INTERNATIONAL JOURNAL OF
NUMERICAL ANALYSIS AND MODELING
Volume 2, Supp , Pages 123–130

PRECONDITIONED HYBRID CONJUGATE GRADIENT ALGORITHM FOR P-LAPLACIAN

GUANGMING ZHOU, YUNQING HUANG* AND CHUNSHENG FENG

Abstract. In this paper, a hybrid conjugate gradient algorithm with weighted preconditioner is proposed. The algorithm can efficiently solve the minimizing problem of general function deriving from finite element discretization of the p-Laplacian. The algorithm is efficient, and its convergence rate is mesh-independent. Numerical experiments show that the hybrid conjugate gradient direction of the algorithm is superior to the steepest descent one when p is large.

Key Words. p-Laplacian, finite element approximation, hybrid conjugate gradient algorithm, numerical experiments

1. Introduction

Let Ω be a bounded open subset of R^2 with a Lipschitz boundary $\partial\Omega$. The p-Laplacian with Dirichlet data is the following equation (1.1):

$$\begin{aligned} -div(| \bigtriangledown u|^{p-2} \bigtriangledown u) &= f, \ in \ \Omega \\ u &= 0, \ on \ \partial\Omega \end{aligned}$$

where $1 < p < \infty$, $f \in L^2(\Omega)$, and $| \cdot |^2 = (\cdot, \cdot)_{R^2}$.

When $p = 2$, the equation (1.1) becomes a linear Laplacian equation. The equation (1.1) occurs in many mathematical models of physical process, for instances, glaciology, nonlinear diffusion and filtration(see Philip [21]), power-law materials(Atkinson and Champion [2]), and quasi-Newtonian flows(Atkinson and Jones [3]). The equation (1.1) is viewed as one of the typical examples of a large class of nonlinear problems. It contains most of the essential difficulties in studies of finite element approximations for this class of degenerate nonlinear systems. For this class of systems, many existing techniques in the finite element method, for example, the linearization method and deformation procedure, do not seem to work well.

Finite element approximations of p-Laplacian have been extensively studied in the literature, for example, in [10, 1, 12, 7, 8, 20]. In particular, the quasi-norm approach has proved quite successful in deriving sharp a priori and a posteriori error bounds for the finite element approximation of the degenerate systems. A priori and a posteriori error bounds for p-Laplacian are proposed by using quasi-norm approach in the paper [14, 15, 16].

Received by the editors January 1, 2004 and, in revised form, March 22, 2004.
2000 *Mathematics Subject Classification.* 49J20, 65N30.
This work was subsidized by the National Basic Research Program of China under the grant 2005CB321701, National Education Ministry of China. The first author was also supported by Research Fund of Hunan Provincial Education Department.

*Corresponding author.

Solving the equation (1.1) is equivalent to solve the following minimization problem:

$$\min_{v\in V} J(v) \tag{1.2}$$

where $V = W_0^{1,p}(\Omega)$,$1 < p < \infty$,and

$$J(v) = \frac{1}{p}\int_\Omega |\bigtriangledown v|^p - \int_\Omega fv \tag{1.3}$$

Huang, Li and Liu[13] proposed a steepest descent algorithm with weighted preconditioner which is solved by an algbric multigrid method. The decent algorithm has excellent computing efficiency for both p large or relatively small, for example, $p = 1000$ and $p = 1.5$, which are obviously superior to past methods. Tai and Xu[22] proposed a pure multigrid algorithm for solving the nonlinear problems including the p-Laplacian. Some theoretical and numerical analysis show the good efficiency.

It is well known that the conjugate gradients or their hybrid algorithms are more efficient than the steepest descent algorithm when solving nonlinear programming. Based on this thought, we proposed a hybrid conjugate gradient algorithm with weighted preconditioner in this paper. The new algorithm is more efficient than the descent one in the paper [13] for p-Laplacian for large p. The paper is organized as follows. Section 2 is devoted to mathematical preliminaries. In Section 3, we propose the hybrid conjugate gradient algorithm with weighted preconditioner. In Section 4, we present numerical results in order to compare and evaluate the performance of the new method and the steepest descent algorithm, and finally end, in Section 5, with some conclusions and discussions.

