Matrix Iteration

Giacomo Boffi

Dipartimento di Ingegneria Civile e Ambientale, Politecnico di Milano

May 13, 2013

Outline

Introduction

Dynamic analysis of *MDOF* systems based on modal superposition is both simple and efficient

- simple: the modal response can be easily computed, analitically or numerically, with the techniques we have seen for *SDOF* systems,
- efficient: in most cases, only the modal responses of a few lower modes are required to accurately describe the structural response.

Introduction

Dynamic analysis of *MDOF* systems based on modal superposition is both simple and efficient

- simple: the modal response can be easily computed, analitically or numerically, with the techniques we have seen for *SDOF* systems,
- efficient: in most cases, only the modal responses of a few lower modes are required to accurately describe the structural response.

As the structural matrices are easily assembled using the *FEM*, our modal superposition procedure is ready to be applied to structures with tenth, thousands or millions of *DOF*'s! except that we can compute the eigenpairs only when the analyzed structure has two, three or maybe four degrees of freedom...

Introduction

Dynamic analysis of *MDOF* systems based on modal superposition is both simple and efficient

- simple: the modal response can be easily computed, analitically or numerically, with the techniques we have seen for *SDOF* systems,
- efficient: in most cases, only the modal responses of a few lower modes are required to accurately describe the structural response.

As the structural matrices are easily assembled using the *FEM*, our modal superposition procedure is ready to be applied to structures with tenth, thousands or millions of *DOF*'s! except that we can compute the eigenpairs only when the analyzed structure has two, three or maybe four degrees of freedom...
Enter the various *Matrix Iterations* procedures!

Equilibrium

First, we will see an iterative procedure whose outputs are the first, or fundamental, mode shape vector and the corresponding eigenvalue.
When an undamped system freely vibrates, the equation of motion is

$$\mathbf{K}\,\boldsymbol{\psi}_i = \omega_i^2 \mathbf{M}\,\boldsymbol{\psi}_i.$$

Equilibrium

First, we will see an iterative procedure whose outputs are the first, or fundamental, mode shape vector and the corresponding eigenvalue.
When an undamped system freely vibrates, the equation of motion is

$$\mathbf{K}\,\boldsymbol{\psi}_i = \omega_i^2\mathbf{M}\,\boldsymbol{\psi}_i.$$

In equilibrium terms, the elastic forces are equal to the inertial forces when the systems oscillates with frequency ω_i^2 and mode shape $\boldsymbol{\psi}_i$

Proposal of an iterative procedure

Our iterative procedure will be based on finding a new displacement vector $\mathbf{x}_{n+1}$ such that the elastic forces $\mathbf{f}_S = \mathbf{K}\,\mathbf{x}_{n+1}$ are in equilibrium with the inertial forces due to the *old* displacement vector $\mathbf{x}_n$, $\mathbf{f}_I = \omega_i^2\mathbf{M}\,\mathbf{x}_n$,

$$\mathbf{K}\,\mathbf{x}_{n+1} = \omega_i^2\mathbf{M}\,\mathbf{x}_n.$$

Premultiplying by the inverse of $\mathbf{K}$ and introducing the *Dynamic Matrix*, $\mathbf{D} = \mathbf{K}^{-1}\mathbf{M}$

$$\mathbf{x}_{n+1} = \omega_i^2\mathbf{K}^{-1}\mathbf{M}\,\mathbf{x}_n = \omega_i^2\mathbf{D}\,\mathbf{x}_n.$$

Proposal of an iterative procedure

Our iterative procedure will be based on finding a new displacement vector $\mathbf{x}_{n+1}$ such that the elastic forces $\mathbf{f}_S = \mathbf{K}\,\mathbf{x}_{n+1}$ are in equilibrium with the inertial forces due to the *old* displacement vector $\mathbf{x}_n$, $\mathbf{f}_I = \omega_i^2 \mathbf{M}\,\mathbf{x}_n$,

$$\mathbf{K}\,\mathbf{x}_{n+1} = \omega_i^2 \mathbf{M}\,\mathbf{x}_n.$$

Premultiplying by the inverse of $\mathbf{K}$ and introducing the *Dynamic Matrix*, $\mathbf{D} = \mathbf{K}^{-1}\mathbf{M}$

$$\mathbf{x}_{n+1} = \omega_i^2 \mathbf{K}^{-1}\mathbf{M}\,\mathbf{x}_n = \omega_i^2 \mathbf{D}\,\mathbf{x}_n.$$

In the generative equation above we miss a fundamental part, the square of the free vibration frequency ω_i^2.

The Matrix Iteration Procedure, 1

This problem is solved considering the $\mathbf{x}_n$ as a sequence of *normalized* vectors and introducing the idea of an *unnormalized* new displacement vector, $\hat{\mathbf{x}}_{n+1}$,

$$\hat{\mathbf{x}}_{n+1} = \mathbf{D}\,\mathbf{x}_n,$$

note that we removed the explicit dependency on ω_i^2.

The Matrix Iteration Procedure, 1

This problem is solved considering the $\mathbf{x}_n$ as a sequence of *normalized* vectors and introducing the idea of an *unnormalized* new displacement vector, $\hat{\mathbf{x}}_{n+1}$,

$$\hat{\mathbf{x}}_{n+1} = \mathbf{D}\,\mathbf{x}_n,$$

note that we removed the explicit dependency on ω_i^2.
The normalized vector is obtained applying to $\hat{\mathbf{x}}_{n+1}$ a normalizing factor, $\mathfrak{F}_{n+1}$,

$$\mathbf{x}_{n+1} = \frac{\hat{\mathbf{x}}_{n+1}}{\mathfrak{F}_{n+1}},$$

$$\text{but} \qquad \mathbf{x}_{n+1} = \omega_i^2 \mathbf{D}\,\mathbf{x}_n = \omega_i^2\,\hat{\mathbf{x}}_{n+1}, \quad \Rightarrow \quad \frac{1}{\mathfrak{F}} = \omega_i^2$$

The Matrix Iteration Procedure, 1

This problem is solved considering the $\mathbf{x}_n$ as a sequence of *normalized* vectors and introducing the idea of an *unnormalized* new displacement vector, $\hat{\mathbf{x}}_{n+1}$,

$$\hat{\mathbf{x}}_{n+1} = \mathbf{D}\,\mathbf{x}_n,$$

note that we removed the explicit dependency on ω_i^2.
The normalized vector is obtained applying to $\hat{\mathbf{x}}_{n+1}$ a normalizing factor, $\mathfrak{F}_{n+1}$,

$$\mathbf{x}_{n+1} = \frac{\hat{\mathbf{x}}_{n+1}}{\mathfrak{F}_{n+1}},$$

$$\text{but} \qquad \mathbf{x}_{n+1} = \omega_i^2 \mathbf{D}\,\mathbf{x}_n = \omega_i^2\,\hat{\mathbf{x}}_{n+1}, \quad \Rightarrow \quad \frac{1}{\mathfrak{F}} = \omega_i^2$$

