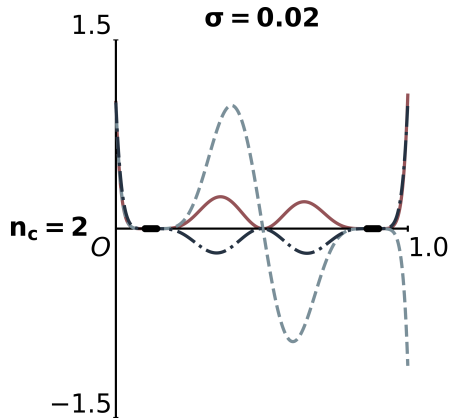
Sharpened CG Iteration Bound for High-contrast Heterogeneous Scalar Elliptic PDEs

Going Beyond Condition Number

P. Soliman[1]

EEMCS-DIAM Numerical Analysis MSc. Thesis Presentation, September, 2025

[1] Delft University of Technology



TU Delft

Opening

Motivation

- Opening question: ”What do X, Y, Z have in common?”. (Show X, Y and Z with single picture slides and make sure to give short descriptions. Go quickly through the single-picture slides.)

Opening

Motivation

- Opening question: "What do X, Y, Z have in common?". (Show X, Y and Z with single picture slides and make sure to give short descriptions. Go quickly through the single-picture slides.)
- Answer: "They can all be modelled by a..." high-contrast diffusion problem for $u, u_D \in H^1(\Omega)$, $f \in L^2(\Omega)$ and $\mathcal{C} \in L^\infty(\Omega)$

$$\begin{aligned} -\nabla \cdot (\mathcal{C} \nabla u) &= f \quad \text{in } \Omega \\ u &= u_D \quad \text{on } \partial\Omega \end{aligned}$$

Opening

Motivation

- Opening question: "What do X, Y, Z have in common?". (Show X, Y and Z with single picture slides and make sure to give short descriptions. Go quickly through the single-picture slides.)
- Answer: "They can all be modelled by a..." high-contrast diffusion problem for $u, u_D \in H^1(\Omega)$, $f \in L^2(\Omega)$ and $\mathcal{C} \in L^\infty(\Omega)$

$$\begin{aligned} -\nabla \cdot (\mathcal{C}\nabla u) &= f \quad \text{in } \Omega \\ u &= u_D \quad \text{on } \partial\Omega \end{aligned}$$

- Understanding the problem: We need to find $u \in H^1(\Omega)$ such that the above equation holds.

Opening

Motivation

- Opening question: "What do X, Y, Z have in common?". (Show X, Y and Z with single picture slides and make sure to give short descriptions. Go quickly through the single-picture slides.)
- Answer: "They can all be modelled by a..." high-contrast diffusion problem for $u, u_D \in H^1(\Omega)$, $f \in L^2(\Omega)$ and $\mathcal{C} \in L^\infty(\Omega)$

$$\begin{aligned} -\nabla \cdot (\mathcal{C}\nabla u) &= f \quad \text{in } \Omega \\ u &= u_D \quad \text{on } \partial\Omega \end{aligned}$$

- Understanding the problem: We need to find $u \in H^1(\Omega)$ such that the above equation holds.
- Briefly explain why discretization is necessary for PDEs, for non-experts.

Opening

Motivation

- Opening question: "What do X, Y, Z have in common?". (Show X, Y and Z with single picture slides and make sure to give short descriptions. Go quickly through the single-picture slides.)
- Answer: "They can all be modelled by a..." high-contrast diffusion problem for $u, u_D \in H^1(\Omega)$, $f \in L^2(\Omega)$ and $\mathcal{C} \in L^\infty(\Omega)$

$$\begin{aligned} -\nabla \cdot (\mathcal{C}\nabla u) &= f \quad \text{in } \Omega \\ u &= u_D \quad \text{on } \partial\Omega \end{aligned}$$

- Understanding the problem: We need to find $u \in H^1(\Omega)$ such that the above equation holds.
- Briefly explain why discretization is necessary for PDEs, for non-experts.
- FEM discretization $\Omega \rightarrow \mathcal{T}$, $\nabla \rightarrow A$, $u \rightarrow \mathbf{u}$, $f \rightarrow \mathbf{b}$

Opening

Motivation

- Opening question: "What do X, Y, Z have in common?". (Show X, Y and Z with single picture slides and make sure to give short descriptions. Go quickly through the single-picture slides.)
- Answer: "They can all be modelled by a..." high-contrast diffusion problem for $u, u_D \in H^1(\Omega)$, $f \in L^2(\Omega)$ and $\mathcal{C} \in L^\infty(\Omega)$

$$\begin{aligned} -\nabla \cdot (\mathcal{C}\nabla u) &= f \quad \text{in } \Omega \\ u &= u_D \quad \text{on } \partial\Omega \end{aligned}$$

