U.S. ARMY COMBAT CAPABILITIES DEVELOPMENT COMMAND
AVIATION & MISSILE CENTER

High-Order Temporal Approaches for Overset Applications and AMR Grids

Dylan Jude, Jayanarayanan Sitaraman, and Andrew Wissink

U.S. Army DEVCOM AvMC

Moffett Field, CA, USA

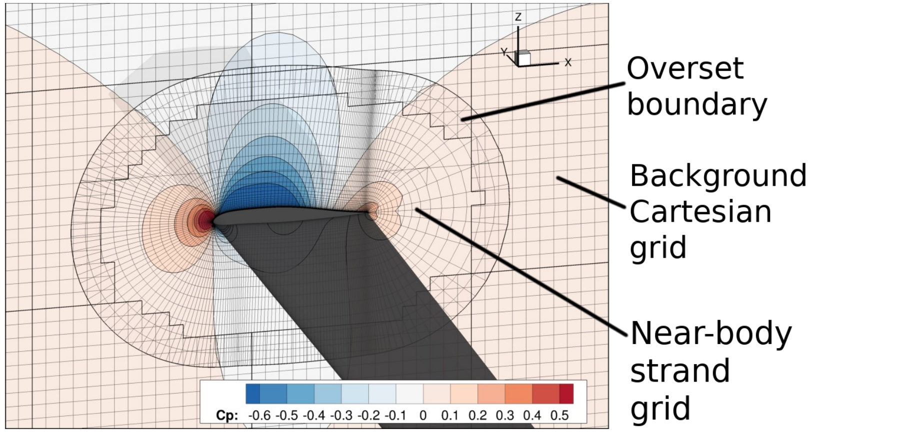
11 07 2022

OUTLINE

- **Introduction and motivation: Rotorcraft CFD**
- **Methodology**
  - Governing equations
  - Time-stepping algorithms
  - Helios framework
- **Results**
  - Vortex convection in Cartesian grids
  - Wing-vortex interaction
  - Pitching wing aerodynamics
- **Conclusions**

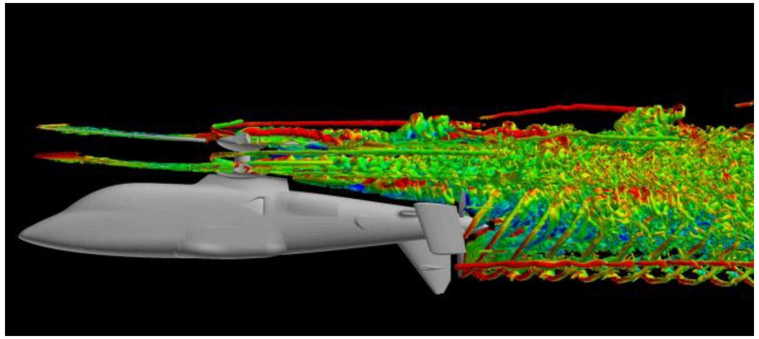
OVERSET CFD FOR ROTORCRAFT APPLICATIONS

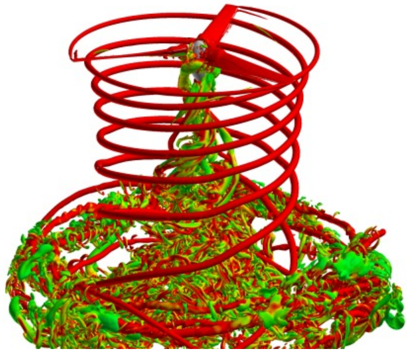
- **Overset or “Chimera” approach:**
  - Near-body and off-body meshes are overlapped in a domain
  - A domain connectivity tool performs hole-cutting and interpolation between meshes
  - Multiple near-body CFD solvers may all be overset within a background mesh

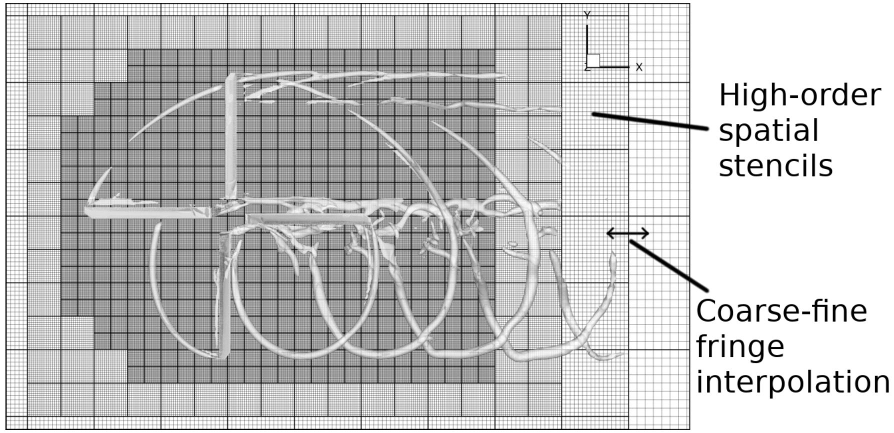


- **Benefits for rotorcraft:**
  - Blades move and deform independently without re-meshing
  - Different mesh resolutions or solver fidelity can be focused on areas of interest (eg. Near-blade, wake, fuselage)

- **HPCMP CREATE™-AV Helios**



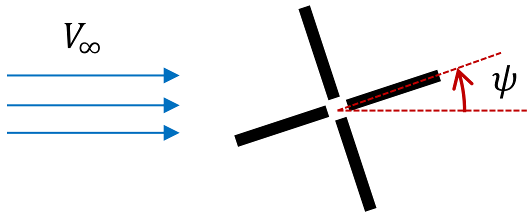




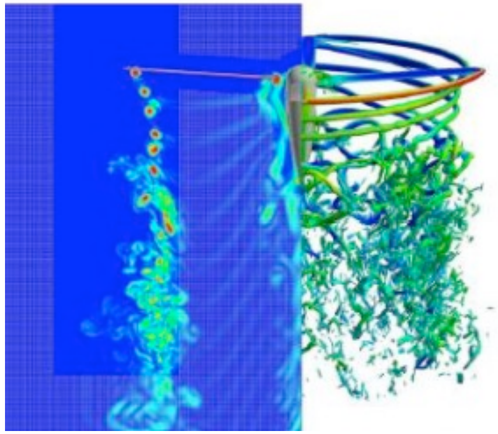
ROTORCRAFT CFD BEST PRACTICES

- **Decades of expert use with codes like Helios and NASA's OVERFLOW best practices:**
  - High-order spatial schemes but not temporal schemes
  - Azimuthal timestep of $\Delta\psi = 0.25°$
  - Second order, implicit Backwards Differencing Formulae (BDF) schemes generally used for timestepping

$$\frac{\partial q}{\partial t} \approx \frac{3q^{n+1} - 4q^{n} + q^{n-1}}{2\Delta t}$$



