A Globally Convergent Method for L_P Problems

Yuying Li*

TR 91-1212
June 1991

Department of Computer Science
Cornell University
Ithaca, NY 14853-7501

*Computer Science Department, Cornell University, Ithaca, New York, 14853.
Research partially supported by the U.S. Army Research Office through Mathematical Sciences Institute, Cornell University and by the Computational Mathematics Program of the National Science Foundation under grant DMS-8706133.

A GLOBALLY CONVERGENT METHOD FOR L_P PROBLEMS*

YUYING LI†

Abstract. The l_p norm discrete estimation problem $\min_{x \in \mathbb{R}^n} \|b - A^T x\|_p$ is troublesome when p is close to unity because the objective function approaches a discontinuous form. In this paper, we present an efficient approach for solving l_p norm problems for all $1 \leq p < 2$. When $p = 1$, it is essentially the method presented in [4], which is a globally and quadratically convergent algorithm under some nondegeneracy assumptions. The existing iteratively reweighted least squares method (IRLS) can be obtained from the new approach by updating some "dual multipliers" in a special fashion. The new method is globally convergent. It is superlinearly convergent when there is no zero residual at the solution. The main computational cost of our new method is the same as the IRLS method: a reweighted least squares solve. Numerical experiments indicate this method is significantly faster than popular iteratively reweighted least squares methods when p is close or equal to one.

Key Words. discrete estimation, data analysis, IRLS method, linear programming, interior-point algorithm, simplex method, Newton method

Subject Classification. AMS/MOS: 65H10, 65K05, 65K10.

* Written June 1991.

† Computer Science Department, Cornell University, Ithaca, New York, 14853. Research partially supported by the U.S. Army Research Office through Mathematical Sciences Institute, Cornell University and by the Computational Mathematics Program of the National Science Foundation under grant DMS-8706133.

1. Introduction. In discrete estimation and data analysis, it is often appropriate to solve the following problem:

$$\min_{x\in\Re^n} \psi(x), \qquad \text{where} \qquad \psi(x) = \sum_{i=1}^{m} |a_i^T x - b_i|^p, \tag{1.1}$$

$A = [a_1, \cdots, a_m] \in \Re^{n\times m}$, $b \in \Re^m$ and $m > n$. In this paper, we focus on the case when $1 \le p < 2$. We assume that A is of rank n and there does not exist any x such that $A^T x - b = 0$.

The most often used measures are 2–norm, 1–norm and ∞–norm. The l_∞ solution offers a worst case guarantee. The 2–norm solution is popular because of its special relationship with the normal distributions. The increasingly important l_1 solution is useful since it is insensitive to a small number of large residuals (*resistant*). Thus, one can imagine situations when minimizing the l_p norm, where $1 < p < 2$, is appropriate ([12],[13]). Moreover, the problem is theoretically interesting since it ranges from a piecewise differentiable minimization problem (equivalent to a constrained minimization) when $p = 1$, through a once-differentiable minimization problem (but not twice differentiable) when $1 < p < 2$, to a twice-differentiable problem when $p = 2$. Clearly, it would be useful, though challenging, to develop a method which works well in all the cases.

The 2–norm problem is easy to solve: a simple least squares problem. The l_1 and l_∞ problems are much more complicated and can be treated as linear programming problems and thus solved by special linear programming methods which usually take advantage of their special structures (e.g, [1], [2]). The objective function $\psi(x)$ becomes piecewise linear when $p = 1$ or ∞.

Until recently, the usual methods for l_1 and l_∞ have been *finite* simplex-type algorithm (e.g, [1], [2]). These methods move along negative projected gradients and follow nondifferentiable hyperplanes whenever possible. In contrast, we have recently developed *iterative* methods for l_1 and l_∞ minimization ([4] and [5]). These algorithms deal with the nondifferentiable hyperplanes by strategically avoiding landing on them exactly and being able to cross when necessary. They are computationally efficient. Under suitable nondegeneracy assumption, both algorithms are proven to be globally convergent with a quadratic convergence rate.

For $1 < p < 2$, the most popular method for solving (1.1) is the iteratively reweighted least squares method (e.g., [10], [12]). This method essentially takes a fixed stepsize along the Newton direction defined by equations $\psi(x) = 0$. The method has been proven to be globally linearly convergent for $1 < p < 2$ [12]. This method can also be applied to the case $p = 1$ though no global convergence has been proven to our knowledge. With a suitable line search, the algorithm can be accelerated to be quadratically convergent when there is no zero residual at a solution. However, the second order derivative of the objective function does not exist when zero residuals occur and this has been regarded as a main problem of the IRLS approach ([3], [8]).

The purpose of this paper is to further investigate the performance of the IRLS method and provide a new method which works well for $1 \le p < 2$. Our experience indicates (§2) that zero residuals alone do not in general impede the speed of the convergence for the IRLS approach; rather, slow convergence occurs when p is close to unity and thus (1.1) is close to a nondifferentiable problem. This is not entirely surprising because when $p = 1$ problem (1.1) is clearly much more difficult than when $p = 2$. In §3, we present a new algorithm for all $1 \le p < 2$. The new method performs significantly better than the IRLS methods and it is the method of [4] when $p = 1$. In §4, we prove that the new algorithm is globally convergent. Superlinear convergence is achieved when there is no zero residual at the solution. Some numerical experience is presented in §5.

Finally, we introduce a few notations. We always use r to denote the residual vector $r = A^T x - b$. Under this transformation, we have $\phi(r) = \|r\|_p, (= \psi(x))$ and the gradient of $\phi(r)$,

when it exists, is denoted by $g = p|r|^{p-1}\mathrm{sgn}(r)$. In this paper, the multiplication and division between vectors are defined as componentwise operations. The operator $|\cdot|$ around a capital script letter, e.g., $|\mathcal{A}|$, denotes the cardinality of a set. Otherwise it denotes the componentwise absolute values of a number, vector or a matrix. The operator $\max(x, y)$ with two vectors as arguments defines a vector whose components are the maximum of the corresponding argument vectors. The $\max(x)$ of a vector x denotes the maximum component of x. The operator $\mathrm{null}(A)$ denotes the matrix whose columns form a basis for the null space of A, i.e., if $B = \mathrm{null}(A)$, $A^T B = 0$ and $rank(B) = n - rank(A)$. The left arrow $x \leftarrow y$ denotes setting y to x.

2. IRLS Methods. It is well known that the l_p norm is differentiable and strictly convex for $1 < p < \infty$; thus the solution occurs at a point where gradient of $\psi(x)$ with respect to x vanishes, i.e.,

$$\nabla\psi(x) = pAg = pAD_r^{p-2}r = 0, \tag{2.1}$$

where $D_r = \mathrm{diag}(|r|)$.

Let $D = \mathrm{diag}(|r|^{\frac{2-p}{2}})$. The motivation of the IRLS method comes from the fact that (2.1) forms the normal equations for the least squares system

$$D^{-1}A^T x = D^{-1}b,$$

which is a weighted least squares problem.

