SAND2017-4784C

Uncertainty Quantification of a Genetic Algorithm for Neutron Energy Spectrum Adjustment[1]

Danielle R. Redhouse[2]

ABSTRACT

GenSpec is software designed to use a genetic algorithm for neutron energy spectrum adjustment [1]. Currently GenSpec can produce adjusted spectra, but the corresponding covariance matrix is not produced. The uncertainty quantification process implemented includes a parametric sensitivity analysis of the genetic algorithm modifiers for population, generations, gene-sites, polynomial order, and mutation rate. A random perturbation analysis was used to characterize the covariance of the GenSpec program using Cholesky decomposition and multivariate normal random sampling. The produced 640 by 640 covariance matrix has retained some characteristic features of the sampled covariance. The uncertainty found in the GenSpec program has minimized the covariance present in a calculated trial spectrum [2].

Keywords

Uncertainty quantification, genetic algorithm, neutron energy spectrum, random perturbation, cholesky decomposition, multivariate normal random sampling

[1] Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525 SAND No. 2017-4728 T

[2] Sandia National Laboratories, Applied Nuclear Technologies, Albuquerque, New Mexico 87185-1146, USA.

INTRODUCTION

Implementation of uncertainty quantification in GenSpec is essential to the completion of the neutron energy spectrum adjustment code. While GenSpec can produce adjusted neutron energy spectra, the question of how uncertainties propagate through the code becomes important in interpreting the merit of the generated data. Therefore, GenSpec is an incomplete program without uncertainty quantification and applicable covariance data. GenSpec relies on neutron activation analysis (NAA) and a trial spectrum that is calculated using a transport code, such as MCNP in its adjustment process. However, it is not known how if the adjusted spectra have minimized errors, as are expected [1][3].

PROBLEM STATEMENT

The overall goals for a neutron adjustment code and of GenSpec, are two-fold [3][4]. First, is to bring the reaction probabilities calculated into a better agreement with experimentally measured reaction probabilities. Secondly, an adjusted spectrum should contain smaller uncertainties than those associated with the calculated trial spectrum [5]. GenSpec, the program, needs to be expanded to produce its uncertainty data so it may be used as a reliable neutron energy adjustment code, and its produced spectra can be assessed for validity.

THEORY

In order to quantify the uncertainty of GenSpec, propagation of the errors will be done using perturbation theory. The sources of uncertainty in GenSpec arise from both calculation uncertainty and adjustment uncertainty. Calculation uncertainties are those present within the model used to provide the calculated data, such as the trial spectrum attained through MCNP. The adjustment

uncertainty is due to experimental measurements and reaction cross-sections [6][7]. We can now simply state GenSpec's sources of uncertainty.

- Nuclear Cross-Sections;
- Trial Spectrum from MCNP;
- Genetic algorithm and its inputs;

We can assume that these input uncertainties for the both the trial spectrum and nuclear data cross-sections will be covariance matrices [7][8]. Another assumption is that these covariance matrices will be square, making them easily factorized. It is likely that these matrices will be positive-definite or positive semi-definite, which implies that they are a complex-square matrix whose diagonal elements must be real and must be their own complex conjugate [3][6]. Should the covariance matrices not be positive definite, a multiplication by its first eigenvalue can be done to make it positive definite [9]. Providing these assumptions, a process such as Cholesky decomposition can be implemented. The basic theory of applying Cholesky decomposition in a numeric process is shown below, beginning with an *n* by *n* matrix called *A*.

$$A = \begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & \ddots & \vdots \\ a_{n,1} & \cdots & a_{n,n} \end{bmatrix} \tag{2.1}$$

For Cholesky decomposition, assume matrix *A* has the form:

$$A = LL^T \tag{2.2}$$

$$\begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & \ddots & \vdots \\ a_{n,1} & \cdots & a_{n,n} \end{bmatrix} = \begin{bmatrix} l_{1,1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ l_{n,1} & \cdots & l_{n,n} \end{bmatrix} \begin{bmatrix} l_{1,1}^T & \cdots & l_{1,n}^T \\ \vdots & \ddots & \vdots \\ 0 & \cdots & l_{n,n}^T \end{bmatrix} \tag{2.3}$$

where L is the lower triangular matrix, with real and positive diagonal entries, and L^T is the

conjugate transpose of L. In Cholesky factorization it is assumed that every positive definite matrix like A can be factored as seen in Equation (2.2), where L is called the Cholesky factor of A and is defined as the "square root" of a positive definite matrix [9]. The next step in developing the uncertainty through perturbation theory, will include multivariate normal random sampling of the covariance data for the trial spectrum and nuclear cross-sections. This will allow for different realizations of the same distribution to be made.

Multivariate normal distributions are simply a higher-dimensional form of a normal distribution. A collection of variables is jointly distributed according to some mean, or mean vector of values, while the covariance matrix signifies the relation between the variables of each. First, we have a collection of data called X, which in this case is the spectrum distribution that is a mix between a Maxwellian distribution and a Watts-Fission distribution. We will consider this data to be a multivariate normal such that, $X \sim \mathcal{N}(\overline{\mu}^{>}, \Sigma)$. This distribution has a vector of mean values μ contained in $\overline{\mu}^{>}$ and a covariance matrix Σ, both of some dimension d. In general, a covariance matrix measures how dependent each individual dimension, and/or energy group, is to the other. Below is the mathematical form for a covariance matrix in expectation form and in expanded form:

$$\Sigma = E\left[(X - \vec{\mu})\left(X - \overrightarrow{\mu^{T}}\right)\right] = \begin{bmatrix} \sigma_{1,1} & \cdots & \sigma_{1,d} \\ \vdots & \ddots & \vdots \\ \sigma_{d,1} & \cdots & \sigma_{d,d} \end{bmatrix} \tag{2.4}$$

where X is some number in a collection of data, $\overline{\mu}^{>}$ is a vector of discrete mean values μ, $\overline{\mu}^{>T}$ is the transpose of the $\overline{\mu}^{>}$, E is the mathematical representation of an expectation calculation, $\sigma_{i,j}$ is a discrete covariance value in some i^{th} row and j^{th} column in dimension d, and Σ is a covariance matrix of dimension d. To sample from this distribution, a Cholesky decomposition of the covariance matrix must be done such that $\Sigma = LL^{T}$, if Σ is positive definite [9]. We can then

generate *d* independent samples from a standard normal random variable

$$\vec{z} = (z_1, \dots, z_d)^T, \;\; Z_i \sim \mathcal{N}(0,1) \tag{2.5}$$

To get a sample from *X*, we then compute

$$x = \mu + LZ \tag{2.6}$$

To demonstrate how this procedure works, we look at a covariance matrix of some normal vector $Z \sim \mathcal{N}(0,1)$. In terms of the expected value, the covariance of *Z* is

$$\Sigma(Z) = E[ZZ^T] = I \tag{2.7}$$

where I is simply the identity matrix. Now, consider a non-normal vector X = LZ. The covariance of a collection of random variables is then

$$E = [XX^T] = E[LZ(LZ)^T] = E[LZZ^TL^T] \tag{2.8}$$

$$\therefore E[XX^T] = LE[ZZ^T] = LL^T = \Sigma(X) \tag{2.9}$$

In changing this result to a variable with a non-zero mean, we simply add the desired mean μ [9].

Additionally, to further understand GenSpec, a local and global sensitivity analysis will be done. The motivation of a sensitivity analysis can be widely viewed as quantifying the relative contributions due to individual parameters or inputs and determining how variation in parameters affects the programs response [10]. A study of just the genetic algorithm modifiers should be done to test the genetic algorithm itself, the major processing component of GenSpec for possible errors. This genetic algorithm local sensitivity analysis is a random perturbation of the modifiers of population, generation, gene-sites, polynomial order, and mutation. The local sensitivity analysis will focus on the local parameter of fitness in each generation, while the global analysis will focus on the resulting adjusted spectrum. Studying these inputs will also reduce the epistemic uncertainty that may be overlooked in the code based on untested, user-selected inputs [10].

