Iterative solutions to sequences of Helmholtz equations

Thesis presentation

Jan de Gier
29 august 2012

TUDelft

Outline

- Introduction
  - Problem description
  - Physical problem
  - Mathematical problem
  - Solution to a test problem
- Solving the mathematical problem
  - IDR(s)
  - Shifted Laplace Preconditioning
- Reducing the computation time
  - Using previous solutions
  - Updating the preconditioner
  - Using spectral information
- Conclusions and future research

TUDelft

Introduction
Problem description

The reduction of noise in a car.

- ◇ Noise is caused by the engine, road contact and head wind.

Model the car and the propagation of the acoustic waves.
Solve the resulting problem for sequencies of frequencies.

Use information of earlier obtained solutions for speeding up the computations.

- ◇ Solution vectors, spectral information and information on the performance of the algorithm.

TU Delft

Introduction
Problem description

The reduction of noise in a car.

- ◇ Noise is caused by the engine, road contact and head wind.

Model the car and the propagation of the acoustic waves.
Solve the resulting problem for sequencies of frequencies.

Use information of earlier obtained solutions for speeding up the computations.

- ◇ Solution vectors, spectral information and information on the performance of the algorithm.

Introduction

Problem description

The reduction of noise in a car.

- ◇ Noise is caused by the engine, road contact and head wind.

Model the car and the propagation of the acoustic waves.
Solve the resulting problem for sequencies of frequencies.

Use information of earlier obtained solutions for speeding up the computations.

- ◇ Solution vectors, spectral information and information on the performance of the algorithm.

Introduction

Physical problem

Sound generates small disturbances in the ambient pressure p.

The wave equation:

$$\frac{1}{c_0^2}\frac{\partial^2}{\partial t^2}p(\mathbf{x},t) - \nabla^2 p(\mathbf{x},t) = s(\mathbf{x},t).$$

The Helmholtz equation:

$$-k^2 P(\mathbf{x}) - \nabla^2 P(\mathbf{x}) = S(\mathbf{x}), \quad \text{with} \quad k = f\frac{2\pi}{c_0}.$$

Introduction

Physical problem

Sound generates small disturbances in the ambient pressure p.

The wave equation:

$$\frac{1}{c_0^2}\frac{\partial^2}{\partial t^2}p(\mathbf{x},t) - \nabla^2 p(\mathbf{x},t) = s(\mathbf{x},t).$$

The Helmholtz equation:

$$-k^2 P(\mathbf{x}) - \nabla^2 P(\mathbf{x}) = S(\mathbf{x}), \quad \text{with} \quad k = f\frac{2\pi}{c_0}.$$

TUDelft

Introduction
Physical problem

Sound generates small disturbances in the ambient pressure p.

The wave equation:

$$\frac{1}{c_0^2}\frac{\partial^2}{\partial t^2}p(\mathbf{x},t) - \nabla^2 p(\mathbf{x},t) = s(\mathbf{x},t).$$

The Helmholtz equation:

$$-k^2 P(\mathbf{x}) - \nabla^2 P(\mathbf{x}) = S(\mathbf{x}), \quad \text{with} \quad k = f\frac{2\pi}{c_0}.$$

Introduction
Physical problem

The Helmholtz equation on a domain

$$-k^2 P(\mathbf{x}) - \nabla^2 P(\mathbf{x}) = S(\mathbf{x}) \text{ on } \Omega \quad (k = f\ 2\pi/c_0),$$

satisfies the boundary condition

$$Z_n \frac{\partial}{\partial n} P(\mathbf{x}) + ikP(\mathbf{x}) = 0 \text{ on } \Gamma.$$

TU Delft

Introduction

Physical problem

The Helmholtz equation on a domain

$$-k^2 P(\mathbf{x}) - \nabla^2 P(\mathbf{x}) = S(\mathbf{x}) \ \text{ on } \ \Omega \quad (k = f\ 2\pi / c_0),$$

satisfies the boundary condition

$$Z_n \frac{\partial}{\partial n} P(\mathbf{x}) + ikP(\mathbf{x}) = 0 \ \text{ on } \ \Gamma.$$

Introduction

Mathematical problem

$$\begin{cases} -k^2 P(\mathbf{x}) - \nabla^2 P(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{x}_s) & \text{on} \quad \Omega, \\ Z_n \frac{\partial}{\partial n} P(\mathbf{x}) + ikP(\mathbf{x}) = 0 & \text{on} \quad \Gamma. \end{cases}$$

The weak form equals

$$-k^2 \int_\Omega \eta P d\Omega + \int_\Omega \nabla P \cdot \nabla \eta d\Omega + ik \oint_\Gamma \frac{1}{Z_n} \eta P d\Gamma = \int_\Omega \eta \delta(\mathbf{x} - \mathbf{x}_s) d\Omega.$$

The finite element method results in a discretised linear system

$$(-f^2 \mathbf{M} + \mathbf{K} + if\mathbf{C})\mathbf{p} = \mathbf{b}.$$

TUDelft

Introduction

Mathematical problem

$$\begin{cases} -k^2 P(\mathbf{x}) - \nabla^2 P(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{x}_s) & \text{on} \quad \Omega, \\ Z_n \frac{\partial}{\partial n} P(\mathbf{x}) + ikP(\mathbf{x}) = 0 & \text{on} \quad \Gamma. \end{cases}$$

The weak form equals

$$-k^2 \int_\Omega \eta P d\Omega + \int_\Omega \nabla P \cdot \nabla \eta d\Omega + ik \oint_\Gamma \frac{1}{Z_n} \eta P d\Gamma = \int_\Omega \eta \delta(\mathbf{x} - \mathbf{x}_s) d\Omega.$$