- Understanding the problem: We need to find $u \in H^1(\Omega)$ such that the above equation holds.
- Briefly explain why discretization is necessary for PDEs, for non-experts.
- FEM discretization $\Omega \rightarrow \mathcal{T}$, $\nabla \rightarrow A$, $u \rightarrow \mathbf{u}$, $f \rightarrow \mathbf{b}$
- Linear system of equations $A\mathbf{u} = \mathbf{b}$

Opening

Motivation

- Opening question: "What do X, Y, Z have in common?". (Show X, Y and Z with single picture slides and make sure to give short descriptions. Go quickly through the single-picture slides.)
- Answer: "They can all be modelled by a..." high-contrast diffusion problem for $u, u_D \in H^1(\Omega)$, $f \in L^2(\Omega)$ and $\mathcal{C} \in L^\infty(\Omega)$

$$\begin{aligned} -\nabla \cdot (\mathcal{C}\nabla u) &= f \quad \text{in } \Omega \\ u &= u_D \quad \text{on } \partial\Omega \end{aligned}$$

- Understanding the problem: We need to find $u \in H^1(\Omega)$ such that the above equation holds.
- Briefly explain why discretization is necessary for PDEs, for non-experts.
- FEM discretization $\Omega \rightarrow \mathcal{T}$, $\nabla \rightarrow A$, $u \rightarrow \mathbf{u}$, $f \rightarrow \mathbf{b}$
- Linear system of equations $A\mathbf{u} = \mathbf{b}$
- We can compute approximations $\mathbf{u}_1, \mathbf{u}_2, \ldots$ efficiently using the CG method.

Opening

Motivation

- Opening question: "What do X, Y, Z have in common?". (Show X, Y and Z with single picture slides and make sure to give short descriptions. Go quickly through the single-picture slides.)
- Answer: "They can all be modelled by a..." high-contrast diffusion problem for $u, u_D \in H^1(\Omega)$, $f \in L^2(\Omega)$ and $\mathcal{C} \in L^\infty(\Omega)$

$$\begin{aligned} -\nabla \cdot (\mathcal{C}\nabla u) &= f \quad \text{in } \Omega \\ u &= u_D \quad \text{on } \partial\Omega \end{aligned}$$

- Understanding the problem: We need to find $u \in H^1(\Omega)$ such that the above equation holds.
- Briefly explain why discretization is necessary for PDEs, for non-experts.
- FEM discretization $\Omega \rightarrow \mathcal{T}$, $\nabla \rightarrow A$, $u \rightarrow \mathbf{u}$, $f \rightarrow \mathbf{b}$
- Linear system of equations $A\mathbf{u} = \mathbf{b}$
- We can compute approximations $\mathbf{u}_1, \mathbf{u}_2, \ldots$ efficiently using the CG method.
- Main R.Q.: "How can we improve existing estimates on the total number of necessary CG iterations?"

Structure

- Introducing CG
- How Does CG Converge? The Role of Eigenvalues
- Preconditioning: Taming High-Contrast Problems
- High-contrast coefficients: split eigenspectrum
- Towards Sharper Iteration Bounds
- Multi-Cluster Spectra
- How Sharp Are the New Bounds?
- New Bounds in Practice: Using Ritz Values
- Conclusion: Key Takeaways & Future Directions

Introducing CG

- CG takes in initial guess $\mathbf{u}_0$ and provides updated solution $\mathbf{u}_k$ after k iterations.

Introducing CG

- CG takes in initial guess $\mathbf{u}_0$ and provides updated solution $\mathbf{u}_k$ after k iterations.
- Classical bound given relative error tolerance $\epsilon_m = \frac{\|\mathbf{e}_m\|_A}{\|\mathbf{e}_0\|_A} \leq \epsilon$ is

  $\epsilon_m \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^m \longrightarrow m \leq \left\lfloor \frac{\sqrt{\kappa}}{2} \ln\left(\frac{2}{\epsilon}\right) + 1 \right\rfloor = m_1(\kappa)$

Introducing CG

- CG takes in initial guess $\mathbf{u}_0$ and provides updated solution $\mathbf{u}_k$ after k iterations.
- Classical bound given relative error tolerance $\epsilon_m = \frac{\|\mathbf{e}_m\|_A}{\|\mathbf{e}_0\|_A} \leq \epsilon$ is

  $\epsilon_m \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^m \longrightarrow m \leq \left\lfloor \frac{\sqrt{\kappa}}{2} \ln\left(\frac{2}{\epsilon}\right) + 1 \right\rfloor = m_1(\kappa)$
- However m_1 can be too pessimistic for high-contrast problems. That is, $m \ll m_1$

Introducing CG

- CG takes in initial guess $\mathbf{u}_0$ and provides updated solution $\mathbf{u}_k$ after k iterations.
- Classical bound given relative error tolerance $\epsilon_m = \frac{\|\mathbf{e}_m\|_A}{\|\mathbf{e}_0\|_A} \leq \epsilon$ is

  $\epsilon_m \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^m \longrightarrow m \leq \left\lfloor \frac{\sqrt{\kappa}}{2} \ln\left(\frac{2}{\epsilon}\right) + 1 \right\rfloor = m_1(\kappa)$
- However m_1 can be too pessimistic for high-contrast problems. That is, $m \ll m_1$
- Restate R.Q.: "How do we improve/sharpen the classical bound m_1 for model problem?"

