Frequency isolation algorithms for hyperbolic quadratic eigenvalue problems

Julio Moro

Departamento de Matemáticas
Universidad Carlos III de Madrid
Madrid (Spain)

Joint work with Juan Egaña and Fernando de Terán

Berlin-Manchester Workshop '07

Manchester, March 22nd '07

Outline

Hyperbolic QEPs

- Consider **quadratic e-value problem (QEP)**

$$Q(\lambda)u = (\lambda^2 M + \lambda C + K)u = 0.$$

with M, C, K symmetric, positive definite $n \times n$.

Hyperbolic QEPs

- Consider **quadratic e-value problem (QEP)**

$$Q(\lambda)u = (\lambda^2 M + \lambda C + K)u = 0.$$

with M, C, K symmetric, positive definite $n \times n$.

- QEP is hyperbolic if

$$(x^T Cx)^2 - 4(x^T Mx)(x^T Kx) > 0$$

for all $x \neq 0$.

Hyperbolic QEPs

- Consider **quadratic e-value problem (QEP)**

$$Q(\lambda)u = (\lambda^2 M + \lambda C + K)u = 0.$$

with M, C, K symmetric, positive definite $n \times n$.

- QEP is hyperbolic if

$$(x^T Cx)^2 - 4(x^T Mx)(x^T Kx) > 0$$

for all $x \neq 0$.

- Nice properties of hyperbolic QEPs:
  - $2n$ real and semisimple e-values
  - e-values can be obtained by bisection

The frequency isolation problem

- When e-values of

$$Q(\lambda) = \lambda^2 M + \lambda C + K$$

fall in certain regions, vibrational system experiences dangerous vibrations (resonance). M, C, K should be chosen in such a way that these spectral regions are avoided

The frequency isolation problem

- When e-values of

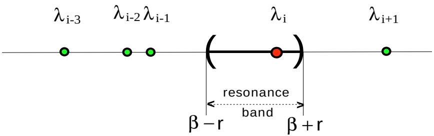
$$Q(\lambda) = \lambda^2 M + \lambda C + K$$

  fall in certain regions, vibrational system experiences dangerous vibrations (resonance). M, C, K should be chosen in such a way that these spectral regions are avoided

- Typically, dangerous region modelled as resonance band

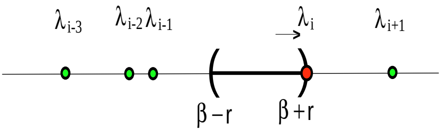
$$I_r = (\beta - r, \beta + r) \subset \mathbb{R},$$

  to be avoided by e-values λ_j of $Q(\lambda)$.



The frequency isolation problem

- When e-values of
$$Q(\lambda) = \lambda^2 M + \lambda C + K$$
fall in certain regions, vibrational system experiences dangerous vibrations (resonance). M, C, K should be chosen in such a way that these spectral regions are avoided
- Typically, dangerous region modelled as resonance band
$$I_r = (\beta - r, \beta + r) \subset \mathbb{R},$$
to be avoided by e-values λ_j of $Q(\lambda)$. Once no eigenvalue is in I_r, we say that spectrum has been isolated



The frequency isolation problem

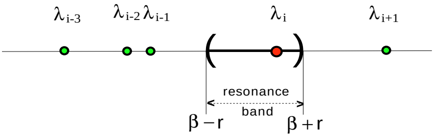
- When e-values of
$$Q(\lambda) = \lambda^2 M + \lambda C + K$$
fall in certain regions, vibrational system experiences dangerous vibrations (resonance). M, C, K should be chosen in such a way that these spectral regions are avoided
- Typically, dangerous region modelled as resonance band
$$I_r = (\beta - r, \beta + r) \subset \mathbb{R},$$
to be avoided by e-values λ_j of $Q(\lambda)$. Once no eigenvalue is in I_r, we say that spectrum has been isolated

Frequency isolation problem: Given a resonance band $I_r = (\beta - r, \beta + r)$ and a vibrational system (M, C, K) with some eigenvalue in $(\beta - r, \beta + r)$, **redesign the system** in such a way that the new system (M^*, C^*, K^*)

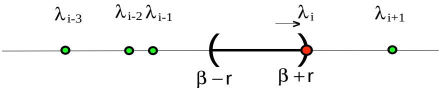
- has **no eigenvalue in the resonance band**, and
- **is close to** (M, C, K) in some sense.