METHODOLOGY

GenSpec aims to adjust neutron energy spectra in nuclear reactors, specifically test reactors that perform experiments. For this analysis we will be assessing the free-field environment of the SNL Annular Core Research Reactor (ACRR) central experimental cavity. It is currently assembled in an annulus configuration in order to accommodate large experiments. The designed epithermal spectrum allows the fluence to be tailored to the desired specifications, using spectrum modifying "buckets." Moderators or absorbers can be used to either thermalize or harden the resulting spectrum within the cavity. For this study none of the spectrum modifying buckets were used as the free-field neutron environment was favored. For an unmoderated/non-absorbed condition, the neutron fluence at the axial centerline of the central cavity is 2.0E13 n/cm^2 per MJ of reactor energy. Approximately 46% of the neutron fluence is above 100 keV and 58% above 10 keV [11]. Figure 1 shows an image of the coupled ACRR and FREC-II. Figure 2 also shows a more detailed ACRR model in MCNP [12].

Neutron Activation Analysis

In total, 21 different foil types were irradiated in ACRR at the central axial centerline, and 33 reactions rates were used in the spectrum adjustment. The foils were all irradiated in a total of 10 different pulse operations at 150 MJ. The four fission reaction foils were irradiated individually in both a cadmium cup and boron ball configuration at steady-state operations. The selection of foils and activation reactions was chosen based on previous studies that have been conducted over several years [13]. The choice of foil and activation reactions were also chosen based on expert judgment and are not believed to be of issue [11][13][14]. Additionally, an ideal set of activation

foils that allows for the entire neutron energy spectrum to be calculated from dosimetry does not exists, but enough reactions do exist to result in a high resolution spectrum after an energy spectrum is adjusted [11].

Uncertainty Quantification

The process to quantify the uncertainty of GenSpec entails sampling components to develop the uncertainty data as well as test the program for faults. A missing piece of analysis data for this research is the possible error associated with just the genetic algorithm and its modifiers, that the user selects when running GenSpec. To also quantify these errors, a random perturbation analysis to quantify its uncertainty via a local and global parametric sensitivity analysis of the genetic algorithm and its modifiers will be done.

Local and Global Parametric Sensitivity Analysis

For the analysis of the genetic algorithm and its modifiers, a base case was chosen. The base case modifiers are as follows in Table 1. These were chosen as the inputs since they were preloaded for the free-field 640-group input file in GenSpec. The base case inputs assume that when a variable is changing, all other parameters remain static at base case values. From the original GenSpec report, variables could easily have limits identified.

The genetic algorithm and its modifiers are explained briefly below, but additional information can be found in the report under citation [1]. Mutation rate is essentially a percentage of the specimen population spectra from 0% to 100% that will randomly undergo a mutation of its gene-sites. Mutation in GenSpec ensures that the solution space is thoroughly explored. Polynomial

order is the order of the fitted polynomial into the gene-sites that creates the shift function after parents have undergone recombination and a child spectrum is being produced. The polynomial order dictates how important each gene-site is to the shift of the original trial spectrum, by either ensuring the polynomial goes through every gene-site or just best fits the gene-site trend. The manual suggests not exceeding a 10th order polynomial, as it can lead to unrealistic spectral artifacts [1]. Gene-sites are the number of points of interest distributed in base 10 logarithmic space between the minimum and maximum energy values. These are initially randomly selected traits for the first generation, and inherited traits in subsequent generations. The manual suggests values should ideally be two to three times the value of the polynomial order, but also suggests that large values increase computation time [1]. Generations is the number of iterative generational runs, where the spectra evolve. Each generation runs the exact same population input. While convergence is problem-dependent, the manual states that most cases will achieve convergence in less than 1000 generations [1]. Population defines the number of specimen spectra produced in each generation and is a fixed value, that is reiterated over each generation. The only notes in the manual on values of population are that at higher values, the likelihood of convergence to a sub-optimal solution is reduced, but can greatly increase computation time [1]. The manual also specifies that population values should be at least greater than 200 [1].

Once these limits were identified, a uniform distribution was made to evenly sample the entire parameter solution space to test each parameter's sensitivity on both a local and global level. In Table 2, are the defined limits of this parametric analysis and the number of cases run in-between each. While the same number of cases for each parameter was desired, values such as polynomial order and mutation rate had limits to the input variable characters, such as whole integer values

and floating digits, respectively. A smaller number of inputs were also favored, as computation times were not characterized for this program.

For the global analysis, GenSpec is run some *N* number of times, with random input for population, generation, gene-sites, polynomial order, and mutation rate swapped in each time. Once a 640 by N matrix is formed, MATLAB's covariance function *cov()* can be simply applied resulting in a 640 by 640 matrix for the covariance of the genetic algorithm modifiers [10] [15].

Random Perturbation Analysis

The implementation step in the uncertainty quantification process was kept relativity fluid in its approach due to the number of component codes and script complexity of the current GenSpec program. Two possible procedures are available for this step. The first will be to add an external *C* GenSpec component, that can be used when a covariance is needed. This external *C* component will contain the mathematical coding for the multivariate normal random sampling and the Cholesky decomposition. Additions to the GenSpec code can be made to carry out perturbation runs.

The second procedure that will be used, is to write a separate uncertainty script in MATLAB which can be run after the optimization is done in GenSpec. For this procedure, some changes to the GenSpec code will still be made to simply create text files in the working directory and print variable values into them. A completely separate MATLAB script can then be made to carry out the perturbation process as a post-processing script.

RESULTS

Local and Global Parametric Sensitivity Analysis

For the global and local sensitivity study, data analysis was done in MATLAB so the quick use of the Statistics and Machine Learning ToolBox could be utilized. For each of the parameters, two graphs were produced to show the effects each modifier had on fitness, the local parameter. The fitness value should ultimately lead us to understand what modifiers may affect the shape of the output spectra and the variance produced by each parameter. The first graph displays the average (green), maximum (red), and minimum (blue) fitness plotted across each variable's limits. This facilitates the display of convergence or divergence within fitness due to the variations in the parameters [9]. Second is a graph of the variance seen in the fitness over the increase in each parameter. Within this second graph, a fitted linear trend line was added to easily asses a positive or negative variance relationship. This graph aims to display either the increase or decrease in the confidence of the values of fitness. While the modifiers could greatly affect fitness, it may be of use to note if little or no variation is reflected in the output. Figures 3 through 12 show these two graphs for each parameter.

As seen with all the figures, each parameter affected the local variable fitness in different ways. While mutation and population increases saw a minimal production of unrealistic spectral artifacts in Figures 13, some of the population results were unexpected. Exceedingly high population values exhibited an increase in variations of the fitness values as seen in Figure 4, making the confidence in those values low. It was observed that due to memory issues, about half of the population runs failed to save ideal spectrum files due to their size. Other issues that were apparent were the production of below zero fitness values, leading to chaotic behavior in GenSpec. The parameters of generation, gene-sites, and mutation all seemed to produce the most positive effects to the

produced spectra. All three of these parameters lead to an overall reduction in the variance of the fitness as seen in Figures 6, 10, and 12. Population had a somewhat odd effect, and led to a noticeable “window" of ideal population values as displayed in Figure 4. The polynomial order Figure 8, displayed a divergence in the fitness function and also exhibited the greatest increase in the fitness variance.

These differences are mirrored in the resulting spectra, indicating that these local issue lead to global issues as well. In Figure 13, we see that very high variance is attributed to the polynomial order, but that the variance is to the 2^{nd} power. These results could have several different explanations. One, while fitness is a facet of the modifying parameters’ effects on the output spectrum, the shift function may also play a large role in changes to the output spectrum as well. The polynomial order is basically the "shift" function in GenSpec, and it is directly applied to the adjusted spectra. Two, this also could mean that large variation in the fitness will not necessarily carry over to the output spectrum, which could be a positive consequence. These could also be underdeveloped results as only 25 cases were run for each parameter. While these effects are important they are only relevant to the base case values, but nevertheless assist in redefining the limits of each parameter in order to random sample the genetic algorithm.

