Proceedings of the 1998 Winter Simulation Conference
D.J. Medeiros, E.F. Watson, J.S. Carson and M.S. Manivannan, eds.

A REVISED SIMPLEX SEARCH PROCEDURE FOR STOCHASTIC SIMULATION RESPONSE-SURFACE OPTIMIZATION

David G. Humphrey

Nortel Public Carrier Networks
2700 Meridian Parkway
Durham, NC 27713, U.S.A.

James R. Wilson

Department of Industrial Engineering
North Carolina State University
Raleigh, NC 27695-7906, U.S.A.

ABSTRACT

We develop a variant of the Nelder-Mead (NM) simplex search procedure for stochastic simulation optimization that is designed to avoid many of the weaknesses encumbering such direct-search methods—in particular, excessive sensitivity to starting values, premature termination at a local optimum, lack of robustness against noisy responses, and lack of computational efficiency. The Revised Simplex Search (RSS) procedure consists of a three-phase application of the NM method in which: (a) the ending values for one phase become the starting values for the next phase; (b) the size of the initial simplex (respectively, the shrink coefficient) decreases geometrically (respectively, increases linearly) over successive phases; and (c) the final estimated optimum is the best of the ending values for the three phases. To compare RSS versus the NM procedure and RS9 (a simplex search procedure recently proposed by Barton and Ivey), we summarize a simulation study based on separate factorial experiments and follow-up multiple comparisons tests for four selected performance measures computed on each of six test problems, with three levels of problem dimensionality and noise variability used in each problem. The experimental results provide substantial evidence of RSS's improved performance with only marginally higher computational effort.

1 INTRODUCTION

Stochastic simulation optimization can be thought of as finding a combination of input parameters that gives the optimal expected response (either minimum or maximum) of some objective function defined in terms of the (random) performance measures generated as the outputs of the simulation. Given a stochastic simulation model of a target system, the experimenter has the responsibility for selecting the values of the input parameters to the model. Let $\mathbf{x}_i \equiv [x_{i,1}, \ldots, x_{i,d}]$ be the vector of input parameters for the simulation operating under the ith scenario, where $i = 1, \ldots, n$; and n denotes the total number of scenarios (or alternative system configurations) in the overall simulation experiment.

The (deterministic) vector of design variables $\mathbf{x}_i$ and a (possibly infinite) input stream of random numbers are used by the simulation model to produce a set of (random) *performance measures* that provide an estimate of how the target system performed at the specified design point. We let $\mathbf{Y}_i^{(j)} \equiv [Y_{i,1}^{(j)}, \ldots, Y_{i,p}^{(j)}]$ denote the set of performance measures (the outputs from the simulation) observed on the jth independent replication of design point i, where p is the number of relevant simulation outputs. Each component of $\mathbf{Y}_i$ may represent any quantity of interest which is an output from the simulation.

One of the components of $\mathbf{Y}_i^{(j)}$ is usually the primary performance measure of interest in terms of optimization and is typically a function of several other outputs. We will arbitrarily let overall cost be the first element in the row vector $\mathbf{Y}_i^{(j)}$; and we let $\theta(\mathbf{x}_i)$ denote the expected value of this primary response at the ith design point $\mathbf{x}_i$ so that $\theta(\mathbf{x}_i) = \mu_1(\mathbf{x}_i) = E[Y_{i,1}^{(j)}]$. (Throughout the rest of this work, if it is unnecessary or inappropriate to identify the design point index i or the replication index j, then we will suppress these indexes and simply write $\theta(\mathbf{x}) = E[Y_1]$.)

We define the *region of interest* for the optimization procedure,

$$\Xi \equiv \{\mathbf{x} \in R^d : \mathbf{x} \text{ defines feasible system operating conditions}\},$$

where R^d denotes d-dimensional Euclidean space. The expected primary response from the simulation, $\theta(\mathbf{x})$, is what the experimenter wants to either maximize or minimize over the region of interest. If the input parameters are continuous variables, then the goal is to find a setting of the input-parameter vector such that the expected primary response is deemed to be "close enough" to the global optimum. Thus, our goal for the optimization procedure (assuming that the primary performance measure is expected total cost, which should be minimized) is to find

$$\theta^* \equiv \min_{\mathbf{x}\in\Xi} \theta(\mathbf{x}) \quad \text{and} \quad \mathbf{x}^* \equiv \arg\min_{\mathbf{x}\in\Xi} \theta(\mathbf{x}).$$

In this article we formulate, implement, and evaluate a stochastic simulation optimization procedure that incorporates many desirable properties of the well-known Nelder-Mead (NM) simplex search procedure (Nelder and Mead 1965) while avoiding some of the critical weaknesses of this procedure. In Section 2 we formulate the Revised Simplex Search (RSS) procedure. Section 3 is a comprehensive experimental performance evaluation of procedure RSS versus the classical procedure NM as well as procedure RS9, a variant of NM that was recently proposed by Barton and Ivey (1996). Finally in Section 4 we summarize the main conclusions of this research, and we present recommendations for future research in this area.

