The SIMPLE Algorithm for Pressure-Velocity Coupling

ME 448/548 Notes

Gerald Recktenwald
Portland State University
Department of Mechanical Engineering
gerry@pdx.edu

February 10, 2014

Overview

CFD modelling involves these steps

1. Define the geometry of the fluid volume
2. Locate boundary regions and assign values to boundary condition parameters
3. Specify fluid physics (e.g., buoyancy, turbulence, radiation) and thermophysical properties
4. Create the mesh
5. Define parameters that control the solution
6. Solve the flow model
7. View the results

Commercial CFD packages have preprocessors (graphical user interfaces) for setting up the model and viewing the results. The order of preprocessing steps may be different for different CFD packages.

Users control step 6 by specifying parameters the affect how the solver updates the velocity, pressure and other scalar fields. That control is indirect, at best.

Basic CFD Solver

We'll survey the features of a basic finite-volume CFD code in the following order.

- Staggered Mesh
- Basic Discretization
- Velocity and Pressure Corrections
- Algorithm Summary
- Convergence

The staggered mesh is not absolutely required, but it is the simplest strategy for incompressible flow calculations.

Many codes use non-staggered or unstructured meshes. We only use a staggered mesh here to explain pressure-velocity coupling.

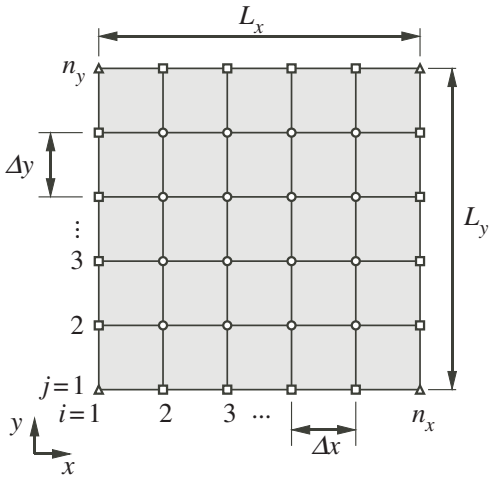
Reference: Ferziger and Perić, §,7.5.1, pp. 188–196

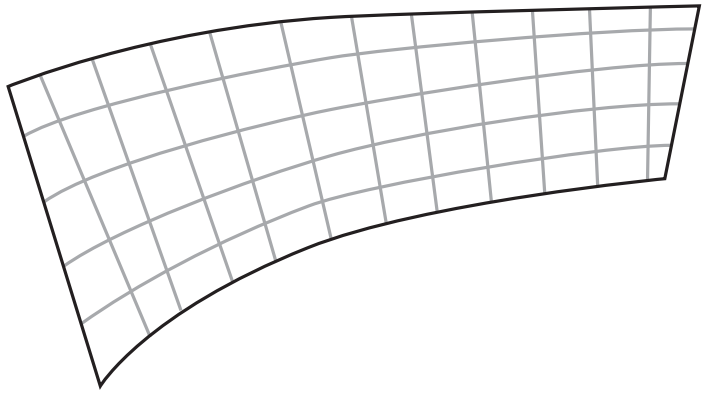
Tu, Yeoh, and Liu, §4.3.3

Structured and Unstructured Meshes

The SIMPLE algorithm can be developed for non-staggered, unstructured meshes. We will use the case of a staggered, structured mesh because the nomenclature is simpler. It is also the historical origin of the method.