For the global sensitivity analysis, a total of 7.6×10^7 spectra in total were adjusted. A method to test for convergence was developed, and fitness convergence was the simplest. With each new iteration on the fitness parameter done, a cumulative average was produced, and an average among the files was reached at about 0.4. Since variation was observed in the resulting spectra, these were ordered into a single N by 640 matrix in MATLAB. Here N represents the number of files run,

while 640 is the spectrum energy group structure. Using the simple covariance function *cov()* from the built-in library [15], a 640 by 640 covariance matrix was found. Most of this covariance is 0 of this analysis, which is ideal [10]. Figure 14 shows the covariance of just eh genetic algorithm modifiers. Generally, simple system modifiers should have little to no effect on global errors [10]. Some covariance is displayed in the very high energy ranges near 20 MeV, which might mean the uncertainty of this region could be due to cross-sections at high energies [16], and not the modifiers.

A marked issue encountered was the continual production of very low fitness values among several files. While these runs do not necessarily 'fail', they seem to indicate a sub-optimal spectrum is converged too, suggesting these results could be incorrect. An example of a failed file feed is seen in Figure 15. A possible explanation for this issue could be a problem that arises with the freedom of the user selection of the polynomial order, as high polynomial orders can lead to very low fitness values, essentially 0. Considering a simple sine function fitted with a polynomial at various orders of M the noticeable problem is exhibited in the 9^{th} order polynomial. This example is displayed in Figure 16. It is noticeable that towards the end of the r values, a drop off of the polynomial fit below -1 occurs. Issues such as this in GenSpec are most likely leading to the occurrence of fitnesses values below zero. This most likely leads to an acceptance of all the specimen spectra, leading to the production of a possible sub-optimum solution.

GenSpec Covariance

For the GenSpec covariance random sampling, the suggested C component stated in the Methodology section was attempted. The additions sparked problems in not only the C

components of GenSpec, it had a cumulative effect on additional PERL and Python file writing scripts. This issue could not be solved, and due to time constraints, the proposed separate MATLAB file was ultimately chosen.

For the external MATLAB script, the trial covariance was found from the trial correlation matrix. Once the covariance was attained, the multivariate normal distribution sampling was done using MATLAB's *mvnrnd()* function [15]. For the trial spectrum and the cross-section data, 300 different realizations of these distributions were made, and the input files were written and stored in the *library* folder of GenSpec [1]. Two images of these 300 realizations can be found below. In Figure 17, an example of the realizations of the 23Na(n,γ)24Na reaction from the International Reactor Dosimetry Fusion and Fission database (IRDFF) is shown. Figure 18 shows the realizations of the trial spectrum for the differential neutron energy fluence. The fluence is displayed as the differential fluence since the fluence is input into GenSpec in this form. Activities were varied about their standard deviations using a random number generator and scaling to the defined limits.

Again, GenSpec underwent another random perturbation process to determine the program's covariance. For this analysis, data files were not omitted for 'failing' as this can skew the results of the covariance as noted in the previous results section [10]. In total, 1.22×10^9 spectra were adjusted for these results All the spectra were loaded into a large matrix, and MATLAB's *cov()* function was simply applied to this matrix to produce the finalized covariance [15]. In Figure 19 is a surface image of the GenSpec covariance after the random perturbation. In order to understand the produced covariance, the original trial covariance is also shown in Figure 20.

As previously mentioned in the problem statement, the goals of GenSpec, and adjustment codes are two-fold. One of these goals is that an adjusted spectrum should contain smaller or minimized uncertainties than those associated with the calculated spectrum [3][17]. In roughly comparing the two covariances, we can see what looks to be minimized covariance values of the GenSpec covariance compared to the trial spectrum covariance, as both figures have similar numeric scaling. At around energy group 475, which corresponds to an energy above 15 MeV, the GenSpec spectrum has a patched pattern of striped smaller covariance values throughout.

While the visual structure of the GenSpec covariance is odd, it is not necessarily a problem. GenSpec's continuous adjustment of the trial spectra though 'evolving' spectra and the use of a polynomial shift function may be the reason behind the results. The more important aspect of this comparison is that additional uncertainty or increases in covariance values are not introduced by GenSpec. Both covariances are predominately zero. A covariance of zero does not necessarily mean that the variables are independent [9][10]. A nonlinear relationship can still exist that would result in a covariance value of zero, but such studies are outside the scope of this research.

CONCLUSION

In conclusion, the covariance for GenSpec was obtained. While a complete reduction in covariance is not statistically possible [9], there are marked points across the covariance were smaller covariance values have been achieved. This final step to the completion of the uncertainty analysis for GenSpec, brings it closer to being ready as a software package. While the covariance for the program has been found, this does not negate the other issues encountered in the global and local sensitivity analysis.

Future work to this research includes the updating of the polynomial regression fit of the shift function to a natural interpolant spline These sections in GenSpec have proved in this research to be problem areas. The polynomial order may indeed be the greatest fault in this program, as given any random set of inputs, the 'failure' rate can be as high as 53%. Additionally, components in GenSpec can be streamlined to make edits across any part of the program easy.

With these noted changes to the code, this work will most likely be revised in the future. An update to the genetic algorithm will require the covariance characterization via random perturbation to be repeated, as a further minimization in the covariance could occur [3].

REFERENCES

[1] R.M. Vega, "GenSpec: A Genetic Algorithm for Neutron Energy Spectrum Adjustment," Technical Report SAND2014-0024 (2014)

[2] D.R. Redhouse, "Uncertainty Quantification of a Genetic Algorithm for Neutron Energy Spectrum Adjustment," Technical Report SAND2017-4728T (2017)

[3] W. Zipj, "Comparison of Neutron Spectrum Unfolding Codes," International Atomic Energy Agency: Specific Nuclear Reactors and Associated Plants, vol. 2, pp. 37-62, (1978)

[4] M. Matzke, "Propagation of Uncertainties in Unfolding Procedures," Nuclear Instruments and methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, vol. 479, pp. 230-241, (2002)

[5] F. Perey, "Uncertainty Analysis of Dosimetry Spectrum Unfolding," Technical Report CONF-771036--9 (1977)

[6] G.F. Knoll, *Radiation Detection and Measurement: 4th Ed*. (2010) John Wiley & Sons

[7] E. M. Zsolnay, R. Capote, H. K. Nolthenius, A. Trkov, "Summary Description of the New International Reactor Dosimetry and Fusion File (IRDFF)," Technical Report INDC-0616 (2012)

[8] MCNP-A General Monte Carlo N-Particle Transport Code, Version 5, Technical Report LA-UR-03-1987 (2003)

[9] R.C. Smith, *Uncertainty Quantification: Theory, Implementation, and Applications.* (2013) SIAM Publishing

[10] J.R. Taylor, *Introduction to Error Analysis, the Study of Uncertainties in Physical Measurements.* (1997) University Science Books

[11] E. J. Parma, G. E. Naranjo, R. M. Vega, L. L. Lippert, D. W. Vehar, P. J. Griffin, "Radiation Characterization Summary: ACRR Central Cavity Free-Field Environment with the 32-Inch Pedestal at the Core Centerline (ACRR-FF-CC-32-cl)," Technical Report SAND2015-6483 (2015)

[12] K. R. DePriest, P. J. Cooper, E. J. Parma, "MCNP/MCNPX Model of the Annular Core Research Reactor," Technical Report SAND2006-3067 (2006)

[13] P. J. Griffin, "A Rigorous Treatment of Self-Shielding and Covers in Neutron Spectra Determination," IEEE Transactions on Nuclear Science, Vol 42, 6, pp. 1878-1885 (1995)

[14] W. McElroy, S. Berg, T. Crockett, and R. Hawkins, "A Computer-Automated Iterative Method for Neutron Flux Spectra Determination by Foil Activation," Technical Report AFWL-TR 67-41.2, (1967)

[15] MathWorks Inc., MATLAB: The Language of Technical Computing, Desktop Tools and Development Environment: ver. 7, vol. 9. MathWorks Press, 2002.