2 REVISED SIMPLEX SEARCH (RSS) PROCEDURE

Procedure RSS operates in three phases and starts with the phase counter φ being initialized to 1. The procedure is provided with initial values of the d variables over which it is to minimize in d-dimensional space. These initial values define the initial vertex $\mathbf{x}_1 \equiv [x_{1,j}, \ldots, x_{1,d}]$.

The prespecified step size ν_1 is determined for the first phase ($\varphi = 1$) using step size parameter τ and the initial vertex $\mathbf{x}_1$ as follows: $\nu_1 = \max\{\tau \cdot x_{1,j} : j = 1, \ldots, d\}$.

From this initial vertex and prespecified step size ν_1, the algorithm determines the remaining vertices $\mathbf{x}_2, \mathbf{x}_3, \ldots, \mathbf{x}_{d+1}$ (that define a d-dimensional general simplex) by moving from the initial vertex in each of the d directions one at a time as follows: $\mathbf{x}_{i+1} = \mathbf{x}_1 + \mathbf{e}_i\nu_1$ for $i = 1, \ldots, d$, where $\mathbf{e}_i$ is the unit vector with one in the ith component and zeros elsewhere.

Therefore, the initial simplex (at stage $q = 0$) and each successive simplex (at stage $q = 1, 2, \ldots$) have vertices denoted $\mathbf{x}_i \equiv [x_{i,1}, \ldots, x_{i,k}, \ldots, x_{i,d}]$ for $i = 1, \ldots, d+1$, so that $\mathbf{x}_i$ is the ith vertex (or point) during the current (that is, the qth) stage of the search. (Although $\mathbf{x}_i^{(q)}$ is a more complete notation for the ith vertex in the stage-q simplex, we suppress the exponent $^{(q)}$ for simplicity since no confusion can result from this usage.) Additionally, the simulation-based estimate of the objective function at vertex $\mathbf{x}_i$ is denoted by $\widehat{\theta}(\mathbf{x}_i)$; and we take

$$\widehat{\theta}(\mathbf{x}_{\max}) \equiv \max\left\{\widehat{\theta}(\mathbf{x}_i) : 1 \le i \le d+1\right\}, \qquad (1)$$

with $\widehat{\theta}(\mathbf{x}_{\min})$ defined similarly. The second highest of the estimates of the response surface (which corresponds to the next-to-worst vertex of the current simplex) is also noted, and we represent this quantity with $\widehat{\theta}(\mathbf{x}_{\text{ntw}})$ and the corresponding vertex with $\mathbf{x}_{\text{ntw}}$.

Some explanation is required for the notation used in the rest of this paper. When we use the notation $\widehat{\theta}(\mathbf{x}_{\max})$, we are referring to the simulation response at the vertex yielding the maximum response on the current stage of the search; and this is precisely the notation we will use when emphasis on the vertex $\mathbf{x}_{\max}$ is important. When it is not important to emphasize the vertex $\mathbf{x}_{\max}$, we will use the simpler notation $\widehat{\theta}_{\max}$. And in a like manner, $\widehat{\theta}(\mathbf{x}_{\min})$ will be represented by $\widehat{\theta}_{\min}$ when no ambiguity can arise from this usage. In the following description of procedure RSS, several points (or vertices) are named and reference is made to their corresponding estimates of the response surface. Within the verbal description and the flow chart of the algorithm, we will use the shorter versions of the notation below rather than the more cumbersome version on the right-hand of each of the following definitions:

$$\left.\begin{array}{ll} \widehat{\theta}_{\max} = \widehat{\theta}(\mathbf{x}_{\max}), & \widehat{\theta}_{\min} = \widehat{\theta}(\mathbf{x}_{\min}), \\ \widehat{\theta}_{\text{cen}} = \widehat{\theta}(\mathbf{x}_{\text{cen}}), & \widehat{\theta}_{\text{refl}} = \widehat{\theta}(\mathbf{x}_{\text{refl}}), \\ \widehat{\theta}_{\text{ntw}} = \widehat{\theta}(\mathbf{x}_{\text{ntw}}), & \widehat{\theta}_{\exp} = \widehat{\theta}(\mathbf{x}_{\exp}), \\ \widehat{\theta}_{\text{cont}} = \widehat{\theta}(\mathbf{x}_{\text{cont}}). & \end{array}\right\} \qquad (2)$$

Finally, we preserve the best answer from each phase φ by letting $\widehat{\mathbf{x}}^*(\varphi)$ denote the final estimate of the optimal solution delivered in phase φ, where $\varphi = 1, 2, 3$.

For $q = 0, 1, \ldots$, the qth stage of the algorithm begins by computing the centroid of all the vertices used in the current simplex excluding $\mathbf{x}_{\max}$; thus we obtain the centroid of a d-dimensional polyhedron in R^d. The centroid is labeled $\mathbf{x}_{\text{cen}}$, and its coordinates are given by

$$\mathbf{x}_{\text{cen}} = \frac{1}{d}\left\{\left[\sum_{i=1}^{d+1} \mathbf{x}_i\right] - \mathbf{x}_{\max}\right\}. \qquad (3)$$

The procedure then proceeds through four operations (reflection, expansion, contraction, and shrinkage) described below until the stopping criterion is satisfied. A flow chart of the algorithm is depicted below in Figure 1, and a formal statement of the algorithm is given below.

