Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Sound and Vibration 285 (2005) 1242–1254
JOURNAL OF SOUND AND VIBRATION
www.elsevier.com/locate/jsvi

Short Communication

A parallel mixed finite element implementation for approximation of eigenvalues and eigenvectors of fourth-order eigenvalue problems

Kshitij Kulshreshtha[a], Neela Nataraj[b,*]

[a]*Institut für Mathematik, Humboldt Universität zu Berlin, Unter den Linden 6, D-10099 Berlin, Germany*
[b]*Department of Mathematics, Indian Institute of Technology Bombay, Powai, Mumbai 400076, India*

Received 2 March 2004; received in revised form 12 November 2004; accepted 17 November 2004
Available online 25 January 2005

Abstract

The paper deals with a parallel implementation of a mixed finite element method of approximation of eigenvalues and eigenvectors of fourth order eigenvalue problems with variable/constant coefficients. The implementation has been done in Silicon Graphics Origin 3800, a four processor Intel Xeon Symmetric Multiprocessor and a beowulf cluster of four Intel Pentium III PCs. The generalised eigenvalue problem obtained after discretization using the mixed finite element method is solved using the package LANSO. The numerical results obtained are compared with existing results (if available). The time, speedup comparisons in different environments for some examples of practical and research interest and importance are also given.

*Corresponding author.
E-mail addresses: kshitij@math.hu-berlin.de (K. Kulshreshtha), neela_nataraj@hotmail.com, neela@math.iitb.ac.in (N. Nataraj).

0022-460X/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jsv.2004.11.016

1. Introduction

Consider the following eigenvalue problem: find $\mu \in \mathbb{R}$, for which $\exists$ *non null* u such that

$$(P^E): \begin{cases} \Lambda u = \mu u & \text{in } \Omega, \\ u|_\Gamma = \left.\dfrac{\partial u}{\partial n}\right|_\Gamma = 0, \end{cases} \tag{1.1}$$

where the plate bending operator Λ is defined by

$$(\Lambda u)(x) = \sum_{i=1}^{2}\sum_{j=1}^{2}\sum_{k=1}^{2}\sum_{l=1}^{2} \frac{\partial^2}{\partial x_k \partial x_l}\left(a_{ijkl}\,\frac{\partial^2 u}{\partial x_i \partial x_j}\right)(x) \quad \forall x \in \Omega, \tag{1.2}$$

Ω being an open convex domain with boundary Γ.

In Ref. [1], a mixed method formulation for the general fourth-order vibration problem (1.1) has been developed with all details of convergence considering the combined effect of numerical integration and a boundary approximation using isoparametric mapping. This method can be applied to the vibration analysis of clamped biharmonic/isotropic/orthotropic/anisotropic plates. The mixed method scheme of Hellan–Hermann–Miyoshi for biharmonic eigenvalue problem [2,3] in a convex polygonal domain can be retrieved as a particular case with a proper choice of coefficients in the equation.

In Ref. [4], a parallel implementation of this very interesting mixed method has been discussed for the source problem for the operator Λ in Eq. (1.2). To our knowledge, this is probably the *first* attempt to study the parallelization of a mixed finite element method for solving general fourth-order elliptic problems in distributed memory environments.

In the case of eigenvalue problems, the discretization using this mixed finite element method leads to a generalized matrix eigenvalue problem. In this paper, a parallel implementation of the method of Ref. [1] has been done. Simplified procedures for computer implementations are developed and the parallelization has been done in SGI Origin 3800, a four processor Intel Xeon Symmetric Processor (SMP) and a beowulf cluster[1] of four Intel Pentium III PCs. Interesting results of numerical experiments on some examples are given with the runtime, speedup comparisons in different platforms.

2. Computer implementation procedure

Using the mixed finite element formulation and discretization as described in Refs. [1,4] we can form the element matrices $\mathbf{A}_T$ and $\mathbf{B}_T$, and the element mass matrix $\mathbf{M}_T$ can be written as follows

$$\mathbf{M}_T = [m_{qr}^T]_{1\leqslant q,r\leqslant 6} \quad \text{with } m_{qr}^T = \langle \chi_T^q, \chi_T^r\rangle_{0,T} = \int_T \chi_T^q \chi_T^r \,\mathrm{d}T, \tag{2.1}$$

with $\{\chi_T^i\}_{i=1}^6$ being the usual polynomial canonical basis functions of $\mathbb{C}^0$ Lagrange finite element associated with the nodes of a triangle T in the triangulation τ_h (see Refs. [1,4] for details).

