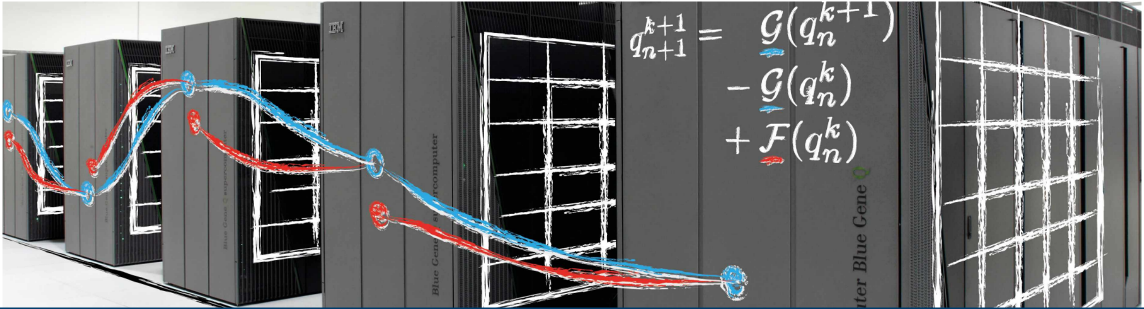
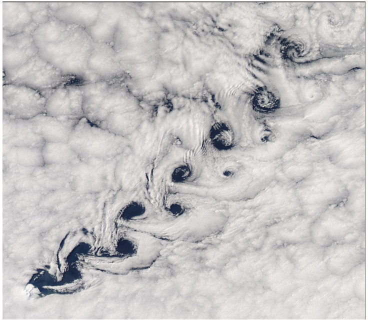


PARALLEL SDC FOR FLUID FLOW SIMULATIONS

December 18th, 2024 | Abdelouahed Ouardghi, Robert Speck | Jülich Supercomputing Centre, Forschungszentrum Jülich

Outline

Motivation



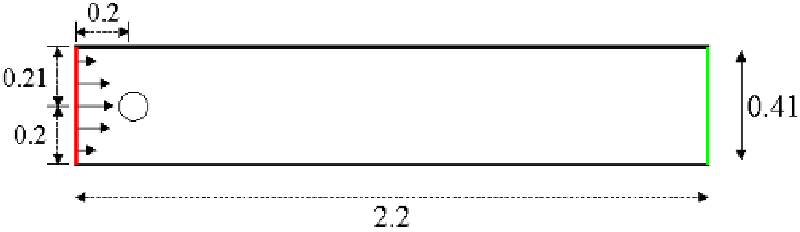
DFG flow around cylinder: Benchmark 2D-3

$$
\left\{
\begin{array}{rclll}
\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} + \nabla p - \nu\Delta\mathbf{u} & = & \mathbf{g} & \text{in} & \Omega\times[0,T] \\
\nabla\cdot\mathbf{u} & = & 0 & \text{in} & \Omega\times[0,T] \\
\mathbf{u} & = & \mathbf{u}_{in} & \text{on} & \Gamma_1\times[0,T] \\
\mathbf{u} & = & 0 & \text{on} & \Gamma_2\times[0,T] \\
\nu\partial_{\mathbf{n}}\mathbf{u} - p\mathbf{n} & = & 0 & \text{on} & \Gamma_3\times[0,T]
\end{array}
\right.
$$

with

$$\mathbf{u}_{in} = (\frac{4Uy(0.41-y)}{0.41^2}, 0),$$

$$U = U(t) = 1.5sin(\pi t/8)$$



JÜLICH Forschungszentrum

IMEX SDC (SISDC) using projection-based splitting scheme

DFG Flow around cylinder: Chorin's projection method

1. **Step 1: Predictor Step:**

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} + (\mathbf{u}^n \cdot \nabla)\mathbf{u}^n = \nu \Delta \mathbf{u}^* + \mathbf{g}$$

