Como resolver o QFEMUP usando otimização

Marina Andretta

ICMC-USP

2 de agosto de 2016

Baseado no artigo M. Andretta, E. G. Birgin and M. Raydan, “An inner-outer nonlinear programming approach for constrained quadratic matrix model updating”, *Mechanical Systems and Signal Processing*, 66-67, pp. 78–88, 2016.

Problema de atualização de modelo de elementos finitos quadrático

O Problema de Atualização de Modelo de Elementos Finitos Quadrático (*Quadratic Finite Element Model Updating Problem* - QFEMUP) consiste em atualizar um modelo gerado por elementos finitos de uma estrutura com vibração da forma

$$M\ddot{x}(t) + D\dot{x}(t) + Kx(t) = 0, \tag{1}$$

com

- M, D e K matrizes reais $n \times n$ conhecidas como massa, amortecimento e rigidez, respectivamente;
- $\dot{x}(t)$ e $\ddot{x}(t)$ denotam a primeira e segunda derivadas do vetor dependente do tempo $x(t)$.

QFEMUP

Os autovalores quadráticos associados a

$$Q(\lambda) = \lambda^2 M + \lambda D + K \tag{2}$$

são relacionados às frequências naturais e os autovetores quadráticos são os modos de vibração do sistema (1).

$Q(\lambda)$ possui $2n$ autovalores e $2n$ autovetores.

Aplicações do QFEMUP

A dinâmica do sistema é modelada por estes autovalores e autovetores. Por exemplo, é sabido que a estabilidade de sistemas de vibração é determinada pela natureza de algumas frequências naturais dominantes.

Também é sabido que às vezes estruturas de vibração sofrem vibrações perigosas, chamadas de *ressonância*, quando uma frequência natural se aproxima ou fica igual à frequência de uma força externa, como um terremoto, pesos de corpos humanos, entre outros. O colapso de várias estruturas como prédios, pontes, asas de aviões e turbinas é atribuído à ressonância.

QFEMUP

As matrizes M, D e K geralmente são muito grandes e esparsas, mas possuem algumas estruturas, como

- M é simétrica e definida positiva ($M = M^T > 0$) e geralmente diagonal;
- D e K são simétricas ($D = D^T$ e $K = K^T$) e a matriz de rigidez K geralmente é tridiagonal e semidefinida positiva.

Definição do QFEMUP

Considere $Q(\lambda)$ dado por (2), com $M, D, K \in \mathbb{R}^{n \times n}$ matrizes dadas tais que M é simétrica definida positiva e D e K são simétricas.

Sejam $1 \leq p < 2n$, $\lambda_i \in \mathbb{C}$ e $x_i \in \mathbb{C}^n$ tais que (λ_i, x_i) para $i = 1, \ldots, p$ são os auto-pares desejados.

O objetivo é encontrar matrizes $\tilde{D}, \tilde{K} \in \mathbb{R}^{n \times n}$ tais que (λ_i, x_i) são auto-pares de $\tilde{Q}(\lambda)$ dado por

$$\tilde{Q}(\lambda) = \lambda^2 M + \lambda \tilde{D} + \tilde{K}. \tag{3}$$

Ou seja,

$$(\lambda_i^2 M + \lambda_i \tilde{D} + \tilde{K}) x_i = 0, \;\; i = 1, \ldots, p. \tag{4}$$

Definição do QFEMUP

Além disso,

- as matrizes atualizadas $\tilde{D} = (\tilde{d}_{ij})$ e $\tilde{K} = (\tilde{k}_{ij})$ devem ser simétricas e devem preservar o padrão de esparsidade de $D = (d_{ij})$ e $K = (k_{ij})$, respectivamente;
- $\tilde{D}$ e $\tilde{K}$ devem estar o mais próximo possível de D e K, respectivamente;
- $\tilde{Q}(\lambda)$ não pode ter autovalores "ruins".

Como resolver o QFEMUP

Defina os conjuntos de índices I_D e I_K como os índices dos elementos não-nulos de D e K, respectivamente.

Considere, por exemplo,

$$D = \begin{pmatrix} d_{11} & d_{12} & 0 & 0 \\ d_{12} & d_{22} & d_{23} & 0 \\ 0 & d_{23} & d_{33} & 0 \\ 0 & 0 & 0 & d_{44} \end{pmatrix}.$$

Como resolver o QFEMUP

Defina os conjuntos de índices I_D e I_K como os índices dos elementos não-nulos de D e K, respectivamente.

