AN ENHANCED RECURSIVE STOPPING RULE FOR STEEPEST ASCENT SEARCHES IN RESPONSE SURFACE METHODOLOGY.

Guillermo Miró and Enrique Del Castillo
Department of Industrial and Manufacturing Engineering The Pennsylvania State University
University Park, PA 16802, USA

Key Words: Process Optimization, Line Searches, Recursive Least Squares Estimation.

ABSTRACT

In traditional Response Surface Methods (RSM) the fitting of a factorial design (possible fractional) is followed by a steepest ascent search using the vector of first order parameter estimates as an approximation for the gradient. In the presence of variability in the responses, there is a need for a stopping rule to determine the optimal point in the search direction. Two formal stopping rules have been proposed in the literature, Myers and Khuri's (MK) stopping rule and Del Castillo's recursive parabolic rule. The first rule requires the specification of an initial guess on the location of the optimum, while the performance of the latter rule has only been studied for quadratic responses. This paper proposes some modifications to the recursive parabolic rule in order to increase its robustness for non-quadratic responses. The modifications include using only a fraction of the experiments made along the search and the estimation of all the parameters in the recursive model. It also compares, using simulation experiments, the performance of the aforementioned rules, together with classical rules of stopping after 1, 2 or 3 consecutive drops, under non-quadratic and non-normally distributed responses. It was observed that the original recursive parabolic rule stops before the optimum under non-quadratic behavior, while the modified parabolic rule and the MK rule perform satisfactorily under most of the simulated conditions.

1. INTRODUCTION.

Since the seminal paper by Box and Wilson (1951), traditional response surface methods have been based on initially conducting steepest ascent/descent searches until significant curvature is detected. It will be assumed in this paper, without loss of generality, that it is of interest to maximize the response so only the steepest ascent case will be discussed. Steepest ascent is conducted based on experiments conducted on the direction defined by the gradient of an estimated main effects

model. The observed responses along the steepest ascent direction are used to estimate when one has reached the maximum on such direction. This paper presents a new rule for determining when to stop a steepest ascent search in an RSM study.

The details of how to conduct steepest ascent are not completely defined in the literature, in particular with respect to when or how to stop a search in the steepest ascent direction. In practice, stopping a search too early before the true maximum over the steepest ascent direction evidently implies that the optimum will be missed. The end result, over many line searches, is the increase in the total number of experiments conducted. Likewise, a rule that stops many steps after the true maximum evidently results in wasted resources in experimentation. These two types of errors become more severe as the sampling error becomes more dominant. Common stopping rules, like stopping after the first observed drop in response or stopping only after 3 consecutive observed drops suffer from either of these two types of problems. Interest is in finding a stopping rule that stops after, but as close as possible, to the true maximum along the steepest ascent direction.

In the literature, two formal rules have been proposed for signaling when to stop a search: a rule by Myers and Khuri (1979) and a rule by Del Castillo (1997). The latter paper contains comparisons of these two rules and the simpler stopping rule of stopping after 1, 2, and 3 drops in a row. The Myers and Khuri rule is a formal test for the hypothesis that a drop in response is significant, and requires to this effect that a preliminary guess on the number of steps to the optimum be given. It also requires normality of the errors. The rule of Del Castillo (1997) recursively fits a parabola to the sequence of observed responses. He proposes to fit recursively the second order coefficient of a quadratic polynomial and to test for the negativeness of the first derivative. Even though this rule does not require normally-distributed errors, its performance was only studied when the response function was quadratic with additive noise. Although classical RSM uses second order models to obtain a local approximation of the true function, during steepest ascent searches significant changes are made to the controllable factors and the quadratic approximation may not longer be valid. Furthermore, the intercept and first order term carried over from the previous factorial design may also need to be changed in order to give a better local approximation of the true function.

The objectives of this paper are 1) to present extensions and improvements to the recursive rule proposed by Del Castillo (1997) with the objective of increasing its robustness against non-quadratic responses, and 2) to evaluate the performance of the modified recursive rule by comparing it to the rule by Myers and Khuri (1979), to the original recursive parabolic rule, and to the other

simpler rules used in practice.

The following five sections are organized as follows. Section 2 reviews previous research in this area and states the scope of the current research. Sections 3 and 4 present the proposed modifications to the recursive rule of Del Castillo (1997). Section 5 provides simulation results where the various stopping rules are compared for non-quadratic responses with normal and non-normal additive noise. Finally, section 6 contains conclusion and recommendations for future research.

2. PREVIOUS RESEARCH AND OBJECTIVES OF CURRENT RESEARCH.

Myers and Khuri Stopping Rule

The Myers and Khuri (1979) stopping rule for steepest ascent involves a sequential test of hypothesis. To apply this rule the user needs to make a preliminary guess, denoted by κ, of the number of steps required to reach the optimum. The value of κ is used in the following equation to obtain the test limits:

$$a = \Phi^{-1}\left(\frac{1}{2\kappa}\right)\sigma_\varepsilon\sqrt{2} = -b \tag{1}$$

As it can be expected, the procedure is rather sensitive to the value of κ chosen (Del Castillo 1997). The procedure is equivalent to testing for the significance of the difference between two responses, where the size of the test is given by $1/2\kappa$, e.g. for $\kappa = 10$ a test of size $\alpha = 0.05$ is obtained. For higher values of κ the sensitivity is reduced because of the inverse normal function. The remarkable characteristic of this rule is the absence of an assumption regarding the type of functional relationship between the response and the controllable factors. However, it uses only part of the information gathered during the search, and assumes normally distributed errors.

Recursive Parabolic Rule

In the derivation of his stopping rule, Del Castillo (1997) used the intercept and the directional derivative in the gradient direction from the previous factorial experiment to estimate the intercept and first order coefficient of the parabolic function given by:

$$Y(t) = \theta_0 + \theta_1 t + \theta_2 t^2 + \epsilon_t \tag{2}$$

where it was assumed that $\epsilon_t \sim N(0, \sigma_\epsilon^2)$ were i.i.d. random variables. These estimates were used

as true values through all the procedure. The procedure consists of re-estimating recursively the curvature coefficient θ_2. This scheme allows to have estimates of all three parameters after just the first experiment.

Let t be the step counter, $\widehat{Y}(0)$ be the response at $t = 0$ (usually obtained from the arithmetic mean of the previous factorial center points), $Y(t)$ the result of the t^{th} experiment, and $\widehat{\theta}_2^{(t)}$ the estimate of the second order coefficients which uses observations up to and including $Y(t)$. With this notation, the algorithm proposed in Del Castillo (1997) was:

1. Update the estimate of $\widehat{\theta}_2^{(t)}$ as follows:

$$\widehat{\theta}_2^{(t)} = \widehat{\theta}_2^{(t-1)} + \frac{P_{t-1}t^2}{1 + t^4 P_{t-1}}(Y(t) - \widehat{Y}(0) - \theta_1 t - \widehat{\theta}_2^{(t-1)}t^2) \tag{3}$$

2. Perform a rank 1 update of the covariance matrix. In the case of a single regressor, this is just a scalar update and it can be simplified to:

$$P_t = \left(1 - \frac{P_{t-1}t^4}{1 + t^4 P_{t-1}}\right) P_{t-1} \tag{4}$$

   The reader can refer to Wellstead and Zarrop (1991) for the general case of multiple regressors.

3. Finally, if

$$\theta_1 + 2\widehat{\theta}_2^{(t)}t < -3\sqrt{\widehat{\sigma}^2_{(\theta_1 + 2\widehat{\theta}_2^{(t)}t)}} \tag{5}$$

   stop the search and return t_{opt} such that $Y(t_{opt}) = max_{l=1,..,t}\{Y(l)\}$. Otherwise, increase the step counter (t) and go back to 1. Here, $\widehat{\sigma}^2_{(\theta_1 + 2\widehat{\theta}_2^{(t)}t)}$ denotes the variance of $\theta_1 + 2\widehat{\theta}_2^{(t)}t$, and P_t denotes the variance of $\widehat{\theta}_2^{(t)}$.

To start up the RLS scheme, prior estimates of P_t and θ_2 are needed. Del Castillo proposed to use $P_0 = 10$ and $\widehat{\theta}_2^{(t)} = -\theta_1/2t_{prior}$, where t_{prior} is an initial guess of how many steps away the directional optimum is located. Although this guess needs to be specified by the user, its impact on the overall performance of the rule was reported to be more moderate than the effect of κ in the Myers-Khuri rule because of the relatively high value of P_0.

The value of P_t converges quickly to $\widehat{\sigma}^2_{(\theta_1 + 2\widehat{\theta}_2^{(t)}t)}$ (the difference is 1.04^{-5} after just three steps when $P_0 = 10$ is used) where:

$$\widehat{\sigma}^2_{(\theta_1 + 2\widehat{\theta}_2^{(t)}t)} = \frac{120t\widehat{\sigma}^2_\varepsilon}{(t+1)(2t+1)(3t^2+3t-1)} \approx 4\widehat{\sigma}^2_\varepsilon t^2 P_t \tag{6}$$

The approximation holds if it is assumed that $\widehat{\theta}_1 \equiv \theta_1$ and therefore $Var(\widehat{\theta}_1) = 0$.

The advantage of the recursive parabolic rule is its exhaustive use of all the information available from past and current experiments. However, its performance under non-quadratic responses has not been validated.

Scope of Current Research

The scope of this research is to propose an extended RLS stopping rule by making the following modifications to the recursive parabolic rule:

1 Recursively fit the intercept and the first order terms in addition to the second order terms in equation (2), to increase the robustness against non-quadratic behavior;

2 Use a coding scheme in order to obtain a near orthogonal design, thus reducing the bias and variance of the parameter estimates;

3 Allow the specification of a maximum number of experiments to be input in the RLS algorithm, from now on called a "rectangular window", in order to fit only a local model, less sensitive to large scale deviations from quadratic behavior.

3. AN ENHANCED RECURSIVE PARABOLIC RULE.

In this section we propose some modifications to the Recursive Parabolic Rule. However, we do not do this by enumerating them, but by presenting significant aspects of them. From these, the proposed modifications follow naturally.

Bias in the RLS parameter estimates

Let the parameter estimates for the i^{th} term in the model be denoted as $\hat{\theta}_i^{(t)}$ where $\hat{\theta}_0$ represents the intercept. In addition, let t denote the current step in the steepest ascent search.

If we use the recursive parabolic given in equations (3-5) scheme and the true function is:

$$Y(t) = \theta_0 + \theta_1 t + \theta_2 t^2 + \theta_3 t^3 + \theta_4 t^4 + \epsilon_t \tag{7}$$

then it can be shown that $E(\widehat{\theta_2})$ is given by:

$$E(\widehat{\theta}_2^{(t)}) = \theta_2 + \frac{5}{2}\frac{t(t+1)(2t^2+2t-1)\theta_3}{(2t+1)(3t^2+3t-1)} + \frac{5}{7}\frac{(3t^4+2t^3-3t+1)\theta_4}{(3t^2+3t-1)} \tag{8}$$

As it can be seen from equation (8) the bias due to third and fourth order terms increases with t, the number of experiments performed. A better fit of the true function can be obtained by increasing the number of parameters estimated in the model and the bias of the estimates can be reduced by using coded regressors centered at zero. In the first case, we are giving the model an increased ability to model changes in curvature. Clearly, when the true function is no longer quadratic the parameter estimates loss practical meaning, however fitting more than one parameter will surely yield a better local approximation of the true function whatever the type of relationship.

