Efficient Preconditioners with Multi-level Sequentially Semiseparable Structure

Y. Qiu[†*], M.B. van Gijzen[‡], J.W. van Wingerden[†], and M. Verhaegen[†]

† Delft Center for System and Control
‡Delft Institute of Applied Mathematics
Delft University of Technology
Delft, the Netherlands
*Email: Y.Qiu@tudelft.nl

26th September, 2013

TUDelft

Outline

Problem to Solve

- We consider solving the linear system
$$Ax = b$$
with preconditioned Krylov solver, where A is large and sparse.
- When A comes from discretized partial differential equation (PDE), the matrix A has an multi-level sequentially semiseparable (MSSS) structure.
- By exploiting the structure of the matrix, we can construct a class of efficient preconditioners based on MSSS matrix computations with linear complexity ($\mathcal{O}(n)$).

TUDelft

Problem to Solve

- We consider solving the linear system
$$Ax = b$$
with preconditioned Krylov solver, where A is large and sparse.
- When A comes from discretized partial differential equation (PDE), the matrix A has an multi-level sequentially semiseparable (MSSS) structure.
- By exploiting the structure of the matrix, we can construct a class of efficient preconditioners based on MSSS matrix computations with linear complexity ($\mathcal{O}(n)$).

TUDelft

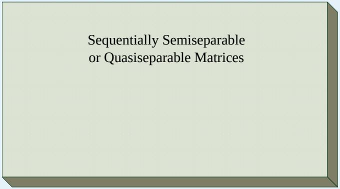
Problem to Solve

- We consider solving the linear system
$$Ax = b$$
with preconditioned Krylov solver, where A is large and sparse.
- When A comes from discretized partial differential equation (PDE), the matrix A has an multi-level sequentially semiseparable (MSSS) structure.
- By exploiting the structure of the matrix, we can construct a class of efficient preconditioners based on MSSS matrix computations with linear complexity ($\mathcal{O}(n)$).

TUDelft

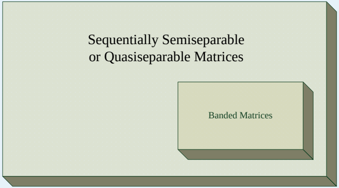
Multi-level Sequentially Semiseparable Matrices

- **Sequentially Semiseparable (SSS) Matrices**
  Off-diagonal blocks from the strictly lower-triangular part and upper-triangular part are of low rank.
- **Semiseparable Matrices**
  All sub-blocks taken from the strictly lower-triangular part and upper-triangular part are of rank 1.



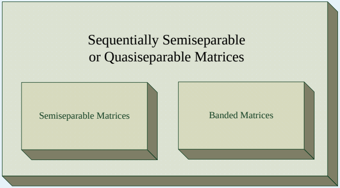
Multi-level Sequentially Semiseparable Matrices

- Sequentially Semiseparable (SSS) Matrices
  Off-diagonal blocks from the strictly lower-triangular part and upper-triangular part are of low rank.
- Semiseparable Matrices
  All sub-blocks taken from the strictly lower-triangular part and upper-triangular part are of rank 1.



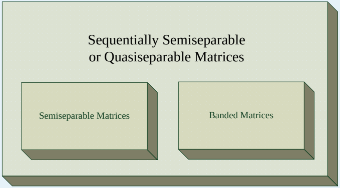
Multi-level Sequentially Semiseparable Matrices

- **Sequentially Semiseparable (SSS) Matrices**
  Off-diagonal blocks from the strictly lower-triangular part and upper-triangular part are of low rank.
- **Semiseparable Matrices**
  All sub-blocks taken from the strictly lower-triangular part and upper-triangular part are of rank 1.



Multi-level Sequentially Semiseparable Matrices

- Sequentially Semiseparable (SSS) Matrices
  Off-diagonal blocks from the strictly lower-triangular part and upper-triangular part are of low rank.
- Semiseparable Matrices
  All sub-blocks taken from the strictly lower-triangular part and upper-triangular part are of rank 1.



Multi-level Sequentially Semiseparable Matrices

Generators Representation of MSSS Matrices

Let A be an $N \times N$ block matrix with SSS structure, then A can be written in the following block partitioned form

$$A_{i,j} = \begin{cases} U_i W_{i+1} \cdots W_{j-1} V_j^T, & \text{if } i < j; \\ D_i, & \text{if } i = j; \\ P_i R_{i-1} \cdots R_{j+1} Q_j^T, & \text{if } i > j. \end{cases}$$

and denoted as $A = \mathcal{SSS}(P_S,\ R_S,\ Q_s,\ D_s,\ U_s,\ W_s,\ V_s)$.

Example

Take $N = 4$ for example, we have

$$A = \begin{bmatrix} D_1 & U_1 V_2^T & U_1 W_2 V_3^T & U_1 W_2 W_3 V_4^T \\ P_2 Q_1^T & D_2 & U_2 V_3^T & U_2 W_3 V_4^T \\ P_3 R_2 Q_1^T & P_3 Q_2^T & D_3 & U_3 V_4^T \\ P_4 R_3 R_2 Q_1^T & P_4 R_3 Q_2^T & P_4 Q_3^T & D_4 \end{bmatrix}$$

Multi-level Sequentially Semiseparable Matrices

Generators Representation of MSSS Matrices (cont'd)

Reference

- S. Chandrasekaran, P. Dewilde et al.*SIAM. J. Matrix Anal. & Appl*, 2005, 27(2), 341–364.

Definition

The matrix is said to be a k-level multi-level sequentially semiseparable (MSSS) matrix, if its generators are $(k-1)$-level MSSS matrices. The 1-level MSSS matrix is the SSS matrix that satisfies the generator representation aforementioned.

