A Performance Analysis of the Minimax
Multivariate Quality Control Chart

Ian J. Rehmert

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Industrial and Systems Engineering

Joel A. Nachlas, Chairman

J. William Schmidt

C. Patrick Koelling

September 18, 1997

Blacksburg, Virginia

A Performance Analysis of the Minimax Multivariate
Quality Control Chart

Ian J. Rehmert

(ABSTRACT)

A performance analysis of three different Minimax control charts is performed with respect to their Chi-Square control chart counterparts under several different conditions. A unique control chart must be constructed for each process described by a unique combination of quality characteristic mean vector and associated covariance matrix. The three different charts under consideration differ in the number of quality characteristic variables of concern. In each case, without loss of generality the in-control quality characteristic mean vector is assumed to have zero entries and the associated covariance matrix is assumed to have non-negative entries. The performance of the Chi-Square and Minimax charts are compared under different values of the sample size, the probability of a Type I error, and selected shifts in the quality characteristic mean vector. Minimax and Chi-Square charts that are compared share identical in-control average run lengths (ARL) making the out-of-control ARL the appropriate performance measure. A combined Tausworthe pseudorandom number generator is used to generate the out-of-control mean vectors. Issues regarding multivariate uniform pseudorandom number generation are addressed.

Acknowledgements

I would first like to thank my advisor, Dr. Joel A. Nachlas, for his guidance and patience during the completion of my thesis and over the last several years. Without his influences in the classroom, as an advisor, and on the lacrosse field I would not have chosen the academic path I have taken.

I would especially like to thank my parents, John C. Rehmert and Linda G. Middlestadt, and Ellen Base for their love and support throughout my college career. Not only have they encouraged me to pursue my academic interests, but have also served as role models and have shaped me into the person that I am today. For this, I will be forever grateful.

I would also like to acknowledge my committee members, Dr. C. Patrick Koelling and Dr. J. William Schmidt, for their comments and suggestions. In addition, I thank the faculty and staff of the Industrial and Systems Engineering Department for preparing me to continue with my graduate studies and to succeed in my future endeavors.

Finally, I would like to thank my colleagues Lisa M. Maillart and David M. Franz and my friends Marcel, Ken, Stymee, and Bill for being the best of friends and making college fun and full of surprises.

Table of Contents

List of Tables

List of Figures

Chapter 1 - Introduction

1.1 Control Charts

A commonly used tool in statistical quality control of a continuous production process is the control chart. A control chart commonly requires samples with fixed size be taken at fixed intervals. It is assumed in both univariate and multivariate control charts that each sample is independent of the previous samples. A univariate control chart is used in cases where there is only one quality characteristic of interest and a multivariate control chart is used in cases where it is desirable to monitor several quality characteristics simultaneously. Each item in the sample is inspected and the values of each of its quality characteristics are determined. The information from the sample, contained in the sample statistic, is used to perform an hypothesis test to determine whether the process appears to be in control, defined as operating with the parameters of interest at their desired values and with reasonable variation. If the mean value of one or more of the parameters appears to have deviated from its desired value, or if the variation in one or more of the parameters seems to have increased, the process is considered out-of control. Multivariate charts are not only sensitive to changes in the variance of individual quality characteristic values, but they are also sensitive to changes in the correlation structure between quality characteristics.

A univariate process is in control if the mean and variance of the chart measure have not changed from their desired values. Likewise a multivariate process is in control if the mean vector and correlation structure of the chart variables have not changed. The detection of a change in the mean vector or correlation structure of the chart variables is facilitated through control limits that are set in the attempt to satisfy two conflicting goals. The first goal is to obtain a low probability of a signal when the variation in the quality characteristics is caused by random noise alone. The second goal is to obtain a high probability of a signal when any of the variation can be assigned to a specific cause. The probability that the sample statistic falls outside of the control limits while the

process is in control is the probability of a Type I error, normally referred to as α. The value of α determines the in-control average-run-length (ARL), which is the expected number of samples until a signal occurs. A signal in the in-control case is a false alarm. Therefore it is desirable to have a large in-control ARL. The probability that the value of a sample statistic falls inside the control limits when the process is out of control is the probability of a Type II error, normally referred to as β. The value of β determines the out-of-control ARL. A signal in the out-of-control case should occur quickly. Therefore a very low out-of-control ARL is desirable. The in-control ARL and the out-of-control ARL are the key performance measures of both univariate and multivariate control charts. Consequently these measures can be used to make meaningful comparisons between control charts.

Instead of using a multivariate control chart, it may seem reasonable to maintain a separate univariate control chart for each quality characteristic. However this is inappropriate. Each of the separate univariate charts will have its own characteristic α value. When the charts are aggregated, the overall probability of a Type I error increases significantly, decreasing the in-control ARL. A multivariate chart uses a multivariate distribution in order to set the control limits. This significantly alters the shape of the acceptance region. This is shown graphically for the bivariate case in Figure 1.1.
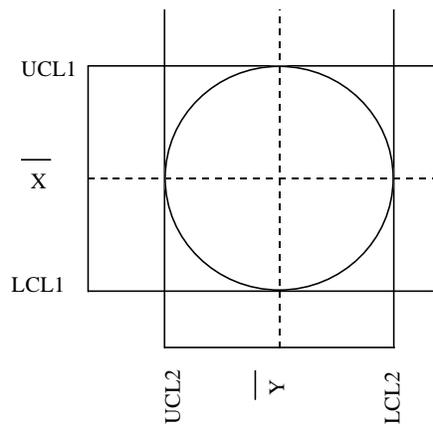
Figure 1.1 – Multivariate Acceptance Regions

2

The square region, common to both univariate charts, is the region of acceptance if two univariate charts are used. The circular region is the region of acceptance for the bivariate chart with independent quality characteristics. The difference in the two areas of acceptance is due to the fact that the probability that both of the characteristics are simultaneously at their extreme values is much lower than the individual probability that either one of them is at its extreme value. In order for the independent univariate charts to have the same probability of Type I error as the bivariate chart, the upper and lower control limits of the univariate charts must be more restrictive. This would cause some of the circular acceptance region of the bivariate chart to be inappropriately considered within the rejection region.

Secondly, independent univariate charts neglect the correlation between the multiple quality characteristics. In the bivariate case, incorporating the correlation between characteristics determines both the degree of distortion of the circular region in Figure 1.1 into an elliptical region and the orientation of the ellipse. The distortion of the acceptance region in the multivariate case causes multiple univariate charts not only to fail to reject in instances where rejection is appropriate, but also to fail to accept in instances where acceptance is appropriate.

The advantage of using a univariate control chart is that it is easily applied and interpreted. A univariate control chart simply requires the elementary manipulation of scalar values to arrive at the sample statistic which is then plotted and compared to the control limits. The interpretation of a signal is straight-forward since the source of the problem obviously arises from a shift in the mean or variance of the quality characteristic being monitored.

Multivariate control charts have the advantage of being able to monitor multiple quality characteristics simultaneously for both changes in the mean value and the correlation structure while maintaining a lower probability of Type I error. Multivariate control charts are at a disadvantage because it is difficult to identify which subset of the

quality characteristics are responsible for a signal since any single characteristic or combination of characteristics could have experienced a shift in mean value, variance, or correlation. This makes it hard to gain insight from a signal in a multivariate chart that would lead to the determination of the source of change in the process behavior. In addition, multivariate charts often require complex matrix manipulation that becomes impractical in higher dimensions without a computerized system. Commonly used multivariate control charts are the χ^2 chart, for a known covariance matrix, and Hotelling's T^2 chart, for an unknown covariance matrix.

1.2 The Minimax Control Chart

After some limited testing, a new control chart developed by Sepulveda (1996), the Minimax control chart, has shown promise in possessing the same desirable characteristics as other multivariate control charts while lacking some of the undesirable characteristics mentioned above. The Minimax chart operates by controlling the minimum and maximum standardized sample means. The correlation between the quality characteristic variables, which is assumed to be known, is incorporated into the construction of the control limits for the minimum and maximum standardized sample mean. In addition, the chart operates at a relatively low and fixed probability of Type I error, α.

There are several advantages of using the Minimax control chart. Although the construction of the chart is fairly complex, the application of the chart is extremely straight-forward. The computations involve the standardization of each of the mean quality characteristic values. The maximum and minimum of these values are then plotted, each against its upper and lower control limits. After limited testing, the Minimax control chart was found to outperform the χ^2 chart in about half of the cases tested.

4

1.3 Problem Description

As mentioned previously, the Minimax control chart has been tested in a limited manner. The purpose of this research is to perform a more thorough investigation into the behavior of the Minimax control chart. An experiment is designed to efficiently test the behavior of the chart over the entire experimental space. A few carefully selected Minimax charts are constructed with identical ARLs, equivalently the identical probability of Type I error. For each Minimax chart, the corresponding χ^2 chart is constructed with the same in-control ARL. The χ^2 chart is an appropriate choice as a benchmark since it is a well-established multivariate control chart based on the identical assumption of a known covariance matrix. Once the charts have been constructed, a verified and validated multivariate uniform random variate generator is used to generate random out-of-control conditions for each of the charts. The out-of-control ARL is evaluated for each of the Minimax and χ^2 charts for each of the out-of-control conditions generated. Since the in-control ARL's for all of the charts are identical, it is valid to make performance evaluations based solely on the out-of-control ARL. Therefore, the control chart with the smaller ARL under each condition is the better control chart under those circumstances. The results of the experiment should lead to conclusions on the strengths and weaknesses of the Minimax control chart as well as the conditions under which the Minimax chart outperforms the χ^2 chart.

Chapter 2 - Literature Review

2.1 Univariate Control Charts

The control chart was first introduced by Dr. Walter Shewhart (1931). Shewhart charts are designed to monitor processes for shifts in the mean or variance of a single quality characteristic. The $\overline{X}$ chart is sensitive to shifts in both the mean and variance. However, the sensitivity of the $\overline{X}$ chart to shifts in the variance is often considered inadequate. For this reason, it is common to use an $\overline{X}$ chart coupled with either an R chart or an s chart, both of which are specifically intended to identify changes in the variance of the quality characteristic. Traditional Shewhart charts only use the current sample statistics to determine if the process has experienced a shift in the mean. The logical extension of the $\overline{X}$ chart is to not only use the current sample statistic, but to incorporate the information that can be gained from the previous sample statistics in some fashion. Page (1955) was one of the first to suggest the use of a separate set of control limits, called warning lines, that lie inside of the ordinary control limits. It was proposed that if two consecutive points fell outside of the warning lines it would be sufficient cause for a signal. This additional signal criterion is called a runs rule. Various runs rules have since been proposed and evaluated, such as Champ and Woodall (1987).

Page's introduction of warning lines allowed for the incorporation of the information from some of the previous sample statistics into the $\overline{X}$ chart. However, Page (1954) proposed the use of the cumulative sum control chart, known as the CUSUM chart, which takes advantage of all of the previous sample statistics. A chart with similar characteristics is the exponentially weighted moving average chart, known as the EWMA chart, first developed by Roberts (1959). Lucas and Saccucci (1990) came to the conclusion that the CUSUM and EWMA charts possess extremely similar performance characteristics. Both the CUSUM chart and the EWMA chart have the ability to detect subtle changes in the value of the process mean and avoid placing too much emphasis on outliers. One of the consequences of combining historical information with current

sample information is the reduction in sensitivity to large abrupt shifts in the mean. Lucas (1973) demonstrated that the Shewhart charts are more responsive to these large shifts in the mean than the EWMA chart. Due to the similar properties of the CUSUM and EWMA charts, it follows that the same is true of the CUSUM chart.

Traditional control charts require that samples are taken at fixed sampling intervals. Reynolds, Amin, Arnold, and Nachlas (1988) propose that variable sampling intervals (VSI) be used with the $\bar{X}$ chart. The time until the next sample is taken with the VSI $\bar{X}$ chart is directly determined by the location of the previous sample mean with respect to the control limits. If the sample mean is relatively close to the control limits, then the time until the next sample is taken is short compared to the situation in which the sample mean is relatively far from either of the control limits. This strategy serves to reduce the average time until signal in cases where the process mean has shifted and reduces the number of samples taken in cases where the process is behaving as expected. Therefore, the VSI $\bar{X}$ chart is shown to be more efficient than the fixed sampling interval $\bar{X}$ chart.

2.2 Multivariate Control Charts

The first development of a multivariate control chart was performed by Hotelling (1947). Hotelling's chart uses Hotelling's T^2 statistic to monitor several variables simultaneously with a specified value of α, the probability of Type I error. Hotelling's T^2 statistic is used in cases where the covariance matrix of the correlated quality characteristic variables is assumed to be unknown. In the case where the covariance matrix is assumed to be known the χ^2 statistic is used. Ghare and Torgerson (1968) developed a bivariate control chart based on the χ^2 statistic. This control chart uses a graphical implementation with an elliptical in-control region in the two-dimensional xy plane. The (x,y) coordinates that correspond to the appropriate value of the χ^2 percentile are plotted for each sample. A more general case of this control chart is the chart commonly referred to as the χ^2 control chart. This differs from Ghare and Torgerson's

control chart in two ways. First, the χ^2 control chart is not limited to the bivariate case. Second, the implementation involves plotting the χ^2 statistic over time and comparing the statistic to the determined critical value. This loses the separation of quality characteristic values that the bivariate method possesses.

As in the univariate $\overline{X}$ chart, the χ^2 chart and Hotelling's T^2 chart only use the information gained from the current sample to determine whether or not to signal. Multivariate versions of the EWMA and CUSUM control charts, MEWMA and MCUSUM respectively, have been proposed which use the information gained from all of the samples. The MCUSUM chart, Woodall and Ncube (1985), has been worked with extensively. Crosier (1988) and Pignatiello and Runger (1990) have developed MCUSUM charts with the ability to detect small shifts in the mean vector more quickly than Hotelling's T^2 chart without sensitivity to the nature of the shift. Lowry and Montgomery (1995) states that the MEWMA chart, developed by Lowry, Woodall, Champ, and Rigdon (1992), has performance characteristics that are at least comparable to the MCUSUM charts proposed by Crosier and Pignatiello and Runger. However, the MEWMA and MCUSUM suffer from the same deficiency as their univariate counterparts. Small shifts in the mean are detected at a faster rate than the χ^2 chart or Hotelling's T^2 chart, but there may be a delayed reaction to a sudden large shift in the mean.

Jackson and Mudholkar (1979) propose the transformation of the correlated quality characteristic variables into a set of independent variables. This is known as the principal component method. The principal components are monitored by comparing a Q statistic, which is the sums of squares of the residuals, to the appropriate critical value. A common criticism of the principal component method is that it does not allow for a meaningful interpretation of a signal in most cases, Hawkins (1991) and Lowry and Montgomery (1995). This is due to the fact that the transformation of the original variables causes information to be lost. Consequently, the means to produce an accurate

translation of a shift in the principal components to a shift in the real system is not generally available.

A method developed by Hawkins (1991) uses the vector of scaled residuals from the regression of each variable on all others. This method is able to separate shifts in the mean from shifts in the covariance structure better than most available multivariate control charts. This makes signal identification a less involved task. Hawkins points out that this method is suited mainly for cases in which only one or two of the quality variables may shift out of control at any time. In addition, it is suggested that cascade-type processes, where a shift at a certain process step will only affect the quality at subsequent process steps, may be effectively monitored using Hawkins' method.

Hayter and Tsui (1994) and Timm (1996) use the Finite Intersection Tests (FIT) method to approach the multivariate quality control problem. Hayter and Tsui's chart is intended to have the ability to diagnose the variables responsible for a process shift while maintaining a desired value of α. The maximum of the absolute difference between the measured value and the mean value, scaled by its associated standard deviation, of each of the quality characteristics is compared to a critical value,

$$M = \max_i \frac{\left| x_i - \mu_i^o \right|}{\sigma_i} > C_{R,\alpha} .$$

(2.1)

The critical value, $C_{R,\alpha}$, is derived using multivariate normal probabilities and the desired value of α. A signal occurs when the critical value is exceeded. The ability to correctly diagnose the responsible variables is shown to be inconsistent by Sepulveda (1996).

The method proposed by Timm is similar except the critical value is derived using the multivariate t distribution with infinite degrees of freedom. If the statistic exceeds the critical value,

$$\max_i \left| T_i \right| = \max_i \frac{\left| x_i - \mu_{0i} \right|}{\sigma_{ii}^{1/2}} > T_{p,\alpha}^2 ,$$

(2.2)

the chart produces a signal.

9

The Minimax control chart developed by Sepulveda (1996) and discussed in Sepulveda and Nachlas (1997) is similar to the charts proposed by Hater and Tsui and Timm. The Minimax control chart also uses the minimum and maximum standardized sample means to make the decision if the process should be considered in control or out of control. However, the Minimax chart uses both lower and upper control limits on both the maximum and minimum standardized sample means. This is facilitated by the development of the capability to determine the value of the joint density function of the maximum and minimum standardized sample means. This not only facilitates a method for setting the control limits, but also allows for the comparison of the performance of the Minimax chart relative to other charts through computation of the out-of-control average run length.

2.3 Pseudorandom Number Generation

Pseudorandom number generators (Law and Kelton 1991) have driven computer simulations and experiments for decades. During this time many different generation techniques have been proposed, many statistical tests for generators have been developed, and many commonly used generators have been proven to possess undesirable characteristics.

Pseudorandom number generators use a deterministic formula to generate a sequence of numbers, U_0, U_1, … that fall in the interval [0,1). The goal is to produce numbers that appear to be randomly chosen from the U(0,1) distribution. The pseudorandom numbers generated are commonly translated into random variates using techniques that transform an observation from a U(0,1) distribution into its counterpart in a more complex distribution. A pseudorandom number generator that possesses undesirable characteristics will result in faulty random variate generation which may lead to misleading or false conclusions. Therefore, it is crucial that the pseudorandom numbers produced by a generator are not distinguishable from true random numbers.

Several statistical and empirical tests exist to identify poor pseudorandom number generators.

A linear congruential generator (LCG) is one of the first types of pseudorandom number generators to be widely implemented and is still common today.   These generators use a deterministic formula to produce an integer, Z_i, from the previous integer, Z_{i-1}.   Each integer produced in the sequence falls in the interval $[0,m)$ and is mapped to the interval $[0,1)$ through division by what is called the modulus of the generator. More specifically, the formula is

$$Z_i = (aZ_{i-1} + c) \bmod m, \tag{2.3}$$

where a is the multiplier, c is the increment, and m is the modulus. A LCG with a value of $c>0$ is a mixed LCG and a LCG with $c=0$ is a multiplicative LCG. The initial value Z_0 is required to start the sequence and is referred to as the seed. The choice of the values for the multiplier, increment, and the modulus determine the properties and behavior of the sequence of numbers produced. In fact, once the seed is specified any number in the sequence can immediately be identified due to the deterministic formula used. It must also be realized that a LCG is only capable of producing the numbers $0/m$, $1/m$, …, $(m-1)/m$. For these two reasons, the numbers produced are termed pseudorandom rather than random. The amount of numbers produced without repeating a previous number is the period of the generator. A repeated number causes the entire sequence to repeat due to the recursive nature of formula 2.3. If certain guidelines are used, the choice of a, c, and m will guarantee a generator with full period, a period of length m. A multiplicative LCG differs in the fact that the maximum period attainable is only of length $m/4$.   The reduction in computation by the time required to add the value of c at each iteration makes the multiplicative LCG attractive in many instances. The LCG has fallen under a great deal of criticism with regard to its statistical properties, specifically the performance exhibited under tests of multidimensional lattice structure (Marsaglia 1968). However, the LCG is still widely used due to its ease of implementation, portability, and ability to generate several independent streams simultaneously.

Several new methods of pseudorandom number generation exist due to an effort to produce numbers with improved statistical properties and longer periods. A second type of generator developed by Tausworthe (1965) relies on a combination of binary operations to produce a binary sequence. The binary sequence b_1, b_2, … is commonly defined by the equation

$$b_i = (b_{i-r} + b_{i-q}) \bmod 2 \quad \text{for integers } 0 < r < q. \tag{2.4}$$

The attractive feature of this implementation is that the sequence defined above is equivalent to performing the *bitwise exclusive or* operation on the two binary digits b_{i-r} and b_{i-q} to obtain the next binary digit in the sequence. The *bitwise exclusive or* operation is much faster than the mathematical computation required in the LCG equation. In order to generate the pseudorandom numbers in the interval [0,1), ℓ of the bits must be chosen from the sequence and then divided by 2^ℓ. The specification of the first q bits of the sequence is necessary to start computation and acts as the seed. Tausworthe generators possess the capability to produce pseudorandom numbers with periods that are several magnitudes greater than any LCG. Although Tausworthe generators generally perform better than LCGs, there has been debate concerning the adequacy of their statistical performance in multiple dimensions (Niederreiter 1992). However, Johnson (1987) recommends Tausworthe generators for multi-dimensional pseudorandom number generation specifically for their statistical properties.

The previous two classes of generators used linear functions of the previous number generated to produce the next number in sequence. Eichenauer, Grothe, and Lehn proposed the first non-linear pseudorandom number generator in 1988 (Niederreiter 1992). The general form of a non-linear generator is

$$Z_i = f(Z_{i-1}) \bmod m, \tag{2.5}$$

where $f(Z_{i-1})$ is a non-linear function of the previous integer generated, m is the modulus of the generator, and Z_i is the integer produced. This generator can have a period as large as m with the appropriate choice of the non-linear function, f. The resulting number, Z_i, can be mapped to the interval [0,1) by setting $U_i = Z_i / m$.

A subclass of non-linear generators, known as inversive congruential generators, is receiving particular attention (Niederreiter 1991, Eicheinauer and Grothe 1992, Niederreiter 1992, Niederreiter 1994). The non-linear formula takes the form

$$f(Z_{i-1}) = a\bar{c} + b \quad \text{where} \quad \bar{c} = \begin{cases} c^{-1} \text{ if } c \neq 0 \\ 0 \quad \text{if } c = 0 \end{cases} \tag{2.6}$$

and a and b are constants. Inversive generators guarantee a reasonable distribution of points over the unit hypercube because they possess lattice structures that have limits on the number of points that fall in a given hyperplane (L'Ecuyer 1990). The study of inversive generators will continue due to their desirable theoretical properties.

Combining more than one generator (L'Ecuyer 1990, Tezuka and L'Ecuyer 1991, L'Ecuyer 1994) to produce each pseudorandom number is another technique that has been investigated. Combined linear congruential generators and combined Tausworthe generators are both capable of achieving longer period lengths and better statistical properties than a single generator of either type (L'Ecuyer 1994). The most well known combination of linear congruential generators uses the first generator to generate several psuedorandorandom numbers and then uses a second LCG to choose which of the several psuedorandom numbers falls next in the sequence.

Tezuka and L'Ecuyer (1991) proposed the use of several Tausworthe generators in parallel in order to produce a sequence of pseudorandom numbers. In order to obtain the next bit in the main sequence, each of the j independent generators that are running in parallel must generate the next bit in its sequence. From equation 2.4, the system of equations that generate the j independent bits can be represented as follows:

$$\begin{aligned}
b_{i,1} &= (b_{i-r_1,1} + b_{i-q_1,1}) \bmod 2 && \text{for integers } 0 < r_1 < q_1 \\
b_{i,2} &= (b_{i-r_2,2} + b_{i-q_2,2}) \bmod 2 && \text{for integers } 0 < r_2 < q_2 \\
&\ \vdots \\
b_{i,j} &= (b_{i-r_j,j} + b_{i-q_j,j}) \bmod 2 && \text{for integers } 0 < r_j < q_j
\end{aligned} \tag{2.7}$$

where $b_{i,j}$ is the ith bit of the jth sequence. These are then translated into the next bit in the main sequence by taking the *bitwise exclusive or* of all of the independent bits, $b_{i,1}$, $b_{i,2}$,

… $b_{i,j}$.   An exhaustive search for a combination of two Tausworthe sequences that performed the best under various multi-dimensional statistical tests in two through fifteen dimensions resulted in three candidate generators. These generators also have periods in the neighborhood of 2^{60}.   For this reason, the generator suggested by Tezuka and L'Ecuyer is used to generate the out of control vectors to evaluate the two multivariate quality control charts (See Apendix B). Although this generator is not specifically stated to be a random vector generator, its tested performance in all of the dimensions that are of concern for this thesis make it a perfect candidate.

2.4 Pseudorandom Vector Generation

In order to address many problems that typical generators have in higher dimensions, there have been a few techniques designed explicitly for multi-dimensional random vector generation.

The matrix LCG method is essentially an extension of the linear congruential method into higher dimensions. The following equation for a mixed matrix linear congruential generator results if equation 2.3 is altered by making the Z elements into k-dimensional vectors, the multiplier into a k x k matrix of integers, and the increment into a k-dimensional vector of integer constants

$$\overline{Z}_i \ = \ (\overline{A}\overline{Z}_{i-1} \ + \ \overline{C}) \ \text{mod} \ m. \tag{2.8}$$

The modulus in this case is taken elementwise with the resulting k-dimensional vector. The multiplicative matrix LCG differs from the mixed matrix LCG by eliminating the addition of the increment vector, $\overline{C}$. The matrix generators have fallen under consistent criticism (Niederreiter 1994, Niederreiter 1992, L'Ecuyer 1990) for possessing the same undesirable multi-dimensional properties as the original linear congruential generators.

Niederreiter extended his work with inversive non-linear congruential generators (Niederreiter 1991, Niederreiter 1992) to include investigation into the properties of inversive vector generators (Niederreiter 1994). The criteria for maximum period length

are stated and proved. In addition, the theoretical performance properties of the inversive vector generator under the serial test and test of discrepancy are explored in detail and found to be promising. The use of inversive congruential pseudorandom vector generators is relatively new and requires further investigation.

