PARALLEL EIGENVALUE EXTRACTION

Fred A. Akl
Ohio University

Abstract

This report presents a new numerical algorithm for the solution of large-order eigenproblems typically encountered in linear elastic finite element systems. The architecture of parallel processing is utilized in the algorithm to achieve increased speed and efficiency of calculations. The algorithm is based on the frontal technique for the solution of linear simultaneous equations and the modified subspace eigenanalysis method for the solution of the eigenproblem. Assembly, elimination and back-substitution of degrees of freedom are performed concurrently, using a number of fronts. All fronts converge to and diverge from a predefined global front during elimination and back-substitution, respectively. In the meantime, reduction of the stiffness and mass matrices required by the modified subspace method can be completed during the convergence/divergence cycle and an estimate of the required eigenpairs obtained. Successive cycles of convergence and divergence are repeated until the desired accuracy of calculations is achieved. The advantages of this new algorithm in parallel computer architecture are discussed.

Generalized Eigenproblem

$$[K][\Phi] = [M][\Phi][\Omega]$$

N - degrees of freedom

Required n eigenpairs, $n \leq N$

[K] positive-definite

- New parallel algorithm for the solution of large scale eigenproblems in finite element applications.
- Work is in progress to implement algorithm on NAS Cray 2 computer at Ames.

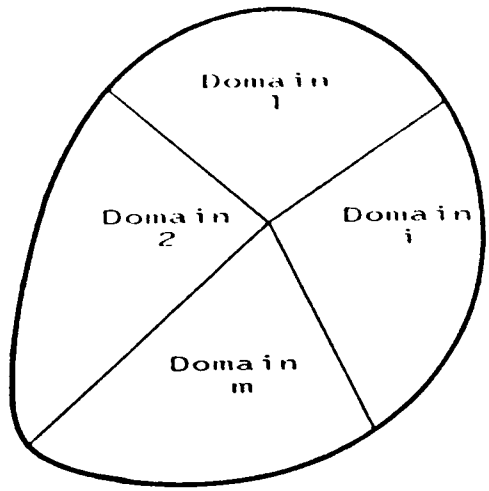
- Assumptions

  1 - Linear elastic finite element models

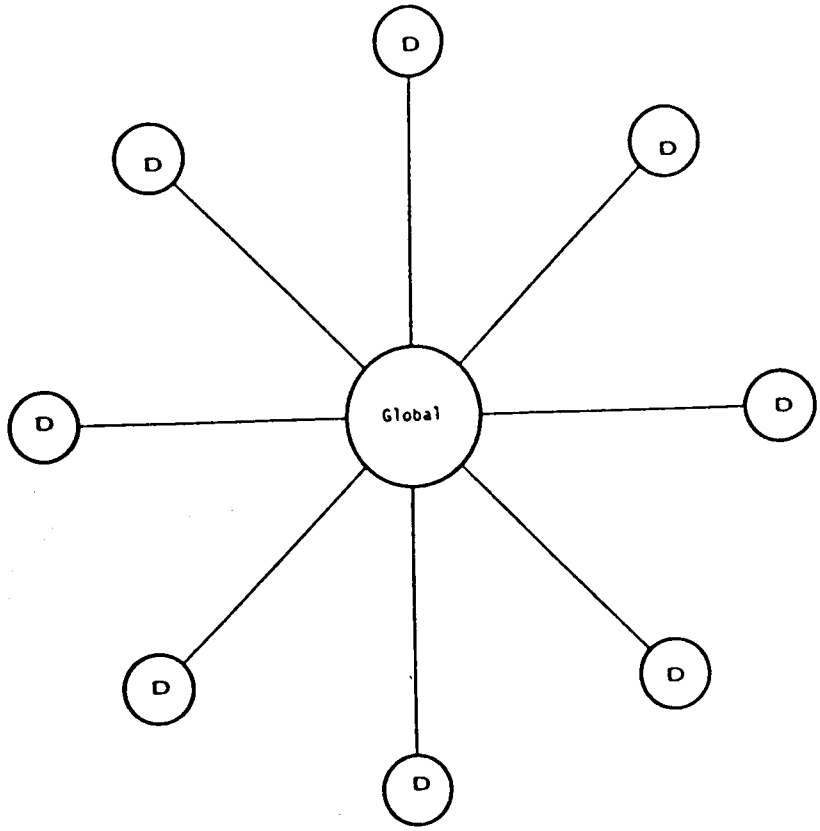
  2 - n lower order eigenpairs are required, i.e. $\omega_1^2 \leq \omega_2^2 \leq \ldots \omega_n^2$

