MATHEMATICS OF COMPUTATION, VOLUME 34, NUMBER 150
APRIL 1980, PAGES 529–541

Nonlinear Curve-Fitting in the L_1 and L_∞ Norms

By Richard I. Shrager and Edward Hill, Jr.

Abstract. In extending the Levenberg-Marquardt L_2 method for nonlinear curve-fitting to the L_1 and L_∞ norms, the following problems arise, but are dealt with successfully:

(1) Trial parameters are generated by linear programming, which can be time-consuming.

(2) Trial parameters are not uniquely specified in some cases.

(3) There are intervals of the search parameter for which the trial parameters remain constant.

(4) In L_1, the trial parameters are discontinuous with respect to the search parameter.

It is shown that linear constraints on the parameters are easily included in the computations. Finally, some numerical results are presented.

I. Introduction. Given a set of independent variables

$$t = \{t_i, i = 1 \text{ to } m\}$$

(in general, each t_i may be a vector), a set of dependent variables

$$y = \{y_i, i = 1 \text{ to } m\},$$

and a set of parameters to be found

$$p = \{p_j, j = 1 \text{ to } n\},$$

we must match the vector y as closely as possible by a computed vector (the model)

$$f = \{f(t_i, p), i = 1 \text{ to } m\}$$

by adjusting the elements of p. Our task is to minimize

$$S(p) = \|r\|, \tag{1.1}$$

where $r = y - f$, and $\|\cdot\|$ is a k-norm $1 \leqslant k \leqslant \infty$ to be specified in context. The k-norm is defined as

$$\|r\| = \left(\sum_{i=1}^{m} |r_i|^k\right)^{1/k}, \qquad 1 \leqslant k < \infty,$$

$$= \max_i |r_i|, \qquad k = \infty.$$

(Weights may be introduced by substituting $w_i y_i$ for y_i and $w_i f_i$ for f_i.)

The choice of k is not entirely arbitrary. For data y whose errors are normally distributed, $k = 2$ is a good choice because of the vast literature on the resulting distributions of the parameters p (e.g. [23, Chapters 2, 5]). If the data y are plagued with outliers, $k = 1$ is a good choice because in L_1, outliers tend to be completely

Received March 23, 1978; revised August 7, 1978 and April 16, 1979.
AMS (MOS) subject classifications (1970). Primary 41A45, 41A50.

0025-5718/80/0000-0061/$04.25

ignored [3]. Finally, if the errors in y are negligible (e.g., if the residuals r are expected to be much larger than the errors in y), then $k = \infty$ will produce the smallest maximum deviation $|r_i|$.

A model f is called linear if at each point, f_i can be written

$$f_i = a_{i1}p_1 + a_{i2}p_2 + \cdots + a_{in}p_n, \tag{1.2}$$

or considering all the a_{ij} as a matrix A,

$$f = Ap. \tag{1.3}$$

The a's may be any functions of t, e.g., $a_{i1} = \sin(t_i)$, but they may not contain p. In other words, A must remain constant when p varies. If the p's appear in exponents or in denominators or in any manner not expressible by (1.2), the model f is called nonlinear.

Linear models are important here because for the linear model in three norms, L_1, L_2, and L_∞, there are effective methods for minimizing S either globally or with linear constraints (e.g. [3]–[10], [24]–[26]). Levenberg-like methods, among others, regard the a_{ij}'s as fixed in each iteration, and use repeated applications of a linear solver to generate increments to p [2], [14], [15], [17], [18], [21], [24], [25].

In order to describe the Levenberg-Marquardt method, we introduce the following notation:

A = the Jacobian with elements

$$a_{ij} = \partial f_i/\partial p_j \quad (= w_i \partial f_i/\partial p_j \text{ in the weighted case}).$$

B = diagonal matrix of k-norms of the columns of A. If any b_{jj} is zero, it should be reset to 1.

x = increments to p, hopefully to improve S.

$R = \|Bx\|$, a measure of step size.

$T = \|r - Ax\|$, a good approximation to $S(p + x)$ when f is nearly linear in the region of interest, e.g. for small enough R.

α = a scalar, $0 \leqslant \alpha \leqslant 1$.

$U = \|\{\alpha T, (1 - \alpha)R\}\|$ the function to be minimized over x for any choice of α. ($\{\alpha T, (1 - \alpha)R\}$ is a vector with two elements.)