[1]Installed in Department of Mathematics, IIT Bombay. Funding provided by Department of Science and Technology, Ministry of Defence and Technology, Government of India.

The mixed finite element eigenvalue problem can be reduced to a matrix form by assembling the element matrices $\mathbf{A}_T$, $\mathbf{B}_T$ and $\mathbf{M}_T$ for all triangles to global matrices $\mathbf{A}$, $\mathbf{B}$ and global mass matrix $\mathbf{M}$ using the standard finite element procedure. The essential boundary conditions and the symmetric conditions (depending on the domain considered) which are involved in the construction of the global matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{M}$ are introduced before the assembly.

The following generalized eigenvalue problem is formed:

Find $(\mu_h, \boldsymbol{\alpha}, \beta) \in \mathbb{R}^+ \times \mathbb{R}^{2M_1+M_2} \times \mathbb{R}^{M_0}$ such that

$$\begin{aligned} \mathbf{A}\boldsymbol{\alpha} + \mathbf{B}\beta &= 0, \\ \mathbf{B}^t\boldsymbol{\alpha} &= \mu_h \mathbf{M}\beta, \end{aligned} \tag{2.2}$$

where $M_0 = \dim\ W_h$ is the total number of nodes in the triangulation τ_h not lying on Γ, M_1 is the total number of nodes in the triangulation τ_h, $M_2 = M_1 -$ (total number of boundary nodes not lying on Γ), $\dim\ \mathbf{V}_h = 2M_1 + M_2$, $\Psi_h = \sum_{j=1}^{2M_1+M_2} \alpha_j \Phi_h^j$, and $u_h = \sum_{k=1}^{M_0} \beta_k \chi_h^k$, where $\Phi_h^j \in \mathbf{V}_h$, $1 \leqslant j \leqslant 2M_1 + M_2$; $\chi_h^k \in W_h$, $1 \leqslant k \leqslant M_0$ are the continuous global basis functions for the admissible spaces of the mixed finite element formulations.

The global matrix $\mathbf{A}$ will have the form

$$\mathbf{A} = \begin{bmatrix} [\mathbf{A}_{11}]_{M_1\times M_1} & [\mathbf{A}_{12}]_{M_1\times M_2} & [\mathbf{A}_{13}]_{M_1\times M_1} \\ [\mathbf{A}_{21}]_{M_2\times M_1} & [\mathbf{A}_{22}]_{M_2\times M_2} & [\mathbf{A}_{23}]_{M_2\times M_1} \\ [\mathbf{A}_{31}]_{M_1\times M_1} & [\mathbf{A}_{32}]_{M_1\times M_2} & [\mathbf{A}_{33}]_{M_1\times M_1} \end{bmatrix}. \tag{2.3}$$

$[\mathbf{B}]^t_{M_0\times(2M_1+M_2)}$ is the transpose of $[\mathbf{B}]_{(2M_1+M_2)\times M_0}$, which will have the form

$$\mathbf{B} = \begin{bmatrix} [\mathbf{B}_1]_{M_1\times M_0} \\ [\mathbf{B}_2]_{M_2\times M_0} \\ [\mathbf{B}_3]_{M_1\times M_0} \end{bmatrix}. \tag{2.4}$$

(See Refs. [1,4] for details.)

2.1. Solution approach

From the first equation in Eq. (2.2), $\boldsymbol{\alpha}$ is eliminated and is substituted in the second equation to obtain

$$\mathbf{K}\beta = \mu_h \mathbf{M}\beta, \tag{2.5}$$

where $[\mathbf{K}]_{M_0\times M_0}$ is the symmetric positive definite global stiffness matrix and $[\mathbf{M}]_{M_0\times M_0}$ is the global mass matrix.

The global stiffness matrix $\mathbf{K}$ is computed as the Schur Complement matrix $-\mathbf{B}^t\mathbf{A}^{-1}\mathbf{B}$ and in the case of constant coefficients a_{ijkl}, the explicit formulation of $\mathbf{K}$ is given in Appendix B of Ref. [4].

The generalized eigenvalue problem (2.5) is solved using the Lanczos method [5].

2.2. Parallelization strategy

The following methodology has been adopted for parallelization.
Triangulation:

- The domain $\overline{\Omega}$ is decomposed into subdomains with the number of subdomains being equal to the number of processors.
- The triangulation of the subdomains is done in parallel in the different processors. The global triangulation of $\overline{\Omega}$ induces triangulations in all the subdomains in such a way that whenever two subdomains share a common interface, the boundary triangles (triangles which have edges on the interface) in these subdomains share the same edges. This means that the interface nodes are common between the neighbouring subdomains and there are no triangles that lie in more than one subdomain. Each subdomain has a local numbering of the nodes along with the global numbering.