2. Preliminaries

Obviously, the functional $J(v)$ decided by (1.3) is strictly convex for $1 < p < \infty$. Furthermore, the equation (1.2) has a unique solution. It is well known that solving the equation (1.2) is equivalent to the following nonlinear PDE-the p-Laplacian:

$$(WP) \quad a(u,v) = \int_\Omega |\bigtriangledown u|^{p-2} \bigtriangledown u \bigtriangledown v = \int_\Omega fv, \quad \forall v \in V. \tag{2.1}$$

A direct calculation yields

$$J'(u)(v) = \int_\Omega |\bigtriangledown u|^{p-2} \bigtriangledown u \bigtriangledown v - \int_\Omega fv. \tag{2.2}$$

One can refer to the paper [9] for other conclusions of $J'(u)(v)$ and $J''(u)(v,w)$. We now introduce the finite element spaces. Let T^h be a regular triangulation of Ω^h, which is composed of disjoint open regular triangles K_i, that is , $\bar{\Omega}^h = \bigcup_{K_k\in T^h} \bar{K}_i$, where $h = \max_{K\in T^h} h_k$, and h_k denotes the diameter of the element K in T^h. When $i \neq j$, $\bar{K}_i \bigcap \bar{K}_j$ is void, or only one common vertex, or a whole edge.

Because of the limited higher order regularity for the solution of the p-Laplacian (see [2, 3, 22]), we shall only discuss the continuous piecewise linear element in this paper. Associated with T^h is a finite dimensional subspace V^h of $C^0(\bar{\Omega}^h)$, such that $\chi|_K \in \mathcal{P}_1$ for all$\chi \in V^h$ and $K \in T^h$, where $\mathcal{P}_1$ is the linear function space. Let

$$V_0^h = \{\chi \in V^h : \chi(x^k) = 0,\ for\ all\ x^k \in \partial\Omega^h\}$$

Then the finite element approximation of (WP) is as follows $(WP)^h$: Find $u_n \in V_0^h$ such that

$$(WP)^h \quad \int_{\Omega^h} |\bigtriangledown u_n|^{p-2} \bigtriangledown u_n \bigtriangledown v_n = \int_{\Omega^h} fv_h \tag{2.3}$$

According to previous discussion,we know that $(WP)^h$ has a unique solution u_h. Also $(WP)^h$ is equivalent to the following minimization problem:

$$\min_{v_h \in V_0^h} J(v_h). \tag{2.4}$$

3. Hybrid conjugate gradient algorithm with weighted preconditioner

In this section,we formulate a hybrid conjugate gradient method with weighted preconditioner for the p-Laplacian. Let $v_h, w \in V_0^h$. The steepest descent direction w of $J(v_h)$ is defined such that

$$J'(v_h)(w) = -\|J'(v_h)\|_* \|w\|. \tag{3.1}$$

For convenience, when computing descent direction w, we shall formulate our algorithm using the $H_0^1(\Omega)$ norm, which is the same as the norm in [13]. Convergence rate of our algorithm is mesh independent.

Let w be the exact solution of (1.2), and $u_n \in V_0^h$ be the current approximation.General formula finding next approximation u_{n+1} is

$$u_{n+1} = u_n + \alpha_n d_n, \tag{3.2}$$

where α_n is step length on search direction d_n. α_n is determined by a line search

$$J(u_n + \alpha_n d_n) = \min_{\alpha \geq 0} J(u_n + \alpha d_n) \tag{3.3}$$

Search direction d_n can be computed by using many different ways. For all $v \in V_0^h$, if d_n is equivalent to solutions of the following two PDE:

$$\int_\Omega \bigtriangledown w_n \bigtriangledown v = -J'(u_n)(v) = -\int_\Omega |\bigtriangledown u_n|^{p-2} \bigtriangledown u_n \bigtriangledown v + \int_\Omega fv, \tag{3.4}$$

$$\int_\Omega (\epsilon + |\bigtriangledown u_n|^{p-2}) \bigtriangledown w_n \bigtriangledown v = -J^{'}(u_n)(v) = -\int_\Omega |\bigtriangledown u_n|^{p-2} \bigtriangledown u_n \bigtriangledown v + \int_\Omega fv, \tag{3.5}$$