The Tausworthe generator is used to generate pseudorandom vectors. In order to generate a p-dimensional pseudorandom vector, p pseudorandom numbers are generated and are then placed in sequence as the vector entries. The multidimensional lattice properties discussed earlier make this method appropriate.

Chapter 3 - Problem Statement

3.1 The Minimax Control Chart (Sepulveda 1996)

The Minimax chart monitors p quality characteristics simultaneously. At fixed time intervals, a sample of size n is taken and the value of each of the quality characteristics is determined. Let X_{ij} represent the value of quality characteristic i of item j for i = {1, 2, ..., p} and j = {1, 2, ..., n}, and let $\bar{X}_i = \frac{1}{n}\sum_{j=1}^{n} X_{ij}$. The vector $\bar{X} = \left\{ \bar{X}_1, \bar{X}_2, \ldots, \bar{X}_p \right\}^T$ contains the sample mean value for each quality characteristic. Although the observations may be correlated within each sample, the samples are assumed independent of each other. In addition, it is assumed that $\bar{X}$ is distributed with a multivariate normal distribution with known process mean vector μ = {μ_1, μ_2, ..., μ_p}^{T} and covariance matrix $\Sigma = \sigma_{ij}^2$, where σ_{ii}^2 is the variance of the ith quality characteristic and σ_{ij}^2 is the covariance between the ith quality characteristic and the jth quality characteristic. The covariance structure is assumed to remain stable over time leaving only a shift in the mean vector to cause an out-of-control situation.

The Minimax chart requires the minimum and maximum standardized sample means to fall between their respective control limits. In order to obtain the minimum standardized sample mean $Z_{[1]}$ and the maximum standardized sample mean $Z_{[p]}$, it is necessary to standardize each of the elements of the sample mean vector, $\bar{X}$, in the following manner:

$$Z_i = \frac{\bar{X}_i - \mu_i}{\sigma_{ii}/\sqrt{n}}. \tag{3.1}$$

Note that Z_i is simply the standardized sample mean for the ith quality measure and $Z_{[i]}$ is an order statistic defining the ith smallest value of the p standardized sample means. Therefore $Z_{[1]}$ and $Z_{[p]}$ are determined from $\mathbf{Z}$ = {Z_1, Z_2, ..., Z_p}.

A shift in the characteristic mean vector of the process is represented by the vector Δ, where $\Delta = (\Delta_1, \Delta_2, \ldots, \Delta_p)$ and Δ_i is the shift in units of standard deviation of the mean of the ith quality measure. It follows that:

$$\Delta_i = \frac{(\mu_i{}' - \mu_i)}{\sigma_{ii}}. \tag{3.2}$$

Since the only shift in the process that may occur is a shift in the process mean vector, the process is in control if $\Delta = 0$ and out of control if $\Delta \neq 0$. For simplicity, the in-control mean vector is assumed to have zero entries without causing a loss in generality.

There are four control limits for the Minimax chart: $UCL_{[1]}$ and $LCL_{[1]}$ are the respective upper and lower control limits for $Z_{[1]}$ and $UCL_{[p]}$ and $LCL_{[p]}$ are the respective upper and lower control limits for $Z_{[p]}$. It should be realized that the Minimax chart is essentially two dependent charts. One chart monitors and controls $Z_{[1]}$ and the other chart monitors and controls $Z_{[p]}$. The charts are dependent because the joint distribution of $Z_{[1]}$ and $Z_{[p]}$ is used to construct the control limits.

The probability of a Type I error, α, is the sum of the probabilities of a Type I error in the $Z_{[1]}$ chart, $\alpha_{[1]}$, and a Type I error in the $Z_{[p]}$ chart, $\alpha_{[p]}$. Assuming that positive and negative shifts in the mean are equally important:

$$\alpha_{[1]} = \alpha_{[p]}, \text{ where} \tag{3.3}$$

$$\alpha_{[1]} = \alpha_1 + \alpha_2 \quad \text{and} \quad \alpha_{[p]} = \alpha_3 + \alpha_4. \tag{3.4}$$

The probabilities α_1 and α_2 are the respective lower and upper in-control tail probabilities associated with the $Z_{[1]}$ chart and α_3 and α_4 are the respective upper and lower in-control tail probabilities associated with the $Z_{[p]}$ chart, such that (see figure 3.1):

$$\alpha_1 = P[Z_{[1]} \in (-\infty, LCL_{[1]}) \mid \Delta = 0], \tag{3.5}$$

$$\alpha_2 = P[Z_{[1]} \in (UCL_{[1]}, \infty) \mid \Delta = 0], \tag{3.6}$$

$$\alpha_3 = P[Z_{[p]} \in (-\infty, LCL_{[p]}) \mid \Delta = 0], \text{ and} \tag{3.7}$$

$$\alpha_4 = P[Z_{[p]} \in (UCL_{[p]}, \infty) \mid \Delta = 0]. \tag{3.8}$$
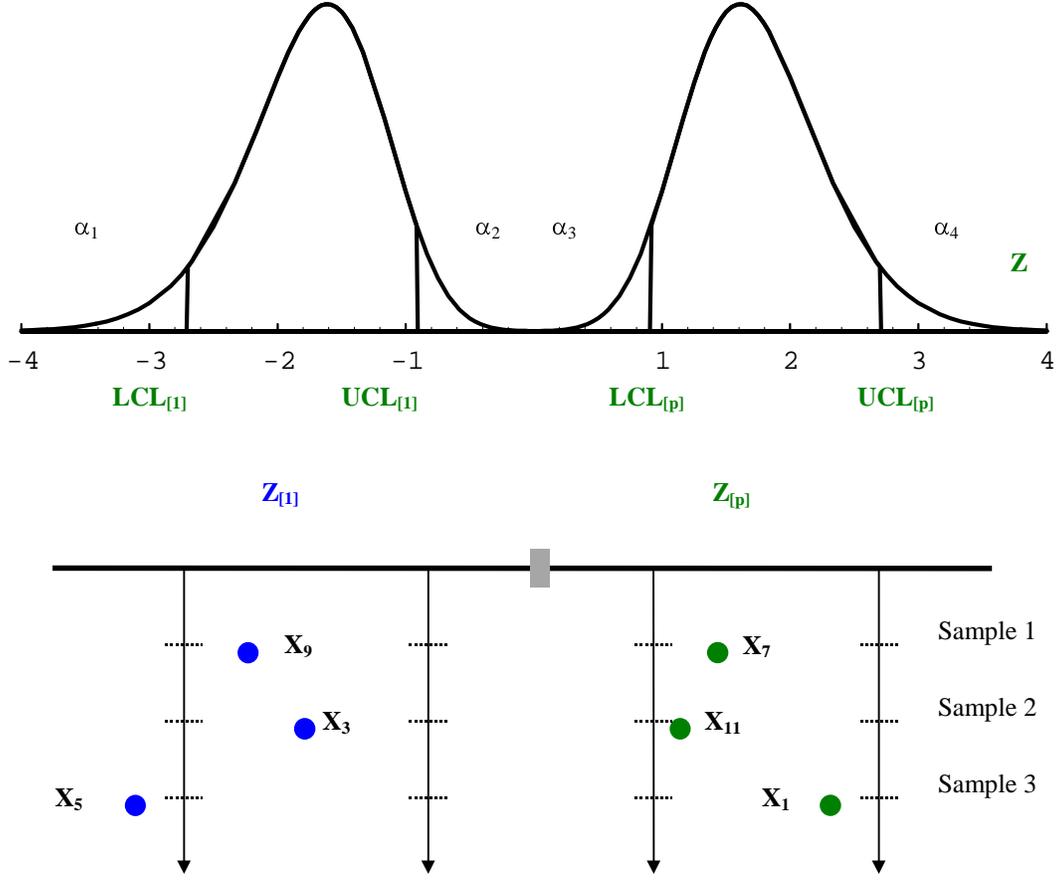
Figure 3.1 - Minimax Tail Probabilities (Sepulveda 1996)

In order to find the values of α, α_1, α_2, α_3, and α_4 it is necessary to divide the sample space into nine mutually exclusive outcomes, which are (see figure 3.2):

$$E_{c,c} = P[Z_{[1]} \in (LCL_{[1]}, UCL_{[1]}) \mid \Delta)] \cap P[Z_{[p]} \in (LCL_{[p]}, UCL_{[p]}) \mid \Delta)] \tag{3.9}$$

$$E_{b,b} = P[Z_{[1]} \in (-\infty, LCL_{[1]}) \mid \Delta)] \cap P[Z_{[p]} \in (-\infty, LCL_{[p]}) \mid \Delta)] \tag{3.10}$$

$$E_{a,a} = P[Z_{[1]} \in (UCL_{[1]}, \infty) \mid \Delta)] \cap P[Z_{[p]} \in (UCL_{[p]}, \infty) \mid \Delta)] \tag{3.11}$$

$$E_{b,a} = P[Z_{[1]} \in (-\infty, LCL_{[1]}) \mid \Delta)] \cap P[Z_{[p]} \in (UCL_{[p]}, \infty) \mid \Delta)] \tag{3.12}$$

$$E_{a,b} = P[Z_{[1]} \in (UCL_{[1]}, \infty) \mid \Delta)] \cap P[Z_{[p]} \in (-\infty, LCL_{[p]}) \mid \Delta)] \tag{3.13}$$

$$E_{b,c} = P[Z_{[1]} \in (-\infty, LCL_{[1]}) \mid \Delta)] \cap P[Z_{[p]} \in (LCL_{[p]}, UCL_{[p]}) \mid \Delta)] \tag{3.14}$$

$$E_{c,b} = P[Z_{[1]} \in (LCL_{[1]}, UCL_{[1]}) \mid \Delta)] \cap P[Z_{[p]} \in (-\infty, LCL_{[p]}) \mid \Delta)] \tag{3.15}$$

$$E_{a,c} = P[Z_{[1]} \in (UCL_{[1]}, \infty) \mid \Delta)] \cap P[Z_{[p]} \in (LCL_{[p]}, UCL_{[p]}) \mid \Delta)] \tag{3.16}$$

$$E_{c,a} = P[Z_{[1]} \in (LCL_{[1]}, UCL_{[1]}) \mid \Delta)] \cap P[Z_{[p]} \in (UCL_{[p]}, \infty) \mid \Delta)] \tag{3.17}$$

18

$E_{b,b}$ $E_{c,b}$ $E_{a,b}$

$E_{b,c}$ $E_{c,c}$ $E_{a,c}$

$E_{b,a}$ $E_{c,a}$ $E_{a,a}$
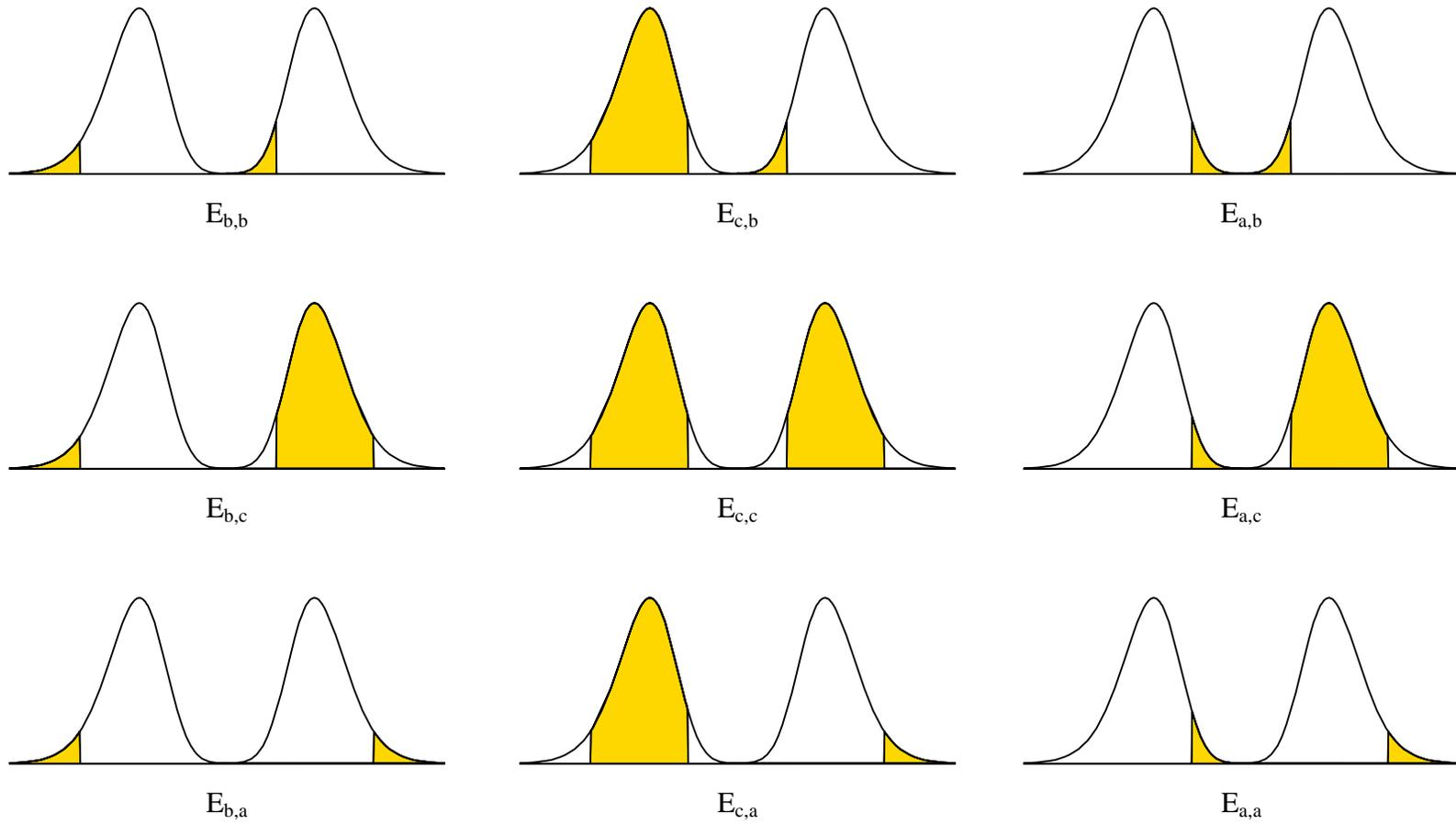
Figure 3.2 - Minimax Control Chart Nine Zone Graph (Sepulveda 1996)

This allows for the calculation of the listed values without the actual joint distribution of $Z_{[1]}$ and $Z_{[p]}$. Instead, calculations based on probability rules, using the nine mutually exclusive outcomes of the sample, provide all of the information necessary. These calculations are derived and explained in Sepulveda (1996) and are outlined in Appendix A. In short, the integral expressions for each of the tail probabilities is used in conjunction with the definition of the above nine mutually exclusive outcomes to form a system of five linearly independent equations with five unknowns. The ability to determine these values makes it possible to perform the necessary comparisons with other multivariate control charts based on the ARL. The same framework that solves for α is used again to find the value for β. This method of calculation makes use of the fact that $E_{c,c}$ is the only event that does not cause a signal. In the case where $\Delta = 0$, the process is in control and no signal occurs. In the case where $\Delta \neq 0$, the process is out of control and no signal occurs. It follows that:

$$P[\text{Type I error}] = \alpha = 1 - P[E_{c,c} \mid \Delta = 0] \text{ and} \tag{3.18}$$

$$P[\text{Type II error}] = \beta = P[E_{c,c} \mid \Delta \neq 0]. \tag{3.19}$$

The values of α and β are used to calculate the in-control and out-of-control average run lengths (ARL) as a measure of chart performance.

3.2 The Chi-Square Control Chart

The χ^2 control chart is used as a performance benchmark for the Minimax control chart. Both charts perform the same hypothesis test for a change in the target mean vector, $\mathbf{\mu}$, assuming no change in the known covariance matrix, $\mathbf{\Sigma}$. The test statistic Z^2 is compared to the value of $\chi^2_{\alpha,p}$, where

$$Z^2 = n\left(\overline{X} - \mu\right)^T \Sigma^{-1}\left(\overline{X} - \mu\right), \tag{3.20}$$

α is the probability of Type I error for the χ^2 control chart, n is the sample size, and p is the number of quality characteristics. If $Z^2 < \chi^2_{\alpha,p}$ the null hypothesis that there has been no change in $\mathbf{\mu}$ is accepted. Otherwise, the null hypothesis is rejected, indicating that there has been a change in $\mathbf{\mu}$.

20

If the process experiences a shift in the mean the test statistic Z^2 is distributed with a Non-Central Chi-Squared distribution with p degrees of freedom and non-centrality parameter $n\lambda^2(\Delta)$. Therefore:

$$\beta = P[(Z^2 \sim \text{Non-Central } \chi^2) < \chi^2_{\alpha,p} \mid \Delta]. \tag{3.21}$$

The value of the non-centrality parameter from above is:

$$\lambda^2(\Delta) = \left(\overline{X} - \mu\right)^T \Sigma^{-1}\left(\overline{X} - \mu\right) = \Delta^T \Sigma^{-1}\Delta, \tag{3.22}$$

the squared distance from the in-control mean vector. It follows that the value

$$\lambda(\Delta) = \sqrt{\Delta^T \Sigma^{-1}\Delta} \tag{3.23}$$

is an indication of the magnitude of the shift experienced in the mean vector.

3.3 Chart Comparison

For each experimental condition, both charts possess the same probability of Type I error, giving them an identical in-control ARL. If the Minimax and the Chi-Square charts have the same in-control ARL they will operate identically until a shift in the mean vector occurs. The expression for the in-control ARL is

$$\text{ARL}_{\text{in-control}} = 1/P[\text{Type I error}] = 1/\alpha, \tag{3.24}$$

and the expression for the out-of-control ARL is

$$\text{ARL}_{\text{out-of-control}} = 1/(1-P[\text{Type II error}) = 1/(1-\beta). \tag{3.25}$$

Since both charts possess the same in-control ARL, the out-of-control ARL is the appropriate measure of relative chart effectiveness. The out-of-control ARL directly determines the average number of samples taken to signal after a shift in the mean vector occurs. Therefore, it is desirable to have the lower out-of-control ARL. It is necessary to have identical in-control ARLs so the relationship between the in-control ARL and the out-of-control ARL is not a factor.

For each of the dimensions of the quality characteristic variable considered an experimental framework is defined in Section 3.5.

3.4 Multivariate Uniform Random Vector Generation

A combined Tausworthe pseudorandom number generator (Tezuka and L'Ecuyer 1991) produces the values of the quality characteristic mean vector that are used to evaluate the out-of-control ARL for the Minimax chart and the χ^2 chart. In addition, it produces the values $\ell_1, \ell_2, \ldots, \ell_8$, discussed in the next section, in order to develop the covariance matrices used. This particular generator combines the information from two Tausworthe sequences to produce a single pseudorandom number stream. As a result, the numbers generated exhibit better statistical behavior and posses a much longer period than single Tausworthe generators or linear congruential generators. Specifically, several sequences generated using this generator passed a battery of statistical tests in two to fifteen dimensions and have a period length on the order of 2^{60}. The demonstrated multi-dimensional statistical properties make it an appropriate tool to generate the out-of-control vectors that will test the performance of the two control charts of interest.

3.5 Experimental Methodology

An experiment consists of the evaluation of the out-of-control ARL for both the Minimax and χ^2 charts for fixed values of the following chart parameters:

- p, the number of quality characteristics
- n, the sample size
- α, the probability of Type I error
- and S, the specific type of shift in the mean vector.

Each experiment requires the evaluation of the ARL eleven times for each of the two control charts. The first evaluation gives the in-control ARL for the fixed values of p, n, α, and S for both charts. Since the in-control ARL is evaluated, the value of the shift vector has zero entries, $\Delta = 0$. The next ten evaluations use out-of-control vectors, $\Delta \neq 0$, generated by the pseudorandom vector generator in order to produce the out-of-control ARL of the two charts, providing a means of chart comparison. Each value of S determines a unique combination of the number of variables that experience a shift and

the vector position each of those variables. Evaluation of the in-control and out-of-control average run lengths for both the Minimax and Chi-Square charts is done using Mathematica (See Appendix C and D).

The parameters p, n, α, and S are varied with each experiment over a relevant range within a structure that facilitates the meaningful interpretation of the resulting data. The values considered for p are 2, 5, and 8, which is representative of the range of the number of quality characteristics monitored by multivariate quality control charts in practice. Covariance matrices, Σ, must be chosen for each of the values of p to represent the relationship between the individual quality variables of interest. In order to simplify the computational intensity of calculating cumulative multivariate normal probabilities, it is necessary to assume a special covariance structure (Tong 1990). The entries of the

covariance matrix, $\Sigma = \begin{pmatrix} 1 & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & 1 & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & 1 \end{pmatrix}$, of the quality characteristic vector is

determined in the following manner:

$$\sigma_{ij} = \begin{cases} l_i l_j & \text{for } i \neq j \\ 1 & \text{for } i = j \end{cases}, \quad \text{where } l_i \in [0,1]. \tag{3.26}$$

The value of ℓ_i is associated with the ith, out of p, quality characteristic variables and is positive because only positive entries for the covariance matrix will be considered for this thesis. This special structure eliminates the need to evaluate a p-dimensional integral and allows the substitution of a single integral. The value $\sigma_{ii} = 1$ is chosen as the variance of the quality characteristic mean values without loss of generality for simplicity.

The Tausworthe pseudorandom number generator is used to generate the values for $\ell_1, \ell_2, \ldots, \ell_8$. The covariance matrix for p = 2 is determined using ℓ_1 and ℓ_2, for p = 5 $\ell_1, \ell_2, \ldots, \ell_5$ are used, and for p = 8, $\ell_1, \ell_2, \ldots, \ell_8$ are used. The values used for each case are in ascending order as follows:

- $p = 2$ $\ell = \{0.410, 0.614\}$,

- $p = 5$ $\ell = \{0.286, 0.410, 0.460, 0.614, 0.698\}$,

- $p = 8$ $\ell = \{0.286, 0.410, 0.460, 0.614, 0.698, 0.795, 0.841, 0.954\}$.

Using the formula described above in equation 3.25, the following covariance matrices result:

- $p = 2$ $\Sigma = \begin{bmatrix} 1.00 & 0.25 \\ 0.25 & 1.00 \end{bmatrix}$,

- $p = 5$ $\Sigma = \begin{bmatrix} 1.00 & 0.12 & 0.13 & 0.18 & 0.20 \\ 0.12 & 1.00 & 0.19 & 0.25 & 0.29 \\ 0.13 & 0.19 & 1.00 & 0.28 & 032 \\ 0.18 & 0.25 & 0.28 & 1.00 & 0.43 \\ 0.20 & 0.29 & 0.32 & 0.43 & 1.00 \end{bmatrix}$, and

- $p = 8$ $\Sigma = \begin{bmatrix} 1.00 & 0.12 & 0.13 & 0.18 & 0.20 & 0.23 & 0.24 & 0.27 \\ 0.12 & 1.00 & 0.19 & 0.25 & 0.29 & 0.33 & 0.34 & 0.39 \\ 0.13 & 0.19 & 1.00 & 0.28 & 0.32 & 0.37 & 0.39 & 0.44 \\ 0.18 & 0.25 & 0.28 & 1.00 & 0.43 & 0.49 & 0.52 & 0.59 \\ 0.20 & 0.29 & 0.32 & 0.43 & 1.00 & 0.55 & 0.59 & 0.67 \\ 0.23 & 0.33 & 0.37 & 0.49 & 0.55 & 1.00 & 0.67 & 0.76 \\ 0.24 & 0.34 & 0.39 & 0.52 & 0.59 & 0.67 & 1.00 & 0.80 \\ 0.27 & 0.39 & 0.44 & 0.59 & 0.67 & 0.76 & 0.80 & 1.00 \end{bmatrix}$.

For each value of p and its associated covariance matrix, the charts are compared over all combinations of 3 values of n, 3 values of α, and several different shift conditions. In order to compare the charts under a range of conditions that are representative of those likely to be encountered in practice, the values considered for the sample size are n = 3, 5, and 7 and the values considered for the probability of Type I error are $\alpha = 0.005, 0.0025$, and 0.002.

In the case of p = 2 there are only two types of shifts that may be considered, a shift in one quality variable and a shift in both quality variables. The quality variable that shifts is irrelevant in the two-dimensional case because the covariance structure is symmetric. If the variable that experiences a shift is represented with a Δ, where

$\Delta \in (0, 3]$, and a variable that remains the same is represented with a 0, then the possible combinations in the two-dimensional case can be represented graphically:

- S = 1: $\quad \Delta = (\Delta, 0)$
- S = 2: $\quad \Delta = (\Delta, \Delta)$

It should be noted that the value of Δ is the shift in units of standard deviation of a particular quality characteristic mean value. This is due to the simplifying assumptions of an in-control mean vector with zero entries and covariance matrices with $\sigma_{ii} = 1$.

