

# Computational Physics 301:

## Exercise 3 - Random Numbers and Monte Carlo

Caspar Fuller - 1310006

17th April 2016

The simulation of complex physical systems using a computer often use Monte Carlo methods of simulation. A random number generator is an essential condition to run the random sampling the Monte Carlo methods rely on and these random numbers can be generated by a computer. Although it is not feasible to generate 'true' random numbers on a computer, one can generate a deterministic set of pseudo-random numbers that have properties close to true random numbers. This report aims to make use of these random number generators to produce a non-uniform distribution of random numbers as well as simulating physical systems using Monte Carlo methods. Finally I shall investigate statistical solutions for finding experimental errors using these methods. Each program was written in the C language.

### Problem 1: Random Numbers in a Distribution

#### Theory and Computation

The GSL (GNU Scientific Library)<sup>[1]</sup> random number generators produce random floating point numbers between 0 and 1 with a uniform probability distribution so one number is just as probable to appear as any other. The first task was to use these generators to generate random number distributions from non-uniform probability distributions, specifically a set of random angles in the range  $0 < \theta < \pi$  with a distribution proportional to  $\sin\theta$ .

$$p(y) = p(x) \left| \frac{dx}{dy} \right| \quad (1)$$

The first sampling method used to perform this was the inverse transformation method. To use this method the cumulative distribution function (CDF) of the distribution must be found. As the task uses a continuous distribution the probability density function must be integrated to find the CDF. This method relies on the fundamental transformation law of probabilities shown in equation (1). Where  $p(x)dx$  represents a uniform probability distribution between 0 and 1 and  $p(y)dy$  represents the probability distribution of  $y(x)$ , the desired final probability distribution. For a desired distribution of  $p(y) = f(y)$ , where  $f$  is positive and has an integral of 1, the solution of the

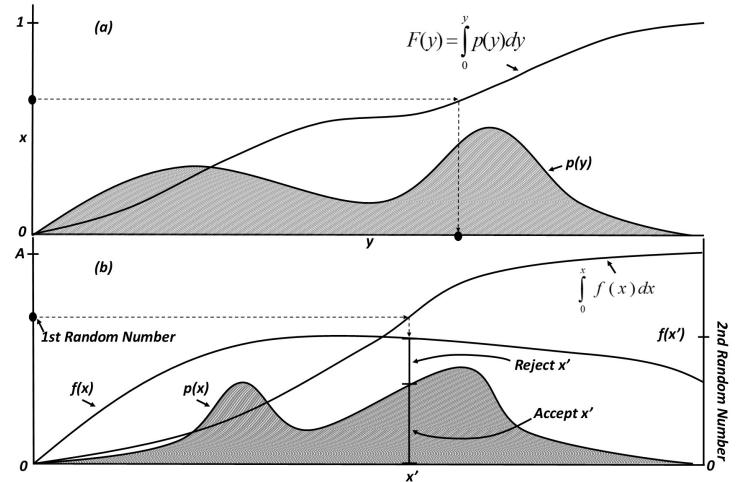


Figure 1: (a) - Inverse Transformation method for generating a random number,  $y$ , from a desired probability distribution  $p(y)$ .  $p(y)$  must be integrable and the integral must be analytically invertible.

(b) - Reject-accept method for generating a random number,  $x$ , from a desired probability distribution  $p(x)$  that lies below the comparison function,  $f(x)$ , everywhere.<sup>[7]</sup>

differential equation becomes  $x = F(y)$  where  $F(y)$  is the indefinite integral of  $f(y)$ . The value for  $y$  is desired so simple re-arrangement can be carried out to give equation (2), the left hand side of which is just the inverse of the CDF.

$$y(x) = F^{-1}(x) \quad (2)$$

This can be seen geometrically in figure (1(a)), a uniform random number between 1 and 0 is sampled. This represents a proportion of the probability area under the curve  $p(y)$ .  $F(y)$  corresponds to the area under the probability curve to the left of  $y$  so is used to find the domain value that as the exact proportion of probability area to its left. Often the integral of the probability density function let alone its inverse cannot be found analytically or numerically. Another sampling method is then required. I shall use the reject-accept sampling method, a general method similar to the standard Monte Carlo method of integration<sup>[2]</sup>.

