레이레이 계수의 최소화에 의한 내부고유치 계산을 위한 병렬준비행렬들의 비교

마 상 백†·장 호 종††

요 약

최근에 CG 반복법을 이용하여 레이레이 계수를 최소화함으로써 대칭행렬의 내부고유치를 구하는 방법이 개발되었다. 그리고 이 방법은 병렬계산에 매우 적합하다. 적절한 준비행렬의 선택은 수렴속도를 향상시킨다. 우리는 본 연구에서 이를 위한 병렬준비행렬들을 비교한다. 고려된 준비행렬들은 Point-SSOR, 다중색채하의 ILU(0)와 Block SSOR이다. 우리는 128개의 노드를 가진 CRAY-T3E에서 구현하였다. 프로세서간의 통신은 MPI 리이브러리를 사용하였다. 최고 512×512 행렬까지 시험하였는데 이 행렬들은 타원형 편미분방정식의 근사화에서 얻어졌다. 그 결과 다중색채 Block SSOR이 가장 성능이 우수한 것으로 판명되었다.

Comparisons of Parallel Preconditioners for the Computation of Interior Eigenvalues by the Minimization of Rayleigh Quotient

Sangback Ma†·Ho-Jong Jang††

ABSTRACT

Recently, CG (Conjugate Gradient) scheme for the optimization of the Rayleigh quotient has been proven a very attractive and promising technique for interior eigenvalues for the following eigenvalue problem,

$$Ax = \lambda x \tag{1}$$

The given matrix A is assummed to be large and sparse, and symmetric. Also, the method is very amenable to parallel computations. A proper choice of the preconditioner significantly improves the convergence of the CG scheme. We compare the parallel preconditioners for the computation of the interior eigenvalues of a symmetric matrix by CG-type method. The considered preconditioners are Point-SSOR, ILU (0) in the multi-coloring order, and Multi-Color Block SSOR (Symmetric Succesive OverRelaxation). We conducted our experiments on the CRAY-T3E with 128 nodes. The MPI (Message Passing Interface) library was adopted for the interprocessor communications. The test matrices are up to 512×512 in dimensions and were created from the discretizations of the elliptic PDE. All things considered the MC-BSSOR seems to be most robust preconditioner.

키워드 : Eigenvalue Problem, CG, Preconditioning, Parallel, Multi-Color Block SSOR

1. Introduction

Recently, an idea was proposed to find the interior eigenvalues by Rayleigh quotient minimization by CG (Conjugate Gradient)-type method [1, 4-7]. Also, this method is very amenable to the parallel computation. Iterative solution of eigenvalue problems or linear systems requires a preconditioning to accelerate the convergence [13]. Incomeple Cholesky factorization is one of the most popular technique. But Incomplete Cholesky factorization is inherently *serial*, and it might not converge for ill-conditioned matrices. Another choice is the block-type parallel preconditioner, Multi-Color Block SSOR (Symmetric Successive OverRelaxarion) preconditioner. Multi-coloring is a simple way to achieve the parallelisim of order (N), where N is the order of the matrix. Block SSOR is believed to reduce interprocessor communications.

We present results from our numerical experiments drawn from the FDM discretizations of the elliptic partial differential equations. The experiments were done on the CRAY-T3E of KISTI, Daejon, Korea.

2. Eigenproblem via Preconditioned CG

2.1 Minimization of Rayleigh Quotients

Consider the eigenvalue problem

† 종신회원 : 한양대학교 전자컴퓨터 공학부 교수
†† 비 회 원 : 한양대학교 수학과 교수
논문접수 : 2002년 5월 7일, 심사완료 : 2002년 9월 13일

$$Ax = \lambda x, \tag{2}$$

where A is a large sparse symmetric positive definite matrix of dimension n.

Let

$$0 < \lambda_1 < \lambda_2 \le \lambda_3 \le \cdots \le \lambda_n$$

be the eigenvalues of equation (1), and let $z_1, z_2, \cdots, z_n$ be the corresponding eigenvectors.

We recall that the eigenvectors of equation (1) are the stationary points of the Rayleigh quotient

$$R(x) = \frac{x^T A x}{x^T x} \tag{3}$$

and the gradient of $R(x)$ is given by

$$g(x) = \frac{2}{x^T x}\,[Ax - R(x)x].$$

For an iterate $x^{(k)}$, the gradient of $R(x^{(k)})$,

$$\nabla R(x^{(k)}) = g^{(k)} = \frac{2}{x^{(k)^T} x^{(k)}}[\,Ax^{(k)} - R(x^{(k)})\,x^{(k)}],$$

is used to fix the direction of descent $p^{(k+1)}$ in which $R(x)$ is minimized.

