SOLUTION METHODS FOR CALCULATIONS OF FREQUENCIES AND MODE SHAPES

LECTURE 12

58 MINUTES

LECTURE 12 Solution methods for finite element eigenproblems

Standard and generalized eigenproblems

Basic concepts of vector iteration methods, polynomial iteration techniques, Sturm sequence methods, transformation methods

Large eigenproblems

Details of the determinant search and subspace iteration methods

Selection of appropriate technique, practical considerations

TEXTBOOK: Sections: 12.1, 12.2.1, 12.2.2, 12.2.3, 12.3.1, 12.3.2, 12.3.3, 12.3.4, 12.3.6 (the material in Chapter 11 is also referred to)

Examples: 12.1, 12.2, 12.3, 12.4

SOLUTION METHODS FOR EIGENPROBLEMS

Standard EVP:

$$\underset{n\times n}{\underline{K}}\ \underline{\phi} = \lambda\ \underline{\phi}$$

Generalized EVP:

$$\underline{K}\ \underline{\phi} = \lambda\ \underline{M}\ \underline{\phi} \quad \longleftarrow \quad (\lambda = \omega^2)$$

Quadratic EVP:

$$(\underline{K} + \lambda\ \underline{C} + \lambda^2\ \underline{M})\underline{\phi} = \underline{0}$$

Most emphasis on the generalized EVP e.g. earthquake engineering

"Large EVP" $n > 500$ $p = 1, \ldots, \frac{1}{3} n$
$m > 60$

In dynamic analysis, proportional damping

$$\underline{K}\ \underline{\phi} = \omega^2\ \underline{M}\ \underline{\phi}$$

If zero freq. are present we can use the following procedure

$$\underline{K}\ \underline{\phi} + \mu\ \underline{M}\ \underline{\phi} = (\omega^2 + \mu)\underline{M}\ \underline{\phi}$$

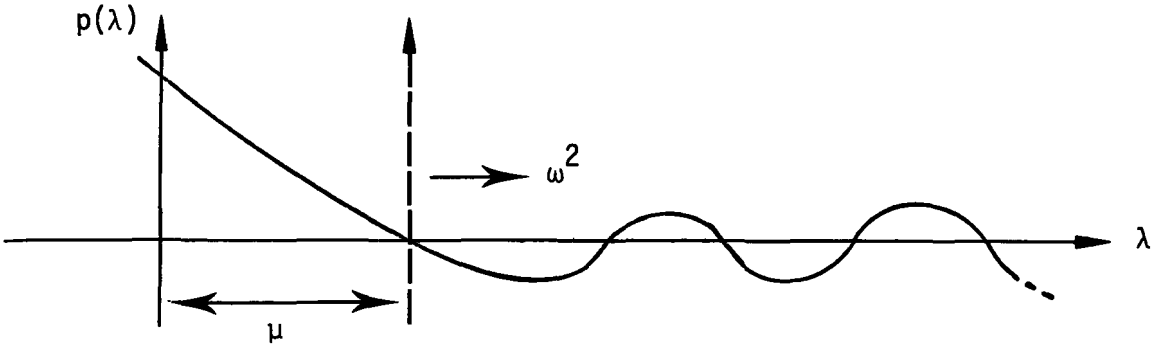
or

$$(\underline{K} + \mu\ \underline{M})\underline{\phi} = \lambda\ \underline{M}\ \underline{\phi}$$