Considere, por exemplo,

$$D = \begin{pmatrix} d_{11} & d_{12} & 0 & 0 \\ d_{12} & d_{22} & d_{23} & 0 \\ 0 & d_{23} & d_{33} & 0 \\ 0 & 0 & 0 & d_{44} \end{pmatrix}.$$

Então I_D é definido como

$$I_D = \{(1,1), (1,2), (2,2), (2,3), (3,3), (3,4), (4,4)\}.$$

Como resolver o QFEMUP

A formulação de programação matemática para o QFEMUP é dada por

$$\begin{array}{ll} \text{Minimizar} & \|\tilde{D} - D\|_F^2 + \|\tilde{K} - K\|_F^2 \\ \text{sujeita a} & M X \Lambda^2 + \tilde{D} X \Lambda + \tilde{K} X = 0, \\ & -U \leq \tilde{d}_{ij} \leq U, \quad \text{para todo} \quad (i,j) \in I_D, \\ & -U \leq \tilde{k}_{ij} \leq U, \quad \text{para todo} \quad (i,j) \in I_K, \end{array}$$

com $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_p) \in \mathbb{C}^{p \times p}$ e $X \in \mathbb{C}^{n \times p}$ tem colunas $x_1, \ldots, x_p$.

Como resolver o QFEMUP

Isso significa que a simetria e o padrão de esparsidade de $\tilde{D}$ e $\tilde{K}$ são garantidos pela escolha das variáveis, sem necessitar que restrições adicionais sejam colocadas no modelo.

Além disso, devido a esta escolha, o número de variáveis é diminuido.

Como resolver o QFEMUP

Note que $\lambda_i \in \mathbb{C}$ e $x_i \in \mathbb{C}^n$ para $i = 1, \ldots, p$.

Portanto, para usar métodos clássicos de programação não-linear, que lidam apenas com problemas no espaço dos números reais (e não complexos), é adequado reescrever as restrições como

$$\begin{aligned} \Re(MX\Lambda^2 + \tilde{D}X\Lambda + \tilde{K}X) &= 0, \\ \Im(MX\Lambda^2 + \tilde{D}X\Lambda + \tilde{K}X) &= 0, \end{aligned}$$

com $\Re(c)$ e $\Im(c)$ a parte real a e a parte imaginária b de um número complexo $c = a + bi$.

Então, temos o modelo

$$
\begin{array}{lrcl}
\text{Minimizar} & \|\tilde{D}-D\|_F^2+\|\tilde{K}-K\|_F^2 & & \\
\text{sujeita a} & \Re(MX\Lambda^2+\tilde{D}X\Lambda+\tilde{K}X) & = & 0, \\
 & \Im(MX\Lambda^2+\tilde{D}X\Lambda+\tilde{K}X) & = & 0, \\
 & -U\leq\tilde{d}_{ij}\leq U,\ \text{para todo } (i,j) & \in & I_D, \\
 & -U\leq\tilde{k}_{ij}\leq U,\ \text{para todo } (i,j) & \in & I_K.
\end{array} \tag{5}
$$

Ao resolver (5), podemos obter autovalores “ruins” em $\tilde{Q}(\lambda)$.

Vamos nos focar na situação frequente em que $\hat{\lambda} \in \mathbb{R}$ é dado e existe uma restrição que diz que todos os autovalores de $\tilde{Q}(\lambda)$ devem ter suas partes reais menores ou iguais a $\hat{\lambda}$.

Como resolver o QFEMUP

Suponha que $(\mu_1, y_1), \ldots, (\mu_{2n}, y_{2n})$ são os auto-pares (desconhecidos) de $Q(\lambda)$ e que $1 \leq p < 2n$ e $(\lambda_1, x_1), \ldots, (\lambda_p, x_p)$ são os (conhecidos) auto-pares desejados.

A seguir, também supomos que

- os p auto-pares desejados são tais que $\Re(\lambda_i) \leq \hat{\lambda}$, para $i = 1, \ldots, p$;
- os p auto-pares que serão substituídos são tais que $\Re(\mu_i) > \hat{\lambda}$, para $i = 1, \ldots, p$;
- os $2n - p$ auto-pares restantes são tais que $\Re(\mu_i) \leq \hat{\lambda}$, para $i = p+1, \ldots, 2n$.