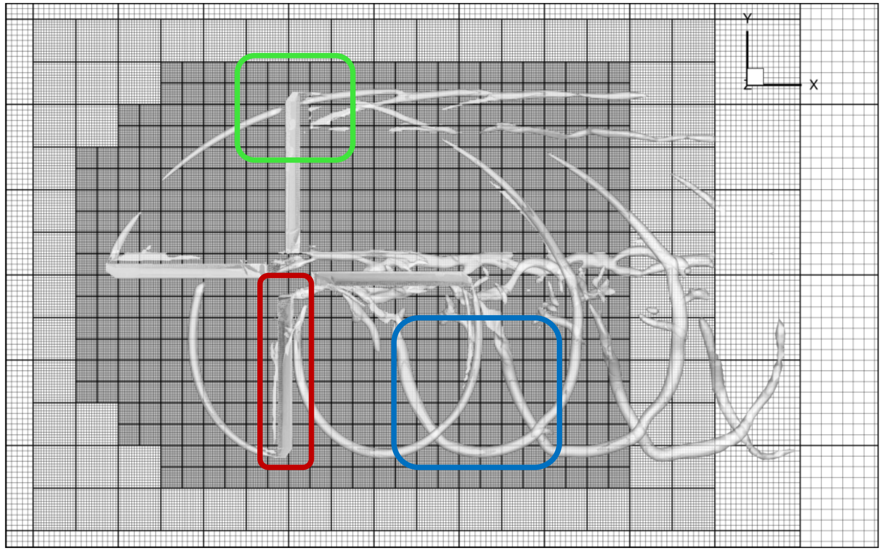
- **Limitations:**
  - BDF2 is not "self-starting". Two old solutions are required to properly advance in time
    - Restarting a simulation with a different timestep size is not possible
    - Changing the temporal order during a simulation is also not possible
  - BDF2 is second-order accurate and dominated by dispersive error
    - Small timesteps required for accurate solutions
    - Dispersion error may contribute to rotor wake breakdown
  - BDF methods of order > 2 are not *A*-stable



Narducci, R., Jain, R., Abras, J., Hariharan, N., SciTech 2021

TIMESTEPPING ALTERNATIVES

- **BDF2 is not the only option for unconditionally stable timestepping**
- **Runge-Kutta methods are also often used for timestepping**
  – Explicit methods suffer from timestep limitations, not good for rotorcraft CFD
  – Implicit RK methods have been successfully used by many other CFD codes
- **Goal for this work: Investigate implicit RK methods for overset, multi-solver frameworks like Helios**
  – Modify overset infrastructure to accommodate multi-stage schemes
  – Implement second, third, and fourth-order SDIRK schemes
  – Validate and compare results using three canonical problems representative of challenges in rotorcraft



1. Vortex convection in Cartesian grids
2. Blade-vortex interactions
3. Pitching wing aerodynamics

Methodology

GOVERNING EQUATIONS

- **Strong form of the Navier-Stokes equations:**

$$\frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial \mathbf{F}_i}{\partial x} + \frac{\partial \mathbf{G}_i}{\partial y} + \frac{\partial \mathbf{H}_i}{\partial z} = \frac{\partial \mathbf{F}_v}{\partial x} + \frac{\partial \mathbf{G}_v}{\partial y} + \frac{\partial \mathbf{H}_v}{\partial z} + \mathbf{S} \quad \Longrightarrow \quad \frac{\partial \mathbf{Q}}{\partial t} + \mathbf{R}(\mathbf{Q}) = 0$$

- **Using a multi-stage RK scheme:**

$$\mathbf{Q}^{n+1} = \mathbf{Q}^n + h\sum_{j=1}^{s} A_{sj} k_j$$

where

$$k_i = -\mathbf{R}(\mathbf{Q}_i), \quad \mathbf{Q}_i = \mathbf{Q}^n + h\sum_{j=1}^{s} A_{ij} k_j, \quad h = \Delta t$$

  - The ***A*** matrix comes from the Butcher table for the scheme:

c_1	a_{11}	a_{12}	…	a_{14}
c_2	a_{21}	a_{22}	…	a_{24}
…	…	…	…	…
c_n	a_{n1}	a_{n2}	…	a_{nn}
1	b_1	b_2	…	b_n

Timestep fraction

Scheme coefficients for linear combinations of solutions at each stage.

  - Assume ***A*** is lower triangular, linearize, and add pseudo-time continuation:

$$\left(\frac{1}{A_{ii}\Delta\tau} + \frac{1}{A_{ii}h} + \frac{\partial \mathbf{R}}{\partial q}\right)\Delta \mathbf{Q}_i = -\frac{\mathbf{Q}_i^p - \mathbf{Q}^*}{A_{ii}h} - \mathbf{R}(\mathbf{Q}_i^p)$$

where

$$\mathbf{Q}^* = \mathbf{Q}^n + h\sum_{j=1}^{i-1} A_{ij} k_j$$

Psuedo-time continuation

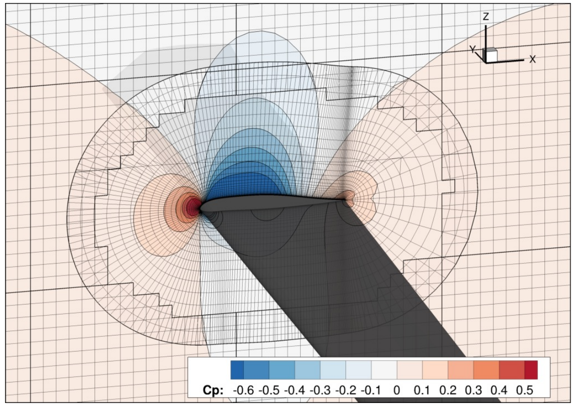
Jacobian

BDF-like term

Spatial residual

Linear combination of residuals from previous stages.

HELIOS FRAMEWORK: BASELINE APPROACH



- **Each module in Helios has a lightweight Python wrapper.**
- **In baseline operation, Helios delegates time-marching to each solver.**
  - Communication is performed once per solver, per timestep.

1: $T \leftarrow 0$
2: **for** $n = 1, ..., N_{\text{steps}}$ **do** .... Main time-stepping loop
3: $\quad T \leftarrow T + \Delta T$ .... Advance solution time
4: $\quad$ MOVE_GRIDS(T) .... Move grids to location at new time — Helios MELODI
5: $\quad$ DOMAIN_CONNECTIVITY() .... Hole-cut and reconnect all grids — Helios PUNDIT
6: $\quad$ **for all** $S \in$ CFD Solvers **do**
7: $\quad\quad$ OVERSET_INTERPOLATE($\mathbf{Q}$) .... Exchange solution at grid boundaries — Helios PUNDIT
8: $\quad\quad$ S.RUN_STEP(n) .... Each solver performs non-linear solve — Helios flow-solver (mStrand, Orchard)
9: $\quad$ **end for**
10: **end for**

- **Helios also has the option to use Overset – GMRES (OGMRES):**
  - Full overset linear system, including overset boundaries, is solved using GMRES at the framework.
  - This improves the formulation of lines 6-9.
  - Domain connectivity with O-GMRES is still once per timestep and limited to second-order accurate in time.

