

**Modeling and Visualizing the Particle Beam  
in the Rare Isotope Accelerator**

Chris Rosenthal  
Office of Science, Pre-Service Teacher Program  
Illinois Institute of Technology  
Argonne National Laboratory  
Argonne, Illinois

Bela Erdelyi  
NIU, Department of Physics  
ANL, Physics Division

August 2005

Prepared in partial fulfillment of the requirements of the Office of Science, DOE Pre-Service Teacher (PST) Program under the direction of Dr. Bela Erdelyi in the Physics Division of Argonne National Laboratory.

## TABLE OF CONTENTS

Abstract	iii
Introduction	1
Materials and Methods	2
Results	7
Discussion and Conclusions	10
Acknowledgements	11
References	11
Tables and Figures	12

## ABSTRACT

Modeling and Visualizing the Particle Beam in the Rare Isotope Accelerator. CHRIS ROSENTHAL (Illinois Institute of Technology, Chicago, IL 60616) BELA ERDELYI (Argonne National Laboratory, Argonne, IL 60439).

Argonne National Laboratory is actively pursuing research and design for a Rare Isotope Accelerator (RIA) facility that will aid basic research in nuclear physics by creating beams of unstable isotopes. Such a facility has been labeled as a high priority by the joint Department of Energy and National Science Foundation Nuclear Science Advisory Committee because it will allow more study on the nature of nucleonic matter, the origin of the elements, the Standard Model, and nuclear medicine. An important part of this research is computer simulations that model the behavior of the particle beam, specifically in the Fragment Separator. The Fragment Separator selects isotopes based on their trajectory in electromagnetic fields and then uses absorbers to separate particles with a certain mass and charge from the rest of the beam. This project focused on the development of a multivariate, correlated Gaussian distribution to model the distribution of particles in the beam as well as visualizations and analysis to view how this distribution changed when passing through an absorber. The distribution was developed in the COSY INFINITY programming language. The user inputs a covariance matrix and a vector of means for the six phase space variables, and the program outputs a vector of correlated, Gaussian random variables. A variety of random test cases were conducted in two, three and six variables. In each case, the expectation values, variances and covariances were calculated and they converged to the input values. The output of the absorber code is a large data set that stores all of the variables for each particle in the distribution. It is impossible to analyze such a large data set

by hand, so visualizations and summary statistics had to be developed. The first visualization is a three-dimensional graph that shows the number of each isotope present after each slice of the absorber. A second graph plots any of the six phase space variables against any of the others to see the change in the beam's distribution. Also, the expectation values, variances and covariances of the phase space variables were calculated after the absorber. The distribution that models the particle beam gives the variability that physicists need to simulate many different situations in the Fragment Separator. The statistics and visualizations will allow quick analysis of the particle beam. Both of these developments will contribute to the overall viability of the RIA proposal.

## INTRODUCTION

Argonne National Laboratory (ANL) is actively pursuing research and design for a Rare Isotope Accelerator (RIA) facility that will aid basic research in nuclear physics. The joint Department of Energy (DOE) and National Science Foundation (NSF) Nuclear Science Advisory Committee stated in their long range plan that “The Rare Isotope Accelerator (RIA) is our highest priority for major new construction” [8]. RIA will allow further study on the nature of nucleonic matter, the origin of the elements, the Standard Model, and nuclear medicine. It will make this research possible by creating beams of unstable, short-lived isotopes for scientific analysis.

There is a lot of planning that goes into the design for such a facility. An important part of this design is the creation of a computer model of the beam line. In the Fragment Separator, isotopes with a specific mass over charge ratio are selected based on their trajectories in electromagnetic fields. Those with the same mass over charge ratio are further separated by an absorber that allows for the selection of a specific isotope by decoupling mass and charge. This project focuses on modeling and visualizing the particle beam in the Fragment Separator.

The development of such a model requires a multivariate, correlated Gaussian distribution to model the particle beam before it passes through the absorber. Each particle in the distribution is given six variables (phase space variables) that track its position, momentum, and energy. This distribution will allow physicists to vary the dependence and mean of the six phase-space variables as they run simulations. This type of flexibility will make it possible to test the fragment separator in a variety of conditions as research is done to design RIA.

As the beam passes through the absorber, the particle distribution changes and new particles are developed. Currently, very large output files of numbers are created as the result of

the absorber simulation. An extension was written to represent the data meaningfully in a graphic form and calculate statistics on the large data set. This will allow physicists and non-physicists alike to make quick and accurate analysis of the particle beam distribution generated by the fragment separator model.

These two important pieces, creating a multivariate, correlated Gaussian distribution and generating graphics to show how the distribution that models the particle beam changes, will be integrated into the existing model in the COSY INFINITY language. “COSY INFINITY is an arbitrary order beam dynamics simulation and analysis code. It allows the study of accelerator lattices, spectrographs, beamlines, electron microscopes, and many other devices” [7]. It is especially useful in the Fragment Separator simulations because its implementation of differential algebra allows it to quickly compute high-order derivatives.

## MATERIALS AND METHODS

The development of the Gaussian distribution that models the particle beam required research in both mathematics and computer science. A random vector  $X = [X_1, \dots, X_N]$  follows a multivariate, Gaussian distribution [6], [12], [13] if there is a vector  $\mu = [\mu_1, \dots, \mu_N]$  and a symmetric, positive definite covariance matrix  $S$  ( $N \times N$  matrix) such that  $X$  has density

$$p(X_1, \dots, X_N) = \frac{1}{\sqrt{(2\pi)^N |S|}} e^{-\frac{1}{2}(X-\mu)^T S^{-1}(X-\mu)}.$$

The covariance matrix of  $x_i$  is  $S$ , which is equal to

$$S \equiv \begin{pmatrix} \langle X_1^2 \rangle & \langle X_1 X_2 \rangle & \dots \\ \dots & \langle X_2^2 \rangle & \dots \\ \dots & \dots & \dots \end{pmatrix},$$

where  $\langle X_1^2 \rangle$ ,  $\langle X_2^2 \rangle$ , ...,  $\langle X_n^2 \rangle$  are the variances of the Gaussian variants and  $\langle X_1 X_2 \rangle$ ,  $\langle X_i X_j \rangle$ , ...,  $\langle X_n X_n \rangle$  are the covariances.