Multi-level Sequentially Semiseparable Matrices

Generators Representation of MSSS Matrices (cont'd)

Example

Take the 2D Laplacian matrix A with homogeneous Dirichlet boundary condition for example,

$$A = \begin{bmatrix} D & F & & & \\ E & D & F & & \\ & E & \ddots & \ddots & \\ & & \ddots & D & F \\ & & & E & D \end{bmatrix}, \text{ where } D = \begin{bmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & -1 & \ddots & \ddots & \\ & & \ddots & 4 & -1 \\ & & & -1 & 4 \end{bmatrix}$$

and $E = F = -I_n$, I_n is the $n \times n$ identity matrix. The matrix A has the multi-level sequentially semiseparable structure and can be denoted as

$$A = \mathcal{MSSS}(I_n,\ 0,\ E,\ D,\ I_n,\ 0,\ F)$$

where $D = \mathcal{SSS}(1,\ 0,\ -1,\ 4,\ 1,\ 0,\ -1)$, $E = F = \mathcal{SSS}(0,\ 0,\ 0,\ -1,\ 0,\ 0,\ 0)$.

Multi-level Sequentially Semiseparable Matrices

Remarks on SSS Matrices

Remarks

1. Generators of MSSS matrices can be variant, even of different sizes, as long as satisfying the block-partitioned form.
2. Almost all the operations (addition, multiplication, inversion, et al) of SSS matrices can be performed in linear computational complexity and preserve the SSS matrix structure.
3. Addition and multiplication will lead to growth of the rank of the generators, which increases the computational complexity.
4. Model order reduction (MOR) is necessary to keep the computational complexity low.

Remarks

1. Generators of MSSS matrices can be variant, even of different sizes, as long as satisfying the block-partitioned form.
2. Almost all the operations (addition, multiplication, inversion, et al) of SSS matrices can be performed in linear computational complexity and preserve the SSS matrix structure.
3. Addition and multiplication will lead to growth of the rank of the generators, which increases the computational complexity.
4. Model order reduction (MOR) is necessary to keep the computational complexity low.

Multi-level Sequentially Semiseparable Matrices

Remarks on SSS Matrices

Remarks

1. Generators of MSSS matrices can be variant, even of different sizes, as long as satisfying the block-partitioned form.
2. Almost all the operations (addition, multiplication, inversion, et al) of SSS matrices can be performed in linear computational complexity and preserve the SSS matrix structure.
3. Addition and multiplication will lead to growth of the rank of the generators, which increases the computational complexity.
4. Model order reduction (MOR) is necessary to keep the computational complexity low.

Multi-level Sequentially Semiseparable Matrices

Remarks on SSS Matrices

Remarks

1. Generators of MSSS matrices can be variant, even of different sizes, as long as satisfying the block-partitioned form.
2. Almost all the operations (addition, multiplication, inversion, et al) of SSS matrices can be performed in linear computational complexity and preserve the SSS matrix structure.
3. Addition and multiplication will lead to growth of the rank of the generators, which increases the computational complexity.
4. Model order reduction (MOR) is necessary to keep the computational complexity low.

Multi-level Sequentially Semiseparable Matrices

Remarks on SSS Matrices

Remarks

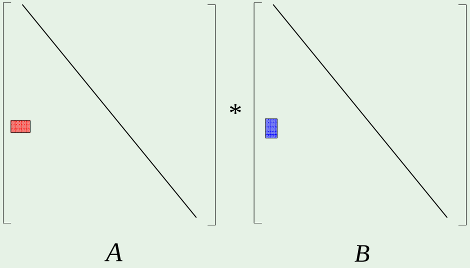
1. Generators of MSSS matrices can be variant, even of different sizes, as long as satisfying the block-partitioned form.
2. Almost all the operations (addition, multiplication, inversion, et al) of SSS matrices can be performed in linear computational complexity and preserve the SSS matrix structure.
3. Addition and multiplication will lead to growth of the rank of the generators, which increases the computational complexity.
4. Model order reduction (MOR) is necessary to keep the computational complexity low.

Multi-level Sequentially Semiseparable Matrices

Growth of the Rank of the Generators

Example

Take the multiplication of 2 SSS matrices A and B for example,

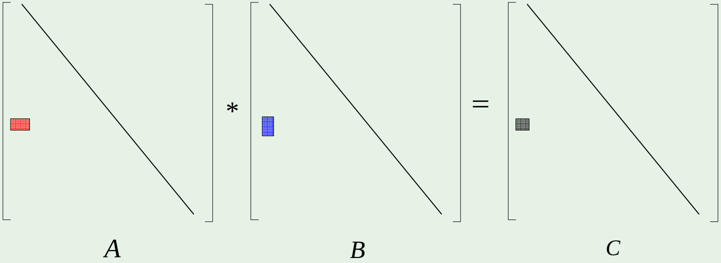


Multi-level Sequentially Semiseparable Matrices

Growth of the Rank of the Generators

Example

Take the multiplication of 2 SSS matrices A and B for example,

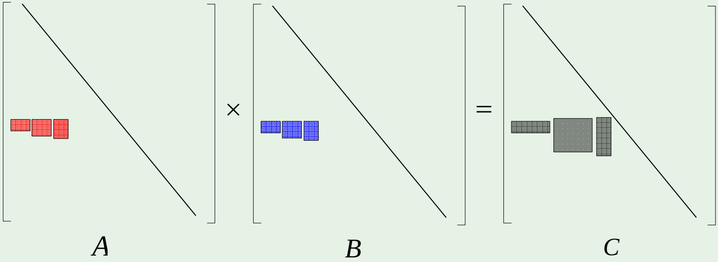


Multi-level Sequentially Semiseparable Matrices

Growth of the Rank of the Generators

Example

Take the multiplication of 2 SSS matrices A and B for example,



Multi-level Sequentially Semiseparable Preconditioners

- To solve the system, ideally we make an LU factorization of the system matrix with MSSS matrix computation technique.
- All the computations are performed on the generators of the system matrix.
- This leads to growth of the rank of the generators.
- To keep the computational complexity low, model reduction is needed.
- This yields an approximate factorization of the system matrix, which can be used as a preconditioner.