How Does CG Converge?

The Role of Eigenvalues

- When discussing eigenvalues, a simple visual or analogy could help non-mathematicians.

How Does CG Converge?

The Role of Eigenvalues

- When discussing eigenvalues, a simple visual or analogy could help non-mathematicians.
- Spectrum $\sigma(A) = \{\lambda_1, \ldots, \lambda_n\}$ and $\kappa(A) = \frac{\lambda_{\max}}{\lambda_{\min}}$

How Does CG Converge?

The Role of Eigenvalues

- When discussing eigenvalues, a simple visual or analogy could help non-mathematicians.
- Spectrum $\sigma(A) = \{\lambda_1, \ldots, \lambda_n\}$ and $\kappa(A) = \frac{\lambda_{\max}}{\lambda_{\min}}$
- CG solution given as $\mathbf{u}_m = \mathbf{u}_0 + q(A)\mathbf{r}_0$, with polynomial q being the *solution polynomial*.

How Does CG Converge?

The Role of Eigenvalues

- When discussing eigenvalues, a simple visual or analogy could help non-mathematicians.
- Spectrum $\sigma(A) = \{\lambda_1, \ldots, \lambda_n\}$ and $\kappa(A) = \frac{\lambda_{\max}}{\lambda_{\min}}$
- CG solution given as $\mathbf{u}_m = \mathbf{u}_0 + q(A)\mathbf{r}_0$, with polynomial q being the *solution polynomial*.
- CG residual polynomial $r(\lambda) = 1 - \lambda q(\lambda)$

How Does CG Converge?

The Role of Eigenvalues

- When discussing eigenvalues, a simple visual or analogy could help non-mathematicians.
- Spectrum $\sigma(A) = \{\lambda_1, \ldots, \lambda_n\}$ and $\kappa(A) = \frac{\lambda_{\max}}{\lambda_{\min}}$
- CG solution given as $\mathbf{u}_m = \mathbf{u}_0 + q(A)\mathbf{r}_0$, with polynomial q being the *solution polynomial*.
- CG residual polynomial $r(\lambda) = 1 - \lambda q(\lambda)$
- Error expression $\epsilon_m = \frac{\|\mathbf{e}_m\|_A}{\|\mathbf{e}_0\|_A} < \min_{r \in \mathcal{P}_{m-1}, r(0)=1} \max_{\lambda \in \sigma(A)} |r(\lambda)|$

How Does CG Converge?

The Role of Eigenvalues

- When discussing eigenvalues, a simple visual or analogy could help non-mathematicians.
- Spectrum $\sigma(A) = \{\lambda_1, \ldots, \lambda_n\}$ and $\kappa(A) = \frac{\lambda_{\max}}{\lambda_{\min}}$
- CG solution given as $\mathbf{u}_m = \mathbf{u}_0 + q(A)\mathbf{r}_0$, with polynomial q being the *solution polynomial*.
- CG residual polynomial $r(\lambda) = 1 - \lambda q(\lambda)$
- Error expression $\epsilon_m = \frac{\|\mathbf{e}_m\|_A}{\|\mathbf{e}_0\|_A} < \min_{r \in \mathcal{P}_{m-1}, r(0)=1} \max_{\lambda \in \sigma(A)} |r(\lambda)|$
- Animation: cg_residual_poly
  - Loop through different randomized, clustered spectra, showing r_m for each.
  - Show best and worst case scenario's for CG convergence

How Does CG Converge?

The Role of Eigenvalues

- When discussing eigenvalues, a simple visual or analogy could help non-mathematicians.
- Spectrum $\sigma(A) = \{\lambda_1, \ldots, \lambda_n\}$ and $\kappa(A) = \frac{\lambda_{\max}}{\lambda_{\min}}$
- CG solution given as $\mathbf{u}_m = \mathbf{u}_0 + q(A)\mathbf{r}_0$, with polynomial q being the *solution polynomial*.
- CG residual polynomial $r(\lambda) = 1 - \lambda q(\lambda)$
- Error expression $\epsilon_m = \frac{\|\mathbf{e}_m\|_A}{\|\mathbf{e}_0\|_A} < \min_{r \in \mathcal{P}_{m-1}, r(0)=1} \max_{\lambda \in \sigma(A)} |r(\lambda)|$
- Animation: cg_residual_poly
  - Loop through different randomized, clustered spectra, showing r_m for each.
  - Show best and worst case scenario's for CG convergence
- Restate R.Q.: "How can we construct an CG iteration bound that accounts for the influence of the eigenvalue distribution?"