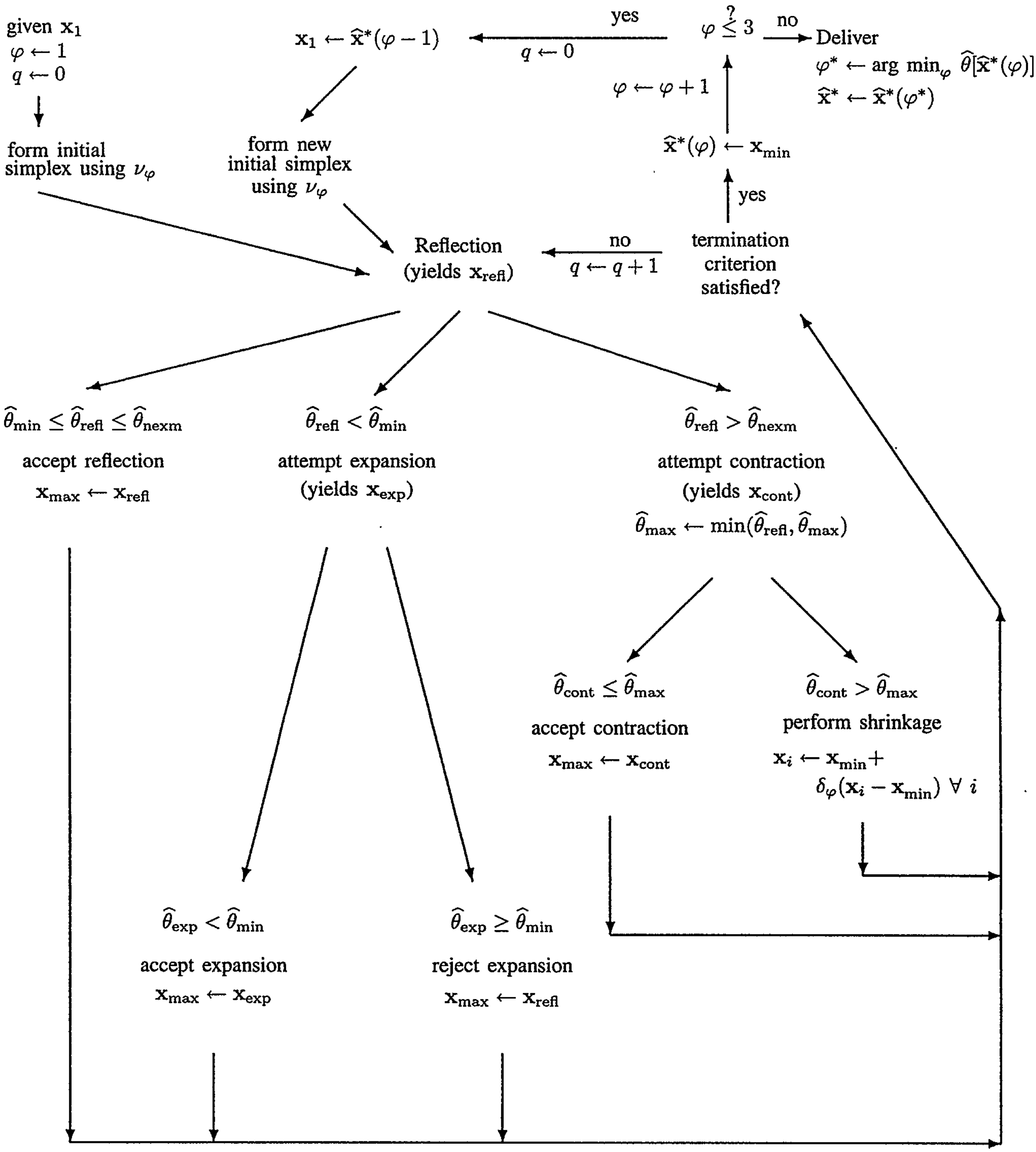

Figure 1: Flow Chart of Procedure RSS

STEPS OF PROCEDURE RSS

0. *Set Up Initial Simplex.* Initialize the phase counter $\varphi \leftarrow 1$, the iteration counter $q \leftarrow 0$, and the vertices $\{\mathbf{x}_i : 1 \leq i \leq d+1\}$ of the initial simplex using the initial prespecified step size ν_1. Go to step 1.

1. *Attempt Reflection.* A new simplex is formed by reflecting $\mathbf{x}_{\text{max}}$ through the centroid $\mathbf{x}_{\text{cen}}$ of the remaining vertices to find a new vertex. Specifically, $\mathbf{x}_{\text{max}}$ is reflected through the centroid to yield the reflected point

$$\mathbf{x}_{\text{refl}} = \mathbf{x}_{\text{cen}} + \alpha \cdot (\mathbf{x}_{\text{cen}} - \mathbf{x}_{\text{max}}),$$

where: $\alpha = 1.0$ is the reflection coefficient, $\mathbf{x}_{\text{cen}}$ is the centroid from (3), and $\mathbf{x}_{\text{max}}$ is the vertex corresponding to $\widehat{\theta}_{\text{max}}$ from (1). If