The finite element method results in a discretised linear system

$$(-f^2 \mathbf{M} + \mathbf{K} + if\mathbf{C})\mathbf{p} = \mathbf{b}.$$

TU Delft

Introduction

Mathematical problem

$$\begin{cases} -k^2 P(\mathbf{x}) - \nabla^2 P(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{x}_s) & \text{on} \quad \Omega, \\ Z_n \frac{\partial}{\partial n} P(\mathbf{x}) + ikP(\mathbf{x}) = 0 & \text{on} \quad \Gamma. \end{cases}$$

The weak form equals

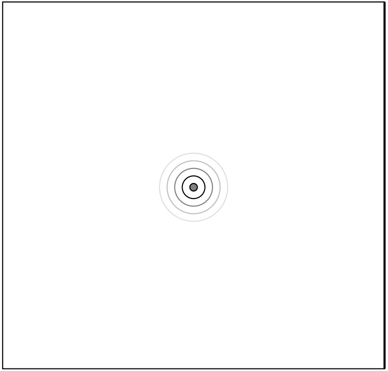
$$-k^2 \int_\Omega \eta P d\Omega + \int_\Omega \nabla P \cdot \nabla \eta d\Omega + ik \oint_\Gamma \frac{1}{Z_n} \eta P d\Gamma = \int_\Omega \eta \delta(\mathbf{x} - \mathbf{x}_s) d\Omega.$$

The finite element method results in a discretised linear system

$$(-f^2 \mathbf{M} + \mathbf{K} + if\mathbf{C})\mathbf{p} = \mathbf{b}.$$

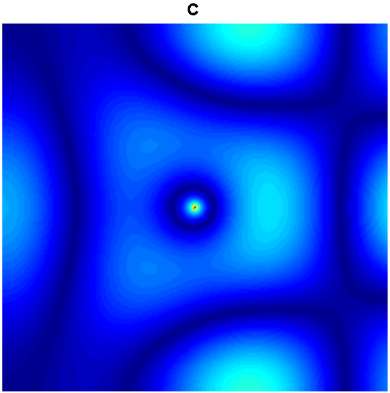
Introduction

Solution to a test problem

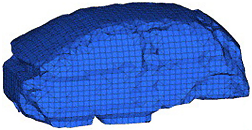


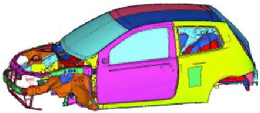
Introduction

Solution to a test problem



Solving the mathematical problem





Solving the mathematical problem

We solve the system

$$\left\{ \begin{pmatrix} \mathbf{K}_s & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_f \end{pmatrix} + if \begin{pmatrix} \mathbf{C}_s & i\mathbf{C}_{sf}^{\top} \\ i\mathbf{C}_{sf} & \mathbf{C}_f \end{pmatrix} - f^2 \begin{pmatrix} \mathbf{M}_s & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_f \end{pmatrix} \right\} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{b}_s \\ \mathbf{b}_f \end{pmatrix}$$

and the (symmetric) system matrix can be written as

$$\mathbf{A}(f) = \mathbf{K} + if\mathbf{C} - f^2\mathbf{M}.$$

TU Delft

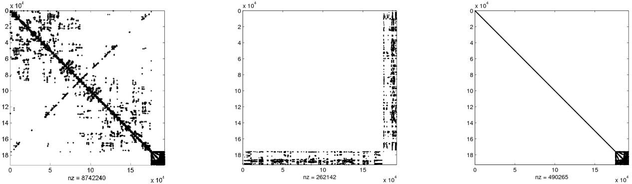
Solving the mathematical problem

We solve the system

$$\left\{ \begin{pmatrix} \mathbf{K}_\mathrm{s} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_\mathrm{f} \end{pmatrix} + if \begin{pmatrix} \mathbf{C}_\mathrm{s} & i\mathbf{C}_\mathrm{sf}^\top \\ i\mathbf{C}_\mathrm{sf} & \mathbf{C}_\mathrm{f} \end{pmatrix} - f^2 \begin{pmatrix} \mathbf{M}_\mathrm{s} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_\mathrm{f} \end{pmatrix} \right\} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{b}_\mathrm{s} \\ \mathbf{b}_\mathrm{f} \end{pmatrix}$$

and the (symmetric) system matrix can be written as

$$\mathbf{A}(f) = \quad \mathbf{K} \quad + \quad if\,\mathbf{C} \quad - \quad f^2\mathbf{M}.$$



Solving the mathematical problem

IDR(s)

IDR(s):

- Iterative method:
  Start with $\mathbf{x}_0$ and determine $\mathbf{x}_1, \mathbf{x}_2, \ldots \to \mathbf{x}$.
- Krylov subspace method:
  $\mathbf{x}_i - \mathbf{x}_0 = P_{i-1}(\mathbf{A})\mathbf{r}_0 \in \mathrm{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots \mathbf{A}^{i-1}\mathbf{r}_0\}$.

$\mathbf{r}_i = \mathbf{b} - \mathbf{A}\mathbf{x}_i$ are the residuals: the amount we are wrong.

$\mathbf{r}_i = R_i(\mathbf{A})\mathbf{r}_0 = [\mathbf{I} - \mathbf{A}P_{i-1}(\mathbf{A})]\mathbf{r}_0$.

Stopping criterium: $\|\mathbf{r}_i\|/\|\mathbf{b}\| \leq 10^{-8}$ (or # iterations ≥ 1000).