respectively, corresponding algorithms are called preconditioned steepest descent one and weighted preconditioned steepest descent one, respectively. In the paper [13], it is proved that w_n determined by (3.4) is the steepest descent direction in $H_0^1(\Omega)$ space, and the direction w_n determined by (3.5) is the steepest descent direction with $V \hookrightarrow H_0^1(\Omega)$ equipped a weighted norm $\|\cdot\|^2_{\epsilon,u_n} = \int_\Omega (\epsilon + |\nabla u_n|^{p-2}) |\nabla \cdot|^2$.

When $n > 0$, let

$$\beta_n = \max\{0, \min\{\beta_n^{FR}, \beta_n^{PRP}\}\}, \tag{3.6}$$

$$\tilde{\alpha}_n = \min_{\alpha \geq 0} J(u_n + \alpha(w_n + \beta_n d_{n-1})). \tag{3.7}$$

β_n^{FR}, β_n^{PRP} in (3.6) are computed by the following two formulae:

$$\beta_n^{FR} = \frac{\|w_n\|^2}{\|w_{n-1}\|^2},$$

$$\beta_n^{PRP} = \frac{(w_n - w_{n-1})^T w_n}{\|w_{n-1}\|^2},$$

respectively. In this paper, search direction d_n shall be determined by the following rule($\mathcal{R}$):

If $n = 0$, *then* $d_n = w_n$;

If $n > 0$, *then* $d_n = w_n$ *when* $\tilde{\alpha}_n = 0$; *or*

$$d_n = w_n + \beta_n d_{n-1}. \tag{3.8}$$

In a way, using the above rule $(\mathcal{R})$, instead of $\beta_n = \beta_n^{PRP}$ or $\beta_n = \max\{0, \beta_n^{PRP}\}$, is reasonable. There are two reasons. Firstly, if one computes β_n according to $\beta_n = \beta_n^{PRP}$,instead of (3.6),then d_n in (3.8) is likely close to $-d_{n-1}$ when β_n^{PRP} is a very large negative number. Obviously, $-d_{n-1}$ is not a good search direction. Secondly, β_k^{FR} has some nice convergence. The details can be found in [11].

Because of that d_n determined by the rule$(\mathcal{R})$ may be the steepest descent direction, or FR-conjugate gradient one, or PRP-conjugate gradient one, we call the following algorithm hybrid conjugate gradient algorithm with weighted preconditioner:

Algorithm 1 *Let* $n := 0$. *For a given initial value* u_0 *and two small positive constants* ϵ_1, ϵ_2, *do the following iterations:*

Step 1 For all $v \in V_0^h$, *solving the equation (3.5);*

Step 2 If $\|w_n\|_{\epsilon_1,u_n}/\|w_0\|_{\epsilon_1,u_0} < \epsilon_2$, *stop;*

Step 3 Computing search direction d_n *according to the rule* $(\mathcal{R})$*;*

Step 4 Finding step length α_n. *If* $\tilde{\alpha}_n \neq 0$, *then* $\alpha_n = \tilde{\alpha}_n$; *or computing* α_n, *such that* $J(u_n + \alpha_n w_n) = \min_{\alpha\geq 0} J(u_n + \alpha w_n)$*;*

Step 5 Updating iterative point. $u_n := u_n + \alpha_n w_n, n := n + 1$; *return Step 1.*

The direction w_n in Step 1 can be solved by fast AMG solvers.

4. Numerical experiments

We test Algorithm 1. The program language is Fortran 90. We used piecewise linear triangle finite element approximation in all our computations, and always used zero as an initial solution in all the iterations. The descent direction w_n is computed by an AMG solver. The stopping rule for the AMG iterations is to reduce the relative defect to 10^{-8} and the maximin V-Cycles in 50. The stopping criterion is $\|w_n\|_{\epsilon_1,u_n}/\|w_0\|_{\epsilon_1,u_0} < 10^{-6}$. We used a 0.618-section algorithm as the line search procedure. The current step length is used as an initial value for the initialization of the search interval at the next step. The parameters ϵ_1 and ϵ_2 are chosen to be 10^{-4} and 10^{-6}, respectively. A great deal of numerical experiments showed that efficiency of the algorithm is very high when $\epsilon_1 = 10^{-4}$. Simultaneously, discretion accuracy of object function and solution can be obtained.