IRLS Algorithm: Given a starting point x^0
Step 1 Compute $r^k = A^T x^k - b$
Step 2 Define $D^k = \mathrm{diag}\{|r^k|^{\frac{2-p}{2}}\}$, solve x^{k+1} from

$$D^{k^{-1}}A^T x^{k+1} \overset{\text{l.s.}}{=} D^{k^{-1}}b. \tag{2.2}$$

and go to Step 1;

Assume $r_i \neq 0, \forall 1 \leq i \leq m$, a Newton step can be defined for (2.1):

$$d_N = -\frac{1}{p-1}(AD^{-2}A^T)^{-1}AD^{-2}r. \tag{2.3}$$

It is easy to see that a Newton direction is always descent for the objective function $\psi(x)$.

Consider the increment $\Delta x = x^+ - x$ obtained from the IRLS:

$$\begin{aligned} \Delta x &= (AD^{-2}A^T)^{-1}AD^{-2}b - (AD^{-2}A^T)^{-1}AD^{-2}A^T x \\ &= -(AD^{-2}A^T)^{-1}AD^{-2}r \qquad (\ p \geq 1\ ) \\ &= (p-1)d_N \qquad (\ p > 1\ ) \end{aligned}$$

Hence, Δx can be considered as a damped Newton step [13].

In [12], it has been proven that, assuming $1 < p < 2$, the limit point of the sequence generated by the IRLS algorithm is a solution to (1.1) and the convergence is linear with convergence constant $|p-2|$. Wolfe obtained the same local convergence property with a rather involved proof [14].

When $p = 1$, however, there is no global convergence result to our knowledge; however, if global convergence is assumed then the convergence rate will be linear [12]. We claim a slight

modification of the proofs in [4] yields that the IRLS, when $p = 1$, is also globally convergent under some nondegeneracy assumptions.

A line search globalization of the Newton's method can be easily made to speed up the convergence for $1 < p < 2$. In this paper, the following line search procedure is used for both the improvement of the IRLS method and the new algorithm. We refer the modified IRLS algorithm as the IRLSL (IRLS with a line search).

Given any descent direction d, we define a suitable stepsize α by attempting to minimize $\psi(x)$ along the direction d.

The objective function $\psi(x)$ is continuously differentiable. Thus, following Theorem 6.3.2 in [7], given $0 < \beta_f < \beta_g < 1$, there exists $0 < \alpha_l^k < \alpha_u^k$ such that, when $\alpha^k \in [\alpha_l, \alpha_u]$, the following conditions are satisfied at $x^{k+1} = x^k + \alpha^k d_x^k$:

$$\psi(x^{k+1}) \leq \psi(x^k) + \beta_f \alpha^k \nabla\psi(x^k)^T d_x^k \tag{2.4}$$

$$\nabla\psi(x^{k+1})^T d_x^k \geq \beta_g \nabla\psi(x^k)^T d_x^k. \tag{2.5}$$

Unfortunately, the objective function $\psi(x)$ (or $\phi(r)$) is not twice differentiable everywhere. Hence conditions (2.4) and (2.5) do not in general guarantee convergence to the solution. The difficulty is that condition (2.5) may not guarantee large enough steplenghs because of the non-smoothness of the derivatives.

Since the function $\phi(r + \alpha d)$ is a strictly convex function for $\alpha > 0$, there can be only one minimum along any descent d. However, an exact line search is not a practical solution. Instead, we exploit the special structure of the objective function $\phi(r)$ and perform the line search in the following fashion.

Along any direction d the points at which the second order derivatives fail to exist can easily be calculated. We refer to the stepsizes corresponding to such points as breakpoints. The set $\mathcal{J}$ identifies the positive breakpoints:

$$\mathcal{J} = \{\alpha_i : \ \alpha_i = -\frac{r_i}{d_i}, \ \sigma_i d_i < 0\}.$$

The basic idea of our line search procedure is to accept a positive breakpoint if it gives sufficient decrease, i.e.,

$$\phi(r^{k+1}) \leq \psi(r^k) + \beta_f \alpha^k \nabla\phi(r^k)^T d^k. \tag{2.6}$$

We will prove in §4 that positive breakpoints, if taken, are large enough to achieve convergence for the directions defined in our algorithm.

If a positive breakpoint fails to give sufficient decrease of the objective function, we use a quadratic interpolation function which approximates $\phi(r)$ around the current point. Consider the following strictly convex quadratic function $U^k(r)$:

$$U^k(r) = \frac{1}{2}\sum_{i=1}^{m} \frac{|g_i^k|}{|r_i^k|} r_i^2 + \sum_{i=1}^{m} (|r_i^k|^p - \frac{1}{2}\frac{|g_i^k|}{|r_i^k|}|r_i^k|^2), \tag{2.7}$$

where $g = \nabla\phi(r)$. It is easy to verify that

$$U^k(r^k) = \phi(r^k), \quad \nabla U^k(r^k) = \nabla\phi(r^k), \quad \text{and} \quad U^k(r^{k+1}) = \phi(r^{k+1}),$$

where r^{k+1} is defined by IRLS algorithm (cf. (2.2)). In other words, $U^k(r)$ is a quadratic interpolation of $\phi(r)$ with one of the interpolation points chosen in a special way: r^{k+1} is defined by a least squares problem (2.2). This quadratic function has been used to prove that $\{r^k\}$ generated by IRLS algorithm decreases the objective function $\phi(r)$ monotonically ([12], page 252).

In this paper, we use this quadratic interpolation to facilitate the line search in both IRLSL and our new method which will be discussed later. For IRLSL, we calculate the minimizer of $U^k(r)$ along direction d defined by (2.3) and use it to approximate the minimizer of $\phi(r)$ along this direction. The optimal stepsize for $U^k(r)$ along the direction $d^k = A^T d_x^k$ is:

$$\check{\alpha}^k = -\frac{g^{k^T} d^k}{d^{k^T} \mathrm{diag}(p|r^k|^{p-2}) d^k} = p - 1. \tag{2.8}$$

Hence, in this case, $\check{\alpha}^k$ is the stepsize that IRLS method takes at each iteration.

The following is true:

LEMMA 1. *The stepsize $\check{\alpha}^k$ as defined by (2.8) satisfies (2.4) with any $0 < \beta_f \leq \frac{2-p}{2p(p-1)}$.*

Proof. Let U^k be as defined by (2.7). Then

$$\begin{aligned}
\psi(x^k) - \psi(x^k + \check{\alpha} d_x^k) &\geq U^k(r^k) - U^k(r^k + \check{\alpha} d^k) \\
&= -\frac{1}{p} g^{k^T} d^k - \frac{1}{2} \check{\alpha}^k d^{k^T} \mathrm{diag}(p|r^k|^{2-p}) d^k \\
&= -\frac{1}{p} g^{k^T} d^k + \frac{1}{2} g^{k^T} d^k \\
&= -\frac{2-p}{2p} g^{k^T} d^k \qquad (\text{ from (2.8) }) \\
&= -\frac{2-p}{2p(p-1)} \check{\alpha}^k g^{k^T} d^k \\
&\geq -\beta_f \check{\alpha}^k g^{k^T} d^k.
\end{aligned}$$

Hence (2.4) is satisfied. ∎

A line search for $\phi(r)$ is conducted in the following manner.