Computation of element matrices: The computation of the element matrices $\mathbf{A}_T$, $\mathbf{B}_T$ and the element mass matrix $\mathbf{M}_T$ is done in parallel in different subdomains, i.e. in this simple SIMD process, each processor computes the element matrices for all those triangles that lie in the respective subdomain allocated to it. This utilizes the distributed memory architecture effectively, because there is no need for communication between two processors as each of the triangles lies in only one processor.

Global assembly: The element matrices are assembled to the global matrices $\mathbf{A}$, $\mathbf{B}$, and the global mass matrix $\mathbf{M}$ in parallel as follows:

- Initially, a local assembly of the matrices is done in parallel within the subdomains for all the nodes. The interior nodes do not require any data from neighbouring subdomains, but the interface nodes do.
- A data exchange of the locally assembled values for the nodes on the interface is then done in order to accumulate values assembled in two neighbouring subdomains.

Solver: Once the assembly process is over, we compute the Schur complement matrix $\mathbf{K}$ using a parallel LU factorization of the matrices $\mathbf{A}_{ii}$, $1 \leqslant i \leqslant 3$ [4].

- The matrices $\mathbf{A}_{ii}$, $1 \leqslant i \leqslant 3$ are stored in a skyline data structure and each processor has a copy of it. During factorization, each processor is symbolically assigned some columns in the skyline structure in a wraparound fashion so that column $j \equiv i \bmod N_{\mathrm{PROCS}}$ goes to processor i, $1 \leqslant i \leqslant N_{\mathrm{PROCS}}$ (N_{PROCS} is the number of processors used). At each step k of the factorization, every processor factorizes the elements in the kth row in the columns assigned to it and the updated values are exchanged among all processors. These are then put in their respective correct positions in the matrix in each processor.
- Once $\mathbf{A}_{ii}$ are factorized, the Schur complement is computed by back and forward substitution followed by matrix vector multiplication.

We find the extreme eigenvalues of the generalized eigenvalue problem $\mathbf{K}\beta = \mu_h \mathbf{M}\beta$ using the LANSO package [5], which uses a version of the Lanczos Method for symmetric generalized eigenvalue problems.

3. Numerical experiments

For the plate bending operator Λ, the eigenvalue problem (1.1) is obtained from the equation of motion for the small transverse displacement U of the vibrating plate under consideration

$$\Lambda U + \rho \frac{\partial^2 U}{\partial t^2} = 0 \quad \forall((x_1, x_2); t) \in \Omega \times (0, T] \tag{3.1}$$

with $U|_\Gamma = 0$, $\partial U/\partial n|_\Gamma = 0\ \forall t \in]0, T]$, ρ being the mass density of the elastic plate per unit area measure of $\bar{\Omega}$, when free natural vibrations are assumed and the motion is defined by

$$U(x_1, x_2; t) = u(x_1, x_2) \cos \omega t \tag{3.2}$$

with ω being the circular frequency expressed in radians/unit time, i.e. a substitution of Eq. (3.2) into Eq. (3.1) will yield Eq. (1.1) with $\mu = \rho\omega^2$.

In the examples considered below, dimensionless coordinates are introduced and instead of $\mu = \rho\omega^2$, some new parameter of convenience, which will depend on ρ, ω, characteristic plate size parameter, flexural rigidity of the plate, etc. will be introduced and *will be still denoted by the same notation* μ by giving its new definition without deduction.

- In the following examples on eigenvalue problems, we consider domains Ω with polygonal/curved boundary Γ. In case the domain has a polygonal boundary Γ, no approximation of boundary is involved. For a domain with a curved boundary Γ, approximations of Γ by a polygonal boundary Γ_h and a curved boundary $\tilde{\Gamma}_h$ (constructed with the help of an isoparametric mapping) are considered.
- N_T denotes the number of triangles in the triangulation of $\bar{\Omega}$.
- N_{PROCS} denotes the number of processors used in the computations.
- Data and results are to be understood in proper units of measurement.
- Numerical results obtained have been compared with existing results, if available.

3.1. Example 3.1. Isotropic parallelogram plate vibration problem

The eigenvalue problem (1.1) with $\mu^2 = \omega a^2 \sqrt{\rho/D} \cos^2 \theta$ corresponding to the natural free vibrations of clamped isotropic parallelogram plates with aspect ratio a/b and skew angle θ (in degrees) (see Fig. 1) is considered. The fundamental eigenvalue μ_{1h} is compared with the results obtained in Ref. [6] in Table 1.