[16] G. Stankunas, P. Batistoni, H. Stöstrand, S. Conroy, and J. Contributors, "Measurements of Neutron Yields by Neutron Activation Cross-Sections," Nuclear Instruments and methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, vol. 788, pp. 168-172, (2015)

[17] M. Reginatto, "Overview of Spectra Unfolding Techniques and Uncertainty Estimation," Radiation Measurements, Vol. 45, pp.1323-1329, (2010)

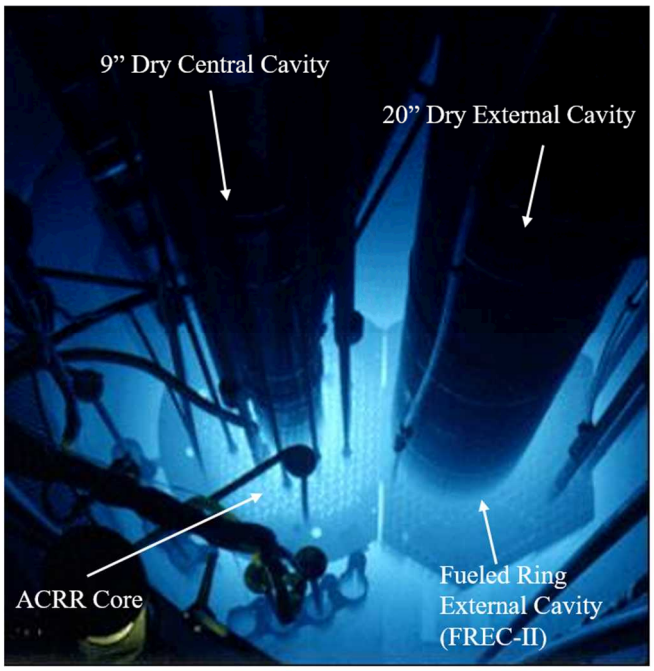

FIGURE. 1 The ACRR and FREC-II Operation at 2-MW Steady-State Power

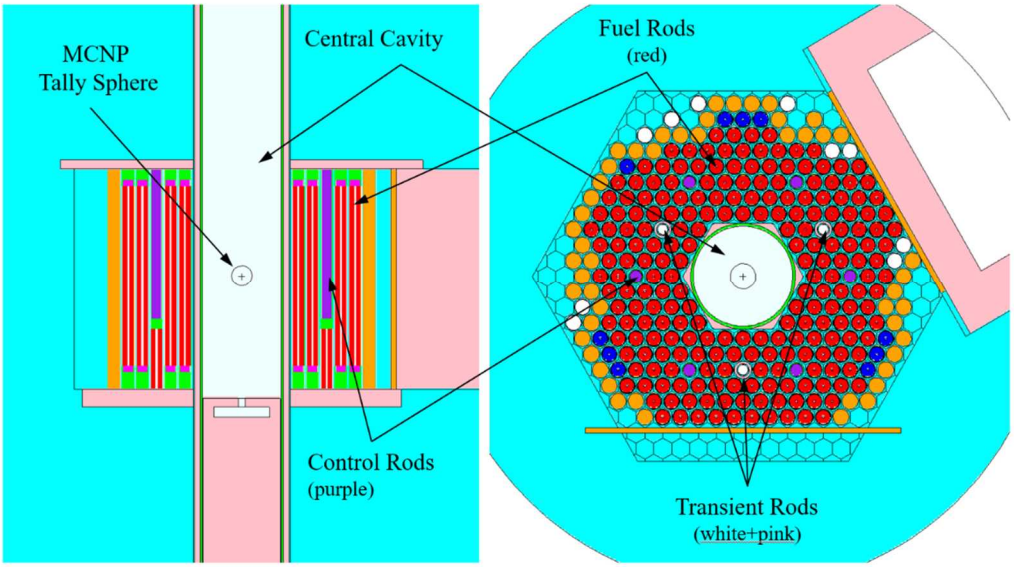

FIGURE. 2 Labeled MCNP Model of the ACRR Decoupled from FREC-II.

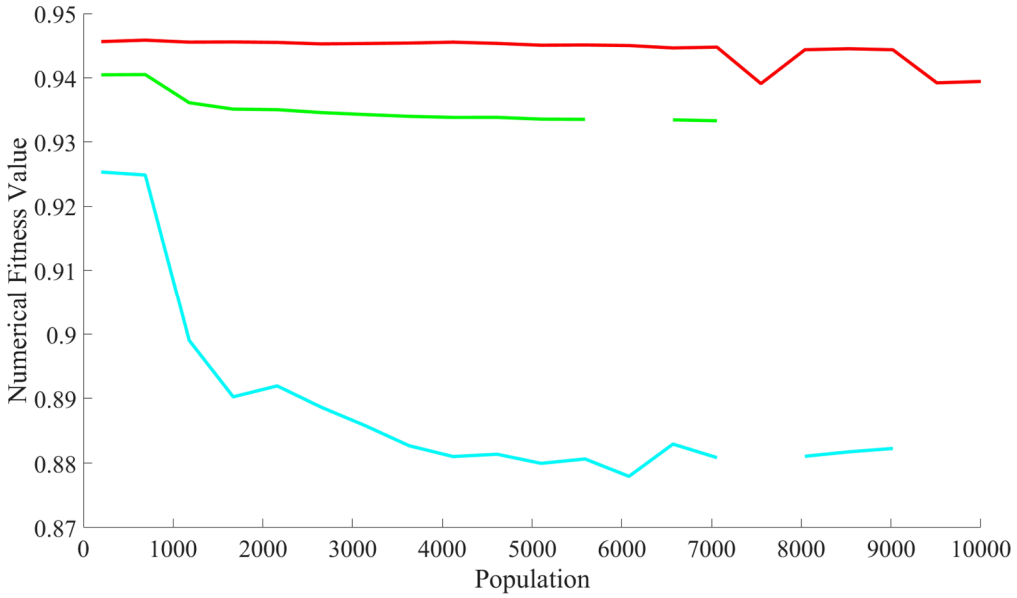

FIGURE. 3 Increasing Population Effects on Fitness

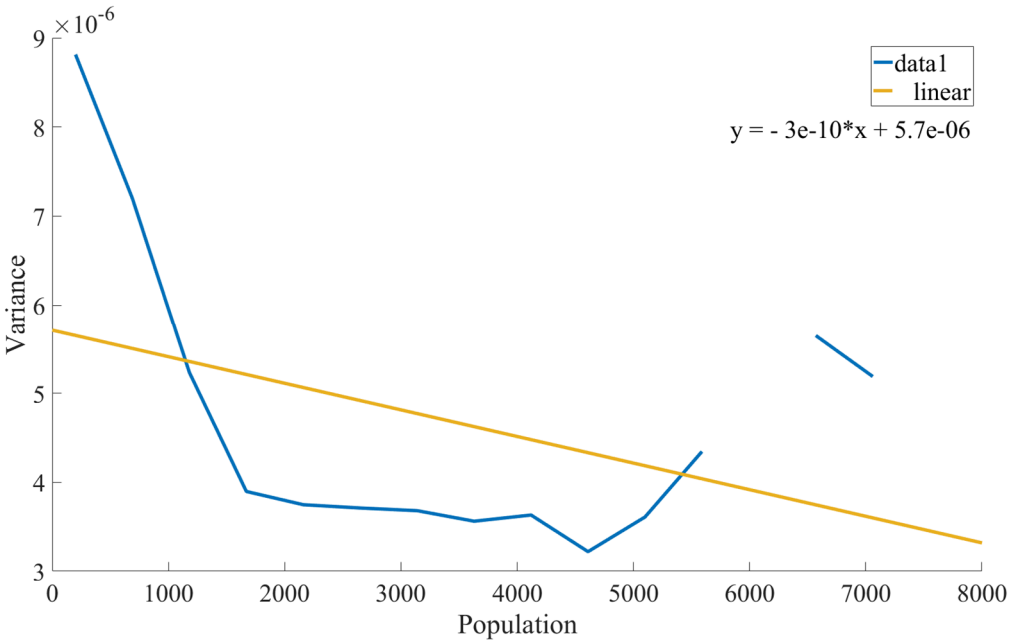

FIGURE. 4 Variance of Fitness during Population Increases.

