------ Raf. Jour. Sci., Vol.17, No.1, pp.59- 66, 2006 ------

New Sampling Three-Term Preconditioned PB Algorithm

Abbas Y. Al Bayati **Nada F. Hassen**
College of Computer and Mathematics Science
Mosul University

(Received 20/6/2001 , Accepted 26/9/2001)

ABSTRACT

In this paper a new sampling three-term preconditioned PB algorithm is developed and implemented.

The proposed sampling algorithm runs parallel to the original PCG Method Once a pair $\{v_k, y_k\}$ has been computed by the PCG Method the new sampling algorithm examines the search direction and decides the suitable selected search direction for the proposed algorithm. Numerical results have been done for (50) well-known test functions with different dimensions. The new proposed algorithm is very effective and promising in general.

خوارزمية Powell-Beal المشروطة ذات المقادير الثلاث النموذجية

الملخص

تم في هذا البحث استحداث وتنفيذ خوارزمية PB المشروطة ذات المقـادير الـثلاث النموذجيـة. الخوارزمية النموذجية المقترحة تم تنفيذها على التوازي مع طريقة PCG الاصلية. فحالما يـتم حـساب زوج $\{v_k,y_k\}$ بطريقة PCG تقوم الخوارزمية النموذجية باختبار اتجاه البحث وتحديد الاتجاه المناسـب. تم اجراء الحسابات العددية على (50) دالة اختبارية معروفة بابعاد مختلفـة. واظهـرت النتـائج الاثـر الايجابي الفعال للخوارزمية الجديدة.

INTRODUCTION

The Conjugate Gradient Method:

The basis of CG method is to determine new directions of search using information related only to the gradient of a quadratic objective function, in such away that successive search directions are conjugate with respect to the matrix G of that quadratic form. At each stage k the direction d_k is obtained by combining linearly the gradient, at $(x_k, -g_k)$, and the previous conjugate directions $d_o,d_1,\dots,d_{k-1}$, where the coefficients of the linear combination is chosen in such away the d_k is conjugate to all previous directions.

In order to improve the local rate of convergence and the efficiency of the classical CG methods, several established algorithms are discussed by Dixon's (1975) gradient predictioned method, Nazareth's (1977) three-term formula and Nazareth and Nocedal (1978a) multi-step method. They have all shown that such algorithms are able to generate conjugate directions for a quadratic without performing exact searches. For non-quadratic models (See Al-Bayati, 2001).

Powell–Beale Restart Algorithm:

This method is one of the most popular conjugate gradient algorithms, which is used for minimizing nonlinear functions of many variables.

Powell developed a new procedure for restarting CG method. He checked that the new direction d_{k+1} will be sufficiently downhill if:

$$-1.2\,||g_{k+1}||^2 > g_{k+1}\, d_{k+1} > -0.8\,||g_{k+1}||^2 \qquad \ldots(1)$$

(See Powell, 1977).

Powell–Beale Restart Algorithm:

```
Step (1): for any starting point $x_k$ , k=1, $\varepsilon$
Step (2): Set $d_k = -g_k$
Step (3): Compute $\lambda_k$ by using the line search procedure to minimize $f(x_k+\lambda_k d_k)$
Step (4): $x_{k+1} = x_k + \lambda_k d_k$
Step (5): Check for convergence if $||g_{k+1}|| < \varepsilon$ Stop, otherwise.
Step(6): if k>1 then
    $d_{k+1} = -g_{k+1} + \frac{g_{k+1}^T y_k}{d_t y_t} d_t + \frac{g_{k+1}^T y_k}{d_k^T y_k} d_k$
    where $y_k = g_{k+1} - g_k$
    otherwise
    $d_{k+1} = -g_{k+1} + \frac{g_{k+1}^T y_k}{d_k y_k} d_k$
    k = k + 1
    Goto Step (3)
    ENDIF
    Step (7): Check if k = n
    or
    $\left\|g_{k+1}^T g_k\right\| \geq 0.2 \left\|g_{k+1}\right\|^2$
    or
    $-1.2\,||g_{k+1}||^2 > d_{k+1}^T\, g_{k+1} > -0.8\,||g_{k+1}||^2$
    are satisfied then
    put k = 1 Goto step (2)
    otherwise
    set        k = k + 1
    Goto step (3)
```

PRECONDITIONING WITH QUASI-NEWTON APPROXIMATION

Preconditioning is a mean of improving the performance of CG methods.