If we agree that, near convergence, $\mathbf{x}_{n+1} \approx \mathbf{x}_n$, substituting in the previous equation we have

$$\mathbf{x}_{n+1} \approx \mathbf{x}_n = \omega_i^2\,\hat{\mathbf{x}}_{n+1} \quad \Rightarrow \quad \omega_i^2 \approx \frac{\mathbf{x}_n}{\hat{\mathbf{x}}_{n+1}}.$$

The Matrix Iteration Procedure, 1

This problem is solved considering the $\mathbf{x}_n$ as a sequence of *normalized* vectors and introducing the idea of an *unnormalized* new displacement vector, $\hat{\mathbf{x}}_{n+1}$,

$$\hat{\mathbf{x}}_{n+1} = \mathbf{D}\,\mathbf{x}_n,$$

note that we removed the explicit dependency on ω_i^2.
The normalized vector is obtained applying to $\hat{\mathbf{x}}_{n+1}$ a normalizing factor, $\mathfrak{F}_{n+1}$,

$$\mathbf{x}_{n+1} = \frac{\hat{\mathbf{x}}_{n+1}}{\mathfrak{F}_{n+1}},$$

$$\text{but} \qquad \mathbf{x}_{n+1} = \omega_i^2 \mathbf{D}\,\mathbf{x}_n = \omega_i^2\,\hat{\mathbf{x}}_{n+1}, \quad \Rightarrow \quad \frac{1}{\mathfrak{F}} = \omega_i^2$$

If we agree that, near convergence, $\mathbf{x}_{n+1} \approx \mathbf{x}_n$, substituting in the previous equation we have

$$\mathbf{x}_{n+1} \approx \mathbf{x}_n = \omega_i^2\,\hat{\mathbf{x}}_{n+1} \quad \Rightarrow \quad \omega_i^2 \approx \frac{\mathbf{x}_n}{\hat{\mathbf{x}}_{n+1}}.$$

Of course the division of two vectors is not an option, so we want to twist it into something useful.

Normalization

First, consider $\mathbf{x}_n = \boldsymbol{\psi}_i$: in this case, for $j = 1, \ldots, N$ it is

$$x_{n,j}/\hat{x}_{n+1,j} = \omega_i^2.$$

Analogously for $\mathbf{x}_n \neq \boldsymbol{\psi}_i$ it was demonstrated that

$$\min_{j=1,\ldots,N} \left\{ \frac{x_{n,j}}{\hat{x}_{n+1,j}} \right\} \le \omega_i^2 \le \max_{j=1,\ldots,N} \left\{ \frac{x_{n,j}}{\hat{x}_{n+1,j}} \right\}.$$

Normalization

First, consider $\mathbf{x}_n = \boldsymbol{\psi}_i$: in this case, for $j = 1, \ldots, N$ it is

$$x_{n,j}/\hat{x}_{n+1,j} = \omega_i^2.$$

Analogously for $\mathbf{x}_n \neq \boldsymbol{\psi}_i$ it was demonstrated that

$$\min_{j=1,\ldots,N}\left\{\frac{x_{n,j}}{\hat{x}_{n+1,j}}\right\} \le \omega_i^2 \le \max_{j=1,\ldots,N}\left\{\frac{x_{n,j}}{\hat{x}_{n+1,j}}\right\}.$$

A more rational approach would make reference to a proper vector norm, so using our preferred vector norm we can write

$$\omega_i^2 \approx \frac{\hat{\mathbf{x}}_{n+1}^T \mathbf{M}\,\mathbf{x}_n}{\hat{\mathbf{x}}_{n+1}^T \mathbf{M}\,\hat{\mathbf{x}}_{n+1}},$$

Normalization

First, consider $\mathbf{x}_n = \boldsymbol{\psi}_i$: in this case, for $j = 1, \ldots, N$ it is

$$x_{n,j}/\hat{x}_{n+1,j} = \omega_i^2.$$

Analogously for $\mathbf{x}_n \neq \boldsymbol{\psi}_i$ it was demonstrated that

$$\min_{j=1,\ldots,N} \left\{ \frac{x_{n,j}}{\hat{x}_{n+1,j}} \right\} \leq \omega_i^2 \leq \max_{j=1,\ldots,N} \left\{ \frac{x_{n,j}}{\hat{x}_{n+1,j}} \right\}.$$

A more rational approach would make reference to a proper vector norm, so using our preferred vector norm we can write

$$\omega_i^2 \approx \frac{\hat{\mathbf{x}}_{n+1}^T \mathbf{M}\, \mathbf{x}_n}{\hat{\mathbf{x}}_{n+1}^T \mathbf{M}\, \hat{\mathbf{x}}_{n+1}},$$

(if memory helps, this is equivalent to the R_{11} approximation, that we introduced studying Rayleigh quotient refinements).

Proof of Convergence, 1

Until now we postulated that the sequence $\mathbf{x}_n$ converges to some, unspecified eigenvector $\boldsymbol{\psi}_i$, now we will demonstrate that the sequence converge to the first, or fundamental mode shape,

$$\lim_{n\to\infty} \mathbf{x}_n = \boldsymbol{\psi}_1.$$

1. Expand $\mathbf{x}_0$ in terms of eigenvectors and modal coordinates:

$$\mathbf{x}_0 = \boldsymbol{\psi}_1 q_{1,0} + \boldsymbol{\psi}_2 q_{2,0} + \boldsymbol{\psi}_3 q_{3,0} + \cdots ,$$

Proof of Convergence, 1

Until now we postulated that the sequence $\mathbf{x}_n$ converges to some, unspecified eigenvector $\boldsymbol{\psi}_i$, now we will demonstrate that the sequence converge to the first, or fundamental mode shape,

$$\lim_{n\to\infty} \mathbf{x}_n = \boldsymbol{\psi}_1.$$

1. Expand $\mathbf{x}_0$ in terms of eigenvectors and modal coordinates:

$$\mathbf{x}_0 = \boldsymbol{\psi}_1 q_{1,0} + \boldsymbol{\psi}_2 q_{2,0} + \boldsymbol{\psi}_3 q_{3,0} + \cdots ,$$