Frequency isolation algorithms: the undamped case

Egaña, Kuhl & Santos '02: Tridiagonal QEP with $C = 0$ (no damping). Given initial, resonant spectrum



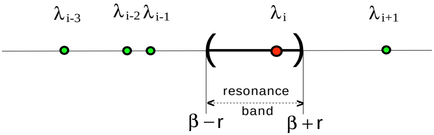
fix target spectrum, e.g.



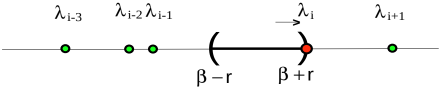
and reconstruct matrices M^* and K^* with that spectrum, i.e., solve tridiagonal inverse eigenvalue problem

Frequency isolation algorithms: the undamped case

Egaña, Kuhl & Santos '02: Tridiagonal QEP with $C = 0$ (no damping). Given initial, resonant spectrum



fix target spectrum, e.g.

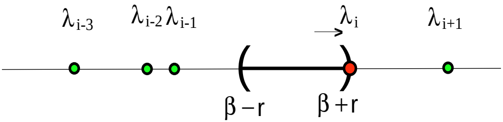


and reconstruct matrices M^* and K^* with that spectrum, i.e., solve tridiagonal inverse eigenvalue problem

Very expensive: cost well over $O(n^4)$ + quite restrictive conditions.

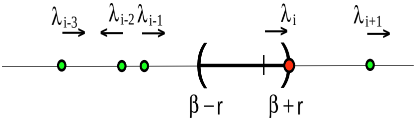
How to improve it?

Fixing a target spectrum is unnatural, since “good" eigenvalues are not allowed to change at all



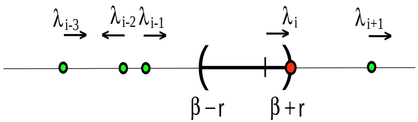
How to improve it?

Should allow “good" eigenvalues to move around (maybe not too much)



How to improve it?

Should allow “good" eigenvalues to move around (maybe not too much)



Idea of new algorithm: Identify a direction in (M, C, K) space along which:

- variation of “bad" eigenvalues is **maximal**, and
- variation of “good" eigenvalues is **minimal**.

Then, modify (M, C, K) along this direction up to isolation.

Outline

A simple situation

As an example, consider hyperbolic QEP

$$Q(\lambda) = \lambda^2 M + \lambda C + K$$

with

$$M = \mathrm{diag}(m) = \mathrm{diag}(m_1, \ldots, m_n), \qquad m_i > 0,$$
$$C = \mathrm{diag}(c) = \mathrm{diag}(c_1, \ldots, c_n), \qquad c_i > 0,$$

$$K = \begin{bmatrix} k_1 + k_2 & -k_2 & & & \\ -k_2 & k_2 + k_3 & -k_3 & & \\ & \ddots & \ddots & \ddots & \\ & & -k_{n-1} & k_{n-1} + k_n & -k_n \\ & & & -k_n & k_n \end{bmatrix}, \qquad k_i > 0$$

A simple situation

As an example, consider hyperbolic QEP

$$Q(\lambda) = \lambda^2 M + \lambda C + K$$

with

$M = \mathrm{diag}(m) = \mathrm{diag}(m_1, \ldots, m_n), \qquad m_i > 0,$

$C = \mathrm{diag}(c) = \mathrm{diag}(c_1, \ldots, c_n), \qquad c_i > 0,$

$$K = \begin{bmatrix} k_1 + k_2 & -k_2 & & & \\ -k_2 & k_2 + k_3 & -k_3 & & \\ & \ddots & \ddots & \ddots & \\ & & -k_{n-1} & k_{n-1} + k_n & -k_n \\ & & & -k_n & k_n \end{bmatrix}, \qquad k_i > 0$$

$\lambda_j = \lambda_j(m,c,k) \quad$ for $\quad (m,c,k) = (m_1, \ldots, m_n, c_1, \ldots, c_n, k_1, \ldots, k_n) \in \mathbb{R}^{3n}.$

Thus,

Work in parameter space (m,c,k) instead of in matrix space (M,C,K)

How to choose directions in (m, c, k) space?

- Consider initial configuration $(m, c, k) = (m_1, \ldots, m_n, c_1, \ldots, c_n, k_1, \ldots, k_n) \in \mathbb{R}^{3n}$, and call

  good e-values those s.t. $\lambda_j(m, c, k) \notin (\beta - r, \beta + r)$

  bad e-values those s.t. $\lambda_j(m, c, k) \in (\beta - r, \beta + r)$

How to choose directions in (m, c, k) space?

