Computers and Fluids 168 (2018) 54–66

Contents lists available at ScienceDirect

Computers and Fluids

journal homepage: www.elsevier.com/locate/compfluid

Benchmark solutions

Implicit boundary conditions for coupled solvers

M. Darwish [a,*], L. Mangani [b], F. Moukalled [a]

[a] Department of Mechanical Engineering, American University of Beirut, P.O. Box 11-0236, Riad El Solh, Beirut 1107 2020, Lebanon
[b] Hochschule Luzern, Technik und Architektur, Horw CH-6048, Switzerland

ARTICLE INFO

Article history:
Received 29 September 2017
Revised 13 February 2018
Accepted 13 March 2018
Available online 14 March 2018

Keywords:
Finite volume method
Pressure equations
Boundary conditions
Coupled algorithm
Velocity–pressure coupling

ABSTRACT

In addition to implicitly resolving the velocity pressure coupling at interior faces, it is also critical for a coupled solver to implicitly resolve this coupling for boundary faces. Failure to account for this leads to substantial degradation in the convergence characteristics of the solver. In this paper, details on the implicit treatment of boundary conditions for a coupled solver are presented in the context of the OpenFOAM® framework. The boundary conditions presented include two geometric conditions namely, cyclicMMI and symmetry condition and a number of physical boundary conditions (inlet, outlet, and wall). The treatment is illustrated with modification to the boundary element coefficients and its effectiveness demonstrated for the case of "NASA Rotor 37", a standard validation case.

1. Introduction

In the pressure-based finite volume method used for Computational Fluid Dynamics (CFD) applications, two approaches have been followed to resolve the velocity–pressure coupling. In the first, denoted by the segregated approach, the velocity components and pressure are solved sequentially one at a time with the inter-equation influences treated explicitly. In the second, denoted by the coupled approach, the velocity and pressure equations are solved simultaneously with the inter-equation influences treated implicitly.

The SIMPLE family of algorithms [1–12] that are at the core of the segregated approach have been quite popular and have been used quite successfully over the last few decades to solve a wide range of flow problems that involve incompressible and compressible flows [13,14], single and multiphase flows [15–20], laminar and turbulent flows, free-surface flows [21,22], and particle laden flows [19] to cite a few. This popularity was partially due to their lower storage requirement in addition to relatively simple and more forgiving implementation as compared to coupled approach. The downside, however, was that the segregated algorithms fail to scale linearly with grid size, and are very sensitive to initial conditions. This has meant that despite their wide adoption and many successes, they continue to suffer from a breakdown in convergence rate when applied to the solution of large scale problems, and decreased robustness associated with sensitivity to initial conditions when simulation complex physics. This failure to scale linearly with grid size is due to the explicit fashion in which the segregated algorithms resolve the velocity–pressure coupling in the discretized Navier–Stokes equations, which in compressible flows extends to a pressure–velocity–density coupling.

The coupled approach requires a more complex set of ingredients to work properly, but when it does it yields drastically increased robustness and near linear scaling with mesh size, a great advantage when dealing with complex physics and/or large scale simulation problems, both of which are becoming more common nowadays. With linear scaling the solver yields nearly constant CPU time per element as mesh size is increased, two, four, ten or even thirty fold [23]. This means that the performance difference between segregated and coupled solver increases substantially with grid size, with ratio of 6–50 times reported on some problems [24].

It is worth noting that the Imperial College CFD group [1,2], which developed the SIMPLE algorithm, had initially developed a coupled pressure-based solver, SIVA, [25]. The SIVA algorithm was later overshadowed by the SIMPLE algorithm that combined low memory requirement with coding simplicity, two substantive advantages given the state of computer technology at that time.

Based on the experience of the authors, a successful implementation of a coupled approach, requires a set of basic ingredients: (i) a diligent linearization of the momentum and pressure equations [25–30] yielding an extended system of coupled equations; (ii) a highly efficient and scalable multigrid linear solver to solve this extended system of equations [31]; and (iii) a fully implicit implementation of boundary conditions. While the first two features

* Corresponding author.
E-mail addresses: darwish@aub.edu.lb, darwish@me.com (M. Darwish).

https://doi.org/10.1016/j.compfluid.2018.03.046
0045-7930/

Nomenclature

$a_C^{uu}, a_F^{uv}, ..$	coefficients in the discretized equations
$\mathbf{B}$	source term in the momentum equation
b_C^{ϕ}	source term in the discretized equations
C	main grid point
$\mathbf{d}_{CF}$	vector joining the grid points C and F
$\mathbf{D}$	operator used in the pressure equation
$\mathbf{E}$	component of the surface vector in the direction of $\mathbf{d}_{CF}$
F	refers to neighbour of the C grid point
g	geometric interpolation factor
$\mathbf{i}, \mathbf{j}$	unit vectors in the x- and y-direction, respectively
$\dot{m}_f$	mass flow rate at control volume face f
p	pressure
C	main grid point
Q	general source term
RMS	root mean square residuals
$\mathbf{S}$	surface vector
t	time
$\mathbf{T}$	Component of the surface vector normal to $\mathbf{S}$
u,v	velocity components in x- and y-direction, respectively
$\mathbf{v}$	velocity vector
V	cell volume