Structured Meshes: Cells (nodes) are arranged in rows and columns

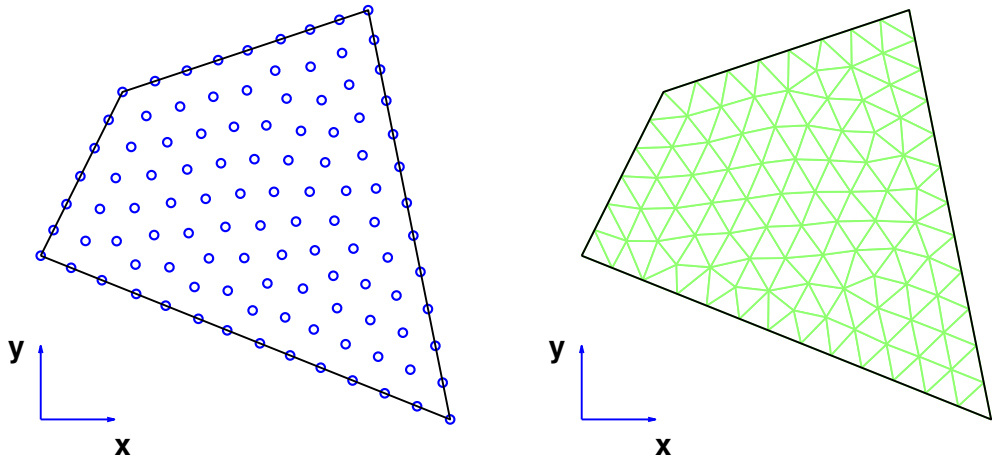




Each cell has the same number of neighbors. The connectivity is regular and uniform.

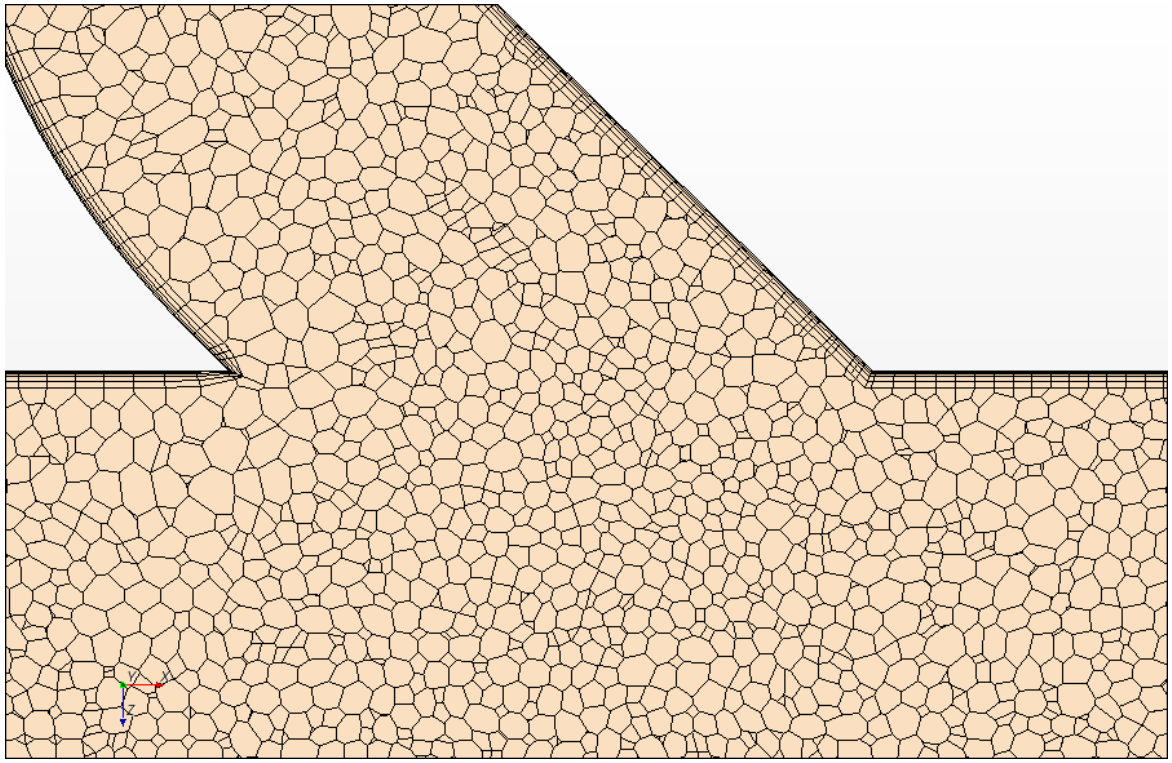
Structured and Unstructured Meshes

Unstructured Meshes: Cells and nodes are not arranged in rows and columns.



Structured and Unstructured Meshes

Unstructured Meshes: Cells and nodes can have arbitrary connectivity.