2. **Step 2: Corrector Step:**

$$\Delta p^{n+1} = \frac{1}{\Delta t} \nabla \cdot \mathbf{u}^*$$

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \Delta t \nabla p^{n+1}$$

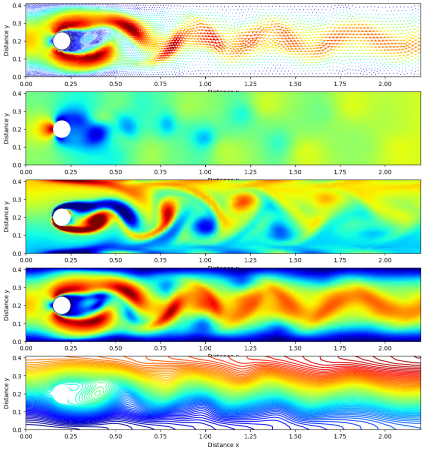


Figure: Velocity, pressure, vorticity, magnitude and streamlines at time $t = 5.5s$

JÜLICH Forschungszentrum

IMEX SDC (SISDC) using projection-based splitting scheme

Convection-diffusion equation

- Convection-diffusion equation

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla)\mathbf{u} + \nu \Delta \mathbf{u} + \mathbf{g}$$

- Spatial discretization

$$[\mathtt{M}]\frac{d\mathbf{u}}{dt} = -[\mathtt{C}_{\mathbf{u}}]\mathbf{u} - [\mathtt{k}]\mathbf{u} + [\mathtt{M}]\mathbf{g}$$

- ODE

$$[\mathtt{M}]\frac{d\mathbf{u}}{dt} = f(\mathbf{u}, t) = f_E(\mathbf{u}, t) + f_I(\mathbf{u}, t),$$

where $f_E(\mathbf{u}, t) = -[\mathtt{C}_{\mathbf{u}}]\mathbf{u}$ and $f_I(\mathbf{u}, t) = -[\mathtt{k}]\mathbf{u} + [\mathtt{M}]\mathbf{g}$

SISDC: Semi-implicit spectral deferred corrections method

- Consider the fully implicit collocation problem

$$(\mathcal{M} - \Delta t Q F)\vec{\mathbf{u}} = \mathcal{M}\vec{\mathbf{u}}_0,$$

where $\mathcal{M} = \mathbf{I}_M \otimes [\mathtt{M}]$, $\vec{\mathbf{u}} = (\mathbf{u}_1, \cdots, \mathbf{u}_M)^T$, $\vec{\mathbf{u}}_0 = (\mathbf{u}_0, \cdots, \mathbf{u}_0)^T$, and $F(\vec{\mathbf{u}}) = (\mathbf{f}(\mathbf{u}_1), \cdots, \mathbf{f}(\mathbf{u}_M))$.

- Precondtionated Picard iteration for the collocation problem

$$(\mathcal{M} - \Delta t Q_\Delta F)\, \vec{\mathbf{u}}^{k+1} = \mathcal{M}\vec{\mathbf{u}}_0 + \Delta t\, (Q - Q_\Delta)\, F(\vec{\mathbf{u}}^k).$$

- IMEX SDC sweep

$$(\mathcal{M} - \Delta t(Q_{\Delta,E}F_E + Q_{\Delta,I}F_I))\, \vec{\mathbf{u}}^{k+1} = \mathcal{M}\vec{\mathbf{u}}_0 + \Delta t Q F(\vec{\mathbf{u}}^k) - \Delta t\, (Q_{\Delta,E}F_E + Q_{\Delta,I}F_I)\, \vec{\mathbf{u}}^k.$$

$\Longrightarrow$ $Q_{\Delta,I}$ lower triangular and $Q_{\Delta,E}$ strictly lower triangular for explicit integration

IMEX SDC (SISDC) for NSE using projection-based splitting scheme

Step 1 Predictor step:

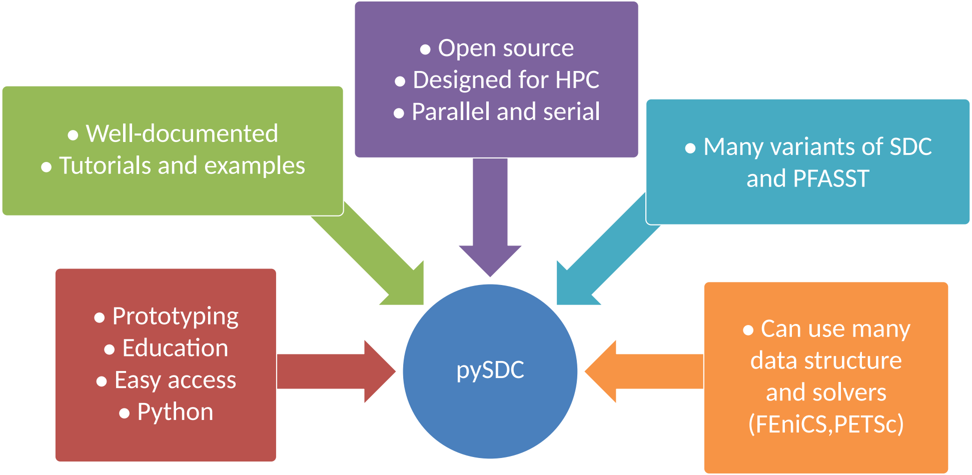
$$
\begin{aligned}
[\mathtt{M}]\mathbf{u}_m^{k+1} \quad = \quad & [\mathtt{M}]\mathbf{u}_0 + \Delta t \sum_{j=1}^{M} q_{m,j} f(\mathbf{u}_j^k, \tau_j) \\
& + \Delta t \sum_{j=1}^{m-1} q_{m,j}^{\Delta,E} \left[f_E(\mathbf{u}_j^{k+1}, \tau_j) - f_E(\mathbf{u}_j^k, \tau_j) \right] + \Delta t \sum_{j=1}^{m} q_{m,j}^{\Delta,I} \left[f_I(\mathbf{u}_j^{k+1}, \tau_j) - f_I(\mathbf{u}_j^k, \tau_j) \right]
\end{aligned}
$$

Step 2 Corrector step:

$$
\Delta p_m^{k+1} = \frac{1}{\Delta t} \nabla \cdot \mathbf{u}_m^{k+1}
$$

$$
\mathbf{u}_m^{k+1} = \mathbf{u}_m^{k+1} - \Delta t \nabla p_m^{k+1}
$$

pySDC prototyping



`https://parallel-in-time.org/pySDC`

IMEX SDC (SISDC) for advection-diffusion equations

Advection-diffusion of a Gaussian hill problem

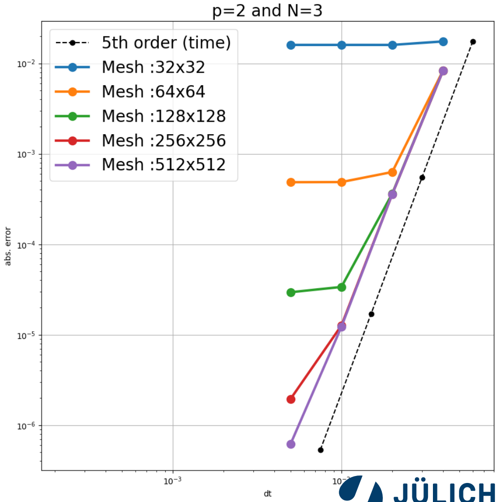
1 Mathematical model

$$\begin{cases} \frac{\partial c}{\partial t} + \mathbf{v} \cdot \nabla c - \nu \Delta c = f & \text{in} \quad \Omega \times [0, T] \\ c = g & \text{on} \quad \partial\Omega \times [0, T] \\ c(x, y, 0) = c_0(x, y) & \text{in} \quad \Omega \end{cases}$$

2 Problem setup

$$c(x, y, t) = \frac{5}{7\sqrt{1 + \frac{4\nu t}{l^2}}} exp \left\{ - \left(\frac{x - x_0 - t}{l\sqrt{1 + \frac{4\nu t}{l^2}}} \right)^2 \right\}$$

with $l = 7\frac{\sqrt{2}}{300}$, $x_0 = \frac{2}{15}$ and $\Omega = [0, 1] \times [0, 1]$