HELIOS FRAMEWORK: SDIRK APPROACH

- **For DIRK schemes, time-marching done at the Python (framework) level.**

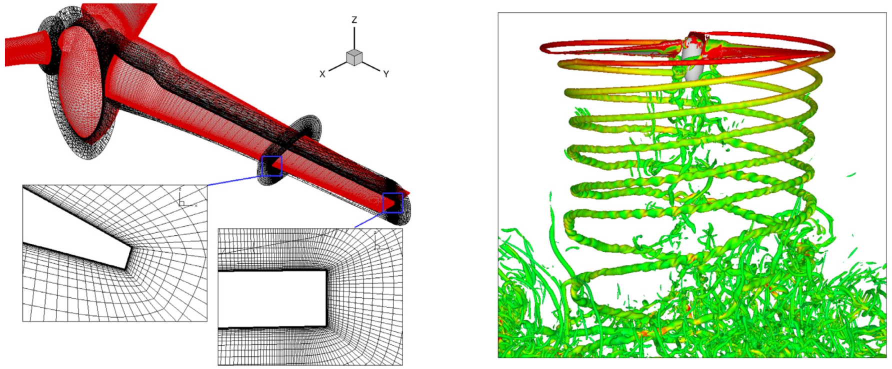
$T \leftarrow 0$	
for $n = 1, \dots, N_{\text{steps}}$ **do**	Main time-stepping loop
$\mathbf{Q}^n \leftarrow \mathbf{Q}$	
$\mathbf{R}_s \leftarrow [\emptyset, \dots, \emptyset]$	
for $i = 1, \dots, N_{\text{stage}}$ **do**	RK-stage loop
$T_i \leftarrow T + \mathbf{C}[i] \cdot \Delta T$	Advance time to value at stage-fraction
MOVE_GRIDS(T_i)	Move the grids to location at new time
DOMAIN_CONNECTIVITY()	Hole-cut and reconnect all grids
$\mathbf{Q}^* \leftarrow \mathbf{Q}^n$	
for $j = 1, \dots, i-1$ **do**	
$\mathbf{Q}^* \leftarrow \mathbf{Q}^* - A_{i,j} \cdot \Delta T \cdot \mathbf{R}[j]$	Build reference ***Q****
end for	
for $m = 1, \dots, N_{\text{sub-iter}}$ **do**	
OVERSET_INTERPOLATE($\mathbf{Q}$)	Exchange ***Q*** each non-linear iteration
for all $S \in$ CFD Solvers **do**	
$h \leftarrow A_{i,i} \cdot \Delta T$	Scale timestep by diagonal of ***A***
$\mathbf{R}, \mathbf{R}_s[i] \leftarrow S$.COMPUTE_RHS($\mathbf{Q}, \mathbf{Q}^*, h$)	Compute full residual and store spatial residual
end for	
$\Delta\mathbf{Q} \leftarrow$ DO_GMRES($\mathbf{R}$)	Perform linear-solve
$\mathbf{Q} \leftarrow \mathbf{Q} + \Delta\mathbf{Q}$	
end for	
end for	
$T \leftarrow T + \Delta T$	
end for	

New interface function (lines for T_i and MOVE_GRIDS)

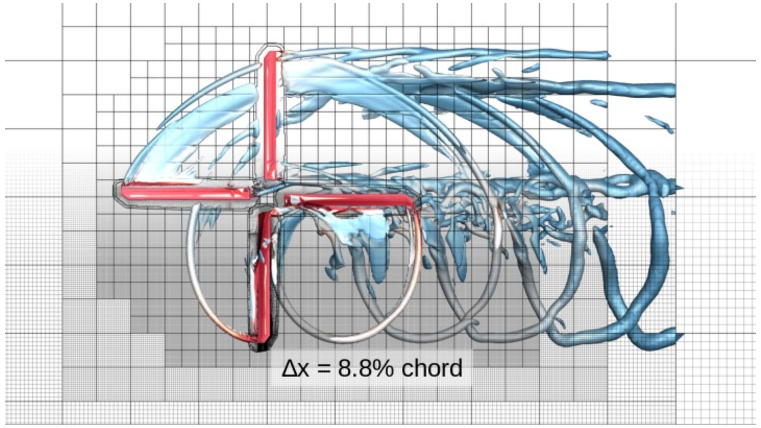
New interface function (lines for h and COMPUTE_RHS)

HELIOS CFD SOLVERS

- **Helios is used with one near-body solver, mStrand, and one off-body solver, Orchard**
  - Inviscid, laminar, or RANS/DES (SA) forms of the governing equations
  - In-house developed codes
- **mStrand:**
  - Unstructured in the surface direction, structured in the strand (surface-normal) direction
  - Developed as part of Helios since 2015
  - Widely tested and validated for both rotorcraft fuselage and blade meshes
- **Orchard**
  - Octree-based Cartesian Adaptive Mesh Refinement (AMR) for background grids
  - Newest solver in Helios, released in 2021
  - Cartesian solver numerics adapted from SAMCart but octree-AMR results in significant overall speedups
  - Compatible with both CPU and GPU architectures, though only CPUs used in this work



mStrand grids and solution for TRAM rotor. *(Lakshminarayan, 2017)*



Orchard refinement and solution for PSP rotor. *(Jude, 2021)*

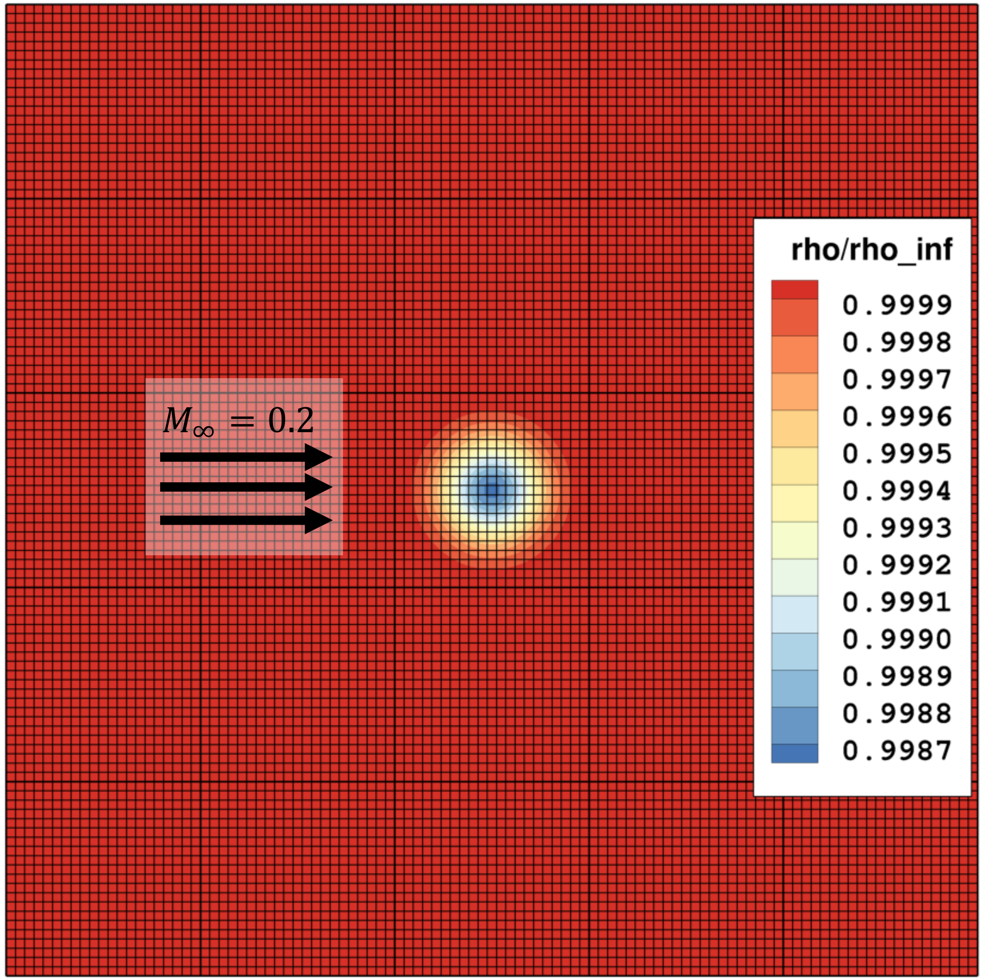
Results

1. Vortex convection on Cartesian grids
2. Wing-vortex interaction
3. Pitching wing

Note: All results use non-dimensional time based on speed of sound and grid unit:

$$t = \frac{t_{\text{phys}} * a_\infty}{D}$$

VORTEX CONVECTION



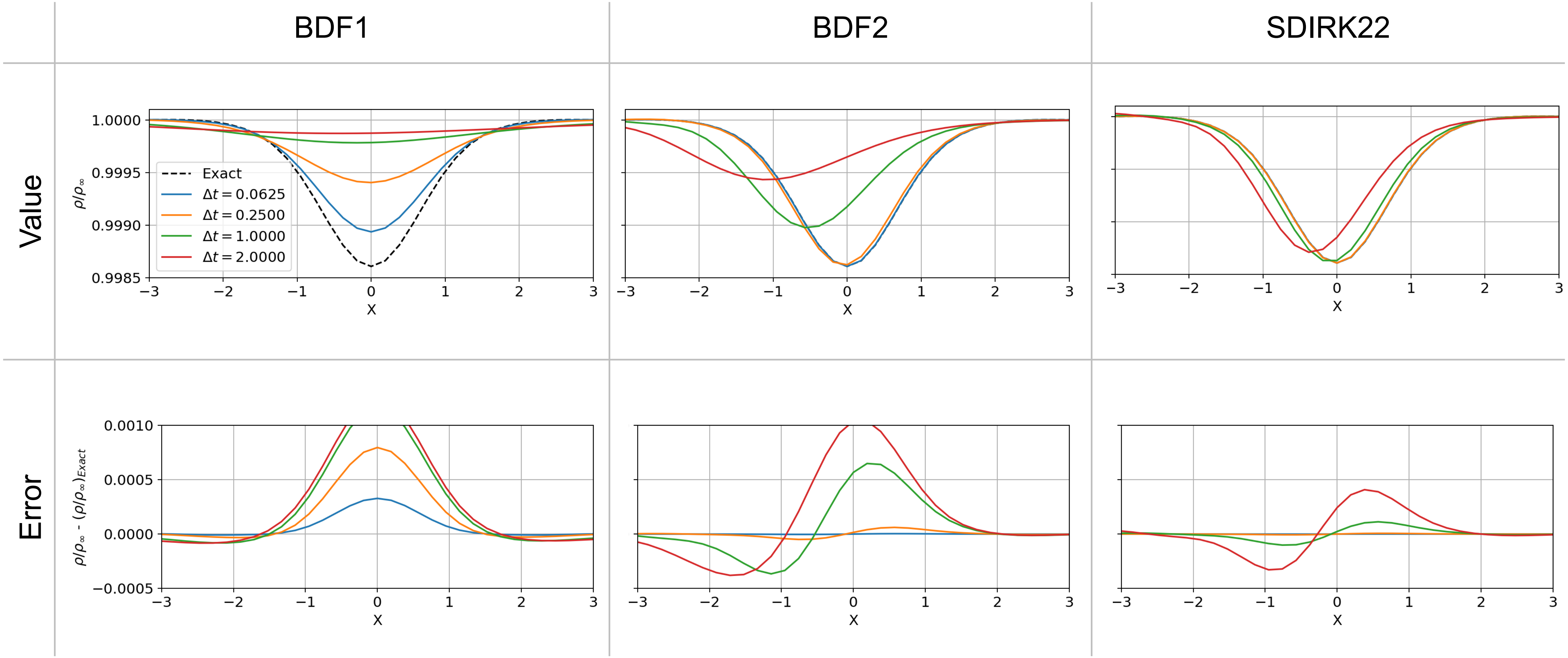
Initial density solution for the vortex.

- **Isentropic vortex convection (no viscosity)**
- **All boundaries are periodic**
- **Vortex travels 20 units (1 period) in $T = 100$**

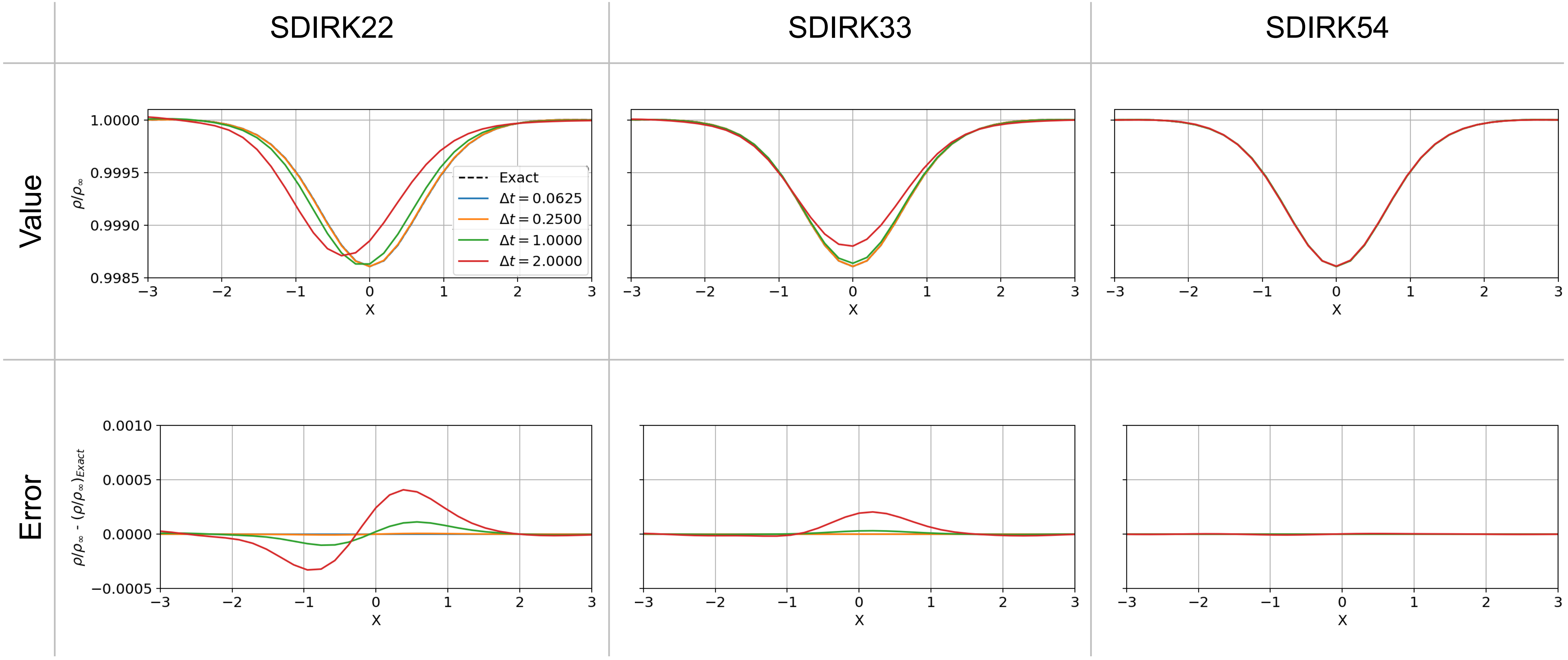
- **Simulation parameters:**
  - Each non-linear iteration is converged over 10 order of magnitude
  - Run using BDF1, BDF2, SDIRK22, SDIRK33, SDIRK54 using various timesteps