These directions of descent are defined by

$$p^{(1)} = -g^{(0)},\ \ p^{(k+1)} = -g^{(k)} + \beta^{(k)} p^{(k)},\ k = 1, 2, \cdots,$$

where $\beta(k) = \dfrac{g^{(k)^T} g^{(k)}}{g^{(k-1)^T} g^{(k-1)}}$ [1, 10]. The subsequent iterate $x^{(k+1)}$ along $p^{(k+1)}$ through $x^{(k)}$ is written as

$$x^{(k+1)} = x^{(k)} + \alpha^{(k+1)} p^{(k+1)},\quad k = 0, 1, \cdots,$$

where $\alpha^{(k+1)}$ is obtained by minimizing $R(x^{(k+1)})$,

$$R(x^{(k+1)}) = \frac{x^{(k)^T} A x^{(k)} + 2\alpha^{(k+1)} p^{(k+1)^T} A x^{(k)} + \alpha^{(k+1)^2} p^{(k+1)^T} A p^{(k+1)}}{x^{(k)^T} x^{(k)} + 2\alpha^{(k+1)} p^{(k+1)^T} x^{(k)} + \alpha^{(k+1)^2} p^{(k+1)^T} p^{(k+1)}}$$

A detailed explanation to get the values for $\alpha^{(k+1)}$ can be found in [7].

The performance of the CG method for computing the eigenpairs of equation (1) can be improved by using a preconditioner [2, 13]. The idea behind the PCG is to apply the "regular" CG scheme to the transformed system

$$\widetilde{A}\,\widetilde{x} = \lambda \widetilde{x},$$

where $\widetilde{A} = C^{-1}AC^{-1}$, $\widetilde{x} = Cx$, and C is nonsingular symmetric matrix. By substituting $x = C^{-1}\widetilde{x}$ into equation (3), we obtain

$$R(\widetilde{x}) = \frac{\widetilde{x}^T C^{-1} A C^{-1} \widetilde{x}}{\widetilde{x}^T C^{-1} C^{-1} \widetilde{x}} = \frac{\widetilde{x}^T \widetilde{A}\,\widetilde{x}}{\widetilde{x}^T \widetilde{x}}, \tag{4}$$

where the matrix $\widetilde{A}$ is symmetric positive definite. The transformation equation (4) leaves the stationary values of equation (3) unchanged, which are eigenvalues of equation (2), while the corresponding stationary points are obtained from $\widetilde{x}_j = C z_j,\ j = 1, 2, \cdots, n$. The matrix $M = C^2$ is called the preconditioner. There are a number of choices of M ranging from simple to complicated forms. In this paper, Multi-Color Block SSOR preconditioner is used with parallel computation aspect. The PCG algorithm for solving the smallest eigenpair with implicit preconditioning is summarized in [7]. After the smallest eigenvalue is found, we apply the orthogonal deflation to compute the next eigenvalues.

3. ILU (0) factorization

Meijerink and Van. der Vorst [8] introduced a so-called Incomplete LU (ILU) preconditioner for symmetric matrices. The following is a modification of the original ILU for nonsymmetric matrices, as described in [3]. Let $A = LU + N$, where $L_{i,j} = U_{i,j} = 0$ if $A_{i,j} = 0$ and $N_{i,j} = 0$ if $A_{i,j} \neq 0$. Let NZ(A), the *nonzero pattern* of A, denote the set of pairs of (i, j) for which $A_{i,j}$, the(i, j) entry of A, is nonzero.

```
For i = 1, ··· , Until N Do
    For j = 1, ··· , Until N Do
        If ((i, j) belongs to NZ (A)) then
            s_{i,j} = A_{i,j} − Σ_{t=1}^{min(i,j)−1} L_{i,t} U_{t,j}
            if (i ≥ j) then L_{i,j} = s_{i,j}
            if (i < j) then U_{i,j} = s_{i,j} / L_{i,i}
        Endif
    Endfor
Endfor
```

(Algorithm 3.1) ILU Factorization

4. Point-SSOR algorithm

1. *Choose* x_0
2. *For* $i = 0, \cdots$ *Do*

$$(D - \omega E)\,x_{i+\frac{1}{2}} = ((1-\omega)D + \omega F)x_i + \omega b \tag{5}$$

$$(D - \omega F)\,x_{i+1} = ((1-\omega)D + \omega E)\,x_{i+\frac{1}{2}} + \omega ba$$

Endfor

(Algorithm 3.2) *Point-SSOR Let A = D − E − F, where D is the diagonal part −E, is the lowertriangular part and −F the uppertriangular part*

5. Multi-Color Block SSOR (Symmetric Successive OverRelaxation) Method

Multi-Coloring is a way to achieve parallelism of order N, where N is the order of the matrix. For example, it is known that for 5-point Laplacian we can order the matrix in 2-colors so that the nodes are not adjacent with the nodes with the same color. This is known as Red/Black ordering. For planar graphs maximum four colors are needed.