$$\lambda = \omega^2 + \mu$$

or

$$\omega^2 = \lambda - \mu$$



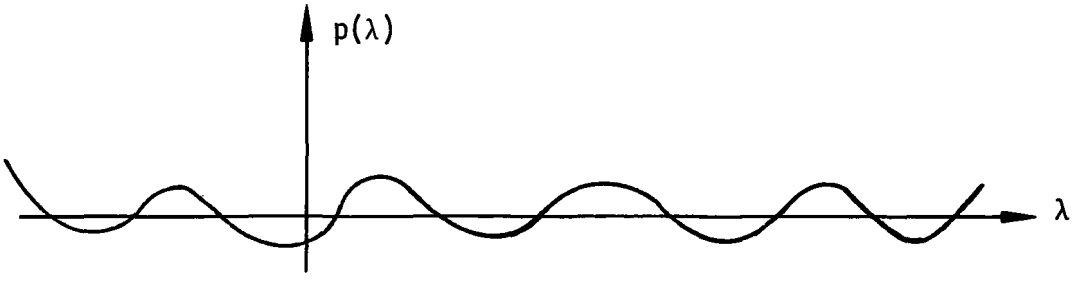
$$p(\lambda) = \det(\overline{K} - \lambda\, \underline{M}) \;;\quad \overline{K} = \underline{K} + \mu\, \underline{M}$$

In buckling analysis

$$\underline{K}\,\underline{\phi} = \lambda\, \underline{K}_G\, \underline{\phi}$$

where

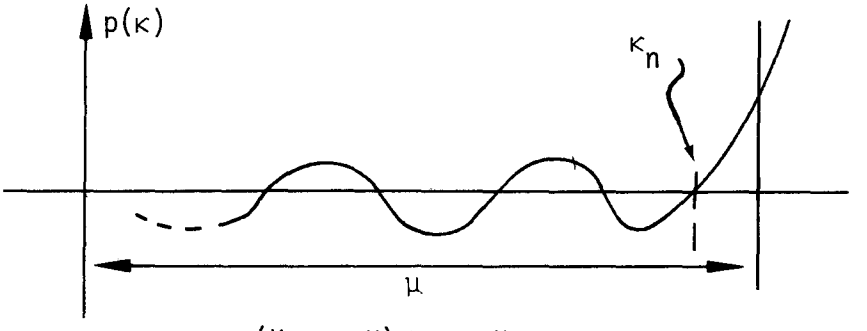
$$p(\lambda) = \det\,(\underline{K} - \lambda\, \underline{K}_G)$$



Rewrite problem as:

$$\underline{K}_G \underline{\phi} = \kappa \underline{K} \underline{\phi} \qquad \kappa = \frac{1}{\lambda}$$

and solve for largest κ:



$$(\underline{K}_G - \mu \underline{K}) \underline{\phi} = \eta \underline{K} \underline{\phi}$$

Traditional Approach: Transform the generalized EVP or quadratic EVP into a standard form, then solve using one of the many techniques available
e.g.

$$\underline{K} \underline{\phi} = \lambda \underline{M} \underline{\phi}$$

$$\underline{M} = \underline{\tilde{L}} \underline{\tilde{L}}^T \quad ; \quad \underline{\tilde{\phi}} = \underline{\tilde{L}}^T \underline{\phi}$$

hence

$$\underline{\tilde{K}} \underline{\tilde{\phi}} = \lambda \underline{\tilde{\phi}} \; ; \quad \underline{\tilde{K}} = \underline{\tilde{L}}^{-1} \underline{K} \underline{\tilde{L}}^{-T}$$

or

$$\underline{M} = \underline{W} \underline{D}^2 \underline{W}^T \qquad \text{etc...}$$

Direct solution is more effective.
Consider the Gen. EVP $\underline{K}\,\underline{\phi} = \lambda\,\underline{M}\,\underline{\phi}$
with

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

$$\underline{\phi}_1 \quad \underline{\phi}_2 \quad \underline{\phi}_3 \cdots \quad \underline{\phi}_n$$

eigenpairs $(\lambda_i , \underline{\phi}_i)$ $i = 1,\ldots,p$
are required or $i = r,\ldots,s$

The solution procedures in use operate on the basic equations that have to be satisfied.