In the cases in which $p = 5$ and $p = 8$ there are 31 and 717 different shifts, respectively, to consider for each combination of n and α. In order to study the experimental space in an efficient manner a subset of the possible shift conditions are studied. The shifts in the mean vector are chosen to cover shifts in the lower correlated quality variables, shifts in the intermediately correlated variables, shifts in the highly correlated variables, and shifts evenly dispersed among the quality variables. The shifts considered in the five-dimensional case are enumerated below:

- S = 1: $\quad \Delta = (\Delta, 0, 0, 0, 0)$
- S = 5: $\quad \Delta = (\Delta, \Delta, \Delta, 0, 0)$
- S = 2: $\quad \Delta = (0, 0, \Delta, 0, 0)$
- S = 6: $\quad \Delta = (0, 0, \Delta, \Delta, \Delta)$
- S = 3: $\quad \Delta = (0, 0, 0, 0, \Delta)$
- S = 7: $\quad \Delta = (0, \Delta, \Delta, \Delta, 0)$
- S = 4: $\quad \Delta = (\Delta, 0, \Delta, 0, \Delta)$
- S = 8: $\quad \Delta = (\Delta, \Delta, \Delta, \Delta, \Delta)$

Likewise, the shifts considered in the eight-dimensional case are:

- S = 1: $\quad \Delta = (\Delta, 0, 0, 0, 0, 0, 0, 0)$
- S = 5: $\quad \Delta = (0, 0, 0, 0, \Delta, \Delta, \Delta, \Delta)$
- S = 2: $\quad \Delta = (0, 0, 0, 0, 0, 0, 0, \Delta)$
- S = 6: $\quad \Delta = (0, 0, \Delta, \Delta, \Delta, \Delta, 0, 0)$
- S = 3: $\quad \Delta = (0, 0, 0, \Delta, 0, 0, 0, 0)$
- S = 7: $\quad \Delta = (\Delta, 0, \Delta, 0, \Delta, 0, \Delta, 0)$
- S = 4: $\quad \Delta = (\Delta, \Delta, \Delta, \Delta, 0, 0, 0, 0)$
- S = 8: $\quad \Delta = (\Delta, \Delta, \Delta, \Delta, \Delta, \Delta, \Delta, \Delta)$

In summary, there are 162 experimental conditions consisting of one set of in-control ARL values and ten sets of out-of-control ARL values. This represents all combinations of

- p = 2, 5, 8
- n = 3, 5, 7
- α = 0.005, 0.0025, 0.002
- for p = 2: S = 1, 2

- for p = 5: S = 1, 2, 3, 4, 5, 6, 7, 8

- for p = 8: S = 1, 2, 3, 4, 5, 6, 7, 8

The values generated for Δ are initially generated uniformly in the interval (0, 3]. After looking at the results of the ARL computations it became apparent that allowing each variable to shift up to 3 standard deviations from the mean provided for less meaningful conclusions. Therefore, it was necessary to scale the values generated for Δ to fall uniformly in the interval (0, 1] and re-evaluate all of the experimental conditions described above. Chapter 4 addresses the results and conclusions derived from the described calculations.

Chapter 4 – Results and Conclusions

After examining the average run lengths resulting from the calculations and experimental conditions explained in Chapter 3, there are several observations that can be made about the performance of the Minimax chart relative to the Chi-Square chart.

4.1 Two Quality Variables (p=2)

In the two dimensional case, p=2, it is obvious that shifts occurring in only one quality variable are detected slightly faster with the Chi-Square chart than with the Minimax chart (Figure 4.1).

ARL vs. Lambda (p=2, n=3, alpha=1/200, S=1)

Lambda	0	0.1	0.3	0.47	0.48	0.48	0.51	0.6	0.72	0.82	0.87
ARL Min	200.1617	189.1776	128.0124	75.23598	75.10053	73.72804	67.41435	49.90011	33.86037	24.60860	21.21313
ARL Chi	200.0037	185.5192	113.1205	60.62777	60.50401	59.25304	53.57107	38.43452	25.38597	18.21962	15.65613
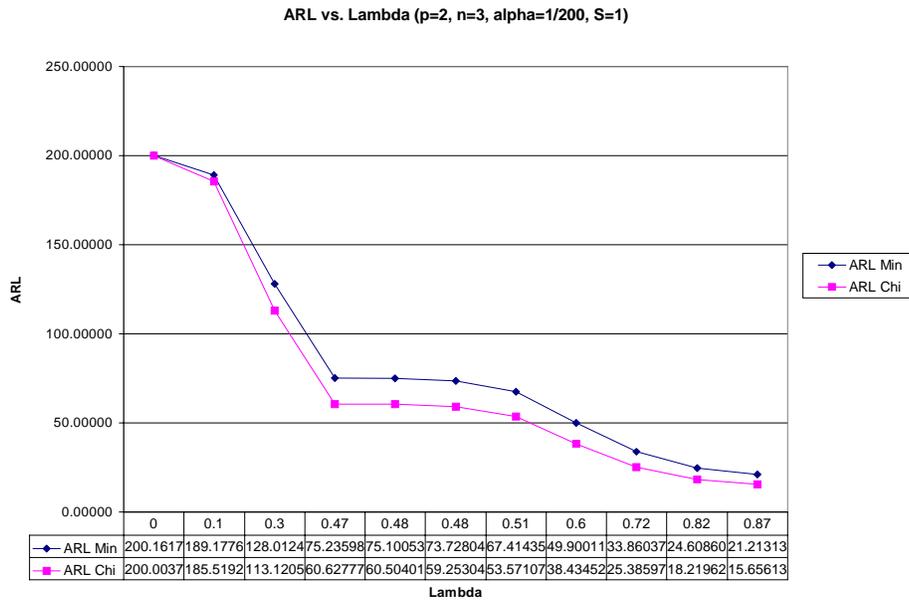
Figure 4.1 – ARL vs. Lambda (p=2, n=3, alpha=1/200, S=1)

As shown in the above graph, the ARL for the Minimax chart is larger than the Chi-Square chart for all values of λ. This is the case for all of the experiments where only one quality variable experiences a shift in the mean, S=1 (Appendix E.1 and E.2).

In the case where there is a shift in both quality variables, S=2, the Minimax chart did not consistently perform better or worse than the Chi-Square chart. When both of the

27

quality variables experiences a similar shift in magnitude, the Minimax chart has a lower out-of-control ARL than the Chi-Square chart. However, when one of the quality variables experiences a shift of a much greater magnitude than the other quality variable, the Chi-Square chart has a lower out-of-control ARL (Table 4.1).
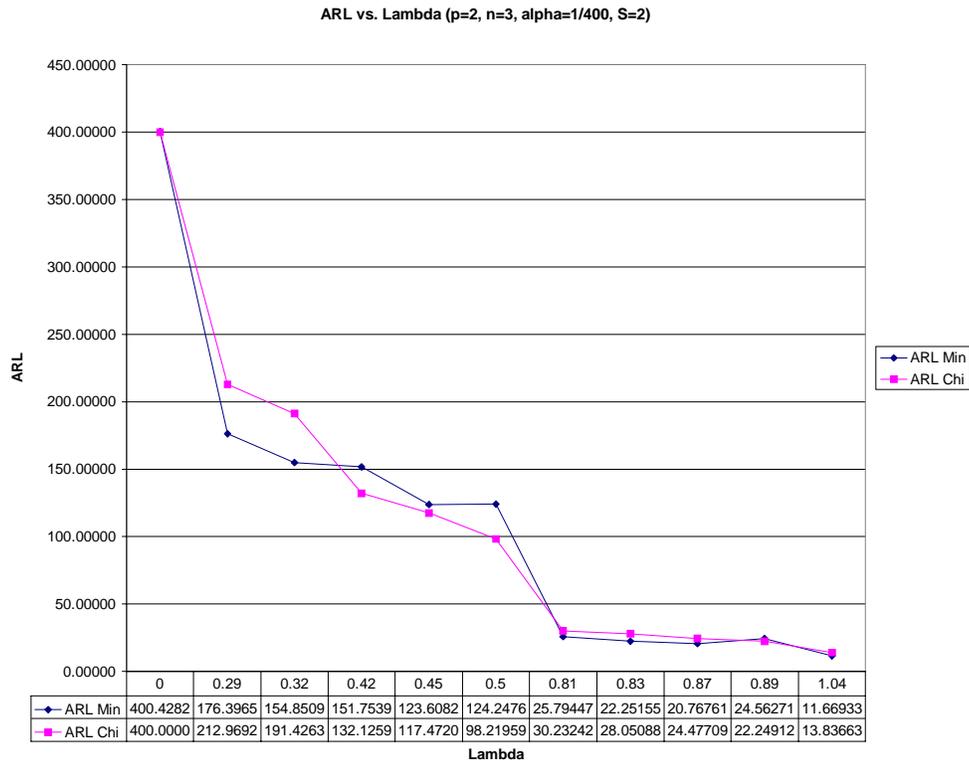
Figure 4.2 – ARL vs. Lambda (p=2, n=3, alpha =1/400, S=2)

Lambda	Delta 1	Delta 2	ARL Min	ARL Chi
0	0	0	400.42822	400.00002
0.29382	0.25016	0.211758	176.39659	212.96927
0.323246	0.248181	0.262577	154.85097	191.42637
0.420541	0.0304946	0.413739	151.75399	132.12594
0.450595	0.449528	0.0823745	123.60822	117.47209
0.496142	0.480565	0.0007132	124.24762	98.21959
0.806585	0.379199	0.784084	25.79447	30.23242
0.827644	0.770424	0.485396	22.25155	28.05088
0.86656	0.838294	0.422125	20.76761	24.47709
0.894306	0.166659	0.892404	24.56271	22.24912
1.039552	0.552334	0.990797	11.66933	13.83663

Table 4.1 – Data for Figure 4.2

This assertion is supported by the data contained in Table 4.1 which corresponds to Figure 4.2. In the case where $\lambda=0.496$, the first quality variable experiences a shift of 0.481 standard deviations and the second quality variable experiences a shift of 0.007 standard deviations. The magnitudes of these two shifts differ drastically. Therefore the ARL of the Minimax chart, 124, is higher than the ARL of the Chi-Square chart, 98. In contrast, when $\lambda=0.323$ the first quality variable shifts by 0.248 standard deviations and the second quality variable shifts by 0.263 standard deviations, the ARL of the Minimax chart, 155, is less than that of the Chi-Square chart, 191. The two quality variables in this case did not shift in a drastically different manner. Additional examples of this behavior can be found in Appendix E.3-E.5.

As expected, both charts generally experience a decrease in ARL with an increase in the value of λ. It should be noted that at the extreme values of λ, both charts behave relatively the same as they approach an ARL of one. This means that the Minimax and Chi-Square charts perform in an equivalent manner for sufficiently large shifts in the mean vector. It can be seen that when the value of λ reaches the neighborhood of 2.0, the ARL of both charts is in the neighborhood of one and both ARLs converge to one as the value of λ increases (Figure 4.3).

ARL vs. Lambda (p=2, n=5, alpha=1/500, S=1)

	0	0.57	1.24	1.82	1.96	2.1	2.37	2.7	2.79	3.07	3.09
ARL Min	499.9979	61.69958	4.57780	1.47416	1.28418	1.16409	1.05078	1.00929	1.00509	1.00080	1.00064
ARL Chi	499.9998	45.38031	3.59882	1.33210	1.19293	1.10681	1.02962	1.00454	1.00233	1.00030	1.00024
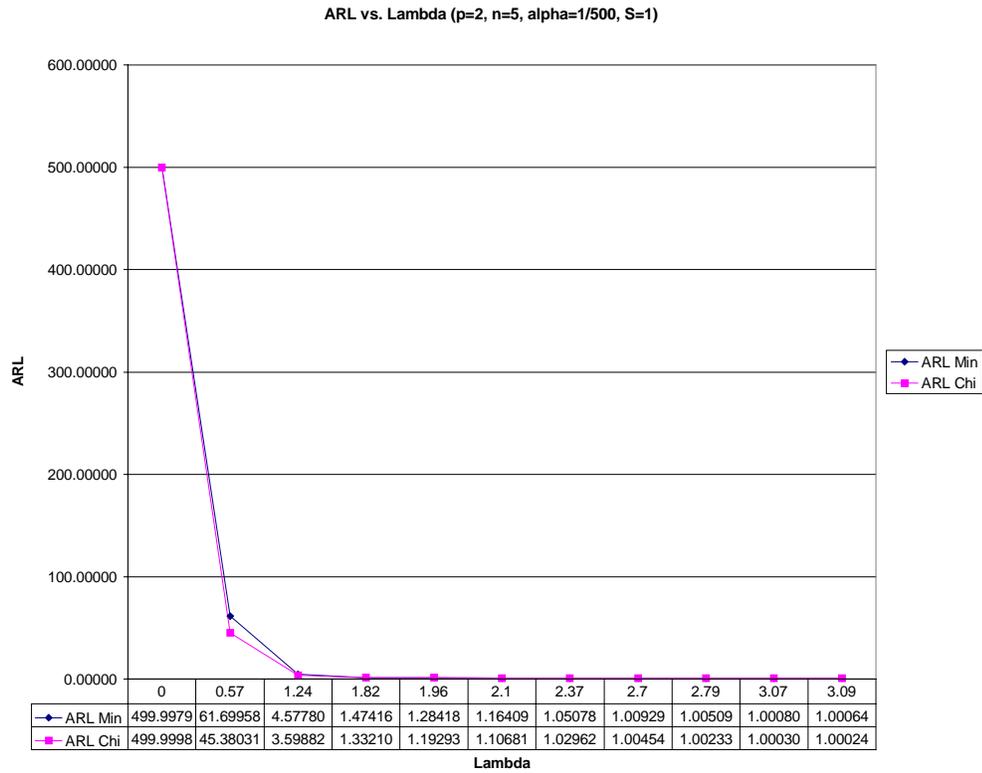
Lambda

Figure 4.3 – ARL vs. Lambda (p=2, n=5, alpha=1/500, S=1)

Although the ARL of the Chi-Square chart consistently decreases as λ increases, this does not always hold true for the Minimax chart. This is attributable to the behavior discussed earlier where the Minimax chart performs differently depending on the relative shift of the two quality variables. Figure 4.4 contains an example of a situation where the ARL changes from 35 to 44 as the value of λ barely increases from 0.724 to 0.727. It is counterintuitive that a shift of larger magnitude could result in an increase in the ARL. However, in the situation where the ARL=35 the magnitude of the shift in both variables is similar, and in the situation where the ARL=44 the magnitude of the shift in the two variables is quite different. Although the value of λ is almost identical, the relative magnitude of the shifts dictates different behavior than might otherwise be expected (also see Appendix E.6).
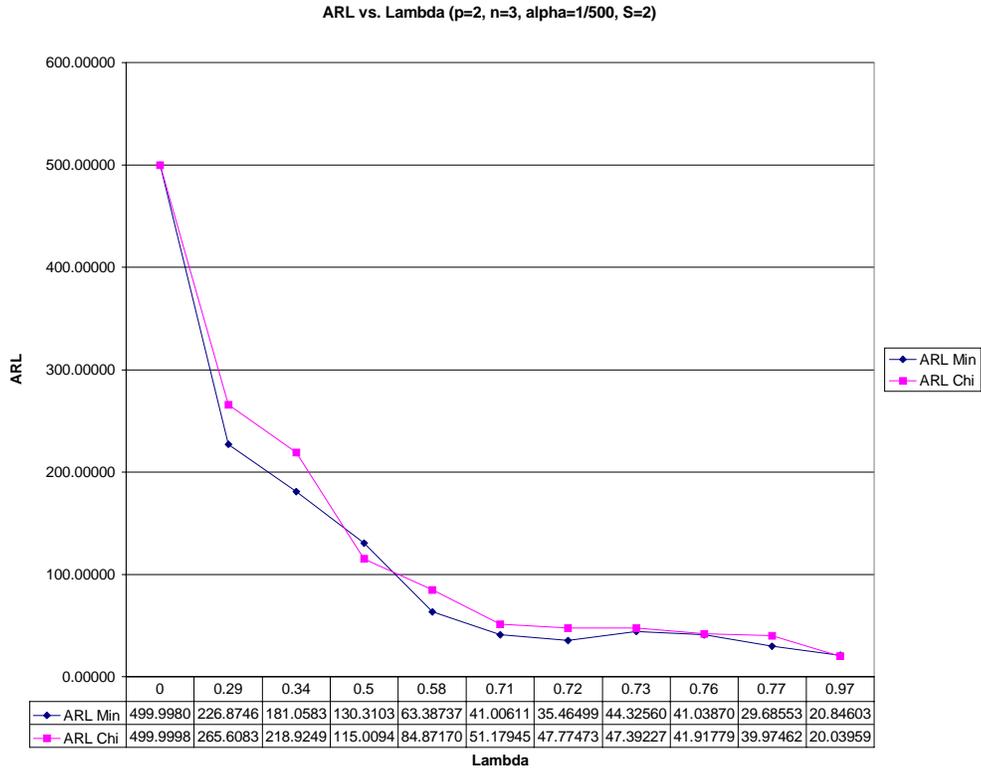
Figure 4.4 – ARL vs. Lambda (p=2, n=3, alpha=1/500, S=2)

Lambda	Delta 1	Delta 2	ARL Min	ARL Chi
0	0	0	499.99800	499.99984
0.28881	0.26986	0.167089	226.87469	265.60834
0.340345	0.20237	0.315547	181.05830	218.92490
0.50198	0.0569204	0.497135	130.31036	115.00947
0.57757	0.406728	0.498733	63.38737	84.87170
0.706456	0.669638	0.38535	41.00611	51.17945
0.724452	0.621852	0.51532	35.46499	47.77473
0.726561	0.723138	0.248995	44.32560	47.39227
0.759016	0.758159	0.22446	41.03870	41.91779
0.771691	0.658187	0.554615	29.68553	39.97462
0.965842	0.965842	0.241631	20.84603	20.03959

Table 4.2 – Data for Figure 4.4

The charts designed to monitor five and eight quality variables behave in a manner similar to that of the charts designed to monitor two quality variables.

4.2 Five Quality Variables (p=5)

For the two-dimensional experiments, the Minimax chart has a larger out-of-control ARL in the cases where there is a large difference in the magnitude of the shift between the two quality variables. Upon examination of the data, it is apparent that the same type of behavior exists in the five-dimensional experiments. There are two situations in which the Minimax chart performs poorly relative to the Chi-Square chart. The first situation is when at least one of the quality variables does not experience a shift in its mean value (Figure 4.5).

ARL vs. Lambda (p=5, n=5, alpha=1/500, S=7)

Lambda	0	0.47	0.5	0.54	0.66	0.87	0.88	0.97	1	1.18	1.3
ARL Min	499.9238	168.8768	150.7755	150.6328	96.25672	47.44148	37.09363	27.81902	18.45722	14.90428	11.82362
ARL Chi	499.9999	132.8910	118.2255	98.04161	58.12033	23.49697	22.52841	15.97954	14.21587	7.61403	5.20278
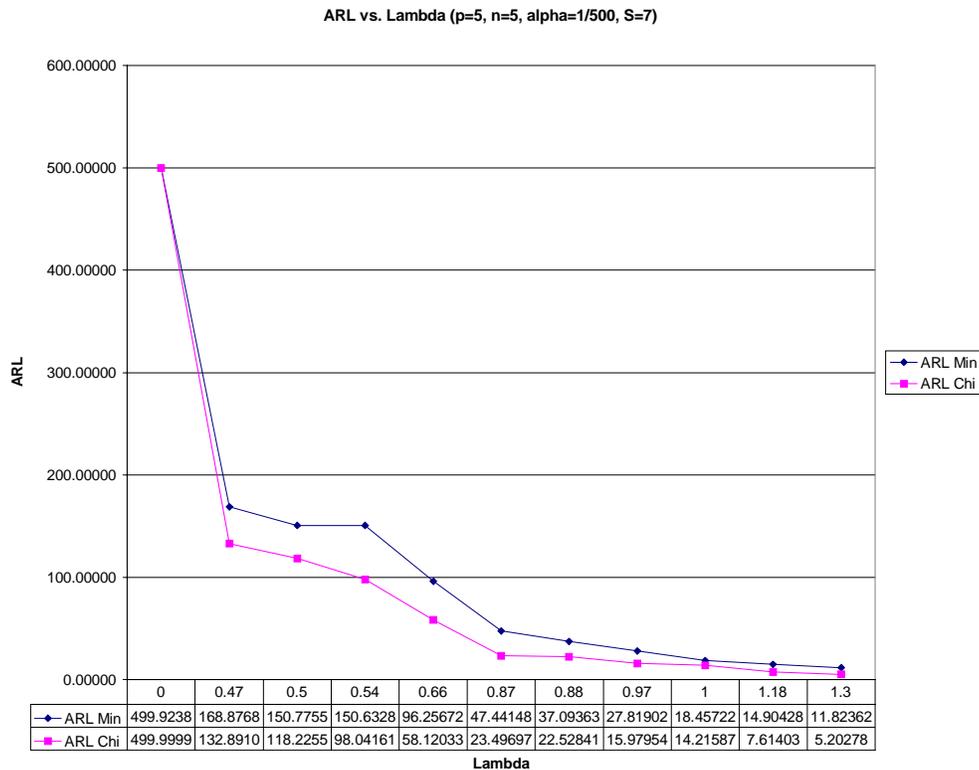
Figure 4.5 – ARL vs. Lambda (p=5, n=5, alpha=1/500, S=7)

In Figure 4.5 three of the five quality variables experience a shift in the mean. The Minimax chart consistently performs worse than the Chi-Square chart for all values of λ. In fact, this behavior holds true for of all of the experiments in which at least one of the quality variables does not experience a shift in the mean, specifically for all experiments with shifts S=1-7 (Appendix E.7-E.10).

The second case in which the Minimax chart does not perform as well as the Chi-Square chart is when all of the quality variables experience a shift, but the magnitude of the shift in at least one of the quality variables is negligible when compared to the shifts in the rest of the quality variables (Figure 4.6 and Table 4.3). This situation is only possible in the experiments associated with the remaining shift condition, S=8.

ARL vs. Lambda (p=5, n=3, alpha=1/400, S=8)

	0	0.58	0.72	0.75	0.79	1.05	1.08	1.14	1.15	1.17	1.53
ARL Min	399.8140	129.0277	90.72167	56.23760	34.05813	15.44255	23.72454	19.40717	19.64745	11.85980	12.39373
ARL Chi	400.0012	122.3705	76.20977	68.66532	60.79189	26.29424	23.64035	19.66223	18.99414	17.90403	6.68891
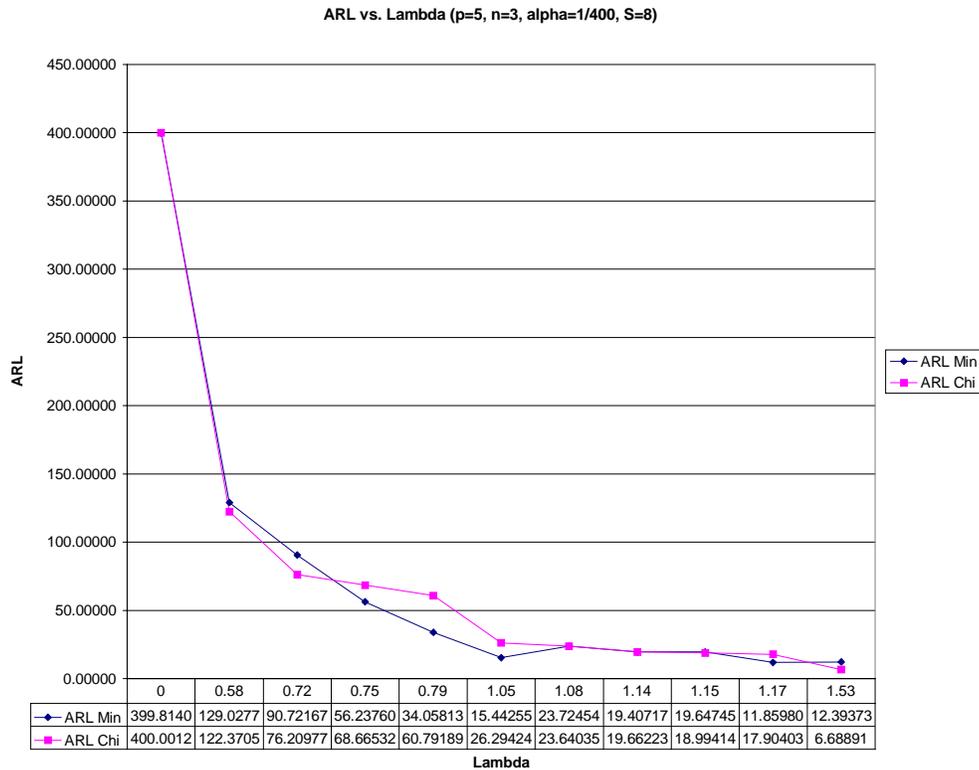
Figure 4.6 – ARL vs. Lambda (p=5, n=3, alpha=1/400, S=8)

Lambda	Delta 1	Delta 2	Delta 3	Delta 4	Delta 5	ARL Min	ARL Chi
0	0	0	0	0	0	399.81406	400.00121
0.576509	0.02416	0.307118	0.51975	0.132739	0.104389	129.02776	122.37051
0.720378	0.067857	0.013123	0.483645	0.20223	0.610189	90.72167	76.20977
0.751667	0.582813	0.490091	0.13542	0.193315	0.408273	56.23760	68.66532
0.788225	0.394072	0.333111	0.550976	0.451957	0.636208	34.05813	60.79189
1.045595	0.445387	0.600967	0.681907	0.534497	0.868009	15.44255	26.29424
1.079634	0.973368	0.257024	0.404746	0.49206	0.141486	23.72454	23.64035
1.139676	0.279212	0.671207	0.734151	0.228464	0.913948	19.40717	19.66223
1.151111	0.010844	0.891688	0.830018	0.440304	0.614211	19.64745	18.99414
1.170798	0.389175	0.72128	0.914443	0.785115	0.585168	11.85980	17.90403
1.533612	0.991506	0.86534	0.382997	0.913761	0.021546	12.39373	6.68891

Table 4.3 – Data for Figure 4.6

For example, when λ=0.577 the first quality variable shifts by 0.024 standard deviations while the rest of the quality variables shift by a much more substantial amount. Therefore, the ARL of the Minimax chart, 129, is greater than the ARL of the Chi-Square chart, 122. On the other hand, the Minimax chart is more effective in detecting shifts of similar magnitude across all variables. For example, when λ=0.788 all of the quality variables experience a moderate shift. Therefore, the ARL of the Minimax chart, 34, is less than the ARL of the Chi-Square chart, 61. More illustrative examples can be found in Appendix E.11 and E.12.