Now we set out two numerical examples and their testing results.

Example 1 $\Omega = \{(x, y)| = r^2 = x^2 + y^2 < 1\},\ f = 1$. The exact solution is

$$u = u(r) = \frac{p-1}{p}\left(\frac{1}{2}\right)^{\frac{1}{p-1}}\left(1 - r^{\frac{p}{p-1}}\right). \tag{4.1}$$

In the tables below, $C1, C2, C3, C4$ represent the meshes with 1601,6221,24444, 97118 nodes, respectively. "ItN" and "CPU" mean iterative numbers and CPU time, respectively. "$\|\cdot\|$" indicates L^2-norm.

Tables 1 to 5 show the computational results using the conjugate gradient algorithm with weighted preconditioner(marked with WPCG) and the steepest descent algorithm with weighted preconditioner(marked with WPSD) in the paper [13].

Table 1 $p = 1.14$

	C1		C2		C3		C4	
	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG
ItN	17	17	18	17	17	16	16	15
CPU	0m26s	0m26s	1m39s	1m34s	5m59s	5m38s	23m16s	21m41s
$\|u - u_h\|$	1.60-5	1.59-5	5.35-6	5.40-6	2.52-6	4.66-6	4.85-6	3.09-6
$\|u_h - u_I\|$	1.23-5	1.22-5	4.39-6	4.44-6	2.21-6	4.44-6	4.80-6	3.03-6

Table 2 $p = 4$

	C1		C2		C3		C4	
	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG
ItN	9	9	9	9	9	9	9	8
CPU	0m13s	0m13s	0m46s	0m46s	2m56s	2m40s	14m06s	10m48s
$\|u - u_h\|$	5.10-4	5.10-4	1.28-4	1.28-4	3.18-5	3.17-5	7.98-6	8.11-6
$\|u_h - u_I\|$	6.75-5	6.66-5	1.66-5	1.63-5	4.03-6	4.03-6	1.70-6	1.05-6

Table 3 $p = 20$

	C1		C2		C3		C4	
	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG
ItN	31	21	28	20	23	19	24	20
CPU	0m40s	0m27s	2m18s	1m41s	7m33s	6m20s	32m08s	26m47s
$\|u - u_h\|$	1.39-3	5.10-4	3.77-4	3.77-4	9.65-5	9.66-5	2.57-5	2.61-5
$\|u_h - u_I\|$	5.63-4	5.64-4	1.68-4	1.68-4	4.75-5	4.80-5	1.26-5	1.31-5

Table 4 $p = 100$

	C1		C2		C3		C4	
	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG
ItN	79	51	86	59	71	60	64	57
CPU	1m49s	1m7s	8m20s	4m56s	28m26s	19m48s	90m06s	78m06s
$\|u - u_h\|$	3.42-3	3.42-3	1.08-3	1.08-3	3.16-4	3.16-4	9.13-5	9.15-5
$\|u_h - u_I\|$	2.61-3	2.61-3	8.74-4	8.74-4	2.67-4	2.67-4	7.81-5	7.84-5

Table 5 $p = 1000$

	C1		C2		C3		C4	
	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG
ItN	161	129	340	200	461	258	419	289
CPU	4m01s	2m55s	27m47s	17m14s	152m25s	85m40s	546m22s	396m23s
$\|u - u_h\|$	6.26-3	6.26-3	2.81-3	2.81-3	1.17-3	1.17-3	4.46-4	4.46-4
$\|u_h - u_I\|$	5.50-3	5.50-3	2.63-3	2.63-3	1.13-3	1.13-3	4.35-4	4.35-4

It is easy to see that the convergence of the two algorithms are almost mesh independent for a fixed p, and convergent rate tends to $O(h)$ as $p \to \infty$. Mostly, we can see that iterative numbers and CPU time of WPCG algorithm are less than that of WPSD algorithm by comparing results of the two algorithms when p is large. Therefore, we can conclude,to a certain extent, that hybrid conjugate gradient direction is superior to the steepest descent one when p is large. In addition, numerical overflow happen when $0 < p < 1.1$ or $p > 1000$. We can utilize WPSD algorithm to get some results when $p = 1.1$, but at the same time, when WPCG algorithm is used, numerical overflow came forth.