The method is as follows, where '$\leftarrow$' denotes assignment of right-side value to left-side variable:

1. Set initial p and tolerance on S.
2. (An iteration begins.)
   $S_{\text{old}} \leftarrow S(p)$; $S_{\text{goal}} \leftarrow S_{\text{old}}$ (1 − tolerance);
   Compute A, B, and r at current p.
3. (Sampling procedure. See formulas (2.3), (2.4), (2.6) for details.)
   For several α, $0 < \alpha \leqslant 1$:
   find any finite solution of $\min_x U(x)$;
   compute $S(p + x)$.
4. If $S(p + x) \geqslant S_{\text{old}}$ for all x's in step 3, stop.
   If $S(p + x) \geqslant S_{\text{goal}}$ for all x's in step 3, then for that x of minimum S,
   $p \leftarrow p + x$ and stop.

If $S(p + x) < S_{goal}$ for some x in step 3, then choose that x of greatest R that satisfies $S(p + x) < S_{goal}$, $p \leftarrow p + x$ and go to step 2.

Note in step 3 that any general method for minimizing T will also minimize U, since both problems are linear, but U involves solving $m + n$ equations:

$$\begin{bmatrix} \alpha A \\ (1 - \alpha)B \end{bmatrix} x \approx \begin{bmatrix} \alpha r \\ 0 \end{bmatrix}. \tag{1.4}$$

Some pleasant properties of this method in L_2 are discussed in [13], [14], [16], [17], [23]. In L_1 and L_∞, Osborne and Watson [21] show, essentially, that if $R > 0$ can be made arbitrarily small by adjusting α, then the method is convergent. They remark that the L_1 method just described does not meet such a requirement. As a result, Anderson and Osborne [2] choose to generate x by:

$$\min_x \max (\alpha\|T\|, (1 - \alpha)\|R\|),$$

a mixed-norm method.

We will show that, with slight alteration the pure L_1 algorithm can be made convergent. This is useful in three ways:

1. A linear L_1 solver can be used without alteration, making one code serve both linear and nonlinear problems.

2. Linear solvers for L_1 are constantly being improved, while the mixed-norm problem receives relatively little attention. As better methods develop, one need only replace the old linear solver with the new one for improved performance.

3. Linear solvers are available with contrasting merits. Some users will want speed [4], [16]; others prefer stability [9]; still others are concerned with constraints on p [6], [7], and so on. Such variety is not available in mixed norm.

Finally, we will show that some other difficulties in both L_1 and L_∞ are resolved by careful choice of α.

II. Special Problems in L_1 and L_∞. In this section, several statements are made, without proof, as to the nature of a particular norm or justification of the search procedure. More formal treatment of these is available in a supplementary paper [26].

In the smooth norms, $1 < k < \infty$, $x(\alpha)$ is unique and continuous (with the exception of $\alpha = 1$ in the rank deficient case), while $T(\alpha)$ and $R(\alpha)$ are unique, continuous and strictly monotonic. These properties are partially or totally missing in L_1 and L_∞ problems, resulting in the following difficulties:

1. Linear programming, a possibly expensive procedure, is required to minimize U. Consequently, we have chosen heuristics, described in this section, to keep the number of such computations small. We have chosen the most efficient available linear methods, those of Barrodale et al. [4], [5]. Also, work by McCormick and Sposito [16] indicates that a good starting estimate reduces the linear computations substantially. Since our nonlinear methods involve exploring several nearby $p + x$ values, we were pleased to learn that our first guess, namely p, was very good on the average, yielding x with much less computation than one would expect from a random problem of the same size. See the section on results.

2. x may be nonunique at special values or entire intervals of α. For this, we rely on the linear solver to choose one solution. The linear solvers given in [4], [5], [6], [9], [10] have this capability.

3. In L_1, $x(\alpha)$ is discontinuous in α (e.g. see Figure 1). This property interferes with our control over the search of x, and may prevent convergence. The best $S(p + x)$ may be in a region skipped over by the locus $x(\alpha)$. We propose a 'continuous' version of the L_1 method using linear interpolation between the discrete points generated by $x(\alpha)$.