The runtime in minutes in different machines is given in Table 2. The time lines for this computation as obtained using the VAMPIR parallel visualization tool [7,8] in SGI Origin 3800 are given in Fig. 2 with $N_{\text{PROCS}} = 16$.

3.2. Example 3.2. Orthotropic square plate vibration problem

In this example, a special kind of orthotropic coefficient [9] has been chosen. The principal directions of orthotropy coincide with the coordinate axes of the square plate. Only three flexural rigidity parameters are required to characterize such plates. To carry out the free vibration analysis, only two parameters involving ratios of the three parameters are required.

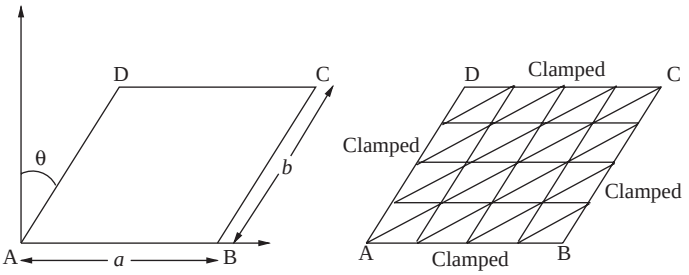

Fig. 1. A clamped isotropic parallelogram plate (Example 3.1).

Table 1
Fundamental eigenvalue μ_{1h} corresponding to the natural free vibration of clamped isotropic parallelogram plate with skew angle θ and aspect ratio a/b, $\mu^2 = \omega a^2\sqrt{\rho/D}\cos^2\theta$, $N_T = 128$ (Example 3.1)

θ in degrees	Aspect ratio a/b				
	1			0.5	
	μ_{1h}	[6]	[6][a]	μ_{1h}	[6]
15	35.6680	35.636	35.625	24.4921	24.484
20	35.3862	35.376	—	24.4032	24.388
30	34.5938	34.624	34.788	24.1630	24.196
45	32.8491	—	32.795	23.6805	—
60	30.4596	—	30.823	23.1327	—

[a]Results from M. Hamada and M. Hasegawa as given in Ref. [6].

Table 2
Runtime in minutes for clamped isotropic parallelogram plate with $N_{\text{PROCS}} = 4$ (Example 3.1)

N_T	SGI Origin 3800	Intel Xeon SMP	Beowulf cluster
8	0:00.243	0:00.19	0:03.78
32	0:00.571	0:00.46	0:05.04
128	0:05.237	0:05.51	1:33.78

The flexural rigidities for the orthotropic case are

$$D_i = \frac{E_i t^3}{12}, \quad D_\tau = \frac{Gt^3}{12} > 0, \quad \left(G = \frac{E_1 E_2}{E_1 + E_2} > 0\right), \quad H = 2D_\tau \quad (i = 1, 2).$$

Here E_i $(i = 1, 2)$ are the Young's moduli, G is the shear modulus, $t = t(x_1, x_2)$ is the thickness of the plate.

We consider the cases where the flexural rigidities are given by

(i) $dhx = 1.0,\ dhy = 1.5;$ (ii) $dhx = 1.5,\ dhy = 1.5;$ (iii) $dhx = 1.5,\ dhy = 2.0.$