- **Comparison criteria:**
  - The density along the centerline of the domain is compared against an “exact” value
  - The “exact” value is obtained by running SDIRK54 at timestep of $\Delta t = 0.03125$
  - Double precision solutions are used to extract density

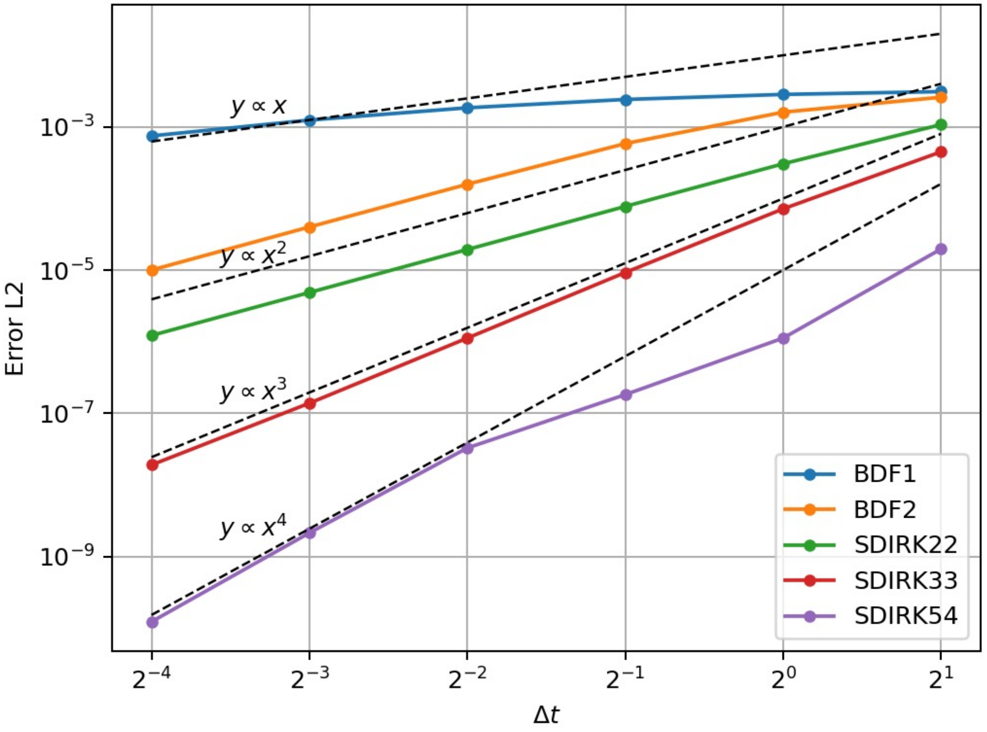
VORTEX CONVECTION



VORTEX CONVECTION

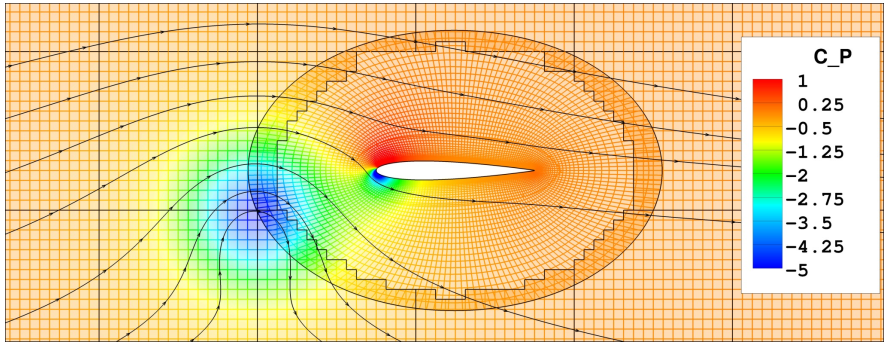


VORTEX CONVECTION

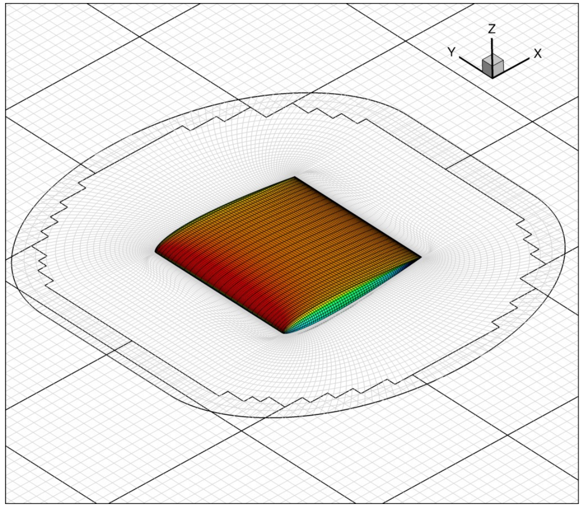


- **Convergence slopes match expected scheme order**
- **As expected:**
  - even-order schemes → dispersion error
  - odd-order schemes → diffusion error
- **Though both methods are 2nd order accurate, SDIRK22 is more accurate than BDF2**
- **The correct error convergence plot could only be if both double precision solutions were used with 10-order drop in residual**

WING-VORTEX INTERACTION



Center-plane pressure at *T*=30.