Solving the mathematical problem
IDR(s)

IDR(s):

- ◇ Iterative method:
  Start with $\mathbf{x}_0$ and determine $\mathbf{x}_1, \mathbf{x}_2, \ldots \to \mathbf{x}$.
- ◇ Krylov subspace method:
  $\mathbf{x}_i - \mathbf{x}_0 = P_{i-1}(\mathbf{A})\mathbf{r}_0 \in \mathrm{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots \mathbf{A}^{i-1}\mathbf{r}_0\}$.

$\mathbf{r}_i = \mathbf{b} - \mathbf{A}\mathbf{x}_i$ are the residuals: the amount we are wrong.

$\mathbf{r}_i = R_i(\mathbf{A})\mathbf{r}_0 = [\mathbf{I} - \mathbf{A}P_{i-1}(\mathbf{A})]\mathbf{r}_0$.

Stopping criterium: $\|\mathbf{r}_i\|/\|\mathbf{b}\| \leq 10^{-8}$ (or # iterations ≥ 1000).

TU Delft

Solving the mathematical problem
IDR(s)

IDR(s):

- Iterative method:
  Start with $\mathbf{x}_0$ and determine $\mathbf{x}_1, \mathbf{x}_2, \ldots \to \mathbf{x}$.
- Krylov subspace method:
  $\mathbf{x}_i - \mathbf{x}_0 = P_{i-1}(\mathbf{A})\mathbf{r}_0 \in \mathrm{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots \mathbf{A}^{i-1}\mathbf{r}_0\}$.

$\mathbf{r}_i = \mathbf{b} - \mathbf{A}\mathbf{x}_i$ are the residuals: the amount we are wrong.

$\mathbf{r}_i = R_i(\mathbf{A})\mathbf{r}_0 = [\mathbf{I} - \mathbf{A}P_{i-1}(\mathbf{A})]\mathbf{r}_0$.

Stopping criterium: $\|\mathbf{r}_i\|/\|\mathbf{b}\| \leq 10^{-8}$ (or # iterations ≥ 1000).

TUDelft

Solving the mathematical problem
IDR(s)

IDR(s):

- ◇ Iterative method:
  Start with $\mathbf{x}_0$ and determine $\mathbf{x}_1, \mathbf{x}_2, \ldots \to \mathbf{x}$.
- ◇ Krylov subspace method:
  $\mathbf{x}_i - \mathbf{x}_0 = P_{i-1}(\mathbf{A})\mathbf{r}_0 \in \mathrm{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots \mathbf{A}^{i-1}\mathbf{r}_0\}$.

$\mathbf{r}_i = \mathbf{b} - \mathbf{A}\mathbf{x}_i$ are the residuals: the amount we are wrong.

$\mathbf{r}_i = R_i(\mathbf{A})\mathbf{r}_0 = [\mathbf{I} - \mathbf{A}P_{i-1}(\mathbf{A})]\mathbf{r}_0$.

Stopping criterium: $\|\mathbf{r}_i\|/\|\mathbf{b}\| \leq 10^{-8}$ (or # iterations $\geq$ 1000).

TUDelft

Solving the mathematical problem
IDR(s)

IDR(s):

- ◇ Iterative method:
  Start with $\mathbf{x}_0$ and determine $\mathbf{x}_1, \mathbf{x}_2, \ldots \to \mathbf{x}$.
- ◇ Krylov subspace method:
  $\mathbf{x}_i - \mathbf{x}_0 = P_{i-1}(\mathbf{A})\mathbf{r}_0 \in \mathrm{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots \mathbf{A}^{i-1}\mathbf{r}_0\}$.

$\mathbf{r}_i = \mathbf{b} - \mathbf{A}\mathbf{x}_i$ are the residuals: the amount we are wrong.

$\mathbf{r}_i = R_i(\mathbf{A})\mathbf{r}_0 = [\mathbf{I} - \mathbf{A}P_{i-1}(\mathbf{A})]\mathbf{r}_0$.

Stopping criterium: $\|\mathbf{r}_i\|/\|\mathbf{b}\| \leq 10^{-8}$ (or $\#$ iterations ≥ 1000).

TUDelft

Solving the mathematical problem

IDR(s)

Generate residuals $\mathbf{r}_i \in \mathcal{G}_j$, where

- $\diamond$ $\mathcal{G}_0 = \mathcal{K}_n(\mathbf{A}, \mathbf{r}_0) = \mathbb{C}^n$,
- $\diamond$ $\mathcal{G}_{j+1} = (\mathbf{I} - \omega_{j+1}\mathbf{A})(\mathcal{G}_j \cap \mathcal{S})$.

Properties of the Sonneveld spaces $\mathcal{G}_j$ are:

- $\diamond$ $\mathcal{G}_{j+1} \subset \mathcal{G}_j$ and even $\dim(\mathcal{G}_{j+1}) = \dim(\mathcal{G}_j) - s$,
- $\diamond$ $\mathcal{G}_k = \{\mathbf{0}\}$ for a $k \ll n$.

TUDelft

Solving the mathematical problem
IDR(s)

Generate residuals $\mathbf{r}_i \in \mathcal{G}_j$, where

- $\diamond$ $\mathcal{G}_0 = \mathcal{K}_n(\mathbf{A}, \mathbf{r}_0) = \mathbb{C}^n$,
- $\diamond$ $\mathcal{G}_{j+1} = (\mathbf{I} - \omega_{j+1}\mathbf{A})(\mathcal{G}_j \cap \mathcal{S})$.