Here $dhx = H/D_1,\ dhy = H/D_2.$

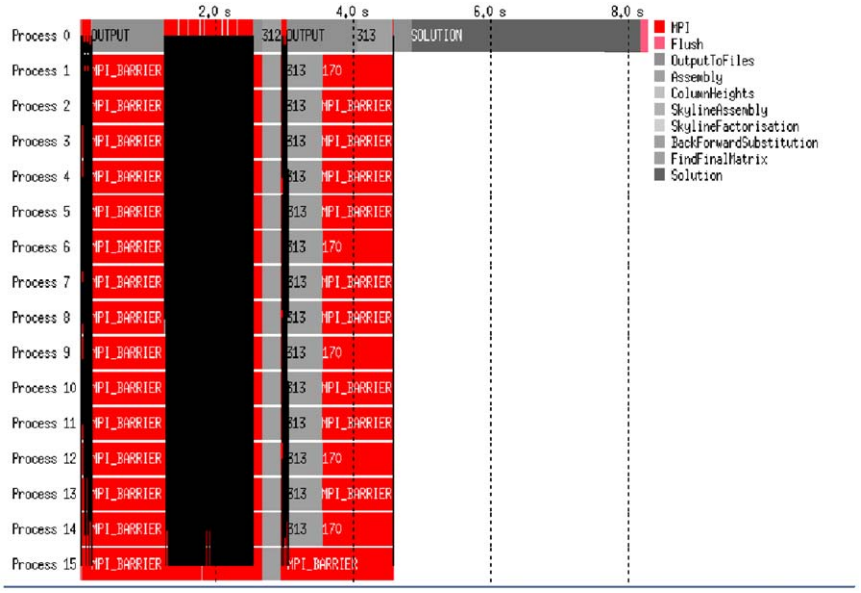

Fig. 2. VAMPIR timeline for the parallel processes for computing the eigenvalues of the clamped isotropic parallelogram plate vibration problem with $N_{\mathrm{PROCS}} = 16$ and $N_T = 128$ (Example 3.1). The figure shows the time spent by the 16 processes in different activities/subroutines like assembly of matrices, factorization in skyline structure, back and forward substitution to compute the schur complement and communication involved, etc. The dark area shows the communication between the processes.

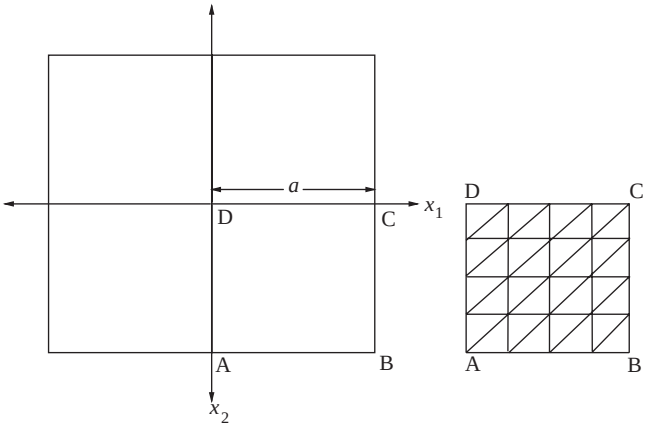

Fig. 3. A clamped orthotropic square plate. ABCD is the reference quarter plate considered for computation (Example 3.2).

The eigenvalue problem (1.1) with $\mu^2 = \omega a^2\sqrt{\rho/D_1}$ corresponding to the natural vibrations of clamped orthotropic square plate with side $2a$ (see Fig. 3) is considered. The first few eigenvalues are compared with those given in Ref. [9] in Tables 3–5. The number of processors vs. runtimes and speedup graph when $N_T = 128$ for the problem run in SGI Origin 3800 is shown in Fig. 4.

Table 3
Eigenvalues $(\mu_{ih})_{i=1}^{4}$ corresponding to the natural free vibration of clamped orthotropic square plate with side $2a$, $\mu^2 = \omega a^2\sqrt{\rho/D_1}$, $N_T = 128$, $dhx = 1.0$, $dhy = 1.5$ (Example 3.2)

Eigenvalues	Computed values	[9]	Nature of mode
μ_{1h}	8.3926	8.39	Doubly symmetric
μ_{2h}	18.0506	18.02	Symmetric–antisymmetric
μ_{3h}	25.5664	25.52	Doubly antisymmetric
μ_{4h}	28.0039	27.97	Second doubly symmetric

Table 4
Eigenvalues $(\mu_{ih})_{i=1}^{4}$ corresponding to the natural free vibration of clamped orthotropic square plate with side $2a$, $\mu^2 = \omega a^2\sqrt{\rho/D_1}$, $N_T = 128$, $dhx = 1.5$, $dhy = 1.5$ (Example 3.2)

Eigenvalues	Computed values	[9]	Nature of mode
μ_{1h}	9.4787	9.476	Doubly symmetric
μ_{2h}	19.2241	19.21	Symmetric–antisymmetric
μ_{3h}	29.3094	29.25	Doubly antisymmetric
μ_{4h}	33.9343	33.90	Second doubly symmetric

Table 5
Eigenvalues $(\mu_{ih})_{i=1}^{4}$ corresponding to the natural free vibration of clamped orthotropic square plate with side $2a$, $\mu^2 = \omega a^2\sqrt{\rho/D_1}$, $N_T = 128$, $dhx = 1.5$, $dhy = 2.0$ (Example 3.2)

Eigenvalues	Computed values	[9]	Nature of mode
μ_{1h}	9.0489	9.046	Doubly symmetric
μ_{2h}	18.9938	18.98	Symmetric–antisymmetric
μ_{3h}	28.2403	28.18	Doubly antisymmetric
μ_{4h}	30.4700	30.44	Second doubly symmetric

3.3. Example 3.3. Isotropic elliptic plate vibration problem

Let Γ_h (resp. $\widetilde{\Gamma}_h$) be a symmetric polygonal (resp. a symmetric isoparametric) approximation to Γ (see Fig. 5). The flexural rigidity for the isotropic case is $D = Et^3/12(1-v^2)$ and Poisson's coefficient $v = 0.3$. $\mu^2 = \omega b^2\sqrt{\rho/D}$ corresponding to the natural vibrations of clamped isotropic elliptic plate with semi-major axis a and semi-minor axis b has been computed.