Regarding the coding of regressors, classical references in RSM explain how the bias in the parameter estimates is reduced by using a coding scheme. See for example (Box Draper 1987 and Myers Montgomery 1995). The single regressor used in the recursive parabolic rule can be coded using the sequence:

$$\{-\frac{t-1}{2}, -\frac{t-3}{2}, -\frac{t-5}{2}, \cdots, \frac{t-5}{2}, \frac{t-3}{2}, \frac{t-1}{2}\} \tag{9}$$

This is a sequence of t numbers centered at zero that provides a new scale . However, if this coding is used and only a second order coefficient is fitted recursively, then an estimate of the response at the midpoint in the original scale is needed to be used as intercept. This will be available when t is odd but will have to be interpolated when t is even. Furthermore, in most cases the intercept without coding comes from repeated "center points" performed in the previous factorial, and therefore it is a better estimate than a single observation. In addition, an estimate of the derivative at $\frac{t-1}{2}$ would also be needed as an estimate of the first order coefficient.

Instead of estimating recursively only the quadratic coefficient, we propose to fit recursively all three parameter estimates and to use the coding convention described by (9). This rule will be denoted by $R3N$. Notice that the estimates from the factorial conducted prior to starting the search can still be used as initial values for the RLS algorithm. The one-parameter-estimate model considered by Del Castillo will be denoted by $R1$.

The expected value of the vector of parameter estimates for the $R3N$ rule is presented in equation (10) where we assume again a fourth order polynomial (equation 7) represents the true function:

$$E(\widehat{\boldsymbol{\theta}}_{R3N}) = \begin{bmatrix} \theta_0 - \dfrac{3}{560}\,(t^4 - 10\,t^2 + 9)\,\theta_4 \\ \theta_1 + \dfrac{1}{20}\,(3\,t^2 - 7)\,\theta_3 \\ \theta_2 + \dfrac{1}{14}\,(3\,t^2 - 13)\,\theta_4 \end{bmatrix} \tag{10}$$

For the purpose of the stopping rule what is relevant is the bias in the estimate of the first derivative at the location of the last experiment. For the un-coded model this derivative is given by:

$$\frac{\partial \hat{Y}}{\partial t} = \widehat{\theta}_1 + 2t\widehat{\theta}_2 \tag{11}$$

and for the coded model this is given by:

$$\frac{\partial \hat{Y}}{\partial t} = \widehat{\theta}_1 + (t-1)\widehat{\theta}_2 \tag{12}$$

since $\frac{t-1}{2}$ is the position of the last run in coded units. Taking expectation in each case we get from (8) and (10):

$$E\left(\frac{\partial \hat{Y}_{R1}}{\partial t}\right) = \theta_1 + 2t\theta_2 + \underbrace{\frac{5t^2(t+1)(2t^2+2t-1)\theta_3}{(2t+1)(3t^2+3t-1)}}_{b_{R1,3}} + \frac{10}{7}\underbrace{\frac{t(3t^4+2t^3-3t+1)\theta_4}{(3t^2+3t-1)}}_{b_{R1,4}} \tag{13}$$

$$E\left(\frac{\partial \hat{Y}_{R3N}}{\partial t}\right) = \theta_1 + (t-1)\theta_2 + \underbrace{\frac{1}{20}(3t^2-7)\theta_3}_{b_{R3N,3}} + \underbrace{\frac{1}{14}(t-1)(3t^2-13)\theta_4}_{b_{R3N,4}} \tag{14}$$

Clearly, for large values of t the bias in the estimates of the derivative obtained from using $R1$ are much higher than those obtained from $R3N$. Figures 1 and 2 compare the bias between the estimates of the first derivative obtained from using $R1$ and $R3N$.

Variance of Parameter Estimates in RLS

An advantage of the $R1$ rule is that the variance of the first derivative comes only from one estimate. Therefore, a trade-off between bias and variance is made when more than one parameter is estimated, and this trade-off will have an impact on the performance of the stopping rule. Equations (15) and (16) give the variance of the first derivatives for the $R1$ and $R3N$ rules, respectively. Figure 3 presents a graphical comparison of these two variances.

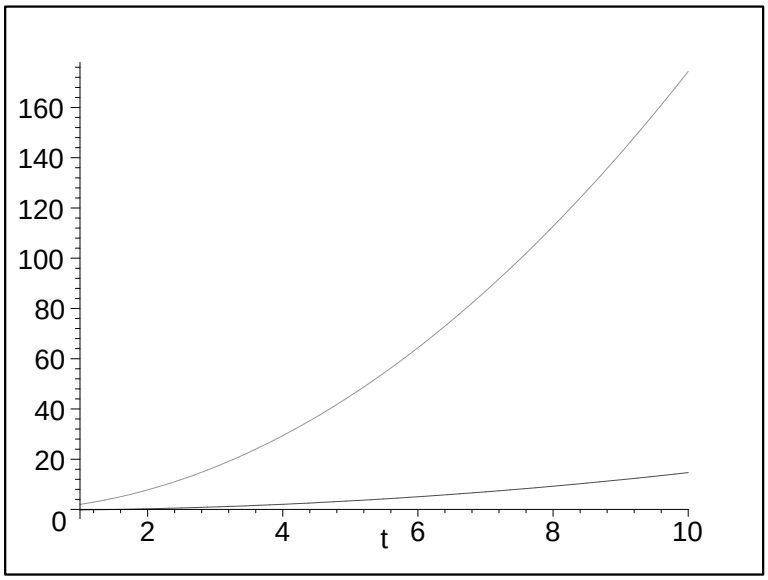

Figure 1: Bias due to third order terms in the first derivative of the quadratic model, lighter line is from $R1$ and darker line is from $R3N$ ($b_{R1,3}$ and $b_{R3N,3}$ in equations 13 and 14)

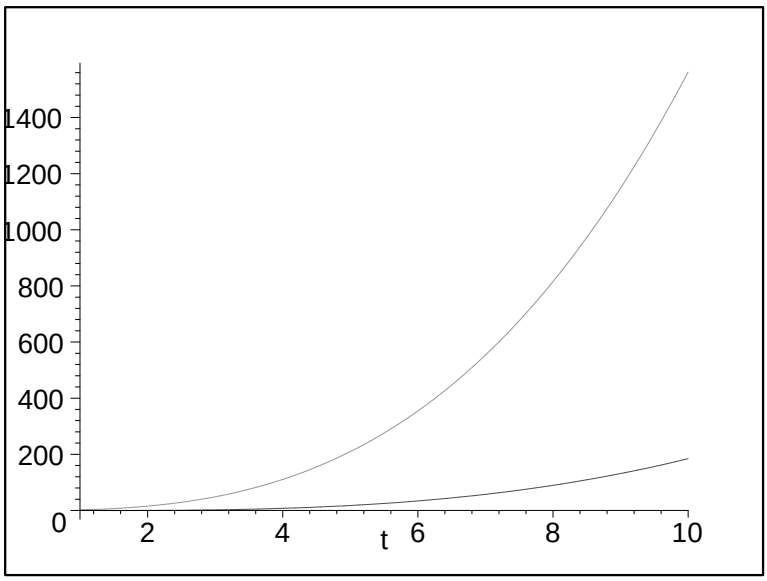

Figure 2: Bias due to fourth order terms in the first derivative of the quadratic model, lighter line is from $R1$ and darker line is from $R3N$ ($b_{R1,4}$ and $b_{R3N,4}$ in equations 13 and 14)

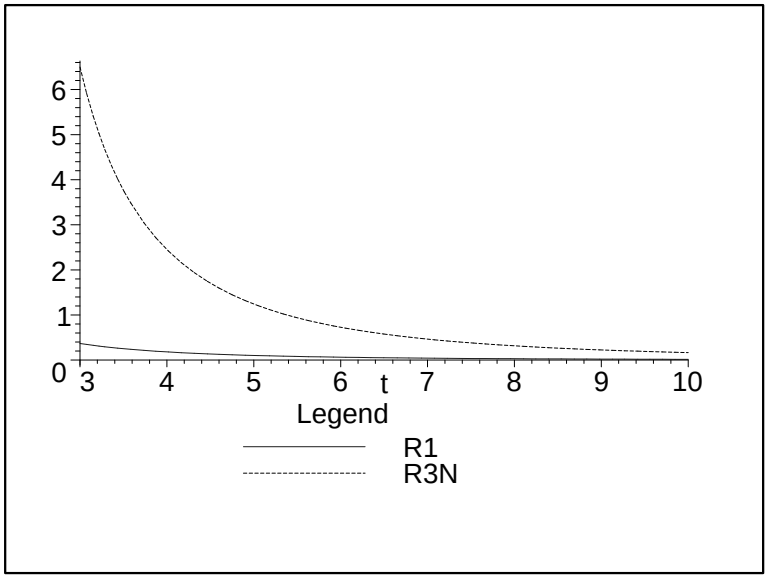

Figure 3: Comparison of the variance of the first derivative

$$Var\left(\frac{\partial \hat{Y}_{R1}}{\partial t}\right) = \frac{120t\sigma_\varepsilon^2}{(2t+1)(t+1)(3t^2+3t-1)} \tag{15}$$

$$Var\left(\frac{\partial \hat{Y}_{R3N}}{\partial t}\right) = \frac{12(2t-1)(8t-11)\sigma_\varepsilon^2}{(t-1)(t-2)(t+2)(t+1)t} \tag{16}$$

As it can be seen, the variance is higher for the $R3N$ rule, specially for small t, thus reducing the sensitivity of the test when the maximum is located close to the starting point.

In both of the proposed rules, the estimate of the intercept for the recursive model can be carried over without any mathematical manipulation. This intercept can be estimated from averaging center points or from the average of factorial points. In either case, the variance of the average utilized should be used in the initial P matrix of the RLS algorithm.

If the steepest ascent direction is chosen for the linear search, the initial estimate for the first order coefficient is given by the directional derivative in the gradient direction, i.e. the norm of the vector of parameter estimates $\|\widehat{\boldsymbol{\theta}}\|$ or $\sqrt{\widehat{\boldsymbol{\theta}}'\widehat{\boldsymbol{\theta}}}$. Because the operation of taking norm is non-linear the distribution of this statistic is rather complex.

