they are generated and collect the data, deleting the old waveforms as new ones are generated and input to the Lynx. Any distribution desired could be derived using the methods in this thesis or other methods as appropriate. The random numbers generated would then be used in the waveform generator to create pulses to mimic a radioactive source or background source shape as needed.

In cases where using a true source is cost-prohibitive or the activity of the source is high enough to render its use cumbersome, the waveform generator may be used in lieu of a true source or even to simulate background distributions. For example, if y^* is known for a particular detector, distributions of random numbers could be fed into the waveform generator, which would then send signals to the detector. Since the expected number of repeated sequences is known or can be calculated, the number of times the detector registers a signal greater than y^* can be compared to the number of times expected, and a base rate of false positive detections can be established for the detector. The waveform generator could then be used to simulate a weak source, and the number of repeated values could be counted and compared to the expected rates under the null hypothesis that no source is present. Ultimately, confirming the expected rates for detectors using simulated data will enable the method to be directly applied to data collected at portal monitors, which will in turn allow heightened sensitivity for them.

Conclusion

In a proof-of-concept effort to confirm that a new statistical approach to source detection would be viable, a number of inverse CDFs were derived. The resulting inverse CDFs were applied to convert uniform, pseudo-randomly generated numbers into distributions of a variety of shapes; specifically, triangular, sinusoidal, normal, and Poisson distributions were generated from a uniform distribution. Distributions from sample size n=10 to n=1×10^7 were generated, with sample sizes n=10, 20, and 30 being directly placed into the Lynx for several distributions. The expected rate of values above y^* is 0.05, except for the Poisson distribution, where y^* was slightly adjusted to compensate for nuances in the methods for that particular distribution. Large sample sizes for each distribution were generated to confirm the expected probabilities for various sequences of values at or above the decision threshold y^*. All investigated scenarios found the 95% confidence interval for the expected sequences greater than y^* to include the observed number of sequences.

The results suggest that probabilities of observing count rate sequences close to but not exceeding y^* may ultimately be computed under the null hypothesis that no source exists. Since current investigation protocols use only y^* as a threshold, if a source is weak or shielded it may be able to slip through a portal monitor undetected. By using suites of detectors intended to measure one type of particle or suites of detectors to measure a variety of particles in conjunction with the method outlined in this thesis, expected rates of sequences may be used to predict presence of a source even under the condition that y^* is never exceeded. Thus additional sensitivity may be afforded, enabling further protection of the environment, humans, and nations. An additional layer of detection sensitivity may reduce false positive measurements while increasing the chances of measuring a weak or shielded source that would otherwise go unnoticed.

References

[1] *Evaluating Testing, Costs, And Benefits Of Advanced Spectroscopic Portals For Screening Cargo At Ports Of Entry*. 1st ed. Washington, D.C.: National Academies Press, 2009. Print.

[2] Ziock, K. P. (2002). The Lost Source, Varying Backgrounds and Why Bigger May Not Be Better. *AIP Conference Proceedings*. doi:10.1063/1.1513955

[3] Pöllänen, R., Siiskonen, T., Ihantola, S., Toivonen, H., Pelikan, A., Inn, K., . . . Bene, B. (2012). Determination of 239Pu/240Pu isotopic ratio by high-resolution alpha-particle spectrometry using the ADAM program. *Applied Radiation and Isotopes, 70*(4), 733-739. doi:10.1016/j.apradiso.2011.12.026

[4] Data produced using the MIRD Program, and extracted from the Evaluated Nuclear Structure Data File (ENSDF), *May 2017*, *[NNDC]*. Additional calculations performed by the program RADLST, T.W. Burrows, *The Program RADLST*, Report BNL-NCS-52142 (1988), National Nuclear Data Center, Brookhaven National Laboratory, U.S.A.

[5] (2004). *Multi-agency radiological laboratory analytical protocols manual: (MARLAP)*. Washington, DC: U.S. Nuclear Regulatory Commission.