- Consider initial configuration $(m, c, k) = (m_1, \ldots, m_n, c_1, \ldots, c_n, k_1, \ldots, k_n) \in \mathbb{R}^{3n}$, and call

  good e-values those s.t. $\lambda_j(m, c, k) \notin (\beta - r, \beta + r)$

  bad e-values those s.t. $\lambda_j(m, c, k) \in (\beta - r, \beta + r)$

- Want to move bad e-values out of resonance band as fast as possible, without any good e-value entering the band.

How to choose directions in (m,c,k) space?

- Consider initial configuration $(m,c,k) = (m_1,\dots,m_n,c_1,\dots,c_n,k_1,\dots,k_n) \in \mathbb{R}^{3n}$, and call

  good e-values those s.t. $\lambda_j(m,c,k) \notin (\beta - r, \beta + r)$

  bad e-values those s.t. $\lambda_j(m,c,k) \in (\beta - r, \beta + r)$

- Want to move bad e-values out of resonance band as fast as possible, without any good e-value entering the band.
- Assume we are close to isolation. Let $\Delta = (\delta m, \delta c, \delta k) \in \mathbb{R}^{3n}$, and write perturbed e-values as

$$\lambda_j((m,c,k)+\Delta) = \lambda_j(m,c,k) + < \nabla \lambda_j(m,c,k), \Delta > + \dots$$

How to choose directions in (m, c, k) space?

- Consider initial configuration $(m, c, k) = (m_1, \dots, m_n, c_1, \dots, c_n, k_1, \dots, k_n) \in \mathbb{R}^{3n}$, and call
  good e-values those s.t. $\lambda_j(m, c, k) \notin (\beta - r, \beta + r)$
  bad e-values those s.t. $\lambda_j(m, c, k) \in (\beta - r, \beta + r)$
- Want to move bad e-values out of resonance band as fast as possible, without any good e-value entering the band.
- Assume we are close to isolation. Let $\Delta = (\delta m, \delta c, \delta k) \in \mathbb{R}^{3n}$, and write perturbed e-values as

$$\lambda_j((m, c, k) + \Delta) = \lambda_j(m, c, k) + < \nabla \lambda_j(m, c, k), \Delta > + \dots$$

- First order term $< \nabla \lambda_j(m, c, k), \Delta >$ should be:

  1) as large as possible for bad λ_j

  2) as small as possible for good λ_j

How to choose directions in (m, c, k) space?

- Consider initial configuration
  $(m, c, k) = (m_1, \dots, m_n, c_1, \dots, c_n, k_1, \dots, k_n) \in \mathbb{R}^{3n}$, and call
  good e-values those s.t. $\lambda_j(m, c, k) \notin (\beta - r, \beta + r)$
  bad e-values those s.t. $\lambda_j(m, c, k) \in (\beta - r, \beta + r)$
- Want to move bad e-values out of resonance band as fast as possible, without any good e-value entering the band.
- Assume we are close to isolation. Let $\Delta = (\delta m, \delta c, \delta k) \in \mathbb{R}^{3n}$, and write perturbed e-values as

$$\lambda_j((m, c, k) + \Delta) = \lambda_j(m, c, k) + < \nabla \lambda_j(m, c, k), \Delta > + \dots$$

- First order term $< \nabla \lambda_j(m, c, k), \Delta >$ should be:

  1) as large as possible for bad λ_j

  2) as small as possible for good λ_j $\rightarrow$ zero for good λ_j **i.e.,**

$$\Delta \perp \nabla \lambda_j(m, c, k) \quad \text{for all} \quad \text{good } \lambda_j.$$

The basic isolation algorithm

Since

$$\Delta = (\delta m, \delta c, \delta k) \perp \nabla \lambda_j(m,c,k) \qquad \text{for all} \quad \text{good} \ \ \lambda_j,$$

denote

$$V_{good}^{\perp} = \{ w \in \mathbb{R}^{3n} : < \nabla \lambda_j(m,c,k), w >= 0 \quad \text{for all} \ \ \text{good} \ \ \lambda_j \}.$$

Basic isolation algorithm organized in two stages:

The basic isolation algorithm

Since

$$\Delta = (\delta m, \delta c, \delta k) \perp \nabla \lambda_j(m,c,k) \qquad \text{for all} \quad \text{good} \quad \lambda_j,$$

denote

$$V_{good}^{\perp} = \{ w \in \mathbb{R}^{3n} : < \nabla \lambda_j(m,c,k), w >= 0 \quad \text{for all} \;\; \text{good} \;\; \lambda_j \}.$$

Basic isolation algorithm organized in two stages:

1) Find a direction $w_{\max} \in V_{good}^{\perp}$ s.t. velocities

$$< \nabla \lambda_j(m,c,k), w_{\max} > \quad \text{for} \;\; \text{bad} \quad \lambda_j$$

are maximal among $w \in V_{good}^{\perp}$.