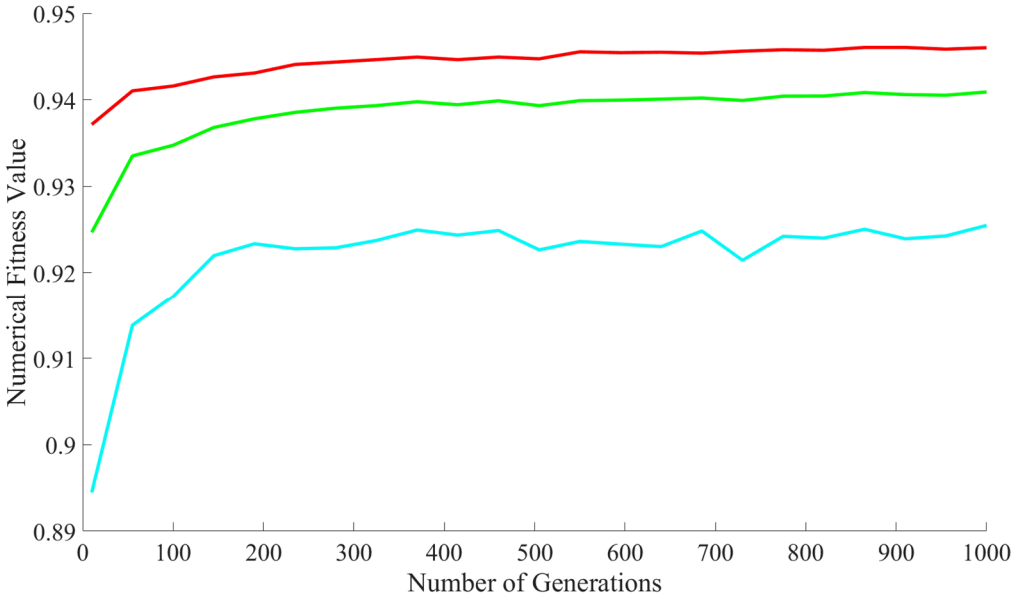

FIGURE. 5 Increasing Generation Effects on Fitness.

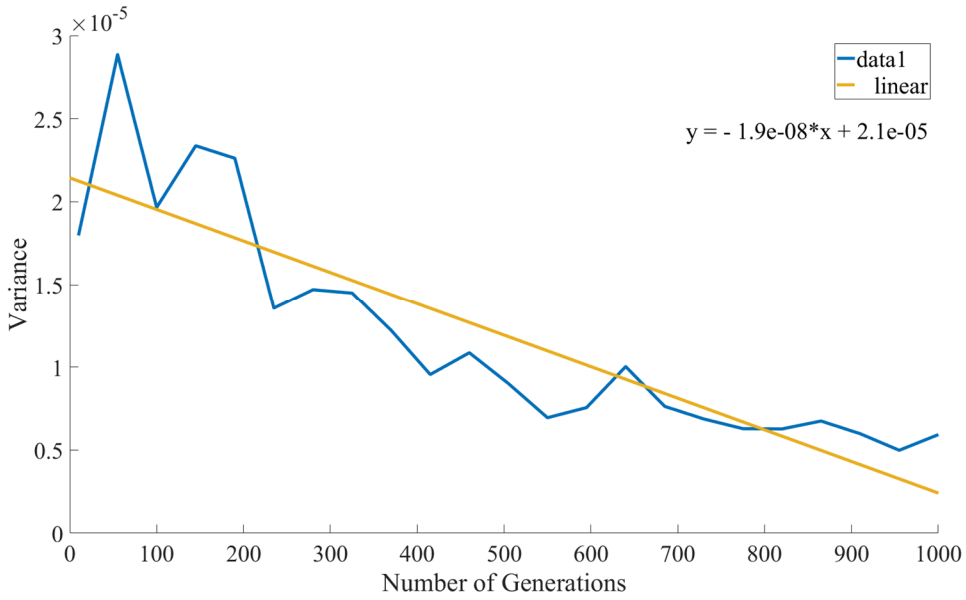

FIGURE. 6 Variance of Fitness due to Increasing Generation Value.

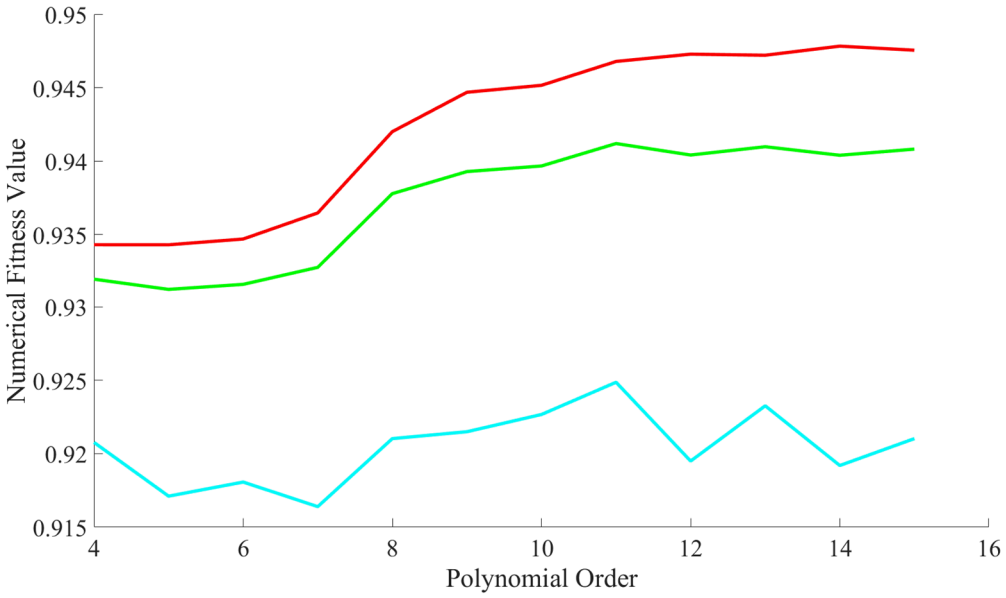

FIGURE. 7 Increasing Polynomial Order Effects on Fitness

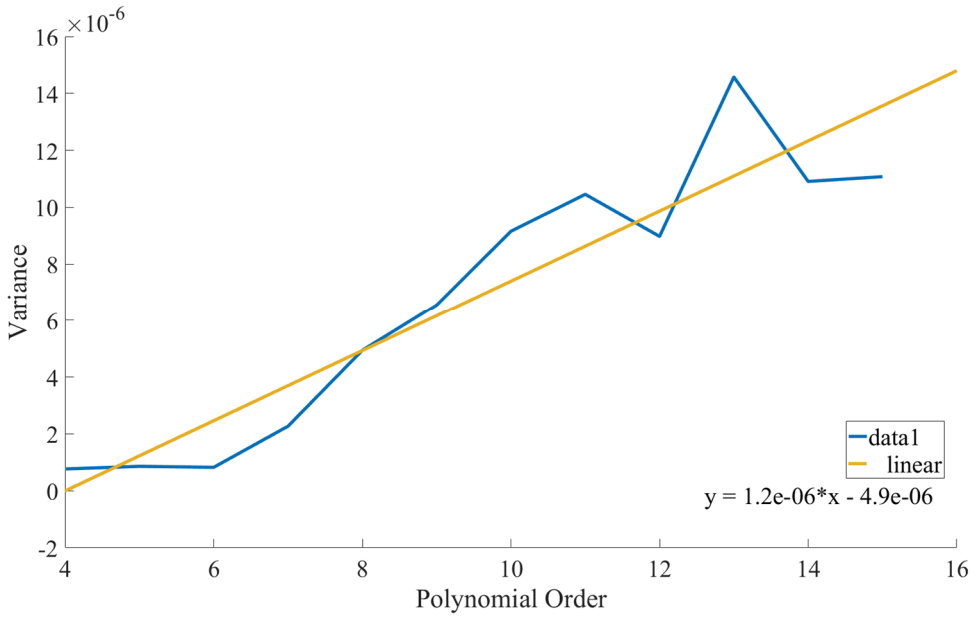

FIGURE. 8 Variance of Fitness due to Increasing Polynomial Order.

