KATHOLIEKE UNIVERSITEIT LEUVEN

OPTEC NALAG

Accelerating PDE-Constrained Optimization by Model Order Reduction with Error Control

Presenter: Yao Yue

Promotor: Karl Meerbergen

July 25, 2012

Outline

1. Optimal Design of Vibration Systems
2. Krylov Padé type MOR and its use in optimization
3. Error-based trust region method
4. Numerical Results
5. Conclusions

Parameterized Vibration Systems

Consider a SISO vibration system with parameter $\gamma = (\gamma_1, \gamma_2, \ldots, \gamma_k)$.

General FEM model:

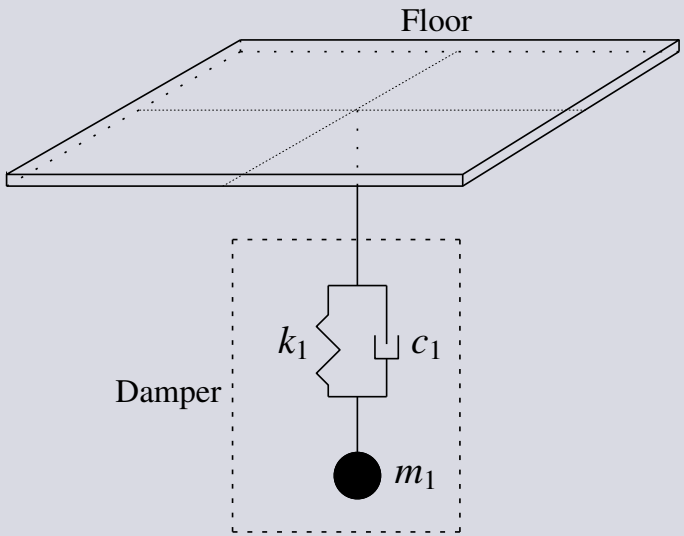
$$\begin{cases} \big(K(\gamma) + i\omega C(\gamma) - \omega^2 M(\gamma)\big)x(\omega,\gamma) = f, \\ y(\omega,\gamma) = \ell^* x(\omega,\gamma), \end{cases}$$

Input $f \in \mathbb{C}^n$ to State $x \in \mathbb{C}^n$
State $x \in \mathbb{C}^n$ to Output $y \in \mathbb{C}^1$

$K(\gamma)$, $C(\gamma)$, $M(\gamma)$: stiffness, damping and mass matrices.

Example 1: Floor Damper Optimization

Problem Statement



- **Goal**: Use a damper to decrease the vibration of a floor.
- **Floor**: Described by a shell element finite element model.
- **Damper**: Described by a K-C-M model.
- **Design Objective:** Minimize the vibration at a point on the floor by choosing k_1 and c_1.

Mathematical Formulation

$$\begin{cases} \left(K_0 + (k_1 + i\omega c_1)K_1 - \omega^2 M\right) x = f, \\ y = \ell^* x. \end{cases}$$

Example 2: Footbridge Damper Optimization



Use four dampers to decrease the vibration of a footbridge.

$$\begin{cases} \left(K_0 + i\omega C_0 + \sum_{i=1}^{4}(k_i + i\omega c_i)K_i - \omega^2 M_0\right)x = f, \\ y = \ell^* x. \end{cases}$$

Design Optimization: Minimize the Energy Norm

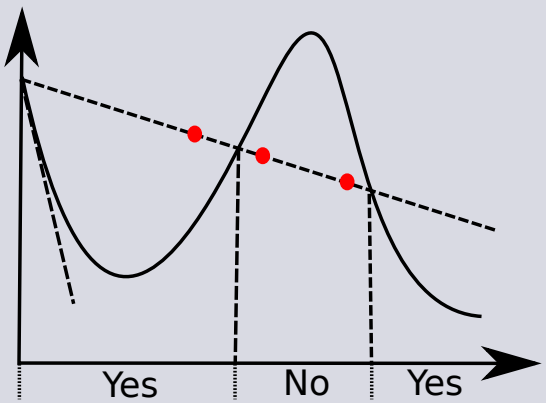
$$\min_{\gamma} \; g(\gamma), \qquad g(\gamma) = \int_{\omega_L}^{\omega_H} |y(\omega, \gamma)|^2 \, \mathrm{d}\omega$$

Algorithm

- To evaluate g: Trapezoidal rule for numerical integration.
- To optimize g: Damped BFGS method.