Preconditioning

Taming High-Contrast Problems

- Why do we want to account for eigenvalue distribution? (practical motivation)

Preconditioning

Taming High-Contrast Problems

- Why do we want to account for eigenvalue distribution? (practical motivation)
- High-contrast C leads to spectral gap and $\kappa(A) \gg 1$

Preconditioning

Taming High-Contrast Problems

- Why do we want to account for eigenvalue distribution? (practical motivation)
- High-contrast C leads to spectral gap and $\kappa(A) \gg 1$
- Clarify what ”preconditioning” means in practical terms (e.g., analogy to making a problem easier to solve).

Preconditioning

Taming High-Contrast Problems

- Why do we want to account for eigenvalue distribution? (practical motivation)
- High-contrast C leads to spectral gap and $\kappa(A) \gg 1$
- Clarify what ”preconditioning” means in practical terms (e.g., analogy to making a problem easier to solve).
- Mathematically: transform $A\mathbf{u} = \mathbf{b}$ into $M^{-1}A\mathbf{u} = M^{-1}\mathbf{b}$, with preconditioner M. This leads to a reduced spectral gap and lower conditioner number. Then, apply CG as usual (now called Preconditioned CG or PCG)

Preconditioning

Taming High-Contrast Problems

- Why do we want to account for eigenvalue distribution? (practical motivation)
- High-contrast C leads to spectral gap and $\kappa(A) \gg 1$
- Clarify what ”preconditioning” means in practical terms (e.g., analogy to making a problem easier to solve).
- Mathematically: transform $A\mathbf{u} = \mathbf{b}$ into $M^{-1}A\mathbf{u} = M^{-1}\mathbf{b}$, with preconditioner M. This leads to a reduced spectral gap and lower conditioner number. Then, apply CG as usual (now called Preconditioned CG or PCG)
- Show Filipe & Alexander’s Research: they used three kinds of preconditioners M_1, M_2, M_3 and all had similar conditioner numbers and thereby similar upper bounds for their iteration number, $m_1(\kappa)$.

Preconditioning

Taming High-Contrast Problems

- Why do we want to account for eigenvalue distribution? (practical motivation)
- High-contrast C leads to spectral gap and $\kappa(A) \gg 1$
- Clarify what ”preconditioning” means in practical terms (e.g., analogy to making a problem easier to solve).
- Mathematically: transform $A\mathbf{u} = \mathbf{b}$ into $M^{-1}A\mathbf{u} = M^{-1}\mathbf{b}$, with preconditioner M. This leads to a reduced spectral gap and lower conditioner number. Then, apply CG as usual (now called Preconditioned CG or PCG)
- Show Filipe & Alexander’s Research: they used three kinds of preconditioners M_1, M_2, M_3 and all had similar conditioner numbers and thereby similar upper bounds for their iteration number, $m_1(\kappa)$.
- However, on a sample problem PCG with M_1, M_2 needed significantly less iterations than PCG with M_3.

Preconditioning

Taming High-Contrast Problems

- Why do we want to account for eigenvalue distribution? (practical motivation)
- High-contrast C leads to spectral gap and $\kappa(A) \gg 1$
- Clarify what ”preconditioning” means in practical terms (e.g., analogy to making a problem easier to solve).
- Mathematically: transform $A\mathbf{u} = \mathbf{b}$ into $M^{-1}A\mathbf{u} = M^{-1}\mathbf{b}$, with preconditioner M. This leads to a reduced spectral gap and lower conditioner number. Then, apply CG as usual (now called Preconditioned CG or PCG)
- Show Filipe & Alexander’s Research: they used three kinds of preconditioners M_1, M_2, M_3 and all had similar conditioner numbers and thereby similar upper bounds for their iteration number, $m_1(\kappa)$.
- However, on a sample problem PCG with M_1, M_2 needed significantly less iterations than PCG with M_3.
- Restate R.Q.: ”How can we construct a CG iteration bound that can distinguish between different preconditioners with similar conditioner numbers?”