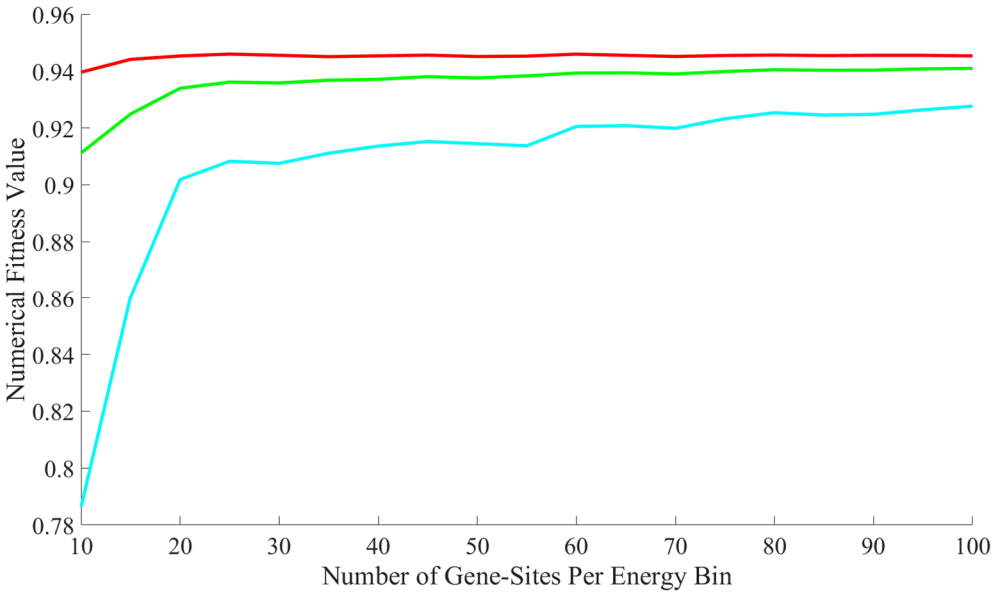

FIGURE. 9 Increasing Gene-site Effects on Fitness.

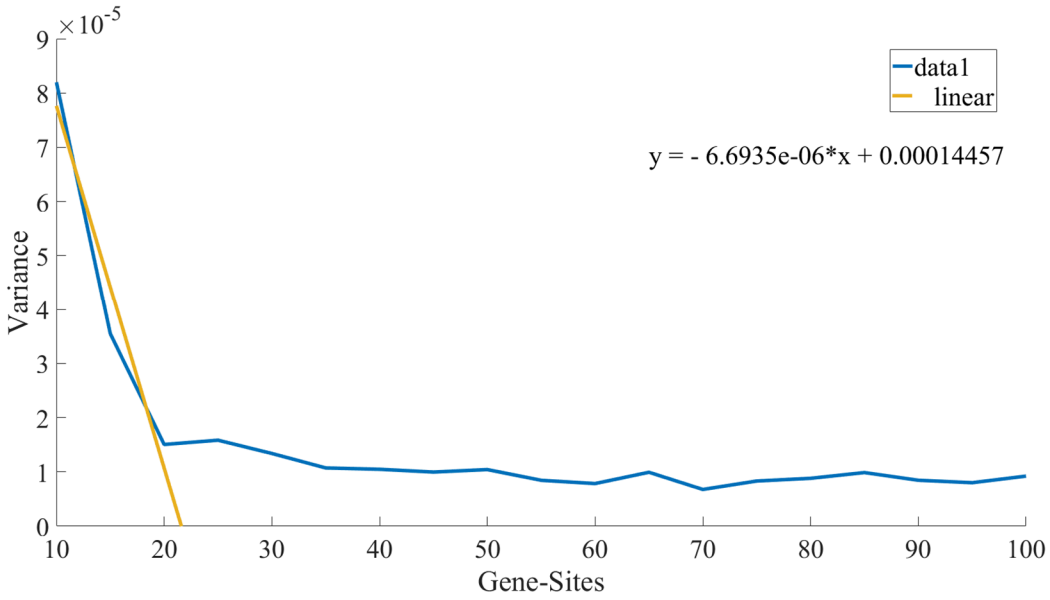

FIGURE. 10 Variance of Fitness due to Increasing Number of Gene-sites.

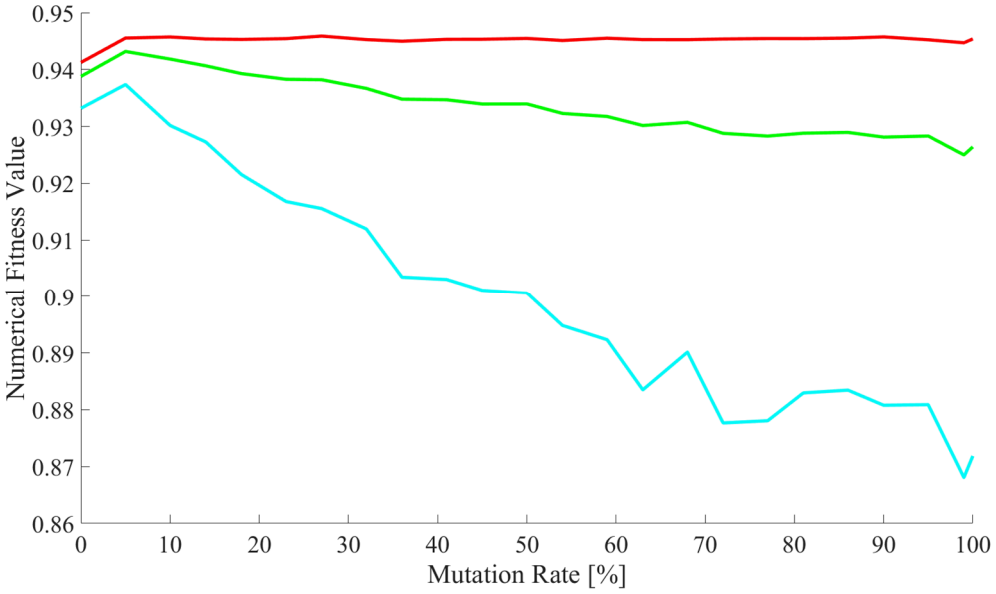

FIGURE. 11 Increasing Mutation Rate Effects on Fitness.

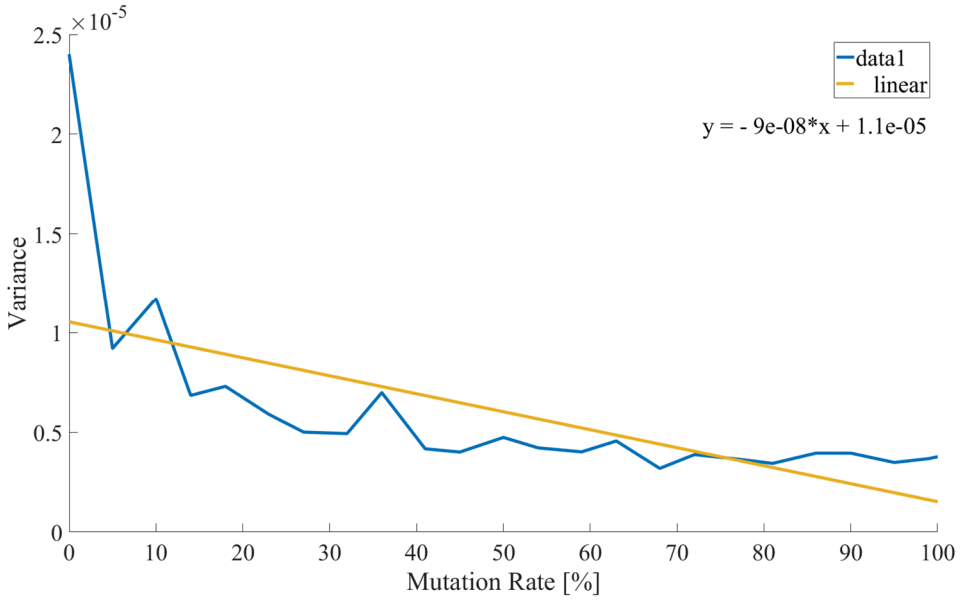

FIGURE. 12 Variance of Fitness due to Increased Mutation Rate.

