Proceedings of the 1991 Winter Simulation Conference
Barry L. Nelson, W. David Kelton, Gordon M. Clark (eds.)

MODIFICATIONS OF THE NELDER-MEAD SIMPLEX METHOD FOR STOCHASTIC SIMULATION RESPONSE OPTIMIZATION

Russell R. Barton
John S. Ivey, Jr.

Department of Industrial and Management Systems Engineering
The Pennsylvania State University
University Park, Pennsylvania 16802

ABSTRACT

The Nelder-Mead simplex method has been used extensively for the optimization of simulation models for over twenty five years. There have been over 2,000 citations to the original paper, with 200 citations in 1989 alone. When used to optimize a response function of a discrete event simulation model, the objective function is usually stochastic, due to the random behavior that is being modeled. The Nelder-Mead rescaling and shrinking steps make it sensitive to random variations in the response function values, and introduce risks of false convergence on stochastic functions. We give analytical and empirical evidence that characterizes false convergence on stochastic functions, and discuss several modifications to reduce the likelihood of false convergence. We describe a computational comparison of several modifications when used to optimize simple stochastic functions of two and ten variables.

1 INTRODUCTION

The Nelder-Mead simplex method (1965) has been used extensively for the optimization of simulation models for over twenty five years. There have been over 2,000 citations to the original paper during this period, with 200 citations in 1989 alone. Most of these citations document the successful application of the technique to a practical problem. The range of application is very broad: researchers in analytical chemistry (Ash *et. al.*, 1989), neurology (Nieminen, Suarez-Isla, and Rapoport, 1988), fishery management (Schnute and Sibert, 1983), and fusion technology (Meier and Morse, 1985) have all made use of the algorithm. When used to optimize some response of a discrete event simulation model, the objective function is usually stochastic, due to the random behavior that is being modeled. This optimization may be based on response functions that are calculated at the end of each simulation run, or the optimization might be based on batch means within a single run, as in Leung and Suri (1990).

This algorithm is based on an earlier "simplex" sequential search strategy developed by Spendley, Hext, and Himsworth (1962). While the precursor was intended for stochastic functions, Nelder-Mead was intended for deterministic functions. The Nelder-Mead rescaling and shrinking modifications make it more sensitive to random variations in the response function values, and introduce new risks of false convergence on stochastic functions.

In Section 2, we give a brief description of the original Nelder-Mead method. In Section 3 we give analytical and empirical evidence that characterizes false convergence on stochastic functions, and discuss several modifications to reduce the likelihood of false convergence. In Section 4 we describe a computational comparison of several modifications used to optimize simple stochastic functions of two and ten variables.

2 THE NELDER-MEAD STRATEGY

2.1 Origin

Nelder and Mead's algorithm was based on the earlier work of Spendley, Hext, and Himsworth (1962), who originated the concept of simplex optimization. For a function of n parameters, the algorithm maintains a set of n+1 function values evaluated at n+1 points in parameter space. This set of points defines a simplex in n dimensions. In two dimensions, the simplex would be a triangle; in three, a tetrahedron. The Spendley *et. al.* algorithm incorporates a regular simplex (i.e., all sides have the same length) which does not vary in size. The function is evaluated at each point of the simplex. The simplex then moves toward the optimum by reflecting the point with the worst function value through the centroid (average) of the remaining n points. In two dimensions, this can be visualized as flipping over a triangle to move it down a hill.

To insure progress, two additional rules apply. If any point of the simplex is retained for n+1 reflections, a new observation is taken at that point to replace the current observation. This rule ensures that the simplex does not remain at a point whose function value is unusually low due to experimental error. To simplify subsequent discussion, this rule will be termed the 'n+1 rule'. The other rule specifies that if a reflected point is still the worst of the simplex, this point is not to be accepted. Instead, the next-to-worst point is reflected through the other points. For discussion purposes, this will be called the next-to-worst rule. This rule causes the Spendley *et. al.* simplex to continually cycle around the region containing the optimum.

This continual cycling, in place of any stopping criterion, was motivated by the concept of evolutionary operation (EVOP), for which Spendley *et. al.* developed the simplex method. Their objective was to optimize an ongoing manufacturing process, such as the operation of production equipment. In such an environment, the optimal operating conditions are likely to change over time, due to tool wear, material variation, and so forth. A constantly moving simplex can pick up these slight variations as they occur and follow the optimum conditions as they shift.

The fixed-size simplex method has several limitations, as pointed out by Deming and Morgan (1973):

1. Equilateral triangles are the only regular simplices which have the ability to close pack. For problems of higher dimension, this means that arrival at the optimum region is not clearly signalled.
2. There is no provision for acceleration toward the optimum.
3. It is possible for the fixed-size simplex to converge to a non-optimal ridge.