IMEX SDC (SISDC) using projection-based splitting scheme

Lift coefficient using different SDC

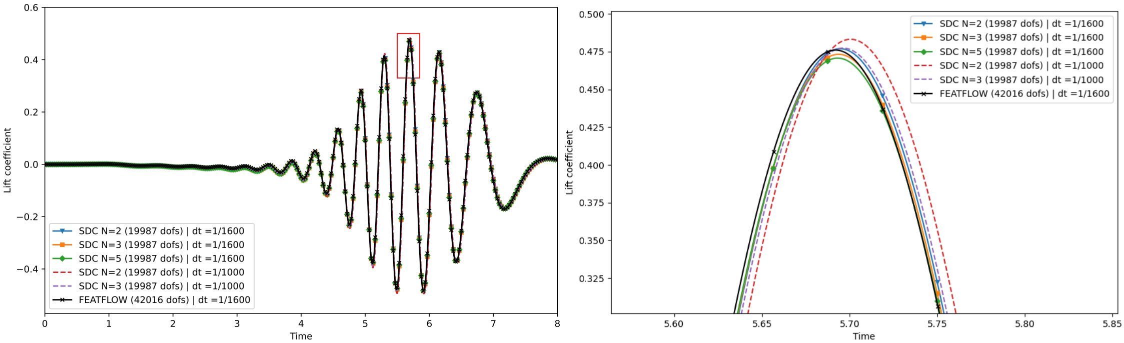


Figure: Lift coefficient using different SDC setups and time steps

Parallel SDC: Diagonal Preconditioners

G. Čaklović et al., Improving Efficiency of Parallel Across the Method Spectral Deferred Corrections. *Preprint in arXiv*

Diagonal SDC: $Q_{\Delta,I} = Q_\Delta^{diag} = diag\left(d_1, \cdots, d_M\right)$ and $Q_{\Delta,E} = 0$

$$[\mathtt{M}]u_m^{k+1} = [\mathtt{M}]u_0 + \Delta t \sum_{j=1}^{M} q_{m,j} f(\tau_m, u_j^k) + \Delta t d_m \left[f_I(\tau_m, u_m^{k+1}) - f_I(\tau_m, u_m^k)\right]$$

1 **Diagonal implicit Euler (IEpar)**

$$Q_\Delta^{IEpar} = diag(\tau_i - t_0),$$

with τ_i the nodes of the quadrature rule.

2 **MIN-SR-NS diagonal coefficients**

For any collocation method on M distinct nodes, then $\rho(Q - Q_\Delta) = 0$ for

$$Q_\Delta = diag\left(\frac{\tau_1}{M}, \cdots, \frac{\tau_M}{M}\right)$$

3 **MIN-SR-S diagonal coefficients**

We minimize $\rho(I - Q_\Delta^{-1} Q)$ by choosing diagonal coefficients such that

$$det\left((1-z)I + zQ_\Delta^{-1}Q\right) = 1, \qquad z \in \{\tau_1, \cdots, \tau_M\}$$

IMEX SDC (SISDC) using projection-based splitting scheme

Comparison of different preconditioners

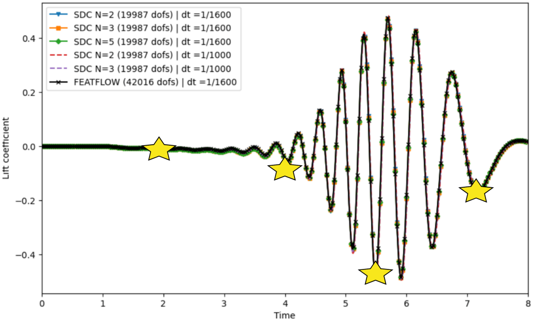


Figure: Lift coefficient using different SDC setups and time steps

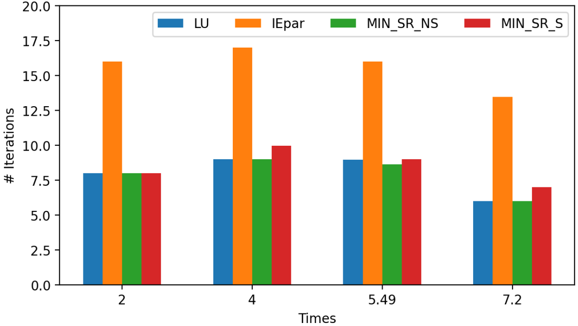


Figure: Average number of iterations needed by various preconditioners after 50 timesteps at four different time points throughout the simulation

Fully implicit SDC using the monolithic approach

- Navier-Stokes equations

$$\begin{aligned}\frac{\partial \mathbf{u}}{\partial t} &= -\mathbf{u}\cdot\nabla\mathbf{u} + \nabla p + \nu\Delta\mathbf{u} + \mathbf{g}\\ 0 &= \nabla\cdot\mathbf{u}\end{aligned}$$