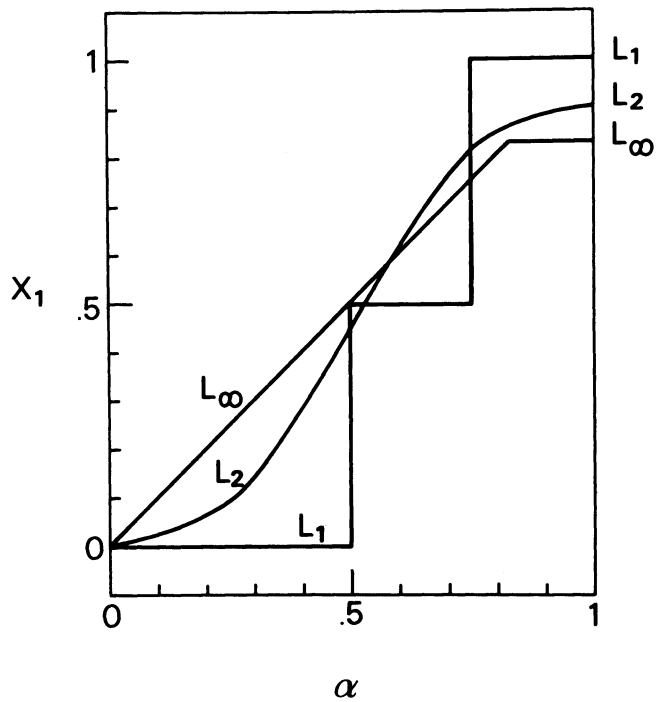

FIGURE 1

x as a function of α in three norms. The problem data is

$$y_1 = .5; \quad y_2 = 1; \quad w_1 = 1; \quad w_2 = 2;$$

$$f_1 = f_2 = p_1; \quad \text{initial } p_1 = 0.$$

The function minimized over x for each α is then

$$L_1\colon \ U(x) = \alpha|.5 - x_1| + 2\alpha|1 - x_1| + 3(1 - \alpha)|x_1|$$

$$L_2\colon \ U(x) = \alpha^2(.5 - x_1)^2 + 4\alpha^2(1 - x_1)^2 + 5(1 - \alpha)^2x_1^2$$

$$L_\infty\colon \ U(x) = \max(\alpha|.5 - x_1|, 2\alpha|1 - x_1|, 2(1 - \alpha)|x_1|).$$

4. There are level intervals, i.e. intervals of α in which x or its associated R and T remain constant. If one has generated an x in such an interval, one certainly wants to avoid the expense of generating that same x again.

The following remarks on L_1 follow from the linear programming nature of solving linear L_1 problems. The L_1 norm is unique in that there is a solution $x(\alpha)$ (that minimizes U and) that satisfies a subset of the $m + n$ equations in (1.4) exactly. For $\alpha < 1$, there is an x that satisfies at least n equations exactly, and these are a subset of rank n. If there is more than one x for a given α, then there is more than one full rank subset; and the set of possible solutions consists of all x, which minimize U and satisfy some full rank subset plus all convex combinations of such x. If we can control which x is chosen, we will have at our disposal a continuous sequence of line segments or possibly a band of x's in $0 \leqslant \alpha \leqslant 1$ ranging from $x = 0$ all the way to those that minimize T. But linear solvers based on linear programming are primed to select only those x uniquely determined by the full rank subsets, of which there can be only a finite number. So $T(\alpha)$ and $R(\alpha)$ instead of being continuous and strictly monotonic, become sums of step functions with coincident discontinuities at a finite number of critical α values.

Let $\bar{\alpha}_j$ be the jth critical value. Let $\bar{T}_j$ and $\bar{R}_j$ be the values of T and R in $\bar{\alpha}_j < \alpha < \bar{\alpha}_{j+1}$. At a critical value $\bar{\alpha}_j$, $U(\bar{\alpha}_j)$ is of course unique, being the minimum over x, but T and R can range from $\bar{T}_j \leqslant T \leqslant \bar{T}_{j-1}$, and $\bar{R}_{j-1} \leqslant R \leqslant \bar{R}_j$. As remarked above, if two distinct x's minimize U, then any convex combination of them will do likewise. So we see that linear interpolation between x's generated by the linear solver is not merely an expedient for controlling the step size R. It is a natural completion of the Levenberg method in L_1, creating a continuous locus $x(\alpha)$ all of which minimize U for some α, and permitting convergence as described in [21].

In L_1, then, we are left with three very practical questions. First, how do we avoid repeating samples, i.e. choosing two α's in the same level interval? Second, how do we know when to stop choosing α, and start interpolating? And finally, how do we stop the search?

At any critical value $\bar{\alpha}_j$, the following relation holds:

$$U(\alpha_j) = \bar{\alpha}_j \bar{T}_{j-1} + (1 - \bar{\alpha}_j)\bar{R}_{j-1} = \bar{\alpha}_j \bar{T}_j + (1 - \bar{\alpha}_j)\bar{R}_j$$

or

$$\bar{\alpha}_j = (\bar{R}_j - \bar{R}_{j-1})/(\bar{R}_j - \bar{R}_{j-1} + \bar{T}_{j-1} - \bar{T}_j). \tag{2.1}$$

Suppose we are given R and T from two distinct (not necessarily adjacent) level intervals u and v, $u + 1 \leqslant v$. Then in the formula

$$\alpha_{\text{new}} = (\bar{R}_v - \bar{R}_u)/(\bar{R}_v - \bar{R}_u + \bar{T}_u - \bar{T}_v) \tag{2.2}$$

the numerator is the sum of the numerators of (2.1), $u + 1 \leqslant j \leqslant v$, and similarly for denominators.