For each iteration of damped BFGS, we force **the Armijo condition**:

$$g(\gamma^{(i)}) - g(\gamma) \geq \kappa_s \left| \nabla_{(\gamma - \gamma^{(i)})} g(\gamma^{(i)}) \right| \left\| \gamma^{(i)} - \gamma \right\|, \quad \kappa_s \in (0, 1)$$

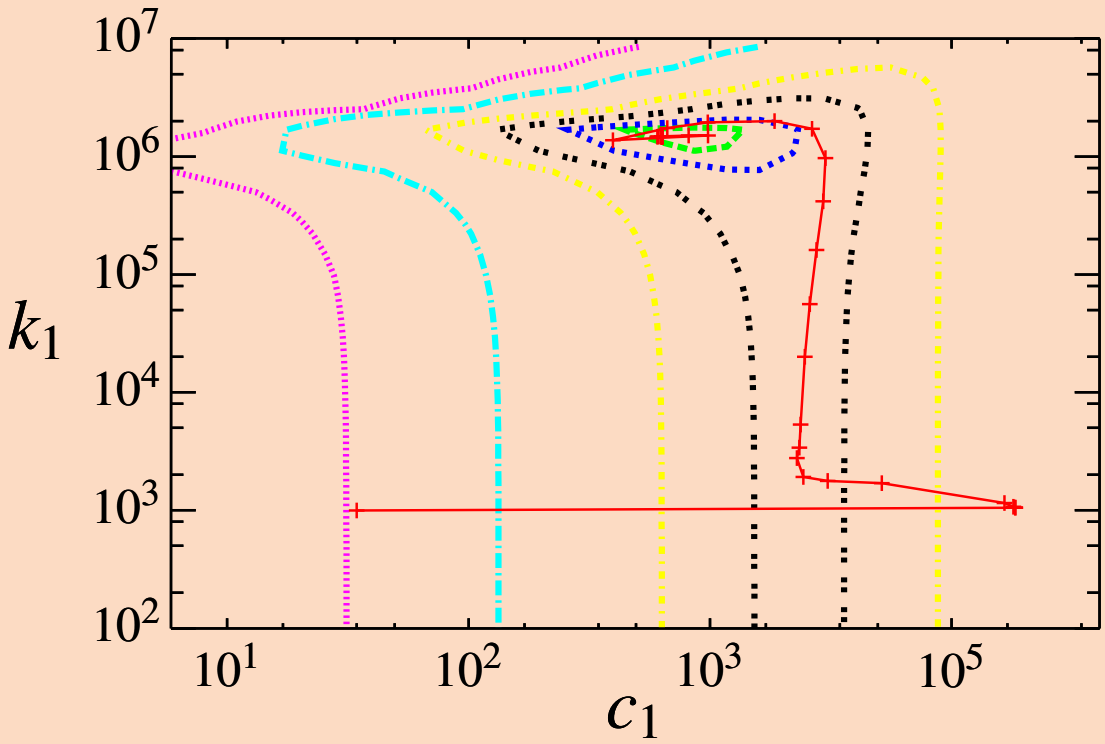


Backtracking strategy:

If the Armijo condition does not satisfy

$\Longrightarrow$ scale the step length with $\kappa_{bt} \in (0, 1)$.

An illustration of damped BFGS



Computational Analysis

Let $\mathcal{L}(\omega,\gamma) = K(\gamma) + i\omega C(\gamma) - \omega^2 M(\gamma)$, then

$$
\begin{aligned}
y &= \ell^* \mathcal{L}(\omega,\gamma)^{-1} f, \\
\frac{\partial y}{\partial \omega} &= -\ell^* \mathcal{L}(\omega,\gamma)^{-1} \frac{\partial \mathcal{L}(\omega,\gamma)}{\partial \omega} \mathcal{L}(\omega,\gamma)^{-1} f, \\
\frac{\partial y}{\partial \gamma_j} &= -\ell^* \mathcal{L}(\omega,\gamma)^{-1} \frac{\partial \mathcal{L}(\omega,\gamma)}{\partial \gamma_j} \mathcal{L}(\omega,\gamma)^{-1} f.
\end{aligned}
$$

- The LU factorization of $\mathcal{L}(\omega,\gamma)$ is *shared* by y, $\frac{\partial y}{\partial \omega}$ and $\frac{\partial y}{\partial \gamma_j}$:
  - Quasi-Newton type methods are suitable.
- Expensive: Numerical integration for $g(\gamma) = \int_{\omega_L}^{\omega_H} |y(\omega,\gamma)|^2 \, \mathrm{d}\omega$ needs LU factorization of $\mathcal{L}(\omega,\gamma)$ for many ω's.
- Our Solution: Use Model Order Reduction (MOR) to reduce the computational cost.

Outline

1. Optimal Design of Vibration Systems
2. Krylov Padé type MOR and its use in optimization
3. Error-based trust region method
4. Numerical Results
5. Conclusions

Projection Based Model Order Reduction

High Order System $\xrightarrow{\text{Projection}}$ *Low Order System*

$$\begin{cases} \mathcal{L}(\omega)x = f \\ y = \ell^* x \end{cases} \qquad \begin{cases} \widehat{\mathcal{L}}(\omega)z = \widehat{f} \\ \widehat{y} = \widehat{\ell}^* z \end{cases}$$

Order n $\qquad n \gg k \qquad$ Order k

We focus on Krylov-Padé type methods: suitable for large-scale linear systems.