Towards Sharper Iteration Bounds

The Two-clusters case

- Recap: $\epsilon_m < \min_{r \in \mathcal{P}_{m-1}, r(0)=1} \max_{\lambda \in \sigma(A)} |r(\lambda)| \leq \epsilon$

Towards Sharper Iteration Bounds

The Two-clusters case

- Recap: $\epsilon_m < \min_{r \in \mathcal{P}_{m-1}, r(0)=1} \max_{\lambda \in \sigma(A)} |r(\lambda)| \leq \epsilon$
- Head-on strategy for a bound: try to find an m^{th}-order polynomial that satisfies the min-max problem.

Towards Sharper Iteration Bounds

The Two-clusters case

- Recap: $\epsilon_m < \min_{r \in \mathcal{P}_{m-1}, r(0)=1} \max_{\lambda \in \sigma(A)} |r(\lambda)| \leq \epsilon$
- Head-on strategy for a bound: try to find an m^{th}-order polynomial that satisfies the min-max problem.
- Classical bound invokes: optimal Chebyshev polynomial $r(\lambda) = \hat{C}_m$, under the assumption of a uniform interval of eigenvalues $\sigma(A) = [\lambda_{\min}, \lambda_{\max}]$ (inaccurate!).

Towards Sharper Iteration Bounds

The Two-clusters case

- Recap: $\epsilon_m < \min_{r \in \mathcal{P}_{m-1}, r(0)=1} \max_{\lambda \in \sigma(A)} |r(\lambda)| \leq \epsilon$
- Head-on strategy for a bound: try to find an m^{th}-order polynomial that satisfies the min-max problem.
- Classical bound invokes: optimal Chebyshev polynomial $r(\lambda) = \hat{C}_m$, under the assumption of a uniform interval of eigenvalues $\sigma(A) = [\lambda_{\min}, \lambda_{\max}]$ (inaccurate!).
- Simplest non-trivial case (one spectral gap): two disjoint clusters

Towards Sharper Iteration Bounds

The Two-clusters case

- Recap: $\epsilon_m < \min_{r \in \mathcal{P}_{m-1}, r(0)=1} \max_{\lambda \in \sigma(A)} |r(\lambda)| \leq \epsilon$
- Head-on strategy for a bound: try to find an m^{th}-order polynomial that satisfies the min-max problem.
- Classical bound invokes: optimal Chebyshev polynomial $r(\lambda) = \hat{C}_m$, under the assumption of a uniform interval of eigenvalues $\sigma(A) = [\lambda_{\min}, \lambda_{\max}]$ (inaccurate!).
- Simplest non-trivial case (one spectral gap): two disjoint clusters
- Axelsson, assume two uniform intervals and invoke $r(\lambda) = \hat{C}_p \hat{C}_{p-m}$

Towards Sharper Iteration Bounds

The Two-clusters case

- Recap: $\epsilon_m < \min_{r \in \mathcal{P}_{m-1}, r(0)=1} \max_{\lambda \in \sigma(A)} |r(\lambda)| \leq \epsilon$
- Head-on strategy for a bound: try to find an m^{th}-order polynomial that satisfies the min-max problem.
- Classical bound invokes: optimal Chebyshev polynomial $r(\lambda) = \hat{C}_m$, under the assumption of a uniform interval of eigenvalues $\sigma(A) = [\lambda_{\min}, \lambda_{\max}]$ (inaccurate!).
- Simplest non-trivial case (one spectral gap): two disjoint clusters
- Axelsson, assume two uniform intervals and invoke $r(\lambda) = \hat{C}_p \hat{C}_{p-m}$
- Resulting two-cluster bound $m_2(\kappa, \kappa_l, \kappa_r) = \left\lfloor \frac{\sqrt{\kappa_r}}{2} \ln(2/\epsilon) + \left(1 + \frac{\sqrt{\kappa_r}}{2} \ln\left(\frac{4\kappa}{\kappa_l}\right)\right) p \right\rfloor$

Towards Sharper Iteration Bounds

The Two-clusters case

- Recap: $\epsilon_m < \min_{r \in \mathcal{P}_{m-1}, r(0)=1} \max_{\lambda \in \sigma(A)} |r(\lambda)| \leq \epsilon$
- Head-on strategy for a bound: try to find an m^{th}-order polynomial that satisfies the min-max problem.
- Classical bound invokes: optimal Chebyshev polynomial $r(\lambda) = \hat{C}_m$, under the assumption of a uniform interval of eigenvalues $\sigma(A) = [\lambda_{\min}, \lambda_{\max}]$ (inaccurate!).
- Simplest non-trivial case (one spectral gap): two disjoint clusters
- Axelsson, assume two uniform intervals and invoke $r(\lambda) = \hat{C}_p \hat{C}_{p-m}$
- Resulting two-cluster bound $m_2(\kappa, \kappa_l, \kappa_r) = \left\lfloor \frac{\sqrt{\kappa_r}}{2} \ln(2/\epsilon) + \left(1 + \frac{\sqrt{\kappa_r}}{2} \ln\left(\frac{4\kappa}{\kappa_l}\right)\right) p \right\rfloor$
- Restate R.Q.: ”How can we construct an CG iteration bound that accounts for the influence of the eigenvalue distribution for general spectra?”