Properties of the Sonneveld spaces $\mathcal{G}_j$ are:

- $\diamond$ $\mathcal{G}_{j+1} \subset \mathcal{G}_j$ and even $\dim(\mathcal{G}_{j+1}) = \dim(\mathcal{G}_j) - s$,
- $\diamond$ $\mathcal{G}_k = \{\mathbf{0}\}$ for a $k \ll n$.

Solving the mathematical problem

Apply preconditioner: condition the problem into a form that is more suitable for the numerical method.

$$\mathbf{A}\mathbf{x} = \mathbf{b} \quad \rightarrow \quad \mathbf{P}^{-1}\mathbf{A}\mathbf{x} = \mathbf{P}^{-1}\mathbf{b}.$$

$\mathbf{P}$ should preferably be

- a good approximation of $\mathbf{A}$,
- easy to construct and apply.

Solving the mathematical problem

Apply preconditioner: condition the problem into a form that is more suitable for the numerical method.

$$\mathbf{A}\mathbf{x} = \mathbf{b} \quad \rightarrow \quad \mathbf{P}^{-1}\mathbf{A}\mathbf{x} = \mathbf{P}^{-1}\mathbf{b}.$$

$\mathbf{P}$ should preferably be

- ◇ a good approximation of $\mathbf{A}$,
- ◇ easy to construct and apply.

Solving the mathematical problem
Shifted Laplace Preconditioning

The system matrix equals $\mathbf{A}(f) = \mathbf{K} + if\mathbf{C} - f^2\mathbf{M}$,

and we consider the preconditioners

- $\diamond$ $\mathbf{P}^i(f_0) = \mathbf{K} + if_0\mathbf{C} + if_0^2\mathbf{M}$ (imaginary shift),
- $\diamond$ $\mathbf{P}^r(f_0) = \mathbf{K} + if_0\mathbf{C} - f_0^2\mathbf{M}$ (real shift),
- $\diamond$ $\mathbf{P}^m(f_0) = \mathrm{Re}(\mathbf{K}) - f_0^2\mathbf{M}$ (modified real shift).

We use LU decomposition of the preconditioner, such that $\mathbf{Q}_1\ \mathbf{P}^*(f_0)\ \mathbf{Q}_2 = \mathbf{L}\ \mathbf{U}$.

TUDelft

Solving the mathematical problem
Shifted Laplace Preconditioning

The system matrix equals $\mathbf{A}(f) = \mathbf{K} + if\mathbf{C} - f^2\mathbf{M}$,

and we consider the preconditioners

- ◇ $\mathbf{P}^i(f_0) = \mathbf{K} + if_0\mathbf{C} + if_0^2\mathbf{M}$ (imaginary shift),
- ◇ $\mathbf{P}^r(f_0) = \mathbf{K} + if_0\mathbf{C} - f_0^2\mathbf{M}$ (real shift),
- ◇ $\mathbf{P}^m(f_0) = \mathrm{Re}(\mathbf{K}) - f_0^2\mathbf{M}$ (modified real shift).

We use LU decomposition of the preconditioner, such that $\mathbf{Q}_1\ \mathbf{P}^*(f_0)\ \mathbf{Q}_2 = \mathbf{L}\ \mathbf{U}$.

TU Delft

Solving the mathematical problem
Shifted Laplace Preconditioning

The system matrix equals $\mathbf{A}(f) = \mathbf{K} + if\mathbf{C} - f^2\mathbf{M}$,

and we consider the preconditioners

- $\diamond$ $\mathbf{P}^i(f_0) = \mathbf{K} + if_0\mathbf{C} + if_0^2\mathbf{M}$ (imaginary shift),
- $\diamond$ $\mathbf{P}^r(f_0) = \mathbf{K} + if_0\mathbf{C} - f_0^2\mathbf{M}$ (real shift),
- $\diamond$ $\mathbf{P}^m(f_0) = \mathrm{Re}(\mathbf{K}) - f_0^2\mathbf{M}$ (modified real shift).

We use LU decomposition of the preconditioner, such that $\mathbf{Q}_1\ \mathbf{P}^*(f_0)\ \mathbf{Q}_2 = \mathbf{L}\ \mathbf{U}$.

TU Delft

Solving the mathematical problem
Shifted Laplace Preconditioning

The system matrix equals $\mathbf{A}(f) = \mathbf{K} + if\mathbf{C} - f^2\mathbf{M}$,

and we consider the preconditioners

- ◇ $\mathbf{P}^i(f_0) = \mathbf{K} + if_0\mathbf{C} + if_0^2\mathbf{M}$ (imaginary shift),
- ◇ $\mathbf{P}^r(f_0) = \mathbf{K} + if_0\mathbf{C} - f_0^2\mathbf{M}$ (real shift),
- ◇ $\mathbf{P}^m(f_0) = \mathrm{Re}(\mathbf{K}) - f_0^2\mathbf{M}$ (modified real shift).

We use LU decomposition of the preconditioner, such that $\mathbf{Q}_1\ \mathbf{P}^*(f_0)\ \mathbf{Q}_2 = \mathbf{L}\ \mathbf{U}$.

Reducing the computation time

Using previous solutions

We solve the linear system $\mathbf{A}(f)\mathbf{x}^f = \mathbf{b}$ for $f = 1, 2, \ldots$ Hz.

For $f = \varphi$ Hz, the solutions to $f = 1, 2, \ldots, \varphi - 1$ Hz are available.

Idea: use (some of) these vectors $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^{\varphi-1}$ for

- ◇ improving the initial guess $\mathbf{x}_0^\varphi$ (by Lagrange extrapolation),
- ◇ the initial search space $\mathcal{U}_0$.