Krylov-Padé type MOR:

Procedure of Two-sided Krylov methods:

- Generate the right Krylov subspace to obtain a basis $V \in \mathbb{C}^{n\times k}$;
- Generate the left Krylov subspace to obtain a basis $W \in \mathbb{C}^{n\times k}$;
- Compute $\widehat{\mathcal{L}}(\omega) = W^*\mathcal{L}(\omega)V$, $\widehat{f} = W^*f$, $\widehat{\ell} = V^*\ell$. $(W^*(\mathcal{L}(\omega)Vz - f) = 0)$.

Algorithm: we use **the SOAR method** to reduce second order systems.

A key property for Krylov methods: moment matching between y and $\widehat{y}$.

- The i-th moment: the i-th coefficient in the Taylor expansion.
- Moment matching implies Padé type approximation.

Moment matching properties

Theorem: For two-sided SOAR

- The first $2k$ moments of $y(\omega, \gamma^{(i)})$ and $\widehat{y}(\omega, \gamma^{(i)})$ match;
- The first $2k-1$ moments of $\frac{\partial y}{\partial \omega}\Big|_{\gamma=\gamma^{(i)}}$ and $\frac{\partial \widehat{y}}{\partial \omega}\Big|_{\gamma=\gamma^{(i)}}$ match;
- The first k moments of $\frac{\partial y}{\partial \gamma_j}\Big|_{\gamma=\gamma^{(i)}}$ and $\frac{\partial \widehat{y}}{\partial \gamma_j}\Big|_{\gamma=\gamma^{(i)}}$ match;

Using MOR in design optimization

Problem:

How to apply MOR on the parameterized system for optimization?

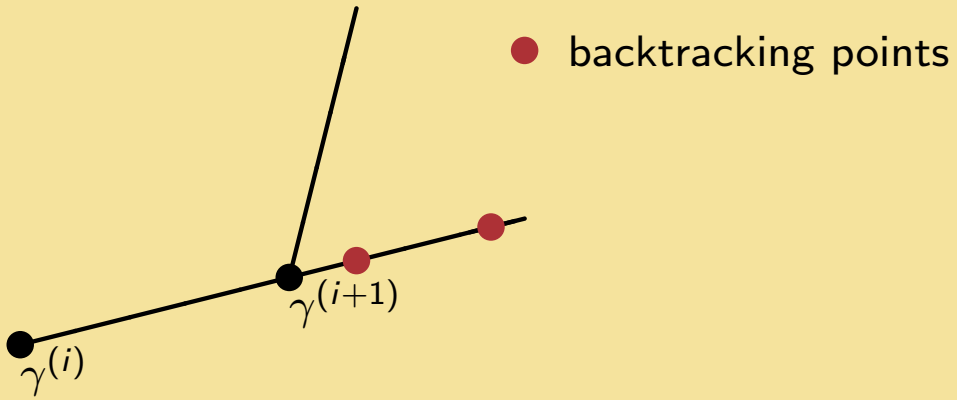
$$\begin{cases} \mathcal{L}(\omega, \gamma) z = f \\ y = \ell^* z \end{cases}$$

Two Existing Approaches:

1. Parameterized MOR. Not the focus of this talk.
2. The MOR Framework.

The MOR Framework

Generate a reduced model for each γ accessed by optimization.



Why does it work?

$g(\gamma) = \int_{\omega_L}^{\omega_H} |y(\omega,\gamma)|^2 \,\mathrm{d}\omega$, $\nabla g(\gamma) = \int_{\omega_L}^{\omega_H} \nabla_\gamma |y(\omega,\gamma)|^2 \,\mathrm{d}\omega$

$\widehat{g}^{(i)}(\gamma^{(i)}) \approx g(\gamma^{(i)})$ and $\nabla\widehat{g}^{(i)}(\gamma^{(i)}) \approx \nabla g(\gamma^{(i)})$ due to the above theorem.

Meeting the requirement of Quasi-Newton methods.

How much is the speedup? (Assume that we use N points in numerical integration)

I. The Direct Approach: Large-sparse LU factorization for N ω's.

II. The MOR Framework:

- Step 1: $g \to \widehat{g}^{(i)}$: One large-sparse LU factorization.
- Step 2: Conduct numerical integration on $\widehat{g}^{(i)}$: very cheap.

Roughly speaking, the factor of speedup is N.

Can we do better?

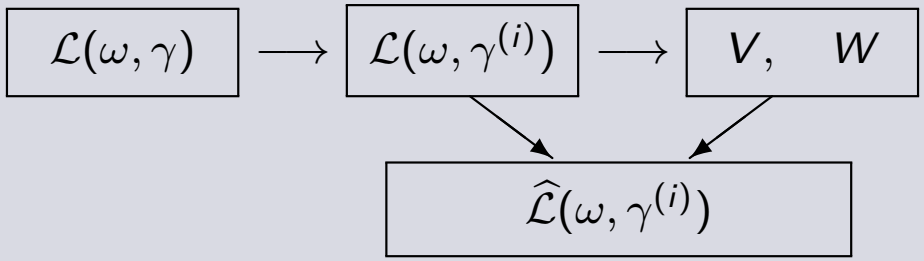
Since $\widehat{g}^{(i)}(\gamma^{(i)}) \approx g(\gamma^{(i)})$ and $\nabla\widehat{g}^{(i)}(\gamma^{(i)}) \approx \nabla g(\gamma^{(i)})$, $\widehat{g}^{(i)}$ should be also accurate in a neighborhood of $\gamma^{(i)}$. Can we exploit this neighborhood?