Example 2 $\Omega = \{(x, y)|x^2 + y^2 < 1\}, f = 2(x + y - x^2 - y^2)$.

We have no way to get analytic solution of the problem, so we only display iterative number and CPU time. Table 6 and 7 show the results which are obtained by using WPSD and WPCG algorithm when $p = 4$, $p = 100$, respectively. From this example, we can also see that WPCG algorithm is superior to WPSD algorithm in the paper [13] when p is large.

Table 6 $p = 4$

	C1		C2		C3		C4	
	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG
ItN	13	12	14	13	15	15	17	16
CPU	0m17s	0m16s	1m11s	1m06s	4m48s	4m53s	20m20s	20m40s

Table 7 $p = 100$

	C1		C2		C3		C4	
	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG	WPSD	WPCG
ItN	193	94	197	122	196	175	232	206
CPU	4m33s	2m17s	16m33s	10m41s	64m27s	58m46s	305m05s	272m11s

‘

Remark In the paper [13], for steepest decent algorithm with weighted preconditioner(WPSD), the inequality

$$J(u_n) - J(u_{n+1}) \geq \frac{c(J(u_n) - J(u))^2}{\|u_0 - u\|^2_{V_0^h}}, \tag{4.2}$$

where c is a positive number, u exact solution of the equation (1.1), u_0 initial value, is proved. It is the inequality (4.2) that guarantees convergence of WPCD algorithm. For Algorithm 1 in this paper, it is very difficult to prove above result (4.2). In order to ensure convergence of WPCG algorithm, we can use a restarting technique, change the rule $(\mathcal{R})$ and get the rule $(\mathcal{R}^*)$:

For a given positive integer,
If n can be divided exactly by l, namely, $mod(n, l) = 0$, then $d_n = w_n$;
If $mod(n, l) \neq 0$, then $d_n = w_n$ when $\tilde{\alpha}_n = 0$; or d_n is decided by (3.8).

In Algorithm 1, if the rule $(\mathcal{R}^*)$ is used, instead of the rule $(\mathcal{R})$, corresponding algorithm(marked with WPCG2) is obviously convergent according to the conclusions in the paper [13].

In Table 8, numerical results of WPCG2 algorithm in which $l = 10$ are displayed when $p = 1000$.

Table 8 $p = 1000$

	C1		C2		C3		C4	
	WPCG2	WPCG	WPCG2	WPCG	WPCG2	WPCG	WPCG2	WPCG
ItN	129	134	200	199	258	252	289	272
CPU	2m55s	3m43s	17m14s	18m16s	85m40s	86m05s	396m23s	372m22s
$\|u - u_h\|$	6.26-3	6.26-3	2.81-3	2.81-3	1.17-3	1.17-3	4.46-4	4.46-4
$\|u_h - u_I\|$	5.50-3	5.50-3	2.65-3	2.63-3	1.13-3	1.13-3	4.35-4	4.35-4

From Table 8 one can see that performance of WPCG2 algorithm is almost the same as that of Algorithm 1. For other p, the similar performance also happens.

5. Conclusions and discussions

Based on quasi-norm and the steepest descent algorithm with weighted preconditioner, we have replaced the steepest descent direction by hybrid conjugate gradient direction, proposed the hybrid conjugate gradient algorithm with weighted preconditioner, and stated convergence of the new algorithm with restarting technique. From the numerical results, we conclude that performance of the new algorithm is superior to the one in the paper [13] when p is large. The new algorithm, of course, has its weakness. For example, it is still a unsolvable problem how to computing the equation (1.1) when p is very close to 1.

Acknowledgments

The authors thank Li Ruo for help at numerical testing. This work was subsidized by the National Basic Research Program of China under the grant 2005CB321701, the National Education Ministry of China and the Research Fund of Hunan Provincial Education Department.

References

[1] Ainsworth, M. and Kay, D., The approximation theory for the p-version finite element method and application to non-linear elliptic PDEs, Numer. Math. Vol. 83(1999), pp.351-388.