The first three eigenvalues $(\mu_{ih})_{i=1}^{3}$ (resp. $(\tilde{\mu}_{ih})_{i=1}^{3}$) corresponding to affine (resp. isoparametric) approximation to Γ symmetric with respect to x_1 and x_2 axes for $b/a = 1, 0.8, 0.5, 0.2$ have been tabulated in Table 6.

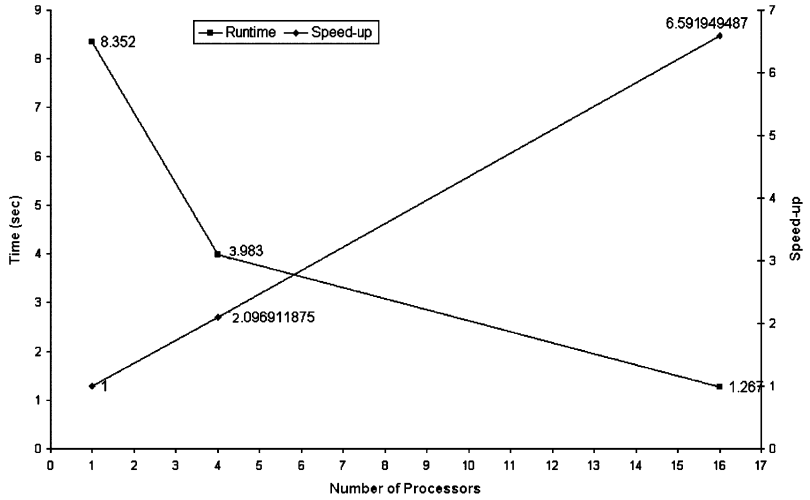

Fig. 4. Number of processors vs. runtime and speedup graph for clamped square orthotropic plate with $N_T = 128$ in SGI Origin 3800 (Example 3.2).

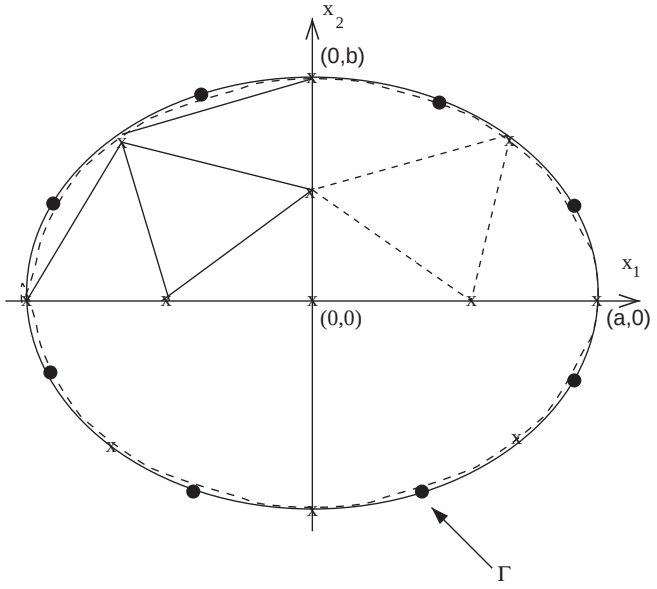

Fig. 5. A clamped isotropic elliptic plate. Isoparametric and affine triangulations are shown in quadrants I and II, respectively (Example 3.3).

A speedup of 5.496 has been achieved in the SGI Origin 3800 machine when $N_T = 64$ and $N_{\text{PROCS}} = 16$.

A comparison of the runtimes in minutes for this problem computed in different platforms, SGI Origin 3800, Intel Xeon SMP and beowulf cluster, for the affine and isoparametric cases when $N_T = 64$ is given in Fig. 6.