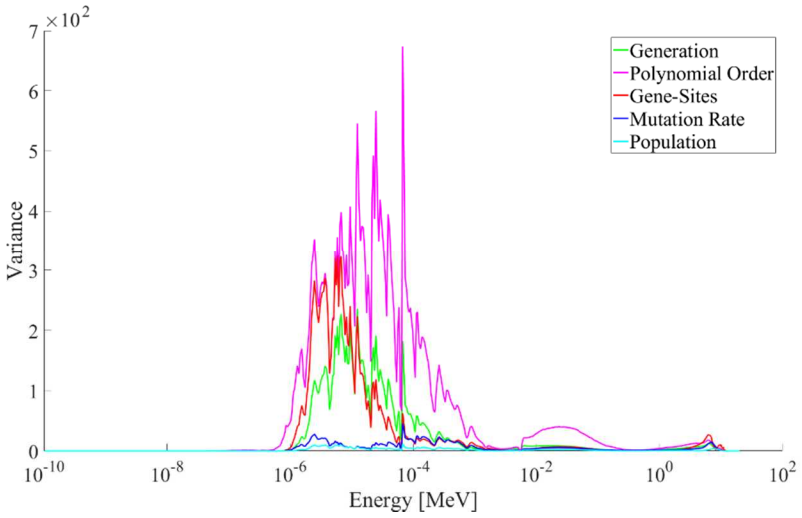

FIGURE. 13 Magnitude of Variance seen in the Fitness for each modifier.

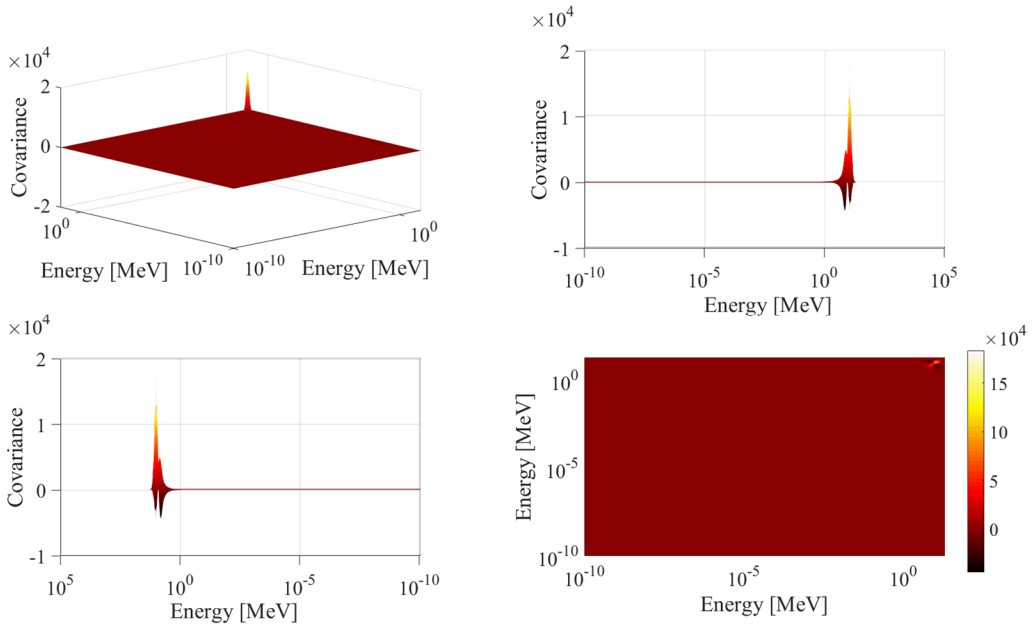

FIGURE. 14 3D and 2D Top and side views of the covariance.

```
0    0.9334541723    0.8677825208    0.9212914230CRLF
1    0.9338322232 -47714885100948542744868511604831550959391717324943785984O.0000000000 -10579797139899899114799
2    0.9338322232 -47714885100948542744868511604831550959391717324943785984O.0000000000 -10579797139899899114799
3    0.9338322232 -47714885100948542744868511604831550959391717324943785984O.0000000000 -10579797139899899114799
4    0.9338322232 -47714885100948542744868511604831550959391717324943785984O.0000000000 -10579797139899899114799
5    0.9338322232 -47714885100948542744868511604831550959391717324943785984O.0000000000 -10579797139899899114799
6    0.9338322232 -47714885100948542744868511604831550959391717324943785984O.0000000000 -10579797139899899114799
```

FIGURE. 15 Suspected Failed File showing Fitness Live Feed. The numbers in the first column are generation numbers, the 2nd, 3rd, and 4th columns are maximum, minimum, and average fitness respectively.

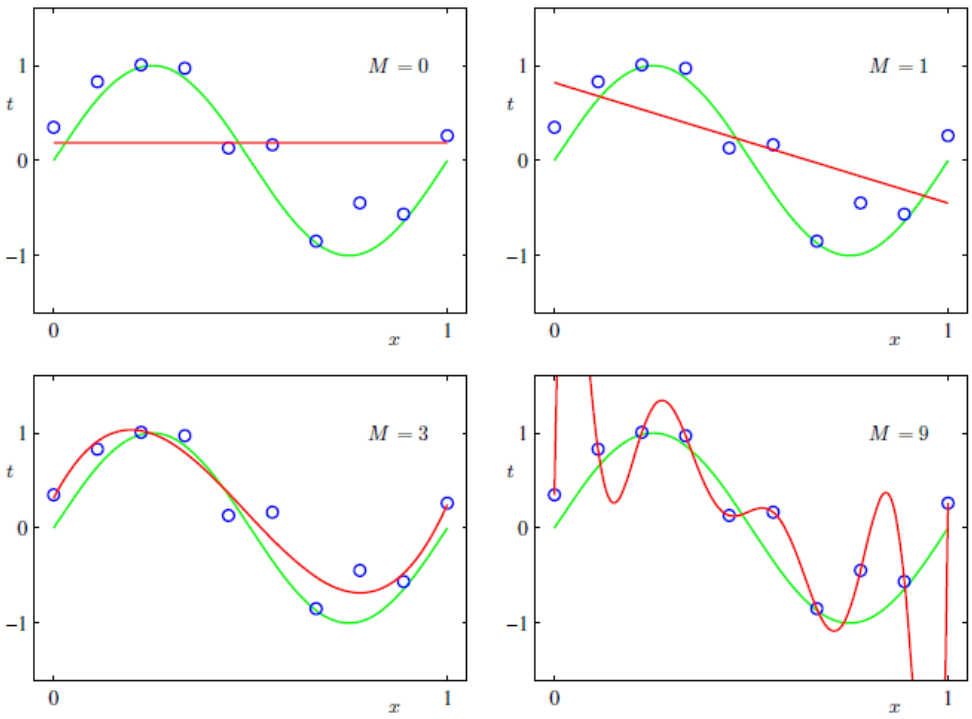

FIGURE. 16 Plot of a polynomial fits having various orders *M*, shown in red, fitted to a simple sine function in green, with gene-sites shown as blue circles.