  3 - [K] is positive-definite

  4 - [M] is semi-positive definite



Finite Element Model Subdivided into m Domains



Domain i

- Consider a parallel computer with (m+1) processors (tasks).
- Designate the first processor as a global processor (task).
- Designate the remaining m-processors as domain processors (tasks).
- A finite element model can be divided into a number of domains equal to m.
- A star architecture (or tree) is the first to be investigated.

$$[K][\phi] = [M][\phi][\Omega]$$

1 - Creation of K^e & M^e

2 - Eigensolution (Modified Subspace)

3 - Equation Solver (Frontal Solution)

- Three major steps of large computational requirements:

  1 - Creation of element stiffness and mass matrices.

  2 - Extraction of a set of eigenpairs.

  3 - Solution of a set simultaneous linear equations.

- The merits of selecting the modified subspace method for step #2 and the frontal solution for step #3 above all discussed in the next new graphs.

Modified Subspace Method

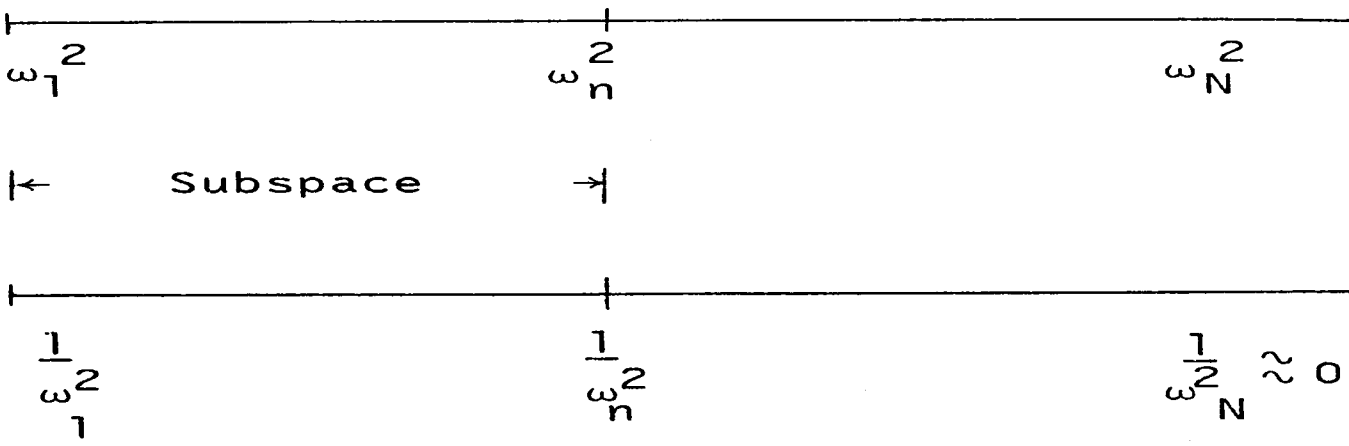
$$[V]^{*}_{\ell+1} = ([K]^{-1}[M] - \beta_{\ell}[I])[V]_{\ell}$$

$$= [K]^{-1}[B]_{\ell} - \beta_{\ell}[V]_{\ell}$$

where $\ell = 1, 2, 3, \ldots$

$$\beta_{\ell} = 1/2(1+r_{\ell-1})/\omega_n^2$$

$$\omega_1^2 \leq \ldots \leq \omega_n^2 \leq \omega_{n+1}^2 \leq \ldots \leq \omega_N^2$$



The Modified Subspace method iterates simultaneously for a subset of eigenpairs $[\phi,\omega]$ of the generalized eigenproblem:

1 - Let $[V]_1$ be n starting eigenvectors. Experience has shown that random numbers can be used here. A number of techniques are available in literature for selecting $[V]_1$.

