SynthEddy

System Verification and Validation Plan

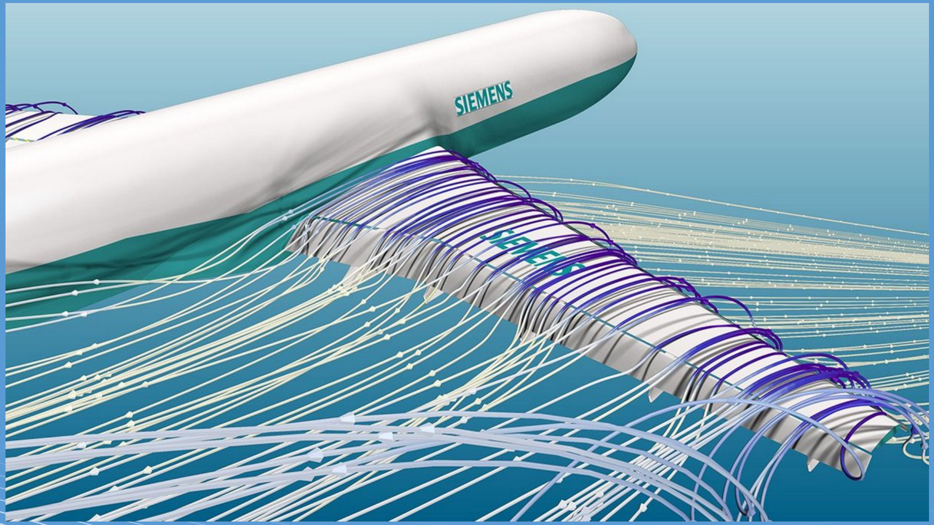
Simulating Turbulent Flow with Synthetic Eddy

Phil Du

McMaster University

What is CFD

- Using computer to simulate fluid (air, water, etc) flow and its interactions with other objects.



Credit: ASCENDTECH GROUP

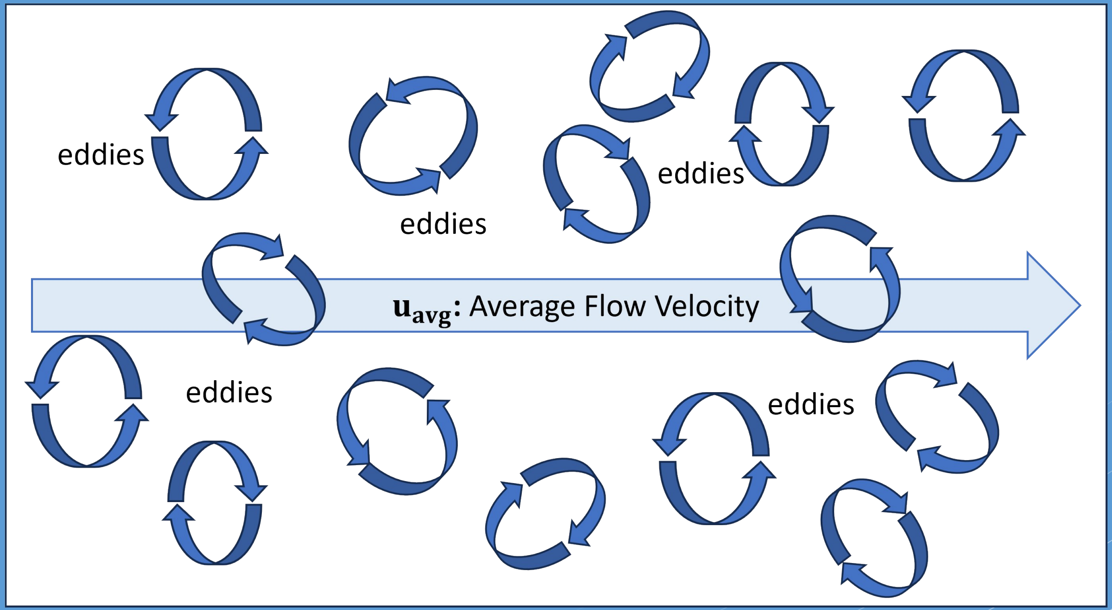
Credit: SolidWorks

Approximate Turbulent Flow

Credit: https://commons.wikimedia.org/wiki/User:Tangopaso

Flow Field (actual one is 3D)

Inlet



Outlet

McMaster University

Main Challenge

- (Almost) no pseudo-oracle.
  – Domain expert's MATLAB code
  – Likely more useful in proof of concept stage
- Only theories on how a correct output should be like.

SRS Verification Plan - Functional

- Automated Testing:
  - R1: Verify that the queried point is within the flow field.
  - R2: Generate a velocity field given any valid eddy profiles input.
  - R3: (Future) Provide a realistic eddy profile before generation.
  - R4: Flow field is divergence-free
- Manual Testing:
  - R5: Allow CFD to interface with this software to obtain IC and BC.

SRS Verification Plan – Nonfunctional

- Manual Testing
  - NFR1: Accuracy: Suitable as CFD IC/BC
    - Need to actually run CFD (Suggestions?)
  - NFR2: Usability: Easy to setup CFD integration
  - NFR3: Maintainability: Modular and well-documented and modifiable.
  - NFR4: Portability: Run on any x86-64 based Windows or Linux computer.