In order to create such a distribution, an uncorrelated X was created, with means  $\mu = 0$ , and a covariance matrix D, such that

$$D \equiv \begin{pmatrix} \langle X_1^2 \rangle & 0 & \dots \\ 0 & \langle X_2^2 \rangle & \dots \\ \dots & \dots & \dots \end{pmatrix}.$$

In this case, D was diagonal with all of the covariances equal to zero. Then the density function was rewritten as

$$p(X_1, \dots, X_N) = \frac{1}{\sqrt{(2\pi)^N |D|}} e^{-\frac{1}{2} X^T D^{-1} X}.$$

Next, a linear transformation A was applied to X, such that  $Y = AX$  and  $\det(A) \neq 0$ . This meant that  $X = A^{-1}Y$  and  $X^T = (A^{-1}Y)^T = Y^T (A^{-1})^T$ . Looking back at the exponential part of the density function,  $X^T D^{-1} X$ , this was rewritten as  $Y^T (A^{-1})^T D^{-1} A^{-1} Y$ . Therefore the density function for a multivariate Gaussian distribution in Y was written as

$$p(Y_1, \dots, Y_N) = \frac{1}{\sqrt{(2\pi)^n |S|}} e^{-\frac{1}{2} Y^T S^{-1} Y},$$

where S, the covariance matrix was given by

$$S = \{(A^{-1})^T D^{-1} A^{-1}\}^{-1} = ADA^T.$$

Since A was chosen to be orthogonal,

$$\begin{aligned} A^{-1} &= A^T, \\ |A| &= \pm 1, \\ |S| &= |ADA^T| = |D|. \end{aligned}$$

This allowed the density function to be rewritten as

$$p(Y_1, \dots, Y_N) = \frac{1}{\sqrt{(2\pi)^n |D|}} e^{-\frac{1}{2} Y^T S^{-1} Y}.$$

Then  $Y$  could generate a correlated Gaussian distribution centered at zero. In order to allow for any mean as an input,  $Z$  was defined as a function of  $\mu$  and  $S$  where

$$Z = Y + \mu.$$

Based on this theory, the COSY function developed took a covariance matrix,  $S$ , as an input to describe the dependence between the Gaussian random variables that were generated. The other inputs were the number of variables ( $n$ ), and a vector ( $\mu$ ) containing the desired means for the variables. It output the vector  $Z$  as a set of correlated Gaussian random variables. Generally, it is expected for this calculation to be done in a six-dimensional phase space, but the algorithm was generalized for any number of variables.

In order to develop such a function, first the covariance matrix  $S$  was diagonalized and the diagonalization matrix  $T$  was stored. Then,  $n$  independent Gaussian random variables were generated, using the Eigen values of  $S$  as the variances. Finally,  $T$  was multiplied by  $W$ , where  $W$  was the vector formed by the independent Gaussian random variables, and the resulting vector was output as a set of correlated Gaussian random variables [9].

COSY has a built-in function that writes a matrix as  $B^{-1}DB$ , where  $D$  is a diagonal matrix with the Eigen values on the diagonal. This Jordan decomposition of sorts was rewritten for better integration with the model and to decompose the matrix into the form  $BDB^{-1}$ . However, this decomposition does not guarantee that the transformation matrix  $B$  is orthogonal. Without this guarantee,  $BDB^{-1}$  does not have to equal the decomposition for  $S$ ,  $ADA^T$ . Therefore, a new decomposition, the Schur decomposition which writes a matrix as  $ADA^T$  (with  $A$  being



orthogonal) was needed. To transform from the Jordan  $BDB^{-1}$  form available in COSY to the Schur  $ADA^T$  form needed, the following calculation was performed:

$$S = ADA^T = BDB^{-1}.$$

Because A was orthogonal,  $A^T = A^{-1}$ . Therefore,  $ADA^{-1} = BDB^{-1}$ . Then it was shown that  $(B^{-1}A)D = D(B^{-1}A)$ , and thus  $B^{-1}A$  commuted with D. D was an arbitrary diagonal matrix, and only diagonal matrices commute with such matrices. Therefore,  $B^{-1}A$  was set equal to some diagonal matrix d. This meant that

$$\begin{aligned} B^{-1}A &= d, \\ A &= Bd. \end{aligned}$$

Because A was orthogonal,

$$\begin{aligned} AA^T &= I, \\ (Bd)(Bd)^T &= I, \\ Bdd^T B^T &= I, \\ BddB^T &= I, \\ Bd^2 B^T &= I, \\ d^2 &= (B^T B)^{-1}, \\ d &= \sqrt{(B^T B)^{-1}}. \end{aligned}$$

Therefore, the transformation was concisely written as

$$A = B\sqrt{(B^T B)^{-1}}.$$

Now that the covariance matrix was diagonalized in the proper form, the next step was creating the independent Gaussian random variables. This was done with the Box-Muller transform [11]. Given r and  $\phi$  independently, uniformly distributed random variables in (0,1], then

$$z_0 = \cos(2\pi\phi) \cdot \sqrt{-2\ln(r)},$$

where  $z_0$  is the Gaussian random variable. A procedure was written in COSY to generate a Gaussian random variable based on this transformation and taking a desired variance and mean. In this case the variance was the Eigen value from the covariance matrix.

Finally, the diagonalization matrix, T, obtained after changing the Jordan decomposition to the Schur decomposition was multiplied by the vector of independent, Gaussian random variables and this created a vector of correlated, Gaussian random variables. This procedure was then called many times, and the results were analyzed to determine their validity.

In order to generate valid covariance matrices to input as test cases, another math theorem was used. If a matrix A is an  $m \times n$  real matrix with  $m > n$ , then A can be written using a singular value decomposition of the form  $A = UDV^T$ . In this case, U is an  $m \times n$  matrix and V is an  $n \times n$  square matrix, both of which have orthogonal columns such that  $U^T U = V^T V = I$ , and D is a  $n \times n$  diagonal matrix [10]. This decomposition was used to generate orthogonal matrices (U,V) which could then be used with a diagonal matrix containing the test Eigen values to generate a symmetric, diagonalizable test covariance matrix S.

As part of the testing, a statistics procedure was written to recalculate the expectation values, variances, and covariances produced and display them alongside the expected input values. Optimizations were added to this computationally intensive code so that the memory for the calculation arrays was dynamically allocated. Another enhancement took advantage of the symmetric nature of the covariance matrix, only calculating the upper half of the covariance matrix and thereby cutting the runtime of the statistics calculation nearly in half.

After completing work on the particle distribution to model the beam before the absorber, the project focused on visualizing and analyzing the output files produced by the absorber simulation code. An isotope file is created after each slice of the absorber that shows how many

of each type of isotope are present in the beam. A particle file is created for each isotope present in the beam after the last slice and it contains the six phase space variables for each particle of that type.