2 - Operate on each $[V]_\ell$ as follows

$$[V]^*_{\ell+1} = [K]^{-1}[M][V]_\ell = [K]^{-1}[B]_\ell$$

where $\ell = 1, 2, 3, \ldots$

3 - Modify $[V]^*_{\ell+1}$ to increase convergence rate by one third on average

$$[V]^*_{\ell+1} \leftarrow V^*_{\ell+1} - \beta_\ell V_\ell$$

where: $\beta_\ell = 0$ for $\ell=1$ and $\ell>11$

$$\beta_\ell = 0.5\,(1+r_{\ell-1})/\omega_n^2 \quad 1\leq\ell\leq 11$$

$r_{\ell-1}$ are the interval points of the 11-th order Labatto rule $[-1, 1]$

Roots of the 11th Order Lobatto Rule (Kopal 1961)

r_1	-0.9533098466	r_6	0.0000000000
r_2	-0.8463475646	r_7	+0.2492869301
r_3	-0.6861884690	r_8	+0.4829098210
r_4	-0.4829298210	r_9	+0.6861884690
r_5	-0.2492869301	r_{10}	+0.8463475646

Subspace

$$[K]^{*}_{\ell+1} = \Sigma[V]^{eT}_{\ell+1}[K]^{e}[V]^{e}_{\ell+1}$$

$$[M]^{*}_{\ell+1} = \Sigma[V]^{eT}_{\ell+1}[M]^{e}[V]^{e}_{\ell+1}$$

The Auxiliary Eigenproblem

$$[K]^{*}_{\ell+1}\ [Q]_{\ell+1} = [M]^{*}_{\ell+1}\ [Q]_{\ell+1}\ [\Omega]_{\ell+1}$$

Improved Eigenvectors

$$[V]^{e}_{\ell+1} = [V]^{*e}_{\ell+1}\ [Q]_{\ell+1}$$

4 - Project K and M onto the required subspace.

5 - Solve the auxiliary eigenproblem to obtain $[Q]_{\ell+1}$ and $[\Omega]_{\ell+1}$.

6 - An improved set of eigenvectors $[V]_{\ell+1}$ can be obtained.

7 - Test for convergence on ω_n^2. Repeat steps 2 to 6 until desired accuracy is achieved.