The reject-accept sampling can be explained geometrically as seen in figure (1(b)). The desired probability distribution  $p(x)$  is drawn as before so any range of  $x$  corresponds to the probability of generating an  $x$  in that range. A comparison function,  $f(x)$ , is then drawn to lie above  $p(x)$  for all  $x$ . A similar method to the transformation method is used meaning that the indefinite integral of  $f(x)$  must be known and must be analytically invertible. Take a random number in the range of 0 to  $A$  where  $A$  represents the total area under  $f(x)$  and use this to find the corresponding value of  $x$  as before, on figure (1(b)) this is  $x'$ . Another ran-

dom number,  $y$ , must now be generated in the range  $0 < y < f(x')$ . We now have a 2D point uniform in the area under the comparison function. If  $y < p(x')$  then accept this point and if not reject the point. The distribution of  $p(x)$  is then built up from the accepted values of  $x'$ .

For the task set the desired probability distribution of angles was required to be proportional to  $\sin\theta$ . For the transformation method this meant:

$$x = F(y) = \int_0^y p(y)dy = \frac{1}{2} \int_0^\theta \sin(\theta)d\theta \quad (3)$$

where  $x$  is the uniform random number generated between  $0 < x < \pi$  and  $\theta$  is the desired random number to fit the distribution. The factor of  $\frac{1}{2}$  is used to restrict the range of the distribution to the required range,  $0 < \sin\theta < 1$ . Equation (4) shows results of the integration and rearrangement to give the final equation for finding sinusoidally distributed random numbers.

$$\theta = \arccos(1 - 2x) \quad (4)$$

For the accept-reject method the comparison function was set to uniformly generate angles in the range  $0 < \theta < \pi$  corresponding to the range of the probability distribution  $0 < p(\theta) = \sin(\theta) < 1$ . A second random number,  $y$ , was then generated between  $0 < y < 1$  as the maximum value for  $p(\theta)$  was found at  $\sin(\frac{\pi}{2}) = 1$ . The accept-reject method was then used to get the required probability distribution.

## Results and Discussion

The resulting distribution from the analytical method, for  $n = 1,000,000$  where  $n$  is the number of simulations, can be seen in figure (2). A sinusoidal fit was applied to the data and a goodness of fit test was performed. The adjusted R-squared value was used as this test and was found to be  $R^2 = 0.99953$  for figure (2) showing a near perfect sinusoidal distribution. Figure (3) shows the resulting distribution for the reject-accept method for 1,000,000 randomly generated angles. An  $R^2$  value of 0.99934 was found which gives a difference of 0.02% between the R-squared values. As no preferred method can be inferred from this further investigation was done to investigate the accuracy of the methods for different  $n$  values. Figure (4) shows the plot of adjusted  $R^2$  values for varying  $n$ .  $N$  was plotted on a logarithmic scale to pronounce the difference in the two methods. As seen before at high orders of  $n$  there is little to no difference between the techniques. However when  $n$  falls below 10,000 the analytical method becomes the preferential technique to use. This difference occurs from the rejection of many data points from the

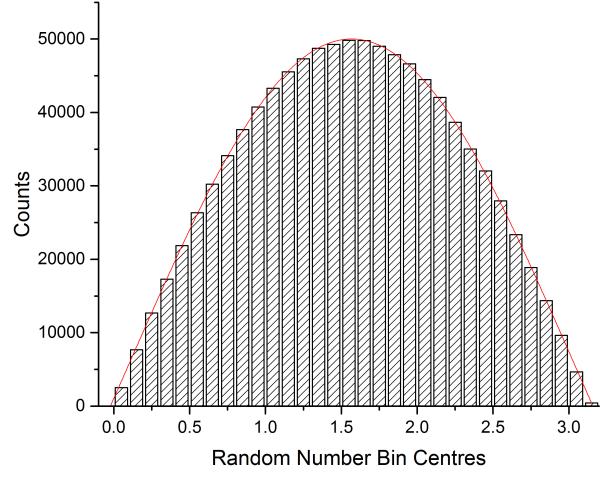


Figure 2: Histogram showing the sinusoidal distribution of random angles in the range  $0 < \theta < \pi$ . 1,000,000 random angles were generated with a seed of 0 using the analytical transformation technique.  $R^2 = 0.99953$