Let α_*^k be the first positive breakpoint in $[\check{\alpha}, \rho_B]$, for some large constant $\rho_B > 0$, with $g(r^k + \alpha_*^k d^k)^T d^k \geq 0$ (if $p = 1$, the gradient just past α_*^k is used) and $\alpha_\sharp^k = \max\{\alpha_i^k : 0 \leq \alpha_i^k < \alpha_*^k\}$. If such a breakpoint α_*^k exists and (2.4) is satisfied at α_*^k, we set

$$\alpha^k = \alpha_\sharp^k + \tau(\alpha_*^k - \alpha_\sharp^k);$$

Otherwise, if at $\alpha = 1$, (2.4) is satisfied, $\alpha^k = 1$; If both the unit stepsize and α_*^k is rejected, $\alpha^k = \check{\alpha}^k$ is accepted.

The parameter τ, $0 < \tau < 1$, is used to avoid points with some $r_i = 0$. The basic idea is to perturb an acceptable point if it lands exactly on a breakpoint.

In §4, we will prove that this line search procedure leads to convergence for the IRLSL algorithm. Moreover, the line search procedure allows us to take a full Newton step or larger steps if possible.

Merle and Spath [10] have done an empirical study of the IRLS algorithm. It was concluded that the IRLS algorithm (without a line search) is satisfactory and seems to converge in all

Number of Iterations ($m = 100, \quad n = 50$)											
$p =$	1	1.01	1.02	1.03	1.04	1.05	1.06	1.07	1.08	1.09	1.1
IRLSL	50	50	36	50	40	38	28	33	30	29	24
IRLS	50	50	50	50	50	50	50	50	50	50	50

FIG. 1. *Behavior of Algorithms when p approaches one*

Number of Iterations ($m = 100, \quad n = 50$)					
p (no $r_i^* = 0$)	IRLS	IRLSL	p (five $r_i^* = 0$)	IRLS	MIRLS
1.3	32	14	1.3	31	18
1.4	23	8	1.4	22	12
1.5	17	9	1.5	17	11
1.6	13	7	1.6	13	8
1.7	10	7	1.7	10	7
1.8	8	5	1.8	8	7
1.9	6	5	1.9	6	5

FIG. 2. *Effect of zero residuals*

cases. We disagree with the claim. In order to investigate the performance of the algorithms more carefully, we apply both the IRLS and IRLSL algorithms to some random generated l_p norm problems. The following stopping criterion is used

$$\textbf{either} \quad \left|\frac{\psi(x^k) - \psi(x^{k+1})}{\psi(x^k)}\right| < \tau_s = \frac{1}{2} \times 10^{-11} \quad \textbf{or} \quad \text{itcount} > \text{maxit},$$

where *maxit* denotes the maximum number of iterations allowed. For more discussion on the stopping criteria, see §5.

Figure 1 and Figure 2 represent typical performance of the IRLS and IRLSL methods.

As indicated in Figure 1, both IRLS algorithms (with or without a line search) converge increasingly slowly when p approaches one. It is clear that, when $p = 1$, one can always find a solution with n zero residuals. Thus, when p is close to one, there usually exist zero residuals or extremely small residuals as well. Because the Hessian matrix of the objective function does not exist at points with zero residuals when $1 < p < 2$, it seems to be reasonable to blame the slow convergence on the occurrence of the zero residuals.

However, we argue this is not the reason. This is supported by the results in Figure 2 which indicate that the presence of zero residuals at solutions do not significantly affect the algorithm when p is further away from one. When a random l_p norm problem, $p > 1.5$, is generated, it usually does not have zero residuals at the solution. For comparison, we generate random l_p norm in a special way to guarantee the zero residuals at the solution: we solve a l_p norm problem first

and add more residuals so that they equal to zero at the solution. We notice that the algorithms seem to produce some superlinear steps asymptotically in some cases as well.

We believe the slow convergence of the both algorithms is due to the fact that the objective function becomes discontinuous when $p = 1$ and the the problem becomes a constrained problem.

In our opinion, the IRLSL is more efficient than IRLS. First, our computational experience indicates that the IRLSL method converges faster than the IRLS algorithm (e.g., Figure 1 and Figure 2): even when both methods fail to find a solution, the IRLSL finds an approximate solution with lower objective function value. The additional cost per iteration for IRLSL is that of the line search which is roughly $O(\kappa m)$ where κ is the number of positive breakpoints in $[\check{\alpha}, \rho_B]$ which have to be inspected in order to find α_*, i.e., an inner product needs to be computed at every such point. In our experiments, this number κ is in general much less than n and decreases quickly as p is away from unity. Thus, the cost of the line search is lower order compared to that of solving a least squares problem ($O(mn^2)$). Hence, we conclude that IRLSL is more efficient than IRLS and subsequently we will compare our new method (§3) to IRLSL only.

3. A New Algorithm. The IRLSL method works well when p is sufficiently away from unity (e.g., $p > 1.3$), as indicated by numerical results. However, when p is close to one, it becomes unsatisfactory. In this section, we develop a new method which works well for all $1 \leq p < 2$.

Let the rows of the matrix Z form a basis for the null space of A, i.e., $AZ^T = 0$. The following nonlinear system is equivalent to (2.1):

$$g - Z^T w = 0. \tag{3.1}$$

Denote $\lambda = Z^T w$. The Newton step for the above equations is defined by

$$\begin{bmatrix} p(p-1)\mathrm{diag}(|r|^{2-p})A^T & -Z^T \end{bmatrix} \begin{bmatrix} d_x \\ d_w \end{bmatrix} = - \begin{bmatrix} g - \lambda \end{bmatrix}. \tag{3.2}$$

Thus

$$d_x = -\frac{1}{p-1}(AD_r^{-1}\mathrm{diag}(|g|)A^T)^{-1}Ag,$$

which is equivalent to the Newton direction (2.3) for $\nabla\psi(x) = 0$.

Recall that $D_r = \mathrm{diag}(|r|)$ and $g = p|r|^{p-1}\sigma$. We consider the following nonlinear system of equations which is equivalent to (3.1) if D_r is nonsingular :

$$D_r(g - Z^T w) = 0. \tag{3.3}$$

If D_r is singular, a solution (r, w) to (3.3) is a solution to (3.1) if for any $r_i = 0$, $\lambda_i = 0$.

Assume for now that the Jacobian of $D_r(g - Z^T w)$ exists and is nonsingular. Let $D_\lambda = \mathrm{diag}(p^2|r|^{p-1} - \mathrm{sgn}(r)\lambda)$. Then the Newton step for (3.3) is defined by

$$\begin{bmatrix} D_\lambda A^T & -D_r Z^T \end{bmatrix} \begin{bmatrix} d_x \\ d_w \end{bmatrix} = - \begin{bmatrix} D_r(g - \lambda) \end{bmatrix}. \tag{3.4}$$

Hence, we obtain

$$AD_r^{-1}D_\lambda A^T d_x = -Ag, \tag{3.5}$$

or equivalently,

$$d_x = -(AD_r^{-1}D_\lambda A^T)^{-1}Ag. \tag{3.6}$$

By considering (3.3) instead of (3.1), we capture both the optimality conditions for smooth minimization ($p > 1$) and part of the optimality conditions (complementarity conditions) for nonsmooth minimization ($p = 1$). Since the objective function $\phi(r)$ becomes nearly nonsmooth, when p is close to one, we argue it is better to consider (3.3) instead of (3.1). Since we are concerned with the l_p norm problem for all $1 \leq p < 2$, taking Newton steps defined by (3.3) is more appropriate than that of (3.1). Indeed, this is one of the key observations that leads to our new method. If we define Newton step by (3.3) and globalize it as follows, it is reasonable to expect that the algorithm thus obtained will work well if p is sufficiently close to one.