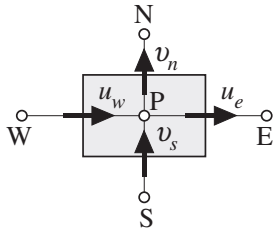


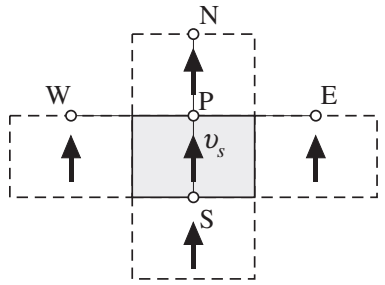
Staggered Grid

Main characteristics

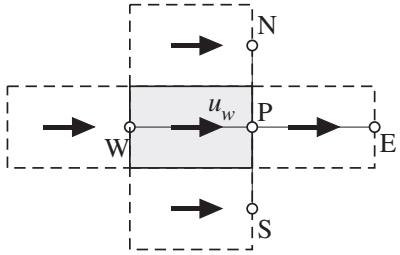
- Velocities are computed at cell faces
- Pressure is computed at cell centers

Main control volume





Control volume and neighbors of v_s



Control volume and neighbors of u_w

Basic Discretization – 2D (1)

For a two-dimensional, incompressible flow, applying the finite volume method on a staggered mesh leads to this set of equations for each interior control volume

$$-a_S^u u_{i,j-1} - a_W^u u_{i-1,j} + a_P^u u_{i,j} - a_E^u u_{i+1,j} - a_N^u u_{i,j+1} = \frac{p_{i-1,j} - p_{i,j}}{\delta x_w} + b_o^{(u)}$$

$$-a_S^v v_{i,j-1} - a_W^v v_{i-1,j} + a_P^v v_{i,j} - a_E^v v_{i+1,j} - a_N^v v_{i,j+1} = \frac{p_{i,j-1} - p_{i,j}}{\delta y_s} + b_o^{(v)}$$

$$\frac{u_{i+1,j} - u_{i,j}}{\Delta x} + \frac{v_{i,j+1} - v_{i,j}}{\Delta y} = 0$$

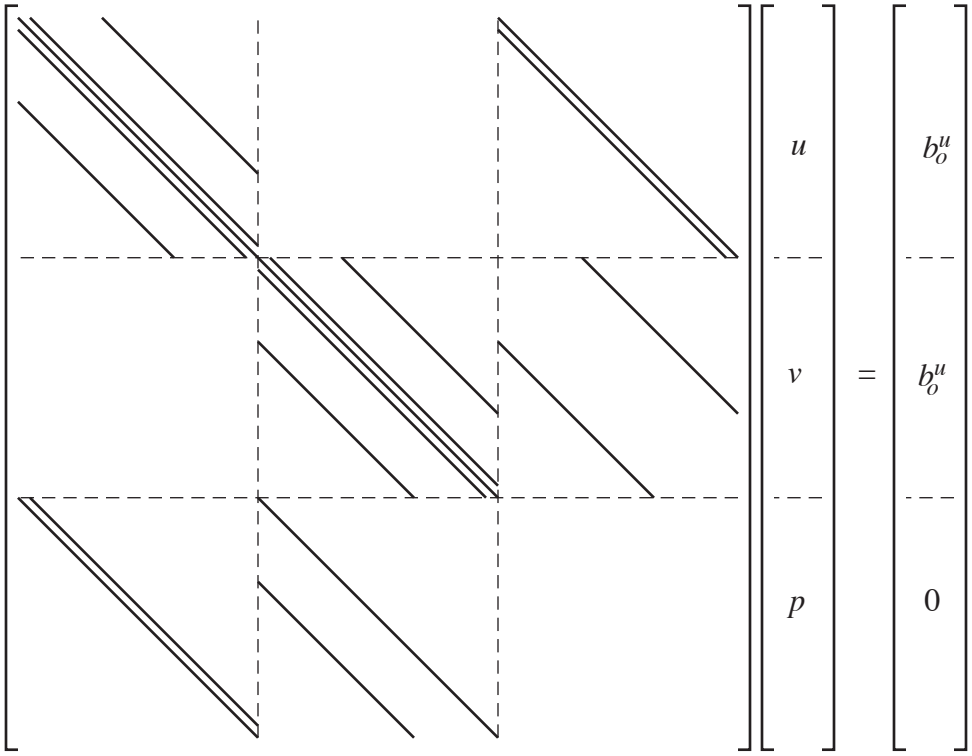
where $b_o^{(u)}$ and $b_o^{(v)}$ are source terms due to non-uniform viscosity, gravity, and other effects.