Note

1. Step #2 is performed using the frontal solution, concurrently within each domain.

2. Steps 1, 3, 4 and 6 are processed concurrently within each domain.

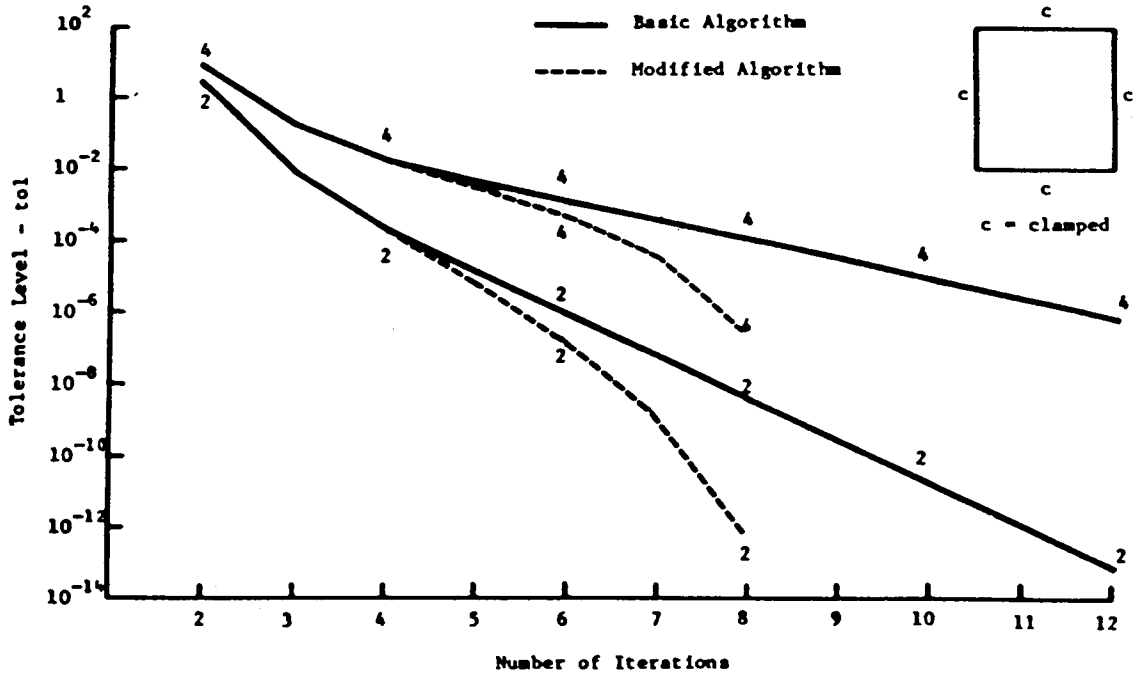
RATE OF CONVERGENCE

- **Basic Subspace**

$$\leq \frac{\omega_i^2}{\omega_{n+1}^2}$$

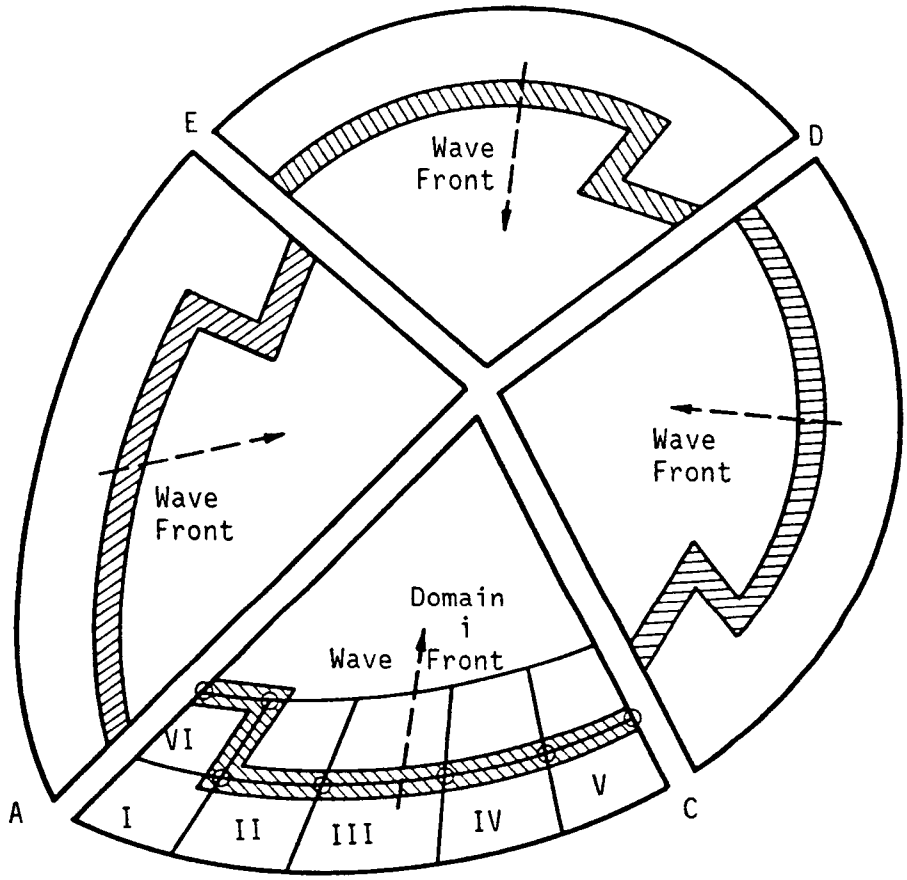
- **Modified Subspace**

$$\leq \frac{\omega_i^2}{\omega_{n+1}^2} \quad \frac{1 - \beta_\ell \omega_{n+1}^2}{1 - \beta_\ell \omega_i^2}$$