$\mathbf{u}_i \in \mathcal{U}_j$ corresponds to $\mathbf{g}_i \in \mathcal{G}_j$ through $\mathbf{g}_i = \mathbf{A}\mathbf{u}_i$.

Reducing the computation time

Using previous solutions

We solve the linear system $\mathbf{A}(f)\mathbf{x}^f = \mathbf{b}$ for $f = 1, 2, \ldots$ Hz.

For $f = \varphi$ Hz, the solutions to $f = 1, 2, \ldots, \varphi - 1$ Hz are available.

Idea: use (some of) these vectors $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^{\varphi-1}$ for

- ◇ improving the initial guess $\mathbf{x}_0^{\varphi}$ (by Lagrange extrapolation),
- ◇ the initial search space $\mathcal{U}_0$.

$\mathbf{u}_i \in \mathcal{U}_j$ corresponds to $\mathbf{g}_i \in \mathcal{G}_j$ through $\mathbf{g}_i = \mathbf{A}\mathbf{u}_i$.

TUDelft

Reducing the computation time

Using previous solutions

We solve the linear system $\mathbf{A}(f)\mathbf{x}^f = \mathbf{b}$ for $f = 1, 2, \ldots$ Hz.

For $f = \varphi$ Hz, the solutions to $f = 1, 2, \ldots, \varphi - 1$ Hz are available.

Idea: use (some of) these vectors $\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^{\varphi-1}$ for

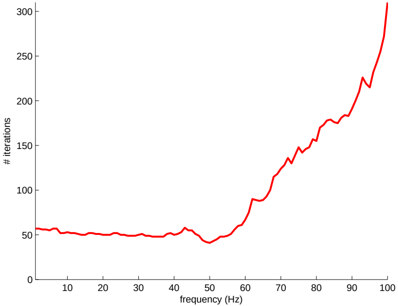
- ◇ improving the initial guess $\mathbf{x}_0^\varphi$ (by Lagrange extrapolation),
- ◇ the initial search space $\mathcal{U}_0$.

$\mathbf{u}_i \in \mathcal{U}_j$ corresponds to $\mathbf{g}_i \in \mathcal{G}_j$ through $\mathbf{g}_i = \mathbf{A}\mathbf{u}_i$.

TU Delft

Reducing the computation time

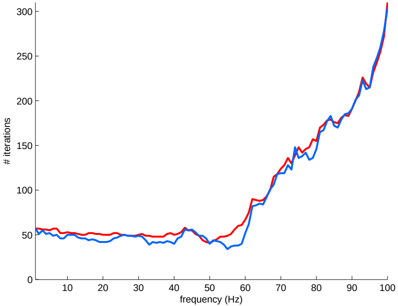
Using previous solutions



	none	$\mathbf{x}_0$	$\mathcal{U}_0$	$\mathbf{x}_0, \mathcal{U}_0$
# iterations	9667			
improvement	-			

Reducing the computation time

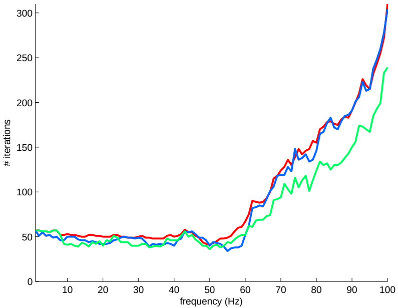
Using previous solutions



	none	$\mathbf{x}_0$	$\mathcal{U}_0$	$\mathbf{x}_0$, $\mathcal{U}_0$
# iterations	9667	8867		
improvement	-	8.3%		

Reducing the computation time

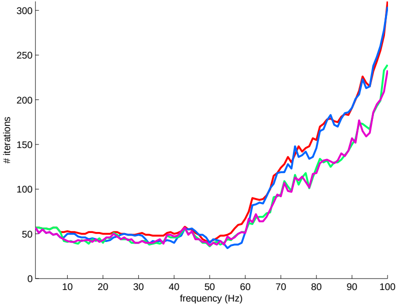
Using previous solutions



	none	$\mathbf{x}_0$	$\mathcal{U}_0$	$\mathbf{x}_0, \mathcal{U}_0$
# iterations	9667	8867	7806	
improvement	-	8.3%	19.3%	

Reducing the computation time

Using previous solutions



	none	$\mathbf{x}_0$	$\mathcal{U}_0$	$\mathbf{x}_0, \mathcal{U}_0$
# iterations	9667	8867	7806	7745
improvement	-	8.3%	19.3%	19.9%

TU Delft

Reducing the computation time

Using previous solutions

Some observations:

- For higher frequencies, the problem is more difficult to solve.
- Using previous solutions results in a significant reduction.
- Different $\mathbf{x}_0$ give equivalent results if we use $\mathcal{U}_0$.

Side notes:

- Other preconditioners lead to very similar results.
- Smaller s in IDR(s) results in more iterations for higher frequencies and in less reduction.
- For the car problem, extrapolation for $\mathbf{x}_0$ and using $\mathcal{U}_0$ give almost identical results.

TU Delft

Reducing the computation time

Using previous solutions

Some observations:

- ◇ For higher frequencies, the problem is more difficult to solve.
- ◇ Using previous solutions results in a significant reduction.
- ◇ Different $\mathbf{x}_0$ give equivalent results if we use $\mathcal{U}_0$.