---

1) VECTOR ITERATION TECHNIQUES

Equation: $\underline{K}\,\underline{\phi} = \lambda\,\underline{M}\,\underline{\phi}$

e.g. Inverse It. $\underline{K}\,\overline{\underline{x}}_{k+1} = M\,\underline{x}_k$

$$\underline{x}_{k+1} = \frac{\overline{\underline{x}}_{k+1}}{(\overline{\underline{x}}_{k+1}^{T}\,\underline{M}\,\overline{\underline{x}}_{k+1})^{\frac{1}{2}}} \longrightarrow \underline{\phi}_1$$

- Forward Iteration
- Rayleigh Quotient Iteration

  can be employed to calculate one eigenvalue and vector, deflate then to calculate additional eigenpair

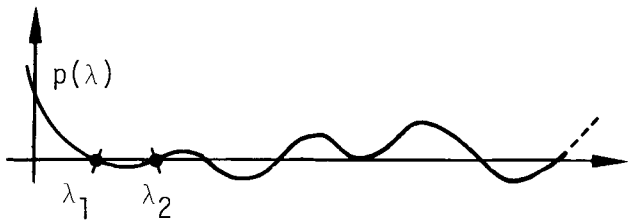
Convergence to "an eigenpair", which one is not guaranteed (convergence may also be slow)

2) POLYNOMIAL ITERATION METHODS

$$\underline{K}\,\underline{\phi} = \lambda\,\underline{M}\,\underline{\phi} \longrightarrow (\underline{K} - \lambda\,\underline{M})\,\underline{\phi} = \underline{0}$$

Hence

$$p(\lambda) = \det\,(\underline{K} - \lambda\underline{M}) = 0$$



Newton Iteration

$$\mu_{i+1} = \mu_i - \frac{p(\mu_i)}{p'(\mu_i)}$$

$$p(\lambda) = a_0 + a_1\lambda + a_2\lambda^2 + \ldots + a_n\lambda^n$$
$$= b_0\,(\lambda-\lambda_1)\,(\lambda-\lambda_2)\,\ldots\,(\lambda-\lambda_n)$$

Explicit polynomial iteration:

- Expand the polynomial and iterate for zeros.
- Technique not suitable for larger problems
  - much work to obtain a_i's
  - unstable process

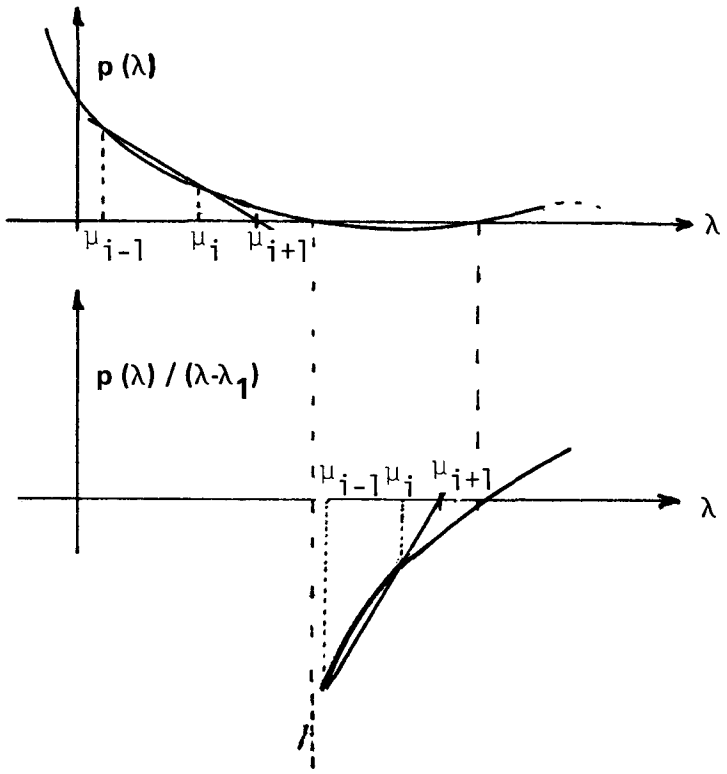
Implicit polynomial iteration:

$$p\,(\mu_i) = \det\,(\underline{K} - \mu_i\,\underline{M})$$
$$= \det\,\underline{L}\,\underline{D}\,\underline{L}^T = \prod_i d_{ii}$$