Trust Region idea?

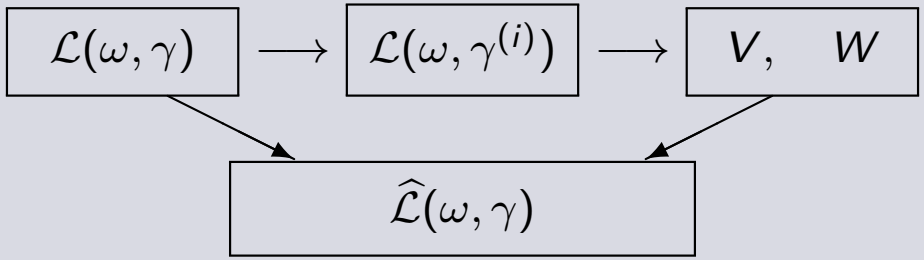
Step 1: Interpolatory MOR

First, we need a reduced model defined for $\gamma \neq \gamma^{(i)}$.

MOR:



Interpolatory MOR: with a single interpolation point



Step 2: Using interpolatory reduced models with error bounds

In optimization, we normally require a surrogate model $\widehat{g}^{(i)}$ to satisfy the

The First Order Condition $\boxed{\widehat{g}^{(i)}(\gamma^{(i)}) = g(\gamma^{(i)}) \quad \text{and} \quad \nabla\widehat{g}^{(i)}(\gamma^{(i)}) = \nabla g(\gamma^{(i)})}$

so that we know our "current status" and in which direction to go.

Question: is $\boxed{\widehat{g}^{(i)}(\gamma^{(i)}) \approx g(\gamma^{(i)}),\ \nabla\widehat{g}^{(i)}(\gamma^{(i)}) \approx \nabla g(\gamma^{(i)})}$ OK?

We need to measure how good it is, and refine it at request. Thus, we developed a heuristic error bound based on the residue $\mathcal{L}(\omega,\gamma)Vz - f$. We omit the details.

Our Situation

The Relaxed First Order Condition

1. Relatively cheap error bounds $e^{(i)}(\gamma)$ and $e_g^{(i)}$ are available, i.e.
$$|\widehat{g}^{(i)}(\gamma) - g(\gamma)| \le e^{(i)}(\gamma) \qquad \text{and} \qquad \|\nabla\widehat{g}^{(i)}(\gamma^{(i)}) - \nabla g(\gamma^{(i)})\| \le e_g^{(i)};$$
2. The surrogate model can be infinitely refined at $\gamma^{(i)}$ (by enlarging the Krylov subspace), i.e., for any given $\tau_g > 0$ and $\tau_{\nabla g} > 0$, there exists a surrogate model $\widehat{g}^{(i)}$ satisfying $\dfrac{e^{(i)}(\gamma^{(i)})}{\widehat{g}^{(i)}(\gamma^{(i)})} \le \tau_g$ and $\dfrac{e_g^{(i)}}{\|\nabla\widehat{g}^{(i)}(\gamma^{(i)})\|} \le \tau_{\nabla g}$.
3. Directly evaluating g is extremely expensive;
4. Generating $\widehat{g}^{(i)}$ is much less expensive, but still not cheap.
5. Evaluating $\widehat{g}^{(i)}$ is very cheap.

Outline

1. Optimal Design of Vibration Systems
2. Krylov Padé type MOR and its use in optimization
3. Error-based trust region method
4. Numerical Results
5. Conclusions

Standard Trust Region Method (STR)

At the i-th iteration, STR generates a surrogate model $\widehat{g}^{(i)}$, and trusts this model within a trust radius Δ_i.

The i-th subproblem: $\boxed{\min_\gamma \widehat{g}^{(i)}(\gamma), \qquad \text{s.t.} \left\|\gamma - \gamma^{(i)}\right\| \leq \Delta_i.}$

Denote the computed optimizer with $\gamma_{\text{cand}}^{(i+1)}$.

We know: $\gamma_{\text{cand}}^{(i+1)}$ achieves sufficient decrease on $\widehat{g}^{(i)}$. (Better than the Cauchy point)

We want: $\gamma_{\text{cand}}^{(i+1)}$ also achieves sufficient decrease on g.

A bridge: $\rho_i = \frac{g(\gamma^{(i)}) - g(\gamma_{\text{cand}}^{(i+1)})}{\widehat{g}^{(i)}(\gamma^{(i)}) - \widehat{g}^{(i)}(\gamma_{\text{cand}}^{(i+1)})}$.