Now, if an orthogonal first order design with n experiments was used to estimate a first order model in p parameters, then all the parameter estimates have the same variance, σ_ε^2/n. Therefore, if the errors are normally distributed we have that:

$$n\frac{\widehat{\boldsymbol{\theta}}'\widehat{\boldsymbol{\theta}}}{\sigma_\varepsilon^2} \sim \chi_p^2(\delta) \tag{17}$$

a non-central chi-square distribution with non-centrality parameter δ given by:

$$\delta = \sum_{i=1}^{n} \theta_i^2 = \boldsymbol{\theta}'\boldsymbol{\theta} \ . \tag{18}$$

The variance and mean of $n\widehat{\boldsymbol{\theta}}'\widehat{\boldsymbol{\theta}}/\sigma_\varepsilon^2$ are:

$$Var\left(\frac{n\widehat{\boldsymbol{\theta}}'\widehat{\boldsymbol{\theta}}}{\sigma_\varepsilon^2}\right) = 4\delta + 2p \tag{19}$$

and

$$E\left(\frac{n\widehat{\boldsymbol{\theta}}'\widehat{\boldsymbol{\theta}}}{\sigma_\varepsilon^2}\right) = \delta + p \ . \tag{20}$$

It can be observed that the variance and the expectation increase with the non-centrality parameter and the number of parameters fitted. However, if the rule for transformation of density functions is used (see, for example, Arnold 1990), it can be shown that the mean of $\sqrt{n\widehat{\boldsymbol{\theta}}'\widehat{\boldsymbol{\theta}}}/\sigma_\varepsilon$ is:

$$E\left(\frac{\sqrt{\widehat{\boldsymbol{\theta}}'\widehat{\boldsymbol{\theta}}}}{\sigma_\varepsilon}\right) = \sqrt{2}\, e^{(-1/2\,\delta)} \sum_{k=0}^{\infty} \frac{2^{(-k)}\, \delta^k\, \Gamma\left(\frac{1}{2}\,p + k + \frac{1}{2}\right)}{\Gamma(k+1)\, \Gamma\left(\frac{1}{2}\,p + k\right)} \tag{21}$$

Now, given that:

$$Var\left(n\frac{\sqrt{\widehat{\boldsymbol{\theta}}'\widehat{\boldsymbol{\theta}}}}{\sigma_\varepsilon}\right) = E\left(n\frac{\widehat{\boldsymbol{\theta}}'\widehat{\boldsymbol{\theta}}}{\sigma_\varepsilon^2}\right) - \left(E\left(n\frac{\sqrt{\widehat{\boldsymbol{\theta}}'\widehat{\boldsymbol{\theta}}}}{\sigma_\varepsilon}\right)\right)^2 \tag{22}$$

we have, substituting (20) and (21) into (22)

$$Var\left(n\frac{\sqrt{\widehat{\boldsymbol{\theta}}'\widehat{\boldsymbol{\theta}}}}{\sigma_\varepsilon}\right) = \delta + p - 2\, e^{(-\delta)} \left(\text{KummerM}\left(\frac{1}{2}\,p + \frac{1}{2},\, \frac{1}{2}\,p,\, \frac{1}{2}\,\delta\right) \frac{\Gamma\left(\frac{1}{2}p + \frac{1}{2}\right)}{\Gamma\left(\frac{1}{2}p\right)}\right)^2 \tag{23}$$

where the summation of equation (21) has been substituted by the KummerM function (Abramowitz and Stegun, 1972) a function readily available in computer algebra systems.

Despite the complexity of equation (23) Figure 4 indicates that it is less than one for finite values of δ and p . Therefore, regardless of the number of controllable factors and the size of their main effects, we can use the scaled variance (i.e. the variance divided by $\widehat{\sigma}_\varepsilon^2$) of the parameter estimates

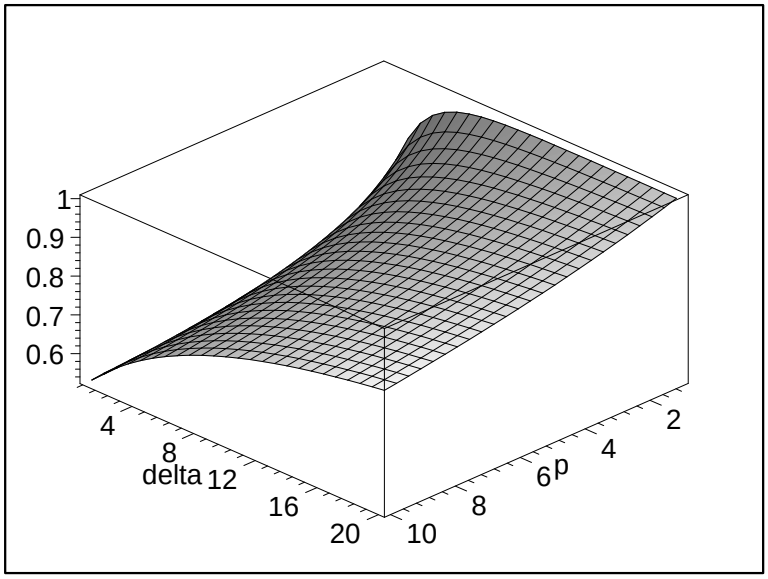

Figure 4: Variance of $\sqrt{\widehat{\boldsymbol{\theta}}'\widehat{\boldsymbol{\theta}}}/\sigma_\varepsilon$ as a function of the number of controllable factors and the non-centrality parameter

as starting values for the RLS algorithm without any risk of stopping before the maximum for this cause, since the probability of a Type I error (false rejections) will not be inflated by underestimating the variance. Another way of viewing this is that an initial low scaled variance will induce the RLS algorithm to give low weight to new observations. However, caution should be exercised in not specifying a too small variance because in such a case the initial estimate will vary little during the RLS updates rendering an inflexible model unable to locally approximate a non-quadratic response. Therefore, unless there is evidence that the true function resembles a quadratic function, we recommend to use a scaled variance of 1 as the starting value for the intercept and first order coefficient. Notice that this implies that the initial variance will be the estimate of σ_ε^2. We retain Del Castillo's suggestion of using 10 for the initial scaled variance of the second degree coefficient.

We point out that under the assumption of a true quadratic response, this result strengthens the approach of using the directional derivative as an estimate of the first order parameter in the $R1$ rule. The results imply that, whatever the variance of the parameter estimates, we can assure that the variance of the directional derivative will be smaller.

4. RECURSIVE LEAST SQUARES WITH A RECTANGULAR WINDOW.

It was mentioned in the introduction that classical RSM assumes the validity of second order

approximations only in a local region of the space of regressors. In addition, it was seen in equations (8) and (10) that the bias of the parameter estimates increases with t. Therefore, reducing the number of observations used to fit the second order model should increase the accuracy of the estimates, making the procedure less sensitive to non-constant curvature. In case the curvature is non-constant, a further modification to the recursive parabolic rule can be implemented as explained in what follows.

Goodwin and Payne (1977) suggested to use a "Rectangular Window" Recursive Least Square Algorithm (RWRLS) in order to track time-changing parameters in adaptive control applications. The algorithm consists of discarding the last observation once a new observation has been made in order to maintain their number constant. The same kind of algorithm can be used to fit a local model during a steepest ascent search. In such a case, the change in the parameters allows to model possible changes in curvature due to non-quadratic behavior.

Notice that if the RWRLS algorithm is used in conjunction with the coding convention of equation (9), i.e. the proposed $R3N$ rule, the matrix of regressors remains fixed at:

$$\mathbf{X}_{R3N} = \begin{bmatrix} 1 & -\frac{(N-1)}{2} & \frac{(N-1)^2}{4} \\ 1 & -\frac{(N-3)}{2} & \frac{(N-3)^2}{4} \\ \vdots & \vdots & \vdots \\ 1 & \frac{(N-3)}{2} & \frac{(N-3)^2}{4} \\ 1 & \frac{(N-1)}{2} & \frac{(N-1)^2}{4} \end{bmatrix} \tag{24}$$

where N is the number of observations used to fit the model and $\mathbf{X}_{R3N}$ denotes the design matrix for the $R3N$ rule. Therefore, it is not necessary to use the rank 1 update of the covariance matrix typical of RLS algorithms. The expressions given in equations (16) and (10) for the variance and bias of the first derivative apply to this scheme just by changing the step counter t for the window size N.

The size of the window must be selected in order to make a compromise between bias and variance. A large window will use much of the information available giving a very powerful test for negativeness of the derivative, although the estimates used will be highly biased due to the wide range over which the function is being fitted. In the presence of negative third and fourth order terms this will lead to stopping before the optimum is reached.

The size of the window can be selected by indicating a desired power for the test. The power function for an $\alpha = 0.05$ test of the null hypothesis $H_o : \frac{\partial Y}{\partial t} > 0$ versus an alternative $H_a : \frac{\partial Y}{\partial t} \leq 0$

is given by:

$$K_{R3c}(\Delta) = P_{\Delta}\left(\frac{\partial \hat{Y}}{\partial t} < -1.645 Stdev\left(\frac{\partial \hat{Y}}{\partial t}\right)\right) \tag{25}$$

where Δ is the true mean of $\frac{\partial \hat{Y}}{\partial t}$ under the assumption of a quadratic response. Now, under $H_a : \Delta \leq 0$ we have that:

$$K_{R3c}(\Delta) = P_{\Delta}\left(\frac{\frac{\partial \hat{Y}}{\partial t} - \Delta}{Stdev\left(\frac{\partial \hat{Y}}{\partial t}\right)} < -1.645 - \frac{\Delta}{Stdev\left(\frac{\partial \hat{Y}}{\partial t}\right)}\right) \tag{26}$$

Assuming again a quadratic response we have that:

$$E\left(\frac{\partial \hat{Y}}{\partial t}\right) = \Delta \Rightarrow \frac{\frac{\partial \hat{Y}}{\partial t} - \Delta}{Stdev\left(\frac{\partial \hat{Y}}{\partial t}\right)} = Z \sim N(0,1) \tag{27}$$

If in equation (26) we substitute $Stdev\left(\frac{\partial \hat{Y}}{\partial t}\right)$ for the squared root of equation (16) and change the t's for N's (i.e. the window size), we get:

$$1 - \beta = \Phi\left(-1.645 - \frac{\Delta}{\sigma_{\varepsilon}}\sqrt{\frac{(N-1)(N-2)(N+2)(N+1)N}{12(2N-1)(8N-11)}}\right) \tag{28}$$

where β is the probability of a Type II error and $1 - \beta$ is the power of the test. Given values of $1 - \beta$ and Δ, equation 28 may be solved to obtain a window size. By doing this we guarantee the minimal window size for a given power and, therefore, the bias due to higher order effects will be reduced (see equation 14).

For example, suppose we want to have a 90% probability of rejecting when the true derivative (Δ) is -4 with an estimate of the white noise standard deviation of 2. Then we will have to solve:

$$0.1 = \Phi\left(-1.645 - \frac{-4}{2}\sqrt{\frac{(N-1)(N-2)(N+2)(N+1)N}{12(2N-1)(8N-11)}}\right) \tag{29}$$

which implies that:

$$-1.645 + 2\sqrt{\frac{(N-1)(N-2)(N+2)(N+1)N}{12(2N-1)(8N-11)}} = -1.282 \ . \tag{30}$$

This can be solved numerically to give, after rounding off, $N \approx 7$. However, notice that in practice it could be difficult to come up with suitable values of Δ. Therefore, assuming that the true function has at least some symmetry in the search direction, we recommend that the value of

Δ be determined as a percentage of the norm of the gradient at the beginning of the search. That is:

$$\Delta = \alpha \parallel \widehat{\boldsymbol{\theta}} \parallel \tag{31}$$

where α is a number between 0 and 1 decided by the practitioner. Therefore we are guaranteeing with some probability the rejection of the null hypothesis when the response is dropping at a percentage of the rate at which it was increasing at the beginning of the search.