- accurate, provided we do not encounter large multipliers
- we directly solve for λ_1, ...
- use SECANT ITERATION:

$$\mu_{i+1} = \mu_i - \frac{p(\mu_i)}{\left(\frac{p(\mu_i) - p(\mu_{i-1})}{\mu_i - \mu_{i-1}}\right)}$$

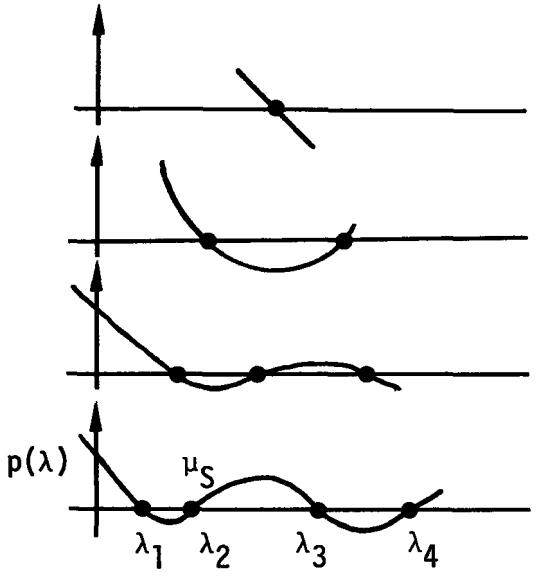
- deflate polynomial after convergence to λ_1



Convergence guaranteed to λ_1 , then λ_2 , etc. but can be slow when we calculate multiple roots.

Care need be taken in $\underline{L}\,\underline{D}\,\underline{L}^T$ factorization.

3) STURM SEQUENCE METHODS

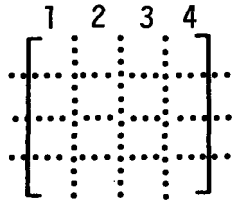


3rd associated constraint problem

2nd associated constraint problem

1st associated constraint problem

$$\underline{K}\,\underline{\phi} = \lambda\,\underline{M}\,\underline{\phi} \Rightarrow \underline{\phi} = \lambda\,\underline{M}\,\underline{\phi}$$



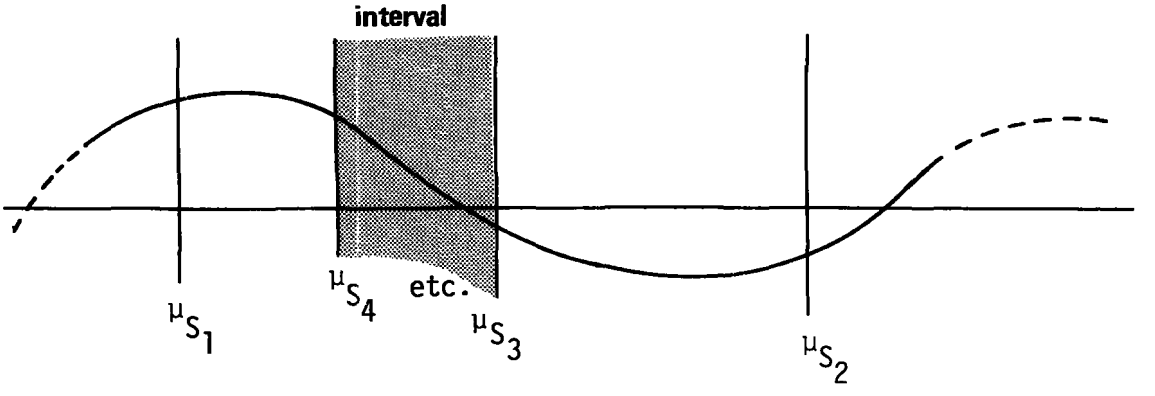
$$\underline{K} - \mu_S\,\underline{M} = \underline{L}\,\underline{D}\,\underline{L}^T$$

Number of negative elements in D is equal to the number of eigenvalues smaller than μ_S .