1. If ρ_i is sufficiently positive, we are lucky. Assign $\gamma^{(i+1)} = \gamma_{\text{cand}}^{(i+1)}$ and $\Delta_{i+1} \geq \Delta_i$.
2. Otherwise, the approximation quality at $\gamma_{\text{cand}}^{(i+1)}$ is not acceptable.
   We reject $\gamma_{\text{cand}}^{(i+1)}$, reduce Δ_i and solve the i-th subproblem again.

STR increases the approximation quality purely by reducing Δ_i because it assumes **The First Order Condition**:

$$\widehat{g}^{(i)}(\gamma^{(i)}) = g(\gamma^{(i)}) \quad \text{and} \quad \nabla\widehat{g}^{(i)}(\gamma^{(i)}) = \nabla g(\gamma^{(i)}).$$

Gradually reducing the trust radius

$\Longrightarrow$ The average error inside the trust radius approaches zero.

Error-based trust region method

Problems under the Relaxed First Order Condition

- Besides reducing the trust radius, we may also need to refine the surrogate model to increase the quality of approximation.
- Computing $\rho_i = \frac{g(\gamma^{(i)}) - g(\gamma_{\text{cand}}^{(i+1)})}{\widehat{g}^{(i)}(\gamma^{(i)}) - \widehat{g}^{(i)}(\gamma_{\text{cand}}^{(i+1)})}$ requires evaluating g directly, which is very expensive.

Our Goal

1. Avoid directly computing $g(\gamma)$.
2. Build as few surrogate (reduced) models as possible, or try to fully exploit each surrogate model.
3. Provable convergence.

Error-based trust region method

A General Framework

1. At the i-th iteration, generate a surrogate model $\widehat{g}^{(i)}$;
2. Reduce $\widehat{g}^{(i)}$ under some constraints to achieve convergence on g.

Connection with g: $\left\|\nabla \widehat{g}^{(i)}(\gamma^{(i)})\right\| \to 0$ implies $\left\|\nabla g(\gamma^{(i)})\right\| \to 0$ since

$1 - \tau_{\nabla g} \leq \frac{\|\nabla g(\gamma^{(i)})\|}{\|\nabla \widehat{g}^{(i)}(\gamma^{(i)})\|} \leq 1 + \tau_{\nabla g}$. (We require $\tau_{\nabla g} < 1$ when we build $\widehat{g}^{(i)}$.)

The approximate generalized Cauchy point $\gamma_{\text{AGC}}^{(i)}$

Consider an Armijo search with the backtracking strategy.

$\gamma_{\text{AGC}}^{(i)}$ is the first point satisfying both

- the Armijo condition on $\widehat{g}^{(i)}(\gamma)$;
- the constraint $c^{(i)}(\gamma) \geq 0$.

What does $\gamma_{\text{AGC}}^{(i)}$ achieve?

Theorem

If both $\widehat{g}^{(i)}$ and $c^{(i)}$ are Lipschitz continuous, with their Lipschitz constants bounded for all iterations, we have:

$$\sum_{i=1}^{\infty} \widehat{g}^{(i)}(\gamma^{(i)}) - \widehat{g}^{(i)}(\gamma_{\text{AGC}}^{(i)})$$

is bounded only if $\left\|\nabla \widehat{g}^{(i)}(\gamma^{(i)})\right\| \to 0$, or $\left\|\nabla g(\gamma^{(i)})\right\| \to 0$.

The Ideal Case

$$\widehat{g}^{(i+1)}(\gamma^{(i+1)}) \leq \widehat{g}^{(i)}(\gamma_{\text{AGC}}^{(i)}) \quad \text{holds for all } i\text{'s.}$$

We have $\lim_{i\to\infty} \widehat{g}^{(i)}(\gamma^{(0)}) - \widehat{g}^{(i)}(\gamma_{\text{AGC}}^{(i)})$ is bounded only if $\left\|\nabla g(\gamma^{(i)})\right\| \to 0$.

Therefore, if all $\widehat{g}^{(i)}(\gamma)$ are bounded from below, we achieve convergence.

Sufficient Decrease Condition

We accept $\gamma_{\text{cand}}^{(i+1)}$ as $\gamma^{(i+1)}$ if $\boxed{\widehat{g}^{(\gamma_{\text{cand}}^{(i+1)})}(\gamma_{\text{cand}}^{(i+1)}) \leq \widehat{g}^{(i)}(\gamma_{\text{AGC}}^{(i)})}$

Don't require evaluating g!

What if it is not satisfied?

Reason 1: $\widehat{g}^{(i)}(\gamma_{\text{AGC}}^{(i)})$ is too small. We expect $\gamma_{\text{AGC}}^{(i)}$ to give a reasonable sufficient decrease that we can achieve, but $\widehat{g}^{(i)}$ may fail to represent g at $\gamma_{\text{AGC}}^{(i)}$.

- Solution:
  1. Conduct backtracking on $\gamma_{\text{AGC}}^{(i)}$ since we expect a smaller step causes better approximation. Also leads to convergence. Very cheap.
  2. Refine the surrogate model $\widehat{g}^{(i)}$ (e.g. by enlarging Krylov subspaces).