By allowing the simplex to change its shape, Nelder and Mead's algorithm (1965) avoids these shortcomings. The shape changes take place through additional geometric operations which provide clear "convergence" toward the optimum, and acceleration toward this optimum. As a whole, the extra operations allow the simplex to adapt to the form of the region it explores, making the method less susceptible to false convergence on deterministic functions.

2.2 Nelder-Mead Rescaling Modifications

Nelder and Mead's simplex method for minimization (1965) incorporates operations to rescale the simplex based on the local behavior of the function. Simplex reflections are expanded in the same direction if the reflected value is particularly good. A poor value results in a contraction. If the function value at the contracted point suggests that the function is not convex, the size of the simplex is reduced. The original Nelder-Mead rules for implementing these added operations are outlined below.

1 Initialization For a function of dimension n, choose n+1 points. Evaluate the function at each point (vertex) of the simplex.

2 Stopping Criterion Iterations continue until the standard error of the function values at the n+1 simplex points falls below a particular value. Nelder and Mead used 10^{-8} as a stopping criterion. (They did not have stochastic functions in mind.)

3 Reflection At the start of each iteration, identify the vertices where highest, second highest, and and lowest function values occur. Let P_H, P_{2H}, P_L denote these points; and let F_H, F_{2H}, F_L represent the corresponding function values. Find C, the centroid of all points other than P_H. Generate a new point P_R by reflecting P_H through C. Reflection is carried out according to the following equation:

$$P_R = (1 + \alpha)C - \alpha P_H$$

where α is a positive constant called the reflection coefficient. The next operation is determined by F_R, the value of the function at P_R.

4a Reflection If $F_L < F_R < F_{2H}$, then P_R replaces P_H in the simplex, and a new iteration begins (step 2 above).

4b Expansion If $F_R < F_L$, the simplex expands, in the hope that more improvement will result from moving further in the same direction:

$$P_E = \gamma P_R + (1 - \gamma)C$$

where γ is the expansion coefficient. Gamma must be greater than 1. If F_E is less than F_L, P_E replaces P_H in the simplex; otherwise, P_R replaces P_H. The next iteration begins with the new simplex.

4c Contraction If the reflected point is still worse than every point in the simplex, (i.e. $F_R > F_H$ or $F_R > F_{2H}$) the simplex contracts, under the assumption that the current move was too far. Contraction generates a point closer to C, on the side which holds the most promise: if $F_R < F_H$, the contracted point lies between C and P_R; if $F_H < F_R$, the contracted point lies between C and P_H. Contraction is defined by the equation:

$$P_C = \beta P\_ + (1 - \beta)C$$

where P_ is either P_H or P_R (whichever has the lowest function value), and ß is the contraction coefficient, a number between 0 and 1. If $F_C < F\_$, P_C replaces P_H and a new iteration begins.

Shrink If $F_C > F\_$, the contraction has failed, and the entire simplex shrinks by the parameter δ, retaining only P_L. This is done by replacing each point P_i by

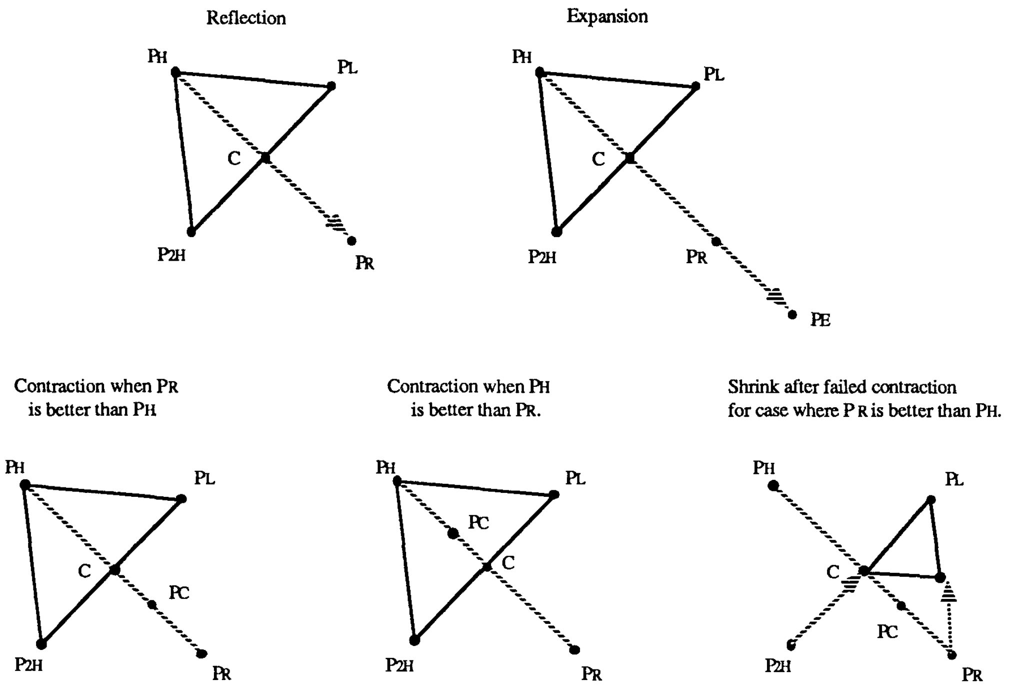
$$P_i = \delta P_i + (1 - \delta)P_L$$

The algorithm then continues with the next iteration.