Towards Sharper Iteration Bounds

The Two-clusters case

- Recap: $\epsilon_m < \min_{r \in \mathcal{P}_{m-1}, r(0)=1} \max_{\lambda \in \sigma(A)} |r(\lambda)| \leq \epsilon$
- Head-on strategy for a bound: try to find an m^{th}-order polynomial that satisfies the min-max problem.
- Classical bound invokes: optimal Chebyshev polynomial $r(\lambda) = \hat{C}_m$, under the assumption of a uniform interval of eigenvalues $\sigma(A) = [\lambda_{\min}, \lambda_{\max}]$ (inaccurate!).
- Simplest non-trivial case (one spectral gap): two disjoint clusters
- Axelsson, assume two uniform intervals and invoke $r(\lambda) = \hat{C}_p \hat{C}_{p-m}$
- Resulting two-cluster bound $m_2(\kappa, \kappa_l, \kappa_r) = \left\lfloor \frac{\sqrt{\kappa_r}}{2} \ln(2/\epsilon) + \left(1 + \frac{\sqrt{\kappa_r}}{2} \ln\left(\frac{4\kappa}{\kappa_l}\right)\right) p \right\rfloor$
- Restate R.Q.: "How can we construct an CG iteration bound that accounts for the influence of the eigenvalue distribution for general spectra?"
- Note: The transition from classical bounds to multi-cluster bounds could be made more explicit—perhaps with a summary slide or visual.

Multi-Cluster Spectra

- Depending on coefficient function and preconditioned systems we can have any number of clusters.

Multi-Cluster Spectra

- Depending on coefficient function and preconditioned systems we can have any number of clusters.
- Extend Axelsson idea: $r(\lambda) = \hat{c}_{p_1}^{(1)} \hat{c}_{p_2}^{(2)} \dots \hat{c}_{p_k}^{(k)} = \prod_{i=1}^{k} \hat{c}_{p_i}^{(i)}$

Multi-Cluster Spectra

- Depending on coefficient function and preconditioned systems we can have any number of clusters.
- Extend Axelsson idea: $r(\lambda) = \hat{C}_{p_1}^{(1)} \hat{C}_{p_2}^{(2)} \dots \hat{C}_{p_k}^{(k)} = \prod_{i=1}^{k} \hat{C}_{p_i}^{(i)}$
- Multi-cluster bound: $p_i \leq \left\lceil \log_{f_i} \frac{\epsilon}{2} + \sum_{j=1}^{i-1} p_j \log_{f_i} \left(\frac{\zeta_2^{(j)}}{\zeta_1^{(i,j)}} \right) \right\rceil$ and $m_{N_{\text{cluster}}} = \sum_{i=1}^{N_{\text{cluster}}} p_i$, depends on the extremal eigenvalues of *each cluster*

Multi-Cluster Spectra

- Depending on coefficient function and preconditioned systems we can have any number of clusters.
- Extend Axelsson idea: $r(\lambda) = \hat{C}_{p_1}^{(1)} \hat{C}_{p_2}^{(2)} \dots \hat{C}_{p_k}^{(k)} = \prod_{i=1}^{k} \hat{C}_{p_i}^{(i)}$
- Multi-cluster bound: $p_i \leq \left\lceil \log_{f_i} \frac{\epsilon}{2} + \sum_{j=1}^{i-1} p_j \log_{f_i} \left(\frac{\zeta_2^{(j)}}{\zeta_1^{(i,j)}} \right) \right\rceil$ and $m_{N_{\text{cluster}}} = \sum_{i=1}^{N_{\text{cluster}}} p_i$, depends on the extremal eigenvalues of *each cluster*
- We need a way of partitioning a given spectrum $\sigma(A)$ into N_{cluster} clusters.

Multi-Cluster Spectra

- Depending on coefficient function and preconditioned systems we can have any number of clusters.
- Extend Axelsson idea: $r(\lambda) = \hat{C}_{p_1}^{(1)} \hat{C}_{p_2}^{(2)} \dots \hat{C}_{p_k}^{(k)} = \prod_{i=1}^{k} \hat{C}_{p_i}^{(i)}$
- Multi-cluster bound: $p_i \leq \left\lceil \log_{f_i} \frac{\epsilon}{2} + \sum_{j=1}^{i-1} p_j \log_{f_i} \left(\frac{\zeta_2^{(j)}}{\zeta_1^{(i,j)}} \right) \right\rceil$ and $m_{N_{\text{cluster}}} = \sum_{i=1}^{N_{\text{cluster}}} p_i$, depends on the extremal eigenvalues of *each cluster*
- We need a way of partitioning a given spectrum $\sigma(A)$ into N_{cluster} clusters.
- Idea: simple, we split at the largest (relative) gap between subsequent eigenvalues, check if the two resulting clusters would give a sharper bound than just one uniform cluster, that is $m_2(\kappa, \kappa_l, \kappa_r) < m_1(\kappa)$. If so, we split the clusters and repeat the process for each created cluster. If not, we stop partitioning. In the process we keep track of all the extremal eigenvalues. After partitioning, we calculate $m_{N_{\text{cluster}}}$.