The basic isolation algorithm

Since

$$\Delta = (\delta m, \delta c, \delta k) \perp \nabla \lambda_j(m,c,k) \qquad \text{for all} \quad \text{good} \quad \lambda_j,$$

denote

$$V_{good}^{\perp} = \{ w \in \mathbb{R}^{3n} : < \nabla \lambda_j(m,c,k), w >= 0 \quad \text{for all} \;\; \text{good} \;\; \lambda_j \}.$$

Basic isolation algorithm organized in two stages:

1) Find a direction $w_{\max} \in V_{good}^{\perp}$ s.t. velocities

$$< \nabla \lambda_j(m,c,k), w_{\max} > \quad \text{for bad} \quad \lambda_j$$

are maximal among $w \in V_{good}^{\perp}$.

2) Given $w_{\max}$ from **Stage 1)**, find smallest $\alpha^* \in \mathbb{R}$ s.t. e-values corresp. to

$$(m^*, c^*, k^*) = (m,c,k) + \alpha^* w_{\max}$$

are all outside the resonance band.

Implementation of Stage 1): finding w_{max}

Stage 1) Find optimal direction w_{max} in $V_{good}^{\perp}$.

Need to compute:

Implementation of Stage 1): finding $w_{\max}$

Stage 1) Find optimal direction $w_{\max}$ in $V_{good}^{\perp}$.

Need to compute:

- Directional derivatives of e-values $\longrightarrow$

  $\longrightarrow$ requires all eigenvectors of initial QEP

Implementation of Stage 1): finding $w_{\max}$

Stage 1) Find optimal direction $w_{\max}$ in $V_{good}^{\perp}$.

Need to compute:

- Directional derivatives of e-values $\longrightarrow$

  $\longrightarrow$ requires all eigenvectors of initial QEP
- Orthonormal basis of $V_{good}^{\perp}$ (e.g., via QR factorization)

Implementation of Stage 1): finding $w_{\max}$

Stage 1) Find optimal direction $w_{\max}$ in $V_{good}^{\perp}$.

Need to compute:

- Directional derivatives of e-values $\longrightarrow$

  $\longrightarrow$ requires all eigenvectors of initial QEP
- Orthonormal basis of $V_{good}^{\perp}$ (e.g., via QR factorization)
- $w_{\max} \equiv$ singular vector corresp. to $\sigma_{\max}$ of scalar product matrix

OVERALL COST: $O(n^3)$

Implementation of Stage 2): detecting isolation

Stage 2) Given optimal direction $w_{\max} \in V_{good}^{\perp}$, find smallest $\alpha^* \in \mathbb{R}$ such that spectrum of

$$(m, c, k) + \alpha^* w_{\max}$$

has been isolated

Implementation of Stage 2): detecting isolation

Stage 2) Given optimal direction $w_{\max} \in V_{good}^{\perp}$, find smallest $\alpha^* \in \mathbb{R}$ such that spectrum of

$$(m,c,k) + \alpha^* w_{\max}$$

has been isolated

QEP is hyperbolic $\longrightarrow$ can use bisection on α to find how many e-values of

$$(m,c,k) + \alpha w_{\max}$$

are inside the resonance band $(\beta - r, \beta + r)$. Stop as soon as that number becomes zero.

Implementation of Stage 2): detecting isolation

Stage 2) Given optimal direction $w_{\max} \in V_{good}^{\perp}$, find smallest $\alpha^* \in \mathbb{R}$ such that spectrum of

$$(m,c,k) + \alpha^* w_{\max}$$

has been isolated

QEP is hyperbolic $\longrightarrow$ can use bisection on α to find how many e-values of

$$(m,c,k) + \alpha \, w_{\max}$$

are inside the resonance band $(\beta - r, \beta + r)$. Stop as soon as that number becomes zero.