$$\widehat{\theta}_{\text{min}} \leq \widehat{\theta}_{\text{refl}} \leq \widehat{\theta}_{\text{ntw}}, \tag{4}$$

that is, if the reflected point $\mathbf{x}_{\text{refl}}$ is better than the next-to-worst vertex $\mathbf{x}_{\text{ntw}}$ in the current simplex but is not better than the best vertex $\mathbf{x}_{\text{min}}$, then the worst vertex $\mathbf{x}_{\text{max}}$ is replaced by the reflected point $\mathbf{x}_{\text{refl}}$:

$$\mathbf{x}_{\text{max}} \leftarrow \mathbf{x}_{\text{refl}}. \tag{5}$$

In this case the procedure continues with the termination criterion (i.e., the procedure goes to step 6). If the condition (4) for accepting the reflection is not satisfied, then the algorithm proceeds to step 2.

2. *Attempt Expansion.* If the reflected point $\mathbf{x}_{\text{refl}}$ is better than the best vertex $\mathbf{x}_{\text{min}}$ in the current simplex, then the reflection operation is expanded in an attempt to obtain even greater improvement. Specifically, if

$$\widehat{\theta}_{\text{refl}} < \widehat{\theta}_{\text{min}} \tag{6}$$

then the vector $\mathbf{x}_{\text{refl}} - \mathbf{x}_{\text{cen}}$ is extended to yield the expansion point

$$\mathbf{x}_{\text{exp}} = \mathbf{x}_{\text{cen}} + \gamma \cdot (\mathbf{x}_{\text{refl}} - \mathbf{x}_{\text{cen}}),$$

where $\gamma = 2.0$ is the expansion coefficient (discussed below). If $\widehat{\theta}_{\text{exp}} < \widehat{\theta}_{\text{min}}$, then the expansion is accepted so that $\mathbf{x}_{\text{max}}$ is replaced by $\mathbf{x}_{\text{exp}}$ in the simplex

$$\mathbf{x}_{\text{max}} \leftarrow \mathbf{x}_{\text{exp}};$$

and the algorithm continues by going to the termination criterion (step 6). If $\widehat{\theta}_{\text{exp}} \geq \widehat{\theta}_{\text{min}}$, then the attempted expansion is rejected and $\mathbf{x}_{\text{max}}$ is replaced by $\mathbf{x}_{\text{refl}}$

$$\mathbf{x}_{\text{max}} \leftarrow \mathbf{x}_{\text{refl}};$$

and the procedure continues by going to step 6. Finally if the condition (6) for attempting expansion is not satisfied, then the algorithm proceeds to step 3.

3. *Set Up Attempted Contraction.* If

$$\widehat{\theta}_{\text{refl}} > \widehat{\theta}_{\text{ntw}}$$

so that the reflected point $\mathbf{x}_{\text{refl}}$ yields a worse (larger) response than the next-to-worst vertex $\mathbf{x}_{\text{ntw}}$ of the current simplex, then some reduction in the size of the current simplex must be performed — either a contraction or a more drastic shrinkage. To set up this reduction in the size of the simplex, the worst (largest) vertex in the current simplex is updated as follows:

$$\text{if } \widehat{\theta}_{\text{refl}} \leq \widehat{\theta}_{\text{max}}, \text{ then} \left\{ \begin{array}{c} \mathbf{x}_{\text{max}} \leftarrow \mathbf{x}_{\text{refl}} \\ \widehat{\theta}_{\text{max}} \leftarrow \widehat{\theta}_{\text{refl}} \end{array} \right\}.$$

The attempted contraction in the next step will seek to replace the current worst vertex $\mathbf{x}_{\text{max}}$ with the new point

$$\mathbf{x}_{\text{cont}} = \mathbf{x}_{\text{cen}} + \beta \cdot (\mathbf{x}_{\text{max}} - \mathbf{x}_{\text{cen}}),$$

where $\beta = 0.5$ is the contraction coefficient.

4. *Accept Contraction.* If

$$\widehat{\theta}_{\text{cont}} \leq \widehat{\theta}_{\text{max}},$$

so that the contracted point $\mathbf{x}_{\text{cont}}$ yields a better (smaller) response than the worst vertex $\mathbf{x}_{\text{max}}$ of the current simplex, then $\mathbf{x}_{\text{max}}$ is replaced by $\mathbf{x}_{\text{cont}}$

$$\mathbf{x}_{\text{max}} \leftarrow \mathbf{x}_{\text{cont}};$$

and the procedure goes to step 6; otherwise the procedure goes to step 5.

5. *Perform Shrinkage.* If the contracted point $\mathbf{x}_{\text{cont}}$ yields a worse (larger) response than every vertex in the current simplex including $\mathbf{x}_{\text{max}}$ so that the shrinkage condition

$$\widehat{\theta}_{\text{cont}} > \widehat{\theta}_{\text{max}}$$

is satisfied, then the size of the current simplex must be reduced to an extent that depends on the current phase φ of procedure RSS since the algorithm has likely "overshot" an area of improvement. In this case, all of the simplex edges with end point $\mathbf{x}_{\text{min}}$,

$$\mathbf{x}_i - \mathbf{x}_{\text{min}}, \quad i = 1, \ldots, d+1,$$

are reduced by the shrinkage factor δ_φ, yielding new vertices

$$\mathbf{x}_i \leftarrow \mathbf{x}_{\text{min}} + \delta_\varphi \cdot (\mathbf{x}_i - \mathbf{x}_{\text{min}}), \quad i = 1, \ldots, d+1,$$

where δ_φ is the shrinkage coefficient for the current phase φ of procedure RSS. The procedure then goes to step 6.