Procedures were written in COSY to read in this data, calculate statistics on the data set, and visualize the important elements. The expectation values, variances and covariances were calculated for each of the phase space variables. These values summarized large amounts of data and were output to a single file. One visualization was developed in three-dimensions to show the amount of each type of isotope present after each slice of the absorber. Another visualization plots any of the phase space variables against any other phase space variable. The statistics and visualizations were tested with a variety of cases to ensure that they would be accurate representations of the data set.

## RESULTS

The multivariate, correlated Gaussian distribution was developed with the following two-variable test case [9]. The covariance matrix  $S$ , was given as  $S = \begin{pmatrix} 3 & 2 \\ 2 & 4 \end{pmatrix}$ . The Eigen values of  $S$  are  $\frac{(7 \pm \sqrt{17})}{2}$ , which is approximately equal to 5.562 or 1.438. The corresponding Eigen vectors are  $s_1 = \begin{pmatrix} 0.615 \\ 0.788 \end{pmatrix}$  and  $s_2 = \begin{pmatrix} 0.788 \\ -0.615 \end{pmatrix}$ . In order to generate pairs  $(z_1, z_2)$  of Gaussian random numbers with the given covariance matrix, draw  $(y_1, y_2)$  Gaussian, uncorrelated, with variances 5.562 and 1.438. Then compute  $(z_1, z_2)$  as

$$\begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} 0.615 & 0.788 \\ 0.788 & -0.615 \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

This test case was used until it was shown that the Eigen values and Eigen vectors could be calculated correctly in order to create correlated, Gaussian random numbers. The uncorrelated random variables were created with the Box-Muller transform from a uniform random number generator. However, when tested with 25,000 iterations centered at 100, the uniform random number generator was not perfectly uniform (Figure 1). This non-uniformity introduces a small amount of error.

The next test was to see if the distribution of one variable was indeed Gaussian. A single, Gaussian, random variable was created using the Box-Muller transform 25,000 times and the distribution is shown in Figure 2. The bell curve is instantly recognizable.

A procedure was written to generate somewhat random test cases in two, three and six variables. In the two variable case, rotation matrices were used as orthogonal matrices to create symmetric, diagonalizable covariance matrices to use as inputs. These rotation matrices were of the form  $\begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix}$ . The three variable tests were also created from orthogonal matrices. These were generated in Mathematica® and were of the form

$$\begin{pmatrix} a & c & b \\ b & a & c \\ c & b & a \end{pmatrix}$$

The six-variable tests took advantage of the singular-value decomposition to create 6 x 6 orthogonal matrices that could be used to generate valid covariance matrices. All of these cases created the covariance matrix from an orthogonal matrix A by multiplying  $A^{-1} * D * A$ , where D was a diagonal matrix with the Eigen values on the diagonal.

While many tests were run in two, three, and six variables during development, an example of a typical six variable test is as follows. The covariance matrix generated is given in

Figure 7. The rounded input Eigen values in the D matrix were 12.469, 4.0517, 13.839, 10.531, 18.005, and 6.1554. After the matrix was decomposed, the Eigen values were output and they were exactly the same out to 15 significant digits. The input covariance matrix was also recalculated and it was exactly the same out to 14 significant digits. The Jordan transformation matrix is given in Figure 8 and the Schur transformation matrix is given in Figure 9.

The test created 25,000 sets of six correlated, Gaussian random variables. All of the expectation values were set to zero and the output expectation values are given in Table 1. The variances and covariances were also calculated and the results are shown in Table 2. These variances and covariances should converge to the values given in the input covariance matrix.

Other tests were conducted to examine the correlation of two variables. Figure 3 shows two variables with a correlation coefficient of zero. Figure 4 shows a perfect correlation, where the coefficient is one. Figures 5 and 6 show correlations of  $\frac{2}{3}$  and  $-\frac{3}{4}$ . With 25,000 pairs of correlated, Gaussian variables in each figure, the shape of the correlation can be seen.

The visualizations were checked to ensure that they were accurate and usable in many situations. The 3D isotope visualization was developed with a small data set of three slices. The test data set can be seen in Table 3. Figure 10 shows the number of each type of isotope present before slice one. Figures 11 and 12 show the number of each type present before slices two and three. Figure 13 shows the final results after slice three. Each isotope was represented in the proper location with a scaled bar to show its height. The scales on the axes adjusted appropriately and new isotopes did show up in the correct locations on subsequent slices.

A larger data set with ten slices was also run and the final graph can be seen in Figure 14. The isotopes with the top five yields have been colored and the rest are dark blue. A filter was

created to allow physicists to focus on a particular subset of data, and an example of this can be seen in Figure 15 where only isotopes with a mass greater than 110 are considered.

The 2D plots of any one phase space variable against any other phase space variable were tested by using this filtering capability to reduce the data to a very small set. Adding in one particle at a time, it could be seen that the scale was correct and the plot was a good representation of the data set. Figure 16 shows X versus Y, which is the horizontal position of the particle versus the vertical position. This is like looking right at the particle beam. The data set is the same as the ten slice run from Figures 14. The colors of each dot correspond to the colors given to each isotope in Figure 14. Figure 17 shows X versus A, the horizontal position versus the horizontal momentum.

The final test was to make sure that the expectation values, variances and covariances that summarized all of the particle data in the particle files were accurate. This was done in a similar fashion, where the data set was first filtered to one particle. Then, as each particle was added, the new statistics were calculated by hand and confirmed.

## **DISCUSSION AND CONCLUSION**

The major question was what was causing the small amounts of error in the Gaussian distribution. It is estimated that most of the error in the calculations is due to the non-uniformity of the random number generator (Figure 1). Another generator that is included in COSY (RERAN) was tested, but was actually a bit worse (Figure 18). However, the amount of error was small and the distribution will still be quite useful for running a variety of simulations in RIA's Fragment Separator. Also, the visualizations and statistics will make it possible for physicists to analyze the results of different types of absorbers. Both of these will contribute to the research necessary to design a Rare Isotope Accelerator.

## ACKNOWLEDGMENT

The author would like to acknowledge the Office of Science and the Department of Energy for providing such a great research opportunity. Argonne National Laboratory and the Physics Division were both excellent hosts. Lou Harnisch, Nancy Nega, Frank Vivio, Natalie Kremer, Lisa Reed and the rest of the education staff were outstanding. Dr. Bela Erdelyi was a brilliant mentor who taught, challenged, and made this project possible. Finally, I would like to thank my wife Amanda, whose support and encouragement was indispensable.