In the experiments where a shift takes place in one variable, S=1,2, and 3, it becomes apparent that the performance of the Minimax chart is influenced by the identity of the variable to experience the shift in addition to the magnitude of the shift. If the variable that experiences the shift is a highly correlated variable, S=3, then the ARL will be larger than if the variable has a lower correlation and experiences a shift of the same number of standard deviations, S=1 (See Figure 4.7 and Appendix E.13-E.14).

ARL vs. Lambda (p=5, n=3, alpha=1/200, S=1 and S=3)
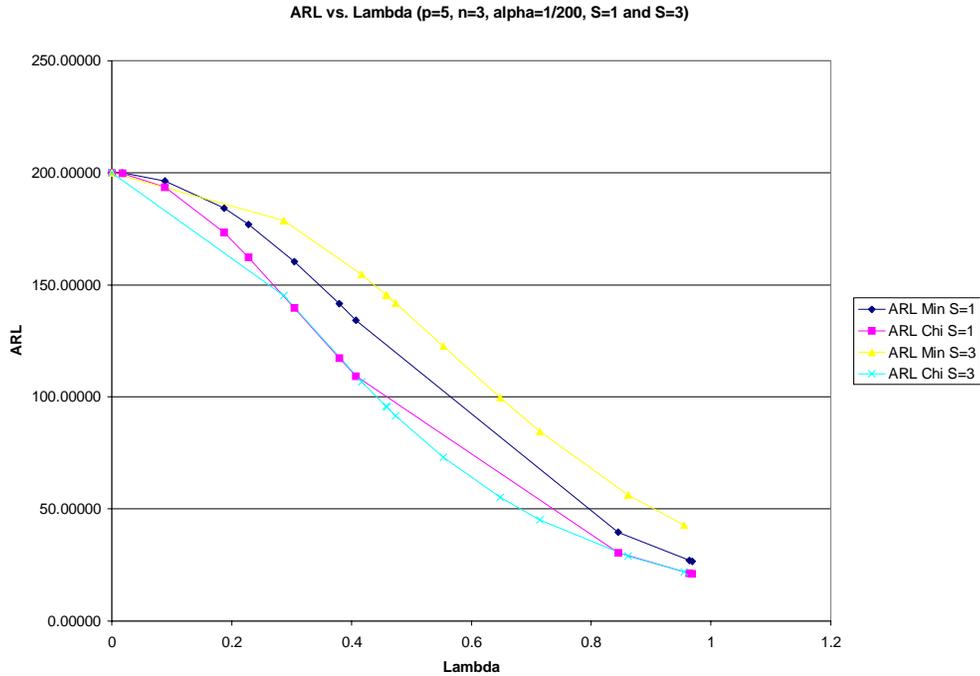
Figure 4.7 – The Relative Sensitivity to the Identity of the Shift Variable (S=1 & S=3)

In Figure 4.7 there is a noticeably larger ARL for the shift in a highly correlated variable, S=3, with the Minimax chart. At the same time, the Chi-Square chart does not seem to be as sensitive to the identity of the shift variable. Therefore, a shift in a highly correlated variable makes the relative performance of the Minimax chart worse than if the shift experienced were in a lower correlated variable. When three variables experience a shift, S=4, 5, 6, and 7, the behavior described above is not nearly as distinct (Figure 4.8 and Appendix E.15).

ARL vs. Lambda (p=5, n=3, alpha=1/200, S=5 and S=6)
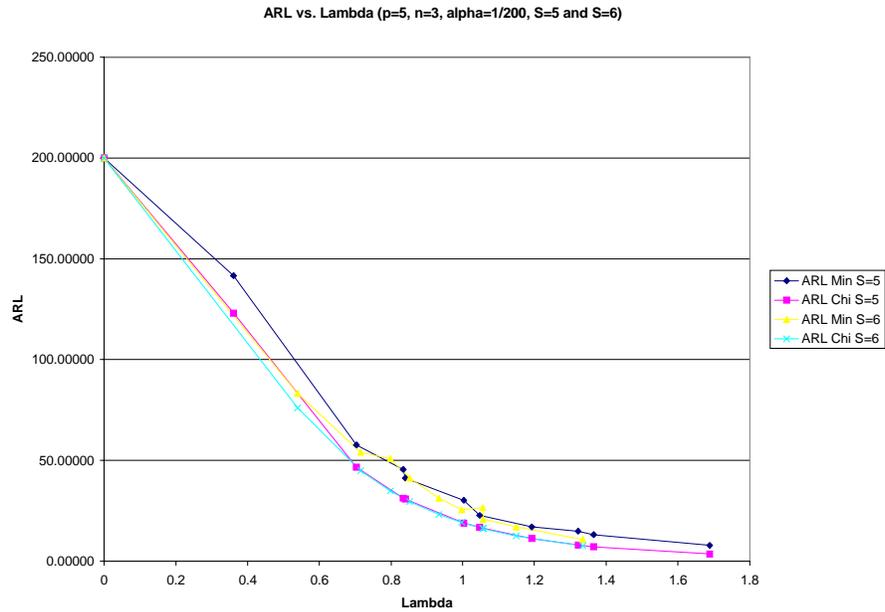
Figure 4.8 – The Relative Sensitivity to the Identity of the Shift Variable (S=5 & S=6)

This is most likely due to the complex relationship of the multiple influences on the ARL mentioned above. When only one variable experiences a shift, the effect of the identity of the quality variable can be isolated.

Both of the charts are almost equally as effective in detecting shifts of sufficiently large magnitude (Figure 4.9). The charts approach and converge to ARL=1 at about the same value for λ. In the example below, the ARL values approach one at λ=1.77 and continue to behave in an almost identical manner from that point on. This behavior can also be seen in the examples in Appendix E.16 and E.17.

ARL vs. Lambda (p=5, n=7, alpha=1/200, S=1)

	0	0.39	0.54	0.67	0.81	1.21	1.77	2.03	2.38	2.47	2.5
ARL Min	199.9806	84.38820	42.49187	22.72439	11.58537	2.77422	1.16191	1.04244	1.00414	1.00200	1.00159
ARL Chi	200.0005	63.89929	32.39923	18.24498	9.96197	2.70476	1.16931	1.04597	1.00471	1.00231	1.00184
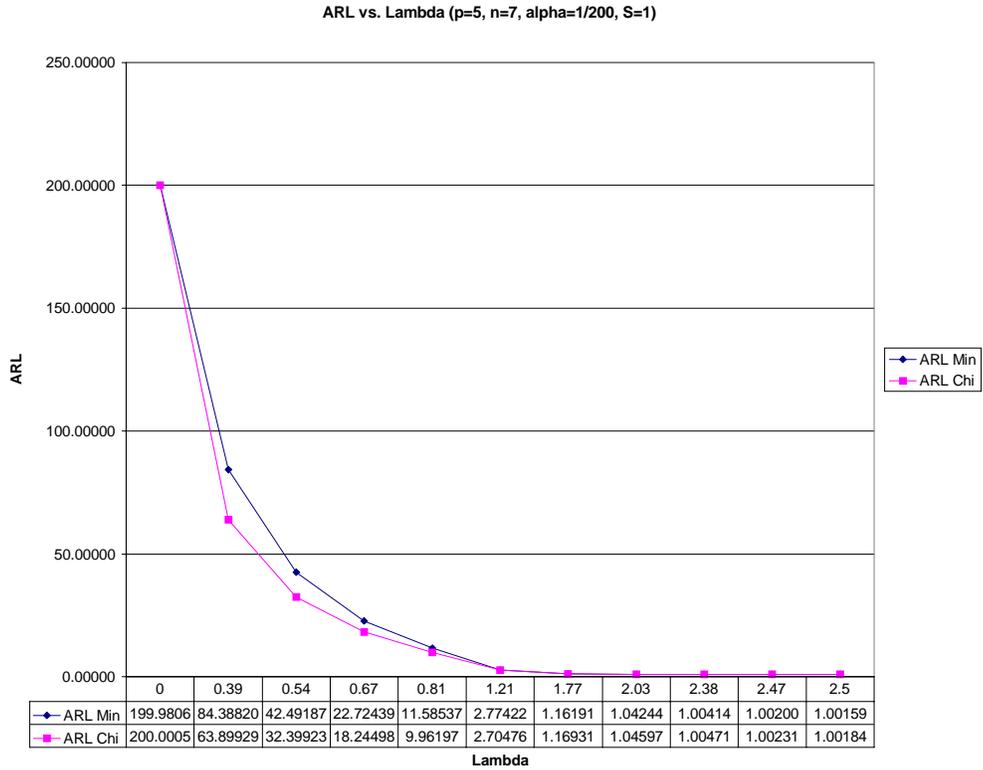
Lambda

Figure 4.9 – ARL vs. Lambda (p=5, n=7, alpha=1/200, S=1)

4.3 Eight Quality Variables (p=8)

The Chi-Square chart outperforms the Minimax chart for all experiments in which there is at least one variable that does not experience a shift in the mean, S=1. Figure 4.10 is an example of the behavior described. Additional examples can be found in Appendix E.18-E.21.

ARL vs. Lambda (p=8, n=7, alpha=1/200, S=7)

Lambda	0	0.63	0.94	1.11	1.41	1.53	1.68	1.69	1.71	1.87	2.23
ARL Min	199.9875	66.60315	21.65671	14.52731	14.15733	4.80320	5.46031	7.14016	5.30493	5.14300	3.01068
ARL Chi	200.0000	29.85217	8.47002	4.80298	2.20146	1.74849	1.42626	1.40392	1.38229	1.19422	1.03148
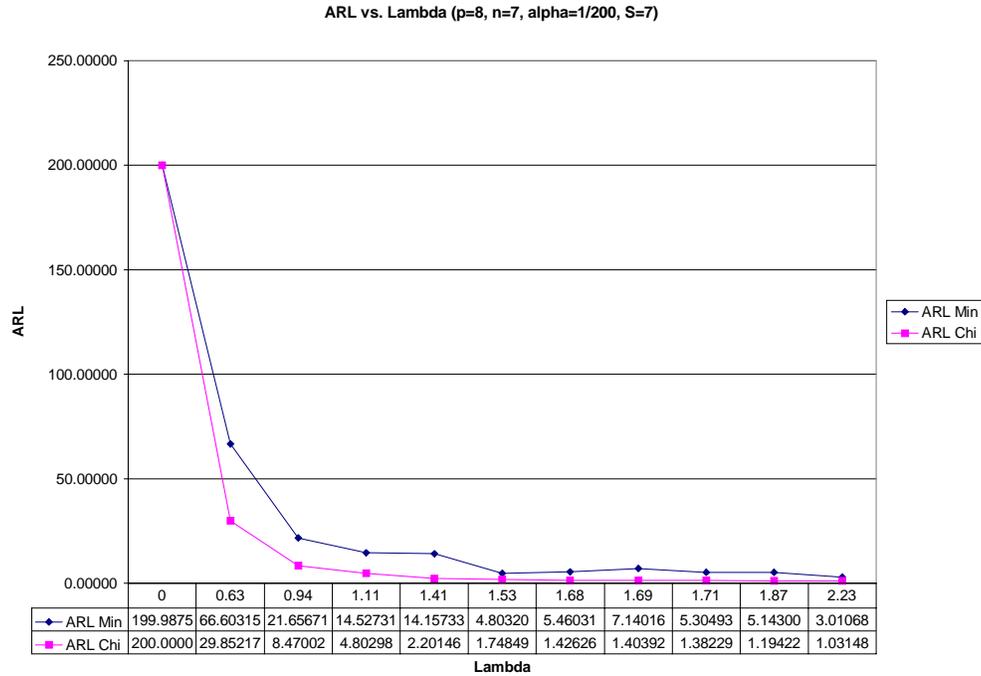
Figure 4.10 – ARL vs. Lambda (p=8, n=7, alpha=1/200, S=7)

As with the charts designed to monitor 2 and 5 variables, when all of the quality variables experience a shift in the mean, neither chart has the lowest out-of-control ARL for all values of λ. When one or more quality variables experience a shift in the mean that is negligible compared to the rest of the quality variables, the Minimax chart does not perform as well as the Chi-Square chart. However, when all of the variables shift by a similar magnitude the Minimax chart performs better than the Chi-Square chart (Figure 4.11 and Appendix E.22-E.23).

ARL vs. Lambda (p=8, n=3, alpha=1/500, S=8)

	0	0.9	0.98	1.17	1.17	1.23	1.29	1.35	1.4	1.65	1.72
◆ ARL Min	499.8350	39.43200	113.6856	26.11148	38.00997	25.46466	54.64782	43.60939	45.95656	23.55801	30.49756
■ ARL Chi	499.9981	71.08501	55.18267	30.42470	29.88712	25.55460	21.29247	17.66752	15.66376	8.07434	6.81724
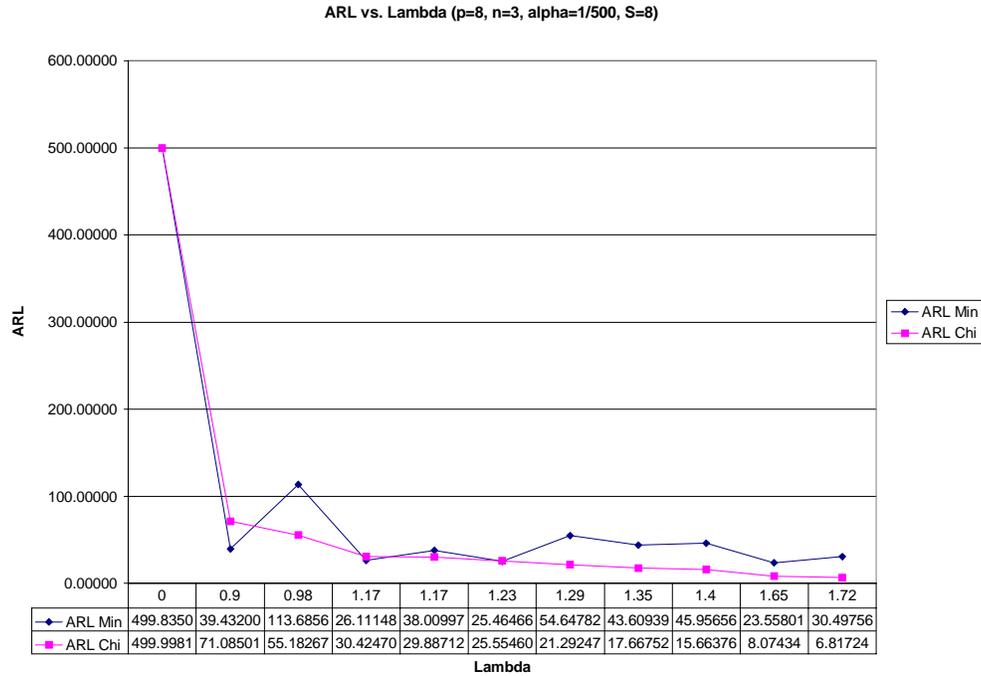
Lambda

Figure 4.11 - ARL vs. Lambda (p=8, n=3, alpha=1/500, S=8)

Lambda	Delta 1	Delta 2	Delta 3	Delta 4	Delta 5	Delta 6	Delta 7	Delta 8	ARL Min	ARL Chi
0	0	0	0	0	0	0	0	0	499.83507	499.99813
0.8962	0.4806	0.1698	0.3186	0.5263	0.6691	0.738	0.5917	0.5825	39.43200	71.08501
0.9766	0.1323	0.3824	0.5936	0.0757	0.1751	0.0095	0.3196	0.5119	113.68568	55.18267
1.1688	0.0814	0.6667	0.9218	0.6219	0.3959	0.3733	0.7134	0.601	26.11148	30.42470
1.1746	0.2882	0.1761	0.2063	0.4084	0.9607	0.2451	0.7492	0.4759	38.00997	29.88712
1.2267	0.1452	0.6681	0.5951	0.3825	0.6293	0.6759	0.9594	0.5319	25.46466	25.55460
1.2885	0.2552	0.3379	0.7379	0.1701	0.0788	0.6901	0.1565	0.7053	54.64782	21.29247
1.3532	0.7601	0.3675	0.7031	0.139	0.1318	0.0024	0.6254	0.5812	43.60939	17.66752
1.3958	0.2146	0.572	0.0715	0.1523	0.5815	0.5986	0.2249	0.9274	45.95656	15.66376
1.648	0.4695	0.5588	0.0635	0.6559	0.9916	0.926	0.0538	0.737	23.55801	8.07434
1.7187	0.0606	0.3647	0.0818	0.3377	0.9732	0.5666	0.0605	0.9851	30.49756	6.81724

Table 4.4 – Data for Figure 4.11

For the case where $\lambda=0.977$ in Table 4.4, the value for the shift of the fourth quality variable is 0.076 standard deviations and the value for the shift of the sixth quality variable is 0.010. The values for the shifts of these two quality variables are negligible in comparison to the other shifts. Therefore, the Minimax chart has a higher ARL than the Chi-Square chart, 114 and 55 respectively. Alternatively, the case in which $\lambda=0.896$

shows that all shifts are substantial. Therefore, the ARL of the Minimax chart is 39 as opposed to the ARL of the Chi-Square chart, 71.

The identity of the variable that experiences the shift in magnitude impacts the ARL just as it does in the five-dimensional case. If the variable that experiences the shift is highly correlated, the Minimax chart performs worse than if the variable that experiences the shift has a lower correlation (Figure 4.12).
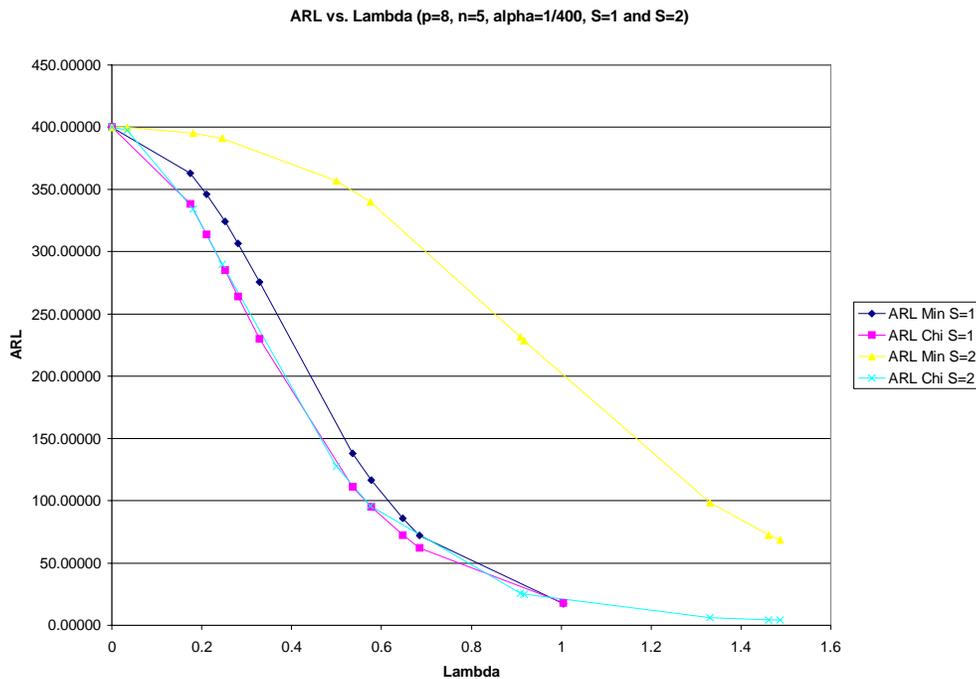
Figure 4.12 – The Relative Sensitivity to the Identity of the Shift Variable (S=1 & S=2)

The sensitivity to the identity of the variable that shifts is not shared to the same degree by the Chi-Square chart. Therefore, the Minimax chart exhibits poorer performance relative to the Chi-Square chart under conditions where the variable that shifts is more highly correlated. This behavior is especially evident when comparing S=1 to S=2 (Appendix E.24-E.26). The behavior just described is not as apparent in the cases where more than one variable experiences a shift, for instance the comparison of S=4 to S=5 (Figure 4.13).

40
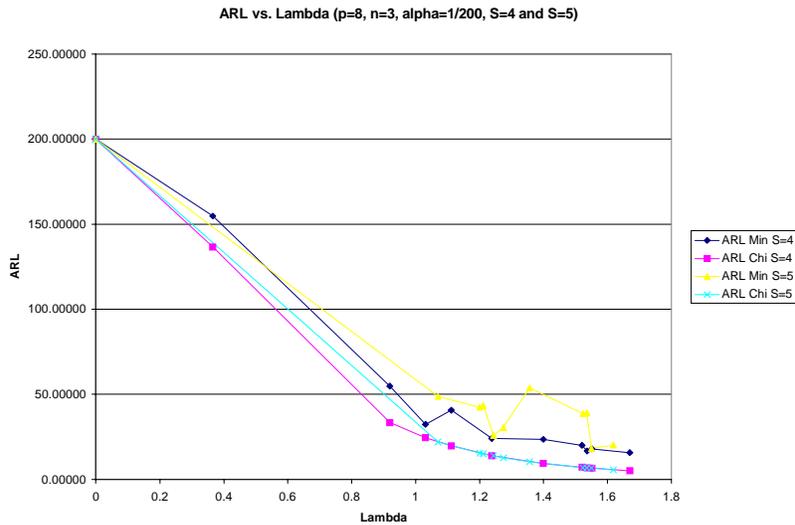
Figure 4.13 – The Relative Sensitivity to the Identity of the Shift Variable (S=4 & S=5)

In Figure 4.13 both of the Minimax charts consistently have a larger ARL than either of the Chi-Square charts (also see Appendix E.27). However, these charts do not seem nearly as well behaved under a shift in three variables as they do in a shift in one variable (Figure 4.12). Although the charts maintain their relative performance, the Minimax chart seems to behave erratically after $\lambda=1$ when three variables experience a shift. This is likely to be the combined effect of the degree of correlation of the shifted variables and the relative magnitudes of the shifts experienced.

Both of the charts are almost equally as effective in detecting shifts of sufficiently large magnitude (Figure 4.14). The charts approach and converge to ARL=1 at about the same value for λ. In the example below, the ARL values approach one at $\lambda=2.82$ and continue to behave in an almost identical manner from that point on. This behavior can also be seen in the examples in Appendix E.28-E.29.

|	0	1.88	2.82	3.33	4.23	4.6	5.04	5.08	5.12	5.6	6.68
ARL Min	199.9875	2.22641	1.03579	1.00522	1.00660	1.00000	1.00000	1.00006	1.00000	1.00000	1.00000
ARL Chi	200.0000	1.17974	1.00037	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
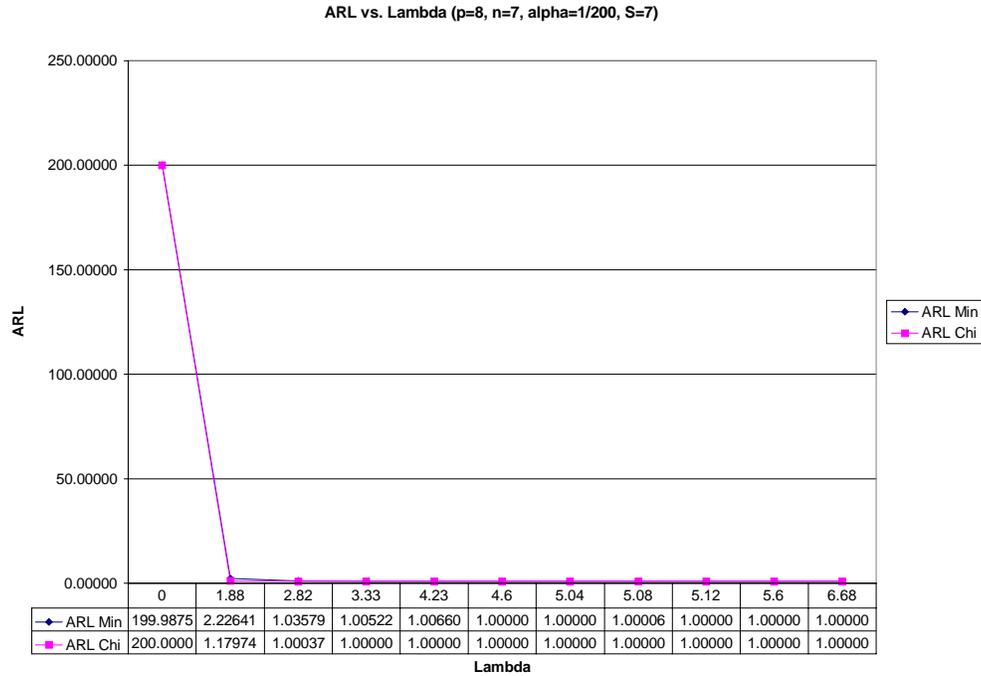
Figure 4.14 – ARL vs. Lambda (p=8, n=7, alpha=1/200, S=7)

4.4 Summary

It is clear that the Minimax chart does not perform as well as the Chi-Square chart when one or more of the quality variables does not experience a shift in the mean value. In cases where all of the variables do experience a shift in the mean value, the performance of the Minimax chart depends upon the combined effect of the magnitude of the shift and the degree of correlation that each of the variables have. The Minimax chart is more sensitive to which variable experiences a shift than the Chi-Square chart. Specifically, if a variable is highly correlated the out-of-control ARL will be higher than if the variable has a low correlation. Both of the charts achieve an ARL near one as the magnitude of the shift increases and are reasonably equivalent after a certain value of λ.