COST: $O(n)$ per bisection step

Preservation of structure

BUT all this can only be done provided the relevant structure is preserved:

Preservation of structure

BUT all this can only be done provided the relevant structure is preserved:

- Entries of $m+\alpha^*\delta m$, $k+\alpha^*\delta k$, $c+\alpha^*\delta c$ must remain positive: $\longrightarrow$ positivity constraints on α.

Preservation of structure

BUT all this can only be done provided the relevant structure is preserved:

- Entries of $m+\alpha^*\delta m$, $k+\alpha^*\delta k$, $c+\alpha^*\delta c$ must remain positive: $\longrightarrow$ positivity constraints on α.
- New system (M^*, C^*, K^*) must stay hyperbolic: hyperbolicity constraints on α: use results on distance to nearest non-hyperbolic QEP

Higham, Tisseur & Van Dooren '02

easiest case: fix M and K, only change C

Preservation of structure

BUT all this can only be done provided the relevant structure is preserved:

- Entries of $m+\alpha^*\delta m$, $k+\alpha^*\delta k$, $c+\alpha^*\delta c$ must remain positive: $\longrightarrow$ positivity constraints on α.
- New system (M^*, C^*, K^*) must stay hyperbolic: hyperbolicity constraints on α: use results on distance to nearest non-hyperbolic QEP

Higham, Tisseur & Van Dooren '02

easiest case: fix M and K, only change C

All these constraints lead to a maximal range for α

$$-\tau^- \leq \alpha \leq \tau^+$$

for appropriate thresholds $\tau^-, \tau^+ > 0$.

Algorithm only works if there are no e-values in $(\beta - r, \beta + r)$ either for $\alpha = -\tau^-$ or for $\alpha = \tau^+$ $\longrightarrow$ provides starting interval for bisection.

The basic isolation algorithm: Shortcomings

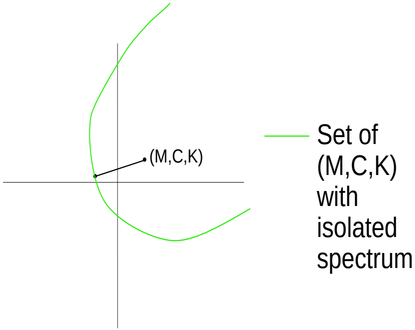
Summarizing: the $O(n^3)$ basic isolation algorithm is

- Likely to give good solutions only if Taylor approximation sufficiently accurate, i.e., if (M^*, C^*, K^*) sufficiently close to (M, C, K).

The basic isolation algorithm: Shortcomings

Summarizing: the $O(n^3)$ basic isolation algorithm is

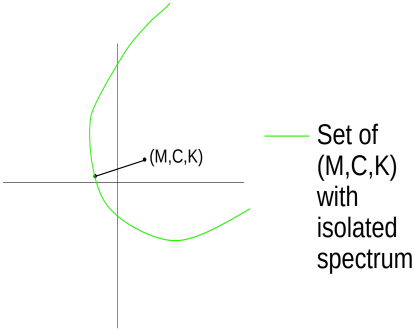
- Likely to give good solutions only if Taylor approximation sufficiently accurate, i.e., if (M^*, C^*, K^*) sufficiently close to (M, C, K).
- May not work at all if constraints prevent the algorithm from going all the way to isolation, e.g., positivity constraints:

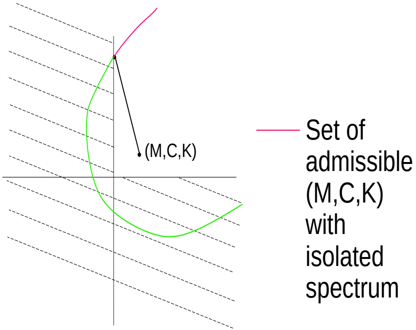


The basic isolation algorithm: Shortcomings

Summarizing: the $O(n^3)$ basic isolation algorithm is

- Likely to give good solutions only if Taylor approximation sufficiently accurate, i.e., if (M^*, C^*, K^*) sufficiently close to (M, C, K).
- May not work at all if constraints prevent the algorithm from going all the way to isolation, e.g., positivity constraints:





The basic isolation algorithm: Shortcomings

Summarizing: the $O(n^3)$ basic isolation algorithm is

- Likely to give good solutions only if Taylor approximation sufficiently accurate, i.e., if (M^*, C^*, K^*) sufficiently close to (M, C, K).
- May not work at all if constraints prevent the algorithm from going all the way to isolation, e.g., positivity constraints:

Both difficulties may be overcome

A continuation algorithm:

Instead of trying to isolate in one single run, repeat basic isolation procedure over and over, setting $(M_0, C_0, K_0) = (M, C, K)$ and updating

$$(M_{i+1}, C_{i+1}, K_{i+1}) = (M_i, C_i, K_i) + h_i\, w_{\max}^{(i)},$$

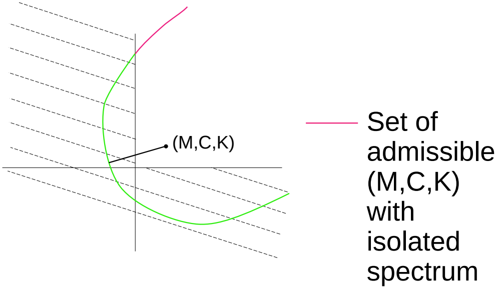
with some appropriate, small step size h_i, where $w_{\max}^{(i)}$ is the optimal direction at step i. Graphically,

A continuation algorithm:

Instead of trying to isolate in one single run, repeat basic isolation procedure over and over, setting $(M_0, C_0, K_0) = (M, C, K)$ and updating

$$(M_{i+1}, C_{i+1}, K_{i+1}) = (M_i, C_i, K_i) + h_i\, w_{\max}^{(i)},$$

with some appropriate, small step size h_i, where $w_{\max}^{(i)}$ is the optimal direction at step i. Graphically,

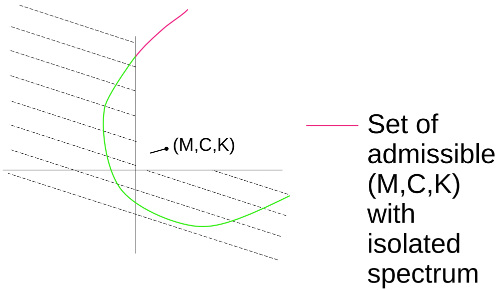


A continuation algorithm:

Instead of trying to isolate in one single run, repeat basic isolation procedure over and over, setting $(M_0, C_0, K_0) = (M, C, K)$ and updating

$$(M_{i+1}, C_{i+1}, K_{i+1}) = (M_i, C_i, K_i) + h_i \, w_{\max}^{(i)},$$

with some appropriate, small step size h_i, where $w_{\max}^{(i)}$ is the optimal direction at step i. Graphically,

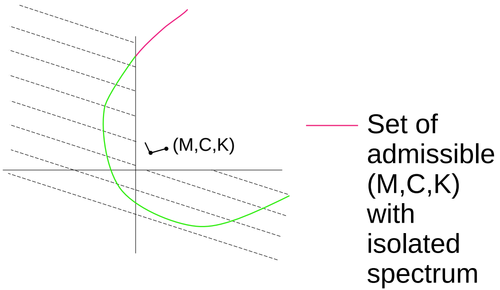


A continuation algorithm:

Instead of trying to isolate in one single run, repeat basic isolation procedure over and over, setting $(M_0, C_0, K_0) = (M, C, K)$ and updating

$$(M_{i+1}, C_{i+1}, K_{i+1}) = (M_i, C_i, K_i) + h_i\, w_{\max}^{(i)},$$

with some appropriate, small step size h_i, where $w_{\max}^{(i)}$ is the optimal direction at step i. Graphically,

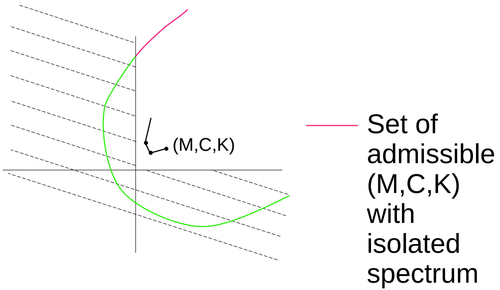


A continuation algorithm:

Instead of trying to isolate in one single run, repeat basic isolation procedure over and over, setting $(M_0, C_0, K_0) = (M, C, K)$ and updating

$$(M_{i+1}, C_{i+1}, K_{i+1}) = (M_i, C_i, K_i) + h_i \, w_{\max}^{(i)},$$

with some appropriate, small step size h_i, where $w_{\max}^{(i)}$ is the optimal direction at step i. Graphically,