2. The inertial forces, assuming that the system is vibrating according to the fundamental frequency, are

$$\begin{aligned}\mathbf{f}_{I,n=0} &= \omega_1^2 \mathbf{M} \left(\boldsymbol{\psi}_1 q_{1,0} + \boldsymbol{\psi}_2 q_{2,0} + \boldsymbol{\psi}_3 q_{3,0} + \cdots\right)\\ &= \mathbf{M}\left(\omega_1^2 \boldsymbol{\psi}_1 q_{1,0} \frac{\omega_1^2}{\omega_1^2} + \omega_2^2 \boldsymbol{\psi}_2 q_{2,0} \frac{\omega_1^2}{\omega_2^2} + \cdots\right).\end{aligned}$$

Proof of Convergence, 2

3. The deflections due to these forces (no hat!, we have multiplied by ω_1^2) are

$$\mathbf{x}_{n=1} = \mathbf{K}^{-1}\mathbf{M}\left(\omega_1^2\boldsymbol{\psi}_1 q_{1,0}\frac{\omega_1^2}{\omega_1^2} + \omega_2^2\boldsymbol{\psi}_2 q_{2,0}\frac{\omega_1^2}{\omega_2^2} + \cdots\right),$$

(note that we have multiplied and divided each term by ω_i^2).

Proof of Convergence, 2

3. The deflections due to these forces (no hat!, we have multiplied by ω_1^2) are

$$\mathbf{x}_{n=1} = \mathbf{K}^{-1}\mathbf{M}\left(\omega_1^2\boldsymbol{\psi}_1 q_{1,0}\frac{\omega_1^2}{\omega_1^2} + \omega_2^2\boldsymbol{\psi}_2 q_{2,0}\frac{\omega_1^2}{\omega_2^2} + \cdots\right),$$

(note that we have multiplied and divided each term by ω_i^2).

4. Using $\omega_j^2\mathbf{M}\,\boldsymbol{\psi}_j = \mathbf{K}\boldsymbol{\psi}_j$,

$$\begin{aligned}\mathbf{x}_{n=1} &= \mathbf{K}^{-1}\left(\mathbf{K}\boldsymbol{\psi}_1 q_{1,0}\frac{\omega_1^2}{\omega_1^2} + \mathbf{K}\boldsymbol{\psi}_2 q_{2,0}\frac{\omega_1^2}{\omega_2^2} + \mathbf{K}\boldsymbol{\psi}_3 q_{3,0}\frac{\omega_1^2}{\omega_3^2} + \cdots\right)\\ &= \boldsymbol{\psi}_1 q_{1,0}\frac{\omega_1^2}{\omega_1^2} + \boldsymbol{\psi}_2 q_{2,0}\frac{\omega_1^2}{\omega_2^2} + \boldsymbol{\psi}_3 q_{3,0}\frac{\omega_1^2}{\omega_3^2} + \cdots\end{aligned}$$

Proof of Convergence, 3

5. Applying again the previous procedure, i.e., premultiply the right member by $\omega_1^2\mathbf{D}$, multiplying and dividing each term by ω_i^2, symplifying, we have

$$\mathbf{x}_{n=2} = \boldsymbol{\psi}_1 q_{1,0} \left(\frac{\omega_1^2}{\omega_1^2}\right)^2 + \boldsymbol{\psi}_2 q_{2,0} \left(\frac{\omega_1^2}{\omega_2^2}\right)^2 + \boldsymbol{\psi}_3 q_{3,0} \left(\frac{\omega_1^2}{\omega_3^2}\right)^2 + \cdots$$

Proof of Convergence, 3

5. Applying again the previous procedure, i.e., premultiply the right member by $\omega_1^2\mathbf{D}$, multiplying and dividing each term by ω_i^2, symplifying, we have

$$\mathbf{x}_{n=2} = \boldsymbol{\psi}_1 q_{1,0}\left(\frac{\omega_1^2}{\omega_1^2}\right)^2 + \boldsymbol{\psi}_2 q_{2,0}\left(\frac{\omega_1^2}{\omega_2^2}\right)^2 + \boldsymbol{\psi}_3 q_{3,0}\left(\frac{\omega_1^2}{\omega_3^2}\right)^2 + \cdots$$

6. repeating the procedure for n times starting from $\mathbf{x}_0$, we have

$$\mathbf{x}_n = \boldsymbol{\psi}_1 q_{1,0}\left(\frac{\omega_1^2}{\omega_1^2}\right)^n + \boldsymbol{\psi}_2 q_{2,0}\left(\frac{\omega_1^2}{\omega_2^2}\right)^n + \boldsymbol{\psi}_3 q_{3,0}\left(\frac{\omega_1^2}{\omega_3^2}\right)^n + \cdots$$

Proof of Convergence, 4

Going to the limit,

$$\lim_{n\to\infty} \mathbf{x}_n = \boldsymbol{\psi}_1 q_{1,0}$$

because

$$\lim_{n\to\infty} \left(\frac{\omega_1^2}{\omega_j^2}\right)^n = \delta_{1j}$$

Consequently,

$$\lim_{n\to\infty} \frac{|\mathbf{x}_n|}{|\hat{\mathbf{x}}_n|} = \omega_1^2$$

Purified Vectors

If we know $\boldsymbol{\psi}_1$ and ω_1^2 from the matrix iteration procedure it is possible to compute the second eigenpair, following a slightly different procedure.

Purified Vectors

If we know $\boldsymbol{\psi}_1$ and ω_1^2 from the matrix iteration procedure it is possible to compute the second eigenpair, following a slightly different procedure.
Express the initial iterate in terms of the (unknown) eigenvectors,

$$\mathbf{x}_{n=0} = \boldsymbol{\Psi}\,\mathbf{q}_{n=0}$$

and premultiply by the (known) $\boldsymbol{\psi}_1^T\mathbf{M}$:

$$\boldsymbol{\psi}_1^T\mathbf{M}\,\mathbf{x}_{n=0} = M_1 q_{1,n=0}$$

solving for $q_{1,n=0}$

$$q_{1,n=0} = \frac{\boldsymbol{\psi}_1^T\mathbf{M}\,\mathbf{x}_{n=0}}{M_1}.$$

Purified Vectors

If we know $\boldsymbol{\psi}_1$ and ω_1^2 from the matrix iteration procedure it is possible to compute the second eigenpair, following a slightly different procedure.
Express the initial iterate in terms of the (unknown) eigenvectors,

$$\mathbf{x}_{n=0} = \mathbf{\Psi}\,\mathbf{q}_{n=0}$$

and premultiply by the (known) $\boldsymbol{\psi}_1^T\mathbf{M}$:

$$\boldsymbol{\psi}_1^T\mathbf{M}\,\mathbf{x}_{n=0} = M_1\,q_{1,n=0}$$

solving for $q_{1,n=0}$

$$q_{1,n=0} = \frac{\boldsymbol{\psi}_1^T\mathbf{M}\,\mathbf{x}_{n=0}}{M_1}.$$

Knowing the amplitude of the 1st modal contribution to $\mathbf{x}_{n=0}$ we can write a *purified* vector,

$$\mathbf{y}_{n=0} = \mathbf{x}_{n=0} - \boldsymbol{\psi}_1\,q_{1,n=0}.$$

Convergence (?)