Remembering

$$\frac{a}{b} < \frac{c}{d} \quad \text{implies} \quad \frac{a}{b} < \frac{a + c}{b + d} < \frac{c}{d} \quad \text{for } a, b, c, d > 0,$$

it is easily shown that for nonadjacent intervals $u + 1 < v$, $\bar{\alpha}_{u+1} < \alpha_{\text{new}} < \bar{\alpha}_v$ holds. I.e., α_{new} does not lie in either of the original level intervals. The only exception is

when $u + 1 = v$. In that case a repeated R may be generated, but this is easily detectable, and one may then switch to linear interpolation between the neighboring x's. There will be at most one repeated x per iteration, and rarely that.

Let $x^{[new]}$ be the newly-generated x, with an associated retinue of values α_{new}, R_{new}, T_{new}, and S_{new}. Let $x^{[in]}$ and $x^{[out]}$ have similar retinues.

The sampling procedure for x is:

$$\text{initially: } \alpha_{in} \leftarrow 0;\ \alpha_{out} \leftarrow 1;$$

(2.3) $$\alpha \text{ search: } \alpha_{new} \leftarrow (R_{out} - R_{in})/(R_{out} - R_{in} + T_{in} - T_{out}).$$

If $R_{new} = R_{in}$ or $R_{new} = R_{out}$, switch to:

(2.4) $$x \text{ search: } x^{[new]} \leftarrow (1 - \beta)x^{[in]} + \beta x^{[out]},$$

where $\beta < 1$ is a refinement constant.

($\beta = .25$ in the authors' programs.)

replacement: If $T_{new} < S_{goal}$ and $S_{new} \geqslant S_{goal}$,

then $x^{[out]} \leftarrow x^{[new]}$; else $x^{[in]} \leftarrow x^{[new]}$,

where $x^{[-]} \leftarrow x^{[new]}$ denotes replacement of the entire retinue.

The rules for stopping the search are:

1. Go to the next iteration if
   (a) $S(\alpha = 1) < S_{goal}$, or
   (b) $S_{in} < S_{goal}$ and $R_{in} \geqslant \beta R_{out}$, or
   (c) $S(p + x) < .75S_{goal}$ for any x.
2. Stop (signal convergence) if

$$S_{in} \geqslant S_{goal}, \quad T_{in} \geqslant S_{goal}, \quad \text{and} \quad R_{in} \geqslant \beta R_{out}.$$

Formulas (2.3) and (2.4) are analogous to interval halving. In effect, a root (not the smallest) of $S(R) = S_{goal}$ is being refined to a crude tolerance, with the final provision that $S < S_{goal}$ must hold to insure convergence. The object of this search is to choose x with rather large R for faster convergence as noted in [1], [15].

In L_∞, there is only one level interval, and it can be easily located. From the formula

$$U = \max(\alpha T, (1 - \alpha)R)$$

we see that $(1 - \alpha)R$ will have no effect on the result until it becomes as large as αT.

Let

$$T^* = \text{the global minimum of } T,$$

$$R^* = \text{minimum } R(x) \text{ for which } T(x) = T^*, \text{ and}$$

$$\alpha^* = R^*/(R^* + T^*), \text{ i.e. where } \alpha^* T^* = (1 - \alpha^*)R^*.$$

In the interval $0 \leqslant \alpha \leqslant \alpha^*$, $T(\alpha)$ and $R(\alpha)$ can be shown to be strictly monotonic, with all the properties necessary for convergence. For some formal results, see [2], [26]. But repeated iterates can occur in the interval $\alpha^* \leqslant \alpha \leqslant 1$. To locate α^* in a

problem with $x(\alpha = 1)$ unique, one simply minimizes $U(\alpha = 1)$ over x, with results $T(x) = T^*$ and $R(x) = R^*$. If $x(\alpha = 1)$ is nonunique, $R(x) > R^*$ will cause α^* to be overestimated, but if the next α is chosen in $\alpha^* \leqslant \alpha < 1$, then $T(\alpha) = T^*$ must hold. Using this last equality and the relations

$$(1 - \alpha_i)R(\alpha_i) \leqslant \alpha_i T^*,$$

we can show that the sequence of α: $\alpha^* < \alpha_i < \alpha_{i-1} < \cdots < \alpha_1 = 1$, where

$$\alpha_{i+1} = \beta R(\alpha_i)/(R(\alpha_i) + T^*) \tag{2.5}$$

produces a sequence $R(\alpha_{i+1}) < \beta R(\alpha_i)$, so that x is guaranteed never to repeat. The L_∞ sampling procedure is similar to that of L_1, but the α search becomes

$$\begin{aligned} &\alpha_{\text{new}} \leftarrow (1 - \beta)\alpha_{\text{in}} + \beta\alpha_{\text{out}}. \\ &\text{If } T_{\text{new}} = T(\alpha = 1) \text{ then replace } \alpha_{\text{new}}: \\ &\quad \alpha_{\text{new}} \leftarrow R_{\text{new}}/(R_{\text{new}} + T_{\text{new}}) \\ &\quad \text{without recomputing } x^{[\text{new}]}, \text{ etc.} \end{aligned} \tag{2.6}$$