These equations are nonlinear. For example, if the central difference scheme is used,

$$a_E^u = \mathcal{F}\left(u_{i,j}, u_{i+1,j}, \nu, \text{geometry}\right)$$

Basic Discretization – 2D (2)

The coupled system of equations looks like this for a staggered, structured, 2D mesh.



Lines in the coefficient matrix represent the position of non-zero entries.

Segregated Solution Procedure (1)

- Apply finite-volume discretization separately to each momentum equation
  - ▷ For u equation, v and p are frozen
  - ▷ For v equation, u and p are frozen
- Each equation results in linearized system

$$A^{(u)}u = b^{(u)}$$

$$A^{(v)}v = b^{(v)}$$

where the momentum source terms consist of pressure terms and other terms due to velocity gradients and non-uniform viscosity.

$$b^{(u)} = b_o^{(u)} - b_p^{(u)}$$

$$b^{(v)} = b_o^{(v)} - b_p^{(v)}$$

$b_p^{(u)}$ and $b_p^{(v)}$ are contributions due to the frozen pressure field (p^*).

Segregated Solution Procedure (2)

Discretization of momentum equations results in non-linear, coupled system of equations for u and v (in 2D).

$$A^{(u)}u = b^{(u)} \qquad A^{(v)}v = b^{(v)}$$

- $A^{(u)}$ is the coefficient matrix, and $b^{(u)}$ is the source term (vector) for the u equation when v and p are frozen.
- $A^{(v)}$ is the coefficient matrix, and $b^{(v)}$ is the source term (vector) for the v equation when u and p are frozen.
- Momentum equations are nonlinear, so $A^{(u)}$ and $A^{(v)}$ depend on u and v.
- Nonlinearity requires iterative solution: $A^{(u)}$ and $A^{(v)}$ are updated with new values of u and v, and the systems are solved again.

What about the continuity equation?

Where is the equation for pressure?

Pressure in Momentum Source Terms

Pressure is the dominant source term in the momentum equations

$$A^{(u)}u = b^{(u)} \quad \text{where} \quad b^{(u)} = b_P^{(u)} + b_o^{(u)} \sim d_u(p_W - p_P) + \ldots + b_o^{(u)}$$

$$A^{(v)}v = b^{(v)} \quad \text{where} \quad b^{(v)} = b_P^{(v)} + b_o^{(v)} \sim d_v(p_S - p_P) + \ldots + b_o^{(v)}$$

The d_u and d_v are coefficients that depend on the geometry of of the control volume.

Note: The $\sim$ symbol is meant to convey the basic features of the terms, and allows a degree of notational abuse. To make these expressions more precise, read the $\sim$ symbol as "contains terms like".

Pressure does not have its own *natural* equation for incompressible flow.

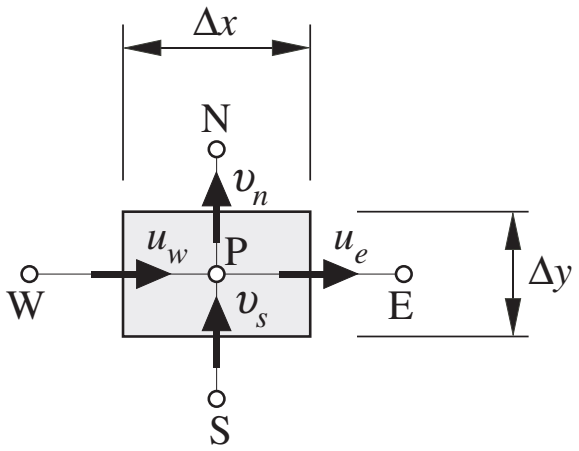
Pressure Correction and Continuity (1)

Solution to the linearized momentum equations

$$A^{(u)}u = b^{(u)}$$

$$A^{(v)}v = b^{(v)}$$

gives updated u and v fields.



In general these new u and v fields will not satisfy the continuity equation for each cell

$$u_w\Delta y - u_e\Delta y + v_s\Delta x - v_n\Delta x \neq 0$$

Pressure Correction and Continuity (2)

Assume that we can find a *velocity correction* at each node that adjusts the velocity field so that it satisfies the continuity equation.

$$u_w = \tilde{u}_w + u'_w \tag{1}$$

$$v_s = \tilde{v}_s + v'_s \tag{2}$$

- u_w and v_s are the velocity components that satisfy the continuity equation.
- $\tilde{u}_w$ and $\tilde{v}_s$ are the velocity components obtained by satisfying the momentum equations.
- u'_w and v'_s are the corrections to the velocity components needed to satisfy the continuity equation.