TUDelft

Multi-level Sequentially Semiseparable Preconditioners

- To solve the system, ideally we make an LU factorization of the system matrix with MSSS matrix computation technique.
- All the computations are performed on the generators of the system matrix.
- This leads to growth of the rank of the generators.
- To keep the computational complexity low, model reduction is needed.
- This yields an approximate factorization of the system matrix, which can be used as a preconditioner.

Multi-level Sequentially Semiseparable Preconditioners

- To solve the system, ideally we make an LU factorization of the system matrix with MSSS matrix computation technique.
- All the computations are performed on the generators of the system matrix.
- This leads to growth of the rank of the generators.
- To keep the computational complexity low, model reduction is needed.
- This yields an approximate factorization of the system matrix, which can be used as a preconditioner.

Multi-level Sequentially Semiseparable Preconditioners

- To solve the system, ideally we make an LU factorization of the system matrix with MSSS matrix computation technique.
- All the computations are performed on the generators of the system matrix.
- This leads to growth of the rank of the generators.
- To keep the computational complexity low, model reduction is needed.
- This yields an approximate factorization of the system matrix, which can be used as a preconditioner.

TUDelft

Multi-level Sequentially Semiseparable Preconditioners

- To solve the system, ideally we make an LU factorization of the system matrix with MSSS matrix computation technique.
- All the computations are performed on the generators of the system matrix.
- This leads to growth of the rank of the generators.
- To keep the computational complexity low, model reduction is needed.
- This yields an approximate factorization of the system matrix, which can be used as a preconditioner.

TUDelft

Multi-level Sequentially Semiseparable Preconditioners

- To solve the system, ideally we make an LU factorization of the system matrix with MSSS matrix computation technique.
- All the computations are performed on the generators of the system matrix.
- This leads to growth of the rank of the generators.
- To keep the computational complexity low, model reduction is needed.
- This yields an approximate factorization of the system matrix, which can be used as a preconditioner.

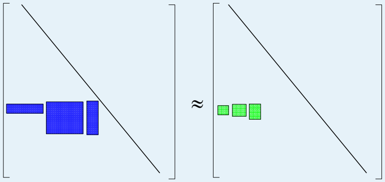
Purpose

Find SSS matrices with generators of smaller sizes that are equivalent to the original SSS matrix to a certain tolerance.

Ways Out

A suitable model order reduction algorithm can be derived from the correspondence between SSS matrices and linear time-varying (LTV) systems.

Illustration



Multi-level Sequentially Semiseparable Preconditioners

Linear Time-Varying System Representation

Mixed-causal linear time-varying system

$$\begin{bmatrix} x^c_{i+1} \\ x^a_{i-1} \end{bmatrix} = \begin{bmatrix} R_i & \\ & W_i \end{bmatrix} \begin{bmatrix} x^c_i \\ x^a_i \end{bmatrix} + \begin{bmatrix} C^f_i \\ C^b_i \end{bmatrix} u_i$$

$$y_i = \begin{bmatrix} B^f_i & B^b_i \end{bmatrix} \begin{bmatrix} x^c_i \\ x^a_i \end{bmatrix} + A_i u_i$$

with zero initial conditions.

Input-Output Relation

Stack all the input and output variables

$$\bar{u} = \begin{bmatrix} u_1^T & u_2^T & \cdots & u_N^T \end{bmatrix}^T$$

$$\bar{y} = \begin{bmatrix} y_1^T & y_2^T & \cdots & y_N^T \end{bmatrix}^T$$

we have the system input-output described as

$$\bar{y} = \bar{A}\bar{u}$$

Example

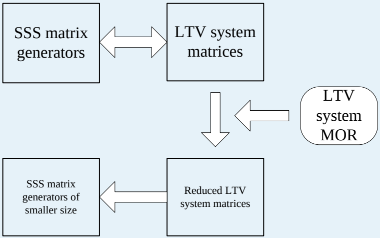
Take $N = 4$ for example, we have $\bar{A}$ described as

$$\begin{bmatrix} A_1 & B^b_1 C^b_2 & B^b_1 W_2 C^b_3 & B^b_1 W_2 W_3 C^b_4 \\ B^f_2 C^f_1 & A_2 & B^b_2 C^b_3 & B_2 W_3 C^b_4 \\ B^f_3 R_2 C^f_1 & B^f_3 C^f_2 & A_3 & B_3 C^b_4 \\ B^f_4 R_3 R_2 C^f_1 & B^f_4 R_3 C^f_2 & B^f_4 C^f_3 & A_4 \end{bmatrix}$$

Multi-level Sequentially Semiseparable Preconditioners

A Novel Model Reduction Algorithm

MOR Illustration



Reference

- Y. Qiu, M.B. van Gijzen, et al. *Tech. Rept. 13-04. Delft Institute of Applied Mathematics, Delft University of Technology, 2013.*

TU Delft

Multi-level Sequentially Semiseparable Preconditioners

A Novel Model Reduction Algorithm (cont'd)

MOR of LTV System

Consider the causal LTV system

$$\begin{cases} x_{k+1} = A_k x_k + B_k u_k \\ y_k = C_k x_k \end{cases}$$

over time interval $[k_0, \ k_f]$ with zero initial states.

Balanced truncation: eliminates the system states that consume more input energy to reach but contributes little to the output energy of the system.