These steps are illustrated in Figure 1 for a function of two parameters. Nelder and Mead recommended the use of $\alpha = 1$, $\beta = .5$, $\gamma = 2$ based on their tests. They did not identify δ as a parameter; instead, they employed a fixed value of 0.5.

3 FALSE CONVERGENCE ON STOCHASTIC FUNCTIONS

The Nelder-Mead algorithm is widely used for simulation optimization, where the functions it optimizes



Reflection

Expansion

Contraction when PR is better than PH

Contraction when PH is better than PR.

Shrink after failed contraction for case where P R is better than PH.

Figure 1: Nelder-Mead Rescaling Operations

are often subject to random noise. The algorithm is robust to small inaccuracies or stochastic perturbations in function values. This is because the method uses only the ranks of the function values to determine the next move, not the function values themselves. Perturbations that do not change the ranks of the values will have no effect on the algorithm's search trajectory.

If this noise is substantial, it will lead to inappropriate rescaling operations, resulting in false convergence. Empirically, this problem often manifests itself as inappropriate shrink steps. Once begun, this reduction in the simplex size can reduce the variance of the simplex function values below the system's inherent variability before the optimum region has been reached.

3.1 Mechanisms for False Convergence

In order to understand how stochastic noise affects the simplex, a closer examination of its movement rules is necessary. Of particular interest is what happens when the simplex is in a region that is fairly constant relative to its size. This will occur after a sequence of shrink and/or contract operations, or when the neighborhood of the minimum has been reached. In either case, much of the perceived difference between function values is dictated by the noise. Consequently, the noise will also dominate the movement of the simplex and the choice of rescaling option.

To see how this might happen, consider the example in Figure 2. The graph of this function would ordinarily be a simple contour map; however, noise blurs the distinction between contours. For example, stochastic variation could cause points sampled from the 375 contour to have function values well above or below 375, depending on the variance of the response. Based on these contours, the simplex should progress downward in the figure. Suppose the current simplex is ABD. Point A has the worst response value, so a reflection is made to point G, where noise causes an unnaturally high value of 380, say. Thus point G will be rejected in favor of a point farther from the optimum: contraction to point H takes place, where a value of 400 is obtained, say. Since this response is worse than all current points, the simplex must now shrink, giving the new simplex BIJ. At this point there is little hope for progress: the simplex is most likely to converge to a point with expected value near 375.

In this example, a reflection that should have been accepted was rejected. In addition, a contraction that should have been accepted wasn't, resulting in an inappropriate shrink. We would like to know how frequently such events might occur in practice. That is, when confronted with a region where function differences are small relative to the stochastic component, how does the simplex behave? These questions can be addressed by studying the behavior of the algorithm on a constant-valued function, where the differences between vertices are

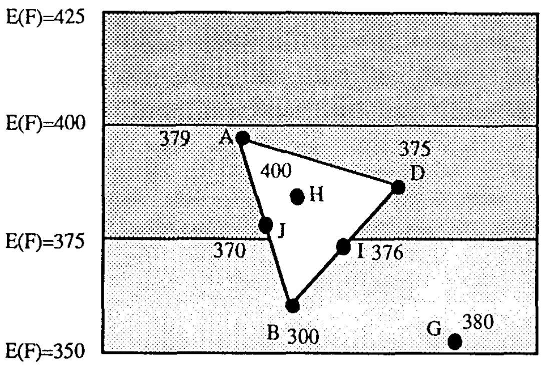


Figure 2: An Example Where Function Variations are Dominated by Noise

entirely due to noise. The path the simplex takes can be described in terms of the state transition diagram shown in Figure 3. We characterize this behavior by finding the probabilities with which it takes each branch, both initially and over the long run.

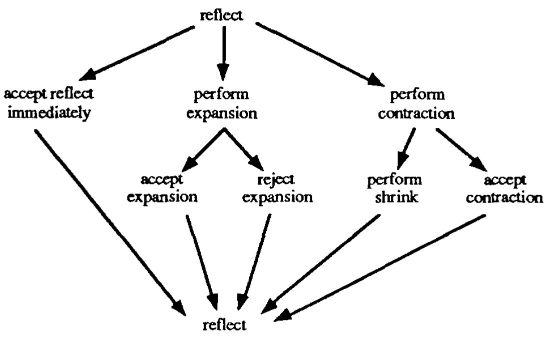


Figure 3: Step Transitions for the Nelder-Mead Algorithm

3.2 Step Transition Probabilities when Noise Dominates

Consider the case where the simulation response function can be modeled as:

$$F(x) = f(x) + W(x)$$

where W is the stochastic component. For the discussion below we will assume that the distribution of W values are independent and do not depend on x, and that f is constant over x. In this case the initial simplex is a set of independent samples from an identical distribution (i.i.d.). The reflected point is also an independent sample. This allows a very simple analytical derivation of the first iteration step transition probabilities. To simplify the discussion, we define the following events:

R = the reflected point is accepted immediately
E = an expansion is attempted
$E \cap A$ = an expansion step is attempted and accepted
C = a contraction is attempted
$C \cap S$ = a contraction fails and a shrink is performed

Consider a two dimensional problem. We have four function values that are i.i.d., one of which belongs to the reflected point. The next action taken is determined by its ranking relative to the other points. The probability that the reflected point has the best value is 1/4. By the same reasoning, the probabilities that it has the worst value, or the second worst value, or the second best value are all equal to 1/4. A contraction will be performed if the reflected point is the worst or the second worst of the four. Thus, P(C) = 1/4 + 1/4 = 1/2. Expansion is attempted when the reflected point is the best, i.e. P(E) = 1/4. By definition, P(R) + P(E) + P(C) = 1, so P(R) = 1/4.

This type of argument can be extended to find the probability that, on the first iteration with a constant function, a shrink will be performed. For a two dimensional problem, we must consider an i.i.d. sampling of 5 points and enumerate all possible rankings of F_R and F_C. Summing the probabilities of rankings that lead to a shrink step gives $P(S \cap C) = (1/5) * (0 + 0 + 1/2 + 1/4 + 1/4) = 0.2$. Hence, the conditional probability of shrinking given that a contraction is performed is $P(S|C) = P(S \cap C) / P(C) = .2 / .5 = .4$ A similar analysis shows that $P(E \cap A) = .1$ and $P(A|E) = .4$. It is interesting to note that these results require only that the noise be i.i.d..

In order to verify this analysis, a computational experiment was performed. The algorithm was run 5000 times on a constant function with a uniform noise distribution ranging from 0 to 1. The algorithm was stopped after only one iteration, and the number of times each operation took place was tallied. The results were in close agreement with the theoretical calculations, as shown in the table below.

Table 1: First Iteration Transition Probabilities

EVENT	P(EVENT)	
	THEORETICAL	EMPIRICAL
R	.25	.2544
E	.25	.2574
$E \cap A$	.10	.1058
C	.50	.4882
$C \cap S$	.20	.2006

When a two-dimensional simplex first reaches a region of constant expected function values, it is twice as likely to reduce in size as it is to expand. These results will hold approximately when the product of the expected function value gradient and the size of the simplex is small compared to the standard deviation of the stochastic component. This difficulty is associated primarily with functions of only a few variables. In higher dimensions, the simplex is much more likely to accept a reflection: for a function of 50 parameters, the probability of performing a reflection is 49/52. The chance of performing an expansion drops to 1/52, and the chance of performing a contraction to 2/52.

This first step information gives some insight into the nature of the difficulty, but it is only a small part of picture. The transition probabilities change very rapidly after the first iteration on the constant function, due to the change in distribution of function values in the current simplex. Bad values are repeatedly discarded in favor of good ones, and the simplex no longer contains n+1 identically distributed observations. Rather, it contains the best points found to date. As better and better values are retained, it is much less likely that any newly sampled point will be better than those in the current simplex. This means that the number of expansions should decrease as iterations continue, regardless of the dimension of the function. On the other hand, contraction and shrinkage should increase as the simplex values become harder to improve upon.

Empirical studies were performed on functions of 2 and 50 dimensions with Gaussian and uniform stochastic components. For the two parameter constant expected value function, the algorithm was run for 200 function evaluations beyond the setup of the initial simplex. The number of operations was then tallied in 14 groups of five iterations (reflection to next reflection). Tests in 50 dimensions were run for 2000 function evaluations, and tallied in 9 groups of 50 iterations. Six replications each were performed using two different noise distributions: uniform (0 to 1), and standard Gaussian truncated to ±3. The 3 sample moving averages for each transition probability are plotted in Figures 4 and 5 for the uniform (0 to 1) case. The bold characters at the left of each figure mark the first iteration transition probabilities.