It is easy to demonstrate that using $\mathbf{y}_{n=0}$ as our starting vector

$$\lim_{n\to\infty}\mathbf{y}_n=\boldsymbol{\psi}_2 q_{2,n=0}, \qquad \lim_{n\to\infty}\frac{|\mathbf{y}_n|}{|\hat{\mathbf{y}}_n|}=\omega_2^2.$$

because the initial amplitude of the first mode is null.

Convergence (?)

It is easy to demonstrate that using $\mathbf{y}_{n=0}$ as our starting vector

$$\lim_{n\to\infty} \mathbf{y}_n = \boldsymbol{\psi}_2 q_{2,n=0}, \qquad \lim_{n\to\infty} \frac{|\mathbf{y}_n|}{|\hat{\mathbf{y}}_n|} = \omega_2^2.$$

because the initial amplitude of the first mode is null.
Due to numerical errors in the determination of fundamental mode and in the procedure itself, using a plain matrix iteration the procedure however converges to the 1st eigenvector, so to preserve convergence to the 2nd mode it is necessary that the iterated vector $\mathbf{y}_n$ is *purified* at each step n.

Purification Procedure

The purification procedure is simple, at each step the amplitude of the 1st mode is first computed, then removed from the iterated vector $\mathbf{y}_n$

$$q_{1,n} = \boldsymbol{\psi}_1^T \mathbf{M} \mathbf{y}_n / M_1,$$

$$\hat{\mathbf{y}}_{n+1} = \mathbf{D}\left(\mathbf{y}_n - \boldsymbol{\psi}_1 q_{1,n}\right) = \mathbf{D}\left(\mathbf{I} - \frac{1}{M_1}\boldsymbol{\psi}_1\boldsymbol{\psi}_1^T\mathbf{M}\right)\mathbf{y}_n$$

Purification Procedure

The purification procedure is simple, at each step the amplitude of the 1st mode is first computed, then removed from the iterated vector $\mathbf{y}_n$

$$q_{1,n} = \boldsymbol{\psi}_1^T \mathbf{M} \mathbf{y}_n / M_1,$$

$$\hat{\mathbf{y}}_{n+1} = \mathbf{D}\left(\mathbf{y}_n - \boldsymbol{\psi}_1 q_{1,n}\right) = \mathbf{D}\left(\mathbf{I} - \frac{1}{M_1}\boldsymbol{\psi}_1\boldsymbol{\psi}_1^T\mathbf{M}\right)\mathbf{y}_n$$

Introducing the *sweeping matrix* $\mathbf{S}_1 = \mathbf{I} - \frac{1}{M_1}\boldsymbol{\psi}_1\boldsymbol{\psi}_1^T\mathbf{M}$ and the modified dynamic matrix $\mathbf{D}_2 = \mathbf{D}\mathbf{S}_1$, we can write

$$\hat{\mathbf{y}}_{n+1} = \mathbf{D}\mathbf{S}_1\mathbf{y}_n = \mathbf{D}_2\mathbf{y}_n.$$

Purification Procedure

The purification procedure is simple, at each step the amplitude of the 1st mode is first computed, then removed from the iterated vector $\mathbf{y}_n$

$$q_{1,n} = \boldsymbol{\psi}_1^T \mathbf{M} \mathbf{y}_n / M_1,$$

$$\hat{\mathbf{y}}_{n+1} = \mathbf{D} \left(\mathbf{y}_n - \boldsymbol{\psi}_1 q_{1,n} \right) = \mathbf{D} \left(\mathbf{I} - \frac{1}{M_1} \boldsymbol{\psi}_1 \boldsymbol{\psi}_1^T \mathbf{M} \right) \mathbf{y}_n$$

Introducing the *sweeping matrix* $\mathbf{S}_1 = \mathbf{I} - \frac{1}{M_1} \boldsymbol{\psi}_1 \boldsymbol{\psi}_1^T \mathbf{M}$ and the modified dynamic matrix $\mathbf{D}_2 = \mathbf{D}\mathbf{S}_1$, we can write

$$\hat{\mathbf{y}}_{n+1} = \mathbf{D}\mathbf{S}_1 \mathbf{y}_n = \mathbf{D}_2 \mathbf{y}_n.$$

This is known as *matrix iteration with sweeps*.

Third Mode

Using again the idea of purifying the iterated vector, starting with the knowledge of the first and the second eigenpair,

$$\hat{\mathbf{y}}_{n+1} = \mathbf{D}\left(\mathbf{y}_n - \boldsymbol{\psi}_1 q_{1,n} - \boldsymbol{\psi}_2 q_{2,n}\right)$$

with $q_{n,1}$ as before and

$$q_{2,n} = \boldsymbol{\psi}_2^T \mathbf{M}\mathbf{y}_n / M_2,$$

substituting in the expression for the purified vector

$$\hat{\mathbf{y}}_{n+1} = \mathbf{D}\big(\underbrace{\mathbf{I} - \frac{1}{M_1}\boldsymbol{\psi}_1\boldsymbol{\psi}_1^T\mathbf{M}}_{\mathbf{S}_1} - \frac{1}{M_2}\boldsymbol{\psi}_2\boldsymbol{\psi}_2^T\mathbf{M}\big)$$

Third Mode

Using again the idea of purifying the iterated vector, starting with the knowledge of the first and the second eigenpair,

$$\hat{\mathbf{y}}_{n+1} = \mathbf{D}\left(\mathbf{y}_n - \boldsymbol{\psi}_1 q_{1,n} - \boldsymbol{\psi}_2 q_{2,n}\right)$$

with $q_{n,1}$ as before and

$$q_{2,n} = \boldsymbol{\psi}_2^T \mathbf{M} \mathbf{y}_n / M_2,$$

substituting in the expression for the purified vector

$$\hat{\mathbf{y}}_{n+1} = \mathbf{D}\big(\underbrace{\mathbf{I} - \frac{1}{M_1}\boldsymbol{\psi}_1\boldsymbol{\psi}_1^T\mathbf{M}}_{\mathbf{S}_1} - \frac{1}{M_2}\boldsymbol{\psi}_2\boldsymbol{\psi}_2^T\mathbf{M}\big)$$

The conclusion is that the sweeping matrix and the modified dynamic matrix to be used to compute the 3rd eigenvector are

$$\mathbf{S}_2 = \mathbf{S}_1 - \frac{1}{M_2}\boldsymbol{\psi}_2\boldsymbol{\psi}_2^T\mathbf{M}, \qquad \mathbf{D}_3 = \mathbf{D}\,\mathbf{S}_2.$$

Generalization to Higher Modes

The results obtained for the third mode are easily generalised.
It is easy to verify that the following procedure can be used to compute all the modes.