Table 6
First few eigenvalues corresponding to the doubly symmetric mode vibrations of a clamped elliptic isotropic plate with semi-major axis a and semi-minor axis b with $\mu^2 = \omega b^2\sqrt{\rho/D}$, $N_T = 64$ (Example 3.3)

$\frac{b}{a}$	μ_{1h}	$\widetilde{\mu}_{1h}$	[6]	μ_{2h}	$\widetilde{\mu}_{2h}$	μ_{3h}	$\widetilde{\mu}_{3h}$
1	10.2909	10.2183	10.216	35.1689	35.9158	40.1193	39.8266
0.8	8.5283	8.4686	8.467	25.8102	25.6170	35.9968	35.7431
0.5	6.8926	6.8462	6.845	14.1341	14.0251	26.0785	25.8500
0.2	6.0283	5.9890	5.996	7.9414	7.8872	10.5421	10.4673

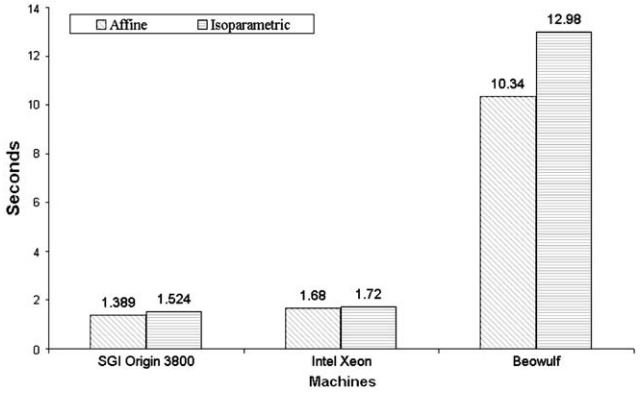

Fig. 6. Runtime comparison for affine and isoparametric elements in SGI Origin 3800, Intel Xeon and beowulf cluster for clamped elliptic isotropic plate with $N_T = 64$ (Example 3.3).

3.4. Example 3.4. Orthotropic circular plate vibration problem

The coefficients for the circular orthotropic case are defined through the flexural rigidities: $D_{22}/D_{11} = 16$, $H/\sqrt{D_{11}D_{22}} = 1.5$, and $v_2 = 0.3$ [10]. $\mu^2 = \omega a^2\sqrt{\rho/D}$ corresponding to the natural vibrations of clamped orthotropic circular plate with radius a (see Fig. 7) for polygonal and isoparametric boundary approximation have been computed and the results have been tabulated in Table 7.

A speedup of 4.956 and an efficiency of 0.309 has been achieved in SGI Origin 3800 machine when $N_T = 64$ and $N_{\text{PROCS}} = 16$.

The runtime comparison in different machines, SGI Origin 3800, Intel Xeon SMP and beowulf cluster, for the affine and isoparametric cases when $N_{\text{PROCS}} = 4$ and $N_T = 4, 16, 64$ is given in Table 8.

4. Conclusions

4.1. About the machines

The performance of the algorithm in the Intel Xeon SMP machine is better than that of the beowulf cluster, as in the cluster, the network speed is the bottle neck. Ethernet-based beowulf is

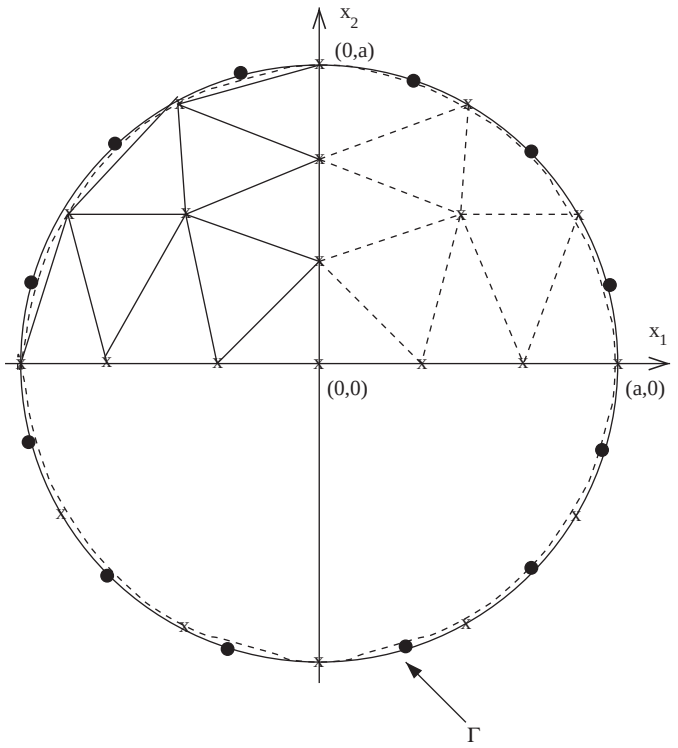

Fig. 7. A clamped orthotropic circular plate. Isoparametric and affine triangulation are shown in quadrants I and II, respectively (Example 3.4).