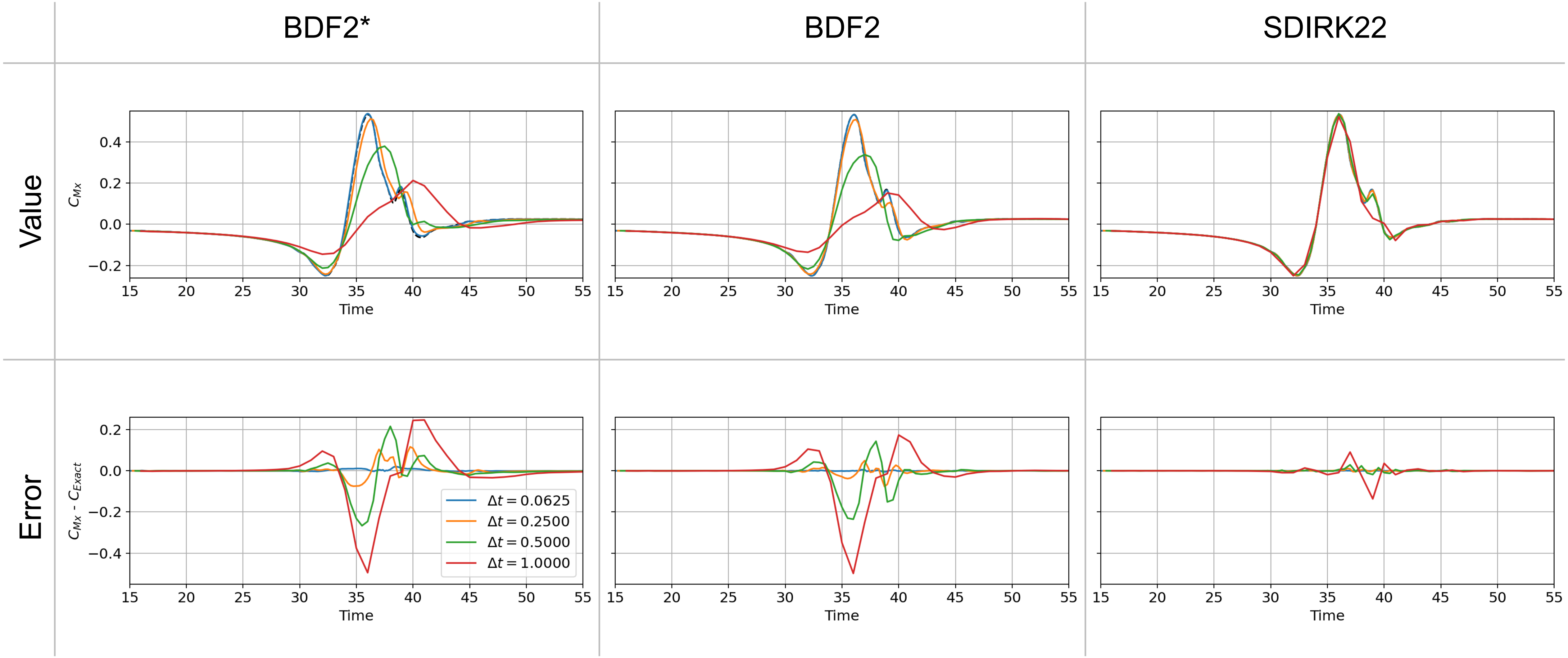
Isometric view of 3D wing with surface pressure solution.

- **A vortex in the Cartesian domain is convected into a NACA0012 wing mesh**
  - Inviscid simulation focuses on temporal accuracy for convection between meshes
  - Vortex is initialized 7-chords up-stream of and ¼-chord below wing
  - Vortex convects 1-chord in non-dimensional time $T = 5$, restarted from $T = 15$

- **Analogy to rotor blade-vortex interaction:**
  - For a rotor with an aspect-ratio of 16, $\Delta\psi = 0.25°$, the blade travels $\frac{\pi R}{720} \approx 0.07c$ per timestep
  - If $M_{tip} = 0.6$, the non-dimensional $\Delta t \approx 0.12$

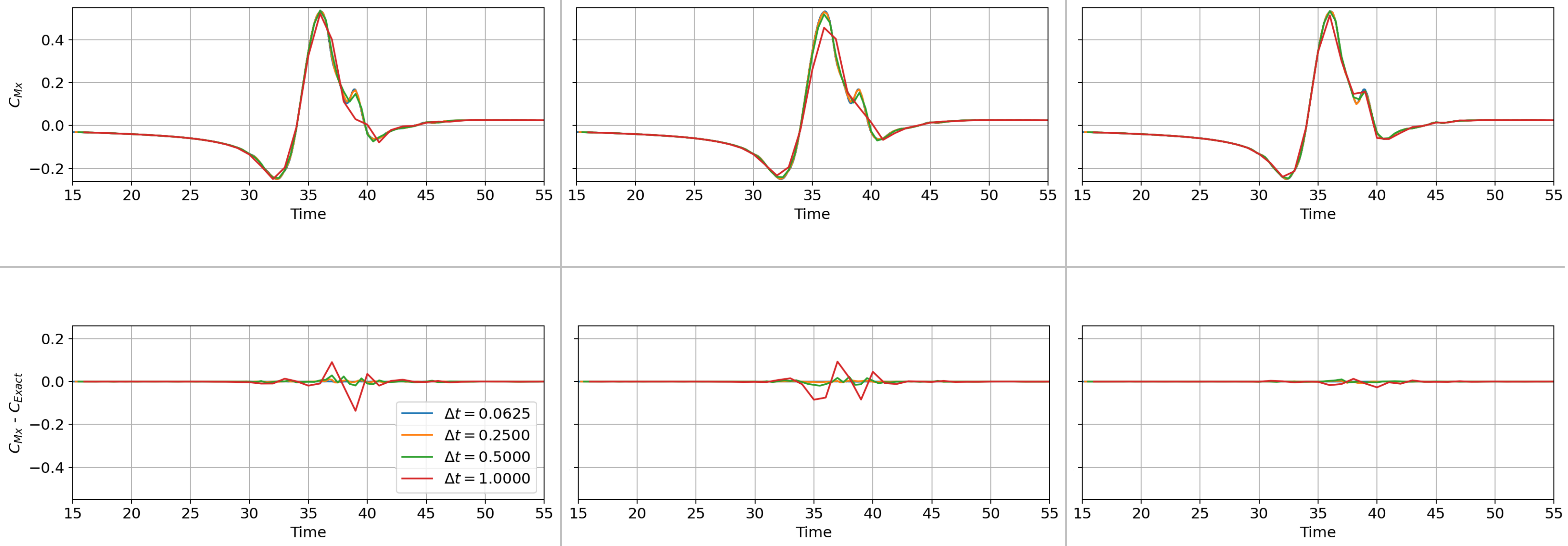
- **Methods used:**
  - BDF1, BDF2, BDF2*, SDIRK22, SDIRK33, SDIRK54
  - BDF2* is an approximation to BDF2 where each solver independently takes a timestep (baseline Helios operation)