Since there are no discontinuities in $x(\alpha)$ for L_∞, there is no need for the line search in (2.4).

III. Linear Constraints. Sometimes, it is desirable to constrain parameter values to certain regions. The most commonly used constraints are linear, i.e., of the form

$$\sum_{j=1}^{n} c_{ij} p_j \tilde{r}_i d_i, \qquad i = 1 \text{ to } L, \tag{3.1}$$

where the relation $\tilde{r}_i$ may be $\leqslant$, $=$, or $\geqslant$, and the c's and d's are constants. A parameter vector that satisfies all the constraints is called feasible.

In order to impose linear constraints on p, the major change in the method is at step 3, which should now read:

$$\text{minimize } U \text{ over } x \text{ subject to } \sum_{j=1}^{n} c_{ij} x_j \tilde{r}_i d_i - \sum_{j=1}^{n} c_{ij} p_j, \qquad i = 1 \text{ to } L. \tag{3.2}$$

To use the method, one must now invoke a procedure that solves the constrained linear problem. Methods for such minimization are available in L_1 and L_2 [6], [7], [18].

One further change is advisable. Initial p should be feasible in order to insure the convergence property that R can be made arbitrarily small. If a user provides infeasible initial p, it can be corrected by minimizing $\|x\|$ subject to the constraints. This is done by setting $A = I_n$, $r = 0$, $\alpha = 1$, and executing (3.2). Thereafter, the algorithm will never generate infeasible $p + x$.

IV. Results and Comparison. We refer to the methods described above as Levenberg-Marquardt (LM). In the L_1, L_2, and L_∞ norms, there is another well-known method called damped Gauss-Newton (DGN). Both methods start by finding an x, call it $\hat{x}$, that minimizes T. Thereafter, if $\hat{x} \neq 0$ and $S(p + \hat{x}) \geqslant S_{\text{goal}}$, DGN searches on γ, $0 < \gamma < 1$, for the relation $S(p + \gamma\hat{x}) < S_{\text{goal}}$. The theory behind this search is explained in the papers of Osborne and Watson [19], [20] where they

prove that if $T(\hat{x}) < S_{old}$, then $\hat{x}$ is a downhill direction in S. In Lemma 3.4 [20] and Lemma 2.2 [19] they show that given smooth enough f, $S(p + \gamma\hat{x}) < S_{old}$ must hold for some small γ.

Osborne and Watson require that γ be searched for a minimum of S. In a later paper, Anderson and Osborne [1] require that γ be chosen large while still satisfying an improvement criterion. We have found that the latter strategy works best, so we used it in the test cases. As illustrated in Table 4, γ was assigned nonpositive powers of 10 in the search, as in [1, Table 3].

Let $\tilde{x}$ be some x produced by the LM method. An important result of minimizing U in any k-norm is that $T(\tilde{x})$ is minimized for a given step size $R(\tilde{x})$ [26]. Therefore, for equal step sizes $R(\tilde{x}) = R(\gamma\hat{x})$, and for sufficiently smooth f, the relation $S(p + \tilde{x}) \leqslant S(p + \gamma\hat{x})$ holds in a right neighborhood of $R = 0$. Marquardt [15] observed that in L_2, the directions taken by the two search paths near $R = 0$ are often quite different. We have observed the same effect in L_1 and L_∞. In fact, there are some cases in which DGN direction $\hat{x}$ is so far from the gradient, that $S(p + \gamma\hat{x}) > S_{goal}$ holds in the entire interval $0 \leqslant \gamma \leqslant 1$, whereas substantial improvement is seen in $S(p + \tilde{x})$ for small R. This contrast in behavior of the two methods is especially common when the Jacobian is ill-conditioned, as illustrated by the following studies.

For each of three functional forms, a solution p^* and a singular point $p^{(s)}$ (at which the Jacobian becomes rank deficient) were chosen, and the initial estimates $p^{(0)}$ were generated by

$$p^{(0)} = (1 - \rho)p^{(s)} + \rho p^*, \qquad 0 < \rho < 1.$$

The object was to measure the amount of work required to find p^* as a function of ρ. The entire study involved every combination of three functional forms of f, two norms, two methods, and eleven values of ρ, 132 problems in all. As a compromise between completeness and brevity, we will describe all the test problems, but we will give only representative results, other results being very similar.