Controllability Gramian $\mathcal{G}_c(k)$

Energy consumed to reach certain states at time step k

$$\begin{aligned} \mathcal{G}_c(k+1) &= A_k \mathcal{G}_c(k) A_k^T + B_k B_k^T \\ \mathcal{G}_c(k_0) &= 0 \end{aligned}$$

Observability Gramian $\mathcal{G}_o(k)$

Energy of the system states contributes to output at time step k

$$\begin{aligned} \mathcal{G}_o(k) &= A_k^T \mathcal{G}_o(k+1) A_k + C_k^T C_k \\ \mathcal{G}_o(k_f) &= 0 \end{aligned}$$

TUDelft

Multi-level Sequentially Semiseparable Preconditioners

A Novel Model Reduction Algorithm (cont'd)

Low-rank Approximation

Both $\mathcal{G}_c(k)$ and $\mathcal{G}_o(k)$ are positive (semi-)definite and are often of low numerical rank. Thus, low-rank approximation could be performed to approximate $\mathcal{G}_c(k)$ and $\mathcal{G}_o(k)$.

$$\mathcal{G}_c(k) \approx \tilde{L_k^c}\tilde{L_k^c}^T, \quad \mathcal{G}_o(k) \approx \tilde{L_k^o}\tilde{L_k^o}^T \; ,$$

where $\tilde{L_k^c} \in \mathbb{R}^{N\times n_k^c}$, $\tilde{L_k^o} \in \mathbb{R}^{N\times n_k^o}$ and $n_k^o, \; n_k^c \ll N$.

Reduced LTV System

With the approximated $\mathcal{G}_c(k)$ and $\mathcal{G}_o(k)$, the reduced LTV system

$$\begin{cases} \hat{x}_{k+1} = \Pi_l(k+1)A_k\Pi_r(k)\hat{x}_k + \Pi_l(k+1)B_k u_k \\ \quad y_k = C_k\Pi_r(k)\hat{x}_k. \end{cases}$$

where $\Pi_l(k) \in \mathbb{R}^{n\times N}$, $\Pi_r(k) \in \mathbb{R}^{N\times n}$, N and n are the system order before and after model reduction.

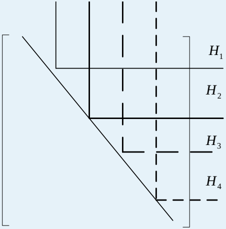
TU Delft

Multi-level Sequentially Semiseparable Preconditioners

Convectional Model Reduction Algorithm

Hankel Blocks Approximation

Approximate the Hankel blocks of the SSS matrix sequentially.



Reference

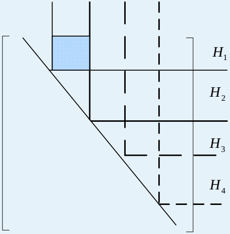
- S. Chandrasekaran, P. Dewilde et al.*SIAM. J. Matrix Anal. & Appl*, 2005, 27(2), 341–364.

Multi-level Sequentially Semiseparable Preconditioners

Convectional Model Reduction Algorithm

Hankel Blocks Approximation

Approximate the Hankel blocks of the SSS matrix sequentially.



Reference

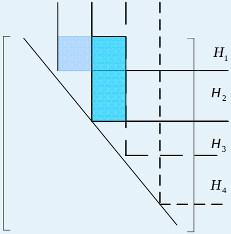
- S. Chandrasekaran, P. Dewilde et al.*SIAM. J. Matrix Anal. & Appl*, 2005, 27(2), 341–364.

Multi-level Sequentially Semiseparable Preconditioners

Convectional Model Reduction Algorithm

Hankel Blocks Approximation

Approximate the Hankel blocks of the SSS matrix sequentially.



Reference

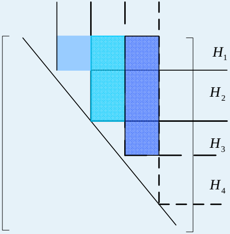
- S. Chandrasekaran, P. Dewilde et al.*SIAM. J. Matrix Anal. & Appl*, 2005, 27(2), 341–364.

Multi-level Sequentially Semiseparable Preconditioners

Convectional Model Reduction Algorithm

Hankel Blocks Approximation

Approximate the Hankel blocks of the SSS matrix sequentially.



Reference

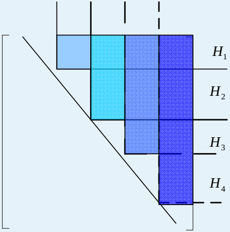
- S. Chandrasekaran, P. Dewilde et al.*SIAM. J. Matrix Anal. & Appl*, 2005, 27(2), 341–364.

Multi-level Sequentially Semiseparable Preconditioners

Convectional Model Reduction Algorithm

Hankel Blocks Approximation

Approximate the Hankel blocks of the SSS matrix sequentially.



Reference

- S. Chandrasekaran, P. Dewilde et al.*SIAM. J. Matrix Anal. & Appl*, 2005, 27(2), 341–364.

Convection-Diffusion Equation

$$
\begin{aligned}
-\epsilon\nabla^2 u + \overrightarrow{\omega}.\nabla u &= f \text{ in } \Omega \\
u_D &= g_D \text{ on } \Gamma_D \\
\frac{\partial u}{\partial n} &= g_N \text{ on } \Gamma_N
\end{aligned}
$$

where

$$
g_D = \begin{cases} (2x-1)^2(2y-1)^2 & \text{if } \mathbf{x} \in [0,\ \frac{1}{2}]^2 \\ 0 & \text{otherwise} \end{cases}, \quad g_N = (2x-1)(2y-1),
$$

$\Omega = [0,\ 1]^2$, $\Gamma_N = \{1\} \times (0,\ 1)$, $\Gamma_D = \Gamma \backslash \Gamma_N$, ϵ is a positive scalar, $\overrightarrow{\omega}$ is the unit directional vector, $\overrightarrow{\omega} = (cos(\theta),\ sin(\theta))^T$ and n is the normal vector on the bounds pointing outward.