3) STURM SEQUENCE METHODS

Calculate $\underline{K} - \mu_{S_i} \underline{M} = \underline{L}\,\underline{D}\,\underline{L}^T$

Count number of negative elements in $\underline{D}$ and use a strategy to isolate eigenvalue(s).



- Need to take care in $\underline{L}\,\underline{D}\,\underline{L}^T$ factorization
- Convergence can be very slow

4) TRANSFORMATION METHODS

$$\underline{K}\,\underline{\phi} = \lambda\,\underline{M}\,\underline{\phi} \longrightarrow \begin{cases} \underline{\Phi}^T \underline{K}\,\underline{\Phi} = \underline{\Lambda} \\ \underline{\Phi}^T \underline{M}\,\Phi = \underline{I} \end{cases}$$

Construct $\underline{\Phi}$ iteratively:

$$\underline{\Phi} = [\underline{\phi}, \ldots\, \underline{\phi}_n]\,; \quad \underline{\Lambda} = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}$$

$$\underline{P}_k^T \cdots \underline{P}_2^T \underline{P}_1^T \underline{K} \underline{P}_1 \underline{P}_2 \cdots \underline{P}_k \rightarrow \underline{\Lambda}$$

$$\underline{P}_k^T \cdots \underline{P}_2^T \underline{P}_1^T \underline{M} \underline{P}_1 \underline{P}_2 \cdots \underline{P}_k \rightarrow \underline{I}$$

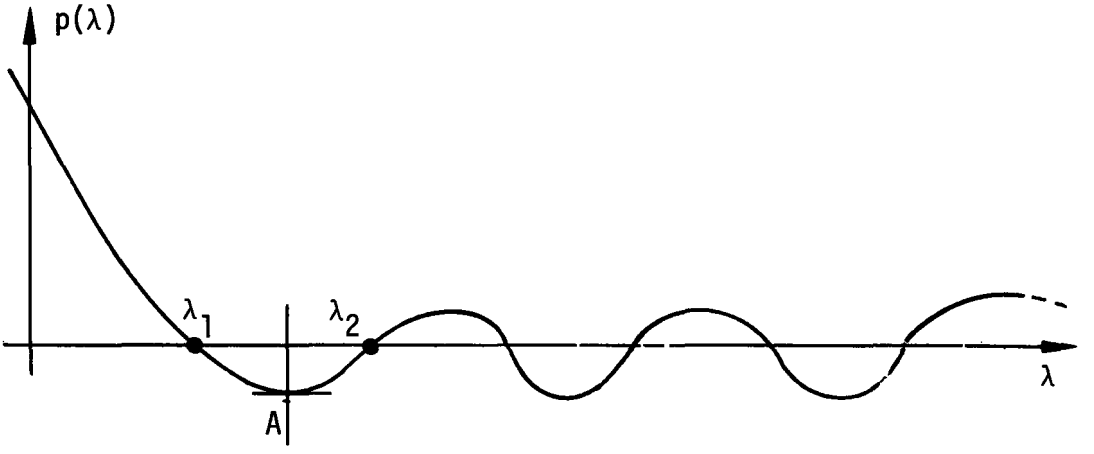
e.g. generalized Jacobi method

- Here we calculate all eigenpairs simultaneously
- Expensive and ineffective (impossible) or large problems.