Although the overwhelming majority of the experiments indicated that the Chi-Square chart outperformed the Minimax chart, there are additional issues that should be considered. If there is a reason to suspect that a similar shift in all variables at once is

more likely than a shift in a subset of the variables, then the Minimax may provide for better performance. In addition, the Minimax chart is easier to use in practice than the Chi-Square chart. The Chi-Square chart requires tedious matrix manipulations after each sample. Alternatively, the Minimax chart only requires the comparison of normalized quality characteristic measurements to the appropriate upper and lower control limits.

Further study of the Minimax chart is necessary to more fully understand and explain the performance characteristics uncovered as a result of this thesis. The use of an experimental framework designed to focus on particular performance aspects may provide a clearer picture of the underlying reasons for the observed behaviors.

Chapter 5 – Future Research

There are several opportunities for the further study of the performance characteristics of the Minimax control chart. A more in-depth look at the behaviors highlighted in this thesis is necessary. Specifically, each performance characteristic should be isolated and examined. This should provide more insights into the performance of the Minimax chart and possibly explain the behaviors observed within the confines of this analysis.

This analysis of the Minimax chart is performed under the assumption that the mean vector and covariance matrix associated with the quality variables of interest are known. In addition, only positive entries in the covariance matrix and positive shifts are considered. The out-of-control conditions that remain to be examined are as follows:

- negative shifts in the mean vector of the quality characteristic variables
- covariance structures with positively and negatively correlated quality characteristics
- a survey of the performance under a variety of covariance structures
- conditions under which an out-of-control shift occurs in the covariance matrix as well as a shift in the mean vector of the quality characteristic variables

It should be emphasized that the study of the performance of the Minimax chart under various covariance structures is essential to gain a fuller understanding its behavior.

References

Champ, C. W., and Woodall, W. H., (1987). "Exact Results for Shewhart Control Charts with Supplementary Runs Rules." Technometrics, 29, 393-399.

Ghare, P. H. and Torgerson, P. E. (1968). "The Multicharacteristic Control Chart." Journal of Industrial Engineering, 19, 269-272.

Eichenauer-Herrmann, J. and Grothe, H. (1992) "A New Inversive Congruential Pseudorandom Number Generator with Power of Two Modulus." ACM Transactions on Modeling and Computer Simulation, 2, 1, 1-11.

Hawkins, D. M., (1991). "Multivariate Quality Control Based on Regression-Adjusted Variables." Technometrics, 33, 61-75.

Hayter, A. J., and Tsui, K. L. (1994). "Identification and Quantification in Multivariate Quality Control Problems." Journal of Quality Technology, 26, 197-208.

Hotelling, H. (1947). "Multivariate Quality Control." Techniques of Statistical Analysis, McGraw-Hill, 111-184.

Jackson, J. E., and Mudholkar, G. S. (1979). "Control Procedures for Residuals Associated with Principal Components Analysis." Technometrics, 21, 341-349.

Law, A. M. and Kelton, W. D. (1991). "Simulation Modeling and Analysis." McGraw-Hill, Inc., New York.

L'Ecuyer, P. (1990). "Random Numbers for Simulation." Communications of the ACM, 33, 10, 85-97.

L'Ecuyer, P. (1994). "Recent Advances in Uniform Random Number Generation."
Proceedings of the 1994 Winter Simulation Conference, 176-183.

Lowry, C. A., and Montgomery, D. C. (1995). "A Review of Multivariate Control
Charts." IIE Transactions, 27, 800-810.

Lowry, C. A., Woodall, W. H., Champ, C. W., and Rigdon, S. E. (1992). "A Multivariate
Exponentially Weighted Moving Average Control Chart." Technometrics, 34, 46-
53.

Lucas, J. M. (1973). "A Modified V-Mask Control Scheme." Technometrics, 15, 833-
847.

Lucas, J. M. and Saccucci, M. S. (1990). "Exponentially Weighted Moving Average
Control Schemes: Properties and Enhancements." Technometrics, 32, 1-12.

Marsaglia, G. (1968). "Random Numbers Fall Mainly In The Planes." Proceedings of the
National Acadamy of the Sciences, 61, 25-28.

Niederreiter, H. (1992). "New Methods For Pseudorandom Number And Pseudorandom
Vector Generation." Preceedings of the 1992 Winter Simulation Conference, 264-
269.

Niederreiter, H. (1994). "Pseudorandom Bector Generation by the Inversive Method."
ACM Transactions on Modeling and Computer Simulation, 4, 2, 191-212.

Page, E. S. (1954). "Continuous Inspection Schemes." Biometrika, 41, 100-114.

Page, E. S. (1955). "Control Charts with Warning Lines." Biometrika, 42, 243-259.

Pignatiello, J. J., Jr. And Runger, G. C. (1990). "Comparisons of Multivariate CUSUM Charts." Journal of Quality Technology, 22, 173-186.

Reynolds, M. R., Jr., Amin, R. W., Arnold, J. C. and Nachlas, J. A. (1988). "X-charts with Variable Sampling Intervals." Technometrics, 30, 181-192.

Roberts, S. W. (1959). "Control Charts Based on Geometric Moving Averages." Technometrics, 1, 239-250.

Sepulveda, A. (1996). "The Minimax Control Chart for Multivariate Quality Control." unpublished Ph.D. dissertation, Virginia Polytechnic Institute and State University, Department of Industrial and Systems Engineering.

Sepulveda, A., and Nachlas, J. A. (1997). "The Minimax Chart for Multivariate Quality Control." paper presently under review for IIE Transactions.

Shewhart, W. A. (1931). "The Economic Control of Quality Manufactured Products." Van Nostrand, New York.

Tezuka, S. and L'Ecuyer, P. (1991). "Efficient and Portable Combined Tausworther Random Number Generators." ACM Transactions on Modeling and Computer Simulation, 1, 2, 99-112.

Timm, N. H. (1996). "Multivariate Quality Control Using Finite Intersection Tests." Journal of Quality Technology, 28, 233-243.

Tong, Y.L. (1990). The Multivariate Normal Distribution, Springer-Verlag.

Woodall, W. H. and Ncube, M. M. (1985). "Multivariate CUSUM Quality-Control Procedures." Technometrics, 27, 285-292.

Appendix

Appendix A - Calculation of α and β for the Minimax Chart

In section 3.1.1 the equations (3.10) through (3.18) define the nine mutually exclusive sample outcomes. It is now necessary to define the tail probabilities for $Z_{[1]}$ and $Z_{[p]}$ as:

$$t_1 = P[Z_{[1]} < LCL_{[1]}] = 1 - \int_{LCL_{[1]}}^{\infty} \int_{LCL_{[1]}}^{\infty} \cdots \int_{LCL_{[1]}}^{\infty} f(Z) \prod_{i=1}^{p} dz_i \qquad (A.1)$$

$$t_2 = P[Z_{[1]} > UCL_{[1]}] = \int_{UCL_{[1]}}^{\infty} \int_{UCL_{[1]}}^{\infty} \cdots \int_{UCL_{[1]}}^{\infty} f(Z) \prod_{i=1}^{p} dz_i \qquad (A.2)$$

$$t_3 = P[Z_{[p]} < LCL_{[p]}] = \int_{-\infty}^{LCL_{[p]}} \int_{-\infty}^{LCL_{[p]}} \cdots \int_{-\infty}^{LCL_{[p]}} f(Z) \prod_{i=1}^{p} dz_i \qquad (A.3)$$

$$t_4 = P[Z_{[p]} > UCL_{[p]}] = 1 - \int_{-\infty}^{UCL_{[p]}} \int_{-\infty}^{UCL_{[p]}} \cdots \int_{-\infty}^{UCL_{[p]}} f(Z) \prod_{i=1}^{p} dz_i \qquad (A.4)$$

From the definitions of α_1, α_2, α_3, and α_4 in equations (3.6) through (3.9), it should be noted that when the process is in control $t_1 = \alpha_1$, $t_2 = \alpha_2$, $t_3 = \alpha_3$, and $t_4 = \alpha_4$. Using equations (3.10) through (3.18) and the above definitions of the tail probabilities, the following relationships can be stated:

$$P[E_{b,b}] + P[E_{b,c}] + P[E_{b,a}] = P[Z_{[1]} \in (-\infty, LCL_{[1]}) \mid \Delta)] = t_1 \qquad (A.5)$$

$$P[E_{c,b}] + P[E_{c,c}] + P[E_{c,a}] = P[Z_{[1]} \in (LCL_{[1]}, UCL_{[1]}) \mid \Delta)] = 1-(t_1 + t_2) \qquad (A.6)$$

$$P[E_{a,b}] + P[E_{a,c}] + P[E_{a,a}] = P[Z_{[1]} \in (UCL_{[1]}, \infty) \mid \Delta)] = t_2 \qquad (A.7)$$

$$P[E_{b,b}] + P[E_{c,b}] + P[E_{a,b}] = P[Z_{[p]} \in (-\infty, LCL_{[p]}) \mid \Delta)] = t_3 \qquad (A.8)$$

$$P[E_{b,c}] + P[E_{c,c}] + P[E_{a,c}] = P[Z_{[p]} \in (LCL_{[p]}, UCL_{[p]}) \mid \Delta)] = 1 - (t_3 + t_4) \qquad (A.9)$$

Equations (A.5) through (A.9) form a set of five linearly independent equations with nine unknowns. Four of the unknowns, $P[E_{b,b}]$, $P[E_{b,a}]$, $P[E_{a,b}]$, and $P[E_{a,a}]$, can be calculated using the fact that:

$$(\gamma_1 < Z_i < \gamma_2), \forall_i \Leftrightarrow Z_{[1]} > \gamma_1 \text{ and } Z_{[p]} < \gamma_2 \qquad \text{for } (\gamma_1, \gamma_2 \in \Re) \qquad (A.10)$$

Therefore, in general

$$P[(Z_{[1]} > \gamma_1) \cap (Z_{[p]} < \gamma_2)] = \int_{\gamma_1}^{\gamma_2} \int_{\gamma_1}^{\gamma_2} \cdots \int_{\gamma_1}^{\gamma_2} f(Z) \prod_{i=1}^{p} dz_i \tag{A.11}$$

and, in particular

$$P[E_{a,b}] = \int_{UCL_{[1]}}^{LCL_{[p]}} \int_{UCL_{[1]}}^{LCL_{[p]}} \cdots \int_{UCL_{[1]}}^{LCL_{[p]}} f(Z) \prod_{i=1}^{p} dz_i \ . \tag{A.12}$$

To find $P[E_{a,b}]$, the following identity is developed for two general events e_1 and e_2:

$$P[e_1 \cap e_2] = P[e_1] + P[e_2] + P[\bar{e}_1 \cap \bar{e}_2] - 1 \tag{A.13}$$

Using this identity and (A.11),

$$P[E_{b,a}] = t_1 + t_4 - 1 + \int_{LCL_{[1]}}^{UCL_{[p]}} \int_{LCL_{[1]}}^{UCL_{[p]}} \cdots \int_{LCL_{[1]}}^{UCL_{[p]}} f(Z) \prod_{i=1}^{p} dz_i \ . \tag{A.14}$$

To compute $P[E_{b,b}]$ and $P[E_{a,a}]$ the following identity is used:

$$P[e_1 \cap e_2] = P[e_1] - P[e_1 \cap \bar{e}_2] \tag{A.15}$$

Using this identity and (A.11),

$$P[E_{a,a}] = t_2 - \int_{LCL_{[1]}}^{UCL_{[p]}} \int_{LCL_{[1]}}^{UCL_{[p]}} \cdots \int_{LCL_{[1]}}^{UCL_{[p]}} f(Z) \prod_{i=1}^{p} dz_i \quad \text{and} \tag{A.16}$$

$$P[E_{b,b}] = t_3 - \int_{LCL_{[1]}}^{UCL_{[p]}} \int_{LCL_{[1]}}^{UCL_{[p]}} \cdots \int_{LCL_{[1]}}^{UCL_{[p]}} f(Z) \prod_{i=1}^{p} dz_i \ . \tag{A.17}$$

With the direct solution of $P[E_{b,b}]$, $P[E_{b,a}]$, $P[E_{a,b}]$, and $P[E_{a,a}]$, equations (A.5) through (A.9) become a set of five linearly independent equations with five unknowns and can be used to solve for α and β.

Appendix B - C++ Code for the Tausworthe Random Number Generator

```cpp
#include <iostream.h>
#include <fstream.h>

ofstream OutFile;
long int Q1=13, Q2=2, S1=12, S2=17, P1mS1=19, P2mS2=12, P1mP2=2;
unsigned long int I1=12345, I2=67890, b, Mask1=2147483647, Mask2=536870911;
double Norm=4.656612873e-10;

double CombTause ();

void main()
{
        OutFile.open("p5s5.dat");

        int count;
        double U,V,W,X,Y;

// Avoid repeating random numbers previously used

        for (count=1; count <= 1710; count++)
        {
        U = CombTause();
        }

// Generates the random numbers used in the analysis

        for (count=1; count <= 90; count++)
        {
        U = CombTause();
        V = CombTause();
        W = CombTause();
        X = CombTause();
        Y = CombTause();

        OutFile <<U<<" "<<V<<" "<<W<<" "<<X<<" "<<Y<< endl;
        }
}

// Subroutine that generates pseudorandom numbers

double CombTause()
{

        b = ((I1 << Q1) ^ I1) & Mask1;
        I1 = ((I1 << S1) ^ (b >> P1mS1)) & Mask1;
        b = ((I2 << Q2) ^ I2) & Mask2;
        I2 = ((I2 << S2) ^ (b >> P2mS2)) & Mask2;

        return ((I1 ^ (I2 << P1mP2)) * Norm);

}
```

Appendix C - Mathematica Notebook to Perform the Chart Comparison

```
(*********************************************************************
   THIS PROGRAM COMPUTES THE ARL VALUES FOR THE MINIMAX AND CHI-SQUARE
                    CHARTS IN EIGHT QUALITY VARIABLES
to compute in dimensions other than eight :
change the value for p
change the entry for Sigma
change the entry for Theta (l from page 23 and 24)
change the dimension on Delta to match that of p
*********************************************************************)


(*********************************************************************
The following lines are the initialization of packages needed for the
program to run properly.  Execute them before starting any session.
*********************************************************************)

Needs["Statistics`NormalDistribution`"]
Needs["Statistics`MultinormalDistribution`"]
Needs["Statistics`ContinuousDistributions`"]
Needs["Minimax`"]

(*********************************************************************
Please input the following parameters for the Minimax control chart :
p = the number of quality variables
Sigma  =  the covariance matrix of the quality variables
Theta  =  the l values discussed on page 23 and 24
          to obtain special structure (Tong 1990)
*********************************************************************)


(*********************************************************************
External text files contain all of the out-of-control vectors (Delta)
DeltaInputFile = external text file containing out-of-control vectors
DeltaData = Mathematica table that stores the information in
            DeltaInputFile
OutFile (X) = partial output file
*********************************************************************)

DeltaInputFile = OpenRead["p8.txt"];
DeltaData = ReadList[DeltaInputFile, Number];
Close[DeltaInputFile];

OutFileA = OpenWrite["p8.a"];
OutFileB = OpenWrite["p8.b"];
OutFileC = OpenWrite["p8.c"];
OutFileD = OpenWrite["p8.d"];
OutFileE = OpenWrite["p8.e"];

Off[General::"Spell1"];
Off[NIntegrate::"inum"];
```

```
Sigma = {{1.00, 0.12, 0.13, 0.18, 0.20, 0.23, 0.24, 0.27},
         {0.12, 1.00, 0.19, 0.25, 0.29, 0.33, 0.34, 0.39},
         {0.13, 0.19, 1.00, 0.28, 0.32, 0.37, 0.39, 0.44},
         {0.18, 0.25, 0.28, 1.00, 0.43, 0.49, 0.52, 0.59},
         {0.20, 0.29, 0.32, 0.43, 1.00, 0.55, 0.59, 0.67},
         {0.23, 0.33, 0.37, 0.49, 0.55, 1.00, 0.67, 0.76},
         {0.24, 0.34, 0.39, 0.52, 0.59, 0.67, 1.00, 0.80},
         {0.27, 0.39, 0.44, 0.59, 0.67, 0.76, 0.80, 1.00}};

Theta = {0.286, 0.410, 0.460, 0.614, 0.698, 0.795, 0.841, 0.954};

Index = 1;

p = 8;

Phi = 5.0/10.0;

(**********************************************************************
Series of loops that cycle through all of the experimental cases for
eight dimensions by changing the type of shift, n, and alpha.
**********************************************************************)

For[shiftcounter = 1, shiftcounter [LessEqual] 8, shiftcounter++,

For[ncounter = 1, ncounter [LessEqual] 3, ncounter++,
  If[ncounter == 1, n = 3;];
  If[ncounter == 2, n = 5;];
  If[ncounter == 3, n = 7;];

  For[alphacounter = 1, alphacounter [LessEqual] 3, alphacounter++,
    If[alphacounter == 1, (DesignAlpha = 1/200.0;];
    If[alphacounter == 2, (DesignAlpha = 1/400.0;];
    If[alphacounter == 3, (DesignAlpha = 1/500.0;];

    For[counter = 1, counter [LessEqual] 11, counter++,
      If[counter == 1,
        Delta = {0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0};
        FindControlLimits = True;

      ,  (* else *)

        Delta =
       {DeltaData[[Index++]],DeltaData[[Index++]],DeltaData[[Index++]],
        DeltaData[[Index++]],DeltaData[[Index++]],DeltaData[[Index++]],
        DeltaData[[Index++]],DeltaData[[Index++]]};

        FindControlLimits = False;
      ];  (* if stmt to determine delta *)


(**********************************************************************
Evaluates each experimental case and generates ARL' s for the Minimax
and Chi-Square charts, as well as other output :

Lambda = measure of the magnitude of the shift in the mean vector
**********************************************************************)
```

```
       Alpha4 = (DesignAlpha Phi)/2;
       Lambda = Sqrt(Delta.Inverse[Sigma].Delta);
       Approved = True;
       If[! FindControlLimits,
          If[Lambda == 0,
             Print["Set Lambda > 0 before calculating
                    Out of Control ARL"];
             Approved = False;
           ,
             ControlLimits = {-LCLp, -UCLp, UCLp, LCLp};
          ];  (* if stmt to check lambda  =  0 *)
        ,

          If[Lambda [NotEqual] 0,
             Print["Set Lambda = 0 before trying to find
                    the control limits!"];
             Approved = False;
          ];  (* if stmt to check lambda  [NotEqual]  0 *)
       ];  (* if stmt to see if control limits should be found *)

       If[Approved,
          {Alpha3,UCLp,LCLp,ARLMinMax}= EvaluateCase[Sigma,Delta,Theta,
                                         Lambda,p,n,DesignAlpha,Alpha4,
                                         FindControlLimits,ControlLimits];
          UCLChiSq = CalcUCLChiSq[DesignAlpha, p];
          ARLChiSq = CalcARLChiSq[p, n, Lambda, UCLChiSq];
          Print["Alpha3= ",Alpha3,"  UCLp= ",UCLp,"  LCLp= ",LCLp,
                "Lambda= ",Lambda," Delta= ",Delta[[1]]," ",Delta[[2]],
                " ",Delta[[3]]," ",Delta[[4]], " ",Delta[[5]]," ",
                Delta[[6]]," ",Delta[[7]]," ",Delta[[8]],"  ARL= ",
                ARLMinMax,"  ARLChiSq= ", ARLChiSq];

          Write[OutFileA,p,q,n,q,DesignAlpha,q,Alpha4,q,N[Alpha3, 6] ];
          Write[OutFileB,N[UCLp, 6],q,N[LCLp, 6],q,
             If[counter == 1, Lambda, N[Lambda, 6]] ];
          Write[OutFileC,Delta[[1]],q,Delta[[2]],q,Delta[[3]] ];
          Write[OutFileD,Delta[[4]],q,Delta[[5]],q,Delta[[6]],q,
                Delta[[7]] ];
          Write[OutFileE,Delta[[8]],q,N[ARLMinMax, 6],q,N[ARLChiSq, 6] ];
        ,
          Print["Some information is missing or input parameters are
                  not logical.
                  \n Please make the appropriate corrections.>"];
       ]; (* if stmt that generates output *)
    ]; (* for loop that generates 11 observations in each experiment *)
  ]; (*  for loop that changes alpha *)
]; (* for loop that changes n *)
]; (* for loop that changes the shift *)
Close[OutFileA];
Close[OutFileB];
Close[OutFileC];
Close[OutFileD];
Close[OutFileE];
```

Appendix D – Mathematica Package to Perform the Chart Comparison

```
(* Minimax Control Chart Statistics *)

BeginPackage[
        "Minimax`",
        "Statistics`InverseStatisticalFunctions`",
        "Statistics`Common`DistributionsCommon`",
        "Statistics`DescriptiveStatistics`",
        "Statistics`NormalDistribution`",
        "Statistics`ContinuousDistributions`"
]

CumF::usage = " Definition of the cummulative multivariate normal function"

EvaluateCase::usage = " EvaluateCase[Sigma_, Delta_, Theta_, Lambda_,  p_, n_,
        DesignAlpha_, Alpha4_, FindControlLimits_, ControlLimits_]
        Returns Alpha3, UCLp, LCLp, and ARLMinMax for the input parameters
        Sigma, Delta, p, n, DesignAlpha,
        Alpha4, and FindControlLimits (True if control limits are unknown)"

CalcAlpha::usage = "CalcAlpha[UCLp, LCLp, UCL1, LCL1, F, Alpha3, Alpha4]
        Calculates THE PROBABILITY OF TYPE I ERROR for the specified control
        limits UCLp, LCLp, UCL1, and LCL1.  F is the joint density of the
        vector Z which must be evaluated from a to b in all the limits of
        the integral."

CalcBeta::usage = "CalcBeta[UCLp, LCLp, UCL1, LCL1, F] Calculates Beta
        for the specified control limits UCLp, LCLp, UCL1, LCL1.  F is the
        joint density of the vector Z which must be evaluated from a to b
        in all the limits of the integral."

CalcUCLChiSq::usage = "CalcUCLChiSq[DesignAlpha_, p_] Calculates the UCL
        of the Chi-Squared control chart for the specified probability of
        Type I error = DesignAlpha and number of variables = p."

CalcARLChiSq::usage = "CalcARLChiSq[p_, n_, Lambda_, UCLChiSq_]
        Calculates the ARL of the Chi-Squared Control Chart for the number
        of variables = p, sample size = n, distance of the shift in the
        mean = Lambda, and the Upper Control Limit of the Chart = UCLChiSq"

CalcLCLp::usage = "CalcLCLp[DesignAlpha, Alpha4, UCLp_, LCL1_, F_]
        Calculates LCLp for the given values of DesignAlpha, Alpha4, and UCLp"

CalcUCLp::usage = "CalcUCLp[Alpha4_, F_] Calculates UCLp for the given Alpha4 and F"

Begin["`Private`"]

CalcUCLChiSq[DesignAlpha_, p_]:=
        Replace[ULCS, FindRoot[CDF[ChiSquareDistribution[p], ULCS]
        == (1-DesignAlpha), ULCS, 1, 100]];

CalcARLChiSq[p_, n_, Lambda_, UCLChiSq_]:=
        1/(1-CDF[NoncentralChiSquareDistribution[p,(n*Lambda^2)], UCLChiSq]);

CalcAlpha[UCLp_, LCLp_, UCL1_, LCL1_, F_, Alpha3_, Alpha4_]:=
Module[t1, t2, t3, t4, PA, PB, PC, PD, PE, PG, PH, PI, Answer,
        t1 = Alpha4;
        t2 = Alpha3;
        t3 = Alpha3;
        t4 = Alpha4;
        If[LCLp > UCL1, PG = F[UCL1, LCLp], PG = 0];
        PC = -1 + t1 + t4 + F[LCL1, UCLp];
        PI = t2 - F[UCL1, UCLp];
        PA = PI;
        PH := t2 - PI - PG;
        (* PD := t3 - PG - PA; *)
        PB := t1 - PA - PC;
        PE := Re[1 - t3 - t4 - PH - PB];
        Answer = 1 - PE
]

CalcBeta[UCLp_, LCLp_, UCL1_, LCL1_, F_]:=
```