Experiment Setup

Perform the experiment with Matlab 2010b on a laptop with Intel Core 2 Duo P8700 2.57GHz and 4Gb memory.

Numerical Experiments

Preconditioning of Convection-Diffusion Equation (cont'd)

Computational Results

Set $\epsilon = 10^{-2}$, $\theta = \frac{\pi}{5}$ and the stopping tolerance $tol = 10^{-6}$.

Table: Pre. time by new MOR and convectional MOR and IDR time

mesh size h	iter. NO.	pre. time (s)	IDR(4) time (s)	total (s)
$2^{-5}(3)$	4	0.43	0.27	0.80
$2^{-6}(3)$	4	0.84	0.37	1.21
$2^{-7}(4)$	4	3.44	0.83	4.27
$2^{-8}(4)$	4	12.58	2.20	14.78
$2^{-9}(4)$	6	48.14	10.20	58.34
mesh size h	iter. NO.	pre. time (s)	IDR(4) time (s)	total (s)
$2^{-5}(3)$	2	0.48	0.08	0.56
$2^{-6}(3)$	2	1.37	0.17	1.54
$2^{-7}(4)$	2	5.47	0.39	5.86
$2^{-8}(4)$	3	21.11	1.54	22.65
$2^{-9}(4)$	4	85.09	6.77	91.86

Computational Results

Set $\epsilon = 10^{-2}$, $\theta = \frac{\pi}{5}$ and the stopping tolerance $tol = 10^{-6}$.

Table: Pre. time by new MOR and convectional MOR and IDR time

mesh size h	iter. NO.	pre. time (s)	IDR(4) time (s)	total (s)
$2^{-5}(3)$	4	0.43	0.27	0.80
$2^{-6}(3)$	4	0.84	0.37	1.21
$2^{-7}(4)$	4	3.44	0.83	4.27
$2^{-8}(4)$	4	12.58	2.20	14.78
$2^{-9}(4)$	6	48.14	10.20	58.34
mesh size h	iter. NO.	pre. time (s)	IDR(4) time (s)	total (s)
$2^{-5}(3)$	2	0.48	0.08	0.56
$2^{-6}(3)$	2	1.37	0.17	1.54
$2^{-7}(4)$	2	5.47	0.39	5.86
$2^{-8}(4)$	3	21.11	1.54	22.65
$2^{-9}(4)$	4	85.09	6.77	91.86

Numerical Experiments

Preconditioning of Convection-Diffusion Equation (cont'd)

Computational Results

Set $\epsilon = 10^{-2}$, $\theta = \frac{\pi}{5}$ and the stopping tolerance $tol = 10^{-6}$.

Table: Pre. time by new MOR and convectional MOR and IDR time

mesh size h	iter. NO.	pre. time (s)	IDR(4) time (s)	total (s)
$2^{-5}(3)$	4	0.43	0.27	0.80
$2^{-6}(3)$	4	0.84	0.37	1.21
$2^{-7}(4)$	4	3.44	0.83	4.27
$2^{-8}(4)$	4	12.58	2.20	14.78
$2^{-9}(4)$	6	48.14	10.20	58.34
mesh size h	iter. NO.	pre. time (s)	IDR(4) time (s)	total (s)
$2^{-5}(3)$	2	0.48	0.08	0.56
$2^{-6}(3)$	2	1.37	0.17	1.54
$2^{-7}(4)$	2	5.47	0.39	5.86
$2^{-8}(4)$	3	21.11	1.54	22.65
$2^{-9}(4)$	4	85.09	6.77	91.86

Numerical Experiments

Preconditioning of Convection-Diffusion Equation (cont'd)

Computational Results

Set $\epsilon = 10^{-2}$, $\theta = \frac{\pi}{5}$ and the stopping tolerance $tol = 10^{-6}$.

Table: Pre. time by new MOR and convectional MOR and IDR time

mesh size h	iter. NO.	pre. time (s)	IDR(4) time (s)	total (s)
$2^{-5}(3)$	4	0.43	0.27	0.80
$2^{-6}(3)$	4	0.84	0.37	1.21
$2^{-7}(4)$	4	3.44	0.83	4.27
$2^{-8}(4)$	4	12.58	2.20	14.78
$2^{-9}(4)$	6	48.14	10.20	58.34
mesh size h	iter. NO.	pre. time (s)	IDR(4) time (s)	total (s)
$2^{-5}(3)$	2	0.48	0.08	0.56
$2^{-6}(3)$	2	1.37	0.17	1.54
$2^{-7}(4)$	2	5.47	0.39	5.86
$2^{-8}(4)$	3	21.11	1.54	22.65
$2^{-9}(4)$	4	85.09	6.77	91.86

TUDelft

Computational Results

Set $\epsilon = 10^{-2}$, $\theta = \frac{\pi}{5}$ and the stopping tolerance $tol = 10^{-6}$.

Table: Pre. time by new MOR and convectional MOR and IDR time

mesh size h	iter. NO.	pre. time (s)	IDR(4) time (s)	total (s)
$2^{-5}(3)$	4	0.43	0.27	0.80
$2^{-6}(3)$	4	0.84	0.37	1.21
$2^{-7}(4)$	4	3.44	0.83	4.27
$2^{-8}(4)$	4	12.58	2.20	14.78
$2^{-9}(4)$	6	48.14	10.20	58.34
mesh size h	iter. NO.	pre. time (s)	IDR(4) time (s)	total (s)
$2^{-5}(3)$	2	0.48	0.08	0.56
$2^{-6}(3)$	2	1.37	0.17	1.54
$2^{-7}(4)$	2	5.47	0.39	5.86
$2^{-8}(4)$	3	21.11	1.54	22.65
$2^{-9}(4)$	4	85.09	6.77	91.86

Numerical Experiments

Preconditioning of Optimal Control of PDEs

IDR(s) Method

To solve the nonsymmetric system, we use the IDR(s) method.