Effective Preconditioning requires second derivative information which is not available. However, approximate curvature information can be accumulated as it is in a Quasi-Newton method. The storage consideration is temporarily neglected so that n×n Quasi-Newton approximations may be used. With some conditions on the line search, and for smooth nonlinear functions, the inverse Hessian approximation matrix H_k are positive definite and symmetric.

In general, a constant preconditioner H must be used throughout to obtain quadratic termination with conjugate gradient method. However, the property of quadratic termination is retained when Quasi–Newton preconditioners are used. In the BFGS updating formula for minimizing a function f (See Fletcher, 1987) we are given a symmetric and positive definite n×n matrix H_k that approximates the inverse of the Hessian of f, and a pair of n-vectors $v_k = x_{k+1} - x_k$ and $y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$ satisfying the condition $v_k^T y_k > 0$. Using this we compute a new inverse Hessian approximation H_{k+1} by means of the updating formula:

$$H_{k+1} = S_k^T H_k S_k + \rho_k v_k v_k^T \qquad \text{…(2)}$$

where

$$\rho_k = 1/\, y_k^T v_k \qquad S_k = I - \rho_k y_k v_k^T \qquad \text{…(3)}$$

We say that the matrix H_{k+1} is obtained by updating H_k once using the correction pair $\{v_k, y_k\}$.

Even if H_k is sparse, the new BFGS matrix H_{k+1} will generally be dense, so that storing and manipulating it is prohibitive when the number of variables is large. To circumvent this problem, the limited memory approach makes use of an alternative representation of the updating process in which Quasi- Newton matrices are not explicitly formed it follows from (2) and (3) that if an initial matrix H is update m times using the BFGS formula and the m pairs $\{v_i, y_i\}$, i= k-1,…,k-m, then the resulting matrix H(m) can be written as:

$$\begin{aligned} H(m) = {} & (S_{k-1}^T \ldots S_{k-m}^T)\overline{H}(S_{k-m} \ldots S_{k-1}) + \rho_{k-m}(S_{k-1}^T \ldots S_{k-m+1}^T) v_{k-m} v_{k-m}^T (S_{k-m+1} \ldots S_{k-1}) \\ & + \rho_{k-m+1}(S_{k-1}^T \ldots S_{k-m+1}^T) v_{k-m+1} v_{k-m+1}^T (S_{k-m+2} \ldots S_{k-1}) \\ & \vdots \\ & + \rho_{k-1} v_{k-1} v_{k-1}^T \end{aligned} \qquad \text{..(4)}$$

Thus instead of forming H(m) we can store the scalars p_i and the vectors $\{v_i, y_i\}$, i=k-1,…,k-m which determine the matrices S_i (See Gill and Murry, 1981).

The so-called BFGS method described in (See Nocedal, 1980; Gilbert and Lemarchal, 1989) updates Hessian approximation as follow. We first choose (usually diagonal) initial Hessian approximation H, and define the first m approximation through (4) as H(1),…, H(m). At this stage the storage is full, and to construct the new Hessian approximation, we first delete the oldest correction pair from the set $\{v_i, y_i\}$, to make room for the newest one, $\{v_k, y_k\}$. The new Hessian approximation H(m+1) is defined by (4), using the new set of pair $\{v_i, y_i\}$. This process is repeated during all subsequent iterations the oldest correction is removed to make space for the newest one.

Theorem: Let f be a strictly convex quadratic function and assume the line search is exact.

$d_0=H_0g_0$,

$$d_{k+1}=-H_kg_{k+1}+\frac{y_k^TH_kg_{k+1}}{y_k^Td_k}d_k \quad , \quad k\geq 0 \qquad \ldots(5)$$

$$H_{k+1}=H_{k-}\frac{H_ky_ky_k^TH_k}{y_k^THy_k}+\frac{v_kv_k^T}{v_k^Ty_k}+\theta_k\, w_k\, w_k^T \qquad \ldots(6)$$

Where

$$w_k=(y_k^THy_k)^{1/2}\left[\frac{v_k^T}{v_k^Ty_k}-\frac{H_ky_k}{y_k^THy_k}\right]$$

Each of H_i, $1\leq i\leq k$ is obtained from H_{I+1} with $\theta_k\geq 0$.