Reason 2: $\widehat{g}^{(\gamma_{\text{cand}}^{(i+1)})}(\gamma_{\text{cand}}^{(i+1)})$ is too big. $\gamma_{\text{cand}}^{(i+1)}$ should be rejected.

- Solution: Modify the subproblem and solve it again.

  Our two choices
  1. Shrink the trust region;
  2. Add a stronger penalty to the objective.

Computational Considerations

Compute $\widehat{g}^{(\gamma_{\text{cand}}^{(i+1)})}(\gamma_{\text{cand}}^{(i+1)})$ requires building a new surrogate model $\widehat{g}^{(\gamma_{\text{cand}}^{(i+1)})}$.
Try to avoid this computation.

Lemma

- A sufficient condition for the sufficient decrease condition is

$$\widehat{g}^{(i)}(\gamma_{\text{cand}}^{(i+1)}) + e^{(i)}(\gamma_{\text{cand}}^{(i+1)}) + \cancel{e^{(\gamma_{\text{cand}}^{(i+1)})}(\gamma_{\text{cand}}^{(i+1)})} \leq \widehat{g}^{(i)}(\gamma_{\text{AGC}}^{(i)}).$$

- A necessary condition for the sufficient decrease condition is

$$\widehat{g}^{(i)}(\gamma_{\text{cand}}^{(i+1)}) - e^{(i)}(\gamma_{\text{cand}}^{(i+1)}) - \cancel{e^{(\gamma_{\text{cand}}^{(i+1)})}(\gamma_{\text{cand}}^{(i+1)})} \leq \widehat{g}^{(i)}(\gamma_{\text{AGC}}^{(i)}).$$

Since $e^{(\gamma_{\text{cand}}^{(i+1)})}(\gamma_{\text{cand}}^{(i+1)}) \approx 0$ and can be controlled when generating $\widehat{g}^{(\gamma_{\text{cand}}^{(i+1)})}$,
we can ignore this term and rely purely on $\widehat{g}^{(i)}$.

Error-based trust region method

A Practical Procedure:

To check $\widehat{g}^{(\gamma_{\text{cand}}^{(i+1)})}(\gamma_{\text{cand}}^{(i+1)}) \leq \widehat{g}^{(i)}(\gamma_{\text{AGC}}^{(i)})$,

1. Check the sufficient condition. If it holds, accept $\gamma_{\text{cand}}^{(i+1)}$.
2. Otherwise, check the necessary condition. If it fails, reject $\gamma_{\text{cand}}^{(i+1)}$, reformulate the subproblem and update $\gamma_{\text{cand}}^{(i+1)}$.
3. Otherwise, if $\gamma_{\text{AGC}}^{(i)}$ is not too close to $\gamma^{(i)}$, conduct backtracking on $\gamma_{\text{AGC}}^{(i)}$ and go to Step 1.
4. Otherwise, generate $\widehat{g}^{(\gamma_{\text{cand}}^{(i+1)})}$ and check the true condition. If it holds, accept $\gamma_{\text{cand}}^{(i+1)}$; otherwise reformulate the subproblem and update $\gamma_{\text{cand}}^{(i+1)}$.

Here, we only need to generate $\widehat{g}^{(\gamma_{\text{cand}}^{(i+1)})}$ if we reach step 4.

Error-based trust region method

How to formulate the optimization subproblem to locate $\gamma_{\text{cand}}^{(i+1)}$?

The ideal case

The sufficient condition $\boxed{\widehat{g}^{(i)}(\gamma_{\text{cand}}^{(i+1)}) + e^{(i)}(\gamma_{\text{cand}}^{(i+1)}) \leq \widehat{g}^{(i)}(\gamma_{\text{AGC}}^{(i)})}$ holds.

Our two approaches:

1. ETR: (Error-based trust region method) only trust $\widehat{g}^{(i)}$ when $e^{(i)}(\gamma)$ is small compared with $\widehat{g}^{(i)}(\gamma)$.