- P. Sonneveld, M.B. Van Gijzen. *SIAM J. Sci. Comput.*, 2008, 31(2), 1035-1062.

Optimal Control of PDEs

Consider the PDE-constrained optimization problem,

$$\begin{aligned}
\min_{u,\ f} \frac{1}{2}\|u-\hat{u}\|^2 \quad &+ \quad \beta\|f\|^2 \\
s.t.\ \mathcal{L}u \quad &= \quad f \text{ in } \Omega \\
u_D \quad &= \quad g_D \text{ on } \Gamma_D \\
\frac{\partial u}{\partial n} \quad &= \quad g_N \text{ on } \Gamma_N
\end{aligned}$$

$\mathcal{L}$ is an operator, $\hat{u}$ is the desired system state and f is the system input. For the optimal control of the convection-diffusion equation, we have $\mathcal{L} = -\epsilon\nabla^2 + \overrightarrow{w}.\nabla$

TUDelft

Numerical Experiments

Preconditioning of Optimal Control of PDEs (cont'd)

Approximation of the Schur Complement

By introducing the weak formulation and discretizing the system with Galerkin method, the discrete analogue of the minimization problem is therefore,

$$\min_{u,\ f} \ \frac{1}{2}u^T M u - u^T b + c + \beta f^T M f \tag{1}$$

$$s.t. \quad Ku = Mf + d \tag{2}$$

The optimal solution of (1)-(2) is given by solving the saddle-point system,

$$\begin{bmatrix} 2\beta M & 0 & -M \\ 0 & M & K^T \\ -M & K & 0 \end{bmatrix} \begin{bmatrix} \mathbf{f} \\ \mathbf{u} \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{b} \\ \mathbf{d} \end{bmatrix}. \tag{3}$$

$$\begin{bmatrix} 2\beta M & 0 & -M \\ 0 & M & K^T \\ -M & K & 0 \end{bmatrix} = \begin{bmatrix} I & & \\ 0 & I & \\ -\frac{1}{2\beta} I & KM^{-1} & I \end{bmatrix} \begin{bmatrix} 2\beta M & 0 & -M \\ & M & K^T \\ & & S \end{bmatrix}$$

where $S = -\left(\frac{1}{2\beta} M + KM^{-1}K^T\right)$ is the Schur complement. Here we use the MSSS computation technique to approximate S.

Numerical Experiments

Preconditioning of Optimal Control of PDEs (cont'd)

Computational Results

Set $\epsilon = 10^{-2}$, $\beta = 10^{-1}$, $\theta = \frac{\pi}{5}$ and the stopping tolerance $tol = 10^{-6}$.

Table: Pre. time by new MOR and convectional MOR and MINRES time

problem size	iter. NO.	pre. time (s)	MINRES time (s)	total (s)
$3 \times 2^{10}(4)$	16	0.77	1.55	2.32
$3 \times 2^{12}(4)$	16	1.21	3.40	4.61
$3 \times 2^{14}(4)$	16	3.82	10.34	14.16
$3 \times 2^{16}(4)$	16	13.47	34.12	47.59
problem size	iter. NO.	pre. time (s)	MINRES time (s)	total (s)
$3 \times 2^{10}(4)$	16	0.71	1.43	2.14
$3 \times 2^{12}(4)$	16	2.25	3.38	5.63
$3 \times 2^{14}(4)$	16	8.08	10.23	18.31
$3 \times 2^{16}(4)$	16	30.71	34.09	64.80

Numerical Experiments

Preconditioning of Optimal Control of PDEs (cont'd)

Computational Results

Set $\epsilon = 10^{-2}$, $\beta = 10^{-1}$, $\theta = \frac{\pi}{5}$ and the stopping tolerance $tol = 10^{-6}$.

Table: Pre. time by new MOR and convectional MOR and MINRES time

problem size	iter. NO.	pre. time (s)	MINRES time (s)	total (s)
$3 \times 2^{10}(4)$	16	0.77	1.55	2.32
$3 \times 2^{12}(4)$	16	1.21	3.40	4.61
$3 \times 2^{14}(4)$	16	3.82	10.34	14.16
$3 \times 2^{16}(4)$	16	13.47	34.12	47.59
problem size	iter. NO.	pre. time (s)	MINRES time (s)	total (s)
$3 \times 2^{10}(4)$	16	0.71	1.43	2.14
$3 \times 2^{12}(4)$	16	2.25	3.38	5.63
$3 \times 2^{14}(4)$	16	8.08	10.23	18.31
$3 \times 2^{16}(4)$	16	30.71	34.09	64.80

Numerical Experiments

Preconditioning of Optimal Control of PDEs (cont'd)

Computational Results

Set $\epsilon = 10^{-2}$, $\beta = 10^{-1}$, $\theta = \frac{\pi}{5}$ and the stopping tolerance $tol = 10^{-6}$.

Table: Pre. time by new MOR and convectional MOR and MINRES time

problem size	iter. NO.	pre. time (s)	MINRES time (s)	total (s)
$3 \times 2^{10}(4)$	16	0.77	1.55	2.32
$3 \times 2^{12}(4)$	16	1.21	3.40	4.61
$3 \times 2^{14}(4)$	16	3.82	10.34	14.16
$3 \times 2^{16}(4)$	16	13.47	34.12	47.59
problem size	iter. NO.	pre. time (s)	MINRES time (s)	total (s)
$3 \times 2^{10}(4)$	16	0.71	1.43	2.14
$3 \times 2^{12}(4)$	16	2.25	3.38	5.63
$3 \times 2^{14}(4)$	16	8.08	10.23	18.31
$3 \times 2^{16}(4)$	16	30.71	34.09	64.80

Computational Results

Set $\epsilon = 10^{-2}$, $\beta = 10^{-1}$, $\theta = \frac{\pi}{5}$ and the stopping tolerance $tol = 10^{-6}$.