55

```
Module[t1, t2, t3, t4, PA, PB, PC, PD, PE, PG, PH, PI, Answer,
        t1 = 1 - F[LCL1, Infinity];
        t2 = F[UCL1, Infinity];
        t3 = F[-Infinity, LCLp];
        t4 = 1 - F[-Infinity, UCLp];
        If[LCLp > UCL1, PG = F[UCL1, LCLp], PG = 0];
        PC = -1 + t1 + t4 + F[LCL1, UCLp];
        PI = t2 - F[UCL1, UCLp];
        PA = t3 - F[LCL1, LCLp];
        PH := t2 - PI - PG;
        (* PD := t3 - PG - PA; *)
        PB := t1 - PA - PC;
        PE := Re[1 - t3 - t4 - PH - PB];
        Answer = PE
]

CalcLCLp[DesignAlpha_, Alpha4_, UCLp_, LCL1_, F_, p_]:=
Module[Err, NearLarge, NearSmall, Shift, Counter, NearestSmallARL,
        NearestLargeARL, UCL1List, Alpha3List, ARLList, Alpha3,
            (* Assigning LCLp assuming independent variables *)
            StdPhi[z_] := CDF[NormalDistribution[0,1], z];
            AlphaMin = 1 - Sqrt[1 - DesignAlpha];
            Alpha3 = AlphaMin - Alpha4;
            LCLp = Replace[t, FindRoot[StdPhi[t] == (Alpha3)^(1/p), t,0]];
            (* Searching for LCLp to meet the required DesignAlpha *)
            Err = 100;
            NearLarge = NearSmall = False;
            Shift = 0.5;
            Counter = 0;
            NearestSmallARL = 0;
            NearestLargeARL = 100/DesignAlpha;
            While[Err > 0.5,
                    Counter++;
                    UCL1 = -LCLp;
                    Alpha3 = F[-Infinity, LCLp];
                    Alpha = CalcAlpha[UCLp, LCLp, UCL1, LCL1, F, Alpha3, Alpha4];
                    ARL = 1/Alpha;
                    Err = Abs[ARL - 1/DesignAlpha];
                    If[(ARL > NearestSmallARL)&&(ARL < 1/DesignAlpha),
                            NearestSmallARL = ARL;
                            NearestSmallLCLp = LCLp;
                            NearSmall = True;
                    ];
                    If[(ARL < NearestLargeARL)&&(ARL > 1/DesignAlpha),
                            NearestLargeARL = ARL;
                            NearestLargeLCLp = LCLp;
                            NearLarge = True;
                    ];
                    If[NearLarge && NearSmall,
                            ForcastLCLp = Interpolation[NearestLargeARL,
                            NearestLargeLCLp,NearestSmallARL, NearestSmallLCLp,
                            InterpolationOrder -> 1];
                            SaveLCLp = LCLp;
                            LCLp = ForcastLCLp[1/DesignAlpha];
                    ,
                            If[Counter > 1,
                                    Fw = Fit[SaveARL, SaveLCLp, ARL, LCLp, 1, PredARL,
                                    PredARL];
                                    PredARL = 1/DesignAlpha;
                                    SaveLCLp = LCLp;
                                    LCLp = Fw;
                            ,
                                    If[1/Err < 0.001, Shift = (1/Err)/0.001 * 0.5];

                                    SaveLCLp = LCLp;

                                    If[ARL < 1/DesignAlpha,
                                            LCLp = LCLp - Shift;
                                    ,
                                            LCLp = LCLp + Shift;
                                    ]; (* End If *)
                            ]; (* End If *)
                    ]; (* End IF *)

                    SaveARL = ARL;
            ]; (* End While *)
```

```
                    LCLp = SaveLCLp;

                    Answer = LCLp, Alpha3, ARL

    ] (* End Module *)

    CalcUCLp[Alpha4_, F_, p_]:=
    Module[Err, NearLarge, NearSmall, Shift, Counter, NearestSmallAlpha4,
                    NearestLargeAlpha4,
                    (* Searching for UCLp to meet the required DesignAlpha *)
                    Err = 1;
                    NearLarge = NearSmall = False;
                    Shift = 0.5;
                    Counter = 0;
                    (* Assuming UCLp assuming independent variables *)
                    StdPhi[z_] := CDF[NormalDistribution[0,1], z];
                    UCLp = Replace[t, FindRoot[StdPhi[t] == (1 - Alpha4)^(1/p),t,0]];
                    NearestSmallTail4 = 0;
                    NearestLargeTail4 = 1;

                    While[Err > 0.00001,
                            Counter++;
                            Tail4 = 1 - F[-Infinity, UCLp];
                            Err = Abs[Tail4 - Alpha4];
                            If[Tail4 > NearestSmallTail4 && (Tail4 < Alpha4),
                                    NearestSmallTail4 = Tail4;
                                    NearestSmallUCLp = UCLp;
                                    NearestSmall = True;
                            ];
                            If[Tail4 < NearestLargeTail4 && (Tail4 > Alpha4),
                                    NearestLargeTail4 = Tail4;
                                    NearestLargeUCLp = UCLp;
                                    NearLarge = True; (* Tail4 > Alpha4 but close to it *)
                            ];
                            If[NearLarge && NearSmall,
                                    ForcastUCLp = Interpolation[NearestLargeTail4,
                                    NearestLargeUCLp, NearestSmallTail4, NearestSmallUCLp,
                                    InterpolationOrder -> 1];
                                    SaveUCLp = UCLp;
                                    UCLp = ForcastUCLp[Alpha4];
                            ,
                                    If[Counter > 1,
                                            Fw = Fit[SaveAlpha4, SaveUCLp,Tail4, UCLp, 1,
                                            PredAlpha4, PredAlpha4];
                                            PredAlpha4 = Alpha4;
                                            SaveUCLp = UCLp;
                                            UCLp = Fw;
                                    ,
                                            If[Err < 0.001, Shift = Err/0.001 * 0.5];
                                            SaveUCLp = UCLp;
                                            If[Tail4 < Alpha4,
                                                    UCLp = UCLp - Shift;
                                            ,
                                                    UCLp = UCLp + Shift;
                                            ]; (* End If *)
                                    ]; (* End If *)
                            ]; (* End If *)

                            SaveAlpha4 = Tail4;
                    ]; (* End While *)

                    UCLp = SaveUCLp;
                    Answer = UCLp
    ] (* End Module *)

    CumF[MuZ_, Theta_, p_, a_, b_] := CompoundExpression[
            StdPhi[z_] := CDF[NormalDistribution[0,1],z];
            F1 := Re[N[Integrate[Simplify[1/Sqrt[2 Pi] Exp[-(Yy^2)/2]*
                    Simplify[Product[
                            StdPhi[ ((b - MuZ[[i]]) + Theta[[i]]*Yy)/Sqrt[1 - Theta[[i]]^2]] -
                            StdPhi[ ((a - MuZ[[i]]) + Theta[[i]]*Yy)/Sqrt[1 - Theta[[i]]^2]],
                            i, 1, p]]],
                    Yy, -Infinity, Infinity]]];
            Return[F1]
```

57

```
] (* End of CompoundExpression *)

EvaluateCase[Sigma_, Delta_, Theta_, Lambda_, p_, n_, DesignAlpha_, Alpha4_,
                     FindControlLimits_, ControlLimits_] := CompoundExpression[
                  (* Defining MuZ, Delta(i) is the shift on the mean of
                        x(i) measured in std dev of x(i) *)

                  MuZ = Delta*Sqrt[n];
                  MuX = Delta;

                  If[FindControlLimits, (* Sets control limits for DesignAlpha *)
                          (* Theta is Lambda in Y.L. Tong 8.1.5 *)
                          F[a_, b_] = CumF[MuZ, Theta, p, a, b];
                          (* Searching for UCLp *)
                          UCLp = CalcUCLp[Alpha4, F, p];
                          LCL1 = -UCLp;

                          (* Searching for LCLp to meet the required DesignAlpha *)
                          LCLp, Alpha3, ARLMinMax = CalcLCLp[DesignAlpha, Alpha4,
                          UCLp, LCL1, F, p];
                          UCL1 = -LCLp;
                  ];
                  If[Lambda > 0, (* Calculate ARL for a given shift in the mean *)
                          UCL1, LCL1, UCLp, LCLp = ControlLimits;
                          (* Theta is Lambda in Y.L. Tong 8.1.5 *)
                          F[a_, b_] = CumF[MuZ, Theta, p, a, b];
                          MyBeta = CalcBeta[UCLp, LCLp, UCL1, LCL1, F];
                          ARLMinMax = 1/(1-MyBeta);
                  ];
                  Return[Alpha3, UCLp, LCLp, ARLMinMax]

] (* End of Compound Expression for EvaluateCase *)

End[] (* End of Private Context *)

EndPackage[]
```

Appendix E – Experimental Data

ARL vs. Lambda (p=2, n=5, alpha=1/400, S=1)

	0	0.1	0.2	0.32	0.34	0.4	0.56	0.94	0.95	0.96	1.03
ARL Min	400.4282	358.0053	272.0505	164.7983	153.6927	117.0372	54.21272	11.20223	10.92118	10.46451	8.08128
ARL Chi	400.0000	345.3243	244.3572	134.9504	124.6062	91.74972	40.10957	8.30541	8.10577	7.78143	6.08880
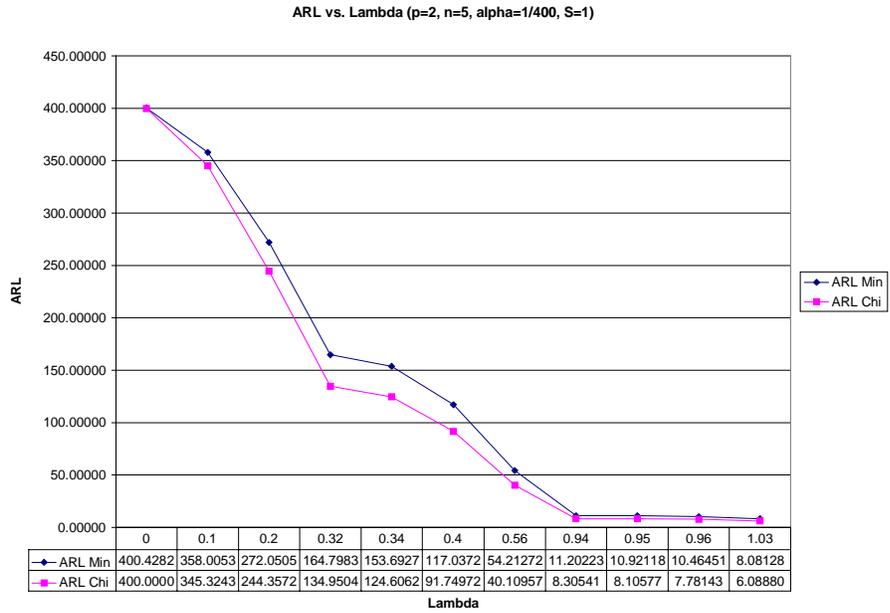
Figure E.1 – ARL vs. Lambda (p=2, n=5, alpha=1/400, S=1)

ARL vs. Lambda (p=2, n=7, alpha=1/500, S=1)

	0	0.09	0.12	0.23	0.23	0.26	0.56	0.6	0.94	0.99	1.01
ARL Min	499.9980	434.2541	397.2566	243.6689	242.5934	215.6339	40.52114	32.06436	6.71250	5.59617	5.28900
ARL Chi	499.9998	415.9246	371.3385	205.3276	204.2674	178.1389	29.48567	23.27689	5.12839	4.33083	4.11072
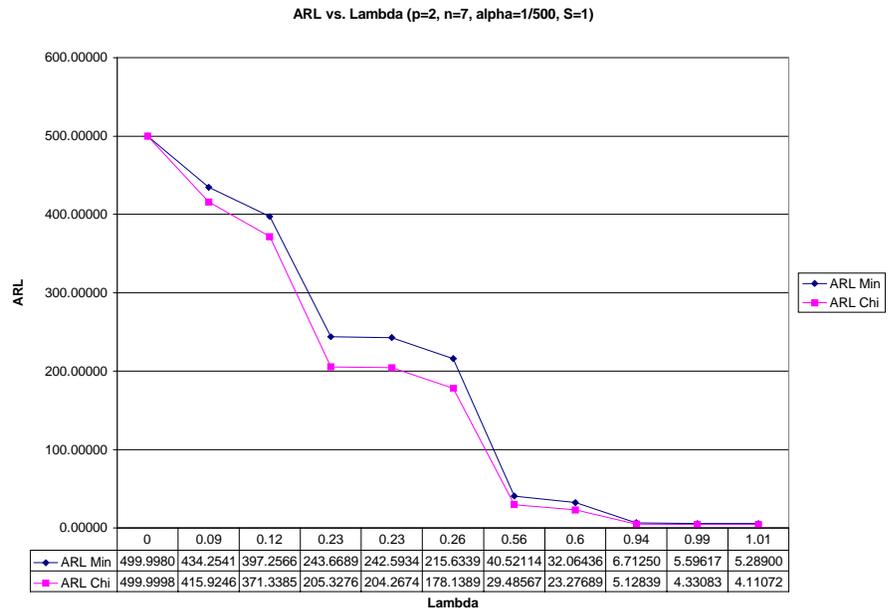
Figure E.2 - ARL vs. Lambda (p=2, n=7, alpha=1/500, S=1)
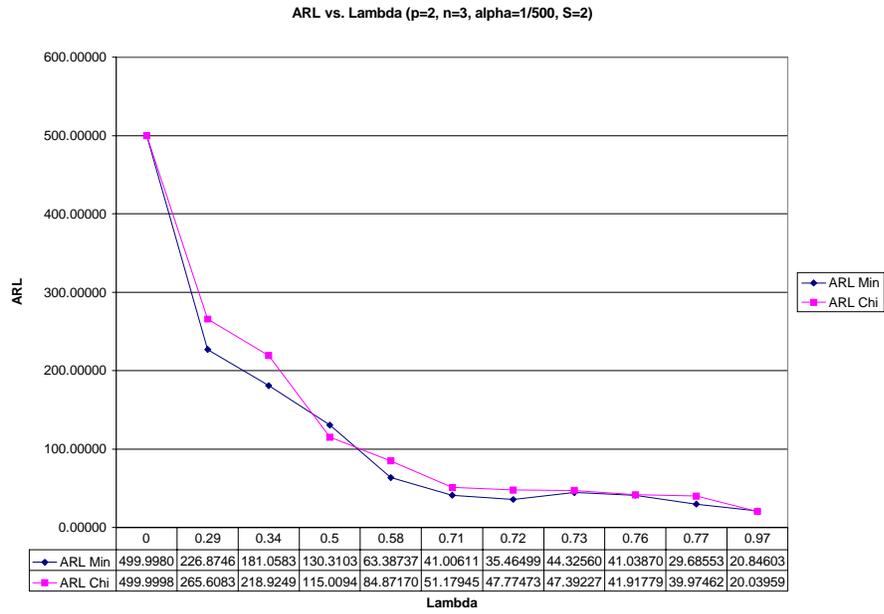
Figure E.3 - ARL vs. Lambda (p=2, n=3, alpha=1/500, S=2)

Lambda	Delta 1	Delta 2	ARL Min	ARL Chi
0	0	0	499.99800	499.99984
0.28881	0.26986	0.167089	226.87469	265.60834
0.340345	0.20237	0.315547	181.05830	218.92490
0.50198	0.0569204	0.497135	130.31036	115.00947
0.57757	0.406728	0.498733	63.38737	84.87170
0.706456	0.669638	0.38535	41.00611	51.17945
0.724452	0.621852	0.51532	35.46499	47.77473
0.726561	0.723138	0.248995	44.32560	47.39227
0.759016	0.758159	0.22446	41.03870	41.91779
0.771691	0.658187	0.554615	29.68553	39.97462
0.965842	0.965842	0.241631	20.84603	20.03959

Table E.3 – Data for Figure E.3
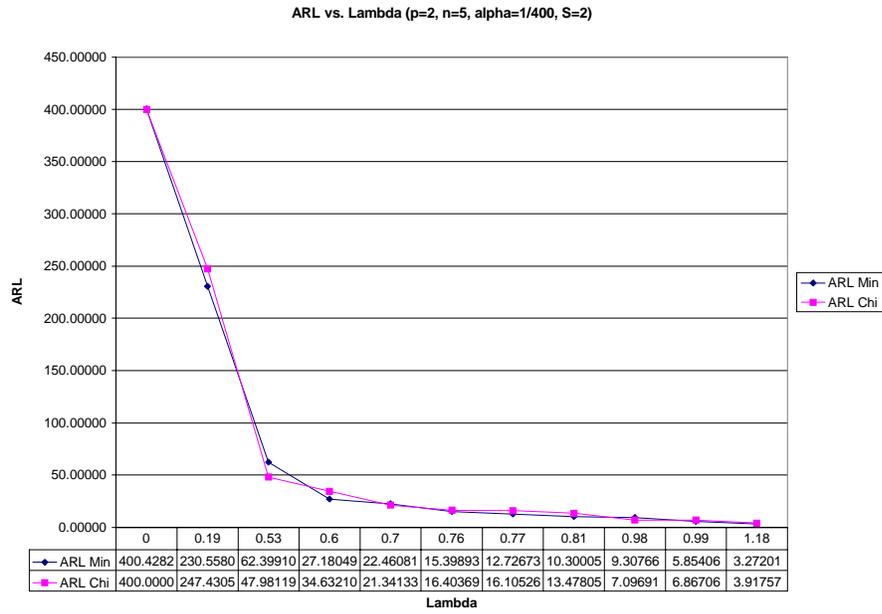
Figure E.4 – ARL vs. Lambda (p=2, n=5, alpha=1/400, S=2)

Lambda	Delta 1	Delta 2	ARL Min	ARL Chi
0	0	0	499.99800	499.99984
0.28881	0.26986	0.167089	226.87469	265.60834
0.340345	0.20237	0.315547	181.05830	218.92490
0.50198	0.0569204	0.497135	130.31036	115.00947
0.57757	0.406728	0.498733	63.38737	84.87170
0.706456	0.669638	0.38535	41.00611	51.17945
0.724452	0.621852	0.51532	35.46499	47.77473
0.726561	0.723138	0.248995	44.32560	47.39227
0.759016	0.758159	0.22446	41.03870	41.91779
0.771691	0.658187	0.554615	29.68553	39.97462
0.965842	0.965842	0.241631	20.84603	20.03959

Table E.4 – Data for Figure E.4
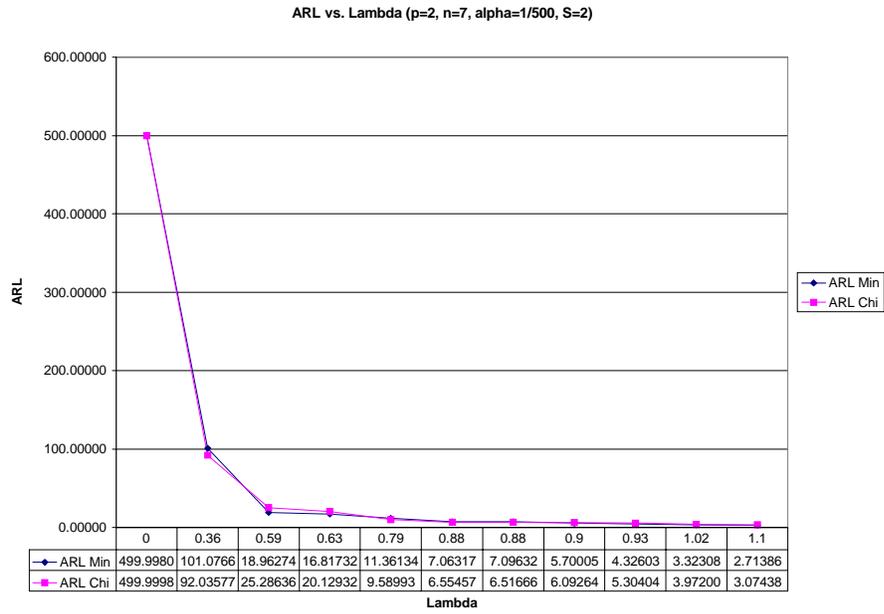
Figure E.5 – ARL vs. Lambda (p=2, n=7, alpha=1/500, S=2)

Lambda	Delta 1	Delta 2	ARL Min	ARL Chi
0	0	0	499.99800	499.99984
0.364858	0.056817	0.363167	101.07660	92.03577
0.58799	0.497337	0.428042	18.96274	25.28636
0.631424	0.603563	0.330495	16.81732	20.12932
0.78667	0.0929472	0.779591	11.36134	9.58993
0.878007	0.192002	0.877551	7.06317	6.55457
0.879481	0.180322	0.878543	7.09632	6.51666
0.896786	0.879123	0.391265	5.70005	6.09264
0.933719	0.748546	0.727544	4.32603	5.30404
1.017415	0.783442	0.824365	3.32308	3.97200
1.101438	0.98784	0.718663	2.71386	3.07438

Table E.5 – Data for Figure E.5

ARL vs. Lambda (p=2, n=3, alpha=1/200, S=2)

	0	0.3	0.4	0.51	0.63	0.67	0.74	0.96	0.99	0.99	1.01
ARL Min	200.1617	97.33362	62.77678	46.64404	30.16350	22.67598	18.48715	9.67140	8.71429	14.02039	9.05779
ARL Chi	200.0037	111.9436	79.13217	53.55822	34.29304	30.15239	23.63697	11.93175	10.89862	10.67713	10.27258
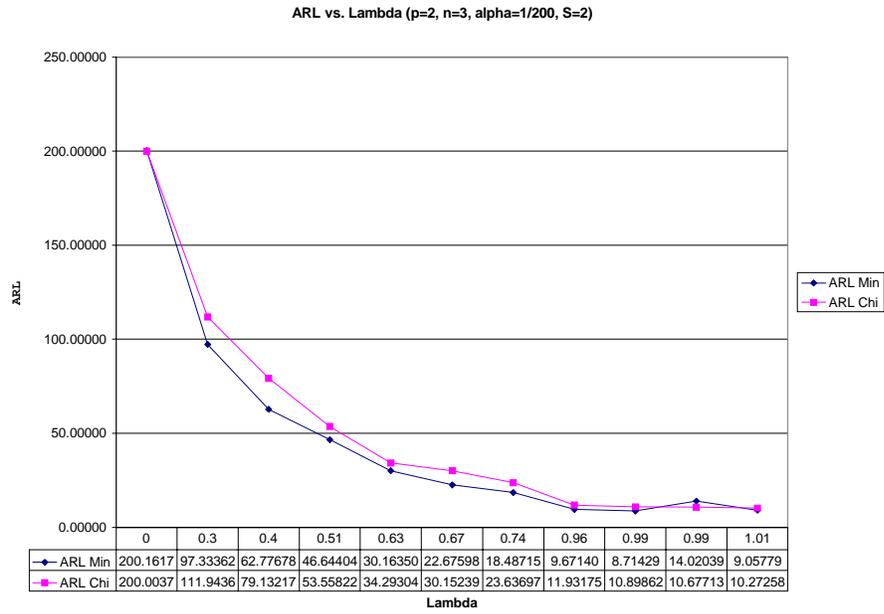
Figure E.6 – ARL vs. Lambda (p=2, n=3, alpha=1/200, S=2)

Lambda	Delta 1	Delta 2	ARL Min	ARL Chi
0	0	0	200.16175	200.00371
0.298452	0.170301	0.279887	97.33362	111.94366
0.400024	0.327859	0.303876	62.77678	79.13217
0.509009	0.226908	0.497893	46.64404	53.55822
0.634038	0.270676	0.62282	30.16350	34.29304
0.670901	0.516676	0.543545	22.67598	30.15239
0.74212	0.468155	0.674578	18.48715	23.63697
0.956977	0.574046	0.884885	9.67140	11.93175
0.987633	0.629593	0.894177	8.71429	10.89862
0.994671	0.0266635	0.969406	14.02039	10.67713
1.008002	0.473856	0.979892	9.05779	10.27258

Table E.6 – Data for Figure E.6

ARL vs. Lambda (p=5, n=3, alpha=1/200, S=1)

	0	0.02	0.09	0.19	0.23	0.3	0.38	0.41	0.85	0.96	0.97
ARL Min	199.9910	199.8534	196.3987	184.2359	176.8885	160.3064	141.5456	134.0519	39.65690	26.96807	26.55936
ARL Chi	200.0005	199.7491	193.5577	173.3830	162.2735	139.6630	117.4072	109.3175	30.37124	21.30528	21.01171
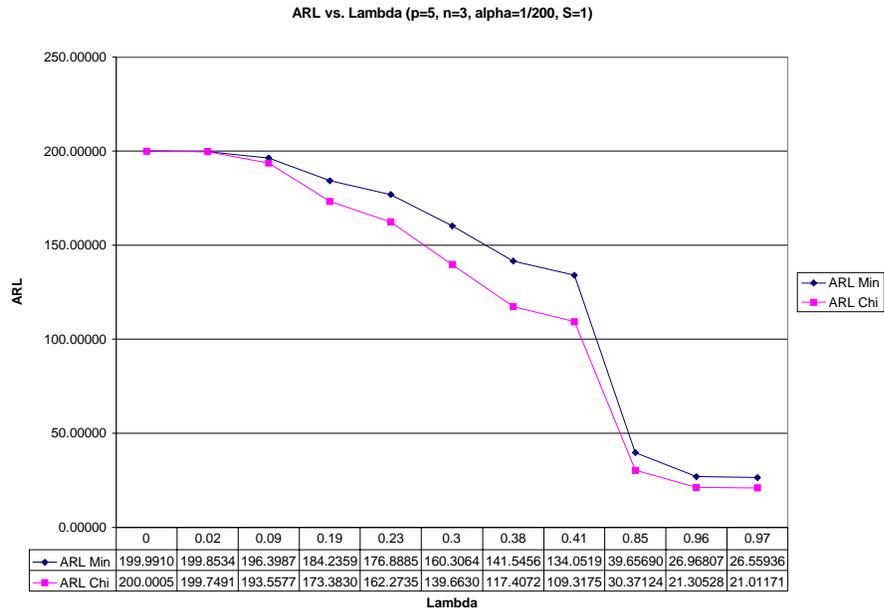
Figure E.7 – ARL vs. Lambda (p=5, n=3, alpha=1/200, S=1)

ARL vs. Lambda (p=5, n=7, alpha=1/400, S=2)