6. *Test Termination Criterion for Current Phase.* After each reflection, expansion, contraction, or shrinkage, the

stopping rule is applied to determine if sufficient progress has been made. The termination criterion is

$$\frac{\max_i \| \mathbf{x}_i - \mathbf{x}_{\min} \|}{\| \mathbf{x}_{\min} \|} \leq \eta, \tag{7}$$

where η is again a user-specified tolerance. This stopping rule examines the largest distance from any vertex in the simplex to the best vertex in the simplex ($\mathbf{x}_{\min}$) in making a decision about termination of the search. If the termination condition (7) is not satisfied, then the iteration counter q is incremented $q \leftarrow q+1$ and the procedure returns to step 1. If the termination condition (7) is satisfied, then the procedure goes to step 7.

7. *End Current Phase.* To complete the current phase of procedure RSS, we record the termination point of the current phase

$$\widehat{\mathbf{x}}^*(\varphi) \leftarrow \mathbf{x}_{\min}$$

and then we increment the phase counter,

$$\varphi \leftarrow \varphi + 1.$$

8. *Test Final Termination Criterion.* If $\varphi > 3$, then procedure RSS delivers the final estimate $\widehat{\mathbf{x}}^*$ of the global optimum according to

$$\varphi^* \leftarrow \arg\min \left\{ \widehat{\theta}[\widehat{\mathbf{x}}^*(\varphi)] : \varphi = 1, 2, 3 \right\}$$

and

$$\widehat{\mathbf{x}}^* \leftarrow \widehat{\mathbf{x}}^*(\varphi^*);$$

then procedure RSS terminates. If $\varphi \leq 3$, then the procedure goes to step 9.

9. *Initialize Next Phase.* Initialize the iteration counter

$$q \leftarrow 0,$$

the prespecified step size

$$\nu_\varphi \leftarrow \frac{1}{2}\nu_{\varphi-1},$$

and the first vertex of the initial simplex,

$$\mathbf{x}_1 \leftarrow \widehat{\mathbf{x}}^*(\varphi - 1).$$

Form the other vertices of the initial simplex

$$\mathbf{x}_{i+1} \leftarrow \mathbf{x}_1 + \mathbf{e}_i \nu_\varphi \text{ for } i = 1, \ldots, d.$$

Go to step 1.

3 EXPERIMENTAL PERFORMANCE EVALUATION

3.1 Formulation of Performance Measures

In this work, four performance measures were used for evaluating stochastic simulation optimization procedures: (i) natural log L of the number of function evaluations; (ii) absolute percentage deviation of the final function value $\widehat{\theta}^* \equiv \widehat{\theta}(\widehat{\mathbf{x}}^*)$ from the optimal function value $\theta^* \equiv \theta(\mathbf{x}^*)$,

$$D = \left| \frac{\widehat{\theta}^* - \theta^*}{\theta^*} \right|;$$

(iii) maximum absolute percentage deviation in each coordinate of the final point $\widehat{\mathbf{x}}^*$ from the corresponding coordinate of the true optimal point $\mathbf{x}^*$,

$$B = \max_{1 \leq j \leq d} \left| \frac{\widehat{x}_j^* - x_j^*}{x_j^*} \right|; \tag{8}$$

and (iv) average absolute percentage deviation in all coordinates of the final point $\widehat{\mathbf{x}}^*$ from the corresponding coordinates of the optimal point $\mathbf{x}^*$,

$$A = \frac{1}{d} \sum_{j=1}^{d} \left| \frac{\widehat{x}_j^* - x_j^*}{x_j^*} \right|. \tag{9}$$

When there are multiple optima, we evaluate the right-hand sides of (8) (respectively, (9)) for each optimum and take the smallest of these quantities as the final value of A (respectively, B).

3.2 Description of Test Problems

We selected six problems to serve as a test-bed for comparing the performance of procedure RSS with that of procedures NM and RS9. All six problems are minimization problems. In this section we describe one of the selected test problems by explaining the function to be minimized, the starting point used, the optimal function value, and the points corresponding to the optimal function value. A complete description of all test problems is given in Humphrey (1997).

The second problem considered is the trigonometric function. The function is defined as

$$\theta(\mathbf{x}) = \sum_{i=1}^{d} [f_i(\mathbf{x})]^2 + 1,$$

where

$$f_i(\mathbf{x}) = d - \sum_{j=1}^{d} \Big\{ \cos(x_j - 1) + i[1 - \cos(x_i - 1)] - \sin(x_i - 1) \Big\} \text{ for } i = 1, \ldots, d.$$

We used dimensionalities $d = 2$, 10, and 18 for our purposes and a starting point of $\mathrm{x}_1 \equiv [1/d, \ldots, 1/d]$. The optimal value of $\theta^* = 1$ is achieved at every point in the lattice of points given by

$$\begin{aligned} \mathrm{x}^*_{k_1 k_2 \cdots k_d} &= [1 + 2\pi k_1, \ldots, 1 + 2\pi k_d], \text{ where} \\ & k_j = 0, \pm 1, \pm 2, \ldots, \text{ for } j = 1, \ldots, d. \end{aligned}$$

The complicated nature of this function is depicted in Figure 2 for $d = 2$. For this test problem, when a given search procedure terminated, we determined which of the optimal points was closest in Euclidean distance to the final estimate $\widehat{\mathrm{x}}^*$; and we used that optimal point for calculating performance measures A and B.