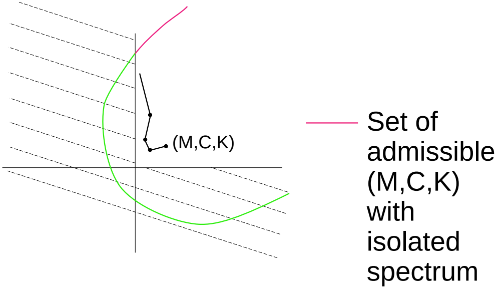


A continuation algorithm:

Instead of trying to isolate in one single run, repeat basic isolation procedure over and over, setting $(M_0, C_0, K_0) = (M, C, K)$ and updating

$$(M_{i+1}, C_{i+1}, K_{i+1}) = (M_i, C_i, K_i) + h_i\, w_{\max}^{(i)},$$

with some appropriate, small step size h_i, where $w_{\max}^{(i)}$ is the optimal direction at step i. Graphically,

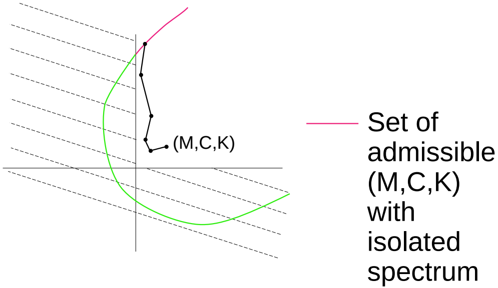


A continuation algorithm:

Instead of trying to isolate in one single run, repeat basic isolation procedure over and over, setting $(M_0, C_0, K_0) = (M, C, K)$ and updating

$$(M_{i+1}, C_{i+1}, K_{i+1}) = (M_i, C_i, K_i) + h_i \, w_{\max}^{(i)},$$

with some appropriate, small step size h_i, where $w_{\max}^{(i)}$ is the optimal direction at step i. Graphically,

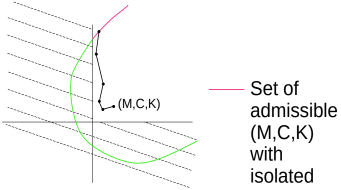


A continuation algorithm:

Instead of trying to isolate in one single run, repeat basic isolation procedure over and over, setting $(M_0, C_0, K_0) = (M, C, K)$ and updating

$$(M_{i+1}, C_{i+1}, K_{i+1}) = (M_i, C_i, K_i) + h_i\, w_{\max}^{(i)},$$

with some appropriate, small step size h_i, where $w_{\max}^{(i)}$ is the optimal direction at step i. Graphically,



How to choose h_i?

How to choose h_i?

$$(M_{i+1}, K_{i+1}) = (M_i, K_i) + h_i\, w_{\max}^{(i)}, \qquad (M_0, K_0) = (M, K)$$

Several possible choices for h_i. Best performance so far:

Greedy version: Try to isolate at each step. If not possible, then advance as far as possible in the optimal direction and repeat.

How to choose h_i?

$$(M_{i+1}, K_{i+1}) = (M_i, K_i) + h_i\, w_{\max}^{(i)}, \qquad (M_0, K_0) = (M, K)$$

Several possible choices for h_i. Best performance so far:

Greedy version: Try to isolate at each step. If not possible, then advance as far as possible in the optimal direction and repeat.

Compute thresholds τ_i^+, τ_i^- and optimal direction $w_{\max}^{(i)}$ at step i.

- If possible, compute α_i^* isolating the spectrum, take $h_i = \alpha_i^*$ and stop.
- If not, take h_i equal either to τ_i^+ or $-\tau_i^-$ and continue.

Outline

Numerical experiments: comparison with EKS

- Both basic and continuation algorithm implemented for tridiagonal undamped case ($C = 0$) in FORTRAN POWER STATION 4.0 on a PC with IEEE arithmetic
- Generate 1000 random 10×10 QEPs with one single e-value in the resonance band. Radius r of resonance band inversely proportional to $\delta < 1$:

  the smaller the parameter δ, the smaller the radius r

- Compare obtained solutions with solutions provided by 'Box' algorithm by Egaña, Kuhl & Santos '02 (EKS) whenever possible:
  - Box algorithm works only for low dimension (too expensive).
  - Box algorithm works only for uniform distributions of initial e-values.

Numerical experiments: Box vs. basic isolation

Generate 1000 random 10×10 mass-spring systems for each value of δ.