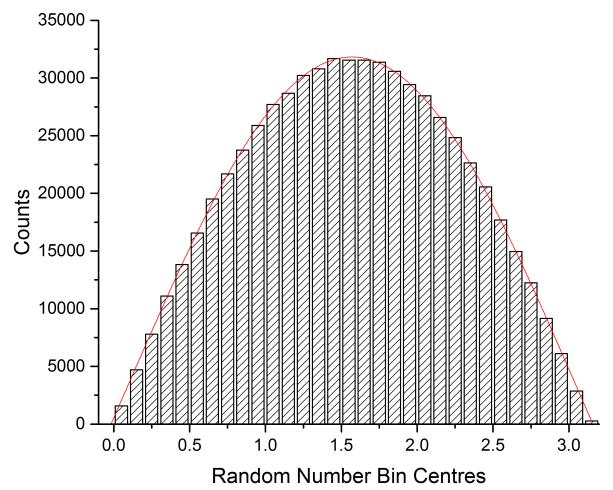


Figure 3: Histogram showing the sinusoidal distribution of random angles in the range  $0 < \theta < \pi$ . 1,000,000 random angles were generated with a seed of 0 using the reject-accept method.  $R^2 = 0.99934$

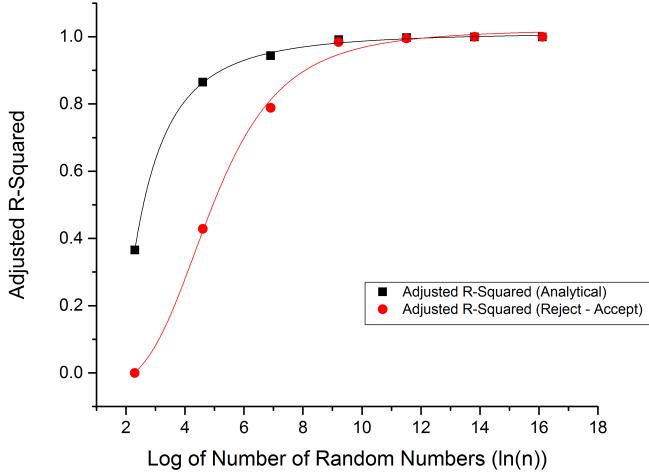


Figure 4: Logarithmic plot of how varying the number of random angles simulated affects the  $R^2$  value.

reject-accept method. Approximately 37% of data points were rejected, corresponding to the ratio of areas between the sinusoidal function and the comparison function. At high values of  $n$  this effect is not noticeable but for low values of  $n$  it becomes more evident. For this specific case therefore the analytical technique would be preferential to use, however generally the reject-accept method is more applicable as it doesn't require the definite integral or the inverse integral of the distribution function.

There are a number of possible optimisation techniques that can be used to improve the efficiency of both routines. For the inverse transformation method a recent more efficient approach has been found for inverting computationally expensive cumulative distribution functions known as the Stochastic Collocation Monte Carlo sampler<sup>[3]</sup>. This would allow for many Monte Carlo samples to be generated with only a few inversions making the method far more efficient. The main disadvantage of the reject-accept method is the loss of data points when the comparison function encloses a much greater area than the desired distribution. Adaptive rejection sampling can be used to solve this problem<sup>[4]</sup>. If the comparison function is defined in log space and rejection sampling is carried out it is possible for the comparison function to update after each rejection to reduce the chances of further rejection, improving the technique.