The RWRLS can only be started once N observations have been collected. Before this, a regular RLS scheme can be used. The initial estimates for the intercept and first order coefficient will be computed as in the $R1$ rule. However they will all be updated at each iteration. Once N steps have been performed the procedure will switch to the RWRLS. Notice that by the coding convention, the amount of computations required actually reduces after switching, since the matrix of regressors becomes fixed.

5. SIMULATION EXPERIMENTS.

This section presents the results obtained from two types of simulated RSM experiments using the different stopping rules mentioned in section 2. In the first experiment, a normally distributed white noise sequence is used to simulate the observations. In the second experiment, a leptokurtic distribution (thick tails) is used to increase the probability of outliers.

Three different polynomial test functions in five controllable factors were used. Of these, two were generated using a recently developed RSM testbed (McDaniel and Ankeman 2000). They were called HH21 and LL21 and differed in the level of curvature introduced in their generation, with the HH21 function being the most curved one. The appendix contains a brief explanation of the procedure used to generate the functions from the testbed. For further details the reader is referred to McDaniel and Ankeman (2000). The third test response was a quartic polynomial model. All the simulations were conducted in MATLAB version 5.3.1.

Normal Noise Simulations

Simulations were conducted according to the following general steps:

1. The global maximum of each surface was computed using MATLAB's "fmincon" command;

2. A starting point was randomly selected from a hyper-sphere of specified radius centered around the global maximum of the surface under consideration;

3. A 2_{IV}^{5-1} factorial design with four center points was run centered at the starting point obtained in step 2, and a main-effects-only model was fitted;

4. The true maximum in the direction of the gradient of the first order model (called t_{max}) was computed using a uniform search and the true value of the polynomial at this point (called Y_{max}) was evaluated;

5. A steepest ascent search was conducted in the same direction as step 4 and a given stopping rule was used to determine when to stop, this point was called t_{stop} and the true response at it Y_{stop};

In addition to the Myers-Khuri (MK) and the parabolic rules ($R1$ and $R3N$), the classical $FirstDrop$, the $2-in-a-row$ and the $3-in-a-row$ stopping rules were tested as well. The other parameters of the simulations were:

1. The distance from the starting point to the global maximum, or radius. Three levels were used: 10, 20 and 30 units were tested;

2. The standard deviation of the white noise, σ_ε. Four levels were used: $1\%, 5\%, 10\%$ and 20% of the potential improvement in the search direction. Hence, this differs for each simulated search. The percentages will be called "noise levels" for the remaining of the paper.

3. An a priori guess on the number of steps to the maximum, called κ. As mentioned before this is required by the Myers and Khuri rule and by Del Castillo's recursive rule. Ten levels were used: The correct value without any error and with bias between -80% and 100%.

One thousand replications were performed for each set of conditions. Within each replication, the seed of the random number generator was reset to the same value after applying each stopping rule.

Two performance measures were used to evaluate the stopping rules. The first one was the mean squared difference (MSD) around the true maximum in the search direction:

$$MSD = \sum_{Replicate=1}^{1000} \frac{(t_{max} - t_{stop})^2}{1000} \tag{32}$$

This performance measure penalizes if the rule used has a very high variability in its stopping point or if it systematically stops after or before the true maximum. Unless experiments are extremely expensive, in most cases a rule that stops after the maximum will be considered better than a rule that stops before, because a better estimate of the true maximum will be obtained. To evaluate this from a practical point of view, the following percentage of improvement will be used.

$$\%Improv = \frac{Y_{start} - Y_{stop}}{Y_{start} - Y_{max}} \tag{33}$$

where $\%Improv$ is the percentage improvement and Y_{start} is the true value of the response at the starting point.

The other parameters of the $R3N$ rule are α and the power $1 - \beta$. Unfortunately, there is no theoretical foundation to specify the power of the test for any given drop in response. Notice that this is the same kind of problem typically encountered in designing quality control charts. To provide some justification of the values chosen, an investigation was done to determine the effect of changing these parameters in the $R3N$ stopping rule. A distance from the maximum of 10 units (radius=10) and the quartic polynomial response were chosen for this experiment. These results are shown via box & whiskers plots of the differences $t_{max} - t_{stop}$ in Figure 5. The box represents 75% quartiles, the whiskers 95% and the cross represents the median of the differences obtained from 1000 replications.

It can be seen from Figure 5 that for very small levels of noise the power and the parameter α have no effect in the stopping point, since the procedure always picks the minimum window size of 3 experiments. For high levels of noise the results are again very similar since the windows are so large than in most of the situations the algorithm never reaches the window size, and therefore never switches from a standard RLS to the RWRLS.

For the two intermediate levels of noise it is noticed that the algorithm presents a tendency to short-stop only when α is small and the power is high, these are the cases where the window size is the biggest.

Therefore it is seen that there is plenty of flexibility in the selection of α and power as long as α is not too small and the power is not too high. Notice that in no case the median of the differences was never negative. A power of 0.8 and a value of α of 0.4 were selected for the remainder of the comparison experiments.

The results of $\%Improv$ for the quartic polynomial are presented in Figure 6. It can be seen that the $R1$ and $FirstDrop$ rules yield the lowest improvement in the mean response. Notice

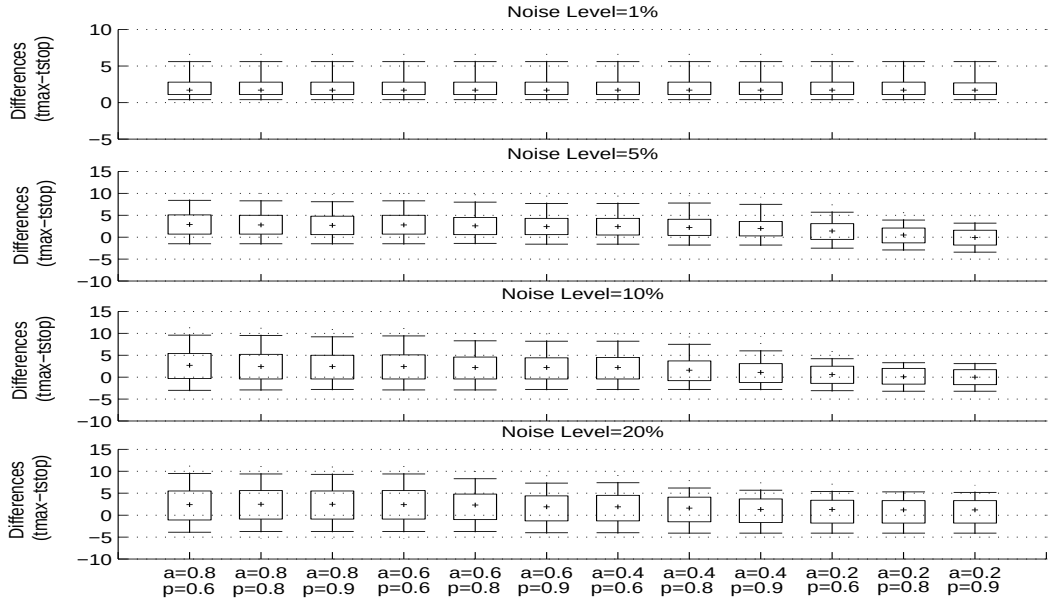

Figure 5: Box & Whiskers plot of differences using the $R3N$ rule for various combinations of noise level, power and α (a is α and p is the power)

how almost every rule gives perfect results for the lowest noise level except the $R1$ rule. However, the performance of this rule improves with increasing noise level while the performance of the $FirstDrop$ rule worsens. It is also seen that the $R3N$ rule performs as good or better than the $R1$ rule in every case. Comparing the MK and the $R3N$ rules we observe that MK slightly outperforms $R3N$. However, this difference is small in most of the cases.

For the differences $t_{max} - t_{stop}$ we have included Tables with the numerical values of their MSD as well as box & whisker plots. As we can see in Table 1, the $R3N$ and the MK rule have similar MSD's. However, it appears that the MK rule slightly outperforms the $R3N$ rule for low levels of noise, while the $R3N$ rule slightly outperforms the MK rule for noise levels greater than 1%. This pattern repeats for all the radii. The $2-in-a-row$ rule behaves very well stopping close and after the maximum except for the cases the search starts far from the optimum (large radius) and the noise level is 20%. The $3-in-a-row$ rule has a clear tendency of stopping after the maximum.

Finally, it is seen in Figure 7 that the MK rule has more variability in the differences $t_{max} - t_{stop}$ than the $R3N$ rule for high levels of noise, while the $R3N$ rule has a slight tendency to short-stop more often than the MK rule. Hence the slightly lower $\%Improv$.