## REFERENCES

- [1] M. Berz, J. Hoefkens, and K. Makino, "COSY INFINITY Version 8.1 Programming Manual," MSUHEP-20703, Department of Physics and Astronomy, Michigan State University, 2002.
- [2] M. Berz and K. Makino, "COSY INFINITY Version 8.1 User's Guide and Reference Manual," MSUHEP-20704, Department of Physics and Astronomy, Michigan State University, 2002.
- [3] O. Knill. (2005, June). Multivariable Calculus in Probability [Online]. Available: [http://abel.math.harvard.edu/archive/21a\\_spring\\_03/handouts/statistics.pdf](http://abel.math.harvard.edu/archive/21a_spring_03/handouts/statistics.pdf)
- [4] T. Barlow, "Task 23: COSY Absorber Addition," Argonne National Laboratory, Argonne, IL, June 2005.
- [5] T. Barlow, "Task 25: COSY Absorber Addition," Argonne National Laboratory, Argonne, IL, June 2005.
- [6] E. Kreyszig, *Advanced Engineering Mathematics*, 8<sup>th</sup> ed. Hoboken: John Wiley & Sons, Inc., 1999.
- [7] The COSY INFINITY web page [Online]. Available: [http://www.bt.pa.msu.edu/index\\_files/cosy.htm](http://www.bt.pa.msu.edu/index_files/cosy.htm)
- [8] *Opportunities in Nuclear Science A Long-Range Plan for the Next Decade*, The DOE/NSF Nuclear Science Advisory Committee, April 2002.
- [9] F. J. Vesely. (2001, October). Introduction to Computational Physics Course Material [Online]. Available: <http://homepage.univie.ac.at/Franz.Vesely/cp0102/dx/dx.html>

- [10] Eric W. Weisstein. "Singular Value Decomposition." From *MathWorld*--A Wolfram Web Resource. <http://mathworld.wolfram.com/SingularValueDecomposition.html>
- [11] Wikipedia, "Box-Muller Transform," [Online]. Available: [http://en.wikipedia.org/wiki/Box-Muller\\_transform](http://en.wikipedia.org/wiki/Box-Muller_transform)
- [12] R. Walpole, R. Myers, S. Myers and K. Ye, *Probability & Statistics for Engineers & Scientist*, 7<sup>th</sup> ed. Upper Saddle River: Prentice-Hall Inc., 2002.
- [13] Wikipedia, "Multivariate Normal Distribution," [Online]. Available: [http://en.wikipedia.org/wiki/Multivariate\\_normal\\_distribution](http://en.wikipedia.org/wiki/Multivariate_normal_distribution)

### TABLES AND FIGURES

	Input Expectation Value	Experimental Expectation Value (25,000 iterations)
E(X)	0.0000000000000000	-.1164935988960410E-001
E(A)	0.0000000000000000	-.2874439761976170E-001
E(Y)	0.0000000000000000	.5855313062040197E-002
E(B)	0.0000000000000000	-.1004700561425346E-001
E(L)	0.0000000000000000	.4222684426054750E-002
E(D)	0.0000000000000000	-.1427094209738337E-001

Table 1 – Expectation Values

	X	A	Y	B	L	D
X	10.671	.20779	-1.4274	.34818	2.6728	1.3420
A		11.307	.69172	4.0452	-2.0736	-2.6828
Y			9.2482	2.3314	1.8719	1.9073
B				11.326	1.4360	-2.1624
L					9.2897	-.80201
D						13.138

Table 2 – Variances and Covariances

	Before slice 1			Before slice 2			Before slice 3			After slice 3		
	Mass	Charge	Count	Mass	Charge	Count	Mass	Charge	Count	Mass	Charge	Count
1	132	50	1500	132	50	1200	132	50	1400	132	50	1700
2	100	38	1200	100	38	1100	100	38	1300	100	38	1000
3	180	10	1000	180	10	1050	180	10	800	180	10	900
4	58	80	800	58	80	900	58	80	400	58	80	750
5	150	20	500	150	20	300	150	20	350	150	20	550
6				120	8	250	120	8	300	120	8	400
7							45	93	150	45	93	100

Table 3 – Test Data Set for Isotopes Graph (three slices)