[6] Mann, J (2017). A Generalized Decision Threshold for Sequences of Independent Radiological Measurements. Submitted to *Journal of Applied Statistics*.

[7] Portnoy, D., Fisher, B., & Phifer, D. (2015). Data and software tools for gamma radiation spectral threat detection and nuclide identification algorithm development and evaluation. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 784*, 274-280. doi:10.1016/j.nima.2014.11.010

[8] Cember, H. (2009). *Introduction to health physics*. New York: McGraw-Hill, Health Professions Division.

[9] R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/

[10] Bell, J. (2015). A Simple and Pragmatic Approximation to the Normal Cumulative Probability Distribution. *SSRN Electronic Journal*. doi:10.2139/ssrn.2579686

[11] Rubin, J. M. (n.d.). Can a computer generate a truly random number? Retrieved May 02, 2017, from http://engineering.mit.edu/ask/can-computer-generate-truly-random-number

[12] Haugh, M. (2004, September 15). Generating Random Variables and Stochastic Processes. Retrieved May 1, 2017, from http://www.columbia.edu/~mh2078/MCS04/MCS_generate_rv.pdf

[13] Turner, J. E. (2010). *Atoms, radiation, and radiation protection* (4th ed.). Weinheim: Wiley-VCH.

[14] Ott, L., Longnecker, M., & Draper, J. D. (2010). *An introduction to statistical methods and data analysis* (6th ed.). Boston, MA: Cengage Learning.

[15] Archer, D. E., Beauchamp, B. R., Mauger, G. J., Nelson, K. E., Mercer, M. B., Pletcher, D. C., . . . Knapp, D. A. (2006, June 20). Adaptable radiation monitoring system and method. Retrieved May 02, 2017, from http://www.osti.gov/scitech/biblio/908392-adaptable-radiation-monitoring-system-method

[16] Knoll, G. F. (2010). *Radiation detection and measurement*. Hoboken, NJ: John Wiley & Sons.

[17] ISO. (2010) *Radiation protection -- Performance criteria for radiobioassay.* (28218:2010). Geneva, Switzerland: International Organization of Standards.

Appendix A

Selection of Solution for Triangular Distribution Quadratic

Consider the derivation of the triangular distribution, Equation 10 and Equation 11. Each CDF, and consequently each inverse CDF, is a quadratic. The choice of solution depends solely on reason. For instance, if the other solution had been used for 6.1 instead:

$$\mathrm{CDF}_{y_1}^{-1} = \mathrm{a} - \sqrt{x(b-a)(c-a)}$$

Assuming the same values used for the generation of the isosceles triangle on [0,2] gives

$$\mathrm{CDF}_{y_1}^{-1} = 0 - \sqrt{x(2-0)(c-0)} = -\sqrt{2cx}$$

The value of $-\sqrt{2cx}$ will always be negative, and since the triangle defined has only positive values, the solution will not suffice. If the desired triangle transformation were entirely on the negative side of the number line, a different derivation or approach would be required, since Equation 14 will generate a negative under the radical. The simplest approach to generate a triangular distribution of negative values would be to generate one first of positive values and subtract some number from every element. However, the equation derived for *y** would no longer be useful.

Proof of Sine Derivation

To support the assertion that the sine derivation will always integrate to 1, provided the parameters are related by $b = 2a$ and $c = \frac{\pi}{\mathrm{b}}$, consider the following:
Begin with the integral of sine, equation 16.

$$1 = \int_0^{\mathrm{c}} a \cdot \sin(bx)\, dx$$

Setting the parameters as described above gives

$$1 = \int_0^{\frac{\pi}{2\mathrm{a}}} a \cdot \sin(2ax)\, dx$$

Completion of the integral on the right hand side yields

$$1 = -\frac{a \cdot \left(\cos\left(\frac{\pi}{2a} \cdot 2a\right) - \cos(0)\right)}{2a}$$

$$1 = -\frac{(-1-1)}{2}$$

$$1 = -\frac{(-2)}{2}$$

$$1 = 1\blacksquare$$

Thus for any sine function integrated with the specified parameters, the result will always be an area of 1. Using this fact enables a convenient means to derive a sine wave of any width, crossing the *x*-axis at any location, or stretching to any height. As long as the other parameters are adjusted to compensate, the area will remain 1 and a distribution can be generated.