Side notes:

- ◇ Other preconditioners lead to very similar results.
- ◇ Smaller s in IDR(s) results in more iterations for higher frequencies and in less reduction.
- ◇ For the car problem, extrapolation for $\mathbf{x}_0$ and using $\mathcal{U}_0$ give almost identical results.

Reducing the computation time

Using previous solutions

Some observations:

- ◇ For higher frequencies, the problem is more difficult to solve.
- ◇ Using previous solutions results in a significant reduction.
- ◇ Different $\mathbf{x}_0$ give equivalent results if we use $\mathcal{U}_0$.

Side notes:

- ◇ Other preconditioners lead to very similar results.
- ◇ Smaller s in IDR(s) results in more iterations for higher frequencies and in less reduction.
- ◇ For the car problem, extrapolation for $\mathbf{x}_0$ and using $\mathcal{U}_0$ give almost identical results.

TUDelft

Reducing the computation time

Using previous solutions

Some observations:

- ◇ For higher frequencies, the problem is more difficult to solve.
- ◇ Using previous solutions results in a significant reduction.
- ◇ Different $\mathbf{x}_0$ give equivalent results if we use $\mathcal{U}_0$.

Side notes:

- ◇ Other preconditioners lead to very similar results.
- ◇ Smaller s in IDR(s) results in more iterations for higher frequencies and in less reduction.
- ◇ For the car problem, extrapolation for $\mathbf{x}_0$ and using $\mathcal{U}_0$ give almost identical results.

TU Delft

Reducing the computation time

Using previous solutions

Some observations:

- ◇ For higher frequencies, the problem is more difficult to solve.
- ◇ Using previous solutions results in a significant reduction.
- ◇ Different $\mathbf{x}_0$ give equivalent results if we use $\mathcal{U}_0$.

Side notes:

- ◇ Other preconditioners lead to very similar results.
- ◇ Smaller s in IDR(s) results in more iterations for higher frequencies and in less reduction.
- ◇ For the car problem, extrapolation for $\mathbf{x}_0$ and using $\mathcal{U}_0$ give almost identical results.

TU Delft

Reducing the computation time

Using previous solutions

Some observations:

- ◇ For higher frequencies, the problem is more difficult to solve.
- ◇ Using previous solutions results in a significant reduction.
- ◇ Different $\mathbf{x}_0$ give equivalent results if we use $\mathcal{U}_0$.

Side notes:

- ◇ Other preconditioners lead to very similar results.
- ◇ Smaller s in IDR(s) results in more iterations for higher frequencies and in less reduction.
- ◇ For the car problem, extrapolation for $\mathbf{x}_0$ and using $\mathcal{U}_0$ give almost identical results.

TUDelft

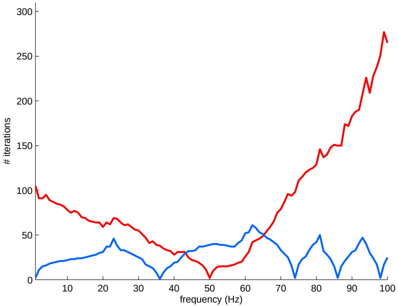
Reducing the computation time

Updating the preconditioner

	$\mathbf{P}^r(50)$	$\mathbf{P}^r(f_0)$	$\mathbf{P}^m(50)$	$\mathbf{P}^m(f_0)$
# iterations	8062			
time (s)	13845			
improvement	-			

Reducing the computation time

Updating the preconditioner

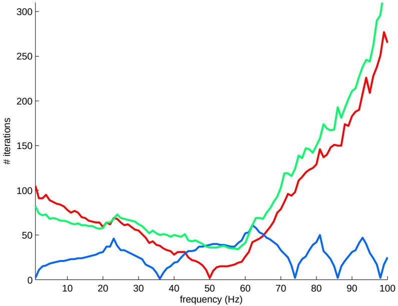


	$\mathbf{P}^r(50)$	$\mathbf{P}^r(f_0)$	$\mathbf{P}^m(50)$	$\mathbf{P}^m(f_0)$
# iterations	8062	2857		
time (s)	13845	5978		
improvement	-	56.8%		

TU Delft

Reducing the computation time

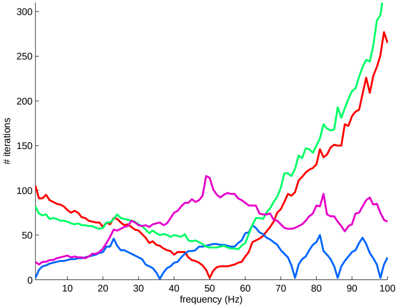
Updating the preconditioner



	$\mathbf{P}^r(50)$	$\mathbf{P}^r(f_0)$	$\mathbf{P}^m(50)$	$\mathbf{P}^m(f_0)$
# iterations	8062	2857	9946	
time (s)	13845	5978	6399	
improvement	-	56.8%	-	

Reducing the computation time

Updating the preconditioner



	$\mathbf{P}^r(50)$	$\mathbf{P}^r(f_0)$	$\mathbf{P}^m(50)$	$\mathbf{P}^m(f_0)$
# iterations	8062	2857	9946	5816
time (s)	13845	5978	6399	3686
improvement	-	56.8%	-	42.4%

Reducing the computation time

Updating the preconditioner

Some observations:

- ◇ Updating the preconditioner approximately halves the computation time.
- ◇ For higher frequencies we need to update more often.
- ◇ The modified shifted Laplace preconditioner is the best preconditioner.

TUDelft

Reducing the computation time

Updating the preconditioner

Some observations:

- ◇ Updating the preconditioner approximately halves the computation time.
- ◇ For higher frequencies we need to update more often.
- ◇ The modified shifted Laplace preconditioner is the best preconditioner.