Blocked methods are useful in that they minimize the interprocessor communications, and increases the convergence rate as compared to point methods. SSOR is a symmetric preconditioner that is expected to perform as efficiently as incomplete Cholesky factorization combined with blocking. Instead we need to invert the diagonal block. In this paper we used the MA48 package from the Harwell library, which is a direct method using reordering strategy to reduce the fill-ins. Since MA48 type employ some form of pivoting strategy, this is expected to perform better for ill-conditioned matrices than Incomplete Cholesky factorization, which does not adopt any type of pivoting strategy.

SSOR needs a ω parameter for overrelaxation. However, it is known that the convergence rate is not so sensitive to the ω parameter.

Let the domain be divided into L blocks. Suppose that we apply a multi-coloring technique, such as a greedy algorithm described in [11], to these blocks so that a block of one color has no coupling with a block of the same color. Let D_j be the coupling within the block j, and color(j) be the color of the j-th block. We denote by $U_{j,k}$, $k = 1, q$, $j < k$ and $L_{j,k}$, $k < j$ the couplings between the j-th color block and the k-th block.

Then, we can describe the Multi-Color Block SSOR as follows.

Let q be the total number of colors, and color (i), i = 1, L, be the array of the color for each block.

1. *Choose* u_0, *and* $\omega > 0$
2. *For i > 0 Until Convergence Do*
3.   *For kolor = 1, q Do*
4.   *For j = 1, L Do*
5.   *if (color (j) == kolor) then*
6. $(u_{i+1/2})_j = D_j^{-1}(b - \omega * \sum_{k \neq kolor}^{k=q} L_{j,k} u_{i+1/2})$
7.   *endif*
8.   *Endfor*
9.   *For kolor = 1, q Do*
10.   *For j = 1, L Do*
11.   *if (color (j) == kolor) then*
12. $(u_{i+1})_j = D_j^{-1}(u_{i+1/2} - \omega * \sum_{k \neq kolor}^{k=q} U_{j,k} u_{i+1})$
13.   *endif*
14.   *Endfor*
15.   *Endfor*
16. *Endfor*

(Algorithm 5.1) Multi-Color Block SSOR

Note that the innermost loop in line six and seven can be executed in parallel.

6. Experiments

6.1 Test problems

- **Problem 1** Elman's problem [3]

$$-(bu_x)_x - (cu_y)_y + fu = g \tag{6}$$
$$\Omega = (0,1) \times (0,1)$$
$$u = 0 \ on \ \delta\Omega$$

where $b = \exp(-xy)$, $c = \exp(xy)$, $f = \frac{1}{(1+xy)}$,

and g is such that exact solution

$$u = x\exp(xy)\sin(\pi x)\sin(\pi y)$$

- **Problem 2** Cylinder Shell problem from Harwell/Boeing Collection

1. *s1rmq4m1.dat*

6.2 Results

<Tables 1>, <Tables 2> contain the timings for the three preconditioners. We used MPI (Message Passing Machine) library for the interprocessor communications. For the first problem we used the Block-Row mapping for the graph partitioning of the matrix. For the second problem we have used the Metis code developed by V. Kumar of the University of Minnesota. The number of colors needed is two for the first problem and reaches 6 for the three dimensional matrix of problem two. For the multi-coloring we have adopted the greedy heuristic as described in [11]. The ε parameter was set to be 10^{-6} for stopping criterion. 'X' stands for the cases with insufficient memory, and 'SL' for the cases where the convergence was not obtained within reasonable amount of time.

The first problem is a well-conditioned matrix. However, the second problem is ill-conditioned, coming from Cylinder Shell problem of Harwell/Boeing collection, with the condition number of 1.8^6. It is reported in [2] that for the second problem Incomplete Cholesky Factorization preconditioning does not achieve the convergence. But MC-BSSOR does achieve the convergence partially. As for the ω parameter we have set ω to be 1.

For the inversion of diagonal blocks in Block SSOR method, we have used the MA48 routine of the Harwell library, which adopts direct methods for sparse matrices with the reordering strategy reducing fill-ins. The cost of the MA48 is roughly proportional to L^2, where L is the size of the matrix. Since L is roughly N/p we expect a quadratic decrease with the increasing number of processors.