Multi-Cluster Spectra

- Depending on coefficient function and preconditioned systems we can have any number of clusters.
- Extend Axelsson idea: $r(\lambda) = \hat{C}_{p_1}^{(1)} \hat{C}_{p_2}^{(2)} \dots \hat{C}_{p_k}^{(k)} = \prod_{i=1}^{k} \hat{C}_{p_i}^{(i)}$
- Multi-cluster bound: $p_i \leq \left\lceil \log_{f_i} \frac{\epsilon}{2} + \sum_{j=1}^{i-1} p_j \log_{f_i} \left(\frac{\zeta_2^{(j)}}{\zeta_1^{(i,j)}} \right) \right\rceil$ and $m_{N_{\text{cluster}}} = \sum_{i=1}^{N_{\text{cluster}}} p_i$, depends on the extremal eigenvalues of *each cluster*
- We need a way of partitioning a given spectrum $\sigma(A)$ into N_{cluster} clusters.
- Idea: simple, we split at the largest (relative) gap between subsequent eigenvalues, check if the two resulting clusters would give a sharper bound than just one uniform cluster, that is $m_2(\kappa, \kappa_l, \kappa_r) < m_1(\kappa)$. If so, we split the clusters and repeat the process for each created cluster. If not, we stop partitioning. In the process we keep track of all the extremal eigenvalues. After partitioning, we calculate $m_{N_{\text{cluster}}}$.
- Animation: cluster_partitioning, visualize above partitioning and subsequent calculation $m_{N_{\text{cluster}}}$.

Multi-Cluster Spectra

- Depending on coefficient function and preconditioned systems we can have any number of clusters.
- Extend Axelsson idea: $r(\lambda) = \hat{C}_{p_1}^{(1)} \hat{C}_{p_2}^{(2)} \dots \hat{C}_{p_k}^{(k)} = \prod_{i=1}^{k} \hat{C}_{p_i}^{(i)}$
- Multi-cluster bound: $p_i \leq \left\lceil \log_{f_i} \frac{\epsilon}{2} + \sum_{j=1}^{i-1} p_j \log_{f_i} \left(\frac{\zeta_2^{(j)}}{\zeta_1^{(i,j)}} \right) \right\rceil$ and $m_{N_{\text{cluster}}} = \sum_{i=1}^{N_{\text{cluster}}} p_i$, depends on the extremal eigenvalues of *each cluster*
- We need a way of partitioning a given spectrum $\sigma(A)$ into N_{cluster} clusters.
- Idea: simple, we split at the largest (relative) gap between subsequent eigenvalues, check if the two resulting clusters would give a sharper bound than just one uniform cluster, that is $m_2(\kappa, \kappa_l, \kappa_r) < m_1(\kappa)$. If so, we split the clusters and repeat the process for each created cluster. If not, we stop partitioning. In the process we keep track of all the extremal eigenvalues. After partitioning, we calculate $m_{N_{\text{cluster}}}$.
- Animation: cluster_partitioning, visualize above partitioning and subsequent calculation $m_{N_{\text{cluster}}}$.
- Restate R.Q.: "How much sharper is $m_{N_{\text{cluster}}}$ compared to $m_1(\kappa)$ for our model problem?"

How Sharp Are the New Bounds?

- We implement the model problem with the two coefficient functions that Filipe and Alexander used and solve them using PCG with preconditioners M_1, M_2, M_3.

How Sharp Are the New Bounds?

- We implement the model problem with the two coefficient functions that Filipe and Alexander used and solve them using PCG with preconditioners M_1, M_2, M_3.
- First experiment: For each $\sigma(M_i^{-1}A)$, we compute $m_{N_{\text{cluster}}}$ and $m_1(\kappa)$ for comparison. As well as, $m_{N_{\text{tail-cluster}}}$, a variant of $m_{N_{\text{cluster}}}$ that can take in clusters as well as individual eigenvalues.

How Sharp Are the New Bounds?