TU Delft

Reducing the computation time
Updating the preconditioner

Some observations:

- ◇ Updating the preconditioner approximately halves the computation time.
- ◇ For higher frequencies we need to update more often.
- ◇ The modified shifted Laplace preconditioner is the best preconditioner.

Reducing the computation time

Using spectral information

In IDR(s), the residuals $\mathbf{r}_i \in \mathcal{G}_j$ and hence $\exists\ \hat{\mathbf{r}}_i \in \mathcal{G}_0$ such that

$$\mathbf{r}_i = \prod_{\ell=1}^{j} (\mathbf{I} - \omega_\ell \mathbf{A}) \hat{\mathbf{r}}_i.$$

Rewriting the residual updates results in the relation

$$\mathbf{A}\hat{\mathbf{r}}_{i-1} = \sum_{\ell=i-s-1}^{i} h_\ell \hat{\mathbf{r}}_\ell,$$

or in matrix form

$$\mathbf{A}\hat{\mathbf{R}}_i = \hat{\mathbf{R}}_i \mathbf{H}_i + h_{i+1,1} \hat{\mathbf{r}}_{i+1} \mathbf{e}_i^\top.$$

Reducing the computation time

Using spectral information

In IDR(s), the residuals $\mathbf{r}_i \in \mathcal{G}_j$ and hence $\exists\ \hat{\mathbf{r}}_i \in \mathcal{G}_0$ such that

$$\mathbf{r}_i = \prod_{\ell=1}^{j} (\mathbf{I} - \omega_\ell \mathbf{A}) \hat{\mathbf{r}}_i .$$

Rewriting the residual updates results in the relation

$$\mathbf{A}\hat{\mathbf{r}}_{i-1} = \sum_{\ell=i-s-1}^{i} h_\ell \hat{\mathbf{r}}_\ell ,$$

or in matrix form

$$\mathbf{A}\hat{\mathbf{R}}_i = \hat{\mathbf{R}}_i \mathbf{H}_i + h_{i+1,1} \hat{\mathbf{r}}_{i+1} \mathbf{e}_i^\top .$$

TU Delft

Reducing the computation time
Using spectral information

The residuals of a Krylov subspace method satisfy $\mathbf{r}_i = R_i(\mathbf{A})\mathbf{r}_0$.

$\|\mathbf{r}_i\| \leq \|R_i(\mathbf{A})\|\|\mathbf{r}_0\|$: $R_i(\xi)$ small on the spectrum of $\mathbf{A}$.

- ⋄ IDR(s): $R_i(\xi) = \prod_{\ell=1}^{j}(1-\omega_\ell\xi)\Psi_{i-j}(\xi) = \Omega_j(\xi)\Psi_{i-j}(\xi)$.
- ⋄ Smallest polynomial on area enclosed by an ellipse: $T_j(\xi)$.

Choose an ellipse that encloses the Ritz values and choose ω_ℓ such that the roots of $\Omega_j(\xi)$ and $T_j(\xi)$ coincide.

Reducing the computation time

Using spectral information

The residuals of a Krylov subspace method satisfy $\mathbf{r}_i = R_i(\mathbf{A})\mathbf{r}_0$.

$\|\mathbf{r}_i\| \leq \|R_i(\mathbf{A})\|\|\mathbf{r}_0\|$: $R_i(\xi)$ small on the spectrum of $\mathbf{A}$.

- ◇ IDR(s): $R_i(\xi) = \prod_{\ell=1}^{j}(1-\omega_\ell\xi)\Psi_{i-j}(\xi) = \Omega_j(\xi)\Psi_{i-j}(\xi)$.
- ◇ Smallest polynomial on area enclosed by an ellipse: $T_j(\xi)$.

Choose an ellipse that encloses the Ritz values and choose ω_ℓ such that the roots of $\Omega_j(\xi)$ and $T_j(\xi)$ coincide.

TUDelft

Reducing the computation time

Using spectral information

The residuals of a Krylov subspace method satisfy $\mathbf{r}_i = R_i(\mathbf{A})\mathbf{r}_0$.

$\|\mathbf{r}_i\| \leq \|R_i(\mathbf{A})\|\|\mathbf{r}_0\|$: $R_i(\xi)$ small on the spectrum of $\mathbf{A}$.

- ◇ IDR(s): $R_i(\xi) = \prod_{\ell=1}^{j}(1 - \omega_\ell \xi)\Psi_{i-j}(\xi) = \Omega_j(\xi)\Psi_{i-j}(\xi)$.
- ◇ Smallest polynomial on area enclosed by an ellipse: $T_j(\xi)$.

Choose an ellipse that encloses the Ritz values and choose ω_ℓ such that the roots of $\Omega_j(\xi)$ and $T_j(\xi)$ coincide.

TUDelft

Reducing the computation time

Using spectral information

The residuals of a Krylov subspace method satisfy $\mathbf{r}_i = R_i(\mathbf{A})\mathbf{r}_0$.

$\|\mathbf{r}_i\| \leq \|R_i(\mathbf{A})\|\|\mathbf{r}_0\|$: $R_i(\xi)$ small on the spectrum of $\mathbf{A}$.