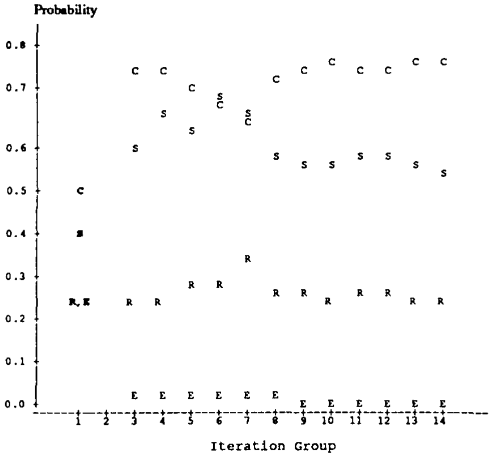


Figure 4: Moving Average Step Transition Probabilities, Two Parameter Function

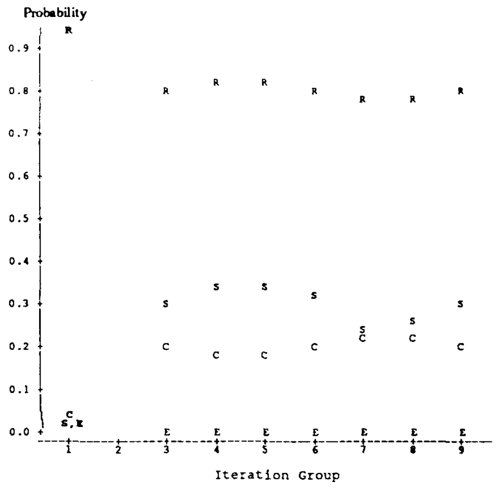


Figure 5: Moving Average Step Transition Probabilities, Fifty Parameter Function

Only the uniform results are shown. Since step transitions are based on the function value ranks, the long term transition probabilities will again be independent of the form of the noise distribution. This was supported by the empirical findings.

There are rapid increases in the probability of contraction and the conditional probability of shrinking given a contraction. After the initial transient, the probabilities remain fairly stable. The difference between the 2 dimensional and 50 dimensional plots highlights the increased resistance to contract and shrink steps in higher dimensions. Even for the fifty parameter function, though, the contract and shrink probabilities are substantially larger than the first iteration values.

3.3 Modifications to reduce the likelihood of false convergence

These results indicate that when noise begins to dominate differences in function values, the size of the simplex is much more likely to decrease than increase, especially in lower dimensions. The likelihood of size reduction also increases dramatically after the first iteration. Some bias toward reduction is clearly necessary for convergence, but an excessive bias increases the chance of false convergence. An effective modification for noise should be able to reduce the probability of contracting and/or shrinking without seriously hampering progress toward the optimum.

Previous approaches have addressed this problem by periodically replicating good points and/or providing alternatives to shrinkage. In using Nelder/Mead to optimize the resolution of a magnetic resonance spectrometer, Ernst (1968) made two useful observations on the difficulties described above:

1. It may happen that the simplex reduces in size

too fast, such that it loses the ability to move. Shrink steps cannot occur on a truly concave surface. Their occurrence is usually caused by erroneous measurements, in which case the size of the simplex should not be reduced.

2. From the logical scheme it is apparent that the lowest value is never replaced. If it was produced by an inaccurate measurement, it may mislead a long series of subsequent steps.

In applying Nelder/Mead to a discrete event simulation model, these two problems pose a major concern. Many experimenters have proposed modifications to the algorithm that attempt to overcome each of them.

The problem of shrinkage appears to be particularly serious, given its potential to rapidly terminate the optimization. For this reason, some authors completely eliminate the use of a shrink step. Ernst (1968) recommended that a failed contraction be followed by a translation of the entire simplex such that it is centered around the location of the current best point. In addition to translation, Deming and Parker (1978) suggest another alternative to shrinking, originally proposed by King (1974): if the contracted point is the worst point of the new simplex, keep it anyway and reflect the second worst point of the new simplex. Note that this is simply the next-to-worst rule from Spendley, Hext, and Himsworth (1962) applied in a different context.

Attempts to solve the second problem, retention of a spuriously low response, all focus on periodic replication of the best point. Ernst (1968), and Deming and Parker (1978) recommend the use of Spendley *et. al.*'s 'n+1 rule' to avoid false convergence. An extension of this rule is implemented in SIMPLEX-V™, a software package for design of experiments (Nachtsheim, 1987). This extension, the 'n+j rule', allows the user to specify j, where $0 \leq j \leq 9$; thus controlling the retention time of a good response.

Replication of all points appears to have no value. Deming and Morgan (1973) point out that if the function values differ greatly compared to the experimental error, the simplex will move in the right direction, making replication a waste of time. They also argue that the simplex will correct its path even if it is temporarily thrown off course due to error. In an empirical investigation Spendley, Hext, and Himsworth (1962) concluded that replication of points seriously hinders the progress of the simplex.

Knowledge of how the simplex behaves on a constant function points to simple modifications that can address the heavy bias toward contraction and shrinkage more directly. The next section presents an empirical study of three modifications that are specifically designed to reduce P(C) and P(S). The first modification is to resample the lowest point during a shrink step. This brings the simplex back to an initial state condition, restoring the transition probabilities to their first iteration values. This should have the greatest effect on low dimensional functions, since the probability of reaching the shrink step is much higher. Unfortunately, the beneficial effect will be short-lived, based on the empirical findings in Figures 4 and 5.