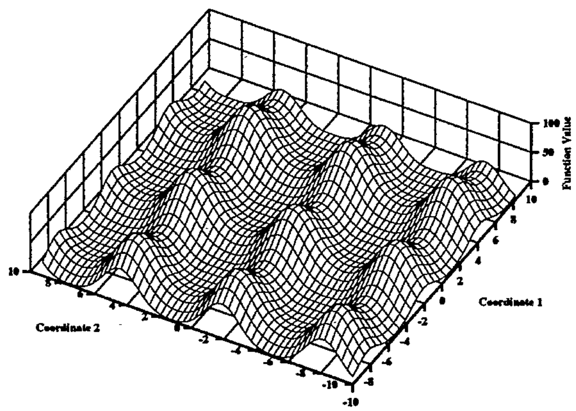

Figure 2: Trigonometric Function (Test Problem 2) for Dimensionality $d = 2$

3.3 Summary of Experimental Results

We decided to examine a variety of problems at different dimensionalities and with different levels of stochastic noise included. The selected dimensionalities are listed in the previous section and were usually 2, 10, and 18 (low, medium, and high) dimensions. We selected 2-dimensional problems as we wanted to see how the procedures compared over "small" problems. Our decision to look at 10-dimensional problems came primarily from the literature which indicates that simplex search type procedures tend to perform well up to about dimensionality 10 (Nelder and Mead 1965). We also wanted to examine the performance of such procedures above dimensions of 10, and that led us to look at 18-dimensional problems. The exception to these general rules is problem 4 (the extended Powell singular function), for which the dimensionality was required to be a multiple of 4. Hence, we used $d = 4, 8,$ and 16 (low, medium, and high) as previously mentioned.

The test problems we used are all deterministic functions, unlike responses generated by a stochastic simulation model. In order to work with stochastic responses, we added a noise component to each deterministic function value. The noise component is a random sample from a normal distribution with a mean of zero and a standard deviation of either 0.75, 1.0, or 1.25 times the magnitude of the optimal response $|\theta^*|$. These three levels of stochastic noise provide us with low, medium, and high levels of variation around the true underlying response surface relative to the optimal function value $\theta^* = 1$ that was common to all six test problems.

Our study of the ith problem $(1 \le i \le 6)$ constituted a complete factorial experiment in which there were three factors each at three levels as defined below:

$$\begin{aligned} P_j &\equiv j\text{th level of optimization procedure} \\ &= \begin{cases} \text{NM} & \text{for } j = 0, \\ \text{RSS} & \text{for } j = 1, \\ \text{RS9} & \text{for } j = 2; \end{cases} \end{aligned}$$

$$\begin{aligned} G_k &\equiv k\text{th level of problem dimensionality} \\ &= \begin{cases} 2 \text{ (4 in problem 4)} & \text{for } k = 1, \\ 10 \text{ (8 in problem 4)} & \text{for } k = 2, \\ 18 \text{ (16 in problem 4)} & \text{for } k = 3; \end{cases} \end{aligned}$$

and

$$\begin{aligned} N_\ell &\equiv \ell\text{th level of noise standard deviation} \\ &= \begin{cases} 0.75 \cdot |\theta^*| & \text{for } \ell = 1, \\ 1.00 \cdot |\theta^*| & \text{for } \ell = 2, \\ 1.25 \cdot |\theta^*| & \text{for } \ell = 3. \end{cases} \end{aligned}$$

In the experiment on problem i $(1 \le i \le 6)$, we let $L_{i,j,k,\ell,m}$ (respectively, $D_{i,j,k,\ell,m}$ or $B_{i,j,k,\ell,m}$ or $A_{i,j,k,\ell,m}$) denote the performance measure L (respectively, D or B or A) observed on the mth replication $(1 \le m \le 9)$ of the treatment combination (P_j, D_k, N_ℓ). We compute the average performance measures $\overline{L}_{i,j}$, $\overline{D}_{i,j}$, $\overline{B}_{i,j}$, and $\overline{A}_{i,j}$ for each problem i $(1 \le i \le 6)$ and optimization procedure j $(0 \le j \le 2)$.