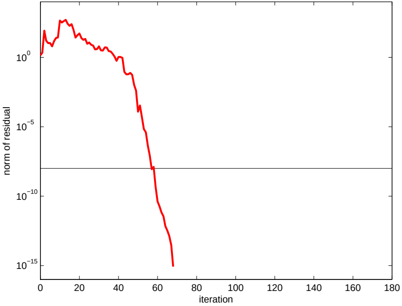
- ◇ IDR(s): $R_i(\xi) = \prod_{\ell=1}^{j}(1 - \omega_\ell \xi)\Psi_{i-j}(\xi) = \Omega_j(\xi)\Psi_{i-j}(\xi)$.
- ◇ Smallest polynomial on area enclosed by an ellipse: $T_j(\xi)$.

Choose an ellipse that encloses the Ritz values and choose ω_ℓ such that the roots of $\Omega_j(\xi)$ and $T_j(\xi)$ coincide.

TUDelft

Reducing the computation time

Using spectral information

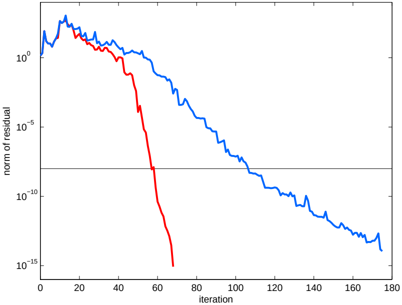


Choice for ω_j:

- ◇ minimise $\|\mathbf{r}_i\|$ w.r.t. ω_j

Reducing the computation time

Using spectral information



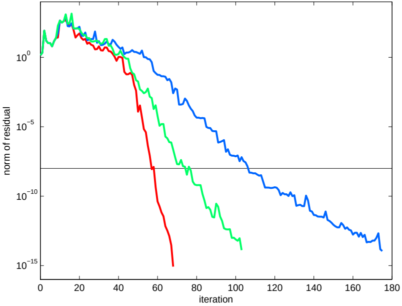
Choice for ω_j:

- ◇ minimise $\|\mathbf{r}_i\|$ w.r.t. ω_j
- ◇ Chebyshev nodes for $f = 19$ Hz

TUDelft

Reducing the computation time

Using spectral information



Choice for ω_j:

- ◇ minimise $\|\mathbf{r}_i\|$ w.r.t. ω_j
- ◇ Chebyshev nodes for $f = 19$ Hz
- ◇ as blue, but $\omega_j = (\mathbf{v}_i \cdot \mathbf{v}_i)/(\mathbf{v}_i \cdot \mathbf{A}\mathbf{v}_i) \in$ FOV for $f = 19$ Hz

TUDelft

Reducing the computation time
Using spectral information

Some observations:

- ◇ There is an increase in the number of iterations with the new approach.
- ◇ The choice of ω_j impacts the Ritz values.

TU Delft

Reducing the computation time

Using spectral information

Some observations:

- ◇ There is an increase in the number of iterations with the new approach.
- ◇ The choice of ω_j impacts the Ritz values.

Conclusions

◇ Using the solution vectors for $\mathbf{x}_0$ or $\mathcal{U}_0$ leads to a significant reduction in iterations.

◇ Updating the right preconditioners leads to serious reduction.

◇ Using spectral information does not lead to better results.

The best results are obtained with

◇ a preconditioner $\mathbf{P}^m(f_0)$ with updating strategy for f_0,

◇ IDR(s) and $s = 8$,

◇ extrapolation of solution vectors for $\mathbf{x}_0$.

TU Delft

Conclusions

- ◇ Using the solution vectors for $\mathbf{x}_0$ or $\mathcal{U}_0$ leads to a significant reduction in iterations.
- ◇ Updating the right preconditioners leads to serious reduction.
- ◇ Using spectral information does not lead to better results.

The best results are obtained with

- ◇ a preconditioner $\mathbf{P}^m(f_0)$ with updating strategy for f_0,
- ◇ IDR(s) and $s = 8$,
- ◇ extrapolation of solution vectors for $\mathbf{x}_0$.

Conclusions

- ⋄ Using the solution vectors for $\mathbf{x}_0$ or $\mathcal{U}_0$ leads to a significant reduction in iterations.
- ⋄ Updating the right preconditioners leads to serious reduction.
- ⋄ Using spectral information does not lead to better results.

The best results are obtained with

- ⋄ a preconditioner $\mathbf{P}^m(f_0)$ with updating strategy for f_0,
- ⋄ IDR(s) and $s = 8$,
- ⋄ extrapolation of solution vectors for $\mathbf{x}_0$.

TUDelft

Conclusions

- ◇ Using the solution vectors for $\mathbf{x}_0$ or $\mathcal{U}_0$ leads to a significant reduction in iterations.

- ◇ Updating the right preconditioners leads to serious reduction.

- ◇ Using spectral information does not lead to better results.

The best results are obtained with

- ◇ a preconditioner $\mathbf{P}^m(f_0)$ with updating strategy for f_0,
- ◇ IDR(s) and $s = 8$,
- ◇ extrapolation of solution vectors for $\mathbf{x}_0$.

Future research

◇ Analyses on Ritz values with IDR(s): dependency of Ritz values on ω_j, convergence of Ritz values.

◇ Consider other residual polynomials: base choices for ω_j on Leja points or Ritz values itself.

◇ Use Fortran and parallisation.

TU Delft

Future research

- ◇ Analyses on Ritz values with IDR(s): dependency of Ritz values on ω_j, convergence of Ritz values.
- ◇ Consider other residual polynomials: base choices for ω_j on Leja points or Ritz values itself.
- ◇ Use Fortran and parallisation.

Future research

- ◇ Analyses on Ritz values with IDR(s): dependency of Ritz values on ω_j, convergence of Ritz values.
- ◇ Consider other residual polynomials: base choices for ω_j on Leja points or Ritz values itself.
- ◇ Use Fortran and parallisation.

Questions