	0	0.07	0.13	0.14	0.57	0.89	0.92	0.93	0.94	0.98	0.99
ARL Min	399.8140	389.2077	367.9321	360.2927	71.90599	13.66505	12.05675	11.58826	10.85307	9.36124	9.06878
ARL Chi	400.0012	376.5609	334.8980	321.4113	47.33687	10.34564	9.23399	8.90751	8.39253	7.33667	7.12782
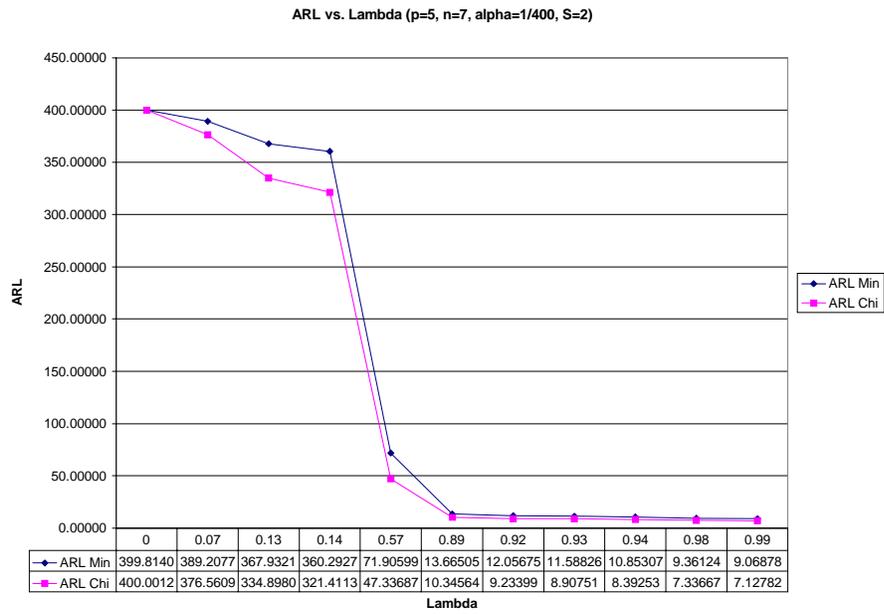
Figure E.8 – ARL vs. Lambda (p=5, n=7, alpha=1/400, S=2)

ARL vs. Lambda (p=5, n=5, alpha=1/200, S=3)

	0	0.04	0.09	0.29	0.43	0.43	0.58	0.66	0.84	1.09	1.15
ARL Min	199.9806	199.1964	196.3011	163.4011	121.7556	121.5080	77.86276	58.69042	29.22327	11.71303	9.59477
ARL Chi	200.0005	197.3674	188.1712	119.0207	72.37125	72.15839	41.07709	30.17705	15.12275	6.52490	5.46123
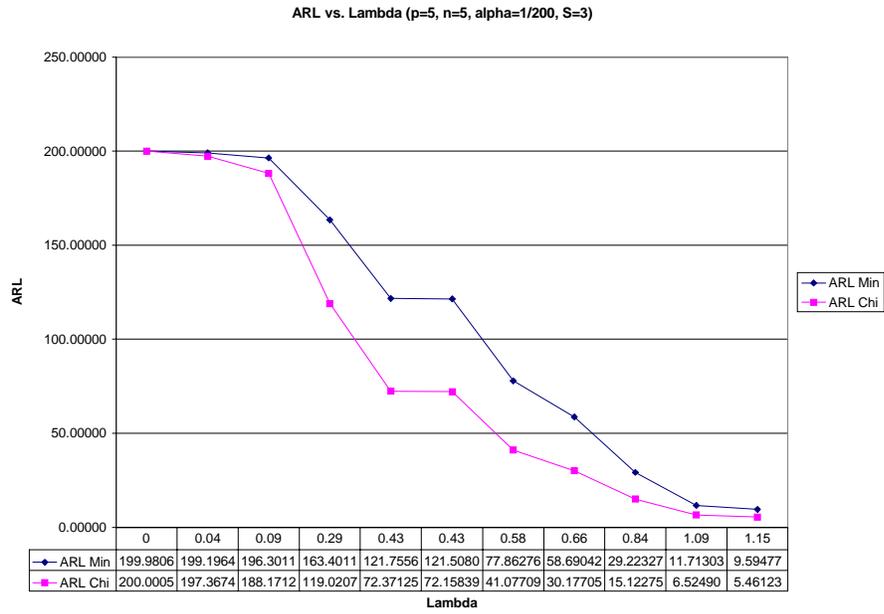
Figure E.9 – ARL vs. Lambda (p=5, n=5, alpha=1/200, S=3)

ARL vs. Lambda (p=5, n=3, alpha=1/400, S=5)

	0	0.65	0.85	0.93	1	1.05	1.17	1.18	1.19	1.44	1.54
ARL Min	399.8140	131.9684	75.20250	54.20328	48.71620	40.49809	31.36796	32.13918	29.91842	18.81720	15.64334
ARL Chi	400.0012	96.95395	50.25558	38.43537	29.98100	26.21604	17.88154	17.62608	16.74355	8.36972	6.57923
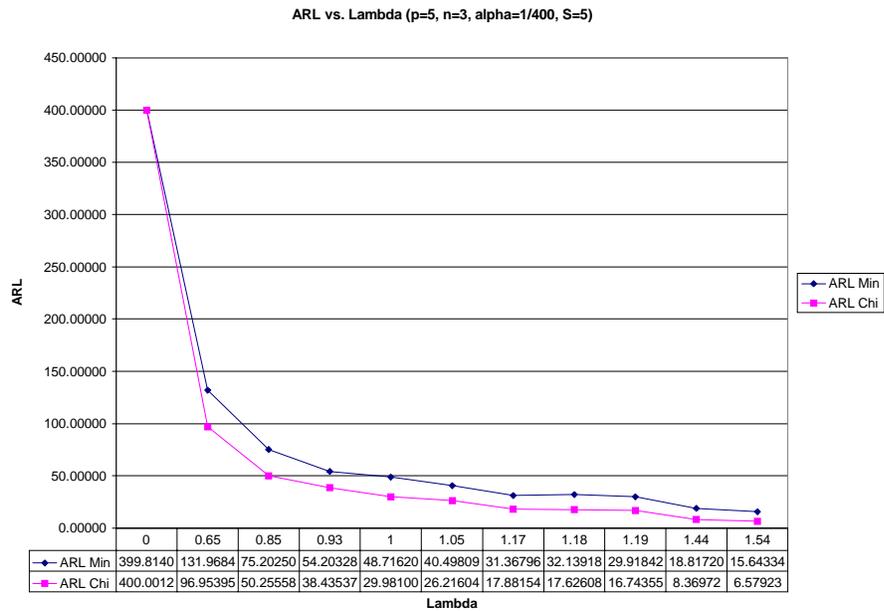
Figure E.10 – ARL vs. Lambda (p=5, n=3, alpha=1/400, S=5)

ARL vs. Lambda (p=5, n=5, alpha=1/500, S=8)

	0	0.74	0.8	0.82	0.97	1.03	1.07	1.11	1.12	1.21	1.33
ARL Min	499.9238	37.38763	29.36496	25.03114	9.13264	16.25149	14.88110	7.20903	12.24100	6.03245	8.49329
ARL Chi	499.9999	40.88273	31.30971	29.38506	15.85549	12.74287	11.25950	9.59766	9.44567	6.92331	4.87430
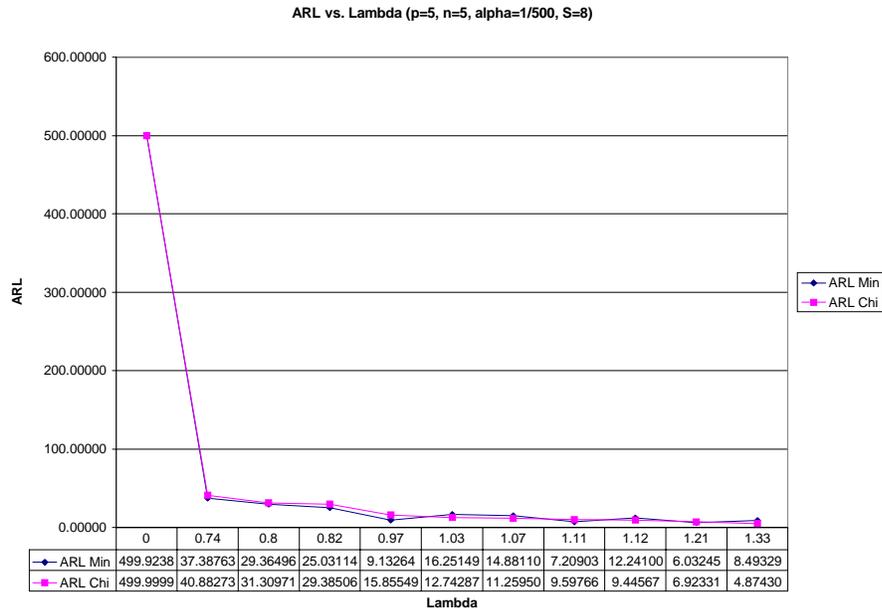
Lambda

Figure E.11 – ARL vs. Lambda (p=5, n=5, alpha=1/500, S=8)

Lambda	Delta 1	Delta 2	Delta 3	Delta 4	Delta 5	ARL Min	ARL Chi
0	0	0	0	0	0	499.92386	499.99999
0.740181	0.496659	0.319883	0.08931	0.30125	0.601193	37.38763	40.88273
0.803046	0.327859	0.303876	0.468155	0.674578	0.170301	29.36496	31.30971
0.818239	0.390816	0.293953	0.139255	0.505187	0.741083	25.03114	29.38506
0.972614	0.470049	0.707416	0.580402	0.637932	0.567664	9.13264	15.85549
1.031011	0.073086	0.545941	0.792531	0.186769	0.736681	16.25149	12.74287
1.065189	0.394212	0.572226	0.471424	0.076636	0.87818	14.88110	11.25950
1.110649	0.493621	0.939356	0.390805	0.703188	0.599836	7.20903	9.59766
1.115285	0.090758	0.928408	0.056564	0.718841	0.607987	12.24100	9.44567
1.209322	0.3025	0.836567	0.89525	0.801536	0.562419	6.03245	6.92331
1.326426	0.279887	0.629593	0.894177	0.026663	0.969406	8.49329	4.87430

Table E.11 – Data for Figure E.11

ARL vs. Lambda (p=5, n=7, alpha=1/400, S=8)

	0	0.82	0.89	1.01	1.09	1.11	1.12	1.19	1.2	1.31	1.38
◆ ARL Min	399.8140	12.29587	9.61079	10.99427	5.23913	5.85946	3.67644	4.06436	4.53005	3.33081	3.16488
■ ARL Chi	400.0012	14.09076	10.52027	6.58302	4.94189	4.59440	4.48737	3.58476	3.50279	2.58404	2.23343
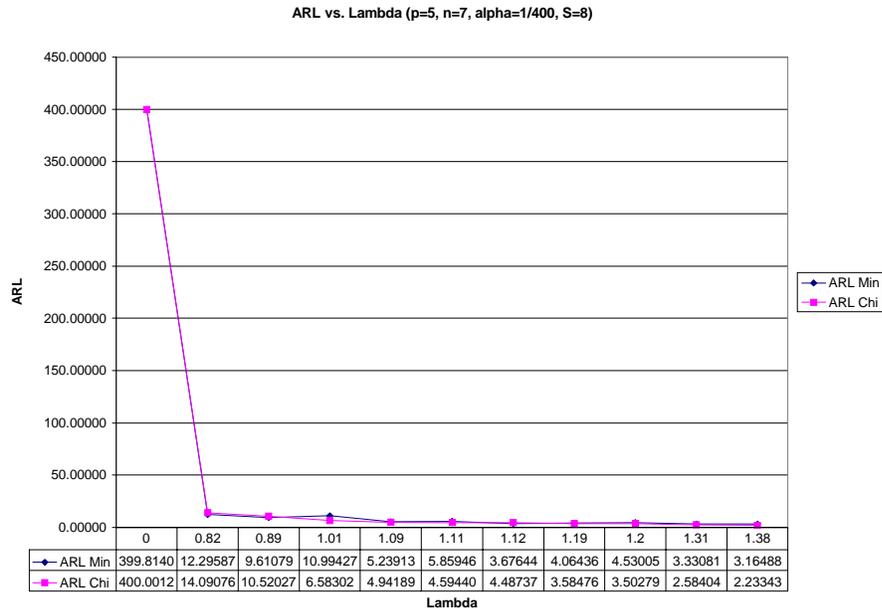
Figure E.12 – ARL vs. Lambda (p=5, n=7, alpha=1/400, S=8)

Lambda	Delta 1	Delta 2	Delta 3	Delta 4	Delta 5	ARL Min	ARL Chi
0	0	0	0	0	0	399.81406	400.00121
0.819952	0.244783	0.213948	0.64282	0.660332	0.367478	12.29587	14.09076
0.888451	0.090834	0.302855	0.527568	0.825629	0.443644	9.61079	10.52027
1.008316	0.422125	0.379199	0.784084	0.480565	0.000713	10.99427	6.58302
1.090001	0.753351	0.252774	0.283036	0.857486	0.687636	5.23913	4.94189
1.112103	0.166659	0.892404	0.25016	0.211758	0.838294	5.85946	4.59440
1.119381	0.642575	0.440069	0.596911	0.523655	0.981353	3.67644	4.48737
1.192452	0.615906	0.964149	0.156574	0.725512	0.603605	4.06436	3.58476
1.200409	0.951494	0.310535	0.0885	0.846328	0.517449	4.53005	3.50279
1.314876	0.995472	0.670197	0.188719	0.896489	0.423547	3.33081	2.58404
1.378239	0.904709	0.990789	0.612578	0.151001	0.642618	3.16488	2.23343

Table E.12 – Data for Figure E.12

ARL vs. Lambda (p=5, n=5, alpha=1/400, S=1 and S=3)
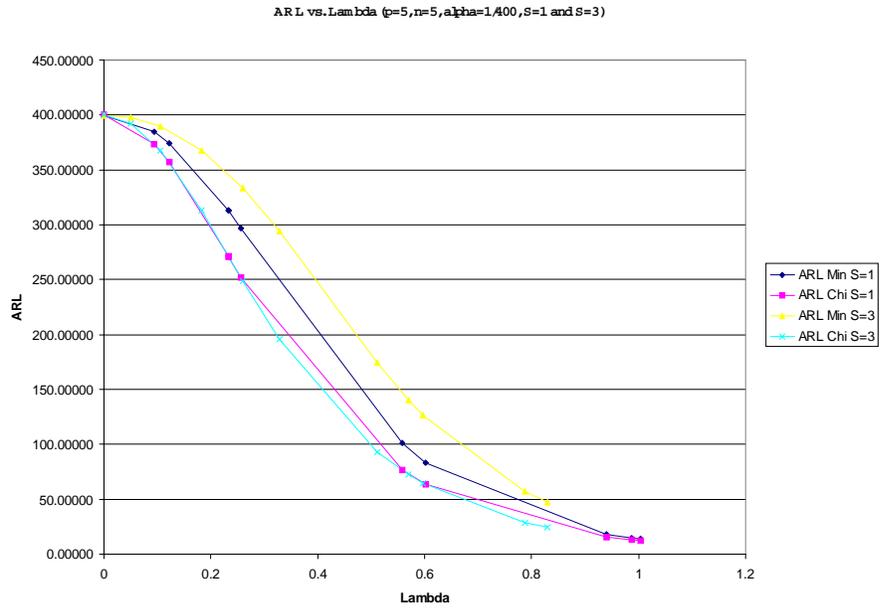
Figure E.13 – Relative Sensitivity to the Identity of the Shift Variable (S=1 & S=3)

**ARL vs. Lambda (p=5, n=7, alpha=1/500, S=1 and S=3)**
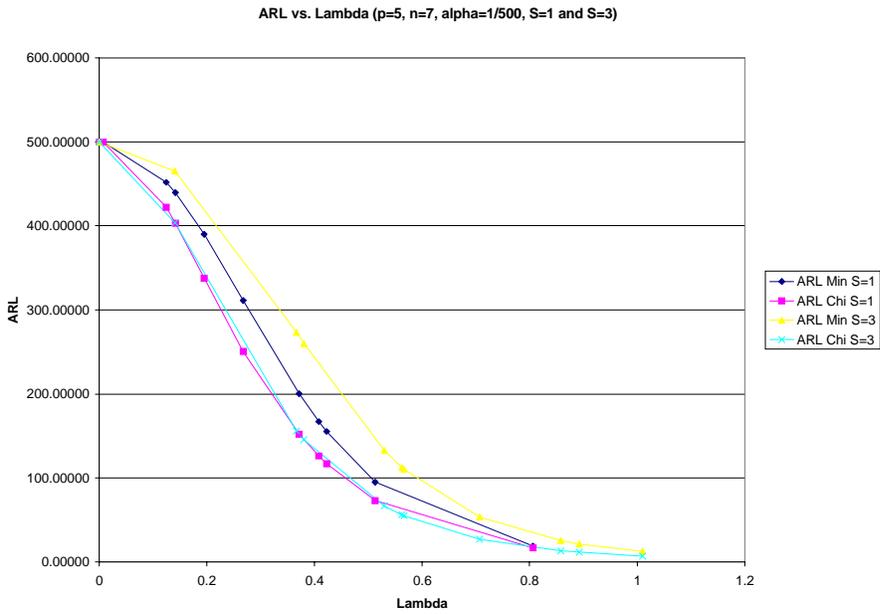
Figure E.14 – Relative Sensitivity to the Identity of the Shift Variable (S=1 & S=3)
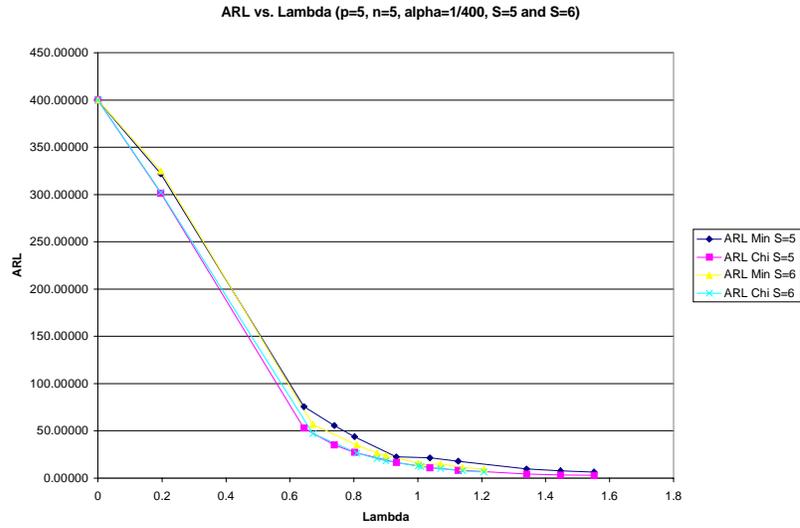
Figure E.15 – Relative Sensitivity to the Identity of the Shift Variable (S=5 & S=6)

	0	0.42	1.1	1.14	1.59	1.69	1.7	2.12	2.57	2.68	3.03
ARL Min	499.9238	219.6031	8.92412	7.68491	2.19475	1.83320	1.80181	1.15757	1.01715	1.00926	1.00078
ARL Chi	499.9999	117.5830	5.22228	4.56995	1.59176	1.39319	1.37612	1.04961	1.00236	1.00099	1.00003
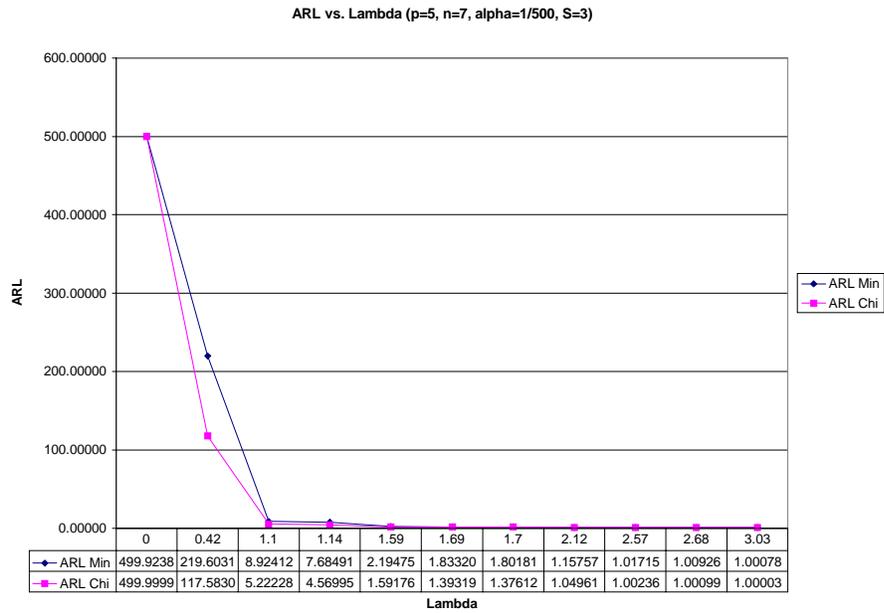
Figure E.16 – ARL vs. Lambda (p=5, n=7, alpha=1/500, S=3)

ARL vs. Lambda (p=5, n=5, alpha=1/200, S=6)

	0	0.67	1.71	1.81	2.05	2.22	2.59	2.59	2.9	2.98	3.81
ARL Min	199.9806	35.01198	1.94050	1.62743	1.23884	1.24517	1.01910	1.04067	1.01700	1.01059	1.00002
ARL Chi	200.0005	28.53637	1.67598	1.48433	1.20038	1.10167	1.01770	1.01738	1.00266	1.00157	1.00000
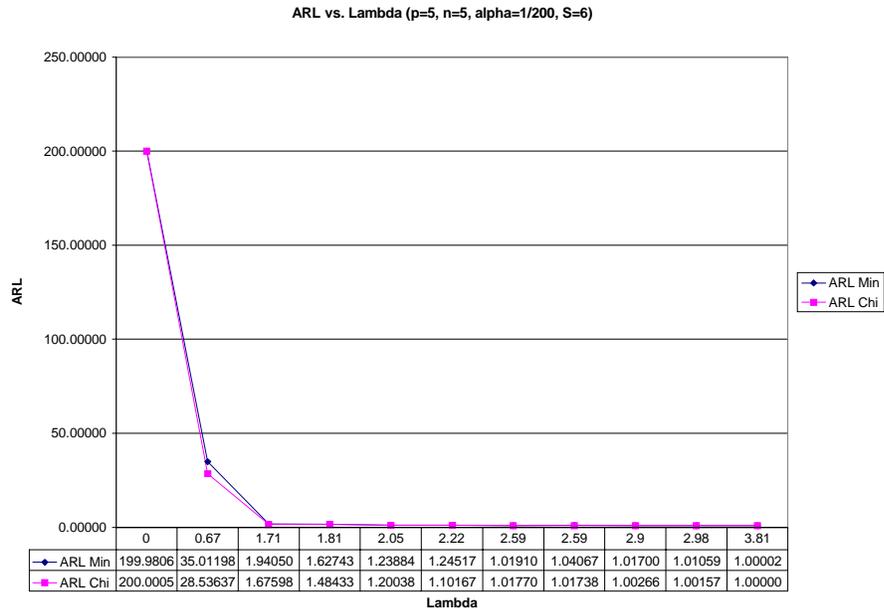
Figure E.17 – ARL vs. Lambda (p=5, n=5, alpha=1/200, S=6)

ARL vs. Lambda (p=8, n=3, alpha=1/200, S=1)

	0	0.14	0.2	0.48	0.51	0.54	0.67	0.78	0.89	0.94	1.01
ARL Min	199.9921	193.5539	187.4079	131.3419	122.9457	116.0559	83.65689	60.41303	43.32595	37.35836	29.85212
ARL Chi	200.0000	188.0092	177.4793	107.1690	99.13802	92.84934	66.01348	48.53888	36.04557	31.66047	26.06440
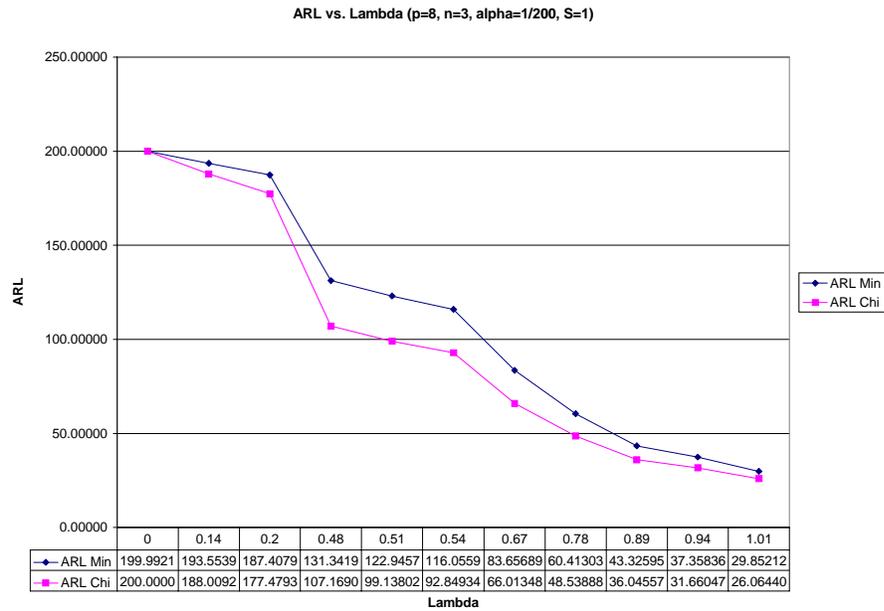
Figure E.18 – ARL vs. Lambda (p=8, n=3, alpha=1/200, S=1)