For large eigenproblems it is best to use combinations of the above basic techniques:

- Determinant search to get near a root
- Vector iteration to obtain eigenvector and eigenvalue
- Transformation method for orthogonalization of iteration vectors.
- Sturm sequence method to ensure that required eigenvalue(s) has (or have) been calculated

THE DETERMINANT SEARCH METHOD



1) Iterate on polynomial to obtain shifts close to λ_1

$$p(\mu_i) = \det\ (\underline{K} - \mu_i\ \underline{M})$$

$$= \det\ \underline{L}\ \underline{D}\ \underline{L}^T = \prod_i d_{ii}$$

$$\mu_{i+1} = \mu_i - \eta \frac{p(\mu_i)}{\frac{p(\mu_i) - p(\mu_{i-1})}{\mu_i - \mu_{i-1}}}$$

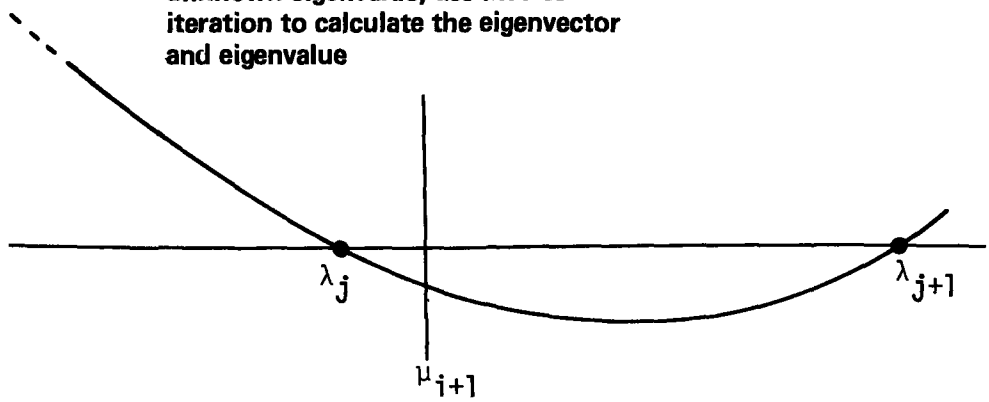
η is normally = 1.0

η = 2., 4., 8.,... when convergence is slow

Same procedure can be employed to obtain shift near λ_i, provided $p(\lambda)$ is deflated of $\lambda_1, \ldots, \lambda_{i-1}$

2) Use Sturm sequence property to check whether μ_{i+1} is larger than an unknown eigenvalue.

3) Once μ_{j+1} is larger than an unknown eigenvalue, use inverse iteration to calculate the eigenvector and eigenvalue



$$(\underline{K} - \mu_{i+1}\,\underline{M})\,\bar{\underline{x}}_{k+1} = \underline{M}\,\underline{x}_k \qquad k = 1,2,\ldots$$

$$\underline{x}_{k+1} = \frac{\bar{\underline{x}}_{k+1}}{(\bar{\underline{x}}_{k+1}^{T}\,\underline{M}\,\bar{\underline{x}}_{k+1})^{\frac{1}{2}}}$$

$$\rho(\bar{\underline{x}}_{k+1}) = \frac{\bar{\underline{x}}_{k+1}^{T}\,\underline{M}\,\underline{x}_k}{\bar{\underline{x}}_{k+1}^{T}\,\underline{M}\,\bar{\underline{x}}_{k+1}}$$

4) Iteration vector must be deflated of the previously calculated eigenvectors using, e.g. Gram-Schmidt orthogonalization.

If convergence is slow use Rayleigh quotient iteration

Advantage:
Calculates only eigenpairs actually required; no prior transformation of eigenproblem

Disadvantage:
Many triangular factorizations

- Effective only for small banded systems

We need an algorithm with less factorizations and more vector iterations when the bandwidth of the system is large.

SUBSPACE ITERATION METHOD

Iterate with q vectors when the lowest p eigenvalues and eigenvectors are required.