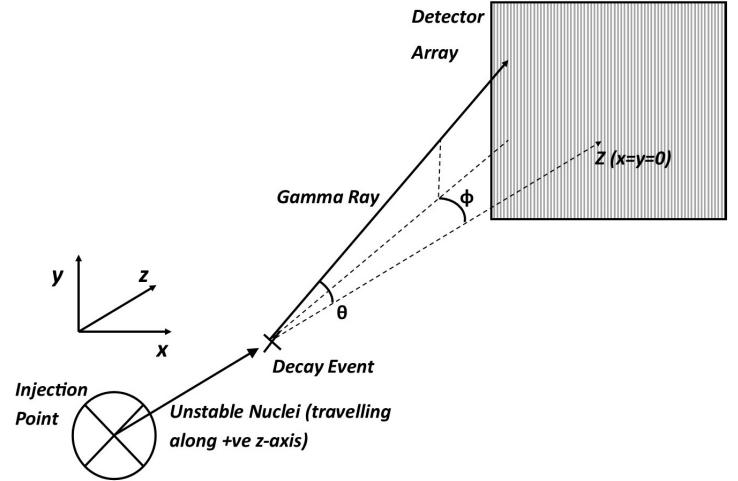


Figure 5: Diagram showing the experimental setup of the detector array for the nuclear physics simulation in task 2. The detector is placed at  $z=2\text{m}$  and the injection point is set at  $z=0\text{m}$ .

## Problem 2: Gamma Ray Distribution Simulation

### Theory and Computation

The second task was to model the physical situation of unstable nuclei decaying to produce gamma rays. A Monte Carlo simulation was run to model this however care was needed to ensure the simulation was setup correctly to give accurate physical modeling. The experimental setup is shown in figure (5). The figure shows a number of random variables will be required for different steps of the simulation. Firstly the decay length for the unstable nuclei must be modeled. The assumption that there are no background counts and the nuclei do not scatter were used to simplify the problem. The nuclei follow the exponential decay law<sup>[5]</sup> which can be easily simulated to find decay length as the mean lifetime and injection velocity are both known. Decay lengths greater than the detector distance are ignored. The remaining data points correspond to a decay event in which a gamma ray is isotropically emitted. To simulate this emission spherical coordinates,  $\theta$  and  $\phi$ , were used to give the gamma ray a specific direction.  $\phi$  was set from the generation of a random number in the range  $0 < \phi < 2\pi$  with a uniform probability distribution. However  $\theta$  was not generated using a uniform probability distribution. For the case of a sphere with radius = 1 an area element on the sphere will be  $dA = \sin\theta d\phi d\theta$  therefore the  $\theta$  distribution needs to be a sinusoidal distribution to avoid unwanted bunching of points at the sphere poles. The analytical technique used in exercise 1 was used for this as  $\sin(x)$  is an easily integrable and invertable function. For the

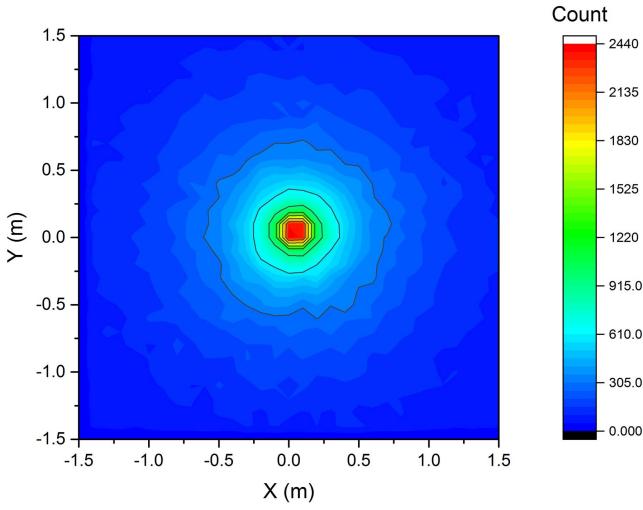


Figure 6: Colour plot showing the distribution of gamma rays recorded in the 3mx3m detector array. 1,000,000 nuclei were simulated with an injection point of 2m away from the detector.

case of a square detector as in figure (5) any gamma rays with  $\theta \geq \frac{\pi}{2}$  were ignored as these would not reach the detector.

$$x = -z\tan\theta\cos\phi \quad (5)$$

$$y = z\tan\theta\sin\phi \quad (6)$$

Using the spherical-cartesian transformation equations it is possible to derive equations, equations (5) and (6) for the (x,y) coordinates of the gamma ray when in line with the 3m x 3m square detector array, which was centred on the z-axis at z=2. Here z is the distance from the decay event to the detector array. From here the rays that hit the detector can be easily found and recorded