The initial approximation is H_0=H, a positive-definite and symmetric matrix. Then the direction (5) identical to the conjugate gradient directions preconditioned by H, that is:

$$d_{k+1}=-H_kg_{k+1}+\frac{y_k^THg_{k+1}}{y_k^Td_k}d_k \quad , \quad k\geq 0$$

and thus an algorithm with these search direction has quadratic termination. For more details (see Al-Bayati and Shareaf, 2001).

Nazareth and Nocedal (1978b) also showed that is not necessary to update the Hessian approximation at every iteration in order to have quadratic termination. The search direction (5) can also be used for general nonlinear functions. The matrices H_{k+1} have the property of hereditary positive definiteness as long as the line search guarantees that $v_k^Ty_k>0$.

NEW SAMPLING THREE-TERM PRECONDITION PB-ALGORITHM

We now present a formal description of the sampling algorithm that collects the pair $\{v_k, y_k\}$ as uniformly as possible the sampling algorithm runs parallel to the PCG method once a pair $\{v_k,y_k\}$ has been computed by the PCG method, the sampling algorithm examined the iteration index k and decides if the pair should be included in H (we denote the set of correction pairs that have been stored as H). When a new pair is accepted the algorithms checks the available space, and if the number of pairs in H is m, then a pair is chosen to leave H.

The Algorithm is started by inserting into H the first m pairs generated by the PCG process. After this, the entering and leaving pairs are chosen to keep an almost uniform distribution at any time.

Some properties of the sampling algorithm:

We now discuss some properties of the sampling algorithm. After the initialization in which the first m pairs are stored, the algorithm performs deletion and insertion operations controlled by the variable N1. For a given value of N1 the algorithm stored m/2 new pairs spaced by a distance 2^{N1}, and deletes the same of number of pairs. Deletion takes place in such away that the space created between two consecutive pairs is 2^{N1}. Therefore when N1 attains a new value, the distribution cases to be uniform and there is a transition period during which a new uniform distribution is generated: this is achieved at the end of the second loop. It follows that the larger m is, the longer it will take to move from one uniform distribution to the next (Morales and Nocedal, 1997).

New Algorithm (Sample).

```
Step (1): for any starting point $x_0$, $\varepsilon$, N, M, k=1, N1=1.
Step (2): Set $d_k = -H_k g_k$
Step (3): Set $ss_k = d_k$. Compute $\lambda_k$ by using line search procedure to minimize $f(x_k+\lambda_k d_k)$
Step (4): $x_{k+1} = x_k + \lambda_k d_k$
Step (5): Check for convergence if $\| g_{k+1}\| < \varepsilon$ then stop.
Step (6): if k<M then
update H by using Al-Bayati formula.
    $H_{k+1} = H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + sc \frac{v_k y_k^T}{v_k^T y_k} + w_k w_k^T$
    Where
    $sc = \frac{y_k^T H_k y_k}{y_k^T v_k}$
    When
    $w_k = (y_k^T H y_k)^{1/2} \left[ \frac{v_k^T}{v_k^T y_k} - \frac{H_k y_k}{y_k^T H y_k} \right]$
    then compute
    $d_{kH} = -H_k g_{k+1} + \frac{g_{k+1} H_k y_k^T}{d_k^T y_k} d_k$
Step (7): if k = N
    or $g_k^T d_k \geq 0$ then
    put k =1 Go to step (2)
Else
    k= k+1 Go To Step (3)
        ENDIF
ENDIF
Step (8): If k = 1+ (M/2+L-1)*2**N1
        then
        $k_2$ = 1+ (2*L-1) *2 **(N1-1)
        N1= N1+1
        Where L = 1,2, …, M/2
Step (9): If k $\leq k_2$ then
        $s_1 = H_k y_k$
        Else
        $s_1 = -H_k y_k$
ENDIF
Step (10): Update H by
```

$$H_{k+1} = H_k + \frac{s_1 s_1^T}{y_k^T s_1} + \frac{y_k^T s_1}{y_k^T v_k}\frac{v_k v_k^T}{v_k^T y_k} + w_k w_k^T$$

$$w_k = (y_k^T s_1)^{1/2}\left[\frac{v_k^T}{v_k^T y_k} - \frac{s_1}{y_k^T s_1}\right]$$

$$d_{k+1} = -H_k g_{k+1} + \frac{g_{k+1}^T H_k y_k}{d_k^T y_k} d_k + \frac{g_{k+1}^T y_k}{d_k^T y_k} d_k$$

Go To step (7).