Equations (1) and (2) define u'_w, and v'_s values *for each cell* in the domain. The $\tilde{u}_w$ and $\tilde{v}_s$ fields are obtained by solving the momentum equations. We need to develop another procedure for computing the u'_w and v'_s fields. Ultimately the u'_w and v'_s fields are obtained by requiring each cell to satisfy the continuity equation.

Pressure Correction and Continuity (3)

Assume that the velocity corrections can be derived from suitable *pressure corrections*

$$p_P = \tilde{p}_P + p'_P \tag{3}$$

- p_P is the desired pressure at the cell center, i.e., the one that brings the velocity field into balance so that continuity equation is satisfied for each cell.
- $\tilde{p}_P$ is the current guess at the pressure at the cell center.
- p'_P is the change in $\tilde{p}_P$ needed to satisfy the continuity equation.

Pressure Correction and Continuity (4)

Unifying assumption:

Assume that the velocity corrections are uniquely determined by the pressure corrections.

$$u'_w \approx c_w(p'_W - p'_P) \tag{4}$$

$$v'_s \approx c_s(p'_S - p'_P) \tag{5}$$

where c_w and c_s are coefficients that depend on terms from the momentum equation (we've skipped a couple of steps here).

Pressure Correction and Continuity (5)

Tying the velocity corrections to the pressure corrections allows us to close the system of equations. An equation for the p' is obtained by substituting the Equations (1), (2), (4), and (5) into the continuity equation. The result is

$$A^{(p')}p' = b^{(p')}$$

where

$$b^{(p')} \sim \tilde{u}_w \Delta y - \tilde{u}_e \Delta y + \tilde{v}_s \Delta x - \tilde{v}_n \Delta x$$

In words: *The source term for the p' field is the* local *error in the continuity equation.*

The Pieces (in 2D)

The momentum equations

$$A^{(u)}u = b^{(u)} \qquad A^{(v)}v = b^{(v)} \tag{$\star$}$$

The pressure correction equation

$$A^{(p')}p' = b^{(p')} \tag{$\star\star$}$$

Momentum and pressure *corrections*

$$u = \tilde{u} + u' \qquad v = \tilde{v} + v' \qquad p = \tilde{p} + p'$$

$\tilde{u}$ and $\tilde{v}$ are the solutions to Equations ($\star$) obtained when the previous field variables (u, v, and p) are used to compute $A^{(u)}$, $b^{(u)}$, $A^{(v)}$ and $b^{(v)}$. $\tilde{p}$ is the pressure field from the previous global iteration.

p' is the solution to Equation ($\star\star$) when $b^{(p')}$ is computed from the newly obtained $\tilde{u}$ and $\tilde{v}$.

SIMPLE Algorithm

Initialize the velocity and pressure fields, then iterate until convergence:

1. Compute $A^{(u)}$ and $b^{(u)}$, and solve $A^{(u)}u = b^{(u)}$ to obtain $\tilde{u}$, a tentative guess at the x-direction velocity field.
2. Compute $A^{(v)}$ and $b^{(v)}$, and solve $A^{(v)}v = b^{(v)}$ to obtain $\tilde{v}$, a tentative guess at the y-direction velocity field. Note that $A^{(v)}$ and $b^{(v)}$ use the most recently available $\tilde{u}$ field, and the v field from the previous iteration
3. Compute $A^{(p')}$ and $b^{(p')}$, and solve $A^{(p')}p' = b^{(p')}$ to obtain the pressure correction field p'.
4. Correct the velocities and pressures.

$$p_P \leftarrow \tilde{p}_P + p'_P \qquad u_w \leftarrow \tilde{u}_w + u'_w \qquad v_s \leftarrow \tilde{v}_s + v'_s$$

5. Test for convergence, and return to Step 1 if not converged.

The SIMPLE algorithm is described by Patankar [2] and Ferzier and Perić [1, § 7.5.1], and Yao et al., §4.3.3.

Convergence Checking (1)

Each time through the SIMPLE loop, $A^{(u)}$ and $b^{(u)}$ are updated with the newest information about the u, v and p fields.