## Results and Discussion

Figure (6) shows the gamma ray distribution for a 3m x 3m square detector array. As expected there is a bunching of points at the centre of the detector corresponding to the restriction on where a specific  $\theta$  value would cause the gamma ray to travel as the decay length approaches the detector distance. The range of  $\theta$  that cause the gamma ray to hit the central bin increases as the decay length increases giving the bunching seen. The detector array was then given a finite resolution of 0.1m in the x-axis and 0.3m in the y-axis which will result in a smearing of the (x,y) coordinates. This smearing was modeled as Gaussian with a randomly generated Gaussian distributed smearing term added to each x and y coordinate. The Gaussian distribution was given a standard deviation

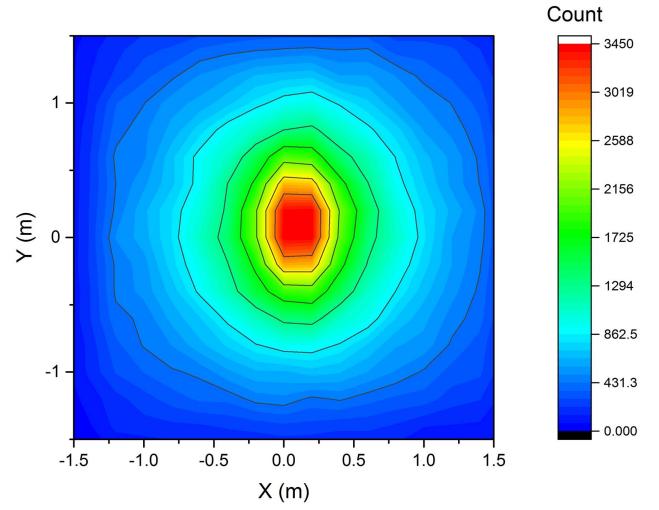


Figure 7: Colour plot showing the distribution of gamma rays recorded in the 3m x 3m detector array. 1,000,000 nuclei were simulated with an injection point of 2m away from the detector. A finite resolution of 0.1m in the x-axis and 0.3m in the y-axis was applied.

equal to the resolution in that direction. Figure (7) shows a plot for this smearing application. There is a noticeable difference to the distribution with a greater broadening seen in the y direction, as expected due to the greater resolution on the y direction. A resolution of 10% of the detector width is unacceptable if meaningful physical results are required.

Next a spherical detector was modeled to confirm the isotropy of the gamma rays. Instead of finding the ( $x, y$ ) position of the gamma ray when it was in line with the detector, the distance,  $d$ , from the gamma ray emission to the edge of a cross section of the spherical detector was found. This was calculated with the cosine rule as relevant distances were known and the value of  $\theta$  could be used as the angle. Care was taken to ensure the correct angle was used as past the sphere midpoint  $\theta$  would represent the wrong angle for the cosine calculations. The cosine rule results in a quadratic which can be solved for the positive root to find the distance to the cross sections edge,  $d$ . Then  $d$  can be put into the spherical-Cartesian transformation equations to provide a data point for where the gamma ray is detected on the sphere. Figure (8) shows the 3D scatter plot of this simulation. As expected there is a bunching of points at the injection point highlighting the short mean lifetime of the unstable nuclei as the solid angle the gamma ray has on the sphere is restricted due to the short distance it is from the edge. Another smaller bunching of points is found across the diameter of the sphere from the injection point. This bunching is due to the linearity of the beam and occurs for the same reason as it did for the square detector. This bunching can be shown in

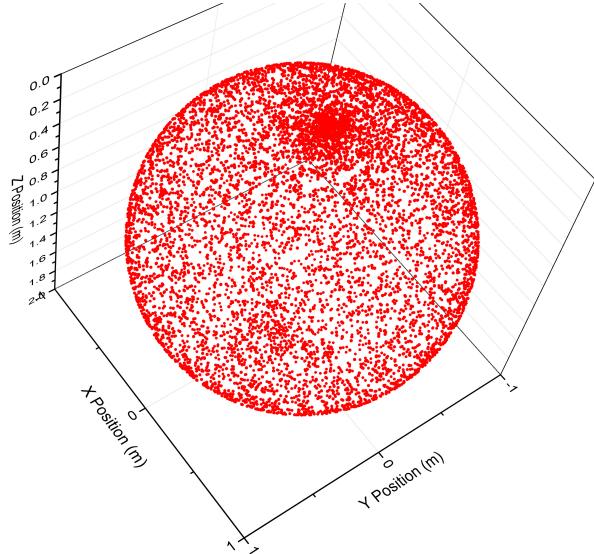


Figure 8: 3D scatter plot for a spherical detector of radius = 1m with 10,000 simulated particles.