In all cases, the values of ρ were

$$(.7, .5, .3, .2, .15, .1, .07, .05, .03, .02, .01).$$

The values of t (the independent variable) were

$$t_i = (i - 1)/48, \qquad i = 1 \text{ to } 49.$$

The values of y (the dependent variable) were

$$L_1\colon y_i = f(t_i; p^*) + e_i$$

$$\text{where } e = (0, -.1, .1, 0, -.1, \ldots, .1, 0)^T.$$

$$L_\infty\colon y_i = f(t_i; p^*) + .01 \cos((n + 2)\pi t_i).$$

The second term of y_i was designed to avoid the exact fit case (which is rare) without changing the solution p^*.

The three test functions with associated solutions and singular points are:

1. Sums of exponentials:
   model: $f(t_i; p) = p_1\exp(-p_2 t_i) + p_3\exp(-p_4 t_i)$
   solution: $p^* = (1, 3, 1, 1)^T$
   sing. pt.: $p^{(s)} = (1, 2, 1, 2)^T$

For models 2 and 3:

   let $z_1(t_i) = (t_i - p_2)/p_3$, $z_2(t_i) = (t_i - p_5)/p_6$.

2. Sums of Gaussians:
   model: $f(t_i; p) = p_1\exp(-z_1^2(t_i)) + p_4\exp(-z_2^2(t_i))$
   solution: $p^* = (1, .4, .4, 1, .7, .2)^T$
   sing. pt.: $p^{(s)} = (1, .55, .3, 1, .55, .3)^T$
3. Sums of derivatives of Lorentzians:
   model: $f(t_i; p) = p_1 z_1(t_i)/(1 + z_1^2(t_i))^2 + p_4 z_2(t_i)/(1 + z_2^2(t_i))^2$
   solution: $p^* = (1, .4, .4, 1, .7, .2)^T$
   sing. pt.: $p^{(s)} = (1, .55, .3, 1, .55, .3)^T$
   note: although p^* and $p^{(s)}$ are the same in models 2 and 3, the functions f are quite different, as are the intermediate iterates in the test problems.

TABLE 1

*Amount of work done in the three major time-consuming phases of solution for the exponential model in the L_1 norm. * indicates failure to find any minimum of S.*

ρ	LM			DGN		
	ITS	FS	LPITS	ITS	FS	LPITS
.7	5	6	37	5	6	37
.5	6	10	66	7	9	53
.3	6	11	72	13	26	96
.2	6	12	75	23	65	175
.15	6	14	96	36	103	286
.1	7	15	104	>30		
.07	7	15	103	>30		
.05	7	15	106	8*		
.03	6	16	115	1*		
.02	6	16	114	1*		
.01	6	16	118	1*		

All arithmetic was done in double precision (≈15 decimal digits) on an IBM 370/168 computer. In all cases, the runs were terminated when S failed to achieve relative improvement of .0001 in an iteration, i.e. $S \geqslant .9999 S_{\text{old}}$. Three indicators of

efficiency were recorded:

1. Number of iterations, denoted ITS in Table 1. These are significant because partial derivatives are computed at the start of each iteration. These are sometimes very expensive to produce.

2. Number of functions sampled (exclusive of partial derivative calculations), i.e. number of parameter vectors tried, denoted FS in Table 1.

TABLE 2

*Amount of work done in the three major time-consuming areas of solution for the Lorentzian model in the L_∞ norm. * indicates failure to find any minimum of S.*

ρ	LM			DGN		
	ITS	FS	LPITS	ITS	FS	LPITS
.7	7	8	98	7	8	98
.5	8	15	195	10	14	141
.3	10	19	246	— local min. —		
.2	10	19	235	33	93	512
.15	8	16	213	>30		
.1	10	19	234	>30		
.07	10	20	255	>30		
.05	10	20	267	12*		
.03	10	19	247	1*		
.02	10	19	247	1*		
.01	10	19	244	2*		

TABLE 3

Selected iterates for the exponential model in L_1 using the LM method. $\rho = 0.2$. Note immediate departure from the singular subspace $p_2 = p_4$. $\alpha = 1$ corresponds to the undamped Gauss-Newton method.