inverse iteration

$$\underline{K}\ \overline{\underline{X}}_{k+1} = \underline{M}\ \underline{X}_k \qquad k = 1,2,\ldots$$

$$\underline{K}_{k+1} = \overline{\underline{X}}_{k+1}^{T}\ \underline{K}\ \overline{\underline{X}}_{k+1}$$

$$\underline{M}_{k+1} = \overline{\underline{X}}_{k+1}^{T}\ \underline{M}\ \overline{\underline{X}}_{k+1}$$

$$\underline{K}_{k+1}\ \underline{Q}_{k+1} = \underline{M}_{k+1}\ \underline{Q}_{k+1}\ \underline{\Lambda}_{k+1}$$

$$\underline{X}_{k+1} = \overline{\underline{X}}_{k+1}\ \underline{Q}_{k+1}$$

"Under conditions" we have

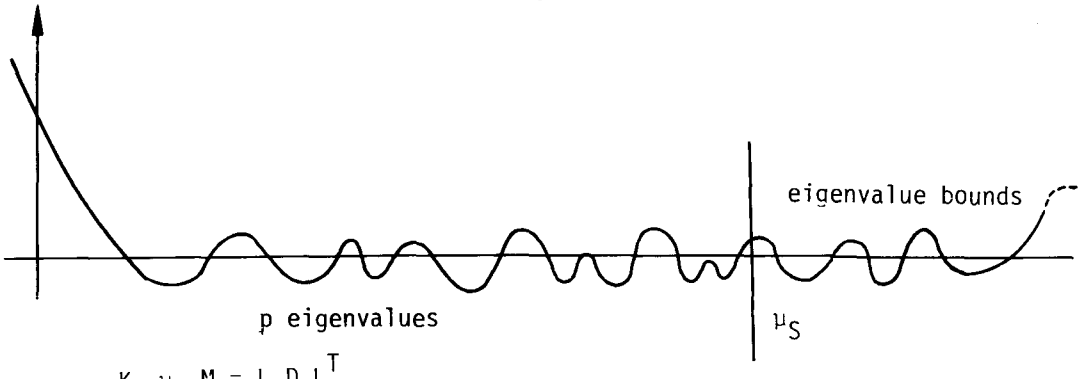
$$\underline{X}_{k+1} \longrightarrow \underline{\Phi}\ ; \quad \underline{\Lambda}_{k+1} \longrightarrow \underline{\Lambda}$$

$$\underline{\Phi} = [\underline{\phi}_1, \ldots, \underline{\phi}_q]\ ; \quad \underline{\Lambda} = \text{diag}\ (\lambda_i)$$

CONDITION:

starting subspace spanned by $\underline{X}_1$ must not be orthogonal to least dominant subspace required.

Use Sturm sequence check



$$\underline{K} - \mu_S \underline{M} = \underline{L}\ \underline{D}\ \underline{L}^T$$

no. of −ve elements in $\underline{D}$ must be equal to p .

Convergence rate:

$$\underline{\phi}_i \Rightarrow {}^{\lambda_i}/_{\lambda_{q+1}} \qquad \lambda_i \Rightarrow \left({}^{\lambda_i}/_{\lambda_{q+1}}\right)^2$$

convergence reached when $\left| \dfrac{\lambda_i^{(k)} - \lambda_i^{(k-1)}}{\lambda_i^{(k)}} \right| \leq \text{tol}$

Starting Vectors

Two choices

1) $\underline{x}_1 = \begin{bmatrix} \vdots \\ \vdots \end{bmatrix}$; $\underline{x}_j = \underline{e}_k$

$j = 2, \ldots, q-1$

$\underline{x}_q$ = random vector

2) **Lanczos method**
Here we need to use q much larger than p .

Checks on eigenpairs

1. **Sturm sequence checks**

2. $$\varepsilon_i = \frac{\| \underline{K}\, \underline{\phi}_i^{(\ell+1)} - \lambda_i^{(\ell+1)}\, \underline{M}\, \underline{\phi}_i^{(\ell+1)} \|_2}{\| \underline{K}\, \underline{\phi}_i^{(\ell+1)} \|_2}$$

important in <u>all</u> solutions.

Reference: An Accelerated Subspace Iteration Method, J. Computer Methods in Applied Mechanics and Engineering, Vol. 23, pp. 313 - 331, 1980.

Resource: Finite Element Procedures for Solids and Structures
Klaus-Jürgen Bathe