Define $\mathbf{S}_0 = \mathbf{I}$, let $i = 1$,

1. compute the modified dynamic matrix to be used for mode i,

$$\mathbf{D}_i = \mathbf{D}\,\mathbf{S}_{i-i}$$

2. compute $\boldsymbol{\psi}_i$ using the modified dynamic matrix;
3. compute the modal mass $M_i = \boldsymbol{\psi}^T\mathbf{M}\,\boldsymbol{\psi}$;
4. compute the sweeping matrix $\mathbf{S}_i$ that *sweeps* the contributions of the first i modes from trial vectors,

$$\mathbf{S}_i = \mathbf{S}_{i-1} - \frac{1}{M_i}\boldsymbol{\psi}_i\boldsymbol{\psi}_i^T\mathbf{M};$$

5. increment i, `GOTO 1`.

Generalization to Higher Modes

The results obtained for the third mode are easily generalised.
It is easy to verify that the following procedure can be used to compute all the modes.

Define $\mathbf{S}_0 = \mathbf{I}$, let $i = 1$,

1. compute the modified dynamic matrix to be used for mode i,

   $$\mathbf{D}_i = \mathbf{D}\,\mathbf{S}_{i-i}$$

2. compute $\boldsymbol{\psi}_i$ using the modified dynamic matrix;
3. compute the modal mass $M_i = \boldsymbol{\psi}^T\mathbf{M}\,\boldsymbol{\psi}$;
4. compute the sweeping matrix $\mathbf{S}_i$ that *sweeps* the contributions of the first i modes from trial vectors,

   $$\mathbf{S}_i = \mathbf{S}_{i-1} - \frac{1}{M_i}\boldsymbol{\psi}_i\boldsymbol{\psi}_i^T\mathbf{M};$$

5. increment i, `GOTO 1.`

Well, we finally have a method that can be used to compute all the eigenpairs of our dynamic problems, full circle!

Discussion

The method of matrix iteration with sweeping is not used in production because

1. **D** is a full matrix, even if **M** and **K** are banded matrices, and the matrix product that is the essential step in every iteration is computationally onerous,
2. the procedure is however affected by numerical errors.

While it is possible to compute all the eigenvectors and eigenvalues of a large problem using our iterative procedure, we can first optimize our procedure and later seek for different, more efficient iterative procedures.

Introduction to Inverse Iteration

Inverse iteration is based on the fact that the symmetric stiffness matrix has a banded structure, that is a relatively large triangular portion of the matrix is composed by zeroes

Introduction to Inverse Iteration

Inverse iteration is based on the fact that the symmetric stiffness matrix has a banded structure, that is a relatively large triangular portion of the matrix is composed by zeroes

The banded structure is due to the *FEM* model that implies that in an equation of equilibrium the only non zero elastic force coefficients are due to degrees of freedom pertaining to *FE* that contains the degree of freedom for which the equilibrium is written).

Introduction to Inverse Iteration

Inverse iteration is based on the fact that the symmetric stiffness matrix has a banded structure, that is a relatively large triangular portion of the matrix is composed by zeroes

The banded structure is due to the *FEM* model that implies that in an equation of equilibrium the only non zero elastic force coefficients are due to degrees of freedom pertaining to *FE* that contains the degree of freedom for which the equilibrium is written).

Definition of LU decomposition

Every symmetric, banded matrix can be subjected to a so called LU decomposition, that is, for $\mathbf{K}$ we write

$$\mathbf{K} = \mathbf{L}\,\mathbf{U}$$

where $\mathbf{L}$ and $\mathbf{U}$ are, respectively, a lower- and an upper-banded matrix.
If we denote with b the bandwidth of $\mathbf{K}$, we have

$$\mathbf{L} = \left[l_{ij}\right] \quad \text{with } l_{ij} \equiv 0 \text{ for } \begin{cases} i < j \\ j < i - b \end{cases}$$

and

$$\mathbf{U} = \left[u_{ij}\right] \quad \text{with } u_{ij} \equiv 0 \text{ for } \begin{cases} i > j \\ j > i + b \end{cases}$$

Twice the equations?

In this case, with $\mathbf{w}_n = \mathbf{M}\,\mathbf{x}_n$, the recursion can be written

$$\mathbf{L}\,\mathbf{U}\,\mathbf{x}_{n+1} = \mathbf{w}_n$$

or as a system of equations,

$$\mathbf{U}\,\mathbf{x}_{n+1} = \mathbf{z}_{n+1}$$
$$\mathbf{L}\,\mathbf{z}_{n+1} = \mathbf{w}_n$$

Apparently, we have doubled the number of unknowns, but the z_j's can be easily computed by the procedure of *back substitution*.

Back Substitution

Temporarily dropping the n and $n+1$ subscripts, we can write

$$\begin{aligned}
z_1 &= (w_1)/l_{11}\\
z_2 &= (w_2 - l_{21}z_1)/l_{22}\\
z_3 &= (w_3 - l_{31}z_1 - l_{32}z_2)/l_{33}\\
&\dots\\
z_j &= (w_j - \sum_{k=1}^{j-1} l_{jk}z_k)/l_{jj}\\
&\dots
\end{aligned}$$

Back Substitution

Temporarily dropping the n and $n+1$ subscripts, we can write

$$\begin{aligned}
z_1 &= (w_1)/l_{11} \\
z_2 &= (w_2 - l_{21}z_1)/l_{22} \\
z_3 &= (w_3 - l_{31}z_1 - l_{32}z_2)/l_{33} \\
&\dots \\
z_j &= (w_j - \sum_{k=1}^{j-1} l_{jk}z_k)/l_{jj} \\
&\dots
\end{aligned}$$

The $\mathbf{x}$ are then given by $\mathbf{U}\,\mathbf{x} = \mathbf{z}$.