When p is sufficiently away from one, by writing (3.1) as (3.3), it seems that we might complicate a smooth minimization problem, assuming there exists $r_i^* = 0$. We argue, however, that when p is far away it is usually the case that $r_i^* \neq 0, \forall 1 \leq i \leq m$. It is also interesting to notice that we can express the fact that the function is smooth through (3.3): optimality conditions simply require $\lambda^* = g^*$. Thus we may set

$$\lambda^{k+1} = g^{k+1}. \tag{3.7}$$

Using this definition of λ^{k+1}, $\eta^{k+1} = 0$ and $\theta^{k+1} = 0$. This leads (3.14) to the same Newton step (3.1).

When $p = 1$, (3.3) represents the complementarity slackness conditions for the l_1 problem. In [4], we have used it to define local Newton steps. We have proven that $AD_r^{-1}D_\lambda A^T$ is positive definite in the neighborhood of the solution, under some nondegeneracy assumptions. In addition, final quadratic convergence has been achieved.

If $1 < p < 2$ and no $r_i^* = 0$, then $D_r^{-1}D_\lambda$ is positive definite when close to the solution ($D_\lambda^* = (p-1)\text{diag}(|g^*|)$ because $\lambda^* = g^*$). Since A is assumed to have full rank, $AD_r^{-1}D_\lambda A^T$ is also positive definite when r is close to r^*. Hence the Newton direction becomes a descent direction in a neighborhood of the solution.

If there exists some $r_i^* = 0$, with $1 < p < 2$, the Jacobian matrix of (3.3) is singular at the solution because $g_i^* - \lambda_i^* = 0$. However, at those points, the Jacobian matrix of the original system (2.1) does not exist as well. Hence, this trouble is not introduced by considering (3.3) instead of (3.1). If, at a solution x^*, some $r_i^* = 0$, it is difficult to achieve quadratic convergence and we are content with a fast linear convergence.

Next, we discuss globalization of the Newton step (3.5).

First, we recall the technique used in [4] for $p = 1$. In [4], we globalize the Newton method by defining a diagonal matrix D_θ such that $AD_r^{-1}D_\theta A^T$ changes from $AD_r^{-1}A^T$ to $AD_r^{-1}D_\lambda A^T$ as solution is approached and replace D_λ by D_θ when determine a direction by (3.4). Thus the hybrid step can be considered as the solution to the following linear equations

$$[D_\theta A^T, D_r Z^T] \begin{bmatrix} d_x \\ d_\lambda \end{bmatrix} = -D_r(g - \lambda). \tag{3.8}$$

Moreover, the directions thus defined lead to global convergence with some suitable line search. Using a controlling variable $0 < \theta < 1$ which measures the closeness to the solution, the diagonal matrix D_θ is defined in the following way

$$D_\theta = |\theta\text{diag}(\sigma g) + (1-\theta)D_\lambda| = \text{diag}(|g - (1-\theta)\lambda|). \tag{3.9}$$

Here θ measures the satisfaction of the complementary slackness condition and the dual feasibility of an l_1 problem

$$\theta = \frac{\eta}{\gamma + \eta}, \qquad \eta = \max\{\max\{\frac{|D_r(g-\lambda)|}{\psi(r^0)}\}, \max\{\max\{|\lambda| - |g|, 0\}\}\}, \tag{3.10}$$

and $0 < \gamma < 1$. In this case, $\theta = 0$ is a necessary and sufficient condition of optimality. (For more detailed discussion, see [4]).

Now we consider the case where $1 < p < 2$. Since we know that the IRLS direction can lead to global convergence, we want to define a diagonal matrix D_θ such that globally the direction defined by replacing D_λ by D_θ is the same direction of that of the IRLS and locally it converges to D_λ. Notice that if we let $D_\theta = \text{diag}(|(p-1)g|)$, the direction defined by (3.8) equals the IRLS direction (2.3). Unfortunately, a simple scalar combination $\theta\text{diag}(g) + (1-\theta)D_\lambda$ does not lead to IRLS directions globally because $\text{diag}(g)$ may not dominant the combination when zero residuals exist. Instead, a controlling vector θ which measures the closeness to the solution is defined:

$$\theta = \frac{\eta e}{\gamma|g| + \eta e}, \qquad \eta = \max\{\max\{\frac{|D_r(g-\lambda)|}{\psi(r^0)}\}, \max\{\max\{|\lambda| - |g|, 0\}\}\}, \tag{3.11}$$

where γ is, again, a constant with $0 < \gamma < 1$. We similarly define a diagonal matrix D_θ in the following way

$$D_\theta = |\text{diag}(\theta)\text{diag}(p\sigma g) + \text{diag}(e-\theta)D_\lambda| = \text{diag}(|pg - (e-\theta)\lambda|). \tag{3.12}$$

The following lemma indicates that D_θ thus defined is appropriate.

LEMMA 2. *Suppose* $0 < \gamma < 1$. *Assume* θ *is defined by (3.12). Then* D_θ *satisfies*

$$(p-1)\text{diag}(|g|) \le |D_\theta| \le (p+2)\text{diag}(|g|). \tag{3.13}$$

Proof. By definition (3.12),

$$D_\theta = \text{diag}(|pg - (e-\theta)\lambda|).$$

From the definition (3.11) of θ,

$$\eta(e-\theta) = \theta\gamma|g|.$$

Hence,

$$(|\lambda| - |g|)(e-\theta) \le \gamma\theta|g|.$$

Therefore

$$\begin{aligned} |\lambda| &\le |g| + \gamma\frac{\theta|g|}{e-\theta} \\ &\le \frac{(e-\theta)|g| + \gamma\theta|g|}{e-\theta} \\ &\le \frac{|g| + |g|(\gamma-1)\theta}{e-\theta}. \end{aligned}$$

Hence

$$(p-1)\text{diag}(|g|) \leq (p-1+(1-\gamma)\theta)\text{diag}(|g|) \leq |D_\theta| \leq (p+2)\text{diag}(|g|).$$

■

It is clear that, when $p = 1$, (3.12) and (3.11) are the same as the definitions (3.9) and (3.10) used in [4]. Moreover, x is optimal if and only if there exists $\lambda = Z^T w$ such that $\eta = 0$. In §4, we prove that this globalization indeed leads to global convergence for $1 < p < 2$. Moreover, when there is no $r_i^* = 0$ at a solution, we have $(AD_r^{-1}D_\theta A^T) \to (AD_r^{-1}D_\lambda A^T)$ fast enough so that we can achieve superlinear convergence (see §4).