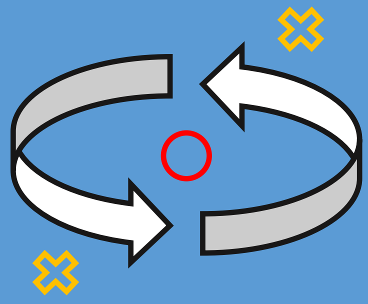
Test Case: Query Outside Flow Region

- Control: Automatic
- Initial State: Any
- Input: x, y, z that is outside field(Lx, Ly, Lz)
- Output: (0, 0, 0), with warning

- How to perform: test script

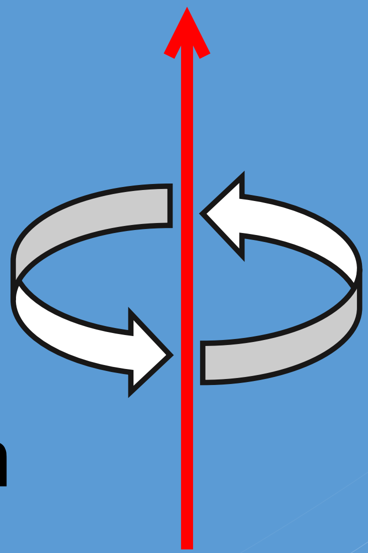
Test Case: Eddy – Shape

- Control: Automatic
- Initial State: Empty Field
- Input:
  - Profile: 1 Eddy with predetermined center location
  - Query: Opposite points pairs relative to center
- Output:
  - Inside eddy: equal and opposite velocities, magnitude follows shape function
  - Outside eddy: both (0,0,0)



McMaster University

Test Case: Eddy – Velocity std

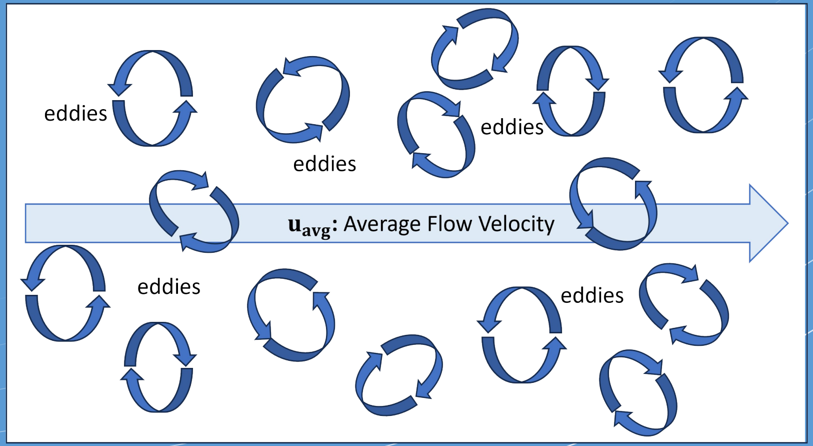


- Control: Automatic
- Initial State: Empty Field
- Input:
  - Profile: 1 Eddy with predetermined orientation
  - Query: Mesh grid of points
- Output: Velocities at each point
  - Should have zero velocity std along the eddy axis

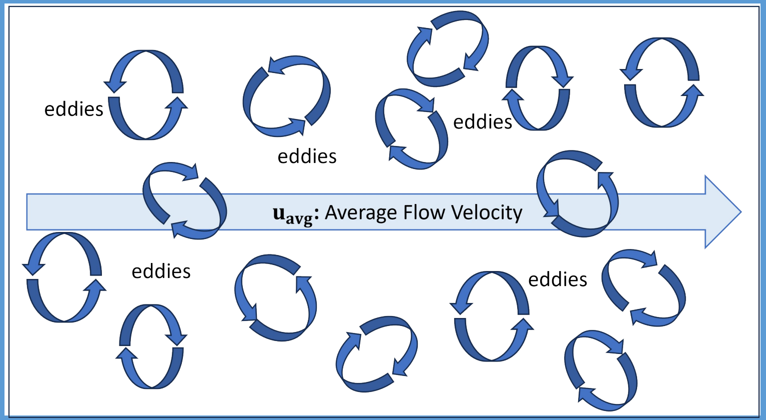
McMaster University

Test Case: Field – Zero Sum

- Control: Automatic
- Initial State: Empty Field
- Input:
  - Profile: Any number of eddies, random orientations
  - Query: Mesh grid of points, any t
- Output: Velocities at each point
  - Vector sum of all output should be $\mathbf{u}_{\mathbf{avg}}$