Back Substitution

We have computed $\mathbf{z}$ by back substitution, we must solve $\mathbf{U}\,\mathbf{x}=\mathbf{z}$ but U is upper triangular, so we have

$$\begin{aligned}
x_N &= (z_N)/u_{NN}\\
x_{N-1} &= (z_{N-1}-u_{N-1,N}z_N)/u_{N-1,N-1}\\
x_{N-2} &= (z_{N-2}-u_{N-2,N}z_N-u_{N-2,N-1}z_{N-1})/u_{N-2,N-2}\\
&\dots\\
x_{N-j} &= (z_{N-j}-\sum_{k=0}^{j-1}u_{N-j,N-k}z_{N-k})/u_{N-j,N-j},
\end{aligned}$$

Back Substitution

We have computed $\mathbf{z}$ by back substitution, we must solve $\mathbf{U}\,\mathbf{x} = \mathbf{z}$ but U is upper triangular, so we have

$$\begin{aligned}
x_N &= (z_N)/u_{NN}\\
x_{N-1} &= (z_{N-1} - u_{N-1,N}z_N)/u_{N-1,N-1}\\
x_{N-2} &= (z_{N-2} - u_{N-2,N}z_N - u_{N-2,N-1}z_{N-1})/u_{N-2,N-2}\\
&\dots\\
x_{N-j} &= (z_{N-j} - \sum_{k=0}^{j-1} u_{N-j,N-k}z_{N-k})/u_{N-j,N-j},
\end{aligned}$$

For moderately large systems, the reduction in operations count given by back substitution with respect to matrix multiplication is so large that the additional cost of the LU decomposition is negligible.

Introduction to Shifts

Inverse iteration can be applied to each step of matrix iteration with sweeps, or to each step of a different procedure intended to compute all the eigenpairs, the *matrix iteration with shifts*.

Matrix Iteration with Shifts, 1

If we write

$$\omega_i^2 = \mu + \lambda_i,$$

where μ is a *shift* and λ_i is a *shifted eigenvalue*, the eigenvalue problem can be formulated as

$$\mathbf{K}\,\boldsymbol{\psi}_i = (\mu + \lambda_i)\mathbf{M}\,\boldsymbol{\psi}_i$$

or

$$(\mathbf{K} - \mu\mathbf{M})\boldsymbol{\psi}_i = \lambda_i\mathbf{M}\,\boldsymbol{\psi}_i.$$

If we introduce a modified stiffness matrix

$$\overline{\mathbf{K}} = \mathbf{K} - \mu\mathbf{M},$$

we recognize that we have a *new* problem, that has *exactly* the same eigenvectors and *shifted* eigenvalues,

$$\overline{\mathbf{K}}\,\boldsymbol{\phi}_i = \lambda_i\mathbf{M}\boldsymbol{\phi}_i,$$

where

$$\boldsymbol{\phi}_i = \boldsymbol{\psi}_i, \qquad \lambda_i = \omega_i^2 - \mu.$$

Matrix Iteration with Shifts, 2

The shifted eigenproblem can be solved, e.g., by matrix iteration and the procedure will converge to the *smallest absolute value* shifted eigenvalue and to the associated eigenvector. After convergence is reached,

$$\boldsymbol{\psi}_i = \boldsymbol{\phi}_i, \qquad \omega_i^2 = \lambda_i + \mu.$$

Matrix Iteration with Shifts, 2

The shifted eigenproblem can be solved, e.g., by matrix iteration and the procedure will converge to the *smallest absolute value* shifted eigenvalue and to the associated eigenvector. After convergence is reached,

$$\boldsymbol{\psi}_i = \boldsymbol{\phi}_i, \qquad \omega_i^2 = \lambda_i + \mu.$$

The convergence of the method can be greatly enhanced if the shift μ is updated every few steps during the iterative procedure using the current best estimate of λ_i,

$$\lambda_{i,n+1} = \frac{\hat{\mathbf{x}}_{n+1} \mathbf{M}\, \mathbf{x}_n}{\hat{\mathbf{x}}_{n+1} \mathbf{M}\, \hat{\mathbf{x}}_n},$$

to improve the modified stiffness matrix to be used in the following iterations,

$$\overline{\mathbf{K}} = \overline{\mathbf{K}} - \lambda_{i,n+1} \mathbf{M}$$

Matrix Iteration with Shifts, 2

The shifted eigenproblem can be solved, e.g., by matrix iteration and the procedure will converge to the *smallest absolute value* shifted eigenvalue and to the associated eigenvector. After convergence is reached,

$$\boldsymbol{\psi}_i = \boldsymbol{\phi}_i, \qquad \omega_i^2 = \lambda_i + \mu.$$

The convergence of the method can be greatly enhanced if the shift μ is updated every few steps during the iterative procedure using the current best estimate of λ_i,

$$\lambda_{i,n+1} = \frac{\hat{\mathbf{x}}_{n+1}\mathbf{M}\,\mathbf{x}_n}{\hat{\mathbf{x}}_{n+1}\mathbf{M}\,\hat{\mathbf{x}}_n},$$

to improve the modified stiffness matrix to be used in the following iterations,

$$\overline{\mathbf{K}} = \overline{\mathbf{K}} - \lambda_{i,n+1}\mathbf{M}$$

Much literature was dedicated to the problem of choosing the initial shifts so that all the eigenvectors can be computed sequentially without missing any of them.

Matrix Iteration with Shifts, 2

The shifted eigenproblem can be solved, e.g., by matrix iteration and the procedure will converge to the *smallest absolute value* shifted eigenvalue and to the associated eigenvector. After convergence is reached,

$$\boldsymbol{\psi}_i = \boldsymbol{\phi}_i, \qquad \omega_i^2 = \lambda_i + \mu.$$

The convergence of the method can be greatly enhanced if the shift μ is updated every few steps during the iterative procedure using the current best estimate of λ_i,

$$\lambda_{i,n+1} = \frac{\hat{\mathbf{x}}_{n+1} \mathbf{M}\, \mathbf{x}_n}{\hat{\mathbf{x}}_{n+1} \mathbf{M}\, \hat{\mathbf{x}}_n},$$

to improve the modified stiffness matrix to be used in the following iterations,

$$\overline{\mathbf{K}} = \overline{\mathbf{K}} - \lambda_{i,n+1}\mathbf{M}$$

Much literature was dedicated to the problem of choosing the initial shifts so that all the eigenvectors can be computed sequentially without missing any of them.

Rayleigh Quotient for Discrete Systems

The matrix iteration procedures are usually used in conjunction with methods derived from the Rayleigh Quotient method.

Rayleigh Quotient for Discrete Systems

The matrix iteration procedures are usually used in conjunction with methods derived from the Rayleigh Quotient method.
The Rayleigh Quotient method was introduced using distributed flexibilty systems and an assumed shape function, but we have seen also an example where the Rayleigh Quotient was computed for a discrete system using an assumed shape vector.