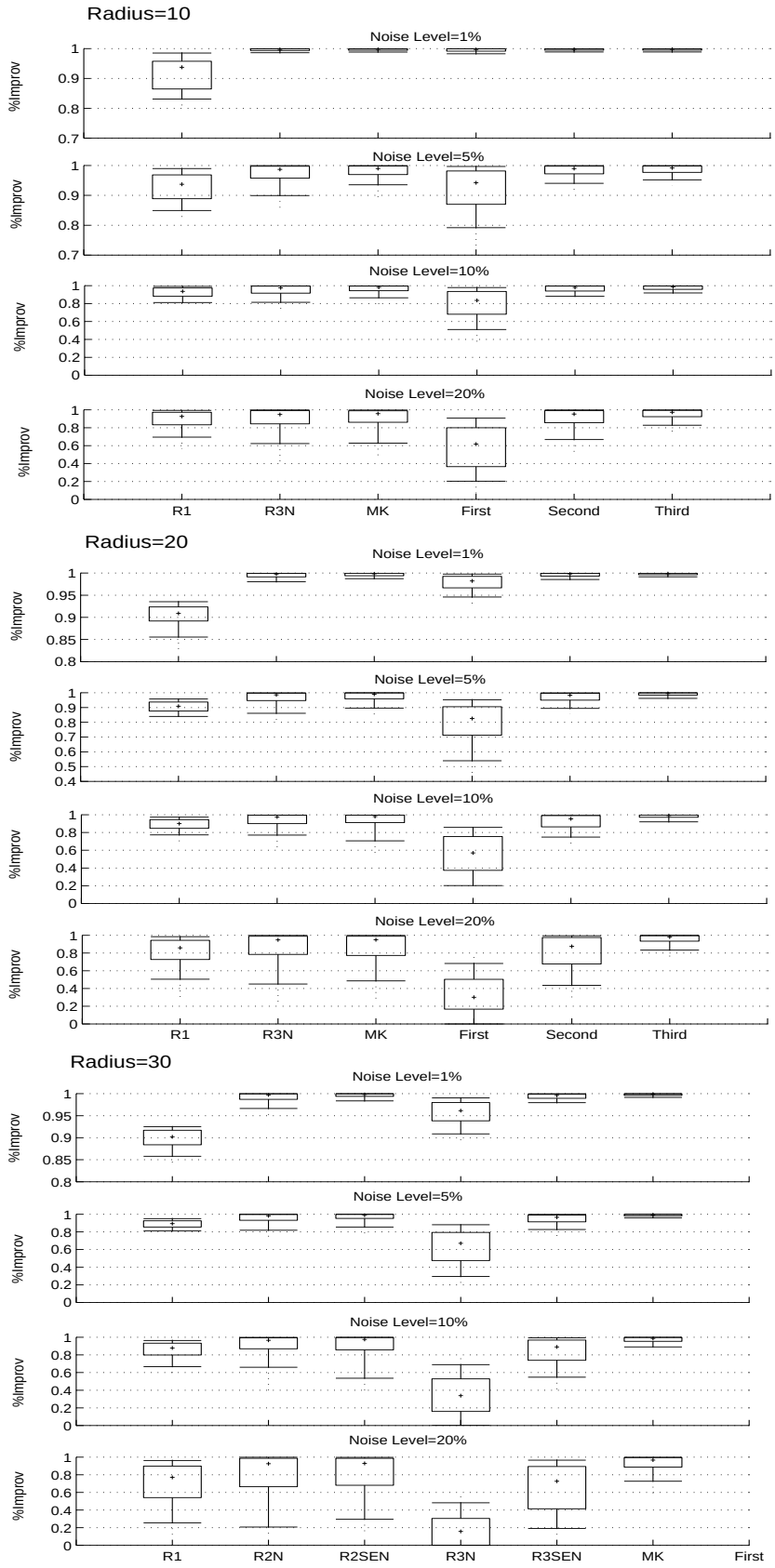

Figure 6: $\%Improv$ computed from 1000 simulations using the quartic polynomial

Table 1: Mean Squared Differences for the Quartic Polynomial

$radius = 10$						
Noise Level	$R1$	$R3N$	MK	1^{st}	2^{nd}	3^{rd}
1%	12.89	8.84	7.19	1.33	8.23	19.55
5%	9.82	22.16	16.96	7.15	13.31	41.14
10%	7.92	21.32	24.57	15.21	15.21	55.94
20%	7.39	20.20	36.02	25.65	19.52	79.51
$radius = 20$						
1%	79.2	35.83	30.62	23.87	14.61	46.55
5%	67.71	63.78	71.15	98.96	26.75	88.21
10%	58.2	71.56	102.27	146.09	41.28	129.07
20%	54.24	92.58	153.22	164.96	61.07	200.53
$radius = 30$						
1%	194.79	77.51	64.12	103.16	26.81	72.49
5%	170.13	136.17	169.68	331.99	87.23	157.29
10%	152.3	177.4	263.68	406.42	142.75	221.93
20%	172.48	207.2	324.64	406.06	192.04	301.76

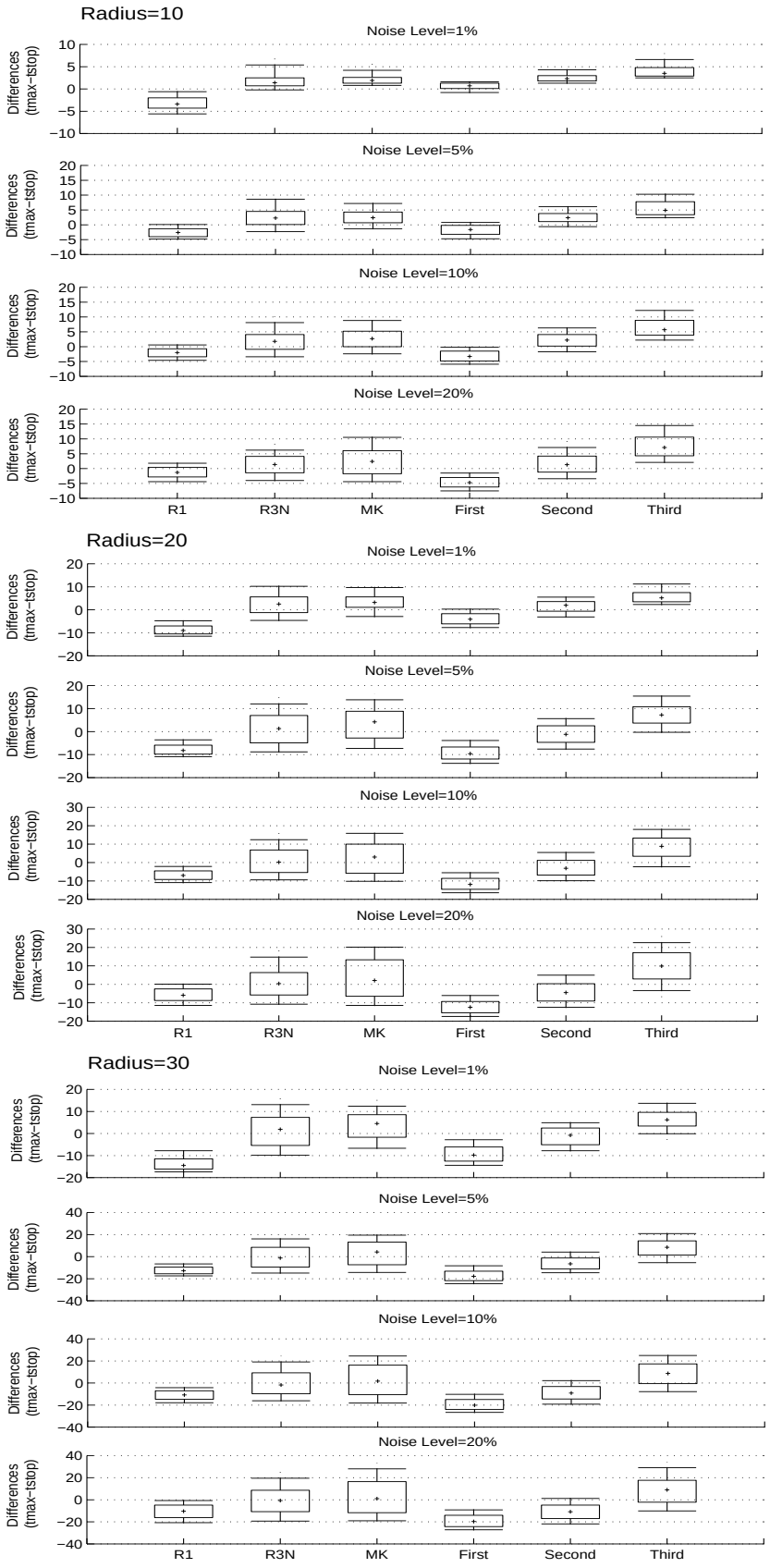

Figure 7: Differences computed from 1000 simulations using the quartic polynomial

Table 2: Mean Squared Differences for the HL21 response

$radius = 10$						
Noise Level	$R1$	$R3N$	MK	1^{st}	2^{nd}	3^{rd}
1%	1.12	2.40	3.21	1.31	4.90	10.30
5%	2.36	7.03	8.92	4.55	9.21	22.87
10%	4.76	9.82	16.71	16.72	11.99	38.39
20%	8.17	17.56	28.43	30.64	15.85	73.26
$radius = 20$						
1%	6.79	10.99	9.39	2.56	8.51	20.07
5%	13.00	26.34	42.42	117.19	19.89	77.02
10%	32.66	43.94	87.46	176.60	48.46	136.72
20%	52.39	84.15	134.47	164.70	71.27	256.50
$radius = 30$						
1%	27.09	28.83	22.46	25.23	12.82	40.09
5%	54.95	74.35	136.10	416.20	85.90	157.35
10%	132.31	140.24	240.93	431.86	179.87	266.59
20%	175.27	235.55	333.85	347.36	197.81	364.03