[2] Atkinson, C. and Champion, C. R, Some boundary value problems for the equation $\nabla \cdot (\|\nabla\phi|^N) = 0$, Quart. J. Mech. Appl. Math.,37(1984), pp. 401-419.

[3] Atkinson, C and Jones, C. W., Similarity solutions in some nonlinear diffusion problems and in boundary-layer flow of a pseudo plastic fluid, Quart. J. Mech. Appl. Math., 27(1974), pp. 193-211.

[4] Barrett, J. W. and Liu, W. B., Finite element approximation of the p-Laplacian, Math. Comp., Vol. 61(1993), 523-537.

[5] Barrett, J. W. and Liu, W. B., Finite element approximation of some degenerate quasi-linear problems, Lecture Notes in Mathematics 303(1994),1-16.

[6] Barrett, J. W. and Liu, W. B., Quasi-norm error bounds for finite element approximation of quasi-Newtonian flows, Numer. Math., Vol.68(1994),437-456.

[7] Barrett, J. W. and Liu, W. B., Finite element approximation of some degenerate quasi-linera problems, Lecture Notes in Mathematics, Vol. 303(1994), pp. 1-16.

[8] Barrett, J. W. and Liu, W. B., Quasi-norm error bounds for finite element approximation of the p-Laplacian, Numer. Math., Vol. 68(1994),pp.437-456.

[9] Bermejo R. and Infante, J., A multigrid algorithm for the p-Laplacian, SIAM, J. Sci. Comput., Vol. 21, pp.1774-1789,2000.

[10] Chow, S. S., Finite element error estimates for non-linear elliptic equations of monitone type, Numer. Math. Vol. 54(1988), pp. 373-393.

[11] Dai, Y. H. and Yuan, Y. Y., Nonlinear Conjugate Gradient Methods (In Chinese), Shanghai:Shanghai Science and Technology Publisher, 2000.

[12] Farhloul, M., A mixed finite element method for a nonlinear Direchlet problem, IMA J. Numer. Anal. ,Vol. 18(1998), pp. 121-132.

[13] Huang, Y. Q., Li, R. and Liu, W. B., Preconditioned Descent Algorithms for p-Laplacian, Submitted, 2003.

[14] Liu, W. B. and Yan, N., Quasi-norm local error estimates for finite element approximation of p-Lapalacian, SIAM. J. Numer. Anal., Vol. 39(2001), 100-127.

[15] Liu, W. B. and Yan, N., Quasi-norm a posteriori error estimates for Non-conforming finite element approximation of p-Laplacian, Numer.Math., Vol.89, pp. 341-378, 2001.

[16] Liu, W. B. and Yan, N., On Quasi-norm Interpolation Error Estimates And a Posteriori Error Estimates for p-Laplacian, SIAM J. Numer. Anal., 2003.

[17] Liu, W. B. and Barrett, J. W., A remark on the regularity of the solutions of p-Laplacian and its applications to their finite element approximation, J. Math. Anal. Appl., Vol. 178(1993), pp. 470-488.

[18] Liu, W. B. and Barrett, J. W., A further remark on the regularity of the solutions of the p-Laplacian and its applications to their finite element approximation,J.Nonlinear Analysis,Vol. 21(1993), pp. 379-387.

[19] Liu, W. B. and Barrett, J. W., Higher order regularity for the solutions of some nonlinear degenerate elliptic equations, SIAM. J. Math. Anal., Vol. 24(1993), pp. 1522-1536.

[20] Padra, C., A posteriori error estimates for nonconforming approximation of some quasi-newtonian flows,SIAM J. Numer. Anal., Vol. 34(1997), 1600-1615.

[21] Philip, J. R., N-diffusion, Austral. J. Phys., 14(1961), pp. 1-13.

[22] Tai, X. C. and Xu, J. C., Global convergence of subspace correction methods for convex optimization problems, Math. Comp., 74(2002), pp. 105-124.

Department of Mathematics, University of Xiangtan, Xiangtan,Hunan 411105, P.R.China
E-mail: huangyq@xtu.edu.cn and zhougm@xtu.edu.cn