- Semi-discrete form of the Navier-Stokes equations.

$$\begin{aligned}[\mathtt{M}]\frac{d\mathbf{u}}{dt} &= -[\mathtt{C}_{\mathbf{u}}]\mathbf{u} - [\mathtt{k}]\mathbf{u} + [\mathtt{B}]p + [\mathtt{M}]\mathbf{g},\\ \mathbf{0} &= [\mathtt{B}^T]\mathbf{u}\end{aligned}$$

- Differential algebraic equations (DAEs)

$$\begin{aligned}[\mathtt{M}]\frac{d\mathbf{u}}{dt} &= \mathcal{F}(\mathbf{u}, p, t),\\ 0 &= \mathcal{G}(\mathbf{u}, t).\end{aligned}$$

Fully implicit SDC using the monolithic approach: DAEs sweeper

- with $\frac{d\mathbf{u}}{dt} = \mathbf{U}$ the SDC iteration for $\mathbf{u}_m^{k+1}$ can be written as

$$\mathbf{u}_m^{k+1} = \mathbf{u}_0 + \Delta t \sum_{j=1}^{M} q_{m,j} \mathbf{U}_j^k + \Delta t \sum_{j=1}^{m} q_{m,j}^{\Delta} \left[\mathbf{U}_j^{k+1} - \mathbf{U}_j^k \right]$$

Thus

$$\begin{aligned}
[\mathtt{M}] \overbrace{\mathbf{U}_m^{k+1}}^{= \frac{d\mathbf{u}}{dt}} &= \mathcal{F}\left(\overbrace{\mathbf{u}_0 + \Delta t \sum_{j=1}^{M} q_{m,j} \mathbf{U}_j^k + \Delta t \sum_{j=1}^{m} q_{m,j}^{\Delta} \left[\mathbf{U}_j^{k+1} - \mathbf{U}_j^k \right]}^{= \mathbf{u}_m^{k+1}}, \quad p_m^{k+1}, \quad \tau_m \right), \\
0 &= \mathcal{G}\left(\mathbf{u}_0 + \Delta t \sum_{j=1}^{M} q_{m,j} \mathbf{U}_j^k + \Delta t \sum_{j=1}^{m} q_{m,j}^{\Delta} \left[\mathbf{U}_j^{k+1} - \mathbf{U}_j^k \right], \tau_m \right).
\end{aligned}$$

Fully implicit SDC using the monolithic approach: DAEs sweeper

Using the matrix notation we obtain

$$\begin{pmatrix} [\mathtt{M}] + \Delta t q^{\Delta}_{m,m}\left([\mathtt{C}_{\mathbf{u}}] + [\mathtt{k}]\right) & [\mathtt{B}] \\ -\Delta t q^{\Delta}_{m,m}[\mathtt{B}^T] & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{U}^{k+1}_m \\ p^{k+1}_m \end{pmatrix} = \begin{pmatrix} -\left([\mathtt{C}_{\mathbf{u}}] + [\mathtt{k}]\right)\tilde{u}_m + [\mathtt{M}]\mathbf{g}_m \\ [\mathtt{B}^T]\tilde{u}_m \end{pmatrix}, \tag{1}$$

with

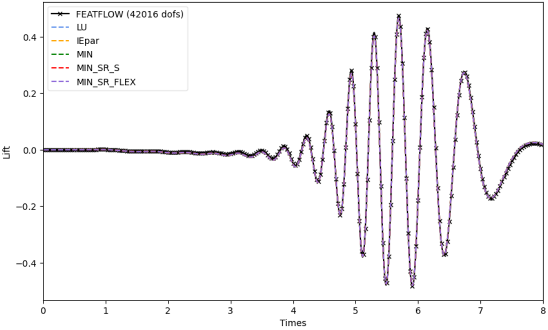
$$\tilde{u}_m = \mathbf{u}_0 + \Delta t \sum_{j=1}^{M} q_{m,j}\mathbf{U}^k_j - \Delta t \sum_{j=1}^{m} q^{\Delta}_{m,j}\mathbf{U}^k_j + \Delta t \sum_{j=1}^{m-1} q^{\Delta}_{m,j}\mathbf{U}^{k+1}_j$$