ENDIF.

Go To step (6).

NUMERICAL RESULTS AND CONCLUSIONS

We shall discuss the computational results of the previous two algorithms (Powell–Beale preconditioned algorithm and the new sampling algorithm).

They are both written in Fortran 77 by using the computer "Pentium I" with double precision. The line search routine employed is the cubic fittings technique, fully described by (Banday, 1984). The stopping criterion is taken to be $\| g_{k+1} \|_2 < 2\times10^{-5}$.

The comparison test involves 50 well- known test functions, by considering both the TOTAL no. of function evaluations and the total no. of iterations. The results are reported in tables (1), (2), (3) with different dimensions.

Table (1) for (low dimensions)	$4 \le n \le 40$.
Table (2) for (medium dimensions)	$42 \le n \le 100$.
Table (3) for (high dimensions)	$200 \le n \le 500$.

Our numerical results indicate, in general, that the new proposed sampling algorithm is always improves the standard preconditioned PB-algorithm.

From Table (1) taking the standard preconditioned PB-algorithm as 100% NOI and NOF the new algorithm improves this standard one in about 50% NOI and 62% NOF for four selected versions namely (m = 1, 9, n/2 and n).

Clearly Table (2) deals with medium size test problems and the new algorithm (for all versions) has an improvement of about 56% NOI and NOF.

Finally, our numerical results in Table (3) show that there are an improvement of about 40% NOI and 45% NOF according to our calculations and our selected test functions.

In general, the new algorithm is promising in the filed of unconstrained optimization.

Table 1: Comparison of algorithms for 4 ≤ n ≤ 40

n	Test function	Standard PPB NOI (NOF)	New sampling algorithm			
			m=1 NOI (NOF)	M=9 NOI (NOF)	m=n/2 NOI (NOF)	m=n NOI (NOF)
4	Wood	59(187)	13(28)	13(28)	15(36)	13(28)
	Powell	21(72)	17(41)	17(41)	30(62)	20(69)
	Wolf	11(31)	9(19)	9(19)	10(22)	10(22)
	Rosen	22(73)	16(40)	16(40)	9(25)	13(34)
	Dixon	10(32)	10(30)	10(30)	10(31)	10(29)
	Cubic	19(58)	12(31)	12(31)	15(36)	13(34)
8	Wood	82(248)	13(28)	13(28)	16(35)	13(28)
	Powell	29(101)	20(48)	32(66)	34(74)	32(66)
	Wolf	17(39)	15(31)	16(33)	17(41)	16(33)
	Rosen	43(132)	14(36)	15(39)	16(41)	15(39)
	Dixon	16(44)	15(43)	15(43)	17(50)	15(43)
	Cubic	41(168)	11(32)	11(32)	11(31)	11(32)
20	Wood	19(47)	13(28)	13(28)	14(40)	13(28)
	Powell	35(104)	27(62)	22(57)	25(61)	28(66)
	Wolf	27(58)	26(53)	28(72)	28(64)	27(55)
	Rosen	34(89)	14(36)	15(39)	14(38)	14(36)
	Dixon	25(70)	21(65)	21(65)	21(65)	22(68)
	Cubic	44(472)	11(32)	11(32)	11(32)	11(32)
40	Wood	23(55)	13(28)	14(33)	13(28)	13(28)
	Powell	38(105)	28(63)	23(60)	29(65)	28(63)
	Cubic	43(103)	11(32)	12(34)	11(32)	11(32)
Total		658(2288)	329(866)	338(850)	366(909)	358(865)

* PPB stands for preconditioned Powell-Beale algorithm.