Instead of using (3.8) to compute (d_x, d_λ), we can compute d_x via solving a least squares problem

$$D^{-1}A^T d_x \overset{\text{l.s.}}{=} -Dg^k$$

where $D = (D_r D_\theta^{-1})^{\frac{1}{2}}$. Hence

$$\begin{cases} D^{-1}A^T d_x \overset{\text{l.s.}}{=} -Dg^k \\ d^k = A^T d_x^k. \end{cases} \tag{3.14}$$

Once $d = A^T d_x$ is computed, λ can be updated by

$$\lambda^+ = D_r^{-1} D_\theta d + g. \tag{3.15}$$

We perform the same line search as used in the IRLSL algorithm. The strict convex quadratic interpolation function $U^k(r)$ defined by (2.7) is again used to determine an appropriate stepsize. Let $\check{\alpha}$ be the minimizer of $U^k(r)$ along d^k defined by (3.14):

$$\check{\alpha}^k = -\frac{g^{kT} d^k}{d^{kT}\text{diag}(p|r^k|^{p-2})d^k} \tag{3.16}$$

The following is true:

LEMMA 3. *The stepsize $\check{\alpha}^k$ as defined by (3.16) is bounded below by $p-1$ and satisfies*

$$\psi(x^k + \check{\alpha}^k d_x^k) \leq \psi(x^k) + \frac{2-p}{2p} g^{kT} d^k.$$

Proof. By definition,

$$\begin{aligned} \check{\alpha}^k &= -\frac{g^{kT} d^k}{d^{kT}\text{diag}(p|r^k|^{p-2})d^k} \\ &= \frac{d^{kT} D_r^{k-1} D_\theta^k d^k}{d^{kT}\text{diag}(p|r^k|^{p-2})d^k} && (\text{ from (3.5) }) \\ &\geq (p-1)\frac{d^{kT} D_r^{k-1}\text{diag}(|g^k|)d^k}{d^{kT}\text{diag}(p|r^k|^{p-2})d^k} && (\text{ from (3.13) }) \\ &= (p-1). \end{aligned}$$

Let U^k be defined as in (2.7). Then we have

$$\begin{aligned}\psi(x^k) - \psi(x^k + \check{\alpha}^k d_x^k) &> U(r^k) - U(r^k + \check{\alpha}^k d^k) \\ &= -\frac{1}{p} g^{k^T} d^k - \frac{1}{2} \check{\alpha}^k d^{k^T} \mathrm{diag}(p|r^k|^{p-2}) d^k \\ &= -\frac{1}{p} g^{k^T} d^k + \frac{1}{2} g^{k^T} d^k \\ &\geq -\frac{2-p}{2p} g^{k^T} d^k.\end{aligned}$$

■

The minimizer of $U^k(r)$ along the direction d^k is then used to approximate the minimizer of $\phi(r)$ along d^k. Hence, we perform the line search for the new method in the exactly the same fashion except $\check{\alpha}^k$ and d^k are different:

Let α_*^k be the first positive breakpoint in $[\check{\alpha}, \rho_B]$, for some large constant $\rho_B > 0$, with $g(r^k + \alpha_*^k d^k)^T d^k \geq 0$ and $\alpha_\sharp^k = \max\{\alpha_i^k : 0 \leq \alpha_i^k < \alpha_*^k\}$. If such a breakpoint α_*^k exists and (2.4) is satisfied at α_*^k, we set

$$\alpha^k = \alpha_\sharp^k + \tau(\alpha_*^k - \alpha_\sharp^k);$$

Otherwise, if at $\alpha = 1$, (2.4) is satisfied, $\alpha^k = 1$; If both the unit stepsize and α_*^k is rejected, $\alpha^k = \check{\alpha}^k$ is accepted.

Quadratic interpolation technique has been used in line search methods for general nonlinear minimization [9]. However, it is worth emphasizing that, for general nonlinear functions, the interpolation function is usually an one dimensional function defined along a search direction instead of approximating the objective function in the entire space. The particular interpolation function $U^k(r)$ used here guarantee the stepsize is acceptable for $\phi(r)$ (i.e., sufficient decrease is achieved and the step is not too small) which cannot be achieved usually for general nonlinear functions.

Now we summarize the new method which is also referred to as GNCS: a globalized Newton method using the complementary slackness conditions for l_1 problems.

New Algorithm (GNCS)

Step 1 Compute a good initial point r^0

Step 2 Define $D^k = (D_r^k D_\theta^{k^{-1}})^{\frac{1}{2}}$ by (3.12) and (3.11) and $g^k = p|r^k|^{p-1}\mathrm{sgn}(r^k)$

Step 3 Compute the direction d^k by

$$\begin{cases} D^{-1} A^T d_x \overset{\text{l.s.}}{=} -Dg^k \\ d^k = A^T d_x^k. \end{cases}$$

Update λ^{k+1} by (3.15);

Step 4 Perform the line search as described on the piecewise function $\psi(r)$ directly. Then

$$r^{k+1} \leftarrow \alpha^k d^k + r^k, \quad k \leftarrow k+1.$$

Go to Step 2.

Notice that if we update λ^{k+1} by (3.7) instead, GNCS becomes the IRLSL. Indeed, the new algorithm and the IRLSL are computationally very similar. The only difference is that, for IRLSL, $D_\theta = \text{diag}(|(p-1)g|)$ and the multiplier information $\{\lambda^k\}$ (which we can obtain almost for free) is not used in defining descent directions.

4. Convergence Properties. As we have mentioned before, the new algorithm is equivalent to the method in [4] for l_1 problems in which the convergence has been established. It has been proven that the method is globally and quadratically convergent under some nondegeneracy assumptions. Thus we only need to consider the convergence of the new algorithm when $p > 1$.

For the IRLS without line search, the global convergence has been established in [12]. However, the convergence of the IRLSL still needs to be established. Moreover, we also need to prove the convergence for GNCS algorithm. The convergence result does not follow automatically from the convergence theory [7] for general descent algorithms because the objective function is not twice differentiable at the points with zero residuals. Furthermore, our line search procedure is not standard.

We first consider the global convergence for both the IRLSL and the new (GNCS) method.

Let P^k be the orthogonal projector onto null(ZD^k), i.e.,

$$P^k = I - D^k Z^T (Z(D^k)^2 Z^T)^{-1} Z D^k.$$

Assume D^k equals either $(D_r^k)^{\frac{1}{2}}$ or $(D_r^k D_\theta^{k-1})^{\frac{1}{2}}$. Then

$$\begin{aligned} d^k &= -A^T(AD^{k-2}A^T)^{-1}Ag^k \\ &= -D^k P^k D^k g^k \\ &= -(D^k)^2(g^k - \lambda^{k+1}), \end{aligned}$$

where $\lambda^{k+1} = Z^T w^{k+1}$ and w^{k+1} is the least squares solution to

$$D^k Z^T w^{k+1} \overset{\text{l.s.}}{=} D^k g^k.$$

From the line search procedure, we have $\alpha^k \geq \check{\alpha}^k$. Using Lemma 3 (or Lemma 1 for IRLSL), the following result becomes apparent.