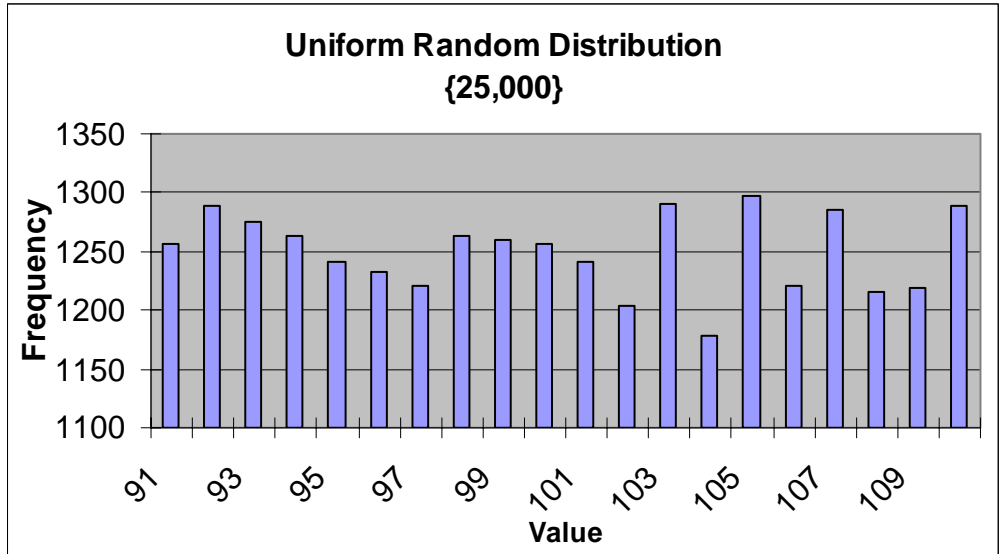


Figure 1 – Uniform Random Distribution

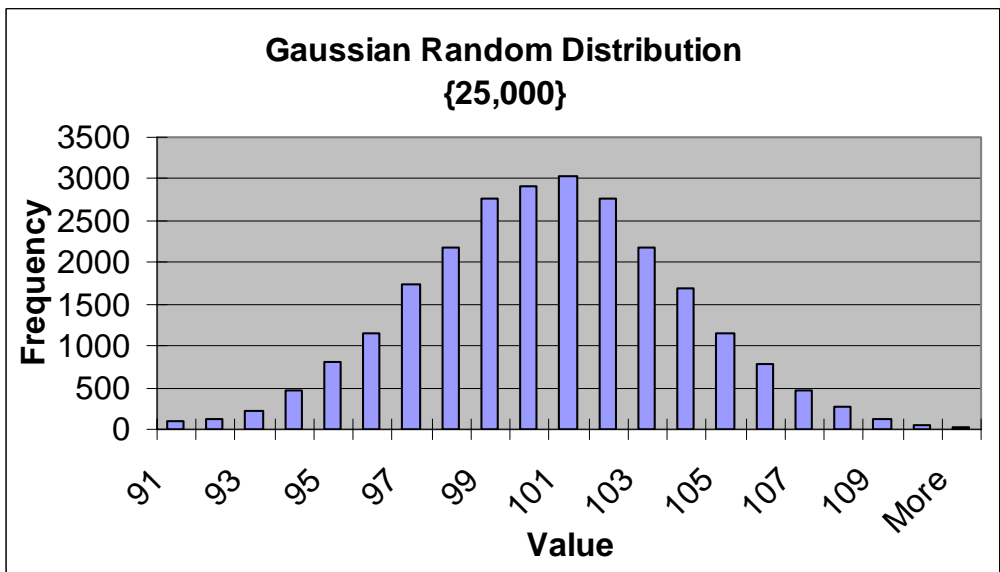


Figure 2 –Gaussian Random Distribution

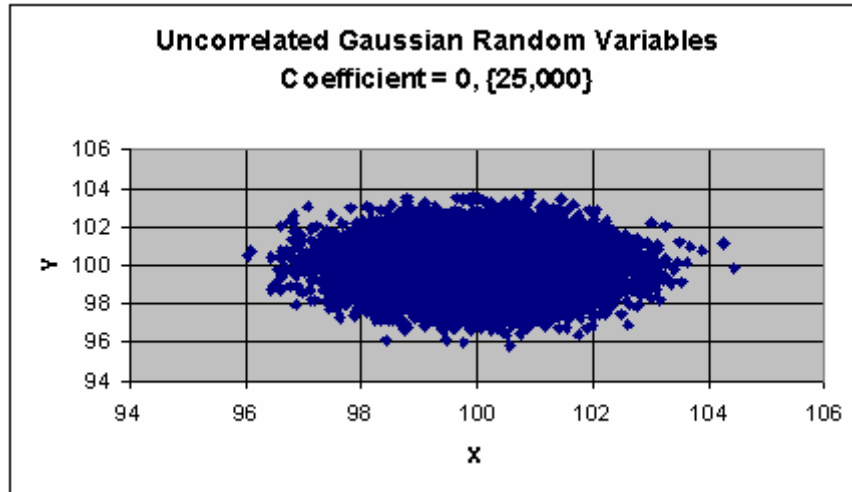


Figure 3 – Uncorrelated Gaussian Random Variables

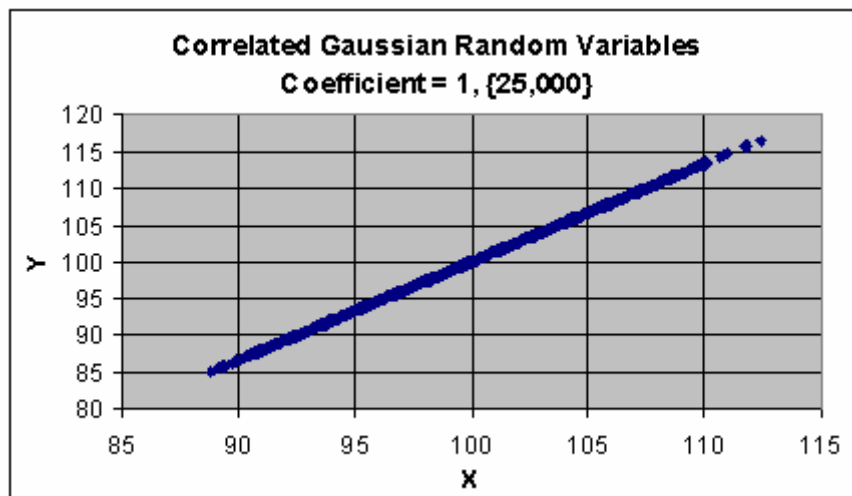


Figure 4 – Correlated Gaussian Random Variables (Coefficient = 1)

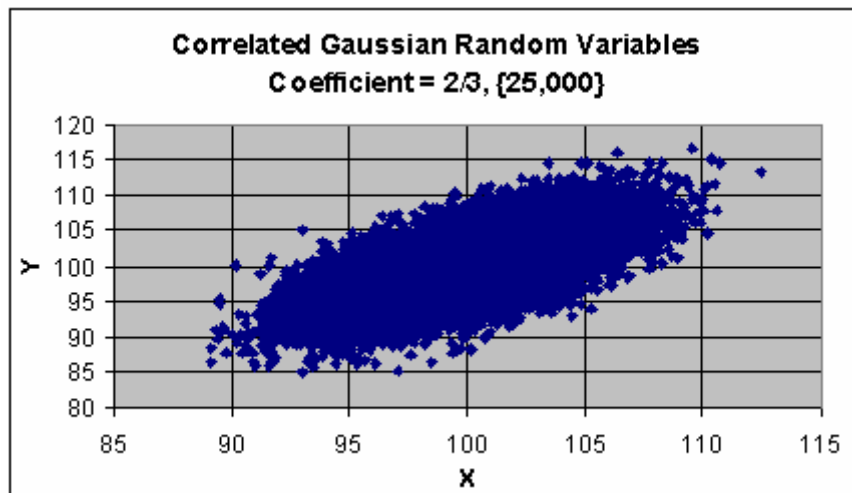


Figure 5 – Correlated Gaussian Random Variables (Coefficient = 2/3)