Convergence of λ_2 and λ_4 for Square Plate

- Rate of convergence of the modified subspace is 33% faster on average compared to the classical subspace method.
- Figure shows typical behavior.
- Most computations are performed on an element by element basis.



Multi-Frontal Parallel Processing

Frontal Solution

1 - Gauss elimination technique.

2 - Underlying philosophy is based on processing of elements one by one.

3 - Simultaneous assembly and elimination of variables.

4 - The optimum frontal width is at most equal to the optimum band width.

5 - Numbering of nodes has no impact on optimality while numbering of elements is important to minimize the frontal width.

6 - More efficient for solid elements and elements with mid-side nodes.

7 - It requires a pre-front to determine last appearance of each node.

8 - It lends itself to parallel solutions.

Multi-Frontal Solution

- Within each domain

$$\hat{k}^d_{ij} \leftarrow k^d_{ij} - \sum^d \left[\frac{k_{is} k_{sj}}{k_{ss}} \right]$$

$$\hat{b}^d_{iq} \leftarrow b^d_{iq} - \sum^d \left[\frac{k_{is} b_{sq}}{k_{ss}} \right]$$

For domain i

$$[K][V]^{*}_{\ell+1} = [B]_{\ell+1}$$

Assembly and elimination gives

$$U_i V^{*}_d + \hat{K}_{dF} V^{*}_F = \hat{B}_d$$

$$\hat{K}_{FF} V^{*}_F = \hat{B}_F$$

where U_i upper Δ matrix for domain i

V^{*}_d variables within domain i

V^{*}_F variables along global front of domain i