The second modification is to reduce the probability that a contraction is performed, which would in turn reduce the probability of shrinkage. One way to accomplish this is to resample both P_R and P_{2H} before attempting a contraction. The points are compared again, without checking to see if the old P_{2H} has changed in rank. The contraction is performed only if the new F_R is still worse than the new F_{2H}. This modification will also have less impact in higher dimensions, since a contraction is attempted much less frequently.

The third modification we consider is used in conjunction with the first, and is an attempt to delay the onset of 'constant function behavior'. Rather than attempt to affect the probability of taking a shrink step, the simplex will be reduced to a lesser extent. The approach tested here is to increase the shrink coefficient δ from .5 to .75 and .9. An alternative would be to use a δ that depended on iteration count, k, for example $\delta_k = k/(k+1)$.

4 COMPUTATIONAL EXPERIMENTS

These three modifications were tested on simple quadratic functions with truncated ($\pm 3\sigma$) Gaussian perturbations. As mentioned earlier, since all decisions in the Nelder-Mead algorithm and the proposed modifications are based on ranks, the behavior near the minimum will be essentially the same for any i.i.d. perturbations. The Gaussian perturbations are appropriate models for discrete event dynamic simulation models, where the response function is often an average of many random variables.

4.1 Experiment Design

We apply these perturbations to simple quadratic test functions of the form $f(x) = x^TQx$, where Q is composed of a diagonal matrix with entries (1,1,...,q) and an orthogonal rotation to move the function contour ellipsoid axes away from the coordinate axes. A quadratic approximation to the expected response function is appropriate for this study, since the main concern is premature convergence in the neighborhood of an optimum. Two values of q were chosen. Test runs with q = 1 evaluate the ability of the algorithm to minimize a well behaved function. Test runs with q = 10 offer contours with a ridge, where the rescaling operations of Nelder-Mead are important for convergence.

The Nelder-Mead variants that were tested were:

O	unmodified original algorithm
A	resample the best value at each shrink step
B	resample the second worst and the contraction vertex before contracting
7 5	shrink coefficient δ = .75 rather than .5
9 0	shrink coefficient δ = .90 rather than .5

Modification A has no negative effect on performance other than requiring one extra function evaluation (simulation run). For this reason, the last three modifications were tested in conjunction with A. Modification B was also tested individually.

Each modification was tested on quadratic functions of two and ten parameters with q = 1 and q = 10, for four functions in all. For each function there were six starting points, and five replications for each starting point, for a total of 120 runs per modification. Total function evaluations were limited to 200 for the two-parameter functions and to 1000 for the ten-parameter functions. Selecting a convergence criterion was not critical for analyzing the results; once the simplex was reduced to a certain size, convergence was rapid for all methods, and the underlying quadratic function value was stable for many iterations before the function evaluation limit was reached.

4.2 Results

Each modification was successful in reducing the fraction of contract and shrink steps. These probabilities are summarized in Table 2 with separate probabilities for the two-parameter and ten-parameter functions. Columns with two modification labels had both modifications implemented together.

Table 2: Step Transition Probabilities for Nelder-Mead Modifications

2-PARAMETER FUNCTIONS

	O	A	A.75	A.90	B	AB
R	.22	.16	.16	.16	.53	.43
E	.05	.20	.20	.19	.06	.15
C	.72	.64	.65	.65	.41	.41
S |C	.53	.45	.44	.44	.37	.35
f/f_0	1.0	.98	.62	.62	.76	.78

10-PARAMETER FUNCTIONS

	O	A	A.75	A.90	B	AB
R	.59	.58	.58	.58	.70	.68
E	.07	.11	.11	.11	.08	.11
C	.33	.30	.31	.31	.22	.20
S |C	.24	.25	.25	.24	.26	.25
f/f_0	1.0	.87	.76	.72	1.1	.95

Modification B produced the greatest decrease in contraction probability, which was its purpose. Also as expected, modification A significantly increased the probability of taking an expansion step (not necessarily accepting it). In all cases, however, the probability of taking a contraction step and the conditional probability of a shrink step are still high.

The last row in the table shows the geometric average improvement in function reduction expressed as a ratio. Since these functions have an optimal expected value of zero, an f/f_0 value of .7 means that the revised method yielded an optimal expected value at convergence that was 70% of the original method's value, on average. While all modifications result in some average improvement, the amount is small. This value is misleading, however, because the modifications guard against early false convergence, which does not happen in every case. Many of the comparisons result in ties, reducing the average contribution in the table above.

The boxplots of $\log_{10}(f/f_0)$ in Figures 6-9 show the dispersion of values about the mean. The box indicates the middle 50% of the data. The horizontal line in the center of the box gives the median. The '+' gives the average. The boxplot at the far right for the original Nelder-Mead has no width, since the log ratio is always zero.