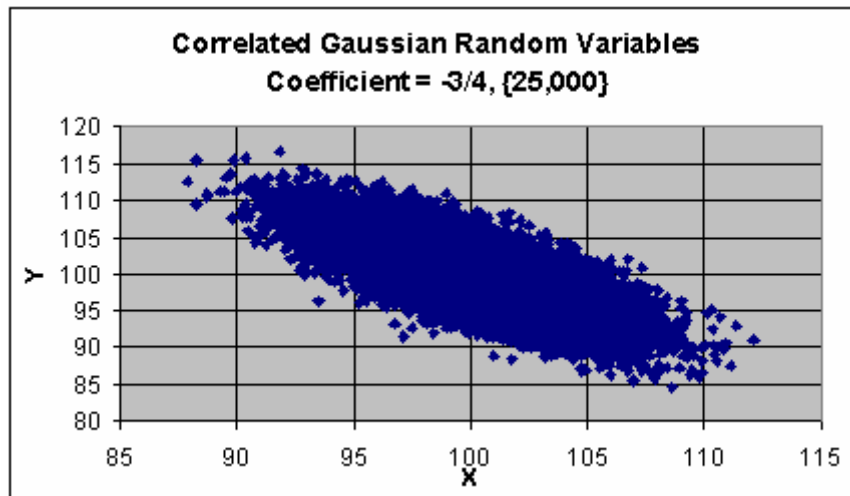


Figure 6 – Correlated Gaussian Random Variables (Coefficient = -3/4)

$$\begin{pmatrix} 10.598 & .18021 & -1.3869 & .39434 & 2.6909 & 1.2784 \\ .18021 & 11.444 & .77670 & 4.1156 & -1.9797 & -2.6044 \\ -1.3869 & .77670 & 9.3385 & 2.3855 & 1.8617 & 2.0057 \\ .39434 & 4.1156 & 2.3855 & 11.462 & 1.4981 & -2.1241 \\ 2.6909 & -1.9797 & 1.8617 & 1.4981 & 9.0835 & -.84000 \\ 1.2784 & -2.6044 & 2.0058 & -2.1241 & -.84000 & 13.125 \end{pmatrix}$$

Figure 7 – Input Covariance Matrix

$$\begin{pmatrix} .41612 & -.07196 & .12966 & -1.1971 & .57873 & 1.1005 \\ -.36983 & .70910 & .45016 & -1.1276 & -.23231 & -.07353 \\ .46467 & .12289 & .47397 & .81362 & -.43161 & 1.2441 \\ .06911 & .70659 & -.69796 & .06343 & -.09364 & 1.0310 \\ -.61197 & .02728 & .18875 & .90775 & .43519 & 1.2549 \\ -.30797 & -.65899 & -.18193 & -.82532 & -.47606 & 1.1901 \end{pmatrix}$$

Figure 8 – Jordan Transformation Matrix

$$\begin{pmatrix} .41612 & -.05961 & .12966 & -.54214 & .57873 & .42144 \\ -.36983 & .58737 & .45016 & -.51068 & -.23231 & -.02816 \\ .46467 & .10179 & .47397 & .36846 & -.43161 & .47644 \\ .06911 & .58530 & -.69796 & .02872 & -.09364 & .39484 \\ -.61197 & .02260 & .18875 & .41109 & .43519 & .48057 \\ -.30797 & -.54587 & -.18193 & -.37376 & -.47606 & .45577 \end{pmatrix}$$

Figure 9 – Schur Transformation Matrix

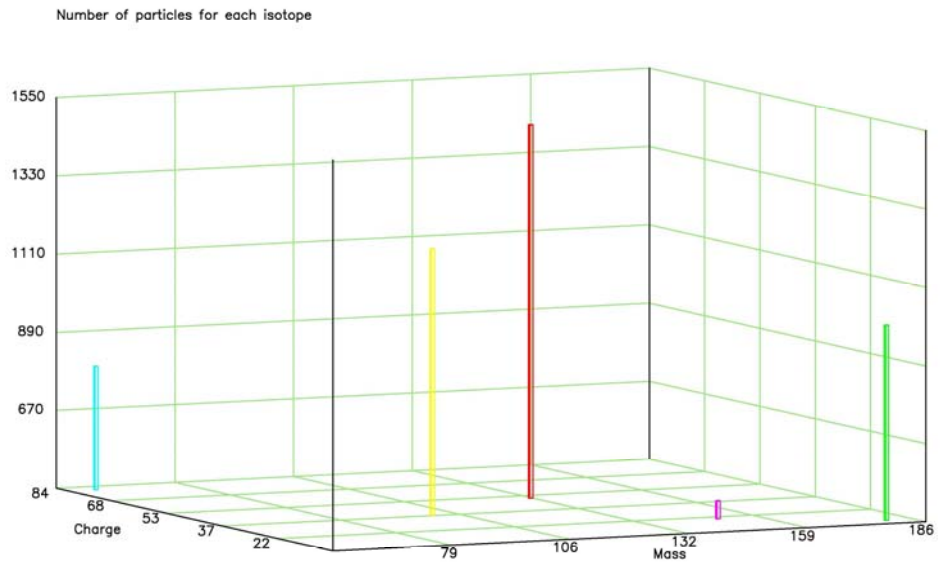


Figure 10 – Isotope Graph Before Slice One (Test Data Set)

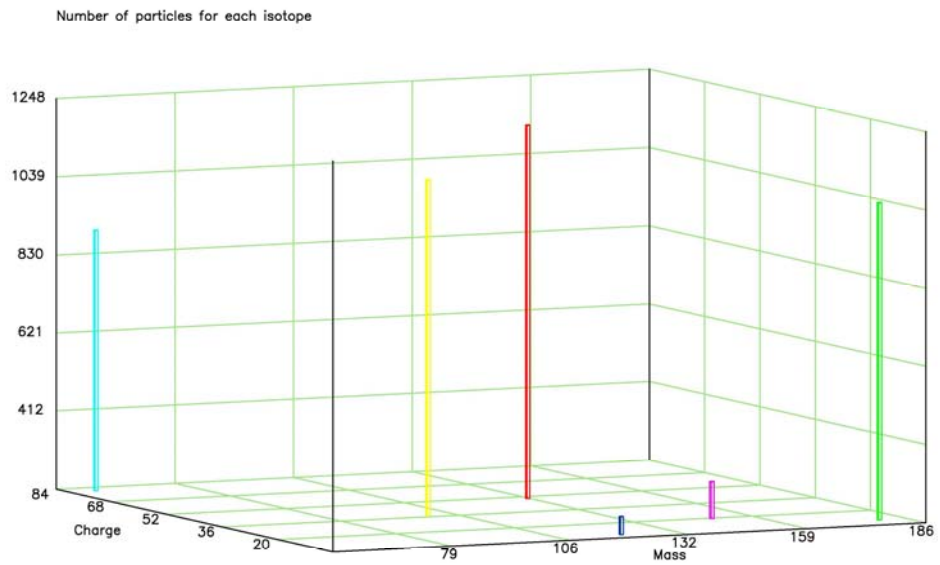


Figure 11 – Isotope Graph Before Slice Two (Test Data Set)