〈Table 1〉 Problem 1 with FDM

	p = 4	p = 8	p = 16	p = 32	p = 64
	Point-SSOR/MC-ILU(0)/MC-BSSOR				
$N = 128^2$	17.4/14.3/19.0	10.2/8.0/9.7	11.0/7.0/10.0	13.5/8.4/5.1	19.7/11.5/6.6
$N = 256^2$	94.3/73.6/139	54.0/40.5/61.7	38.7/28.2/56.6	57.2/22.7/19.6	140/25.3/15.1
$N = 512^2$	×	×	184/135/225	129/85.4/118	118/69.7/66.9

〈Table 2〉 Cylinder Shell problem from the Hatwell/Boeing Collection

	p = 4	p = 8	p = 16	p = 32	p = 64
	Point-SSOR/MC-ILU(0)/MC-BSSOR				
s1rmq4ml	SL/SL/102	SL/SL/52.1	SL/SL/SL	SL/SL/152.4	SL/SL/SL

7. Conclusions

- For the first problem with the small number of processors Multi-Color ILU (0) gives better performance, but with the large number of processors MC-BSSOR shows the best performance. For the cylinder shell problem only the MC-BSSOR converges partially. We believe that for the shell problem we need a different approach.
- Due to the nature of MA48 library, we expect MC-BSSOR to be scalable with the increasing number of processors.
- Out of the three preconditioner considered the MC-BSSOR seems to be the most robust preconditioner.
- Our CG-type method is far simpler than the Jacobi-Davidson method, in terms of the preconditioning.

References

[1] W. W. Bradbury and R. Fletcher, *New iterative methods for the solution of the eigenproblem*, Numer. Math., 9, pp. 259-267, 1966.

[2] Y. Cho and Y. K. Yong, *A multi-mesh, preconditioned conjugate gradient solver for eigenvalue problems in finite element models*, Comput. Struct., 58, pp.575-583, 1996.

[3] H. Elman, *Iterative methods for large, sparse, nonsymmetric systems of linear equations*, Ph. D Thesis, Yale University, 1982.

[4] Y. T. Feng and D. R. J. Owen, *Conjugate gradient methods for solving the smallest eigenpair of large symmetric eigenvalue problems*, Internat. J. Numer. Methods Engrg., 39, pp.2209-2229, 1996.

[5] G. Gambolati, G. Pini and M. Putti, *Nested iterations for symmetric eigenproblems*, SIAM J. Sci. Comput., 16, pp. 173-191, 1995.

[6] G. Gambolati, F. Sartoretto and P. Florian, *An orthogonal accelerated deflation technique for large symmetric eigenproblems*, Comput. Methods Appl. Mech. Engrg., 94, pp.13-23, 1992.

[7] D. E. Longsine and S. F. McCormick, *Simultaneous Rayleigh quotient minimization methods for $Ax = \lambda Bx$*, Linear Algebra Appl., 34, pp.195-234, 1980.

[8] Sangback Ma, *Comparisons of the parallel preconditioners on the CRAY-T3E for large nonsymmetric linear systems*, International Journal of High Speed Computing, 10, pp. 285-300, 1999.

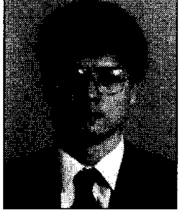
[9] B. N. Parlett, *The Symmetric Eigenvalue Problem, Prentice-Hall, Englewood Cliffs*, NJ, 1980.

[10] A. Ruhe, *Computation of eigenvalues and eigenvectors,* in Sparse Matrix Techniques, V. A. Baker, ed., Springer-Verlag, Berlin, pp.130-184, 1977.

[11] Y. Saad, *Highly parallel preconditioner for general sparse matrices,* in Recent Advances in Iterative Methods, IMA Volumes in Mathematics and its Applications, G. Golub, M. Luskin and A. Greenbaum, eds, Springer-Verlag, Berlin, Vol.60, pp.165-199, 1994.

[12] F. Sartoretto, G. Pini and G. Gambolati, Accelerated simultaneous iterations for large finite element eigenproblems, J. Comput. Phys., 81, pp.53-69, 1989.

[13] H. R. Schwarz, *Eigenvalue problems and preconditioning,* ISNM, 96, pp.191-208, 1991.



마 상 백

e-mail : sangback@cse.hanyang.ac.kr
1978년 서울대학교 계산통계학과(학사)
1978년~1983년 국방과학연구소 연구원
1987년 미국 미네소타 주립대학 수학과 석사
1993년 동대학 전자계산학과 박사
1994년~1996년 한양대학교 공학대학 전자계산학과 전임강사
1996년~2001년 한양대학교 전자컴퓨터 공학부 조교수
2001년~현재 한양대학교 전자컴퓨터 공학부 부교수
관심분야 : 수치해석, 암호학, 병렬처리