WING VORTEX INTERACTION

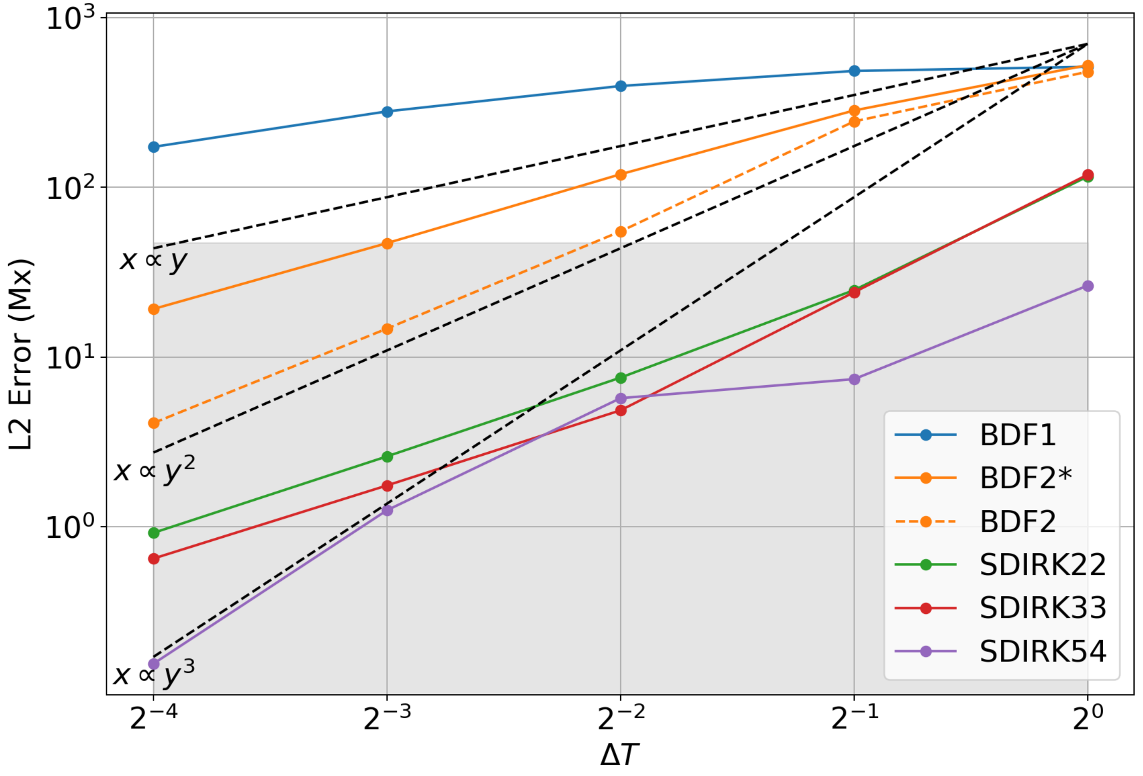


WING VORTEX INTERACTION

	SDIRK22	SDIRK33	SDIRK54
Value			
Error			

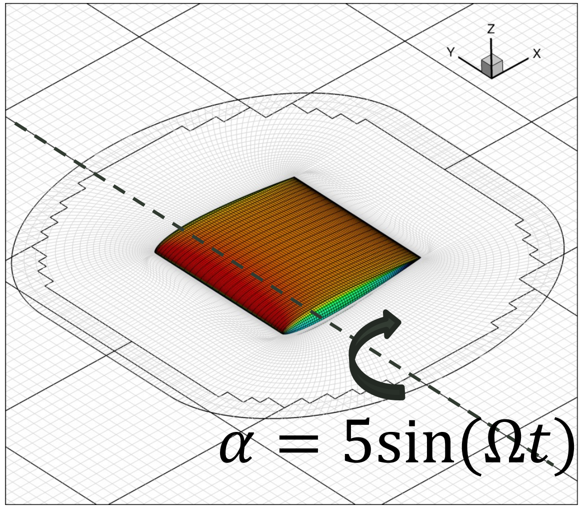


WING VORTEX INTERACTION



- **Temporal convergence:**
  - Slopes match order only for formal BDF2 scheme
  - SDIRK schemes are more accurate but the slope stalls at smaller timesteps
  - Possible reason: convergence, floating-precision of compared moments
- **From the rotor analogy, $\Delta\psi = 0.25 \rightarrow \Delta t \approx 0.12$:**
  - Everything in the shaded region is more accurate than a baseline Helios simulation (BDF2*)
  - Using SDIRK22 or SDIRK33 a $4\times$ increase in timestep is possible without loss of accuracy
  - Using SDIRK54 a $8\times$ increase in timestep is possible
- **Use of O-GMRES:**
  - Without O-GMRES convergence always stalled at $T \approx 30$ when the vortex enters the near-body mesh
  - Using O-GMRES, the overall residual drops >6 orders almost every iteration

PITCHING WING

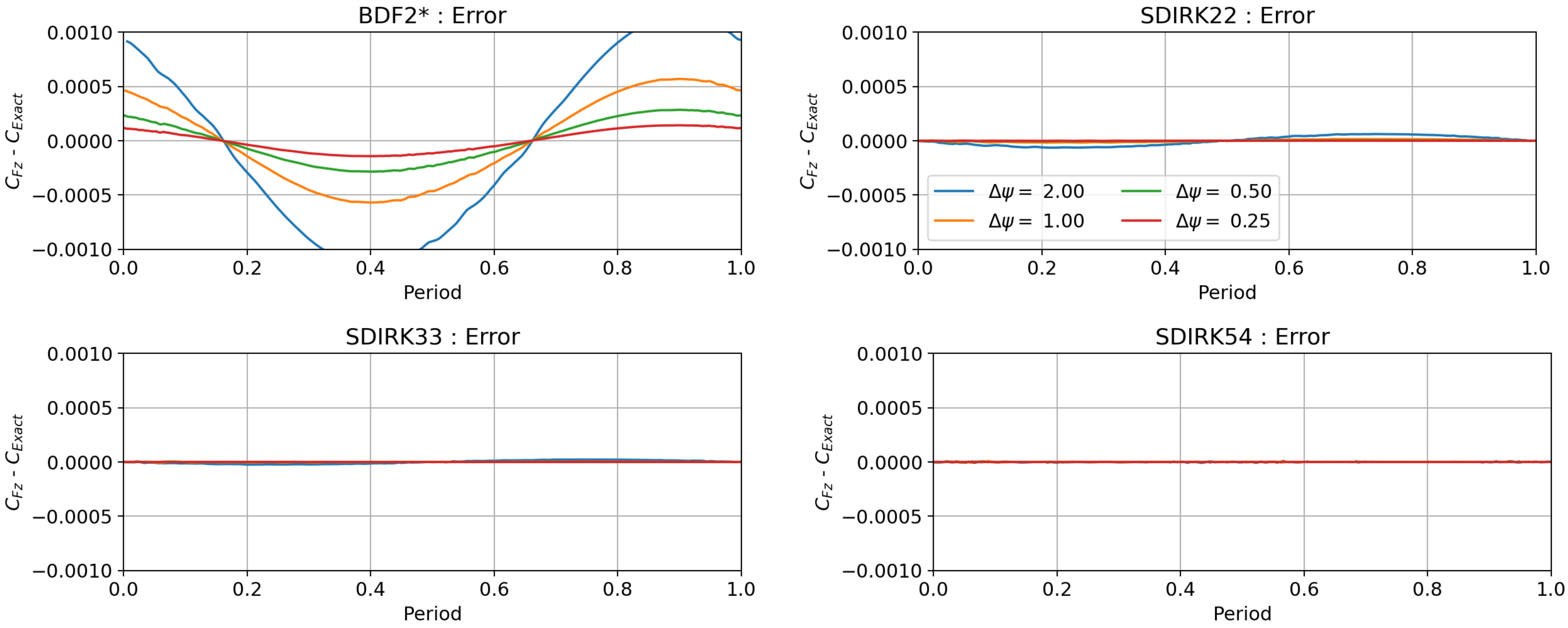


The same wing from wing-vortex case but now with oscillatory pitching.