Within the ith experiment and for each of the selected performance measures that were observed on the mth replication of the treatment combination (P_j, G_k, N_ℓ), we postulate a linear statistical model of the form

$$\begin{aligned} Z_{jk\ell m} = \beta_0 &+ \beta_P W_{P_j} + \beta_D W_{G_k} + \beta_N W_{N_\ell} \\ &+ \beta_{PD} W_{P_j} W_{G_k} + \beta_{PN} W_{P_j} W_{N_\ell} \\ &+ \beta_{DN} W_{G_k} W_{N_\ell} + \varepsilon_{jklm}, \end{aligned} \tag{10}$$

where the "coded" independent variables W_{P_j}, W_{G_k}, and W_{N_ℓ} are defined as follows:

$$W_{P_j} = \begin{cases} -1, & \text{for } j = 0, \\ 0, & \text{for } j = 1, \\ +1, & \text{for } j = 2; \end{cases}$$

and W_{G_k} and W_{N_ℓ} are defined similarly. Note that in the linear model (10), the dependent variable $Z_{jk\ell m}$ is taken to be $L_{jk\ell m}$, $D_{jk\ell m}$, $B_{jk\ell m}$, or $A_{jk\ell m}$, depending on the performance measure under consideration.

3.4 Analysis of Experimental Results

Most of the analysis of this section will be drawn directly from the numerical information presented in Table 1. We will consider each of the performance measures independently over the six problems studied.

3.4.1 Linear Statistical Model

The linear statistical model (10) presented previously appears to provide an adequate fit for our purposes. All r^2 values were above 0.93 and most were above 0.99. As expected, two factors presented significant effects: problem dimensionality and search procedure. As the dimensionality of the test problems increases, the search obviously becomes more difficult and this manifests itself in large F-values for the dimensionality factor.

The results of the analysis of variance also indicate to us that the search procedure factor is significant and this is precisely what we want – evidence that procedure RSS is significantly different from one or both of NM and RS9 (as is discussed here) and evidence that procedure RSS is significantly better than, in at least a large number of cases, one or both of NM and RS9. The F-values corresponding to the search procedure simply indicate that there is some significant difference in the search procedures being studied. Other discussion and results, mostly associated with Table 1, will be presented to show that not only are there significant differences among procedures RSS, NM, and RS9, but that there is evidence to conclude that RSS performs better, on the whole, than NM or RS9.

Of the interactions within the linear model (10), only the problem dimensionality–search procedure interaction at times yielded F-values that appeared to indicate some significance. A closer examination of search procedures RSS, NM, and RS9 over a wider variety of problem dimensionalities is a likely area of future investigation. We felt that our goal of presenting an improved search procedure for a broad class of test problems at a variety of different dimensions was adequately met without a more thorough and detailed examination of search-procedure performance over a greater number of problem dimensionalities.

3.4.2 Number of Function Evaluations

The first performance measure considered here is L, the natural log of the number of function evaluations performed. This is particularly important to us as it gives us an idea of the computer time required to complete the desired simulation optimization. We were willing at the start of this work to accept an improved procedure which would require considerably more computational effort, provided that effort produced better results. We were willing to consider an increase of an order in magnitude in the number of function evaluations as being tolerable since computer time is relatively inexpensive and becoming more so with each passing year. What we really hoped for, however, was an increase of not more than 5 to 8 times the number of function evaluations of standard Nelder-Mead. From Table 1 it is apparent that our procedure is more costly in terms of function evaluations required. However, the increased effort is only about 3 or 4 times greater than that of procedure NM for both RSS and RS9. If we temporarily disregard the natural log transformation and take the number of standard Nelder-Mead evaluations as a sort of baseline, Table 2 shows a summary of the relative amount of computational effort required by each of the three procedures on each of the six problems.

Procedure RSS is doing more work than NM and about as much as much work as RS9; and this is more than acceptable for, as we shall see, our procedure is providing better results at only a marginally higher cost.

3.4.3 Final Function Value at Estimated Optimum

In looking at performance measure D we make the following observations. In five of the six problems considered, RSS produced an average value of D that is statistically significantly better than the average D-values produced by either NM or RS9. In problem number 2 (that is, the trigonometric function), RSS and RS9 had results that were not distinguishable from each other but were significantly better than those from NM. Additionally, RSS is *much* better than either RS9 or NM on two of the six problems (namely, problems 1 and 4).

Table 2: Relative Computational Effort of Procedures

Procedure	Problem 1	2	3	4	5	6	Average
NM	1	1	1	1	1	1	1
RSS	3.2	3.6	3.1	4.5	3.1	3.9	3.6
RS9	3.6	4.7	3.3	3.5	3.8	4.3	3.8

3.4.4 Maximum Relative Component Deviation from Global Optimum

The next performance measure considered, B, deals with the maximum deviation of any single coordinate of the best

Table 1: Results of Multiple Comparisons Tests on Optimization Procedures NM, RS9, and RSS for Level of Significance $\alpha = 0.05$