Before solving

$$A^{(u)}u = b^{(u)}$$

compute the residual

$$r^{(u)} = b^{(u)} - A^{(u)}u$$

If $r^{(u)}$ is small, then the old u is a good approximation to the system of equations defined by the newly updated coefficients in $A^{(u)}$ and $b^{(u)}$.

Similarly,

$$r^{(v)} = b^{(v)} - A^{(v)}v$$

is a measure of how well the old v satisfies the linearized equations based on the newly updated coefficients in $A^{(v)}$ and $b^{(v)}$

Convergence Checking (2)

$r^{(u)}$ and $r^{(v)}$ are residual vectors. To check convergence, we need to look at a scalar value. Let σ be a normalized scalar value of the residual.

$$\sigma_u = \frac{\|r_u\|}{\|r_{u,\mathrm{ref}}\|}, \qquad \sigma_v = \frac{\|r_v\|}{\|r_{v,\mathrm{ref}}\|}$$

The values of $\|r_{u,\mathrm{ref}}\|$ and $\|r_{v,\mathrm{ref}}\|$ are chosen so that $\sigma_u \sim 1$ and $\sigma_v \sim 1$ on the first iteration, or close to the first iteration.

Convergence Checking (3)

For the pressure correction equation, the residuals are not as informative as the source term

$$b^{(p')} \sim \tilde{u}_w \Delta y - \tilde{u}_e \Delta y + \tilde{v}_s \Delta x - \tilde{v}_n \Delta x$$

If $b^{(p')}$ is small, then the current guess at the velocity field comes close to satisfying the continuity equation *before* the pressure and velocity corrections are applied.

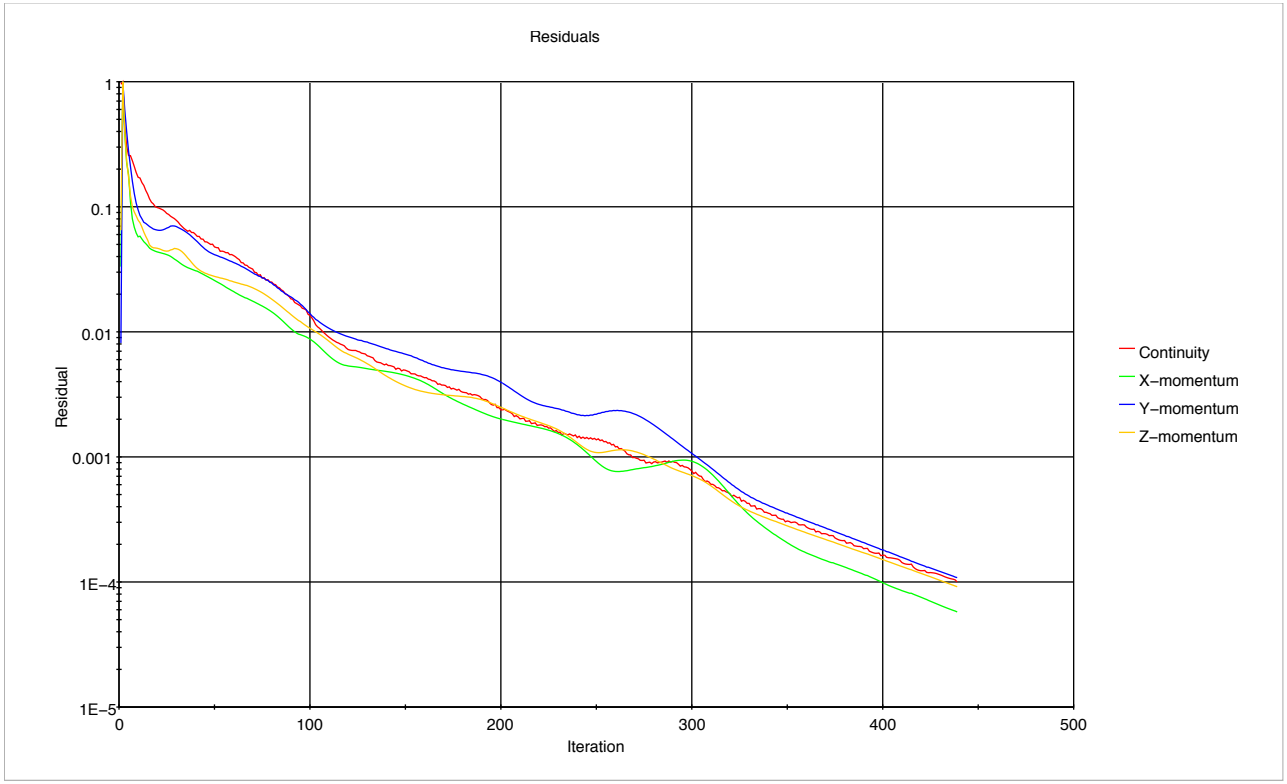
To check convergence of the continuity equation, monitor σ_c