Table 7
Eigenvalues $(\mu_{ih})_{i=1}^{4}$ corresponding to the natural free vibration of clamped orthotropic circular plate with radius a, $\mu^2 = \omega a^2\sqrt{\rho/D_{11}}$, $N_T = 64$ (Example 3.4)

Eigenvalues	Affine	Isop.	Nature of mode	Nodal pattern
μ_{1h}	10.1141	10.0256	Doubly symmetric	
μ_{2h}	15.3093	14.6800	Symmetric–antisymmetric	
μ_{3h}	21.6437	21.4525	Second doubly symmetric	
μ_{4h}	29.4785	28.7209	Second symmetric antisymmetric	

Table 8
Runtime in minutes for clamped circular orthotropic plate of radius a in different platforms with four processors (Example 3.4)

N_T	SGI Origin 3800		Intel Xeon SMP		Beowulf cluster	
	Affine	Isop.	Affine	Isop.	Affine	Isop.
4	0:00.431	0:00.446	0:00.32	0:00.33	0:02.47	0:02.56
16	0:00.496	0:00.548	0:00.34	0:00.39	0:02.59	0:03.02
64	0:01.382	0:01.551	0:01.69	0:01.72	0:10.33	0:12.74

attractive as it provides a very cheap parallel processing platform. A MYRINET/Fibre optic network-based beowulf cluster can be a compromise between a large processor array like SGI Origin 3800 and an Ethernet-based beowulf cluster.

4.2. About the method

The results of the numerical experiments establish the fact that the mixed finite element method gives satisfactory results for all the problems considered. Also, the quality of solution obtained by the mixed finite element approximation for domains with curved boundary has been improved by a better boundary approximation using isoparametric mapping without much change in the computational time as compared to the polygonal approximation of the boundary.

4.3. Limitations

The memory usage in the parallel implementation of the mixed finite element eigenvalue problem for fourth-order elliptic problems has a trade off with the communication needed. Hence, to minimize the communication, a large amount of memory is used. Domain decomposition algorithms for general fourth-order elliptic eigenvalue problems are yet to be explored.

Acknowledgements

We would like to thank Prof. Wolfgang E. Nagel, Zentrum für Hochleistungsrechnen, Technische Universität Dresden, Germany for providing the computing resources of SGI Origin 3800. We would also like to thank the Department of Science and Technology, Ministry of Defence and Technology, Government of India for providing the funds to build a beowulf cluster. This research was conducted within the scientific cooperation supported by the Max Plank research prize for international cooperation awarded in 2001 to Prof. Andreas Griewank of Technische Universität Dresden (now at Humboldt Universität zu Berlin).

References

[1] P.K. Bhattacharyya, N. Nataraj, Isoparametric mixed finite element approximation of eigenvalues and eigenvectors of fourth order eigenvalue problems with variable coefficients, *Modélisation Mathématique et Analyse Numérique* 36 (1) (2002) 1–32.
[2] C. Canuto, Eigenvalue approximation by mixed methods, *R.A.I.R.O. Anal. Numer.* R3 12 (1978) 27–50.
[3] B. Mercier, J. Osborn, J. Rappaz, P. Raviart, Eigenvalue approximation by mixed and hybrid methods, *Mathematics of Computation* 36 (154) (1981) 427–453.
[4] K. Kulshreshtha, N. Nataraj, M. Jung, Performance of a parallel mixed finite element implementation for fourth order clamped anisotropic plate bending problems in distributed memory environments, *Applied Mathematics and Computation* 155 (3) (2004) 753–777.
[5] B.N. Parlett, D. Scott, The lanczos algorithm with selective orthogonalisation, *Mathematics of Computation* 33 (1979) 217–238.
[6] A.W. Leissa, *Vibration of Plates*, NASA SP-160, 1969.
[7] W.E. Nagel, A. Arnold, M. Weber, H.-C. Hoppe, K. Solchenbach, Vampir: visualization and analysis of mpi resources, *Supercomputer* 63 (12) (1996) 69–80.
[8] H. Brunst, H.-C. Hoppe, W.E. Nagel, M. Winkler, Performance optimization for large scale computing: the scalable vampir approach, in: *Proceedings of ICCS2001*, Springer, Berlin, 2001.
[9] D.J. Gorman, Accurate free vibration analysis of clamped orthotropic plates by the method of superposition, *Journal of Sound and Vibration* 140 (3) (1990) 319–411.
[10] S.G. Leknitskii, *Anisotropic Plates*, Gordon and Breach Science, New York, 1968.