Rayleigh Quotient for Discrete Systems

The matrix iteration procedures are usually used in conjunction with methods derived from the Rayleigh Quotient method.
The Rayleigh Quotient method was introduced using distributed flexibilty systems and an assumed shape function, but we have seen also an example where the Rayleigh Quotient was computed for a discrete system using an assumed shape vector.
The procedure to be used for discrete systems can be summarized as

$$\mathbf{x}(t) = \boldsymbol{\phi} Z_0 \sin \omega t, \qquad \dot{\mathbf{x}}(t) = \omega \boldsymbol{\phi} Z_0 \cos \omega t,$$

$$2T_{\max} = \omega^2 \boldsymbol{\phi}^T \mathbf{M}\, \boldsymbol{\phi}, \qquad 2V_{\max} = \boldsymbol{\phi}^T \mathbf{K}\, \boldsymbol{\phi},$$

Rayleigh Quotient for Discrete Systems

The matrix iteration procedures are usually used in conjunction with methods derived from the Rayleigh Quotient method.
The Rayleigh Quotient method was introduced using distributed flexibilty systems and an assumed shape function, but we have seen also an example where the Rayleigh Quotient was computed for a discrete system using an assumed shape vector.
The procedure to be used for discrete systems can be summarized as

$$\mathbf{x}(t) = \boldsymbol{\phi} Z_0 \sin \omega t, \qquad \dot{\mathbf{x}}(t) = \omega \boldsymbol{\phi} Z_0 \cos \omega t,$$

$$2 T_{\text{max}} = \omega^2 \boldsymbol{\phi}^T \mathbf{M}\, \boldsymbol{\phi}, \qquad 2 V_{\text{max}} = \boldsymbol{\phi}^T \mathbf{K}\, \boldsymbol{\phi},$$

equating the maxima, we have

$$\omega^2 = \frac{\boldsymbol{\phi}^T \mathbf{K}\, \boldsymbol{\phi}}{\boldsymbol{\phi}^T \mathbf{M}\, \boldsymbol{\phi}} = \frac{k^\star}{m^\star}.$$

Take note that $\boldsymbol{\phi}$ is an assumed shape vector, not an eigenvector.

Ritz Coordinates

For a *N DOF* system, an *approximation* to a displacement vector $\mathbf{x}$ can be written in terms of a set of $M < N$ assumed shape, linearly independent vectors,

$$\boldsymbol{\phi}_i, \quad i = 1, \ldots, M < N$$

and a set of *Ritz coordinates* z_i, $i - 1, \ldots, M < N$:

$$\mathbf{x} = \sum_i \boldsymbol{\phi}_i z_i = \mathbf{\Phi}\, \mathbf{z}.$$

We say *approximation* because a linear combination of $M < N$ vectors cannot describe every point in a N-space.

Rayleigh Quotient in Ritz Coordinates

We can write the Rayleigh quotient as a function of the Ritz coordinates,

$$\omega^2(\mathbf{z}) = \frac{\mathbf{z}^T\mathbf{\Phi}^T\mathbf{K}\,\mathbf{\Phi}\mathbf{z}}{\mathbf{z}^T\boldsymbol{\phi}^T\mathbf{M}\,\boldsymbol{\phi}\mathbf{z}} = \frac{\overline{k}(\mathbf{z})}{\overline{m}(\mathbf{z})},$$

but this is not an explicit fuction for any modal frequency...

Rayleigh Quotient in Ritz Coordinates

We can write the Rayleigh quotient as a function of the Ritz coordinates,

$$\omega^2(\mathbf{z}) = \frac{\mathbf{z}^T \mathbf{\Phi}^T \mathbf{K}\,\mathbf{\Phi}\mathbf{z}}{\mathbf{z}^T \boldsymbol{\phi}^T \mathbf{M}\,\boldsymbol{\phi}\mathbf{z}} = \frac{\overline{k}(\mathbf{z})}{\overline{m}(\mathbf{z})},$$

but this is not an explicit fuction for any modal frequency...
On the other hand, we have seen that frequency estimates are always greater than true frequencies, so our best estimates are the the local minima of $\omega^2(\mathbf{z})$, or the points where all the derivatives of $\omega^2(\mathbf{z})$ with respect to z_i are zero:

$$\frac{\partial\omega^2(\mathbf{z})}{\partial z_j} = \frac{\overline{m}(\mathbf{z})\dfrac{\partial\overline{k}(\mathbf{z})}{\partial z_i} - \overline{k}(\mathbf{z})\dfrac{\partial\overline{m}(\mathbf{z})}{\partial z_i}}{(\overline{m}(\mathbf{z}))^2} = 0, \qquad \text{for } i = 1, \ldots, M < N$$

Reduced Eigenproblem

Observing that

$$\overline{k}(\mathbf{z}) = \omega^2(\mathbf{z})\overline{m}(\mathbf{z})$$

we can substitute into and simplify the preceding equation,

$$\frac{\partial\overline{k}(\mathbf{z})}{\partial z_i} - \omega^2(\mathbf{z})\frac{\partial\overline{m}(\mathbf{z})}{\partial z_i} = 0, \qquad \text{for } i = 1, \ldots, M < N$$

Reduced Eigenproblem

Observing that

$$\overline{k}(\mathbf{z}) = \omega^2(\mathbf{z})\overline{m}(\mathbf{z})$$

we can substitute into and simplify the preceding equation,

$$\frac{\partial\overline{k}(\mathbf{z})}{\partial z_i} - \omega^2(\mathbf{z})\frac{\partial\overline{m}(\mathbf{z})}{\partial z_i} = 0, \qquad \text{for } i = 1, \ldots, M < N$$

With the positions

$$\overline{\mathbf{K}} = \mathbf{\Phi}^T\mathbf{K}\,\mathbf{\Phi}, \qquad \overline{,\mathbf{M}} = \mathbf{\Phi}^T\mathbf{M}\,\mathbf{\Phi}$$

we have

$$\overline{k}(\mathbf{z}) = \mathbf{z}^T\overline{\mathbf{K}}\mathbf{z} = \sum_i\sum_j \overline{k}_{ij}z_jz_i,$$

and

$$\frac{\partial\overline{k}(\mathbf{z})}{\partial z_i} = 2\sum_j \overline{k}_{ij}z_j = 2\overline{\mathbf{K}}\mathbf{z}, \text{ and, analogously, } \frac{\partial\overline{m}(\mathbf{z})}{\partial z_i} = 2\overline{\mathbf{M}}\mathbf{z}.$$

Reduced Eigenproblem

Observing that

$$\overline{k}(\mathbf{z}) = \omega^2(\mathbf{z})\overline{m}(\mathbf{z})$$

we can substitute into and simplify the preceding equation,

$$\frac{\partial\overline{k}(\mathbf{z})}{\partial z_i} - \omega^2(\mathbf{z})\frac{\partial\overline{m}(\mathbf{z})}{\partial z_i} = 0, \qquad \text{for } i = 1, \ldots, M < N$$