When this problem converges to a solution $(\mathbf{U}^s_m, p^s_m)$ for $m = 1, \cdots, M$, the final solutions can be computed as follows

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \sum_{j=1}^{M} q_{m,j}\mathbf{U}^s_j \quad \text{and} \quad p^{n+1} = p^s_M \tag{2}$$

Fully implicit SDC using the monolithic approach: DAEs sweeper

Comparison of different preconditioners



Fully implicit SDC using the monolithic approach: DAEs sweeper

Comparison of different preconditioners

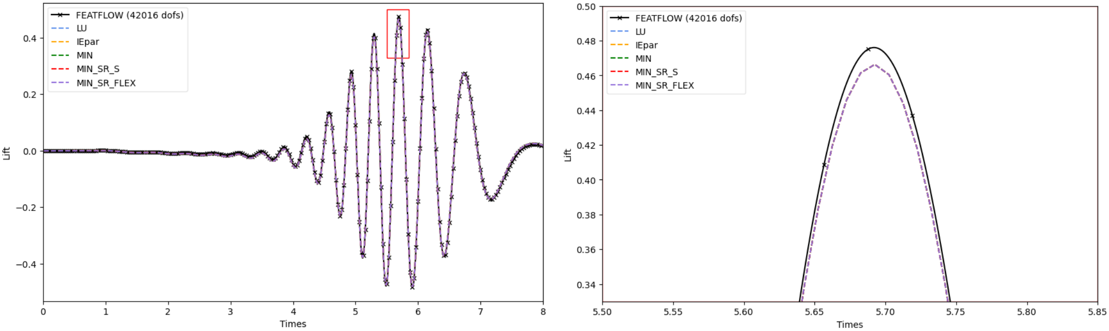


Figure: Lift coefficient using different SDC setups and time steps

Fully implicit SDC using the monolithic approach: DAEs sweeper

Comparison of different preconditioners

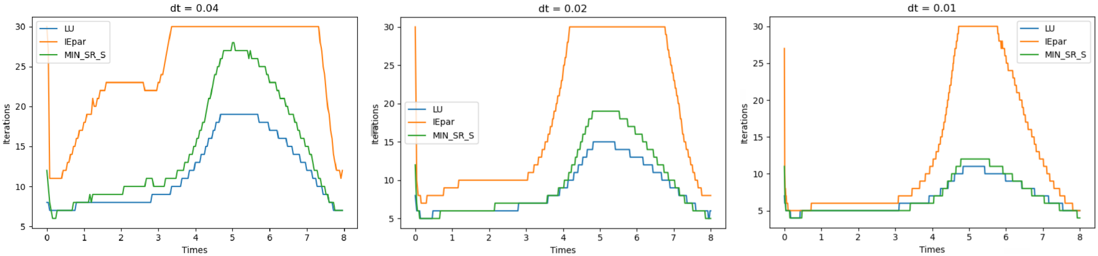


Figure: Number of iterations needed by various preconditioners throughout the simulation

Fully implicit SDC using the monolithic approach: PDEs sweeper

- Semi-discrete form of the Navier-Stokes equations.

$$
\begin{aligned}
[\mathtt{M}]\frac{d\mathbf{u}}{dt} &= -[\mathtt{C}_{\mathbf{u}}]\mathbf{u} - [\mathtt{k}]\mathbf{u} + [\mathtt{B}]p + [\mathtt{M}]\mathbf{g}, \\
\mathbf{0} &= [\mathtt{B}^T]\mathbf{u}
\end{aligned}
$$

- Matrix-vector form of the semi-discrete form

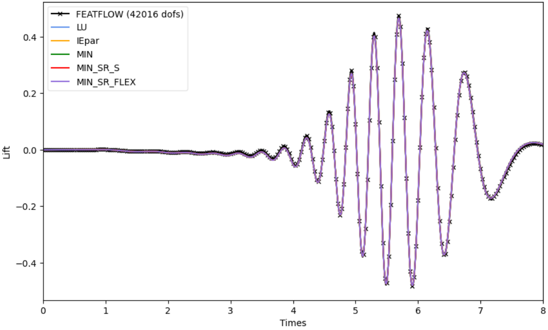
$$
\begin{pmatrix} [\mathtt{M}] & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \frac{d\mathbf{u}}{dt} \\ \frac{dp}{dt} \end{pmatrix} = \begin{pmatrix} -[\mathtt{C}_{\mathbf{u}}] - [\mathtt{k}] & [\mathtt{B}] \\ [\mathtt{B}^T] & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ p \end{pmatrix} + \begin{pmatrix} [\mathtt{M}]\mathbf{g} \\ 0 \end{pmatrix}.
$$