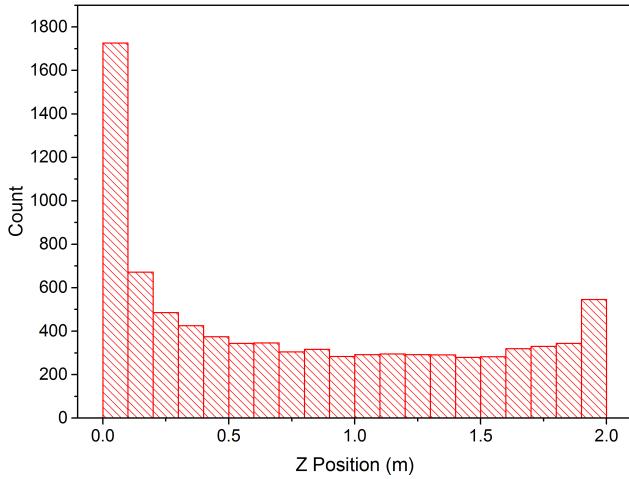


Figure 9: Histogram of the distribution of particles in the spherical detector along the z-axis. The spherical detector had radius = 1m with 10,000 simulated particles.

figure (9) where a histogram has been made from the distribution of particles along the z-axis. Both figures (8) and (9) show that apart from these poles the distribution of particles was indeed isotropic, validating this aspect of the simulation.

### Problem 3: Statistical Solutions - Toy Monte Carlo Method

#### Theory and Computation

It is often required to investigate the statistical properties of analysis procedures. This could be for problems such as the determination of confidence limits for experimental results. Many ensembles of pseudo-experiments are generated (this is known as the Toy

Monte Carlo method), where each experiment represents one possible version of the true experiment and model parameters are changed each time. The model parameters corresponding to the observed results (a model with the result within its distribution at the correct probability) can be easily identified. These pseudo-experiments are used extensively in nuclear physics<sup>[6]</sup> where applying approximations and a limit on the statistics available make the classic way of estimating experimental error unviable.

This task simulated the search for a hypothesised new particle, X. The number of X particles expected to be produced was expressed as  $n = \mathcal{L}\sigma$ , where  $\mathcal{L} = 10/nb$  is the known integrated luminosity and  $\sigma$  is the unknown production cross section. The task was to find the limit on the cross section such that it is set at the 95% confidence level. The number of background events that satisfy the criteria needed was given at  $5.8 \pm 0.4$ . A mean from a Gaussian distribution centred at 5.8 with a standard deviation was found for each pseudo-experiment. Each pseudo-experiment then generated random numbers in a Poisson distribution with the mean being the sum of the mean background radiation (the mean from the Gaussian distribution) and the number of X particles expected,  $\mathcal{L}\sigma$ . The probability of observing a value of 5 or less is then found for the pseudo-experiment when run many times. The 95% confidence limit is reached when the probability of observing 5 or less is 5% for the smallest possible cross section. To find the correct upper bound for the cross section each new pseudo-experiment used a different cross-section and an iterative convergence method, of under and overshooting the true value until the required accuracy is met, was employed. To make the results more physical a random Gaussian uncertainty term was added to the integrated luminosity each pseudo-experiment. This was set with a standard deviation of  $0.15/nb$ .