IT #	p_1	p_2	p_3	p_4	α	S
0	1.00	2.20	1.00	1.80	—	4.98
1	0.97	2.20	1.00	1.26	0.95	3.56
2	0.97	2.85	1.00	0.95	0.97	3.55
3	0.99	3.02	1.01	1.00	1.00	3.21
4	1.00	3.00	1.00	1.00	1.00	3.20
6	1.00	3.00	1.00	1.00	1.00	3.20

3. Number of LP iterations, as described in McCormick and Sposito [16], denoted LPITS in Table 1. This is a sum of iterations from all LP solutions invoked in solving the nonlinear problem, and constitutes a reasonable measure of work in generating trial parameters.

These three activities are the major time consumers. Any measure of total work for varying $p^{(0)}$ (with a given f, computer system, compiler, and implementation) would be very close to a weighted combination of the above three measures. A "typical" measure of work might be

$$\text{work} = (1.5n)(\text{ITS}) + (\text{FS}) + (1/6)(\text{LPITS}),$$

but the particular weights chosen are unimportant for this discussion.

TABLE 4

Selected iterates for the exponential model in L_1 using the DGN method. $\rho = 0.2$. Note the tendency to linger near the singular subspace $p_2 = p_4$ in the early iterations. $\gamma = 1$ corresponds to the undamped Gauss-Newton method.

IT #	p_1	p_2	p_3	p_4	γ	S
0	1.00	2.20	1.00	1.80	—	4.98
1	1.03	2.20	0.97	1.79	.001	4.97
9	1.18	2.18	0.82	1.74	.001	4.95
10	1.19	2.18	0.81	1.73	.001	4.95
11	1.31	2.18	0.69	1.68	.01	4.92
12	1.36	2.18	0.64	1.63	.01	4.92
13	1.38	2.19	0.62	1.60	.01	4.90
14	1.47	2.27	0.53	1.32	.1	4.61
17	1.21	2.51	0.79	1.22	.1	4.07
18	1.17	2.57	0.82	1.19	.1	3.96
19	0.90	3.05	1.10	0.96	1.0	3.82
20	1.00	2.99	1.00	1.00	1.0	3.22
23	1.00	3.00	1.00	1.00	1.0	3.20

Typical results are shown in Tables 1 and 2. In all cases, (qualitatively) as $p^{(0)}$ was moved from p^* to $p^{(s)}$, the amount of work increased linearly for LM but hyperbolically for DGN. If, after 30 iterations, there was any evidence of progress toward the desired solution, the problem was run to completion. Otherwise, the ITS entry reads > 30 with no other statistics. The ITS entry "N^*", where N is an integer, indicates spurious convergence to a nonminimum. In every such case, LM could start from the final GN parameters and find substantial improvement in S, often reaching the desired solution.

Any method will fail on occasion, but DGN does so in problems where substantial local improvement in S is easily found by, say, steepest descent. The problem seems to be DGN's affinity for singular subspaces (e.g. in model 1 when $p_2 = p_4$ and in models 2 and 3 when $p_2 = p_5$ and $p_3 = p_6$). Given a first guess or an intermediate iterate near such a subspace, DGN tends to stay there longer then we would prefer, perhaps to the point of stopping before a minimum is reached. Tables 3 and 4 list intermediate results for one problem using both DGN and LM. This problem illustrates, in mild form, the comparative attraction that singular subspaces hold for DGN.

V. Summary. The four apparent liabilities of LM methods, summarized at the beginning of Section II, can be overcome to produce effective methods in the L_1 and L_∞ norms. The discontinuities of $x(\alpha)$ in L_1 are eliminated by interpolation in formula (2.4). Repeated iterates are avoided by formulas (2.3) in L_1 and (2.6) in L_∞. Efficiency seems competitive with DGN methods in cases where both work well, but LM appears more reliable in the test problems. In addition, Anderson and Osborne have noted superior behavior of their own LM methods when the multiplier condition, defined in [1], [2], fails to hold. Watson [27] shows that one may solve continuous L_∞ problems (where y and f are used for all t in an interval) by solving discrete L_∞ problems with suitably chosen abscissas. He makes use of both DGM and LM, favoring some LM features when the Jacobian becomes singular at the solution. It appears, then, that despite the apparent expense of generating x in LM methods (one LP problem for each x in LM, but only one LP problem for each iteration in DGN), robustness and convergence in fewer iterations makes LM the more attractive of the two approaches. It may be possible to merge the two in some way to get the best of both worlds.

Laboratory of Applied Studies
Division of Computer Research and Technology
National Institutes of Health
Bethesda, Maryland 20205

Department of Energy
Energy Information Administration
Division of Coal and Electric Power Statistics
1000 Independence Avenue, S.W.
Washington, D. C. 20585

1. D. H. ANDERSON & M. R. OSBORNE, "Discrete, nonlinear approximation problems in polyhedral norms," *Numer. Math.*, v. 28, 1977, pp. 143–156.