Table 2: Comparison of algorithms for 42≤n≤100

N	Test function	Standard PPB NOI (NOF)	New sampling algorithm			
			m=1 NOI(NOF)	m=9 NOI(NOF)	m=n/2 NOI(NOF)	m=n NOI(NOF)
42	Rosen	40(97)	14(36)	15(39)	14(38)	14(36)
0	Wood	24(60)	13(28)	14(33)	13(28)	13(28)
	Powell	47(125)	27(62)	34(70)	27(62)	27(62)
	Rosen	52(120)	14(36)	15(43)	14(36)	14(36)
	Dixon	34(98)	23(72)	24(76)	23(72)	23(72)
	Cubic	47(133)	11(32)	12(34)	11(32)	11(32)
80	Wood	23(55)	13(28)	14(30)	13(28)	13(28)
	Powell	48(141)	28(62)	34(72)	28(62)	28(62)
	Wolf	75(153)	41(83)	42(85)	42(86)	41(83)
	Rosen	52(126)	14(36)	15(43)	14(36)	14(36)
	Dixon	34(98)	22(69)	24(82)	22(69)	22(69)
82	Cubic	63(154)	11(32)	12(34)	11(32)	11(32)
100	Powell	52(162)	27(61)	33(73)	27(61)	27(61)
	Wolf	86(175)	42(85)	42(85)	42(85)	42(85)
	Rosen	64(163)	14(36)	15(39)	14(36)	14(36)
	Dixon	34(97)	22(69)	24(88)	22(69)	22(69)
	Cubic	75(199)	11(32)	12(34)	11(32)	11(32)
Total		857(2156)	347(859)	381(960)	348(864)	347(859)

Table 3: Comparison of algorithm for $200 \leq n \leq 500$

n	Test function	Standard PPB NOI (NOF)	New sampling algorithm m=1 NOI(NOF)	m=9 NOI(NOF)	m=n/2 NOI(NOF)	m=n NOI(NOF)
200	Powell	52(149)	29(66)	28(73)	29(66)	29(66)
	Dixon	39(109)	22(68)	22(68)	22(68)	22(68)
	Edgar	7(19)	7(18)	7(18)	7(18)	7(18)
	Wolf	78(168)	43(87)	43(87)	43(87)	43(87)
	Rosen	37(92)	14(36)	15(39)	14(36)	14(36)
300	Powell	47(121)	26(60)	33(68)	26(60)	26(60)
	Dixon	38(107)	22(68)	22(68)	22(68)	22(68)
	Edgar	7(19)	7(18)	7(18)	7(18)	7(18)
400	Powell	49(133)	27(63)	28(76)	27(63)	27(63)
	Dixon	38(104)	22(68)	23(73)	22(68)	22(68)
500	Edgar	7(19)	7(18)	7(18)	7(18)	7(18)
Total		406(1059)	233(588)	242(624)	233(588)	233(588)

- PPB stands for preconditioned Powell-Beale algorithm.

REFERENCES

Al-Bayati, A.Y., 2001. New generalized CG-Method for the non–quadratic model in unconstrained optimization. J. Al-Yarmouk, Jordan, Vol.101, pp.1-17.

Al-Bayati, A.Y. and Shareaf, S.A., 2001. A modified PCG-method for non linear optimization. Raf. Jour. Sci., Vol.12, 60-70.

Banday, B.D., 1984. Basic optimization methods. Edward Arnold, London.

Dixon, L.C.W., 1975. Conjugate gradient algorithm: quadratic termination without linear searches. Jour. Inst. Math. App.

Fletcher, R., 1987. Practical Methods of Optimization, 2nd Edition, John Wiley and Sons, New York.

Gilbert, J.C. and Lemarechal, C., 1989. Some numerical experiments with variable storage quasi-Newton algorithms. Math. Prog., Vol.45, pp.407-436.

Morales, J. and Nocedal, J., 1977. Automatic Preconditioning by limited memory Quasi-Newton update. Technical Report OTC 97/08, Optimization Technology Center, Northwestern University.

Nazareth, L., 1977. A Conjugate direction without line searches, JOTA, 23 p.

Nazareth, L. and Nocedal, J., 1978a. Properties of Conjugate gradient algorithm with in exact line searches. Report 71, Systems optimization laboratory, Dept. O. R., Standford University.

Nazareth, L. and Nocedal, J., 1978b. A study of Conjugate gradient methods. Technical report sol 78-29, Dept. of operational research, Stanford University.

Nocedal, J., 1980. Updating Quasi-Newton matrices with limit storage. Math. Comp., Vol.35, pp.773-782.

Powell, M.J.D., 1977. Restart procedure for the conjugate gradients. Math. Programming.