With the positions

$$\overline{\mathbf{K}} = \mathbf{\Phi}^T\mathbf{K}\,\mathbf{\Phi}, \qquad \overline{,\mathbf{M}} = \mathbf{\Phi}^T\mathbf{M}\,\mathbf{\Phi}$$

we have

$$\overline{k}(\mathbf{z}) = \mathbf{z}^T\overline{\mathbf{K}}\mathbf{z} = \sum_i\sum_j \overline{k}_{ij}z_j z_i,$$

and

$$\frac{\partial\overline{k}(\mathbf{z})}{\partial z_i} = 2\sum_j \overline{k}_{ij}z_j = 2\overline{\mathbf{K}}\mathbf{z}, \text{ and, analogously, } \frac{\partial\overline{m}(\mathbf{z})}{\partial z_i} = 2\overline{\mathbf{M}}\mathbf{z}.$$

Substituting these results in $\frac{\partial\overline{k}(\mathbf{z})}{\partial z_i} - \omega^2(\mathbf{z})\frac{\partial\overline{m}(\mathbf{z})}{\partial z_i} = 0$ we can write a *new homogeneous system* in the Ritz coordinates, whose non trivial solutions are the solutions of a *reduced eigenvector problem* in the M *DOF* Ritz coordinates space, with reduced $M \times M$ matrices:

$$\overline{\mathbf{K}}\,\mathbf{z} - \omega^2\overline{\mathbf{M}}\,\mathbf{z} = \mathbf{0}.$$

Modal Superposition?

After solving the reduced eigenproblem, we have a set of M eigenvalues $\overline{\omega}_i^2$ and a corresponding set of M eigenvectors $\bar{\mathbf{z}}_i$. What is the relation between these results and the eigenpairs of the original problem?
The $\overline{\omega}_i^2$ clearly are approximations from above to the real eigenvalues, and if we write $\overline{\boldsymbol{\psi}}_i = \boldsymbol{\Phi}\bar{\mathbf{z}}_i$ we see that, being

$$\overline{\boldsymbol{\psi}}_i^T \mathbf{M} \overline{\boldsymbol{\psi}}_j = \bar{\mathbf{z}}_i^T \underbrace{\boldsymbol{\Phi}^T \mathbf{M} \boldsymbol{\Phi}}_{\overline{\mathbf{M}}} \bar{\mathbf{z}}_j = \overline{M}_i \delta_{ij},$$

the approximated eigenvectors $\overline{\boldsymbol{\psi}}_i$ are orthogonal with respect to the structural matrices and can be used in ordinary modal superposition techniques.

A Last Question

One last question: how many $\overline{\omega}_i^2$ and $\overline{\boldsymbol{\psi}}_i$ are *effective* approximations to the true eigenpairs? Experience tells that an effective approximation is to be expected for the first $M/2$ eigenthings.

Block Matrix Iteration

If we collect all the eigenvalues into a diagonal matrix $\mathbf{\Lambda}$, we can write the following equation,

$$\mathbf{K}\,\mathbf{\Psi} = \mathbf{M}\,\mathbf{\Psi}\,\mathbf{\Lambda}$$

where every matrix is a square, $N \times N$ matrix.
The *Subspace Iteration* method uses a reduced set of trials vectors, packed in $N \times M$ matrix $\mathbf{\Phi}_0$ and applies the procedure of matrix iteration to the whole set of trial vectors at once:

$$\hat{\mathbf{\Phi}}_1 = \mathbf{K}^{-1}\mathbf{M}\,\mathbf{\Phi}_0.$$

We used, again, the hat notation to visualize that the iterated vectors are not normalized by the application of the unknown $\mathbf{\Lambda}$.

Block Matrix Iteration

If we collect all the eigenvalues into a diagonal matrix $\mathbf{\Lambda}$, we can write the following equation,

$$\mathbf{K}\,\mathbf{\Psi} = \mathbf{M}\,\mathbf{\Psi}\,\mathbf{\Lambda}$$

where every matrix is a square, $N \times N$ matrix.
The *Subspace Iteration* method uses a reduced set of trials vectors, packed in $N \times M$ matrix $\mathbf{\Phi}_0$ and applies the procedure of matrix iteration to the whole set of trial vectors at once:

$$\hat{\mathbf{\Phi}}_1 = \mathbf{K}^{-1}\mathbf{M}\,\mathbf{\Phi}_0.$$

We used, again, the hat notation to visualize that the iterated vectors are not normalized by the application of the unknown $\mathbf{\Lambda}$.
Should we proceed naively down this road, though, all the columns in $\mathbf{\Phi}_n$ would converge to the first eigenvector, subspace iteration being only an expensive manner of applying matrix iteration without sweeps or shifts...

Subspace Iteration

Different options that comes to mind:

1. force all step n non-normalized vectors to be orthogonal with respect to $\mathbf{M}$, difficult, essentially we have to solve an eigenvalue problem…
2. use the step n non-normalized vectors as a reduced base for the Rayleigh-Ritz procedure, solve an eigenvalue problem

$$\overline{\mathbf{K}}_n = \hat{\mathbf{\Phi}}_n^T \mathbf{K}\, \hat{\mathbf{\Phi}}_n = \hat{\mathbf{\Phi}}_n^T \mathbf{M}\, \mathbf{\Phi}_{n-1}$$

$$\overline{\mathbf{M}}_n = \hat{\mathbf{\Phi}}_n^T \mathbf{M}\, \hat{\mathbf{\Phi}}_n$$

$$\overline{\mathbf{K}}_n\, \overline{\mathbf{Z}}_n = \overline{\mathbf{M}}_n\, \overline{\mathbf{Z}}_n \overline{\mathbf{\Lambda}}_n$$

whose outcome $\overline{\mathbf{\Lambda}}_n$, $\overline{\mathbf{Z}}_n$ *is* correlated to the structural eigenvalues, and use the normalized $\mathbf{Z}_n$ eigenvectors as the normalized, un-hatted $\mathbf{\Phi}_n$.

Subspace Iteration, 2

The second procedure is exactly what we want: we use $\overline{\mathbf{Z}}$ to start an iteration that will lead to a new set of base vectors that, being computed from the equation of dynamic equilibrium, will be a *better* base for the successive estimation of the eigenvectors, a new *subspace* where the eigenvectors can be more closely approximated.

Final Considerations

The procedure converges very fast and with excellent approximation to a number of eigenvalues and eigenvector p, $p = M - q$ where q is the number of required *guard* eigenpairs.
Experience shows that we can safely use $q = \min\{p, 8\}$.