- Ordinary differential equation (ODE)

$$
[\alpha]\frac{d\mathbf{w}}{dt} = [\beta]\mathbf{w} + \gamma = f(\mathbf{w}, t),
$$

Fully implicit SDC using the monolithic approach: PDEs sweeper

Comparison of different preconditioners



Fully implicit SDC using the monolithic approach: PDEs sweeper

Comparison of different preconditioners

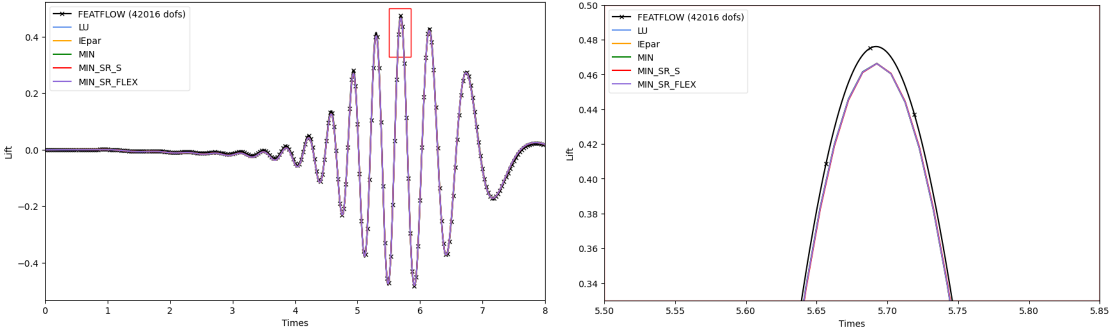


Figure: Lift coefficient using different SDC setups and time steps

Fully implicit SDC using the monolithic approach: PDEs sweeper

Comparison of different preconditioners

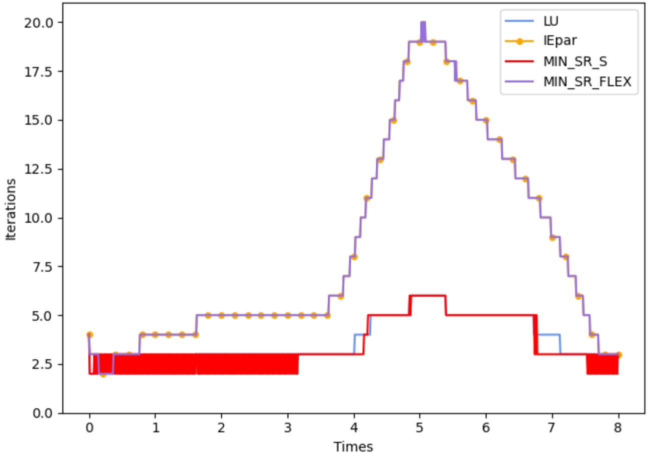


Figure: Number of iterations needed by various preconditioners throughout the simulation