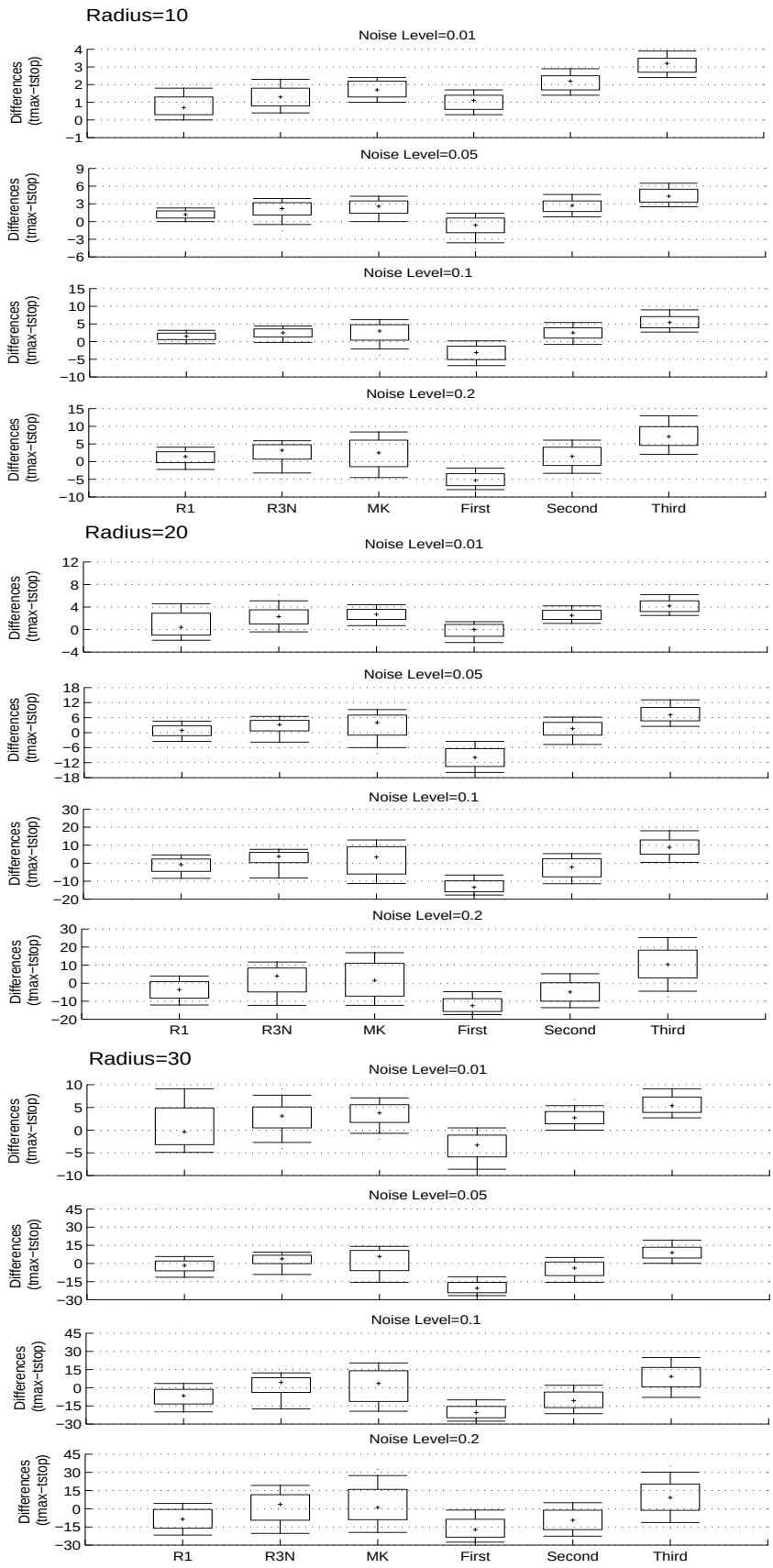

Figure 8: Differences computed from 1000 simulations using the function HL21

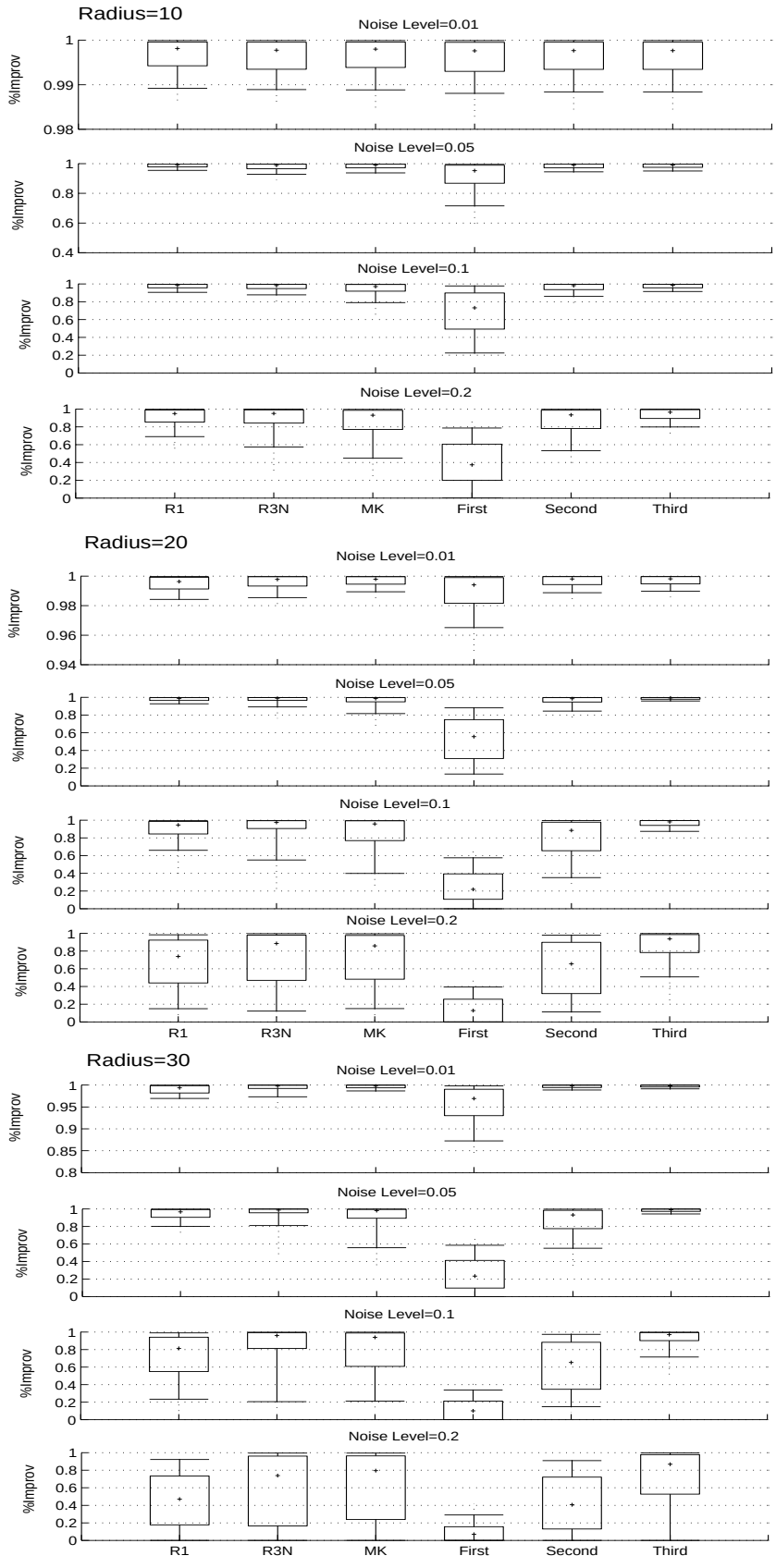

Figure 9: $\%Improv$ computed from 1000 simulations using the function HL21

The results of $\%Improv$ for the HL21 function are showed in Figure 9. Notice how the $R1$ rule behaves better in this case, particularly when the noise level is low, since the surface resembles more a quadratic polynomial. Nonetheless, for long searches it still suffers from short-stopping. For this function the $R3N$ rule outperforms the MK rule in almost every case, except the ones with the highest noise level and radius 20 and 30.

The MSD's for the HL21 function are presented in Table 2 and in Figure 8. In the plots it can be observed that the $R3N$ rule showed a consistent desirable behavior, having a great proportion of stops just after the maximum and low variance. Again it is seen that the $R1$ rule performs well for the short searches (small radius). Notice how when it stops in the right place it is typically the one with the minimum variance. Again, the MK rule gave best results for the cases with low level of noise and short searches.

The results for the LL21 function are showed in Table 3 and in Figures 10 and 11. For this function the $R3N$ rule gave better or similar $\%Improv$ values than the MK rule. However, in some situations it was outperformed by the $R1$ rule. Notice that the $R3N$ rule was not the worst in any case. As in previous cases the $3 - in - a - row$ rule gave the highest $\%Improv$, since it systematically stops after the maximum.

In Table 3 it is seen that again the MK rule has lower MSD value than the $R3N$ rule only for some of the cases where the noise level is low. In addition, the tendency to short-stopping is less pronounced in the $R3N$ rule for all the cases except for the sigma=0.01, radius=30 case.

Simulations Under Non-Normal Noise

Additional simulations were carried over in the same way as those in the previous section with the exception that the noise in the observations was sampled instead from the following distribution:

$$\epsilon \sim \begin{cases} U(-6,-3) & if\ r < 0.1 \\ N(0,1) & if\ 0.1 \leq r \leq 0.9 \\ U(3,6) & if\ r > 0.9 \end{cases} \tag{34}$$

where $r \sim Unif(0,1)$ and ϵ is the noise added to the respective polynomial to obtain the observations. The $\epsilon's$ were multiplied by the same values as in the normal case. Notice that because ϵ does not have variance one, the variance in this case is inflated with respect to the normal case.

Now, equation (34) gives a symmetric, leptokurtic distribution. To create negative and positive skewed distributions the following was used:

Table 3: Mean Squared Differences for the LL21 response

$radius = 10$						
Noise Level	$R1$	$R3N$	MK	1^{st}	2^{nd}	3^{rd}
1%	0.69	1.75	2.58	1.37	4.84	10.13
5%	1.88	5.17	6.36	2.45	7.86	17.57
10%	3.46	6.63	10.36	7.36	9.46	27.34
20%	7.01	12.53	15.75	15.53	14.16	52.21
$radius = 20$						
1%	2.54	7.12	6.50	1.70	6.87	15.32
5%	6.65	17.56	28.78	54.70	15.45	51.28
10%	17.70	28.86	56.75	94.55	27.96	107.23
20%	33.63	56.80	91.44	103.09	43.99	175.98
$radius = 30$						
1%	9.73	24.02	16.23	12.98	10.51	30.33
5%	28.07	46.10	76.77	200.90	39.86	102.60
10%	64.23	80.45	149.11	255.40	95.17	196.59
20%	116.34	181.29	242.39	248.91	132.92	317.43