2. D. H. ANDERSON & M. R. OSBORNE, "Discrete, nonlinear approximation problems in polyhedral norms–a Levenberg-like algorithm," *Numer. Math.*, v. 28, 1977, pp. 157–170.

3. I. BARRODALE & F. D. K. ROBERTS, "An improved algorithm for L_1 linear approximation," *SIAM J. Numer. Anal.*, v. 10, 1973, pp. 839–848.

4. I. BARRODALE & F. D. K. ROBERTS, "Solution of an over-determined system of equations in the L_1 norm," *Comm. ACM*, v. 17, 1974, pp. 319–320.

5. I. BARRODALE & C. PHILLIPS, "Algorithm 495: Solution of an over-determined system of linear equations in the Chebyshev norm," *ACM Trans. Math. Software*, v. 1, 1975, pp. 264–270.

6. I. BARRODALE & F. D. K. ROBERTS, *An Efficient Algorithm for Discrete L_1 Linear Approximation with Linear Constraints*, Tech. Dept. DM-103-IR, Dept. of Math., Univ. of Victoria Victoria, B. C., Canada, July 1977.

7. I. BARRODALE & F. D. K. ROBERTS, *Solution of the Constrained L_1 Linear Approximation Problem*, Tech. Dept. DM-104-IR, Dept. of Math., Univ. of Victoria, Victoria, B. C., Canada, July 1977.

8. R. H. BARTELS & G. H. GOLUB, "Algorithm 328," *Comm. ACM*, v. 11, 1968, pp. 428–430.

9. R. H. BARTELS, A. R. CONN & J. W. SINCLAIR, "Minimization techniques for piecewise differentiable functions: the L_1 solution to an overdetermined linear system," *SIAM J. Numer. Anal.*, v. 15, 1978, pp. 224–241.

10. R. H. BARTELS, A. R. CONN & C. CHARALAMBOUS, "On Cline's direct method for solving overdetermined linear systems in the L_∞ sense," *SIAM J. Numer. Anal.*, v. 15, 1978, pp. 255–270.

11. E. W. CHENEY, *Introduction to Approximation Theory*, McGraw-Hill, New York, 1966.

12. S. I. GASS, *Linear Programming*, McGraw-Hill, New York, 1969.

13. P. LaFATA & J. B. ROSEN, "An interactive display for approximation by linear programming," *Comm. ACM*, v. 13, 1970, pp. 651–659.

14. K. LEVENBERG, "A method for the solution of certain non-linear problems in least squares," *Quart. Appl. Math.*, v. 2, 1944, pp. 164–168.

15. D. W. MARQUARDT, "An algorithm for least-squares determination of nonlinear parameters," *SIAM J. Appl. Math.*, v. 11, 1963, pp. 431–441.

16. G. F. McCORMICK & V. A. SPOSITO, "Using the L_2-estimator in L_1-estimation," *SIAM J. Numer. Anal.*, v. 13, 1976, pp. 337–343.

17. D. A. MEETER, "On a theorem used in non-linear least squares," *SIAM J. Appl. Math.*, v. 14, 1966, pp. 1176–1179.

18. D. D. MORRISON, *Methods for Non-Linear Least Squares Problems and Convergence Proofs. Tracking Programs and Orbit Determination*, Proc. Jet Propulsion Lab. Seminar, 1970, pp. 1–9.

19. M. R. OSBORNE & G. A. WATSON, "An algorithm for minimax approximation in the nonlinear case," *Comput. J.*, v. 12, 1969, pp. 63–68.

20. M. R. OSBORNE & G. A. WATSON, "On an algorithm for discrete nonlinear L_1 approximation," *Comput. J.*, v. 14, 1971, pp. 184–188.

21. M. R. OSBORNE & G. A. WATSON, *Nonlinear Approximation Problems in Vector Norms* (G. A. Watson, Ed.), Lecture Notes in Math., Vol. 630, Springer-Verlag, Berlin, 1978.

22. J. R. RICE, *The Approximation of Functions*, Addison-Wesley, Reading, Mass., 1964.

23. S. R. SEARLE, *Linear Models*, Wiley, New York, 1971.

24. R. I. SHRAGER, "Nonlinear regression with linear constraints: an extension of the magnified diagonal method," *J. AMC*, v. 17, 1970, pp. 446–452.

25. R. I. SHRAGER, "Quadratic programming for nonlinear regression," *Comm. ACM*, v. 15, 1972, pp. 41–45.

26. R. I. SHRAGER & E. HILL, "Some properties of the Levenberg method in the L_1 and L_∞ norms," 1979. Available from the authors.

27. G. A. WATSON, "A method for calculating best non-linear Chebyshev approximations," *J. Inst. Math. Appl.*, v. 18, 1976, pp. 351–360.