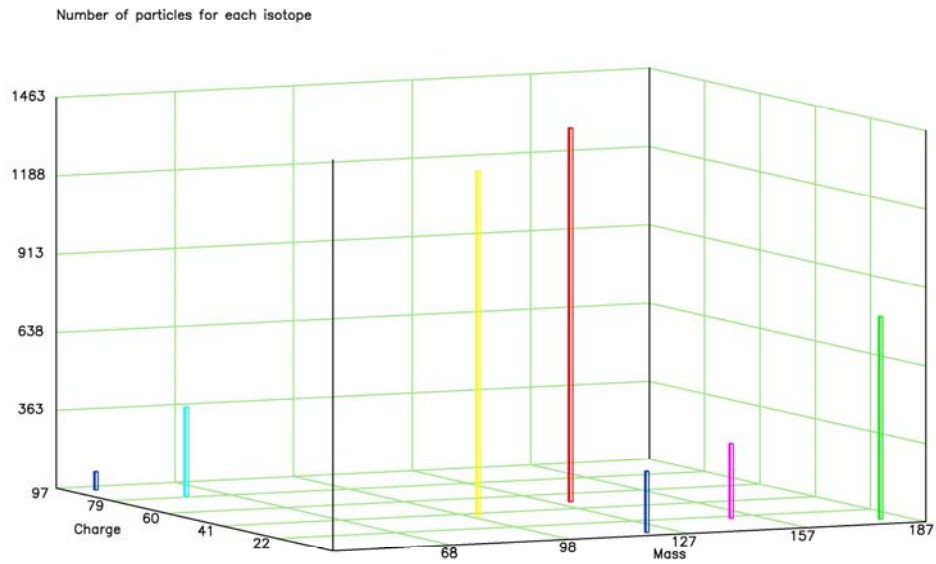


Figure 12 – Isotope Graph Before Slice Three (Test Data Set)

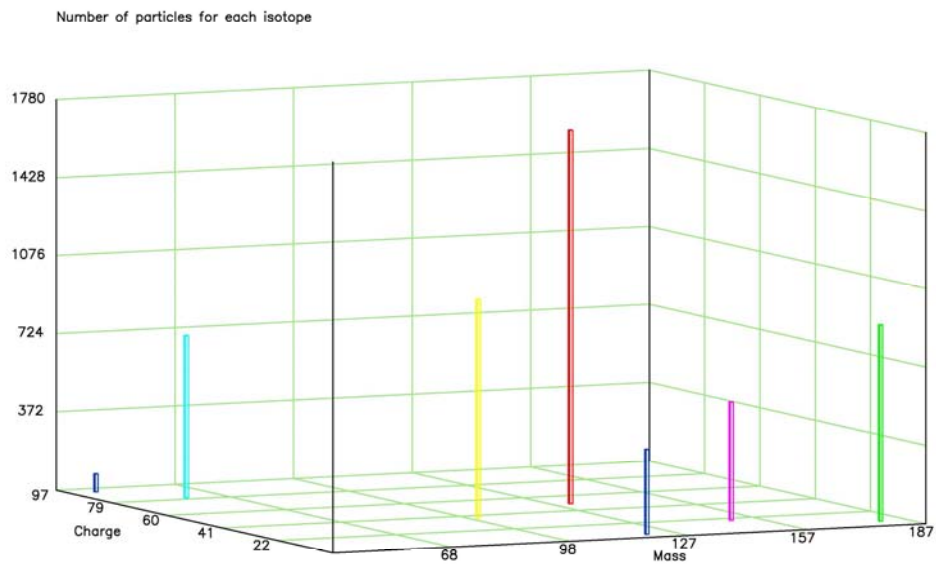


Figure 13 – Isotope Graph After Slice Three (Test Data Set)

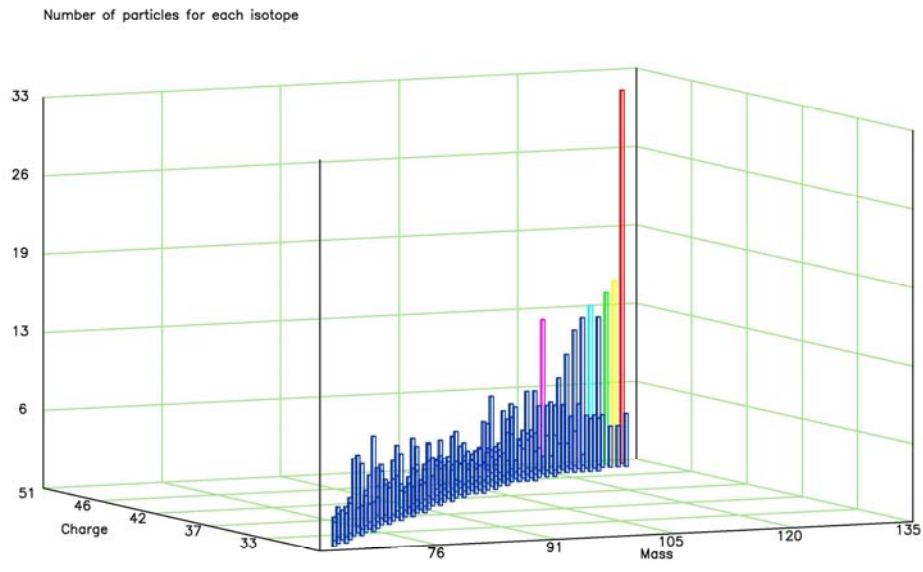


Figure 14 – Isotope Graph After Slice Ten (Large Data Set)

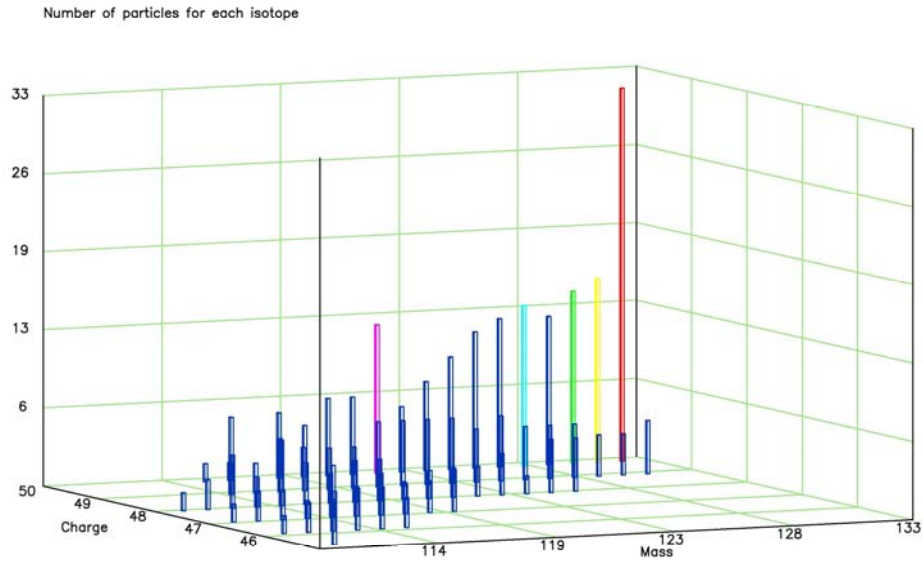


Figure 15 – Filtered Data Subset (mass > 110)