LEMMA 4. *The stepsize $\{\alpha^k\}$ for GNCS algorithm (or IRLSL) is bounded below by $p-1$, i.e., $\alpha^k > p-1$.*

THEOREM 5. *Let D^k and d^k be defined by GNCS algorithm (or IRLSL). Then $\lim_{k\to\infty} \|P^k D^k g^k\|_2 = 0$ and $\lim_{k\to\infty} d^k = 0$.*

Proof. From the fact that $\psi(x)$ is bounded below, we have

$$\psi(x^k) - \psi(x^0) = \sum_{j=0}^{k-1} (\psi(x^{j+1}) - \psi(x^j)) \tag{4.1}$$

$$\leq \sum_{j=0}^{k-1} \max(\beta_f \alpha^j \nabla\psi^T d_x^j, \frac{2-p}{2p} \nabla\psi^T d_x^j) \tag{4.2}$$

with $\beta_f > 0$. Thus, $\lim_{k\to\infty} \max(\alpha^k \nabla\psi^T d_x^k, \nabla\psi^T d_x^k) = 0$.

From Lemma 4, $\alpha^k \geq (p-1)$. Hence,

$$\lim_{k\to\infty} \nabla\psi^T d_x^k = 0.$$

But $\nabla\psi^T d_x^k = g^{kT} d^k = -\|P^k D^k g^k\|_2^2$. Thus

$$\lim_{k\to\infty} \|P^k D^k g^k\|_2 = 0 \quad \text{and} \quad \lim_{k\to\infty} d^k = 0.$$

■

LEMMA 6. *Let $\{r^k\}$ be obtained by GNCS algorithm (or the IRLSL algorithm). Then the set of limit points of $\{r^k\}$ is closed and connected.*

Proof. We first prove that $\{\alpha^k d^k\}$ converges to zero. From the line search, α^k can be either unity, $\check{\alpha}^k$, or α^k satisfies (2.4). For Algorithm 1, $\check{\alpha}^k = p-1$. For Algorithm 2,

$$\begin{aligned}
\check{\alpha}^k &= -\frac{g^{kT} d^k}{d^{kT}\mathrm{diag}(p|r^k|^{p-2})d^k} \\
&= \frac{d^{kT} D_r^{k-1} D_\theta^k d^k}{d^{kT}\mathrm{diag}(p|r^k|^{p-2})d^k} \qquad (\text{ from (3.5) }) \\
&\leq \max\left(\frac{|pg^k - (e-\theta^k)\lambda^k|}{|g^k|}\right) \\
&\leq \max(|p|) + \max\left(\frac{\gamma|\lambda^k|}{\gamma|g^k| + \max\{|\lambda^k| - |g^k|, 0\}}\right) \qquad (\text{by definition (3.11)}).
\end{aligned}$$

But, it is easy to verify that

$$\max\left(\frac{\gamma|\lambda^k|}{\gamma|g^k| + \max\{|\lambda^k| - |g^k|, 0\}}\right) \leq 1.$$

Since $\{g^k\}$ is bounded, there exists $M > 0$, such that

$$\check{\alpha}^k \leq M.$$

Using the line search described, $\alpha^k \leq \rho_B$, we have

$$\lim_{k\to\infty} \alpha^k d^k = 0.$$

Since $\{r^k\}$ are bounded and $\{\alpha^k d^k\}$ converges to zero, the limit points of $\{r^k\}$ form a closed connected set ([11], page 478). ■

LEMMA 7. *Let $\{r^k\}$ be obtained by GNCS algorithm (or the IRLSL algorithm). Then $\{r^k\}$ converges to r^*.*

Proof. Let $\mathcal{S} = \{r \ : \ r \text{ is a limit point of } \{r^k\}\}$. Following Lemma 6, $\mathcal{S}$ is closed and connected.

Since $\{\phi(r^k) = \psi(x^k)\}$ is monotonically decreasing, bounded below and $\psi(r)$ is continuous, there exists an r^*, such that

$$\lim_{k\to\infty} \phi(r^k) = \phi(r^*).$$

Hence, for any limit point $r \in \mathcal{S}$, $\phi(r) = \phi(r^*)$. In addition, since $\mathcal{S}$ is closed, connected and $\phi(r)$ is strictly convex, $\mathcal{S}$ can only have one point.

Hence $\{r^k\}$ converges to r^*. ∎

THEOREM 8. *Let $\{r^k\}$ be obtained by GNCS algorithm (or the IRLSL algorithm). Assume that, at the limit point r^*, $\{a_i : b_i - a_i^T x = 0\}$ is linearly independent. Then $\{\lambda^k\}$ converges to λ^* and r^* is the solution of (1.1).*

Proof. Following Lemma 7, there exists r^* such that $\lim_{k\to\infty}\{r^k\} = r^*$.

From $\lim_{k\to\infty} D^k(g^k - Z^T w^{k+1}) = 0$, any limit point $\bar{w}$ of $\{\lambda^{k+1}\}$ satisfies $Z_1^T \bar{w} = g_1^*$. By assumption that, at the limit point r^*, $\{a_i : b_i - a_i^T x = 0\}$ is linearly independent, $Z_1^T w = g_1^*$ has a unique solution. Hence $\{\lambda^k\}$ is bounded and converges to λ^*.

We prove that $r^* \in \mathcal{S}$ is a solution by showing that $\lambda_i^* = 0$ if $r_i^* = 0$.

Assume otherwise, i.e., there exists some $\lambda_j^* \neq 0$ with $r_j^* = 0$. Assume there exists k_1 such that when $k \geq k_1$, all nonzero λ_j^k remains the same sign and $|r_j^k|^{p-1} < \lambda_j^k$ for all $r_j^* = 0$ with $\lambda_j^* \neq 0$.

Now consider the breaking point α_j^k defined by $r_j = 0$. Then

$$\alpha_j^k = \begin{cases} \frac{|pg_j^k - (1-\theta_j^k)\lambda_j^k|}{|g_j^k - \lambda_j^{k+1}|} & \text{for GNCS algorithm;} \\ -\frac{r_j^k}{\sigma_j^k |r_j^k|^{2-p}(g_j^k - \lambda_j^{k+1})} & \text{for the IRLSL.} \end{cases}$$

It is clear that $\{\alpha_j^k\}$ converges to zero because $\{g_j^k\}$ and $\{1 - \theta_j^k\}$ converge to zero.

Using Lemma 3 (or Lemma 1), if $r_j^{k_1}$ and $\lambda_j^{k_1+1}$ have different signs, $\alpha_j^k > 0$ can be passed because $\alpha^k > \check{\alpha}^k$. Thus, at the iteration $k = k_1 + 1$, λ_j^{k+2} and r_j^{k+1} will have the same sign. If, for $\hat{k} > k_1$, $r_j^{\hat{k}}$ and $\lambda_j^{\hat{k}+1}$ have the same sign, it will remain so for $k > \hat{k}$ because $g_j^k d_j^k > 0$. But this means $|r_j^k|$ will be increased for $k > k_1$. This contradicts $r_j^* = 0$. ∎

Now we discuss the local convergence properties of the two algorithms. If, at the solution r^*, there is some $r_i^* = 0$, the Hessian matrix of $\psi(x)$ does not exist at a corresponding x^*. Hence, theoretically, we do not expect superlinear convergence for either IRLSL or our new algorithm.