Appendix B

Two of the codes used in R to generate the distributions are provided.

Triangular Distribution R-Code

```
#Generate uniform random numbers
options(digits=16)
counts=100000
prec=100000
#note that higher precision eliminates or reduces the small triangle
#at the right tail. sample must be run from 1 to desired precision
#(prec) or the density plot will fail.
x=sample(1:prec, counts, replace=T)
x=x/prec
qts=quantile(x,probs=c(0.95))
q100=quantile(x,probs=c(1))
count=0
r=rep(1,length(x))

#Set parameters for triangle
a=0
b=2

#Set c for scalene or isosceles
c=1
#c=1
check=(c-a)/((b-a))
ystar=b-sqrt((b-a)*(b-c))/(2*sqrt(5))

#scaling factor for height of density curve
height=1.2*(2*(c-a)/((b-a)*(c-a)))
y=c(x)

#Select location relative to apex of triangle.
for(i in 1:length(y))
{
        if(x[i] < check) {
                y[i]=a+sqrt(x[i]*(b-a)*(c-a))
        } else {
                y[i]=b-sqrt((1-x[i])*(b-a)*(b-c))
        }
}

#clt.mean[int]=mean(y)
#}

#hist(y)
```

```
d=density(y)
plot(d,main="Density Curve: Isosceles n=1E7",xlab="Voltage",ylab="Transformed Random
Numbers",ylim=c(0,height),lwd=2)

for(j in 1:length(y))
{
	if(y[j]>=ystar)
	{
		r[j]=1
	}
	else
	{
		r[j]=0
	}
}

for(j in 1:(length(y)))
{
	if(r[j] + r[j+1] + r[j+2] + r[j+3] + r[j+4] + r[j+5] == 3) {
		count=count+1
	}
	else {
		count=count
	}
}

his=hist(y,breaks=c(0,.25,.5,.75,1,1.25,1.5,1.75,2))
null.probs=c(0.03125,0.09375,0.15625,0.21875,0.21875,0.15625,0.09375,0.03125)
chisq.test(his$counts,p=null.probs)

f=table(cut(y,breaks=c(-Inf,0.25,0.5,0.75,1.0,1.25,1.5,1.75,Inf),labels=c('<0.25','0.25 to 0.5','0.5 to 0.75','0.75 to
1.0','1.0 to 1.25','1.25 to 1.5','1.5 to 1.75','1.75 to 2.0')))
table(f)
freq=as.numeric(as.matrix(f))
real.freq=freq/sum(freq)

null.freq=counts*null.probs
chisq.test(freq,p=null.probs)
ks.test(freq,null.freq)

qts=quantile(y,probs=c(0.95))
q100=quantile(y,probs=c(1))
qystar=quantile(y,probs=c(.689202437605))

his=hist(y, plot=FALSE)

k=0
j=0
for(i in 1:length(his$breaks))
```

```
{
	if(his$breaks[i]>=c) {
		k=k+1
	}
	else {
		j=j+1
	}
}

i=1
py1=c(rep(0,length(his$counts)-k))
py2=c(rep(0,length(his$counts)-j))
while(his$breaks[i] < c)
{
	avgheight=((his$breaks[i]-a)+(his$breaks[i+1]-a))/((b-a)*(c-a))
	py1[i]=(his$breaks[i+1]-his$breaks[i])*avgheight
	i=i+1
}

k=1
while(his$breaks[i] < b)
{
	avgheight=((b-his$breaks[i])+(b-his$breaks[i+1]))/((b-a)*(b-c))
	py2[k]=(his$breaks[i+1]-his$breaks[i])*avgheight
	k=k+1
	i=i+1

}

E=c(py1,py2)

O=c(rep(0,length(his$counts)))
for(i in 1:length(his$counts))
{
	O[i]=his$counts[i]/sum(his$counts)
}

OminE=c(rep(0,length(E)))

for(i in 1:length(E))
{
	OminE[i]=(O[i]-E[i])^2
}



dense=density(y)
plot(dense,main="Density Curve: Isosceles 1E7",xlab="Transformed Random Numbers",ylab="Transformed
Random Numbers",ylim=c(0,height),lwd=2)
```

```
#change x1 or x2; this will shade between x1 and x2
x1=min(which(dense$x>=ystar))
x2=1000
#x2=max(which(dense$x<=q100))
with(dense,polygon(x=c(x[c(x1,x1:x2,x2)]),y=c(0,y[x1:x2],0),col="red"))
```