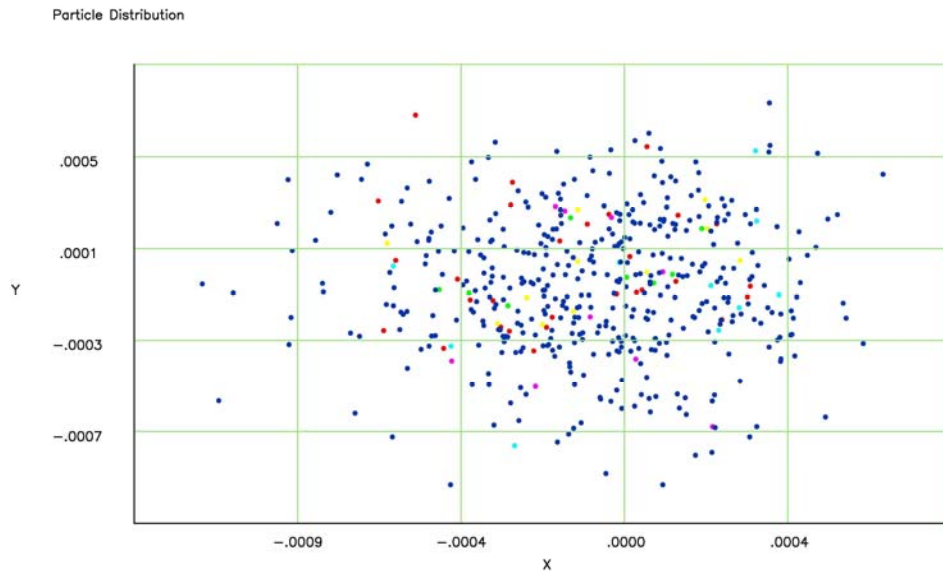


Figure 16 – Particle Plot of X vs. Y (horizontal vs. vertical position)

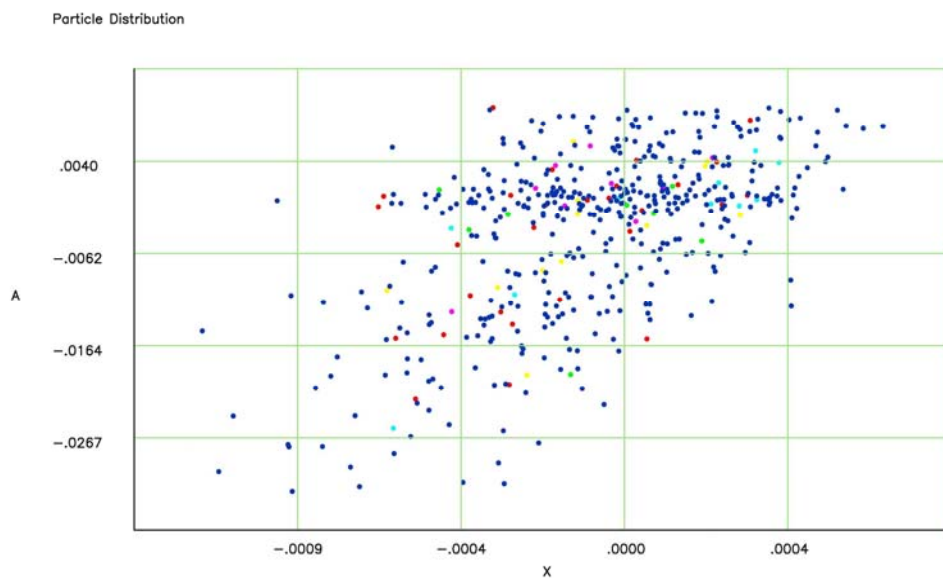


Figure 17 – Particle Plot of X vs. A (horizontal position vs. horizontal momentum)

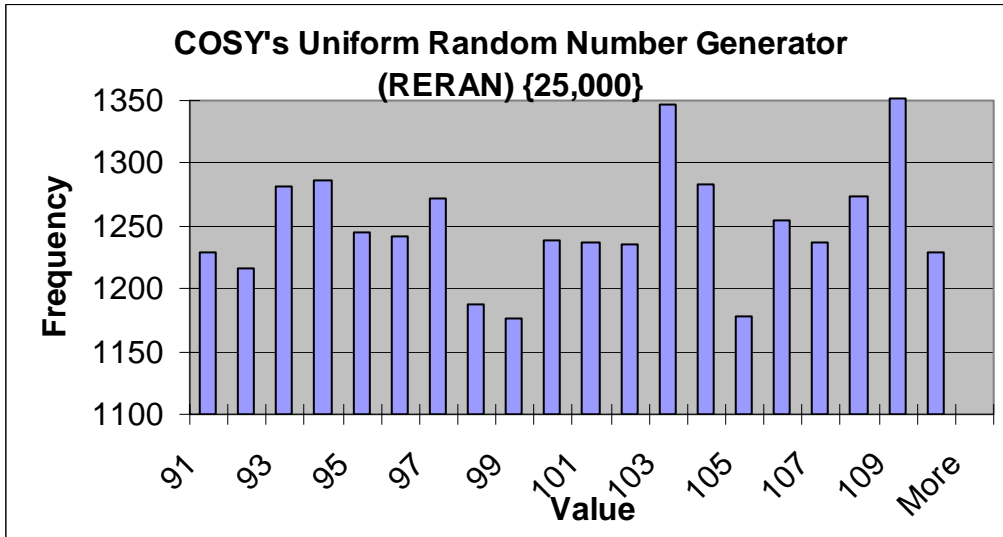


Figure 18 – COSY's Uniform Random Number Generator (RERAN)