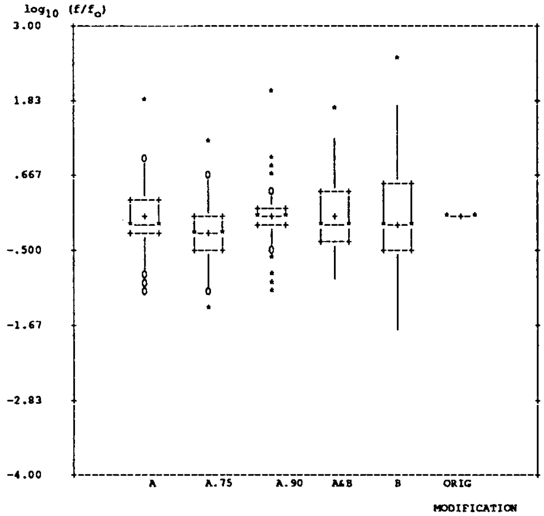


Figure 6: Boxplots of log Function Reduction for Two-Parameter Quadratic, q=1

All of the modifications occasionally make dramatic improvements over the original algorithm. In general we see the best performance for the two methods which change the shrink coefficient, A.75 and A.90. In many of the runs the expected function value was reduced by 50% or more, and rarely was the function value worse by a similar amount. The contraction resample modifications, B and A&B, have high variability and occasionally lead to significantly higher values.

The performance of the contraction resampling modification (B) was disappointing. Based on studies with deterministic functions, the contraction step is often invoked by Nelder-Mead, while the shrink step is rarely taken. This suggests that shrink step modifications are free, in the sense that they do not affect progress in the region where the function variation dominates the stochastic component. On the other hand, modifications

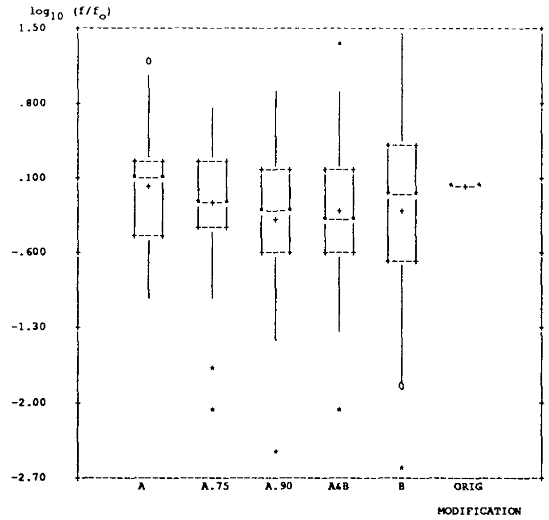


Figure 7: Boxplots of log Function Reduction for Two-Parameter Quadratic, q=10

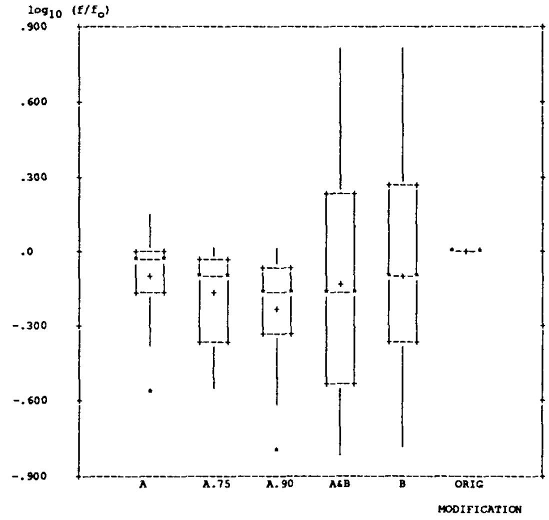


Figure 8: Boxplots of log Function Reduction for Ten-Parameter Quadratic, q=1

which directly affect contract or expand steps may adversely affect algorithm performance in the early iterations.

Finally, this study showed that the unmodified Nelder-Mead was effective even when presented with significant noise. In many cases the function values were reduced below .1, while the variance of the stochastic component was 1.

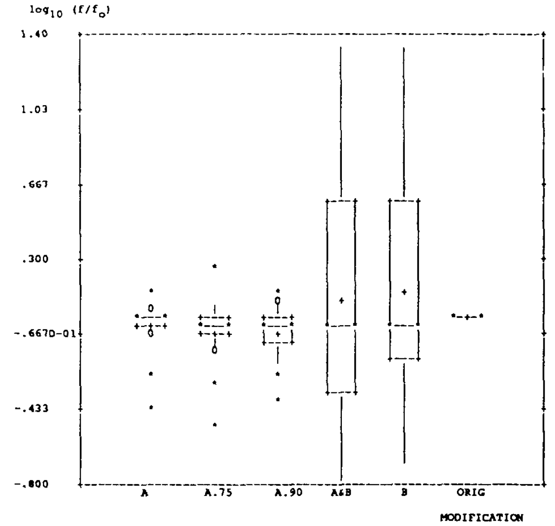


Figure 9: Boxplots of log Function Reduction for Ten-Parameter Quadratic, q=10

5 CONCLUSIONS

The Nelder-Mead simplex algorithm is a popular choice for optimizing simulation model responses. The steps are based solely on the ranks of the function values, which makes the method robust to small variations in function values. When the simulation model output has significant stochastic variation, the Nelder-Mead algorithm's rescaling steps make it vulnerable to premature convergence on stochastic functions.