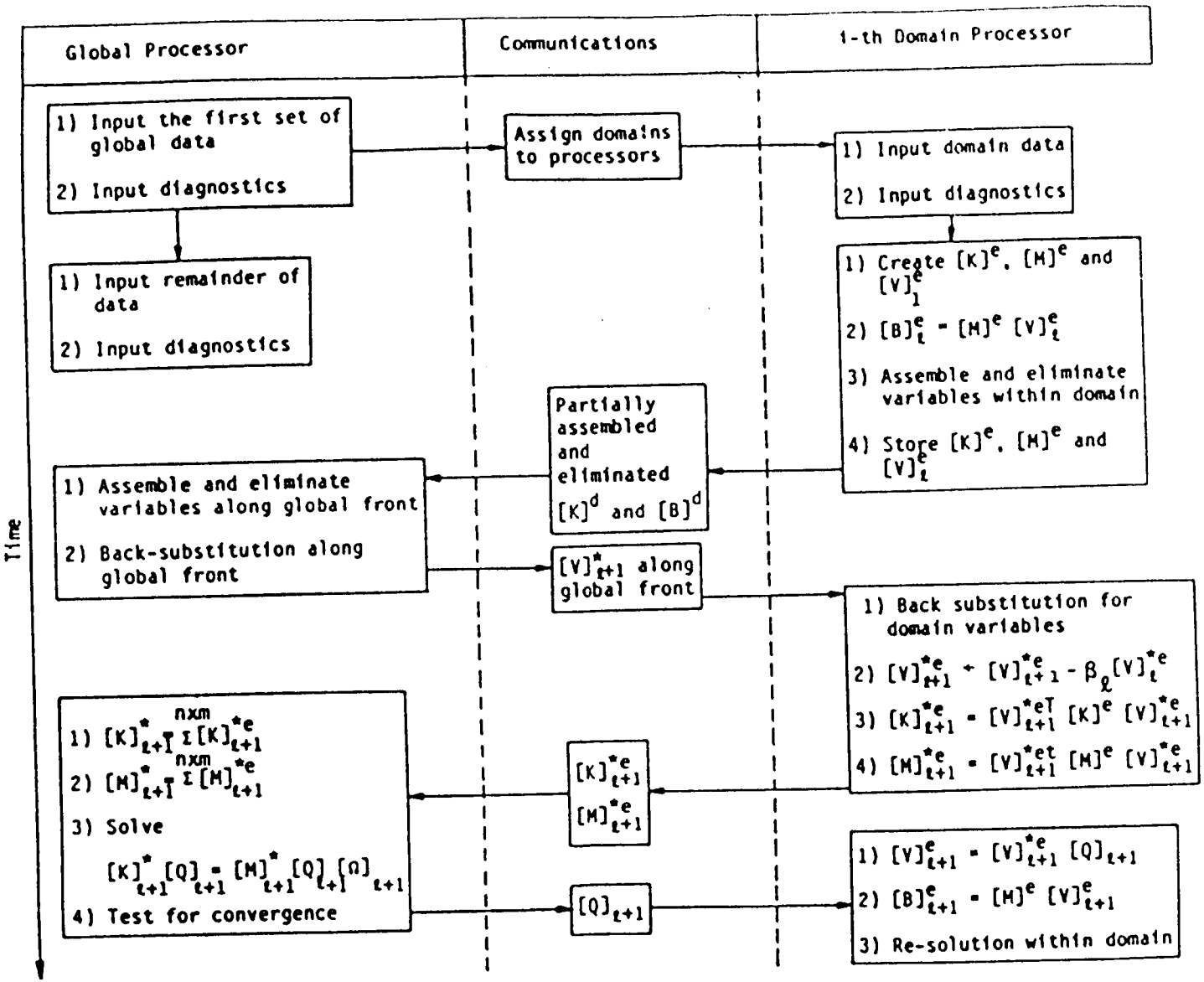
$\hat{B}_d$ & $\hat{B}_F$ are right-hand sides for domain & global front, respectively

For global fronts

$$\hat{K} = \sum^{m} K_{FF}$$

$$\hat{B} = \sum^{m} B_F$$

$$\hat{K}\ \hat{V}_F = \hat{B}_F$$



Computational Tasks and Communications

- Successful implementation of the new parallel algorithm depends on:

  1 - Maximizing the efficiency of communication links between the global task and the domains

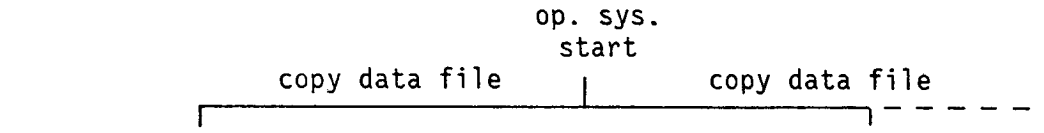
  2 - Minimizing sequential computational steps

  3 - Multi-threaded I/O

- Final report will be available in the Summer 1988

Anticipated Benefits

- Parallel eigenvalue extraction algorithm to maximize efficiency and speed-up of computations.
- A general purpose eigenproblem solver for finite element analysis in parallel computing environment.



GLBFRONT	DOMFRONT
1. read/check first data card	1. read/check first data card
2. set-up VEC for data input	2. set-up VEC for data input
3. data input and check	3. data input and synthesis
4. reset VEC for global fronts	4. reset VEC for domain
5. pre-front for global fronts	5. pre-front for domain
	5. element K and M matrices
	6. domain assembly/elimination
	7. K_{FF} to GLBFRONT
6. global fronts solution	
7. V_F to DOMFRONT	
	8. domain solution and subspace
	9. K_d^* and M_d^* to DOMFRONT
8. subspace solution	
9. Q to DOMFRONT	
10. convergence test	10. Improved eigenvectors V^e