Poisson R-Code

```
#Poisson by Accept-Reject Criteria

#Set parameters and sample size
lambda=5
n=1e5
#Pre-allocate size of vector for x
x=c(rep(0,n))
#Reset counts (in case re-running after previous run)
count=0
#Pre-allocate vector for measuring hits above y*
r=rep(0,length(x))
#Generate Poisson Random Numbers
for(i in 1:n)
{
        U=runif(1)
        j=0
        p=exp(-lambda)
        F=p
        while (U>F)
        {
                p=lambda*p/(j+1)
                F=F+p
                j=j+1
        }
        x[i]=j
        i=i+1
}
#Find y*
target=0
iter=0
while(target <= 0.95) {
        ptarg=target
        piter=iter
        target = lambda^iter*exp(-lambda)/factorial(iter) + target
        iter=iter+1
        print(iter)
}
ystar=piter

#To be used to color histogram above y*.
header=ystar
#Calculate null probabilities for chi-squared test
null.probs=lambda^(0)*exp(-lambda)/(factorial(0))
```

```
i=1
while(null.probs<1)
{
	null.probs=null.probs+lambda^(i)*exp(-lambda)/(factorial(i))
	if(null.probs=1)
	{
		numelements=null.probs
		break
	}
	i=i+1
}
null.probs=rep(0,i)
null.probs[1]=lambda^(0)*exp(-lambda)/(factorial(0))
i=1

while(sum((null.probs))<1)
{
	null.probs[i+1]=lambda^(i)*exp(-lambda)/(factorial(i))
	print(i)
	print(null.probs[i])
	i=i+1
}
#Pre-allocate vector for observed frequencies
act.probs=rep(0,length(null.probs))
setbreaks=seq(from = 0, to = i, by = 1)
#Color area to right of 0.95
his <- hist(x,breaks=setbreaks,plot=FALSE)
plot(his, main="Poisson: Sample Size 1e7",xlab="Counts",ylab="Frequency",col=ifelse(abs(his$breaks)
< header, "white", ifelse (abs(his$breaks) >=header, "red", "gray50")))

#Populate vector for observed frequencies
for(i in 1:length(act.probs))
{
	if(his$breaks[i]!= setbreaks[i])
	{
		act.probs[i]=0
	}
	else
	{
		act.probs[i]=his$counts[i]
	}
}

#Convert vector to fractional values
act.probs=act.probs/n
#Test Goodness-of-Fit
chisq.test(act.probs,p=null.probs)
#Search for values exceeding y*
for(j in 1:length(x))
{
	if(x[j]>=ystar)
```

```
	{
		r[j]=1
	}
	else
	{
		r[j]=0
	}
}
#Check for sequences; change number of r[j] values to correspond to n, change sum to correspond to k
#(for n-choose-k)
for(j in 1:length(x))
{
	if(r[j] + r[j+1] + r[j+2] + r[j+3] + r[j+4] + r[j+5] + r[j+6] + r[j+7] + r[j+8] + r[j+9] == 2)
	{
		count=count+1
	}
	else
	{
		count=count
	}
}
#Return counts
count
```