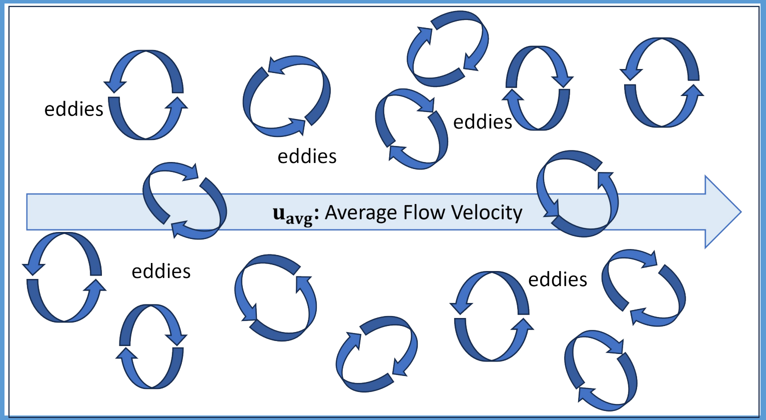


Test Case: Field – Velocity std vs t



- Control: Automatic
- Initial State: Empty Field
- Input:
  - Profile: Any number of eddies, random orientations
  - Query: Mesh grid of points, increasing t
- Output: Velocities at each point
  - Velocity std should stay the same (some tolerance)

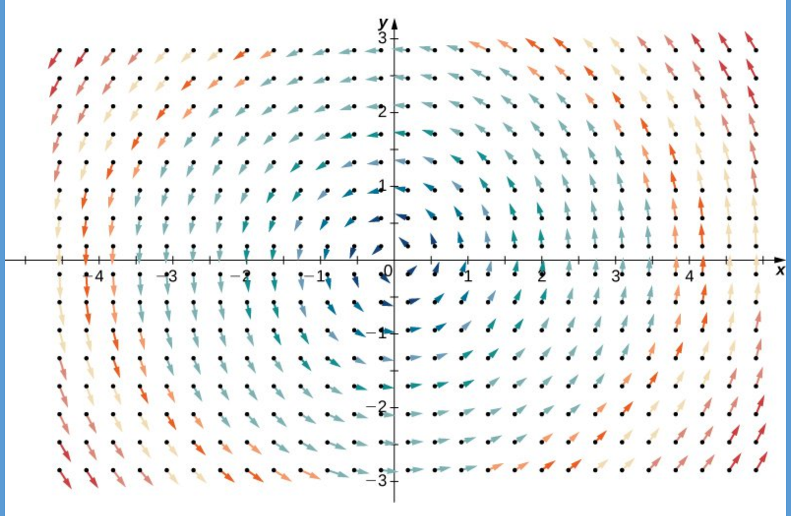
Test Case: Field – Velocity std vs α



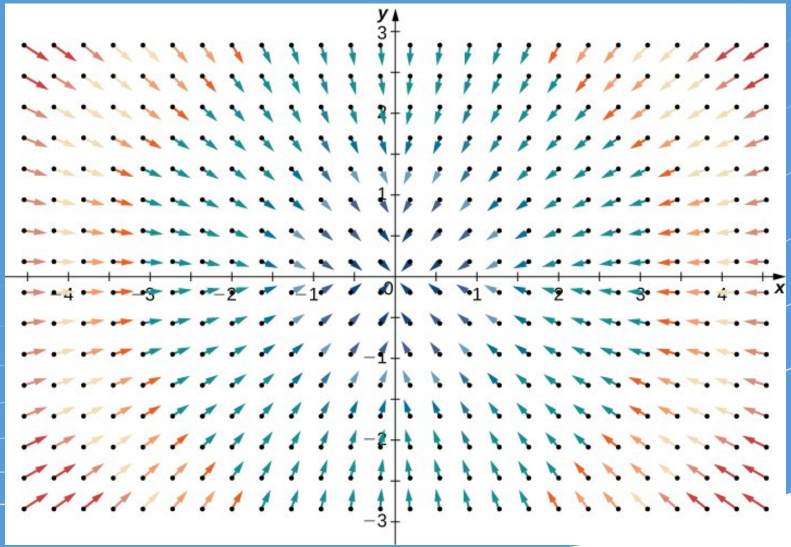
- Control: Automatic
- Initial State: Empty Field
- Input:
  - Profile: Any number of eddies, random orientations
  - Query: Mesh grid of points, increasing α: eddy intensity
- Output: Velocities at each point
  - Velocity std should increase with α (trend TBD)

Test Case: Divergence-Free

- Control: Automatic
- Initial State: Empty Field
- Input:
  - Profile: Any
  - Query: Mesh grid of points
- Output: Velocities at each point
  - Div(field) < some tolerance



(b)



McMaster University

Credit: https://openstax.org/books/calculus-volume-3/pages/6-5-divergence-and-curl

Test Case: NFR – Maintainability

- Type: Static + Dynamic
- How test will be performed:
  - Provide an alternative shape function (mathematical)
  - Ask the reviewer to modify the code to implement it
    - With no help other than documentation that comes with the software
  - Survey how easy it is
  - Check if the code still passes all the automatic tests.

Credit: https://openstax.org/books/calculus-volume-3/pages/6-5-divergence-and-curl

Automated Tools

- Profiling Tool: likely no need
- Continuous Integration
  - Test scripts
  - GitHub Actions
- Linter: VS Code Extension
- Code Coverage: Coverage.py

Really, I’m here for suggestions...

- Anything goes, thanks!

McMaster University