Table: Pre. time by new MOR and convectional MOR and MINRES time

problem size	iter. NO.	pre. time (s)	MINRES time (s)	total (s)
$3 \times 2^{10}(4)$	16	0.77	1.55	2.32
$3 \times 2^{12}(4)$	16	1.21	3.40	4.61
$3 \times 2^{14}(4)$	16	3.82	10.34	14.16
$3 \times 2^{16}(4)$	16	13.47	34.12	47.59
problem size	iter. NO.	pre. time (s)	MINRES time (s)	total (s)
$3 \times 2^{10}(4)$	16	0.71	1.43	2.14
$3 \times 2^{12}(4)$	16	2.25	3.38	5.63
$3 \times 2^{14}(4)$	16	8.08	10.23	18.31
$3 \times 2^{16}(4)$	16	30.71	34.09	64.80

Numerical Experiments

Preconditioning of Optimal Control of PDEs (cont'd)

Computational Results

Set $\epsilon = 10^{-2}$, $\beta = 10^{-1}$, $\theta = \frac{\pi}{5}$ and the stopping tolerance $tol = 10^{-6}$.

Table: Pre. time by new MOR and convectional MOR and MINRES time

problem size	iter. NO.	pre. time (s)	MINRES time (s)	total (s)
$3 \times 2^{10}(4)$	16	0.77	1.55	2.32
$3 \times 2^{12}(4)$	16	1.21	3.40	4.61
$3 \times 2^{14}(4)$	16	3.82	10.34	14.16
$3 \times 2^{16}(4)$	16	13.47	34.12	47.59
problem size	iter. NO.	pre. time (s)	MINRES time (s)	total (s)
$3 \times 2^{10}(4)$	16	0.71	1.43	2.14
$3 \times 2^{12}(4)$	16	2.25	3.38	5.63
$3 \times 2^{14}(4)$	16	8.08	10.23	18.31
$3 \times 2^{16}(4)$	16	30.71	34.09	64.80

Conclusions & Remarks

1. MSSS matrices computation provides an efficient approach to construct a preconditioner.
2. The new model order reduction algorithm and the convectional algorithm gives efficient preconditioner of linear computational complexity.
3. Both preconditioner are almost independent of the mesh size.
4. The new algorithm is computationally cheaper than the convectional one.
5. Extension to 3D is a big challenge because of the model reduction of the higher level, some work has already been done

Conclusions & Remarks

1. MSSS matrices computation provides an efficient approach to construct a preconditioner.
2. The new model order reduction algorithm and the convectional algorithm gives efficient preconditioner of linear computational complexity.
3. Both preconditioner are almost independent of the mesh size.
4. The new algorithm is computationally cheaper than the convectional one.
5. Extension to 3D is a big challenge because of the model reduction of the higher level, some work has already been done

Conclusions & Remarks

1. MSSS matrices computation provides an efficient approach to construct a preconditioner.
2. The new model order reduction algorithm and the convectional algorithm gives efficient preconditioner of linear computational complexity.
3. Both preconditioner are almost independent of the mesh size.
4. The new algorithm is computationally cheaper than the convectional one.
5. Extension to 3D is a big challenge because of the model reduction of the higher level, some work has already been done

Conclusions & Remarks

1. MSSS matrices computation provides an efficient approach to construct a preconditioner.
2. The new model order reduction algorithm and the convectional algorithm gives efficient preconditioner of linear computational complexity.
3. Both preconditioner are almost independent of the mesh size.
4. The new algorithm is computationally cheaper than the convectional one.
5. Extension to 3D is a big challenge because of the model reduction of the higher level, some work has already been done

TUDelft

Conclusions & Remarks

1. MSSS matrices computation provides an efficient approach to construct a preconditioner.
2. The new model order reduction algorithm and the convectional algorithm gives efficient preconditioner of linear computational complexity.
3. Both preconditioner are almost independent of the mesh size.
4. The new algorithm is computationally cheaper than the convectional one.
5. Extension to 3D is a big challenge because of the model reduction of the higher level, some work has already been done

Conclusions & Remarks

1. MSSS matrices computation provides an efficient approach to construct a preconditioner.
2. The new model order reduction algorithm and the convectional algorithm gives efficient preconditioner of linear computational complexity.
3. Both preconditioner are almost independent of the mesh size.
4. The new algorithm is computationally cheaper than the convectional one.
5. Extension to 3D is a big challenge because of the model reduction of the higher level, some work has already been done

TUDelft

Preconditioning of 3D Problems

Precondition of 3D Poisson Equation

Consider the 3D Poisson equation

$$-\left(\frac{\partial}{\partial x}u(x,\ y,\ z)+\frac{\partial}{\partial y}u(x,\ y,\ z)+\frac{\partial}{\partial z}u(x,\ y,\ z)\right) \quad = \quad f$$

$$u = g_D \text{ on } \Gamma_D$$

Discretize the 3D Poisson equation with finite difference method, we have the following system,

$$\Phi u = b$$

with

$$\Phi = \begin{bmatrix} M & -L & & & \\ -L & M & -L & & \\ & -L & M & -L & \\ & & \ddots & \ddots & \ddots \\ & & & -L & M \end{bmatrix},\ M = \begin{bmatrix} D & -P & & & \\ -P & D & -P & & \\ & -P & D & -P & \\ & & \ddots & \ddots & \ddots \\ & & & -P & D \end{bmatrix}$$