Como resolver o QFEMUP

Ao resolver o problema de programação matemática (5), encontramos matrizes $\tilde{D}$ e $\tilde{K}$ tais que o $\tilde{Q}(\lambda)$ tem os auto-pares desejados $(\lambda_1, x_1), \ldots, (\lambda_p, x_p)$, para $i = 1, \ldots, p$, isto é, tais que vale (4).

Então, um autovalor $\tau \in \mathbb{C}$ com maior parte real de (3) é calculado.

Se sua parte real $\Re(\tau)$ for maior do que $\hat{\lambda}$, então

- um autovetor (normalizado) associado $\tilde{x}$ é calculado;
- uma restrição, que depende de $\tilde{x}$, é adicionada ao modelo (5) com a intenção de eliminar o autovalor "ruim" detectado;
- um novo problema de programação não-linear é resolvido.

Esse processo é repetido até que $\tilde{Q}(\lambda)$ não tenha autovalores "ruins".

Como resolver o QFEMUP

Da condição de Galerkin, segue que τ deve ser uma das duas soluções da equação quadrática

$$q(\theta) = \theta^2(\tilde{x}^* M \tilde{x}) + \theta(\tilde{x}^* \tilde{D} \tilde{x}) + (\tilde{x}^* \tilde{K} \tilde{x}),$$

que são dadas por

$$\theta_1 = \frac{-(\tilde{x}^* \tilde{D} \tilde{x}) + \sqrt{(\tilde{x}^* \tilde{D} \tilde{x})^2 - 4(\tilde{x}^* M \tilde{x})(\tilde{x}^* \tilde{K} \tilde{x})}}{2(\tilde{x}^* M \tilde{x})}$$

e

$$\theta_2 = \frac{-(\tilde{x}^* \tilde{D} \tilde{x}) - \sqrt{(\tilde{x}^* \tilde{D} \tilde{x})^2 - 4(\tilde{x}^* M \tilde{x})(\tilde{x}^* \tilde{K} \tilde{x})}}{2(\tilde{x}^* M \tilde{x})}.$$

Como resolver o QFEMUP

Se $\tau = \theta_1$ então a nova restrição a ser acrescentada ao modelo é dada por

$$\Re\left(\frac{-(\tilde{x}^*\tilde{D}\tilde{x}) + \sqrt{(\tilde{x}^*\tilde{D}\tilde{x})^2 - 4(\tilde{x}^*M\tilde{x})(\tilde{x}^*\tilde{K}\tilde{x})}}{2(\tilde{x}^*M\tilde{x})}\right) \leq \hat{\lambda} - \varepsilon, \tag{6}$$

com $\varepsilon > 0$ uma pequena tolerância, para que a restrição seja um pouco mais forte do que o que realmente queremos.

Como resolver o QFEMUP

Caso contrário, temos que $\tau = \theta_2$ e então a nova restrição a ser adicionada é dada por

$$\Re\left(\frac{-(\tilde{x}^* \tilde{D}\tilde{x}) - \sqrt{(\tilde{x}^* \tilde{D}\tilde{x})^2 - 4(\tilde{x}^* M\tilde{x})(\tilde{x}^* \tilde{K}\tilde{x})}}{2(\tilde{x}^* M\tilde{x})}\right) \leq \hat{\lambda} - \varepsilon. \qquad (7)$$

Lembre-se que quando resolvemos o problema (5) com as restrições adicionais, o vetor $\tilde{x}$ e os escalares $\hat{\lambda}$ e ε são dados, então as variáveis do problema continuam as mesmas.

Algoritmo

Algoritmo 1. Sejam $M, D, K \in \mathbb{R}^{n\times n}$ matrizes dadas tais que M é definida positiva e $D = (d_{ij})$ e $K = (k_{ij})$ são simétricas. Seja $1 \leq p < 2n$ e sejam $(\lambda_i, x_i) \in \mathbb{C} \times \mathbb{C}^n$, para $i = 1, \dots, p$, os auto-pares quadráticos desejados. Seja $\hat{\lambda} \in \mathbb{R}$ o parâmetro que descreve a região proibida para os autovalores quadráticos de $\tilde{Q}(\lambda)$. Seja $U > 0$ um número real grande e seja $\varepsilon > 0$ uma tolerância pequena dada. Faça $\kappa \leftarrow 0$.