$$\sigma_c = \frac{\|b^{(p')}\|_\infty}{\|b^{(p')}\|_{\infty,\text{ref}}}$$

Recall that $\|x\|_\infty$ is equivalent to $\max |x_i|$

Monitoring Residuals

Convergence history for a well-behaved, laminar flow simulation with STAR-CCM+:



Summary

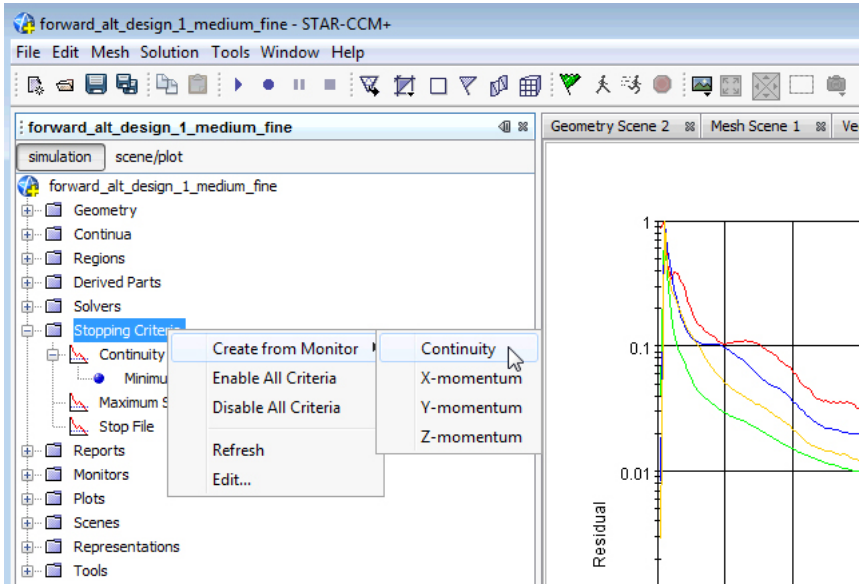
- SIMPLE is a *segregated* solution method: The u, v, w, and p fields are solved separately. Coupling between these field variables is achieved via velocity and pressure corrections.
- Convergence occurs when the residuals for each of the equations is reduced to a value below a tolerance.
- The "residuals" are normalized scalars. The normalization is chosen so that $\sigma \sim 1$ on the first iteration.
- σ_c is the largest local (cell-wise) error in the continuity equation *before* the pressure-based corrections to the velocity fields are applied. After the velocity field is corrected, local mass conservation is obtained for all cells. Thus, at the end of each iteration of SIMPLE, both global *and* local mass conservation is guaranteed.
- In version 8.04 of StarCCM+, the default stopping criterion is to run the simulation for 1000 outer iterations. Turning on convergence checking requires user action.

Adding a Convergence Criterion in Star-CCM+

To add a convergence criterion in Star-CCM+

1. Right click on Stopping Criteria
2. Select Create from Monitor and select one of the field variables.

I recommend using *Continuity* as a baseline convergence constraint.
For harder problems, one of the turbulence field variables or temperature might also be used as a convergence contraint.



Recommendations

- *Always* inspect the residual plot after each run.
- If the residual tolerances have not been met, restart the solution, using the last iteration as a starting point. (Don't start from scratch.)
- For some hard problems, other adjustments to the solution parameters might be necessary in order obtain convergence.
- STAR-CCM+ also has a *coupled flow solver*, which may converge better (and faster) than the segregated solver. The coupled flow solver uses more memory than the segregated solver.

References

[1] Joel H. Ferziger and Milovan Perić. *Computational Methods for Fluid Dynamics*. Springer-Verlag, Berlin, third edition, 2001.

[2] S.V. Patankar. *Numerical Heat Transfer and Fluid Flow*. Hemisphere, Washington D.C., 1980.