Greek symbols

μ	dynamic viscosity
ρ	fluid density
τ	deviatoric stress tensor

Subscripts

C	refers to main grid point
f	refers to control volume face
F	refers to the F grid point
nb	refers to values at the faces obtained by interpolation between C and its neighbours
NB	refers to the neighbours of the C grid point

Superscripts

p	refers to pressure
u	refers to the u-velocity component
v	refers to the v-velocity component
$\circ$	refers to value at the previous time step
$*$	refers to value at the previous iteration
$**$	refers to value at the current iteration
—	refers to an interpolated value

have been summarized by the authors in several papers [25–31], the last component has been rarely addressed on its own and has generally received a cursory overview.

To this end, the aim of the paper is to report on the fully implicit implementation of boundary conditions in a coupled solver. The details are presented in the context of the OpenFOAM® framework. The two types of boundary conditions encountered in solving flow problems, namely physical and geometric boundary conditions [32], are considered. Physical boundary conditions represent the conditions imposed at inlets, outlets, and walls. Geometric boundary conditions are generally used to simplify the geometric domain, such as the symmetry boundary condition and the various types of cyclic boundary conditions (rotational, translational, etc.).

In the remainder of this article, the governing equations and the coupled discretization approach are first briefly reviewed. The two types of boundary conditions encountered in solving flow problems, namely physical (inlets, outlets, and walls) and geometric (symmetry, cyclic translational, and cyclic rotational) boundary conditions [32] are then derived in details and implementation issues clarified. To assist in this endeavour, the topology structures used in the OpenFOAM® framework for the representation of the discretized equations in the form of upper, lower, and diagonal arrays are also presented. Finally, the approach is evaluated using a test case that incorporates many of the treated boundary conditions.

2. Governing equations

The mass, momentum, and energy equations governing fluid flow and heat transfer problems are written as

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{1}$$

$$\frac{\partial [\rho \mathbf{v}]}{\partial t} + \nabla \cdot \{\rho \mathbf{v}\mathbf{v}\} = -\nabla p + [\nabla \cdot \tau] + \mathbf{f}_b \tag{2}$$

$$\frac{\partial}{\partial t}(\rho c_p T) + \nabla \cdot [\rho c_p \mathbf{v} T] = \nabla \cdot [k \nabla T] + \rho T \frac{Dc_p}{Dt} + \frac{Dp}{Dt} - \frac{2}{3}\mu \Psi + \mu \Phi + \dot{q}_V \tag{3}$$

where $\mathbf{v}$, p, T, ρ, $\mathbf{f}_b$, c_p, k, μ,and $\dot{q}_V$ represent the velocity vector, pressure, temperature, density, body force per unit volume, specific heat at constant pressure, thermal conductivity, dynamic viscosity, and heat generation per unit volume, respectively. In addition, τ is the deviatoric stress tensor that can be written for a Newtonian fluid as

$$\tau = \mu\left(\nabla \mathbf{v} + (\nabla \mathbf{v})^T\right) - \frac{2}{3}\mu(\nabla \cdot \mathbf{v}) \tag{4}$$

Since the energy equation is solved after simultaneously solving the continuity and momentum equations, its discretization and the implementation of its boundary conditions follow the segregated approach [32] and will not be discussed here. Therefore, the paper concentrates on the implementation of boundary conditions for the continuity and momentum equations in the coupledFOAM solver for both incompressible and compressible flows. For compressible flow, an equation of state relating density to temperature (T) and pressure is required, which for an ideal gas is given by

$$\rho = \frac{P}{RT} = C_\rho p \tag{5}$$

where R is the gas constant.

3. The discretized equations

The fully implicit coupled discretization has been covered in details in a number of papers [24–30]. Only the basic features of the coupled discretization will be described in this section.

For the discretized momentum equations can be written in vector form as follows:

$$a_C^{\mathbf{vv}} \mathbf{v}_C + \sum_{F=NB(C)} a_F^{\mathbf{vv}} \mathbf{v}_F + a_C^{\mathbf{v}p} p_C + \sum_{F=NB(C)} a_F^{\mathbf{v}p} p_F = \mathbf{b}_C^{\mathbf{v}} \tag{6}$$

where the coefficients are given by

$$a_C^{\mathbf{vv}} = \frac{\rho_C^* V_C}{\Delta t} + \sum_{f=nb(C)} \left(\mu_f \frac{E_f}{d_{CF}} + \left\| \dot{m}_f^*, 0 \right\| \right)$$

$$a_F^{\mathbf{vv}} = -\mu_f \frac{E_f}{d_{CF}} - \left\| -\dot{m}_f^*, 0 \right\|$$

$$a_C^{\mathbf{v}p} = g_C \mathbf{S}_f$$

$$a_F^{\mathbf{v}p} = g_F \mathbf{S}_f$$