#### Results and Discussion

Figure (10) shows the resulting Poisson distribution for a valid upper bound on the cross section of  $\sigma = 0.4718099m^2$  when 100,000 pseudo-experiments were run for each test cross section. The mean on the Poisson distribution was found to be  $\lambda = 10.389$ . A simulation was then run with 10,000,000 pseudo-experiments performed each time resulting in a cross section of  $\sigma = 0.471469m^2$ . These two results differ by 0.07% meaning that 100,000 pseudo-experiments is enough for a meaningful value for the upper bound of the cross section. Investigations were then undertaken to find the minimum value for which the upper bound of the cross section can be taken as a mean-

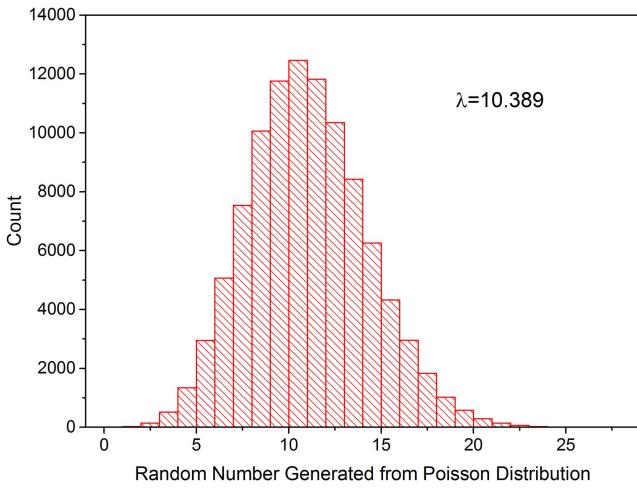


Figure 10: Histogram of the Poisson distribution for a valid upper bound on the cross section of  $\sigma = 0.4718099m^2$  when 100,000 pseudo-experiments were run for each test cross section. The mean on the Poisson distribution was found to be  $\lambda = 10.389$ .

ingful result. Figure (11) shows a plot of varying  $n$ , the number of pseudo-experiments, and the resulting cross section bound given. The scale of  $n$  is logged to show the relationship clearly. It seems that at around 800 pseudo-experiments the results converge on the true value however this is found to be a statistical fluctuation as when  $n$  is increased further this accuracy is lost. The first meaningful result occurs at  $n = 5,000$  pseudo-experiments where a value of  $\sigma = 0.46765$  is found. This is 0.81% from the best estimate and this accuracy is not lost again if  $n$  is increased further.

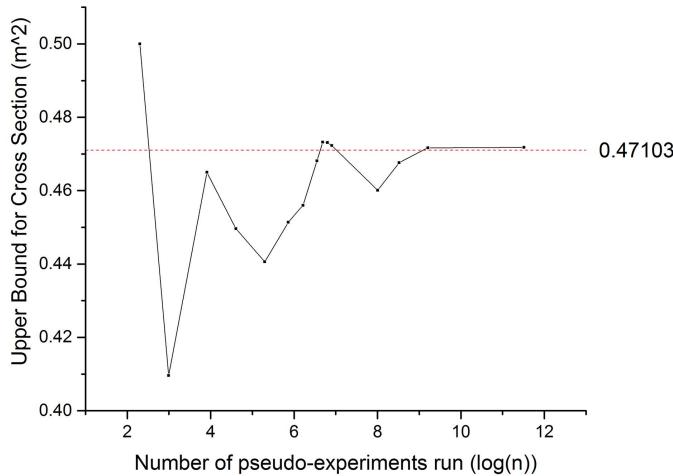


Figure 11: Plot of how the resultant cross section bound varies with the natural logarithm of the number of pseudo-experiments.

- [1] GNU Scientific Library (GSL), 1.13, (<http://www.gnu.org/software/gsl>) accessed 17/04/2016
- [2] Kroese, Dirk P., Thomas Taimre, and Zdravko I. Botev. "Handbook of Monte Carlo Methods." Vol. 706. John Wiley & Sons, 2013.
- [3] Grzelak, Lech A., et al. "The Stochastic Collocation Monte Carlo Sampler: Highly Efficient Sampling from 'Expensive' Distributions." (2014).
- [4] Ripley, Brian D. "Stochastic simulation." Vol. 316. John Wiley & Sons, 2009.
- [5] Yang, Fujia, and Joseph H. Hamilton. "Modern atomic and nuclear physics." World scientific, 2010.
- [6] Gross, Eilam, and Ofer Vitells. "Trial factors for the look elsewhere effect in high energy physics." The European Physical Journal C **70.1-2** (2010): 525-530.
- [7] Press, William H. "Numerical recipes 3rd edition: The art of scientific computing". Cambridge university press, 2007.