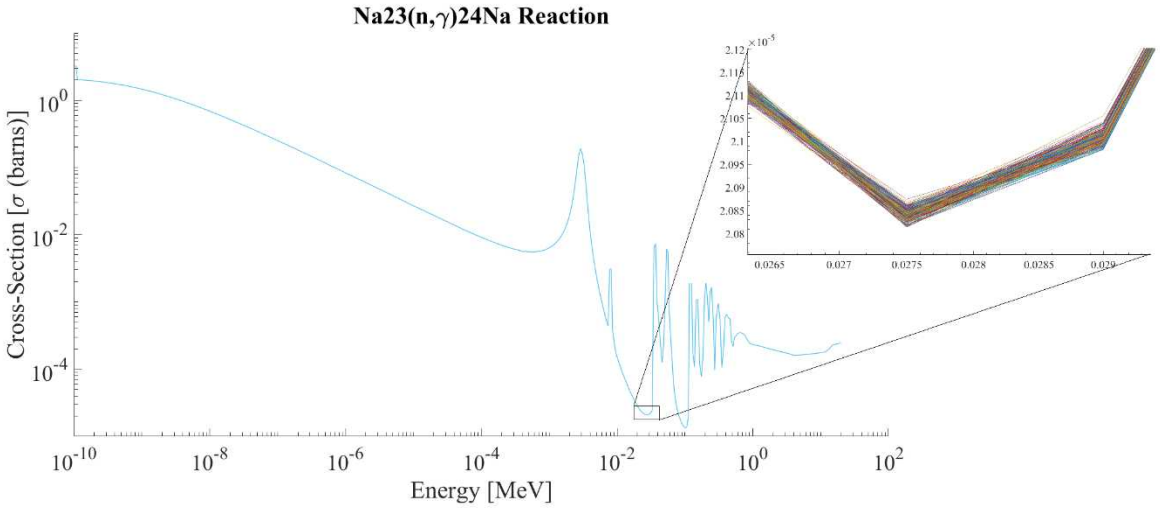

FIGURE. 17 Realizations of the IRDFF Na23(n,γ)24Na reaction cross-section

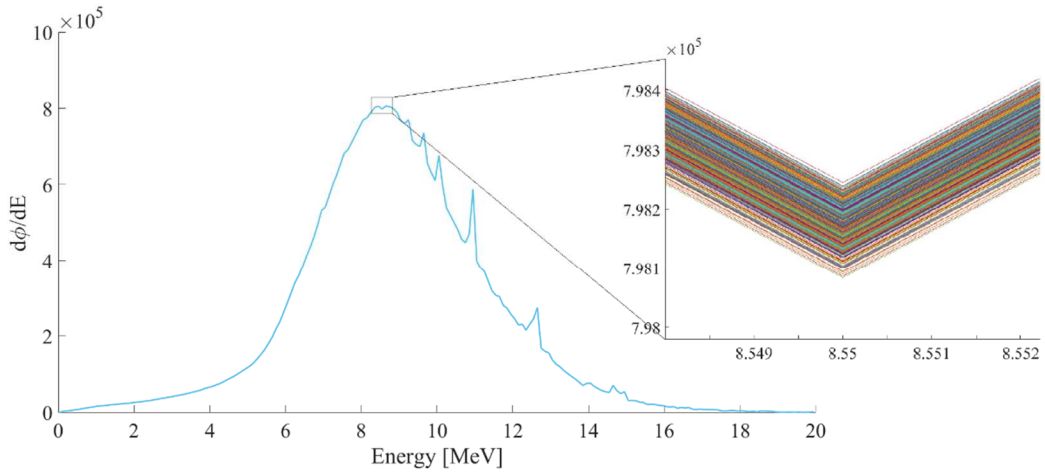

FIGURE. 18 Realizations of the differential neutron energy fluence.

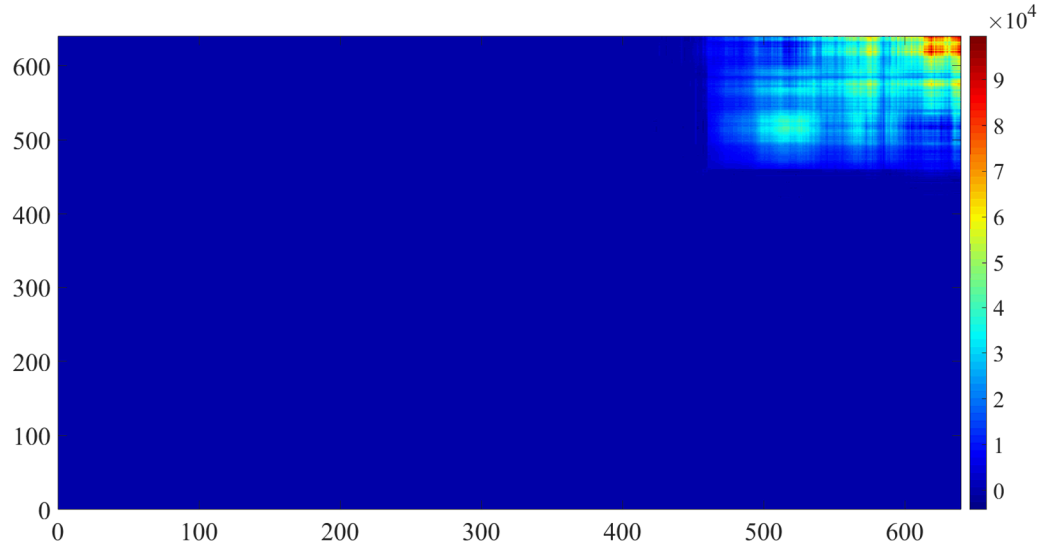

FIGURE. 19 The produced GenSpec covariance surface plot.

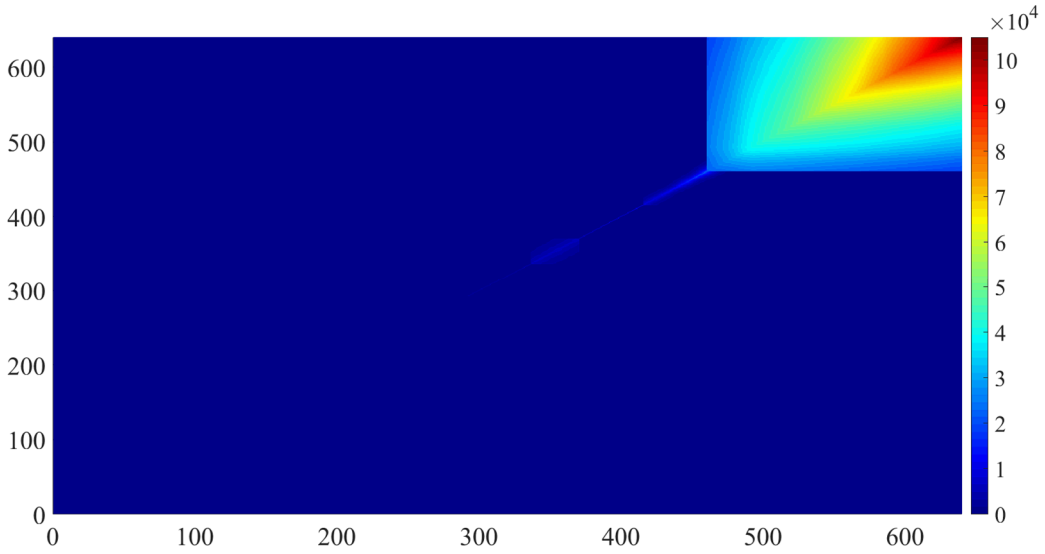

FIGURE. 20 The MCNP trial covariance surface plot.

TABLE 1

Base Case Values of the Genetic Algorithm Modifiers

Modifier Parameter	Value
Population	200
Number of Generations	600
Polynomial Order	10
Number of Gene-Sites	80
Mutation Rate	0.15

TABLE 2

Parametric Sensitivity Analysis Limits and Number of Cases

Modifier Parameter	Minimum	Maximum	Number of Cases
Population	200	10,000	25
Number of Generations	10	1000	25
Polynomial Order	4	15	12
Number of Gene-Sites	10	1000	19
Mutation Rate	0%	100%	25