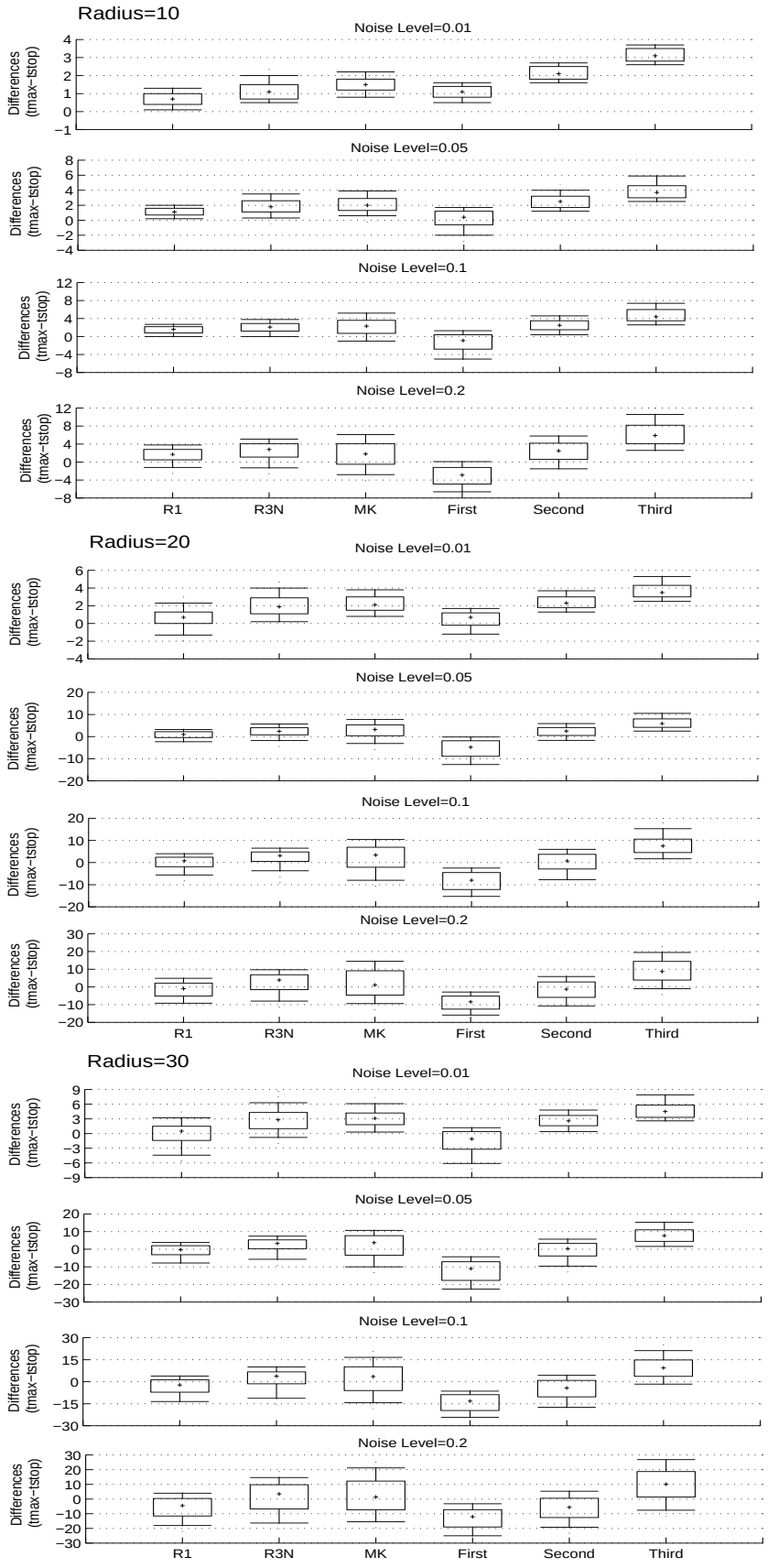

Figure 10: Differences computed from 1000 simulations using the LL21 polynomial

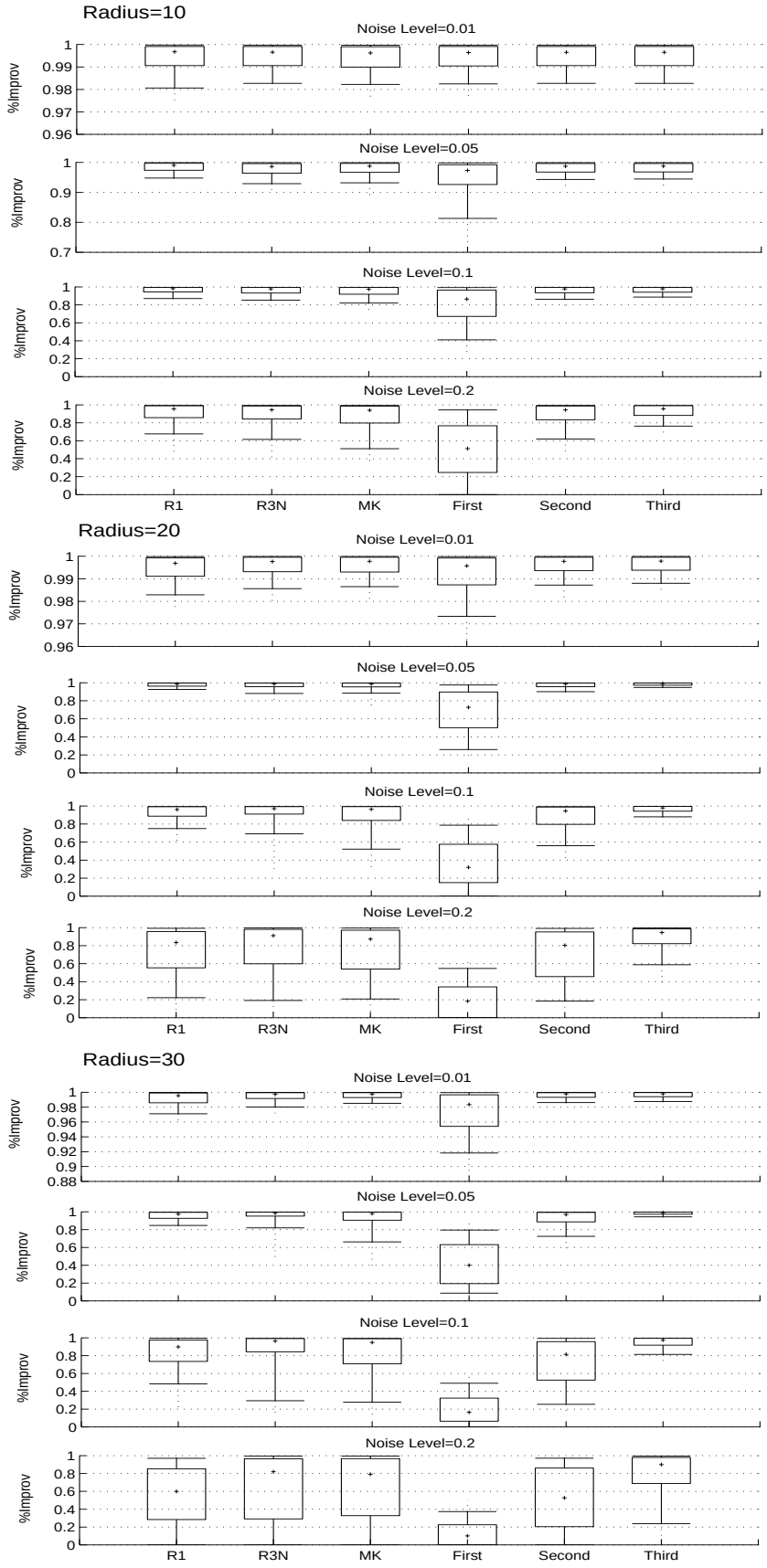

Figure 11: $\%Improv$ computed from 1000 simulations using the LL21 polynomial

Table 4: Mean Squared Differences for Quartic Polynomial (Non-Normal Noise)

Symmetric						
Noise Level	$R1$	$R3N$	MK	1^{st}	2^{nd}	3^{rd}
1%	11.70	16.30	13.80	2.55	10.06	28.13
5%	8.56	25.21	34.19	16.99	14.98	59.35
10%	8.36	25.95	49.15	22.96	18.71	75.19
20%	11.22	42.28	71.02	26.44	20.24	108.67
Positive Skewness						
1%	11.74	18.14	14.58	2.42	10.43	27.53
5%	7.75	26.73	33.12	16.03	14.81	51.67
10%	5.96	29.93	52.27	22.62	17.76	69.77
20%	7.27	48.96	70.05	27.49	20.15	104.27
Negative Skewness						
1%	12.80	15.94	14.56	2.47	10.59	27.52
5%	9.99	20.93	36.32	16.33	14.92	54.43
10%	9.48	23.19	56.73	24.40	19.72	73.62
20%	12.37	35.47	84.06	25.75	20.40	103.62

$$\epsilon \sim \begin{cases} N(0,1) & if\ r < 0.8 \\ U(3,6) & if\ r > 0.8 \end{cases} \tag{35}$$

$$\epsilon \sim \begin{cases} U(-6,-3) & if\ r < 0.2 \\ N(0,1) & if\ r > 0.2 \end{cases} \tag{36}$$

where (35) gives a positively skewed distribution while (36) gives a negatively skewed distribution. The results for these three distributions using the quartic polynomial and a radius of 10 units are presented in Table 4.

The $R1$ rule appears to be the more robust rule against deviations from the normal assumption, it even outperforms the $R3N$ and MK rules for some cases. However, this rule still presents some problems of short-stopping.

As expected, the performance of the MK rule deteriorates more by the relaxation of the normal

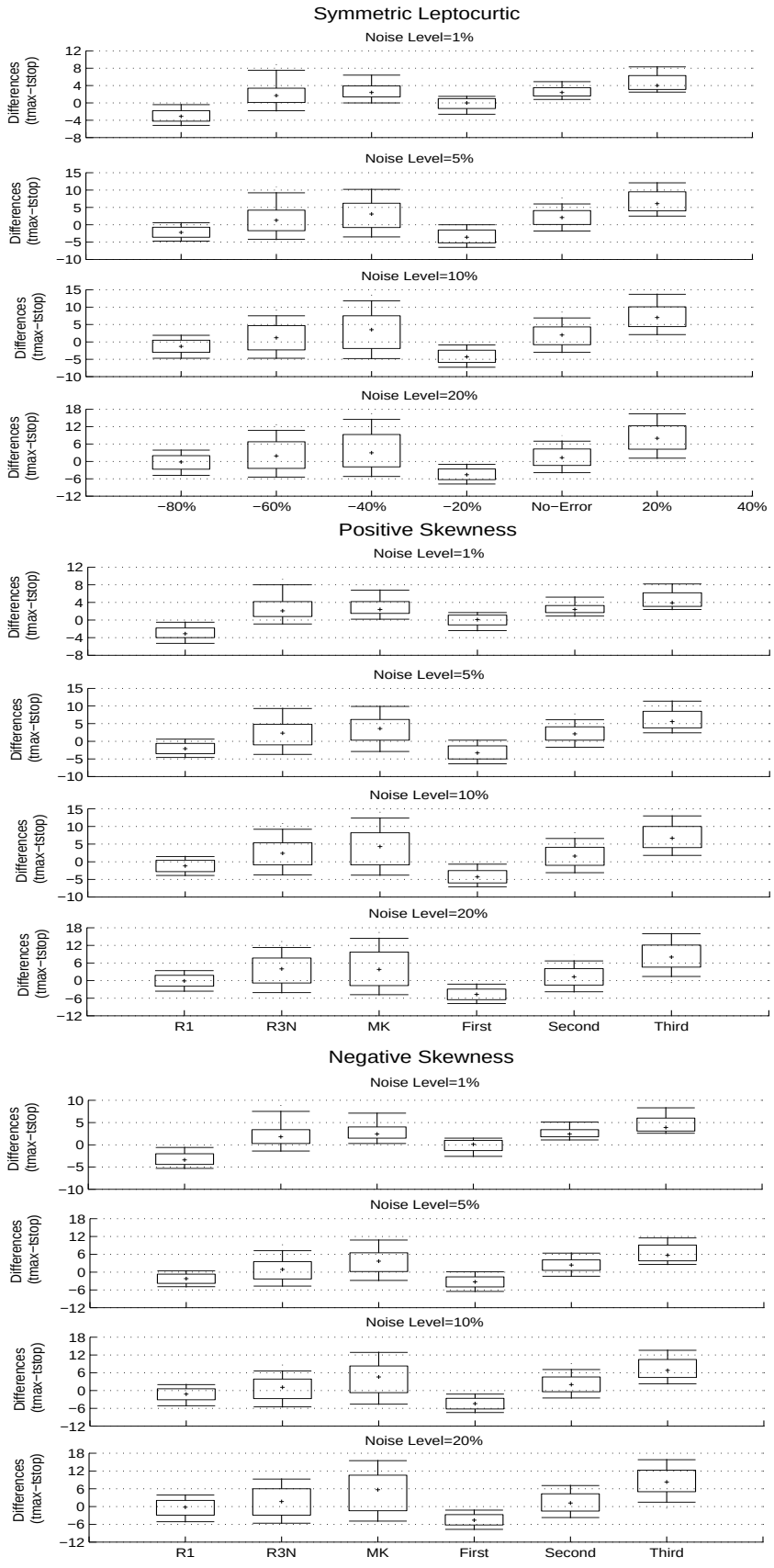

Figure 12: Differences computed from 1000 Non-normal noise simulations using the quartic polynomial and radius=10.