Passo 1. *Define os padrões de esparsidade*
Calcule

$$I_D = \{(i,j) \mid 1 \leq i \leq j \leq n \text{ tais que } d_{ij} \neq 0\}$$

e

$$I_K = \{(i,j) \mid 1 \leq i \leq j \leq n \text{ tais que } k_{ij} \neq 0\}.$$

Passo 2. *Passo de otimização*

Resolva o problema de programação não-linear com $|I_D| + |I_K|$ variáveis dado por

$$
\begin{array}{lrcl}
\text{Minimizar} & \|\tilde{D} - D\|_F^2 + \|\tilde{K} - K\|_F^2 & & \\
\text{sujeita a} & \Re(MX\Lambda^2 + \tilde{D}X\Lambda + \tilde{K}X) & = & 0, \\
 & \Im(MX\Lambda^2 + \tilde{D}X\Lambda + \tilde{K}X) & = & 0, \\
 & -U \leq \tilde{d}_{ij} \leq U, \text{ para todo } (i,j) & \in & I_D, \\
 & -U \leq \tilde{k}_{ij} \leq U, \text{ para todo } (i,j) & \in & I_K,
\end{array}
\tag{8}
$$

Algoritmo

mais

$$\Re\left(\frac{-(u_j^*\tilde{D}u_j)+s_j\sqrt{(u_j^*\tilde{D}u_j)^2-4(u_j^*Mu_j)(u_j^*\tilde{K}u_j)}}{2(u_j^*Mu_j)}\right)\leq\hat{\lambda}-\varepsilon, \qquad (9)$$

para todo $j=0,\ldots,\kappa-1$, com $\Lambda=\text{diag}(\lambda_1,\ldots,\lambda_p)\in\mathbb{C}^{p\times p}$ e $X\in\mathbb{C}^{n\times p}$ com colunas $x_1,\ldots,x_p$, para encontrar as matrizes $\tilde{D}_\kappa$ e $\tilde{K}_\kappa$ tais que

$$(\lambda_i^2M+\lambda_i\tilde{D}_\kappa+\tilde{K}_\kappa)x_i=0,\ \ i=1,\ldots,p.$$

Algoritmo

Passo 3. *Verifica se há autovalores quadráticos "ruins"*

Passo 3.1. Calcule o autovalor quadrático τ com maior parte real de

$$\tilde{Q}(\lambda) = \lambda^2 M + \lambda \tilde{D}_\kappa + \tilde{K}_\kappa.$$

Passo 3.2. Se $\Re(\tau) \leq \hat{\lambda}$ então pare e devolva $\tilde{D}_\kappa$ e $\tilde{K}_\kappa$.

Algoritmo

Passo 4. *Adicione novo corte e itere*

Passo 4.1. Calcule o autovetor quadrático u_κ associado a τ (tal que $\|u_\kappa\|_2 = 1$).

Passo 4.2. Se

$$\tau = \frac{-(u_\kappa^* \tilde{D}_\kappa u_\kappa) + \sqrt{(u_\kappa^* \tilde{D}_\kappa u_\kappa)^2 - 4(u_\kappa^* M u_\kappa)(u_\kappa^* \tilde{K}_\kappa u_\kappa)}}{2(u_\kappa^* M u_\kappa)}$$

então defina $s_\kappa = 1$. Caso contrário, se

$$\tau = \frac{-(u_\kappa^* \tilde{D}_\kappa u_\kappa) - \sqrt{(u_\kappa{}^* \tilde{D}_\kappa u_\kappa)^2 - 4(u_\kappa^* M u_\kappa)(u_\kappa^* \tilde{K}_\kappa u_\kappa)}}{2(u_\kappa^* M u_\kappa)}$$

então defina $s_\kappa = -1$.

Passo 4.3. Defina $\kappa \leftarrow \kappa + 1$ e volte para o Passo 2.

Remark

Note que, a cada iteração $\kappa \geq 1$, todos os cortes em (9) de iterações anteriores são mantidos como restrições no problema de programação não-linear a ser resolvido.

Portanto, a região viável é monotonicamente reduzida quando κ aumenta.

Teorema de convergência

Theorem

Sejam $\hat{\lambda} \in \mathbb{R}$ e $\varepsilon > 0$ dados. Se o Algoritmo 1 é aplicado para resolver o problema de otimização (5) com a restrição adicional de que todos os autovalores de $\tilde{Q}(\lambda)$ devem ter suas partes reais menores ou iguais a $\hat{\lambda}$, então ele termina em um número finito de iterações.