70

ARL vs. Lambda (p=8, n=5, alpha=1/400, S=2)

	0	0.03	0.18	0.25	0.5	0.58	0.91	0.92	1.33	1.46	1.49
ARL Min	399.92	399.76	395.31	391.15	356.92	340.06	231.53	228.42	98.503	72.612	68.459
ARL Chi	400.00	397.44	334.18	289.69	127.59	95.723	25.872	25.059	6.1651	4.3080	4.0428
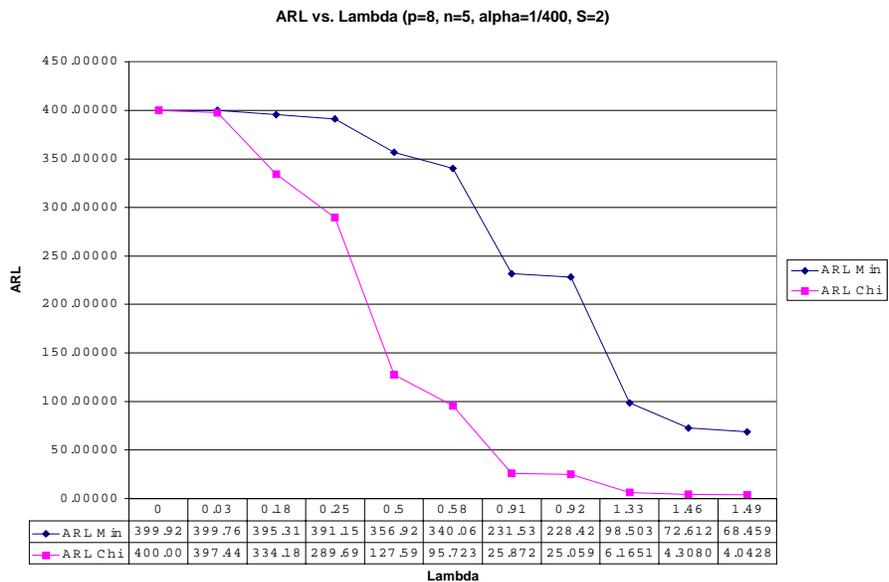
Figure E.19 - ARL vs. Lambda (p=8, n=5, alpha=1/400, S=2)

ARL vs. Lambda (p=8, n=7, alpha=1/500, S=3)

|	0	0.05	0.07	0.35	0.38	0.54	0.82	0.84	1.12	1.2	1.21
ARL Min	499.8350	497.8559	495.3340	357.2230	329.6706	184.0422	50.37914	44.45663	13.19232	9.66476	9.34927
ARL Chi	499.9981	491.5412	481.2141	214.0277	187.3778	87.97268	23.90951	21.24405	6.88620	5.19365	5.04084
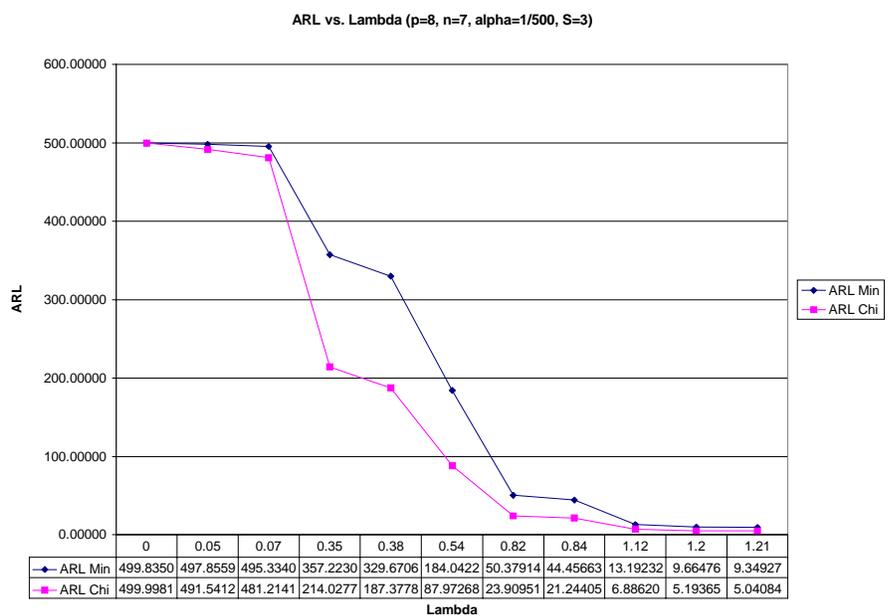
Figure E.20 – ARL vs. Lambda (p=8, n=7, alpha=1/500, S=3)
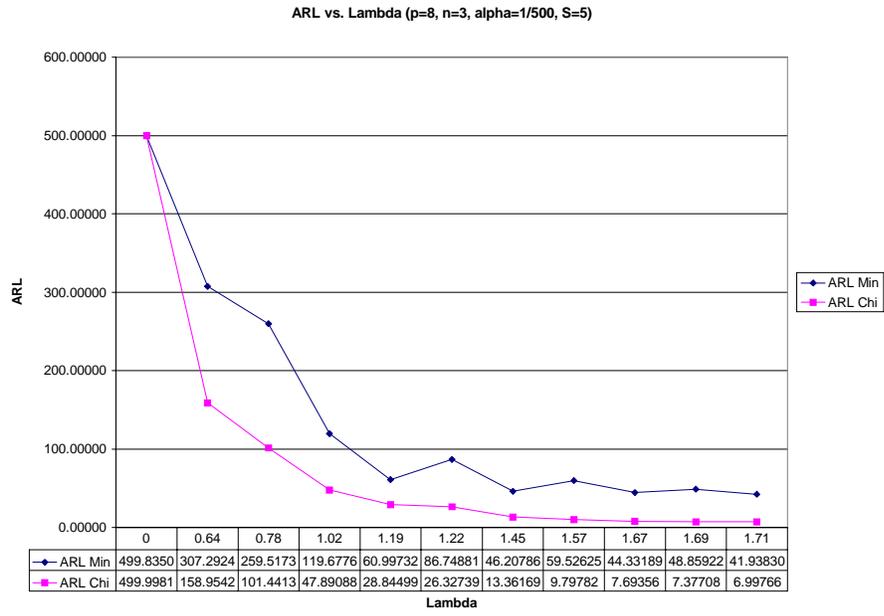
Figure E.21 – ARL vs. Lambda (p=8, n=3, alpha=1/500, S=5)
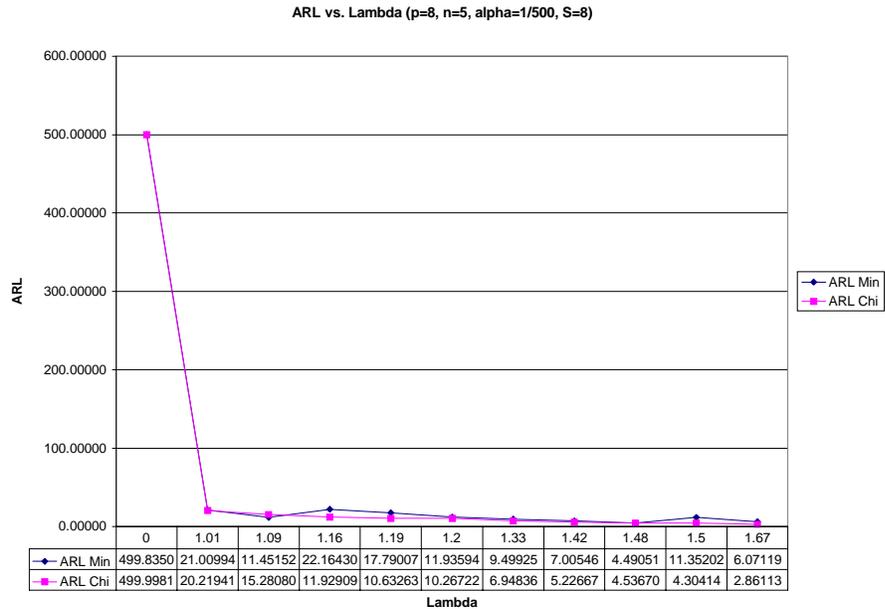
Figure E.22 - ARL vs. Lambda (p=8, n=5, alpha=1/500, S=8)

Lambda	Delta 1	Delta 2	Delta 3	Delta 4	Delta 5	Delta 6	Delta 7	Delta 8	ARL Min	ARL Chi
0	0	0	0	0	0	0	0	0	499.83507	499.99813
0.8962	0.4806	0.1698	0.3186	0.5263	0.6691	0.738	0.5917	0.5825	39.43200	71.08501
0.9766	0.1323	0.3824	0.5936	0.0757	0.1751	0.0095	0.3196	0.5119	113.68568	55.18267
1.1688	0.0814	0.6667	0.9218	0.6219	0.3959	0.3733	0.7134	0.601	26.11148	30.42470
1.1746	0.2882	0.1761	0.2063	0.4084	0.9607	0.2451	0.7492	0.4759	38.00997	29.88712
1.2267	0.1452	0.6681	0.5951	0.3825	0.6293	0.6759	0.9594	0.5319	25.46466	25.55460
1.2885	0.2552	0.3379	0.7379	0.1701	0.0788	0.6901	0.1565	0.7053	54.64782	21.29247
1.3532	0.7601	0.3675	0.7031	0.139	0.1318	0.0024	0.6254	0.5812	43.60939	17.66752
1.3958	0.2146	0.572	0.0715	0.1523	0.5815	0.5986	0.2249	0.9274	45.95656	15.66376
1.648	0.4695	0.5588	0.0635	0.6559	0.9916	0.926	0.0538	0.737	23.55801	8.07434
1.7187	0.0606	0.3647	0.0818	0.3377	0.9732	0.5666	0.0605	0.9851	30.49756	6.81724

Table E.22 – Data for Figure E.22

ARL vs. Lambda (p=8, n=7, alpha=1/500, S=8)

|	0	1.18	1.19	1.37	1.45	1.47	1.48	1.5	1.54	1.69	1.78
ARL Min	499.8350	4.42400	7.16872	9.14573	5.98396	3.79299	4.84540	7.34217	4.96935	3.72609	4.99506
ARL Chi	499.9981	5.71704	5.43471	3.15444	2.61924	2.50176	2.42738	2.29250	2.14887	1.63857	1.44179
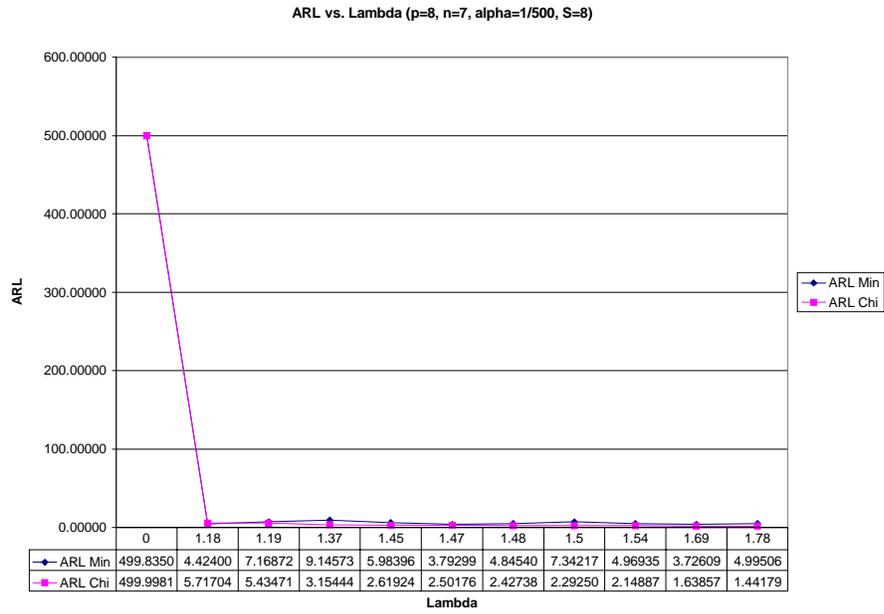
Figure E.23 - ARL vs. Lambda (p=8, n=7, alpha=1/500, S=8)

| Lambda	Delta 1	Delta 2	Delta 3	Delta 4	Delta 5	Delta 6	Delta 7	Delta 8	ARL Min	ARL Chi
0	0	0	0	0	0	0	0	0	499.83507	499.99813
0.8962	0.4806	0.1698	0.3186	0.5263	0.6691	0.738	0.5917	0.5825	39.43200	71.08501
0.9766	0.1323	0.3824	0.5936	0.0757	0.1751	0.0095	0.3196	0.5119	113.68568	55.18267
1.1688	0.0814	0.6667	0.9218	0.6219	0.3959	0.3733	0.7134	0.601	26.11148	30.42470
1.1746	0.2882	0.1761	0.2063	0.4084	0.9607	0.2451	0.7492	0.4759	38.00997	29.88712
1.2267	0.1452	0.6681	0.5951	0.3825	0.6293	0.6759	0.9594	0.5319	25.46466	25.55460
1.2885	0.2552	0.3379	0.7379	0.1701	0.0788	0.6901	0.1565	0.7053	54.64782	21.29247
1.3532	0.7601	0.3675	0.7031	0.139	0.1318	0.0024	0.6254	0.5812	43.60939	17.66752
1.3958	0.2146	0.572	0.0715	0.1523	0.5815	0.5986	0.2249	0.9274	45.95656	15.66376
1.648	0.4695	0.5588	0.0635	0.6559	0.9916	0.926	0.0538	0.737	23.55801	8.07434
1.7187	0.0606	0.3647	0.0818	0.3377	0.9732	0.5666	0.0605	0.9851	30.49756	6.81724

Table E.23 – Data for Figure E.23
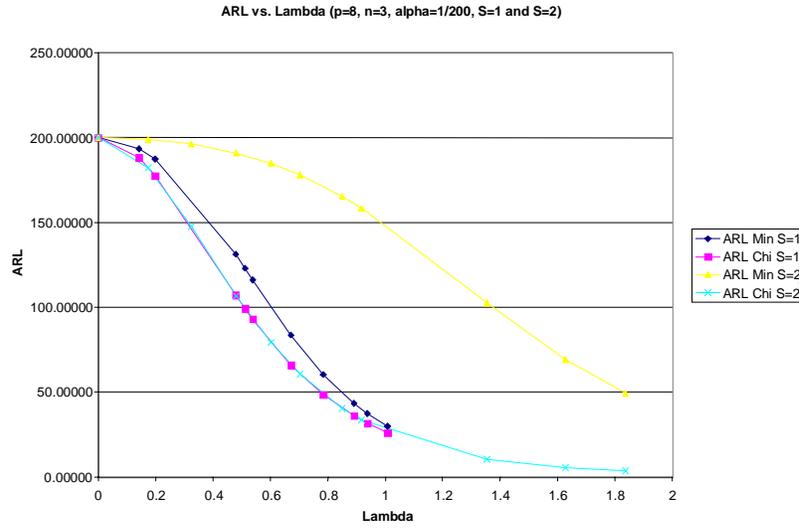
Figure E.24 – Relative Sensitivity to the Shift Variable (S=1 & S=3)
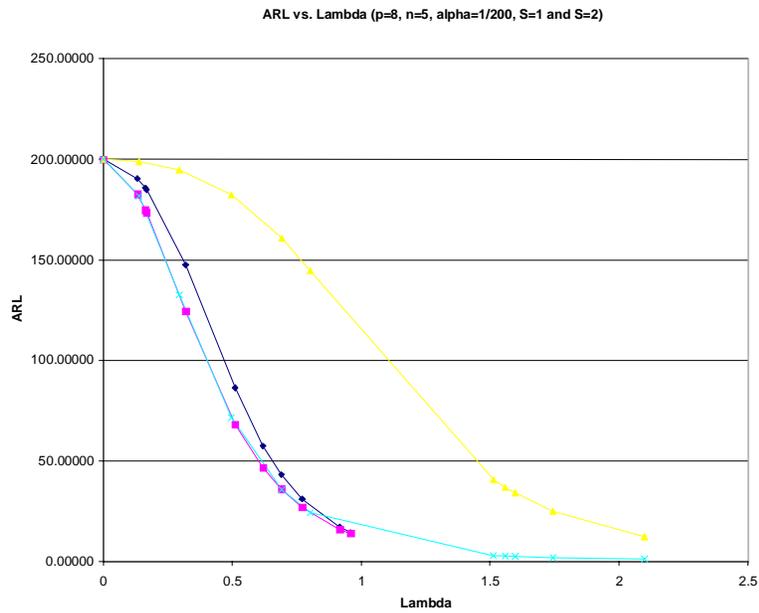
Figure E.25 – Relative Sensitivity to the Identity of the Shift Variable (S=1 & S=3)
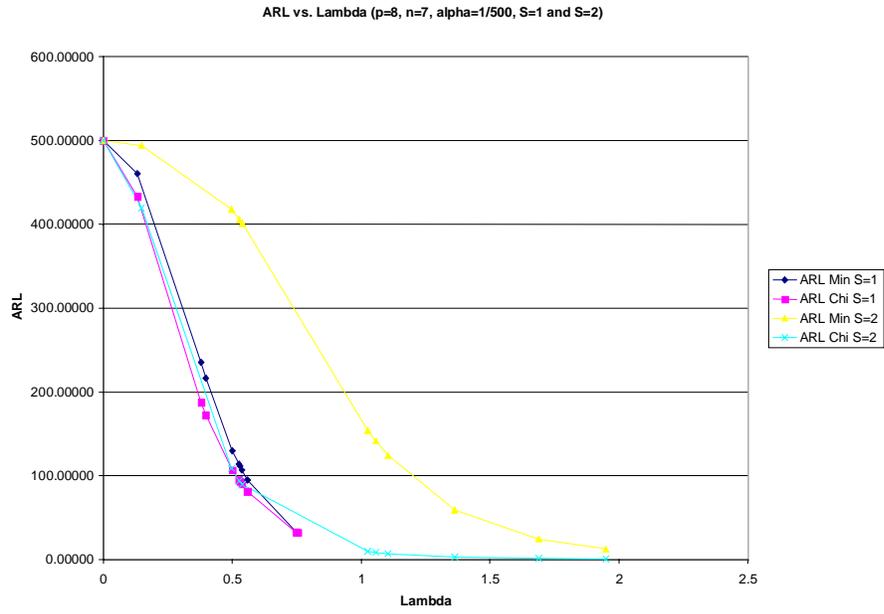
Figure E.26 – Relative Sensitivity to the Identity of the Shift Variable (S=1 & S=3)
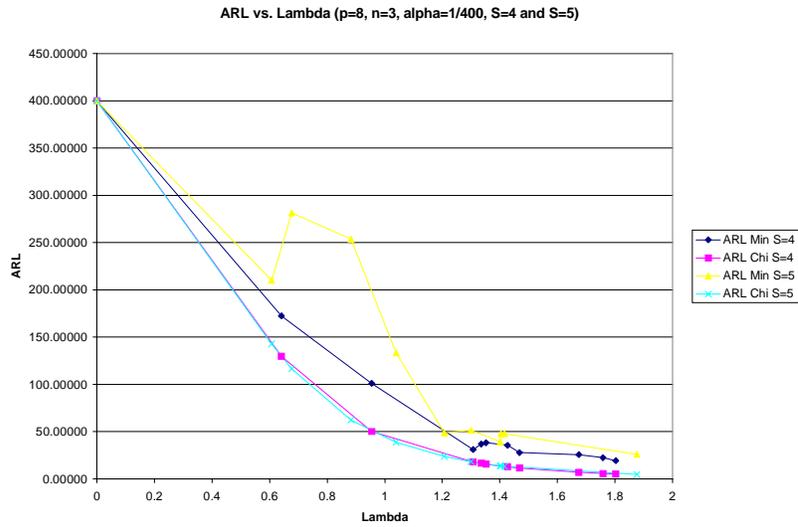

Figure E.27 – Relative Sensitivity to the Identity of the Shift Variable (S=4 & S=5)

ARL vs. Lambda (p=8, n=7, alpha=1/500, S=1)

	0	0.4	1.14	1.19	1.51	1.58	1.59	1.61	1.68	2.24	2.26
ARL Min	499.8350	214.1537	5.59640	4.56429	1.98351	1.72649	1.69099	1.62991	1.48423	1.03032	1.02718
ARL Chi	499.9981	170.6197	6.52872	5.36038	2.29159	1.97002	1.92524	1.84797	1.66223	1.05169	1.04678
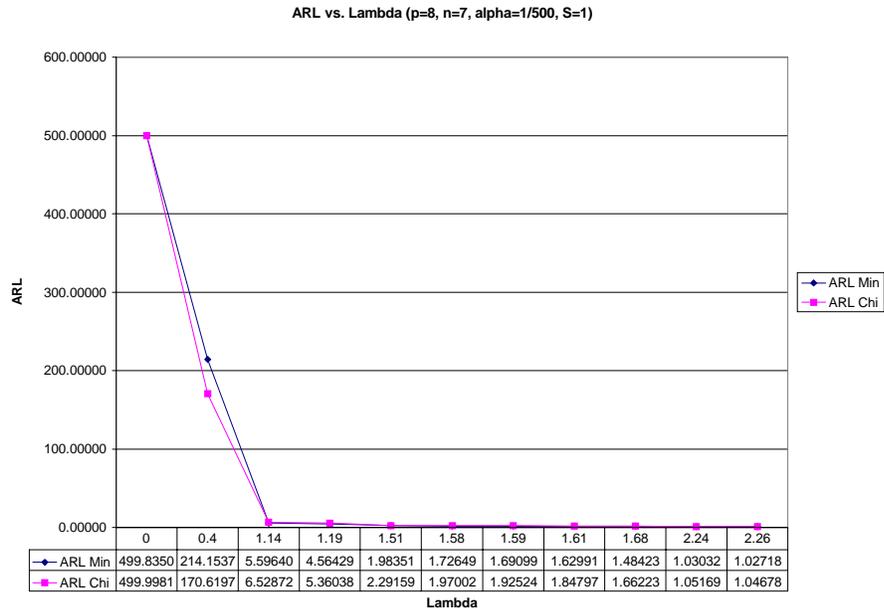
Figure E.28 – ARL vs. Lambda (p=8, n=7, alpha=1/500, S=1)

ARL vs. Lambda (p=8, n=7, alpha=1/200, S=3)

|	0	0.06	0.29	0.72	2.17	2.4	2.41	3.4	3.48	3.58	3.59
ARL Min	199.9875	198.7047	167.5521	43.15055	1.18322	1.07038	1.06786	1.00013	1.00007	1.00003	1.00002
ARL Chi	200.0000	194.3360	116.1365	20.45108	1.04361	1.01058	1.01002	1.00000	1.00000	1.00000	1.00000
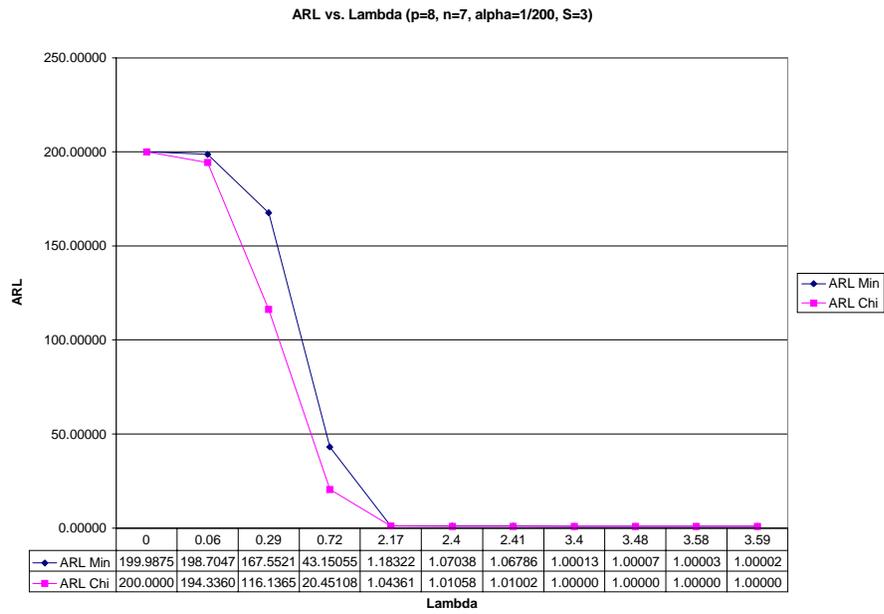
Figure E.29 – ARL vs. Lambda (p=8, n=7, alpha=1/200, S=3)

77

Vita

Ian Jon Rehmert

Ian Jon Rehmert was born to John Carroll Rehmert and Linda Gertrude Middlestadt in Baltimore, Maryland on September 21, 1973. Ian received his Bachelor of Science Cum Laude in Industrial and Systems Engineering from Virginia Polytechnic Institute and State University in Blacksburg, Virginia on May 13, 1995. During the completion of his undergraduate degree at Virginia Tech, he was inducted into Alpha Pi Mu and was a four-year member of the Virginia Tech Lacrosse Team. Ian will graduate with his Master of Science in Industrial and Systems Engineering with a concentration in Operations Research from Virginia Polytechnic Institute and State University in December of 1997. While pursuing his Masters degree, he held an internship position with the Operations Research department of USAir and a part-time position with the Operational Evaluation Division of the Institute for Defense Analyses. Ian will continue his studies in Blacksburg at Virginia Tech where he will pursue his Ph.D. in Industrial and Systems Engineering concentrating in Operations Research. He is a member of the Institute for Operations Research and the Management Sciences.