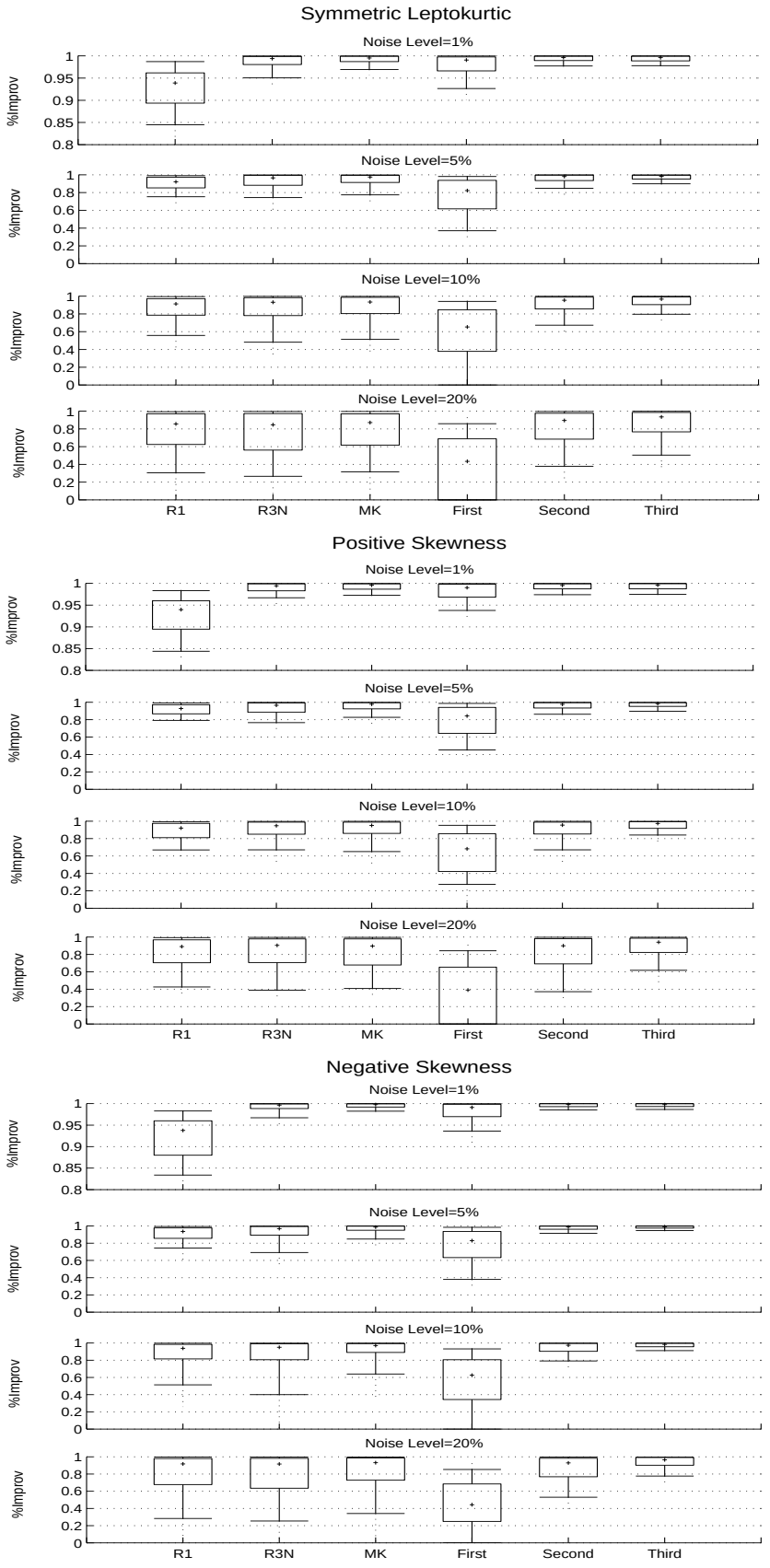

Figure 13: $\%Improv$ computed from 1000 Non-normal noise simulations using the quartic polynomial and radius=10.

noise than for the $R3N$ rule. A possible explanation for this behavior relies in the fact that OLS estimates are still BLUE under non-normal distributions while as it can be seen in the derivation of the MK rule, it seriously relies on the normality assumption. The impact on the $R3N$ rule probably comes from the test of negativeness of the derivative, which is a function of the no longer normally distributed parameters. However, notice that the distribution of the parameters converges to normal as the number of experiments increases. Because the $R1$ rule uses more runs it is more robust to departures from the normal assumption.

Effect of Uncertainty in MK's rule κ parameter

As it was mentioned before, the MK stopping rule requires that the user specifies a parameter called κ which represents an initial guess on how many steps away is the optimum in the steepest ascent search. Clearly this information will not be available in most of the cases, since it is precisely the reason why the search is being performed.

In the previous simulation experiments the true value of κ was used in the searches, which represented an advantage for the MK rule in the comparisons. To asses the impact that uncertainty in this parameter has on the performance of the rule, simulations were conducted introducing a systematic error. The plots in Figure 14 contain the results obtained when the value of κ used differed from -80% to 100% from the true value. It is seen that sub-estimating κ incurs in short-stopping, while overestimating κ slightly increases the variance. However, only when the sub-estimation is greater than 40% this has a significant effect on the improvement. Notice also that for the case of smallest noise level all the stops where done after the maximum, regardless of the amount of bias in the estimate of κ. In general it is seen that overestimating κ reduces the performance slightly and therefore a user should not be too cautious when specifying this parameter in actual applications. This agrees with the recommendations in Del Castillo (1997) who studied the performance of this rule under a quadratic function.

6. CONCLUSIONS.

1. The recursive parabolic rule may stop short of the optimum under non-quadratic responses.
2. By recursively estimating all the quadratic model and implementing a "Rectangular Window", the sensitivity of the recursive rule to non-quadratic responses may be considerably reduced.
3. The performance of the proposed rule and the MK rule was not considerably different under

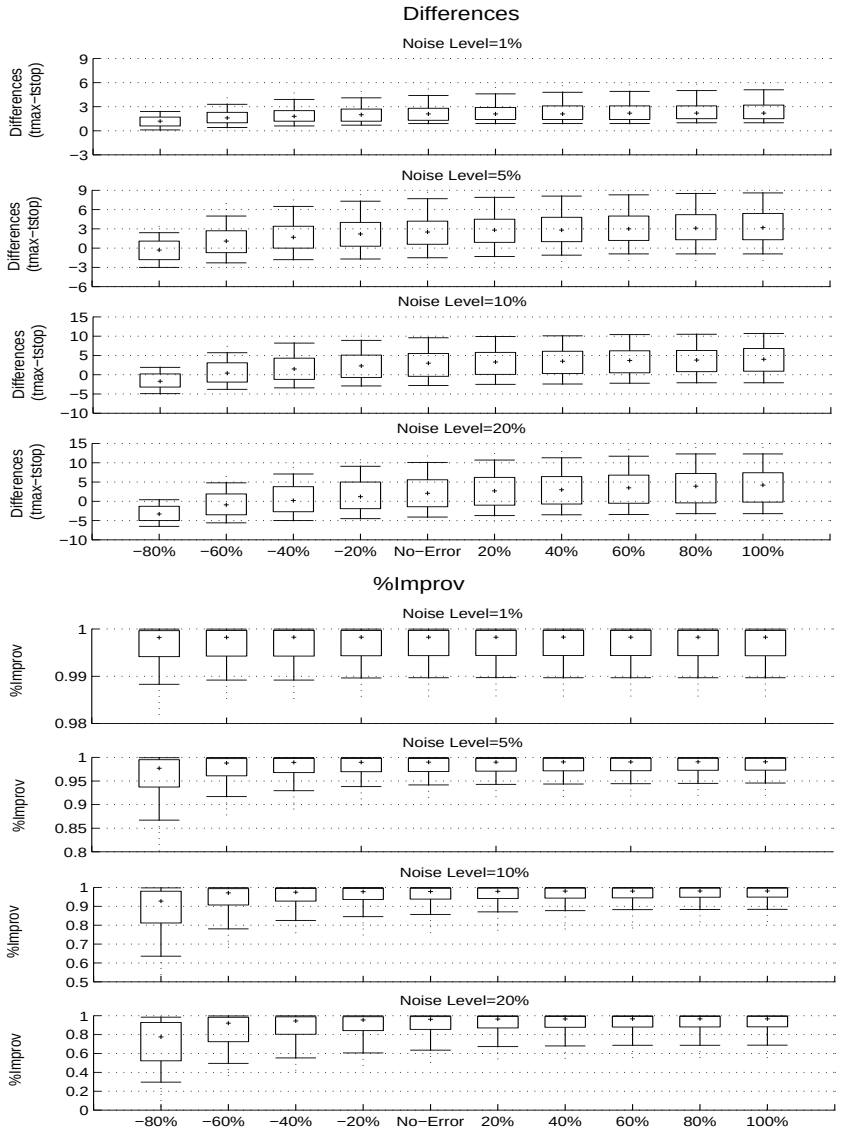

Figure 14: Effect of Bias in κ on the Performance of the MK rule. Top: Differences; Bottom: $\%Improv$

most of the conditions studied. However, the proposed rule is more robust to departures from the normality assumption and does not require an initial estimate of the number of steps to reach the optimum.

4. The performance of the $FirstDrop$ rule was poor in almost all the conditions studies and its use is strongly discouraged. The performance of the $3-in-a-row$ rule demands too many experiments and it is only recommended if the experiments are not expensive to perform. The performance of the $2-in-a-row$ rule was surprisingly good, although not as good as the MK and the enhanced recursive rule.

APPENDIX: TEST FUNCTIONS FROM RSM TEST BED.

Here we present a brief explanation of the procedure used to obtain the HL21 and LL21 polynomials. For a more complete description of the procedure the reader is referred to (McDaniel and Ankeman 2000).

The RSM testbed requires the specification of a "S" and "T" matrices and a "flatness" index. The S matrix controls the presence of main effects and their powers, that is the presence of the $x_i^{l_i}$ terms, where i represents the i^{th} controllable factor and l_i is the order (power at which x_i is raised) of the term in the polynomial. The T matrix controls the presence of two and three order interactions, $x_i^{l_i}x_j^{l_j}x_k^{l_k}\ \forall i \neq j \neq k$. Currently the testbed only allows interactions such that $\sum_{r=1}^{n} l_r \leq 3$, where n is the total number of controllable factors in the polynomial. These matrices control the "form" of the final polynomial by specifying the probabilities of appearance of a given term.

The flatness index controls the values given to the coefficients once the form has been established by the T and S matrices. It should be at least equal to the inverse of the number of variables, i.e. $f \geq 1/n$. Larger values of f produce flatter surfaces.

Two different S matrices were used, one to produce a highly "bumpy" surface and the other one to produce a surface less bumpy. Only one T matrix were used two produce slightly "twisted" surfaces. The test bed was unable to to produce highly twisted convex surfaces.

The aforementioned three matrices were combined to produce two different function forms. The flatness was set to the lowest possible value of 0.21. The surface obtained with the low T and low S matrices was named LL21 and with the high S and low T was named HL21.

In every case a large number of functions were obtained and only the ones that had a finite maximum for an unbounded experimental region were selected. This was done to assure convergence

of the steepest ascent searches. Unfortunately, the test bed does not give any means for controlling the convexity of the generated functions. Although the functions used are not convex their Hessians are negative definite for points located sufficiently far away from the center and, therefore, they have a maximum in an unbounded region.

References

Abramowitz, M. Stegun I.A. (eds.) (1972). *Handbook of Mathematical Functions.* NY: Dover.

Arnold, S.F. (1990). *Mathematical Statistics.* Englewood Cliffs, NJ: Prentice Hall.

Box, G.E.P. and Draper, N.R. (1987). *Empirical Model-Building and Response Surfaces.* NY: Jon Wiley & Sons.

Box, G.E.P. and Wilson, K.B. (1951). "On the Experimental Attainment of Optimum Conditions," *Journal of the Royal Statistical Society*, B13, 1-38.

Del Castillo, E. (1997). "Stopping Rules for Steepest Ascent in Experimental Optimization," Communications in Statistics, Simulations, 26(4), 1599-1615.

Goodwin, G.C. and Payne, R.L. (1977). *Dynamic System Identification: Experiment Design and Data Analysis.* NY: Academic Press.

McDaniel, W. R. and Ankenman, B.E. (2000). "A Response Surface Test Bed," Quality and Reliability Engineering International, 16, 363-372.

Myers, R.H. and Montgomery, D.C. (1995). *Response Surface Methodology.* NY: Wiley Series in Probability and Mathematical Statistics.

Myers, R.H. and Khuri, A.I. (1979). "A New Procedure for Steepest Ascent," Communications in Statistics, Theory and Methods, A8(14), 1359-1376.

Wellstead, P.E. and Zarrop, M.B. (1991). *Self-Tuning Systems: Control and Signal Processing.* NY: Wiley.