IMPLEMENTATION OF PRESSURE BASED SOLVER FOR SU2

3rd SU2 Developers Meet | Akshay K.R, Huseyin Ozdemir, Edwin van der Weide
Content

› ECN part of TNO
› SU2 applications at ECN
› Incompressible flow solver
› Pressure-based solver
› Research plan
› SU2 in past and current projects
ECN part of TNO

- **ECN** - Energy Research Center of the Netherlands merged with **TNO** as of April 2018.
- Research on many topics of **renewable energy technologies**
  - Wind energy, Solar energy, Biomass, etc.
- Wind energy unit mainly focused on **low-fidelity tools** that are tailor made for wind energy applications (about 50 researchers on various topics)
- **SU2** is our first serious attempt to include **CFD** in our research/design tool chain
SU2 Applications at ECN

- Airfoil analysis:
  - Thick airfoils (30%-40% thickness)
  - Blade sections with add-ons like VGs

- Wind turbine rotor simulations
  - Rotating and periodic simulations
  - Load computations

- Wind farm simulations (planned)
  - Actuator disk models
Incompressible Flow Solver

- Wind energy applications have typical $\text{Ma} \sim 0$ and $\text{Re} \sim 10^6$.

- Artificial compressibility method not suitable in this regime – very high mesh requirements and accuracy issues.

- Need an accurate incompressible solver for wind energy applications.

- Segregated pressure-based solver – SIMPLE family of algorithms.
Pressure Based Solver: Governing Equations

Momentum Equation:

$$\partial_t U + \nabla \cdot \left( \overrightarrow{F^c} - \overrightarrow{F^v} \right) - Q = -\nabla P$$

Pressure correction equation:

$$\nabla \cdot \nabla kP' = \sum_f m^* + \text{RHS}$$

**SIMPLE**

1. Start with $u^n, v^n, ...$ as the initial guess
2. Solve momentum equations to get $u^*, v^*, w^*$
3. Update mass flux, $m_f^*$ and other terms in the RHS
4. Solve pressure correction equation
5. Update pressure and velocities to get $u^{**}, v^{**}, w^{**}$
6. Repeat till convergence
Research Plan

- **Research code**
  - Develop *research code* to test and analyze different features of the code
  - Implement an *artificial compressibility* version and a *pressure-based* version

- **SU2**
  - A finite volume discretization for the *Poisson problem* in SU2
  - *Euler solver* with the convective terms and *N-S solver* with the viscous terms following the code structure of SU2
Results: Research Code

Lid driven cavity problem

- Re = 400
- Vertex based FVM
- Uniform, cartesian grid
- Spatial discretization:
  - 1st order upwind
  - Central
- Time discretization:
  - Implicit Euler
  - Explicit Euler
- Linear solver:
  - Tri-Diagonal Matrix Algorithm (with ADI)
  - Gauss-Seidel
  - Plan to add MG
Results: Research Code

Lid driven cavity problem

Comparison of velocity profile for Lid driven cavity, Re = 400

Centreline velocity
Results: SU2

Poisson Solver FVM

- Implement a FV discretization of Poisson equation.
- Solve an analytical test case to check solution.
- Needs to solved multiple times within every iteration.
- Enable multigrid to obtain a fast solution from the Poisson solver.
- Algebraic multigrid might be necessary for unsteady problems to obtain acceptable computational times.
Current Status of Research

SU2 Euler/NS solver

- Implementing the solver, numerical, variable and other associated routines for a pressure based system.
- Coupled the flow solver with the Poisson solver.
- Experimenting with different spatial discretization methods for the pressure-based solver.
  - Plan to implement first and second order upwind and central schemes initially.
- Testing in the research code
- Extend the multigrid to not only Poisson but also to flow solver.
- Works only on single node so far.
- Add ALE, periodic options.
SU2 in Current Projects at ECN

Airfoil with add-ons
Aero polar for flap angle = -5deg

Airfoil with add-ons
Load Computation

Power plane – Novel concepts for wind power generation.

An alternative to traditional wind turbine designs.
Wind farm modelling

- Use actuator disk to approximate the turbine.
- Compute power output and optimize wind farm layouts.
- Currently use parabolized N-S solver, plan to use SU2.

Wake behind a turbine

Wind farm layout
THANK YOU FOR YOUR ATTENTION

TNO.NL/ECNPARTOFTNO
A typical case consists of
- 10x10 = 100 wind turbines
- 25 wind speed levels
- 72 wind directions
- 1 turbulence intensity level

In total
- (10x9)x25x72x1 = 162000 cases to simulate
- 162000 x 3 seconds / 3600

~ 135 hours
FarmFlow simulation time for a single case

- Simulation time (CPU time) for a single:
  - Wind turbine (single wake)
  - Wind speed
  - Wind direction
  - Turbulence intensity level

- On a:
  - Intel® Xeon® CPU E5620 @2.40 GHz
  - 8 GB ram
  - Windows 64 bit operating system

- ~ 3.0 seconds
ABL stability model

Stable conditions:

\[ u = \frac{u_0}{\kappa} \left[ \ln \left( \frac{z}{z_0} \right) - \psi_m \left( \frac{z}{L} \right) \left( 1 - \frac{z}{2z_i} \right) + \frac{z}{L_{MBL}} - \frac{z}{z_i} \left( \frac{z}{2L_{MBL}} \right) \right] \]

Unstable conditions:

\[ u = \frac{u_0}{\kappa} \left[ \ln \left( \frac{z}{z_0} \right) - \psi_m \left( \frac{z}{L} \right) + \frac{z}{L_{MBL}} - \frac{z}{2z_i} \left( \frac{z}{2L_{MBL}} \right) \right] \]

Neutral conditions:

\[ u = \frac{u_0}{\kappa} \left[ \ln \left( \frac{z}{z_0} \right) + \frac{z}{L_{MBL}} - \frac{z}{z_i} \left( \frac{z}{2L_{MBL}} \right) \right] \]

Empirical fit for scaling parameter, \( L_{MBL} \):

\[ \frac{u_0}{fL_{MBL}} = \left( -2 \ln \left( \frac{u_0}{fz_0} \right) + 55 \right) e^{\left( \frac{(u_0/fL_{MBL})^2}{400} \right)} \]