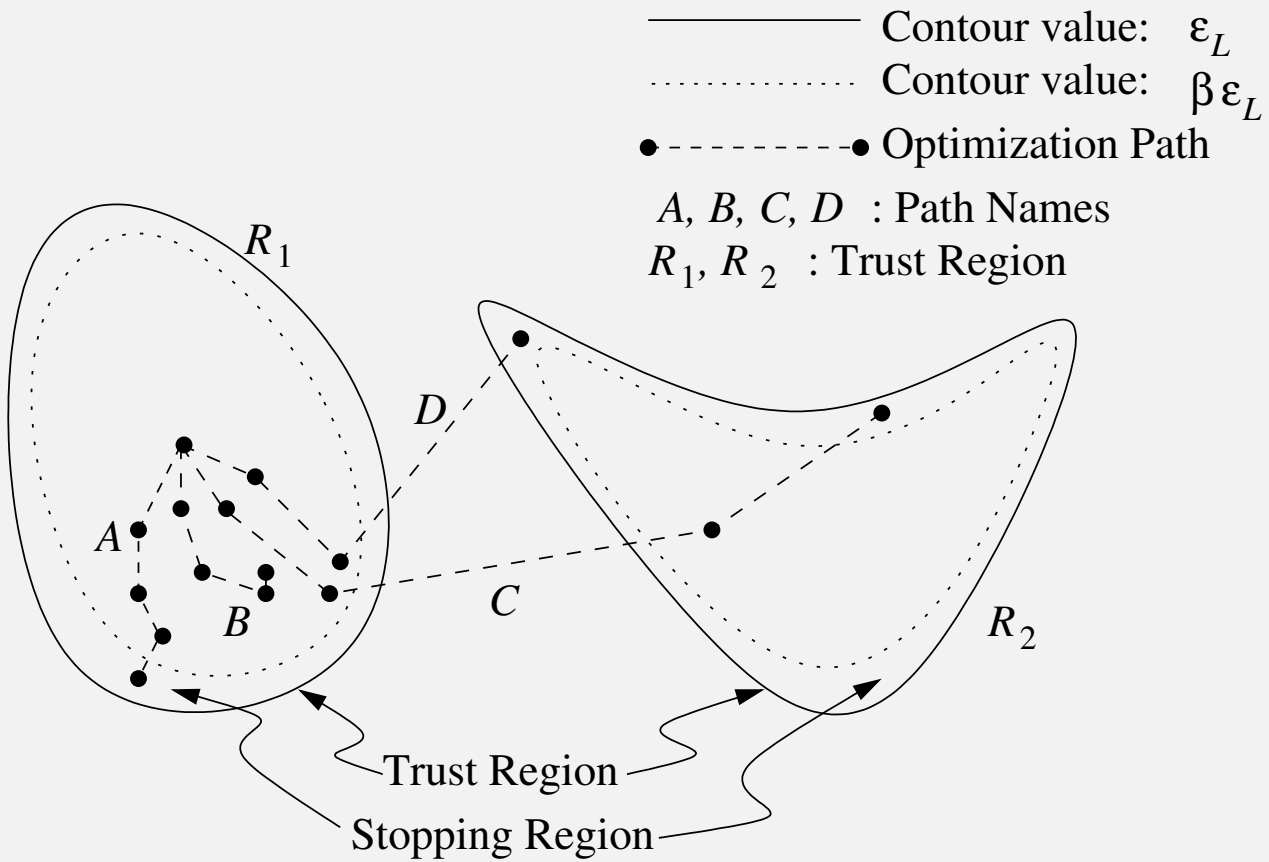
$$\boxed{\min_\gamma \ \widehat{g}^{(i)}(\gamma) \qquad \text{s.t.} \ \frac{e^{(i)}(\gamma)}{\widehat{g}^{(i)}(\gamma)} \leq \epsilon_L. \quad (\epsilon_L > 0)}$$

2. EP: (Error-based penalty method) consider $\widehat{g}^{(i)}(\gamma) + e^{(i)}(\gamma)$ as a whole. Using a differentiable weighting function:
$w(\epsilon) = 0$ when $\epsilon \leq \epsilon_L$, $\quad w(\epsilon) = 1$, when $\epsilon \geq \epsilon_H$, $\quad (\epsilon_L < \epsilon_H)$.

$$\boxed{\min_\gamma \quad F^{(i)}(\gamma) = \widehat{g}^{(i)}(\gamma) + w\left(\frac{e^{(i)}(\gamma)}{\widehat{g}^{(i)}(\gamma)}\right) e^{(i)}(\gamma).}$$