Our analysis showed that the Nelder-Mead algorithm is strongly biased toward contraction in the presence of significant noise. Empirical studies of the frequency of contract and shrink iterations support the probabilistic analysis.

The results of this computational study suggest that the unmodified Nelder-Mead is often effective even in the presence of significant noise. Resampling the best point after a shrink and changing the shrink coefficient from .5 to .75 or .90 were effective in the cases where the original Nelder-Mead method failed. Resampling strategies to reduce the frequency of contraction were not effective in improving algorithm performance.

Even with the successful modifications, the algorithm frequently terminates prematurely on stochastic functions via a mix of contract and shrink steps. Further research is needed to identify effective modifications for this false convergence that do not adversely affect the performance of the algorithm in deterministic regions.

ACKNOWLEDGMENTS

Computational experiments were assisted by the resources of the Cornell Theory Center, which is supported by IBM, the National Science Foundation, New York State, and the members of the Corporate Research Institute.

REFERENCES

Ash, P. W., N. W. Barnett, L. Ebdon, and S. J. Rowley. 1989. Configuration and optimization of semiconductor gas sensor as gas chromatographic detectors. *Analytica Chimica Acta* **216**: 147-161.

Deming, S. N. and S. L. Morgan. 1973. Simplex optimization of variables in analytical chemistry. *Analytical Chemistry* **45**: 278A-283A.

Deming, S. N. and L. R. Parker, Jr. 1978. A review of simplex optimization in analytical chemistry. *CRC Critical Reviews in Analytical Chemistry* **7**: 187-202.

Ernst, R. R. 1968. Measurement and control of magnetic field homogeneity. *The Review of Scientific Instruments* **39**: 998-1012.

King, P. G. 1974. Ph.D. dissertation, Emory University, Atlanta, Georgia.

Leung, Y.T., and R. Suri. 1990. Finite-time behavior of two simulation optimization algorithms. In *Proceedings of the 1990 Winter Simulation Conference*, eds. O. Balci, R. Sadowski, and R. Nance, 372-376. Institute of Electrical and Electronics Engineers, Piscataway, New Jersey.

Meier, W. R., and E. C. Morse. 1985. A nonlinear, multivariable method for fusion reactor blanket optimization. *Fusion Technology* **8**: 2665-2680.

Nachtsheim, C. J. 1987. Tools for computer-aided design of experiments. *Journal of Quality Technology* **19**: 132-160.

Nelder, J. A. and R. Mead. 1965. A simplex method for function minimization. *The Computer Journal* **7**: 308-313.

Nieminen, K., B.A. Suarez-Isla, and S. I. Rapaport. 1988. Electrical properties of cultured dorsal root ganglion neurons from normal and trisomy 21 human fetal tissue. *Brain Research* **474**: 246-254.

Schnute, J., and J. Sibert. 1983. The salmon terminal fishery: a practical, comprehensive timing model. *Canadian Journal of Fisheries and Aquatic Sciences* **40**: 835-853.

Spendley, W., G. R. Hext, and F. R. Himsworth. 1962. Sequential application of simplex designs in optimisation and evolutionary operation. *Technometrics* **4**: 441-461.

AUTHOR BIOGRAPHIES

RUSSELL R. BARTON is an associate professor in the Department of Industrial and Management Systems Engineering at The Pennsylvania State University. He received a B.S. in electrical engineering from Princeton University in 1973 and a Ph.D. in operations research from Cornell University in 1978. Before joining Penn State he was a visiting associate professor in the School of Operations Research and Industrial Engineering at Cornell University (1987-1990) and a member of technical staff at RCA Laboratories (1978-1987). His experience at RCA involved design of experiments, process optimization, and the optimization of differential equation-based simulation models. His current research interests include graphical methods for experiment design, robust optimization methods for simulation models, and the statistical issues in modeling manufacturing yield.

JOHN S. IVEY, JR. is a graduate student in the Department of Industrial and Management Systems Engineering at The Pennsylvania State University. He received a B.S. in industrial engineering from Penn State in May 1991 and is currently working on his master's degree in Industrial Engineering and Operations Research. His research focuses on issues involved with applying optimization techniques to simulation models. He is a member of IIE, Alpha Pi Mu, and Tau Beta Pi.