Fully implicit SDC using the monolithic approach: PDEs sweeper

Comparison of different preconditioners

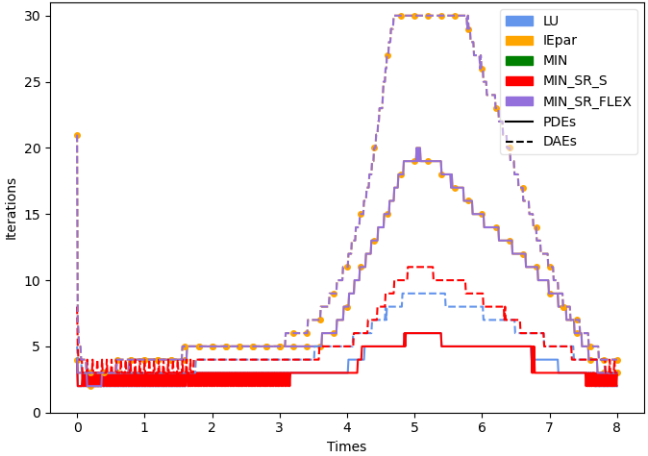
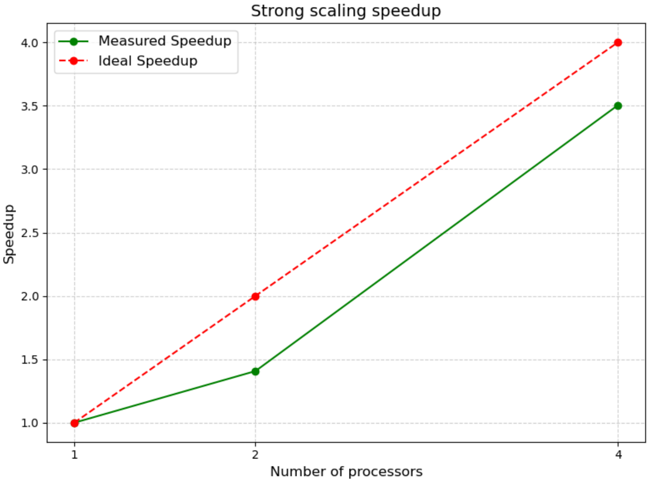


Figure: Number of iterations needed by various preconditioners throughout the simulation

Fully implicit SDC using the monolithic approach: PDEs sweeper



Summary

- SDC method can use simple numerical method (even a first-order method) to compute a solution with higher-order accuracy.
- SDC can be used for various initial value problems, using explicit, implicit or implicit- explicit schemes.
- parallelization can be done across the method (i.e. using diagonal preconditioners).

what's next?

- Assess code performance with advanced numerical test cases to ensure robustness.
- Optimize parallel execution and analyze the space-time speedup.
- Implement SDC in the FEAT3 toolbox.

Acknowledgment

Federal Ministry of Education and Research

Novel Exascale-Architectures with Heterogeneous Hardware Components for CFD Simulations

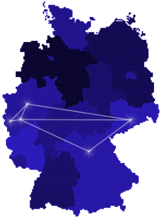
STROEMUNGSRAUM
SCALEXA HPC DEEPTECH

Project goals

- Enhance scalability and efficiency of FEATFLOW
- Develop exascale-ready methods
- Provide IANUS customers with access to methods, codes, and knowledge.

Project partners

- IANUS Simulation GmbH
- TU Dortmund University - Coordinator
- Jülich Supercomputing Centre
- University of Cologne
- Friedrich Alexander University Erlangen -Nuremberg
- Freiberg University of Technology



JÜLICH Forschungszentrum

Fully implicit SDC using the monolithic approach

Lift coefficient using different SDC

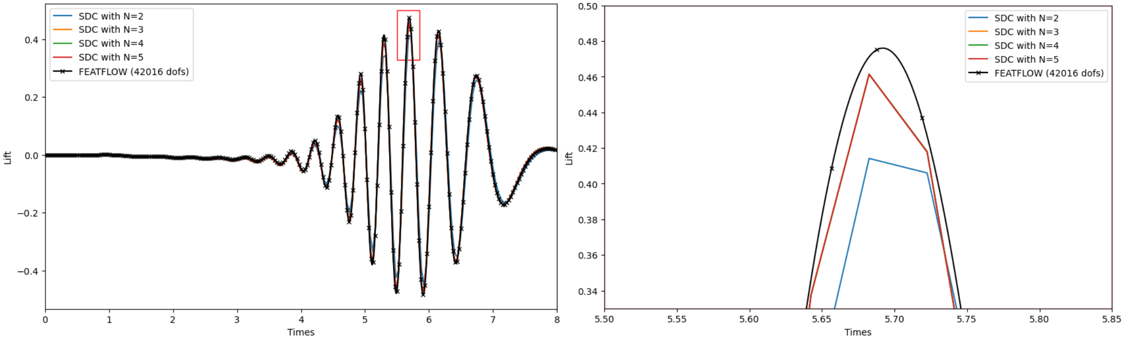


Figure: Lift coefficient using various SDC methods with $\Delta t = \frac{1}{25}$, compared to FEATFLOW reference data with $\Delta t = \frac{1}{1600}$