Assume that $r_i^* \neq 0$, for any $1 \leq i \leq m$. Then the Hessian matrix of $\psi(x)$ is positive definite at x^*. The IRLSL algorithm is locally equivalent to the Newton method for minimizing $\psi(x)$ which is a locally twice continuously differentiable function. Hence, following standard unconstrained minimization convergence analysis [7], the IRLSL is locally quadratically convergent. The new algorithm is locally equivalently to a quasi-Newton method for the minimization of a twice continuously differentiable function $\psi(x)$, with the Hessian matrix replaced by the matrix $A^T D_r^{-1} D_\theta A$. Moreover, we have

$$\begin{aligned} \lim_{k\to\infty} \frac{\|\nabla\psi(x^k) - \nabla^2\psi(x^k)d^k\|}{\|d^k\|} &= \lim_{k\to\infty} \frac{\|(A^T D_r^{k^{-1}} D_\theta^k A - A^T D_r^{k^{-1}} \mathrm{diag}(|(p-1)g^k|)A)d_x\|}{\|d^k\|} \\ &= 0, \end{aligned}$$

since $\{D_\theta^k - \mathrm{diag}((p-1)|g^k|)\}$ converges to zero. Following Theorem 6.4 in [6], $\{x^k\}$ converges superlinearly to x^*.

In summary, we have shown that both the IRLSL and our new algorithm are globally convergent under mild conditions. Assuming there is no zero residual at the solution, the IRLSL is *locally quadratically* convergent while the new algorithm is *locally superlinearly* convergent.

5. Numerical Experiments. In this section, we compare the computational performance of the IRLS with the line search (IRLSL) and GNCS algorithm proposed. The numerical results clearly show the superiority of the new approach over the IRLSL method (and thus IRLS method as well).

The dominant cost of the computation of the two methods is the same: solving a weighted least squares problem of the same dimension and structure per iteration. Moreover, the same line search procedure is used.

Now we discuss possible stopping criteria for problem (1.1).

Assume $1 < p < 2$. The optimality condition is simply $g^* = 0$ or equivalently $\eta^* = 0$. We point out, however, that the norm of g is generally not a good stopping criterion because it may not be satisfied numerically. When p is close to one, the gradient function $AD_r^{p-1}\sigma$ is ill conditioned in the neighborhood of a point where some $r_i = 0$, i.e., a small change of a variable may lead to a big change in the gradient. As an example, let us consider a simple scalar function $\psi(\xi) = |\xi|^{1.001}$. The gradient function is equal to $1.001\xi^{0.001}$. Even when $\xi = 2.2204 \times 10^{-16}$ which is machine precision on the SPARCstation, the gradient $\nabla\psi(\xi)$ equals 0.9656. Since the gradient should be zero when $\xi = 0$, it is clear the gradient function is extremely unstable and it is not possible to numerically satisfy gradient being zero.

In our computation, we terminate the calculation when an algorithm has stopped decreasing the objective function. More specifically, we stop the computation when

$$\textbf{either } \frac{|\psi^+ - \psi|}{\psi^+} < \tau_s \textbf{ or } \eta < \tau_s \textbf{ or } \text{itcount} > \text{maxit}$$

where *maxit* denotes the maximum number of iterations allowed and τ_s has been set to $\frac{1}{2}10^{-11}$.

The parameters required by the algorithms are set as follows:

$$\tau = 0.975, \qquad b_f = \epsilon, \qquad \gamma = 0.99$$

where ϵ is machine precision. We first generate some test problems from discrete approximation.

Function Approximation Problems: Approximate $f(z)$, evaluated at $z = 0, \frac{1}{m}, \cdots, 1$ by a polynomial of degree $n-1$: $\sum_{j=1}^{n} x_j z^{j-1}$ such that the l_p norm residuals is minimized. The two test functions are used

$$f_1(z) = \sqrt{1+z}, \qquad f_2(z) = exp(z) + \begin{cases} 5 & \text{if } 0.1 < z < 0.2; \\ 0 & \text{otherwise.} \end{cases}$$

The number of iterations required by both methods are tabulated in Figure 3. The new method (GNCS)is consistently better than IRLSL method. The first function $f_1(z)$ is continuous while the second function $f_2(z)$ is not smooth. For $f_1(z)$ and $p = 1.9$, the best l_p norm residual equals $\phi(r^*) = 4.97528518113 \times 10^{-10}$. For the second function $f_2(z)$, if $p = 1.9$, l_p norm residual $\phi(r^*)$ equals 1.7535105×10^2.

Random Problems: we also generate random test problems by producing the random elements for matrix A and right hand side b.

Figure 4 exhibits the number of iterations required by the GNCS algorithm and the IRLSL method whn $p = 1$. The IRLSL method stops after 50 iterations with the objective function having only a few digits of accuracy. The GNCS algorithm is essentially the method presented in [4] and demonstrates fast convergence.

$m = 200$, $n = 6$, $f_1(z)$		
p	GNCS	IRLSL
1	11	20
1.001	13	30
1.01	12	25
1.1	11	22
1.2	10	45
1.3	8	26
1.4	9	22
1.5	8	17
1.6	7	17
1.7	6	12
1.8	5	8
1.9	4	5

$m = 200$, $n = 10$, $f_2(z)$		
p	GNCS	IRLSL
1	12	50
1.001	11	50
1.01	15	50
1.1	10	33
1.2	9	23
1.3	7	27
1.4	8	20
1.5	6	15
1.6	6	12
1.7	6	11
1.8	6	7
1.9	4	6

FIG. 3. *function approximation problems,* maxit $= 50$, $\tau_s = \frac{1}{2}10^{-11}$

Number of Steps $m = 100$		
n	GNCS	IRLSL
10	12	50
30	14	50
50	12	50
70	13	50
90	14	50

Number of Steps $m = 200$		
n	GNCS	IRLSL
10	17	50
30	17	50
50	15	50
70	21	50
90	15	50
110	14	50
130	17	50
150	13	50
170	13	50
190	9	50

FIG. 4. $p = 1$, maxit $= 50$, $\tau_s = \frac{1}{2}10^{-11}$

Number of Steps			$m = 100$
n	GNCS	IRLSL	
10	11	27	9
20	14	46	11
30	20	50	6
40	16	50	6
50	16	50	4
60	17	50	6
70	14	50	5
80	11	50	6
90	13	37	4

Number of Steps			$m = 200$
n	GNCS	IRLSL	
10	15	38	12
30	18	50	4
50	15	50	8
70	17	50	5
90	21	50	5
110	15	50	7
130	17	50	5
150	14	50	5
170	18	50	3
190	13	50	6

FIG. 5. $p = 1.001$, maxit $= 50$, $r_s = \frac{1}{2}10^{-11}$

Number of Steps $m = 100$		
n	GNCS	IRLSL
10	12	34
30	12	50
50	13	50
70	13	50
90	16	50