Set

- $m^*_{box}, k^*_{box} \equiv$ solutions computed by the 'Box' algorithm of Egaña, et al.
- m^*_{β}, k^*_{β}: solutions computed by the basic isolation algorithm.

$$Dist^{(\square)}_{m,k} = \frac{\|(m,k) - (m^*,k^*)_{\square}\|}{\|(m,k)\|},$$

where $\square$ is either 'box´ or 'β'. Define the quotients

$$Q^{(\beta)}_{m,k} = \frac{Dist^{(box)}_{m,k}}{Dist^{(\beta)}_{m,k}}, \qquad Q^{(\beta)}_{T} = \frac{\text{CPU time for Box alg.}}{\text{CPU time for basic isolation alg.}}$$

and the percentage $p_{m,k}$ of cases with $Q^{(\beta)}_{m,k} < 1$.

Numerical experiments: Box vs. basic isolation

Generate 1000 random 10×10 mass-spring systems for each value of δ.

$$Dist_{m,k}^{(\square)} = \frac{\|(m,k) - (m^*,k^*)_{\square}\|}{\|(m,k)\|},$$

where $\square$ is either 'box' or 'β'. Define the quotients

$$Q_{m,k}^{(\beta)} = \frac{Dist_{m,k}^{(box)}}{Dist_{m,k}^{(\beta)}}, \qquad Q_T^{(\beta)} = \frac{\text{CPU time for Box alg.}}{\text{CPU time for basic isolation alg.}}$$

and the percentage $p_{m,k}$ of cases with $Q_{m,k}^{(\beta)} < 1$.

	$Q_{m,k}^{(\beta)}$			$Q_T^{(\beta)}$		
δ	Average	Min	$p_{m,k}$	Average	Min	isolated
0.1	0.74	0.09	53.1%	514	61	97.8%
0.15	0.9	0.1	66.4%	420	53	96.9%
0.2	1.31	0.15	82.2%	359	51	89.1%
0.4	2.38	0.39	97.3%	228	48	59.6%

Numerical experiments: Box vs. continuation

Generate 1000 random 10×10 mass-spring systems for each value of δ.

Set

- $m^*_{box}, k^*_{box} \equiv$ solutions computed by the 'Box' algorithm of Egaña, et al.
- m^*_{cont}, k^*_{cont}: solutions computed by the continuation algorithm (greedy version).

$$Dist^{(\square)}_{m,k} = \frac{\|(m,k) - (m^*,k^*)_{\square}\|}{\|(m,k)\|},$$

where $\square$ is either 'box´ or '*cont*'. Define the quotients

$$Q^{(cont)}_{m,k} = \frac{Dist^{(box)}_{m,k}}{Dist^{(cont)}_{m,k}}, \qquad Q^{(cont)}_{T} = \frac{\text{CPU time for Box alg.}}{\text{CPU time for continuation alg.}}$$

and the percentage $p_{m,k}$ of cases with $Q^{(cont)}_{m,k} < 1$.

Numerical experiments: Box vs. continuation

Generate 1000 random 10×10 mass-spring systems for each value of δ.

$$Dist_{m,k}^{(\square)} = \frac{\|(m,k) - (m^*,k^*)_{\square}\|}{\|(m,k)\|},$$

where $\square$ is either 'box' or '*cont*'. Define the quotients

$$Q_{m,k}^{(cont)} = \frac{Dist_{m,k}^{(box)}}{Dist_{m,k}^{(cont)}}, \qquad Q_T^{(cont)} = \frac{\text{CPU time for Box alg.}}{\text{CPU time for continuation alg.}}$$

and the percentage $p_{m,k}$ of cases with $Q_{m,k}^{(cont)} < 1$.

	$Q_{m,k}^{(cont)}$			$Q_T^{(cont)}$	
δ	Average	Min	$p_{m,k}$	Average	Min
0.2	0.66	0.04	61.2%	532	67
0.3	0.67	0.009	65.3%	398	65
0.4	0.83	0.016	69.3%	276	50
0.6	1.06	0.04	77.4%	218	36

Summary:

New, directional algorithm for the frequency isolation problem proposed for hyperbolic QEPs.

Tested only for tridiagonal undamped case.

- **Basic algorithm**: $O(n^3)$, but isolation guaranteed only for systems close to non-resonance.
- **Continuation algorithm**:
  - Cost $O(n^3)$ per step.
  - Much faster than 'Box' algorithm.
  - Same quality of approximations as Box alg.
  - More robust: works irrespective of spectral distribution or distance to non-resonance.