Preconditioning of 3D Problems

Precondition of 3D Poisson Equation (cont'd)

$$L = \begin{bmatrix} R & Q & & & \\ Q & R & Q & & \\ & Q & R & Q & \\ & & \ddots & \ddots & \ddots \\ & & & Q & R \end{bmatrix}, \quad D = \frac{1}{30}\begin{bmatrix} 128 & -14 & & & \\ -14 & 128 & -14 & & \\ & -14 & 128 & -14 & \\ & & \ddots & \ddots & \ddots \\ & & & -14 & 128 \end{bmatrix}$$

$$R = P = \frac{1}{30}\begin{bmatrix} 14 & 3 & & & \\ 3 & 14 & 3 & & \\ & 3 & 14 & 3 & \\ & & \ddots & \ddots & \ddots \\ & & & 3 & 14 \end{bmatrix}, \quad Q = \frac{1}{30}\begin{bmatrix} 3 & 1 & & & \\ 1 & 3 & 1 & & \\ & 1 & 3 & 1 & \\ & & \ddots & \ddots & \ddots \\ & & & 1 & 3 \end{bmatrix}$$

To do the LDU factorization of the matrix Φ, the Schur complement S_i are

$$\begin{cases} S_1 = M \\ S_{i+1} = M - L S_i^{-1} L^T \end{cases}$$

TU Delft

Preconditioning of 3D Problems

Preconditioning of 3D Poisson Equation (cont'd)

Approximate the Schur complement S_i with MSSS computation technique and solve the linear system with conjugate gradient (CG) method, set the stopping tolerance $tol = 10^{-6}$, we have the following results.

Table: Pre. time by new MOR and the CG time

problem size	iter. NO.	pre. time (s)	CG time (s)	total (s)
$2^6(4)$	3	1.97	0.11	2.08
$2^9(4)$	4	4.32	0.72	5.04
$2^{12}(4)$	5	8.68	3.92	12.60
$2^{15}(4)$	8	29.46	30.64	60.10
$2^{18}(4)$	12	112.71	235.80	348.51

Reference

- Y. Qiu, M.B. van Gijzen, et al. *Tech. Rept. 13-04. Delft Institute of Applied Mathematics, Delft University of Technology, 2013.*

Preconditioning of 3D Problems

Preconditioning of 3D Poisson Equation (cont'd)

Approximate the Schur complement S_i with MSSS computation technique and solve the linear system with conjugate gradient (CG) method, set the stopping tolerance $tol = 10^{-6}$, we have the following results.

Table: Pre. time by new MOR and the CG time

problem size	iter. NO.	pre. time (s)	CG time (s)	total (s)
$2^6(4)$	3	1.97	0.11	2.08
$2^9(4)$	4	4.32	0.72	5.04
$2^{12}(4)$	5	8.68	3.92	12.60
$2^{15}(4)$	8	29.46	30.64	60.10
$2^{18}(4)$	12	112.71	235.80	348.51

Reference

- Y. Qiu, M.B. van Gijzen, et al. *Tech. Rept. 13-04. Delft Institute of Applied Mathematics, Delft University of Technology, 2013.*

Preconditioning of 3D Problems

Preconditioning of 3D Poisson Equation (cont'd)

Approximate the Schur complement S_i with MSSS computation technique and solve the linear system with conjugate gradient (CG) method, set the stopping tolerance $tol = 10^{-6}$, we have the following results.

Table: Pre. time by new MOR and the CG time

problem size	iter. NO.	pre. time (s)	CG time (s)	total (s)
$2^6(4)$	3	1.97	0.11	2.08
$2^9(4)$	4	4.32	0.72	5.04
$2^{12}(4)$	5	8.68	3.92	12.60
$2^{15}(4)$	8	29.46	30.64	60.10
$2^{18}(4)$	12	112.71	235.80	348.51

Reference

- Y. Qiu, M.B. van Gijzen, et al. *Tech. Rept. 13-04. Delft Institute of Applied Mathematics, Delft University of Technology, 2013.*

Preconditioning of 3D Problems

Preconditioning of 3D Poisson Equation (cont'd)

Approximate the Schur complement S_i with MSSS computation technique and solve the linear system with conjugate gradient (CG) method, set the stopping tolerance $tol = 10^{-6}$, we have the following results.

Table: Pre. time by new MOR and the CG time

problem size	iter. NO.	pre. time (s)	CG time (s)	total (s)
$2^6(4)$	3	1.97	0.11	2.08
$2^9(4)$	4	4.32	0.72	5.04
$2^{12}(4)$	5	8.68	3.92	12.60
$2^{15}(4)$	8	29.46	30.64	60.10
$2^{18}(4)$	12	112.71	235.80	348.51

Reference

- Y. Qiu, M.B. van Gijzen, et al. *Tech. Rept. 13-04. Delft Institute of Applied Mathematics, Delft University of Technology, 2013.*

TUDelft

Preconditioning of 3D Problems

Preconditioning of 3D Poisson Equation (cont'd)

Approximate the Schur complement S_i with MSSS computation technique and solve the linear system with conjugate gradient (CG) method, set the stopping tolerance $tol = 10^{-6}$, we have the following results.

Table: Pre. time by new MOR and the CG time

problem size	iter. NO.	pre. time (s)	CG time (s)	total (s)
$2^6(4)$	3	1.97	0.11	2.08
$2^9(4)$	4	4.32	0.72	5.04
$2^{12}(4)$	5	8.68	3.92	12.60
$2^{15}(4)$	8	29.46	30.64	60.10
$2^{18}(4)$	12	112.71	235.80	348.51

Reference

- Y. Qiu, M.B. van Gijzen, et al. *Tech. Rept. 13-04. Delft Institute of Applied Mathematics, Delft University of Technology, 2013.*

Thanks for your attention! Any questions, suggestions or remarks?