Number of Steps $m = 200$		
n	GNCS	IRLSL
10	11	33
30	18	50
50	18	48
70	19	41
90	17	50
110	17	50
130	17	50
150	15	47
170	13	50
190	17	50

FIG. 6. $p = 1.01$, maxit $= 50$, $r_s = \frac{1}{2}10^{-11}$

When p is very close to one (e.g., see Figure 5 and 6 with $p = 1.001, 1.01$), the number of activities at a solution is usually slightly less than n. The GNCS algorithm exhibits final superlinear convergence behavior. This is because n of the residuals are in fact near activities. The algorithm treats it as a vertex and thus demonstrates superlinear behavior when approaching the neighborhood of the solution. The first column under IRLSL shows the number of iterations required to satisfy the stopping criteria. The second column indicates how many correct digits the approximate solution from IRLSL method have (relative to the solutions found by GNCS method which have lower values). The IRLSL method again shows extremely slow convergence and fails to find a solution after 50 iterations for majority of problems.

When p is further away from one (e.g., Figure 7 and 8 with $p = 1.1, 1.3$), the number of activities at the solution is less. However, many residuals are still relatively small. Hence the algorithm again approaches the neighborhood of solution with a few final superlinear steps. Here the IRLSL method found a solution with the required accuracy. But the number of iterations required is more than twice of that of the new algorithm (See Figure 7 and 8).

In the last case, p is significantly bigger than one, (e.g., Figure 9 with $p \geq 1.5$) there usually exists no activities at the solution. Thus, both the GNCS algorithm and the IRLSL method converge quickly to solutions and exhibit fast convergence. For these problems, the two methods behave roughly identically.

In summary, the GNCS algorithm works very well for all $1 \leq p < 2$. It always performs

Number of Steps		$m = 100$
n	GNCS	IRLSL
10	11	19
30	9	24
50	11	24
70	10	25
90	10	28

Number of Steps		$m = 200$
n	GNCS	IRLSL
10	10	15
30	11	26
50	12	28
70	11	26
90	11	29
110	10	34
130	10	27
150	12	27
170	10	37
190	10	32

FIG. 7. $p = 1.1$, maxit $= 50$, $\tau_s = \frac{1}{2}10^{-11}$

Number of Steps	$m = 100$	
n	GNCS	IRLSL
10	7	8
30	8	10
50	8	11
70	9	13
90	8	15

Number of Steps	$m = 200$	
n	GNCS	IRLSL
10	8	13
30	9	13
50	8	13
70	8	13
90	8	13
110	9	15
130	9	16
150	9	17
170	9	17
190	9	19

FIG. 8. $p = 1.3$, maxit $= 50$, $\tau_s = \frac{1}{2}10^{-11}$

Number of Steps	$m = 100$	
n	GNCS	IRLSL
10	6	5
30	6	8
50	7	7
70	9	7
90	8	11

Number of Steps	$m = 200$	
n	GNCS	IRLSL
10	5	5
30	6	7
50	6	7
70	6	6
90	7	7
110	7	7
130	6	6
150	7	10
170	8	9
190	7	11

FIG. 9. $p = 1.7$, maxit $= 50$, $\tau_s = \frac{1}{2}10^{-11}$

significantly better than the IRLSL when p is close to one ($p < 1.5$). When p is *very close or equal* to one, the IRLSL method is extremely inefficient while the GNCS method find the solutions in about 18 iterations in general. If p is *close* to one, the new algorithm is more than twice as fast as the IRLSL method. It is slightly better than the IRLSL when $p \geq 1.5$ and there exists no zero residual at a solution.

Finally, we emphasize that the number of iterations required by the GNCS method appears to be independent of the size of the problem, while the IRLSL seems to depend on the problem size when p is close to one (e.g., $p = 1.1$).

6. Conclusion. In this paper, we develop a new efficient method which solves the l_p norm minimization problem with $1 \leq p < 2$. We also further investigate the performance of the classical IRLS method and compare it with the new approach. We observe that the slow convergence of the IRLS (or IRLSL) method is not entirely due to the zero residuals at a solution but the fact that the constrained aspect is not taken care of.

The new method (GNCS) is attractive because of its capability of efficiently solving the l_p minimization problem with the entire range $1 \leq p < 2$. It is exactly the approach for l_1 presented in [4] when $p = 1$. It is quadratically convergent in this case. When $p > 1$, the new method is superlinearly convergent when there are no zero residuals at the solution.

The new method (GNCS) is significantly better than the IRLSL algorithm when p is close or equal to unity. Computational cost of each iteration of the two methods is the same: the main cost is solving a weighted least squares problem of the same size and structure. The difference between the two methods lies only in the definition of the different diagonal scaling matrix which defines descent directions: in our new method the multiplier information is incorporated in the diagonal scaling matrix and this is the key to a significant improvement.

7. Acknowledgement. The author would like to thank Thomas Coleman for numerous helpful suggestions.

REFERENCES

[1] I. BARRODALE AND F. ROBERTS, *An improved algorithm for discrete l_1 linear approximation*, SIAM J. Num. Anal., 10 (1972), pp. 839–848.

[2] R. H. BARTELS, A. R. CONN, AND J. W. SINCLAIR, *Minimization techniques for piecewise differentiable functions: the l_1 solution to an overdetermined linear system*, Siam J. Numer. Anal., 15 (1978), pp. 224–240.

[3] R. H. BYRD, *Algorithms for robust regression*, in Nonlinear Optimization 1981, M. Powell, ed., Academic Press, 1982, pp. 79–89.

[4] T. F. COLEMAN AND Y. LI, *A global and quadratic affine scaling method for linear l_1 problems*, Tech. Rep. 89–1026, Computer Science Department, Cornell University, 1989.

[5] ———, *A global and quadratic method for linear l_∞ problems*, Tech. Rep. In preparation, Computer Science Department, Cornell University, 1990.

[6] J. E. DENNIS AND J. J. MORÉ, *Quasi-Newton methods, motivation and theory*, SIAM Review, 19 (1977), pp. 46–89.

[7] J. E. DENNIS AND R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization*, Prentice-Hall, 1983.

[8] J. FISHER, *An algorithm for discrete linear l_p approximation*, Numerishche Mathematik, 38 (1981), pp. 129–139.

[9] P. GILL, W. MURRAY, AND M. WRIGHT, *Practical Optimization*, Academic Press, 1981.

[10] G. MERLE AND H. SPATH, *Computational experience with discrete l_p-approximation*, Computing, 12 (1974), pp. 315–321.

[11] J. M. ORTEGA AND W. C. RHEINBOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, 1970.

[12] M. R. OSBORNE, *Finite Algorithms in Optimization and Data Analysis*, Wiley Series in Probability and Mathematical Statistics, 1985.

[13] G. A. WATSON, *On two methods for discrete l_p approximation*, Computing, 18 (1977), pp. 263–266.

[14] J. M. WOLF, *On the convergence of an algorithm for discrete l_p approximation*, Numer. Math., 32 (1979), pp. 439–459.