Test Problem	Performance Measure $\overline{L}_{i,j}$			$\overline{D}_{i,j}$			$\overline{B}_{i,j}$			$\overline{A}_{i,j}$		
	Grouping	Calc. Val.	Opt. Proc.	Grouping	Calc. Val.	Opt. Proc.	Grouping	Calc. Val.	Opt. Proc.	Grouping	Calc. Val.	Opt. Proc.
	1	6.86	RS9	1	5.10	NM	1	1.28	NM	1	0.41	NM
							1			1		
1	2	6.68	RSS	2	4.94	RS9	1	1.28	RS9	1	0.40	RS9
	3	5.57	NM	3	0.48	RSS	2	0.38	RSS	2	0.19	RSS
	1	7.06	RS9	1	0.22	NM	1	0.47	NM	1	0.29	NM
2	2	6.83	RSS	2	0.12	RSS	2	0.39	RS9	2	0.24	RS9
				2								
	3	5.56	NM	2	0.10	RS9	3	0.35	RSS	3	0.20	RSS
	1	6.69	RS9	1	20.2	NM	1	2.01	NM	1	1.04	RS9
	1			1			1			1		
3	1	6.68	RSS	1	20.0	RS9	1	2.01	RS9	1	1.04	NM
	2	5.50	NM	2	18.2	RSS	2	1.74	RSS	2	0.96	RSS
	1	7.25	RSS	1	11.0	NM	1	1.66	NM	1	0.84	NM
4	2	7.02	RS9	2	10.1	RS9	2	1.59	RS9	2	0.79	RS9
	3	5.78	NM	3	3.76	RSS	3	0.95	RSS	3	0.41	RSS
	1	6.89	RS9	1	1.85	NM	1	0.78	RS9	1	0.32	RS9
				1			1			1		
5	2	6.68	RSS	1	1.56	RS9	1	0.78	NM	1	0.30	NM
										1		
	3	5.57	NM	2	0.53	RSS	2	0.55	RSS	1	0.29	RSS
	1	6.98	RS9	1	245.2	RS9	1	1.75	RSS	1	0.99	NM
							1			1		
6	2	6.91	RSS	2	238.3	NM	21	1.53	NM	1	0.92	RS9
							2			1		
	3	5.60	NM	3	229.8	RSS	2	1.40	RS9	1	0.92	RSS

point found via the given procedure and the corresponding coordinate of the nearest optimal point. The information in Table 1 shows that RSS is statistically significantly better than NM or RS9 in this respect. While the results here may not appear as dramatic as those concerning the average function values, they still clearly favor RSS. In problem 1, RSS has a B value of about 0.38 while the corresponding values for RS9 and NM are each about 1.28. The results are less dramatic for problems 2–5, but they still clearly favor RSS. Only for problem 6 do the results not clearly indicate the superiority of RSS.

3.4.5 Average Relative Component Deviation from Global Optimum

The final performance measure considered is A, which looks at the average deviation in the coordinates of the best point located versus those of the known best point. Once again, the performance of RSS is considerably better than that of either RS9 or NM. In four of the six problems, RSS is shown to again be statistically significantly better that RS9 and NM. In problems 5 and 6 the performance of RSS is not significantly better than that of NM and RS9, but RSS is also shown to be no worse than NM and RS9. In problems 1–4, RSS is grouped by itself with a lower (better) average value of A. The single best performance in this regard is that of problem 1 where RSS scores a 0.19, RS9 scores a 0.40, and NM scores a 0.41.

In summary, the information in Table 1 shows that RSS is doing about as much work as RS9 and about 3 or 4 times as much work as NM. But in exchange for the additional work, RSS is producing results which are consistently at least as good as the results from RS9 and NM and in most cases are significantly better than RS9 and NM (and in a few cases dramatically better). It seems that the additional work will consistently improve the results without any danger of producing results worse that would be achieved with RS9 or NM.

4 CONCLUSIONS AND RECOMMENDATIONS

The experimental analysis summarize in Section 3 led us to conclude that procedure RSS successfully avoids some of the weaknesses of traditional search procedures and provides significant improvement over procedures NM and RS9 in terms of three of our four performance measures: D, B, and A. In terms of our fourth performance measure, L, we see that some additional computational effort is required by procedure RSS in comparison to procedure NM, but procedures RSS and RS9 displayed no significant differences in terms of performance measure L. We do not consider the differences in performance measure L for the three procedures to be of a significant nature.

The analysis in Section 3 raises questions and issues that merit consideration for future work. The suite of six test problems could be enlarged to provide for analysis on a collection of test problems that encompasses an even greater degree of difficulty, dimensionality, and geometry of response surfaces. The experimental performance could also be expanded to include other variants of procedure NM as well as techniques such as simulated annealing, genetic algorithms, and so forth. Finally, an effort should be made to formulate some rules of thumb for the use of procedure RSS in practice, including how to establish initial values for the search procedure and how to set some reasonable stopping tolerance so that procedure RSS may be effectively used by practitioners.

REFERENCES

Barton, R. R., and J. S. Ivey, Jr. 1996. Nelder-Mead simplex modifications for simulation optimization. *Management Science* 42 (7): 954–973.

Humphrey, D. G. 1997. A revised simplex search procedure for stochastic simulation response surface optimization. Ph.D. dissertation, Department of Industrial Engineering, North Carolina State University, Raleigh, North Carolina.

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

AUTHOR BIOGRAPHIES

DAVID G. HUMPHREY is a member of the Operations Research Department of Nortel Public Carrier Networks in Durham, North Carolina. He is a member of IIE and INFORMS.

JAMES R. WILSON is Professor and Director of Graduate Programs in the Department of Industrial Engineering at North Carolina State University. Currently he serves as a corepresentative of the INFORMS College on Simulation to the WSC Board of Directors. He is a member of ASA, ACM, IIE, and INFORMS.