- We implement the model problem with the two coefficient functions that Filipe and Alexander used and solve them using PCG with preconditioners M_1, M_2, M_3.
- First experiment: For each $\sigma(M_i^{-1}A)$, we compute $m_{N_{\text{cluster}}}$ and $m_1(\kappa)$ for comparison. As well as, $m_{N_{\text{tail-cluster}}}$, a variant of $m_{N_{\text{cluster}}}$ that can take in clusters as well as individual eigenvalues.
- Results: the new bounds are incredibly sharp as they are anywhere from 2 to 4 times as big as m. For comparison, m_1 is anywhere from 100 to 1000 times bigger than m, where the difference increases for spectra with larger spectral gaps

How Sharp Are the New Bounds?

- We implement the model problem with the two coefficient functions that Filipe and Alexander used and solve them using PCG with preconditioners M_1, M_2, M_3.
- First experiment: For each $\sigma(M_i^{-1}A)$, we compute $m_{N_{\text{cluster}}}$ and $m_1(\kappa)$ for comparison. As well as, $m_{N_{\text{tail-cluster}}}$, a variant of $m_{N_{\text{cluster}}}$ that can take in clusters as well as individual eigenvalues.
- Results: the new bounds are incredibly sharp as they are anywhere from 2 to 4 times as big as m. For comparison, m_1 is anywhere from 100 to 1000 times bigger than m, where the difference increases for spectra with larger spectral gaps
- Restate R.Q.: "How can we compute $m_{N_{\text{cluster}}}$ for our model problem in practice?"

New Bounds in Practice:

Using Ritz Values

- Problem: computing $m_{N_{\text{cluster}}}$ requires a good estimate of $\sigma(M_i^{-1}A)$.

New Bounds in Practice:

Using Ritz Values

- Problem: computing $m_{N_{\text{cluster}}}$ requires a good estimate of $\sigma(M_i^{-1}A)$.
- Luckily, PCG gives us exactly that, every iteration it produces a set of so-called Ritz values that is 'close' to the true eigenvalue distribution. In particular, at the k^{th} iteration, we get a set of k Ritz values that we can use as an approximation of the full spectrum $\sigma(M_i^{-1}A)$.

New Bounds in Practice:

Using Ritz Values

- Problem: computing $m_{N_{\text{cluster}}}$ requires a good estimate of $\sigma(M_i^{-1}A)$.
- Luckily, PCG gives us exactly that, every iteration it produces a set of so-called Ritz values that is 'close' to the true eigenvalue distribution. In particular, at the k^{th} iteration, we get a set of k Ritz values that we can use as an approximation of the full spectrum $\sigma(M_i^{-1}A)$.
- Second experiment, we use the Ritz values from the PCG iterations to compute $m_{N_{\text{cluster}}}$ and $m_1(\kappa)$ for our model problem and observe how good of an upper bound for m we can obtain within the first 300 iterations.

New Bounds in Practice:

Using Ritz Values

- Problem: computing $m_{N_{\text{cluster}}}$ requires a good estimate of $\sigma(M_i^{-1}A)$.
- Luckily, PCG gives us exactly that, every iteration it produces a set of so-called Ritz values that is 'close' to the true eigenvalue distribution. In particular, at the k^{th} iteration, we get a set of k Ritz values that we can use as an approximation of the full spectrum $\sigma(M_i^{-1}A)$.
- Second experiment, we use the Ritz values from the PCG iterations to compute $m_{N_{\text{cluster}}}$ and $m_1(\kappa)$ for our model problem and observe how good of an upper bound for m we can obtain within the first 300 iterations.
- Results: For PCG with preconditioners M_1, M_2 (fast converging) $m_{N_{\text{cluster}}}$ gives sharp upper bounds for m in all cases. However, for PCG with M_3 (slow converging) $m_{N_{\text{cluster}}}$ underestimates m in some cases. This is because the Ritz values do not yet capture the full spectrum well enough within 300 iterations.

New Bounds in Practice:

Using Ritz Values

- Problem: computing $m_{N_{\text{cluster}}}$ requires a good estimate of $\sigma(M_i^{-1}A)$.
- Luckily, PCG gives us exactly that, every iteration it produces a set of so-called Ritz values that is 'close' to the true eigenvalue distribution. In particular, at the k^{th} iteration, we get a set of k Ritz values that we can use as an approximation of the full spectrum $\sigma(M_i^{-1}A)$.
- Second experiment, we use the Ritz values from the PCG iterations to compute $m_{N_{\text{cluster}}}$ and $m_1(\kappa)$ for our model problem and observe how good of an upper bound for m we can obtain within the first 300 iterations.
- Results: For PCG with preconditioners M_1, M_2 (fast converging) $m_{N_{\text{cluster}}}$ gives sharp upper bounds for m in all cases. However, for PCG with M_3 (slow converging) $m_{N_{\text{cluster}}}$ underestimates m in some cases. This is because the Ritz values do not yet capture the full spectrum well enough within 300 iterations.
- Animation: ritz_value_migration