- **The same NACA0012 wing from the previous case:**
  - Still inviscid but no vortex initialized in Orchard mesh
  - Wing pitches $\pm 5°$ with reduced frequency $k = \frac{\Omega c}{2U_\infty} = 0.2$
  - Using $\psi = \Omega t$, timesteps are chosen as $\Delta\psi = [0.25°, 0.5°, 1.0°, 2.0°]$
- **Rotor analogy:**
  - Though a rotor blade will undergo pitch, flap, and lag deflections at multiple frequencies, a single $1/rev$ forcing frequency is input via pilot controls
  - Trivial analogy between rotor azimuth and ψ for this case
- **Grid motion:**
  - Grids are moved and reconnected each stage of each RK step
  - Analytic grid speeds are used in mStrand

PITCHING WING

- **Lift is compared between all timestepping methods**
  - Lift is a sinusoid based on forcing frequency, only the error is of interest



Conclusions and Future Work

CONCLUSIONS AND FUTURE WORK

- **Three canonical problems representative of challenges in rotorcraft CFD presented:**
  - Vortex convection, wing/blade-vortex interaction, and pitching wing aerodynamics
  - SDIRK methods more accurate than BDF2* (baseline Helios)
  - Formal accuracy not achieved for SDIRK33 and SDIRK54 for overset cases
  - Generally higher-order SDIRK schemes reduce error
  - SDIRK methods enable use of $4\times$ or $8\times$ the baseline timestep with improved accuracy

- **Future simulations with deforming meshes:**
  - Yang and Mavriplis proposed method for high-order, GCL grid speeds
  - Preliminary implementation in mStrand being validated

For SDIRK33:

$$\begin{pmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{Bmatrix} \dot{\mathbf{X}}^1 \\ \dot{\mathbf{X}}^2 \\ \dot{\mathbf{X}}^3 \end{Bmatrix} = \frac{1}{\Delta t} \begin{Bmatrix} V_1 - V_n \\ V_2 - V_n \\ V_3 - V_n \end{Bmatrix}$$

face velocity at each stage

volume “swept” by face

RK coefficients

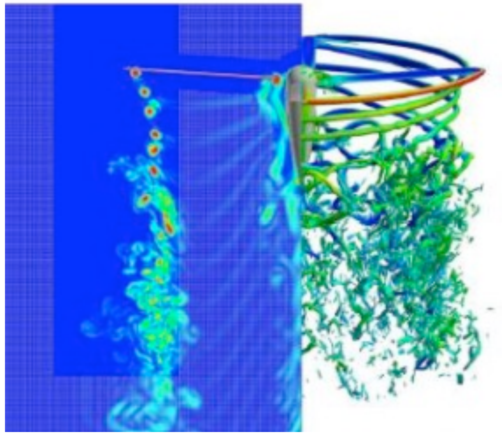
- **Future temporal adaption:**
  - At timesteps/azimuths of interest, reduce timestep or increase scheme order
  - Mirror high-order approaches currently used in spatial terms

FUTURE WORK: WAKE BREAKDOWN STUDY

- **Non-physical wake breakdown of a hovering rotor is often observed in CFD simulations**

- **Further study involving high-order temporal schemes would answer:**
  - Is dispersion-dominated BDF2* is the primary cause?
  - Is a diffusion-dominated scheme (like SDIRK33) better for resolving rotor wakes?
  - How do the wake structures differ using varying orders of accuracy?

- **Accurately resolving wakes is key for interactional aero analysis:**
  - Multi-rotor vehicles have many vortex-rotor and vortex-body interactions
  - Acoustics and aircraft vibration require accurate, well-resolved flow fields



Narducci, R., Jain, R., Abras, J., Hariharan, N., SciTech 2021

ACKNOWLEDGEMENTS

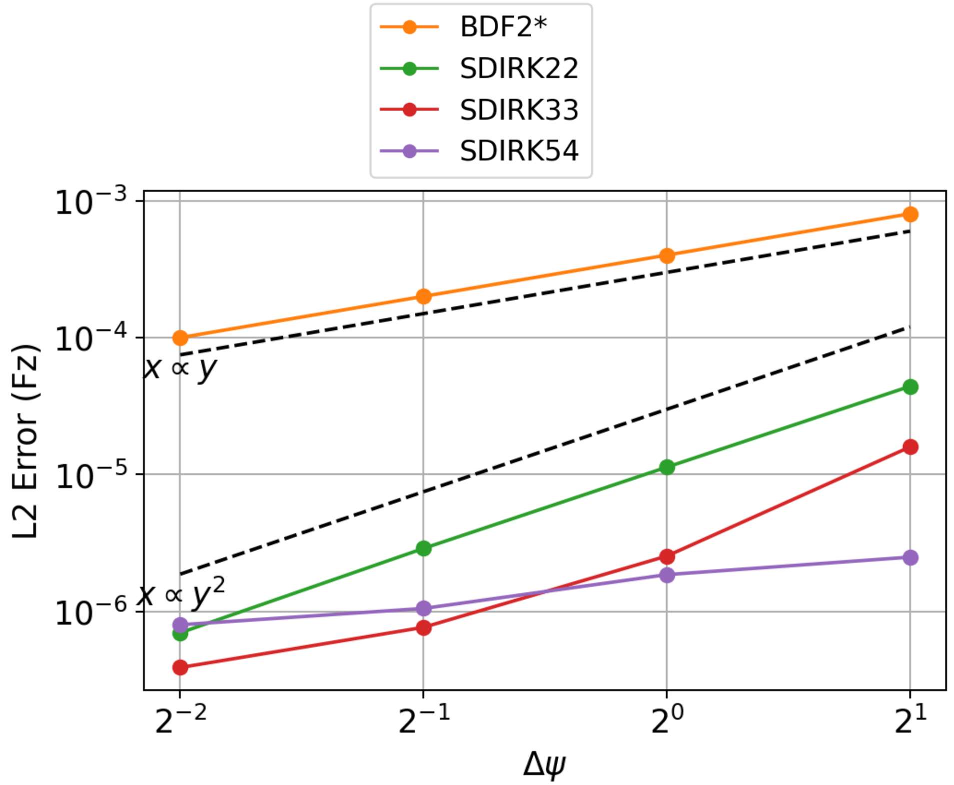
Many of the mStrand-related improvements for this work were developed and debugged with help of Vinod Lakshminarayan.

Presented materials are products of the CREATE-AV Element of the Computational Research and Engineering for Acquisition Tools and Environments (CREATE) Program sponsored by the U.S. Department of Defense HPC Modernization Program Office.

Funding and support for the presented research is also provided by the U.S. Army Combat Capabilities Development Command, Aviation & Missile Center.

Backup Slides

PITCHING WING



- **BDF2* (baseline Helios with separate time-marching):**
  - Converges at a rate of order 1, not 2.
  - Much less accurate than 2nd -order SDIRK22 scheme. This was seen as huge dispersion error in the previous slide.
- **SDIRK22:**
  - Matches expected 2nd -order slope and comparable to higher-order SDIRK methods at small timesteps.
- **SDIRK33 and SDIRK54:**
  - Do not match expected slopes. SDIRK33 has slope between 1 and 2; SDIRK54 has slope $\approx$1.
  - SDIRK33 performs even better that SDIRK54 for some timesteps, which is puzzling.
- **Analysis:**
  - Forces are recorded using single-precision so there may not be enough digits for comparison of methods with low-error.
  - Use of high-order methods allows $8\times$ larger timesteps with all SDIRK methods compared to BDF2*.