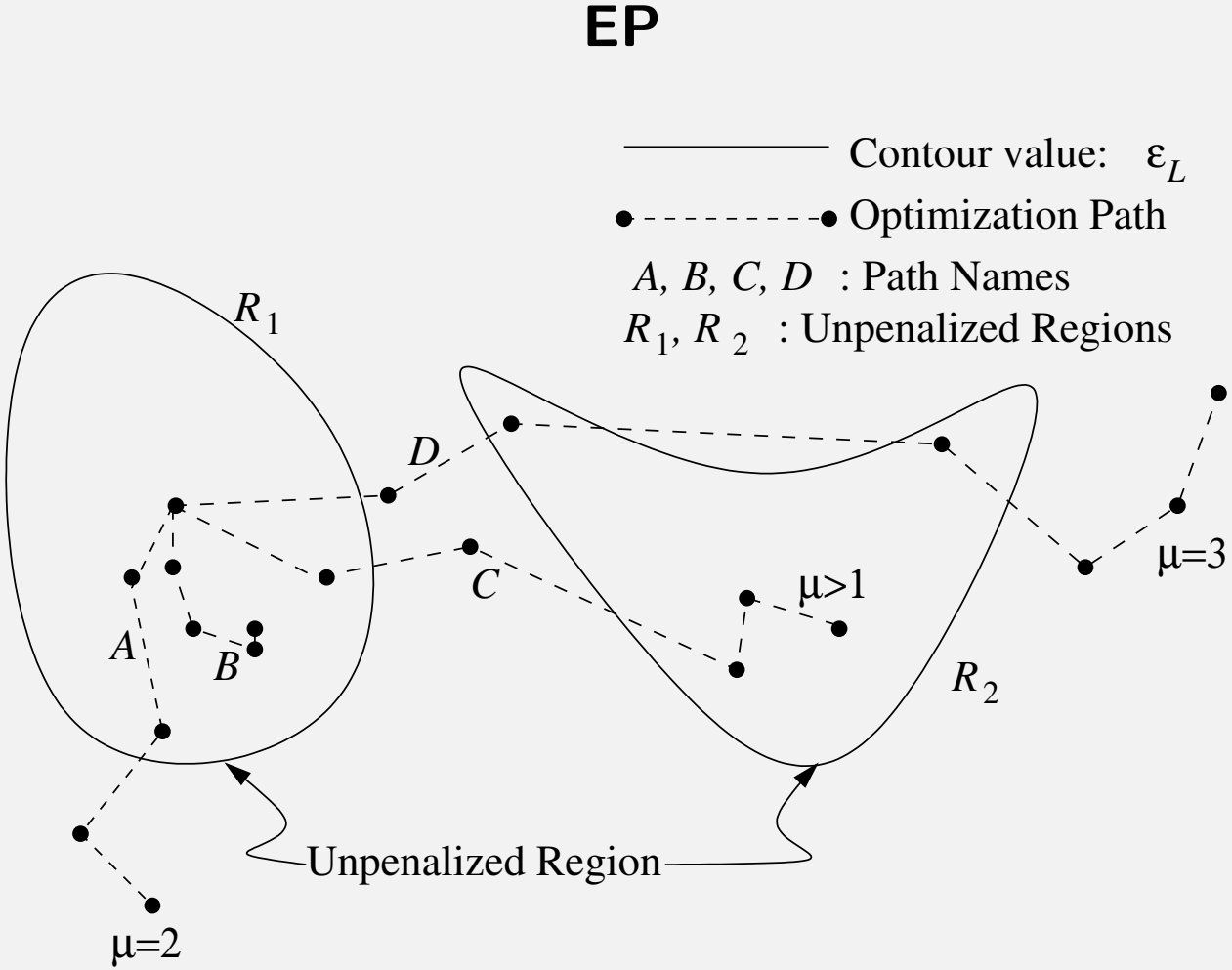
Error-based trust region method

ETR



We backtrack if we lie out of the trust region.
We terminate when we reach the stopping region.

Error-based trust region method



We terminate if we stay in the penalized region for μ successive steps. If $\nabla e^{(i)}(\gamma)$ is expensive to compute, we prefer $\mu = 1$.

Outline

1. Optimal Design of Vibration Systems
2. Krylov Padé type MOR and its use in optimization
3. Error-based trust region method
4. Numerical Results
5. Conclusions

Numerical Results

For the Floor Damper Optimization Problem

	Order	Optimum	CPU Time	Relative Error
ETR	7	1.3162e+10	220s	2.626620741e-09
	10	1.3162e+10	35s	2.424547354e-11
EP(1)	7	1.3162e+10	129s	2.616067046e-09
	10	1.3162e+10	41s	1.729372585e-11
EP(2)	7	1.3162e+10	169s	2.585171968e-09
	10	1.3162e+10	45s	3.963751122e-11
EP(μ) ($\mu \geq 8$)	10	1.3162e+10	58s	7.083806541e-11
ETR & EP(μ) ($\mu \geq 1$)	12	1.3162e+10	25s	3.87328218e-11
The MOR Framework	7	1.3162e+10	89s	2.6865095e-09
Direct Method	29800	1.3162e+10	7202s	—

When Order=7, ETR and EP requires 16–20 reduced models.
When Order=10, ETR and EP requires 3–4 reduced models.

The MOR Framework requires 9 reduced models.

Numerical Results

For the Footbridge Damper Optimization Problem

	Order	Optimum	CPU Time	Relative Error
ETR	18	24.77751668	705s	1.313402699e-12
	20	24.78594112	205s	4.465054737e-12
EP(1)	18	24.77751694	1245s	1.033587692e-11
	20	24.7762798	295s	3.730479306e-12
EP(2)	18	24.77751799	542s	7.982648976e-12
	20	24.7775166	190s	7.944365742e-12
ETR & EP(μ) ($\mu \geq 1$)	32	24.80397266	194s	1.092742827e-11
The MOR Framework	12	24.77751651	879s	2.023983216e-07

Outline

1. Optimal Design of Vibration Systems
2. Krylov Padé type MOR and its use in optimization
3. Error-based trust region method
4. Numerical Results
5. Conclusions

Conclusions

- Design optimization of large scale dynamical systems is usually very expensive.
- Two-sided MOR can be used to reduce the computational cost of Quasi-Newton type optimization.
- Under the relaxed first order condition, we proposed a sufficient decrease condition, under which we achieve convergence. We also discuss how to check this condition efficiently.
- We developed two algorithms: ETR and EP. Both of them are efficient in speeding up design optimization.

Thank you!