A Domain Decomposition Strategy for Simulation of Industrial Fluid Flows

Rune Teigland

1 Introduction

Multiblock methods are often employed to compute flows in complex geometries. This paper describes a robust and efficient multiblock solution procedure for general three dimensional flow situations involving both single- and multi-phase flows. The multiblock approach is implemented within the framework of the well known SIMPLE solution strategy. The multiblock linear solvers are based upon acceleration of the basic additive Schwarz method using Krylov subspace methods in the 'outer iterations'. The effect of incorporating the multiblock linear solver within the SIMPLE solution procedure is being discussed in some detail. Test results of the numerical solution and convergence behavior on several problems involving incompressible single- and multi-phase flows are presented. The problems chosen involve fearly simple geometries, yet they illustrate the effect of using the multiblock procedure in a general purpose code.

The multiblock approach (in 3D) is to segment the physical region into contiguous subregions, each bounded by six curved sides and each of which transforms to a cubic block in the computational region. Each grid block is assumed to be topologically cubic (in 3D) and has its own curvilinear coordinate system. Furthermore, the use of multiblock grids is advantageous in terms of economic use of memory and the possibility to use different flow equations in different blocks. One should also notice the potential for significant speedup on parallel machines in using the multiblock approach as compared to using a single block approach. Since multiblock grids are unstructured on the block level, information is needed on the connectivity of neighbouring blocks along with their orientation. Each block has its own local coordinate system, needed to provide geometrical flexibility. The multiblock approach can also be used to handle situations involving cyclic boundary conditions, here a block can 'wrap around' the same edge.

The Krylov methods used here are the standard conjugate gradient method (used for solving the pressure correction equation in situations involving incompressible flows)

Ninth International Conference on Domain Decomposition Methods
Editor Petter E. Bjørstad, Magne S. Espedal and David E. Keyes

and the BiCGSTAB method proposed in [VdV92]. One should notice that for general compressible flows the arising pressure correction equation is also nonsymmetric. In the case of incompressible flow the pressure correction equation can be solved using the Conjugate Gradient method, an option given in our code. Multiblock approaches to solving viscous fluid flow problems can be found in e.g. [KG93, TF92].

2 Governing Equations and Discretization

The multiblock linear solver presented here has been implemented into a computer program [GST93] for simulation of fluid flows in complex three dimensional geometries. The code is based on the use of general curvilinear coordinates and is applicable to both laminar and turbulent flows as well as multi-phase flows using a model for dispersed flow. It is assumed that the fluid is a Newtonian fluid such that the flow is governed by the Navier-Stokes equations which express conservation of mass and momentum. In the case of turbulent reactive flows the equations are augmented by equations for the conservation of enthalpy, a standard turbulence model as well as equations describing the conservation of mass fraction of a chemical specie. The conservation equations may be written in general form as follows:

$$\frac{\partial(\rho\phi)}{\partial t} + \frac{\partial(\rho u_j \phi)}{\partial x_j} = \frac{\partial}{\partial x_j}(\Gamma_\phi \frac{\partial \phi}{\partial x_j}) + S_\phi \tag{2.1}$$

where ϕ represents different conserved quantities such as momentum, continuity, enthalpy, turbulent kinetic energy etc. In the steady state case this is a prototype of a scalar advection-diffusion equation. These equations are discretised on a non-orthogonal co-located grid arrangement. If a flux vector $\boldsymbol{J}_j$ containing convection and diffusion is defined as

$$\boldsymbol{J}_j = \rho u_j \phi - \Gamma_\phi \frac{\partial \phi}{\partial x_j} \tag{2.2}$$

equation (2.1) can be written as

$$\frac{\partial(\rho\phi)}{\partial t} + \frac{\partial \boldsymbol{J}_j}{\partial x_j} = S_\phi \tag{2.3}$$

We use a finite volume discretization scheme where the equations for the conserved quantities are integrated over a general non-orthogonal control volume. Multiblock grids are handled using a local coordinate system. These local coordinate systems may be nontrivially related as block boundaries are crossed, (e.g. orientation change as well as discontinuous grid line slopes as in the example shown in Figure 1). Thus, the discretization of the equations proceeds block by block, exactly as in the single block case. Implicit time discretization (backward Euler) for the transient term yields

$$\int_{\delta V} \frac{\partial(\rho\phi)}{\partial t} = \frac{(\rho_p \phi_p - \rho_p^0 \phi_p^0)}{\triangle t} \delta V_p \tag{2.4}$$

where superscript 0 denotes values from the previous time step. The general conservation equation for ϕ, equation (2.1), is integrated over a three-dimensional control volume δV_p in physical space such that after employing Gauss divergence theorem one gets

$$\frac{(\rho_p\phi_p - \rho_p^0\phi_p^0)}{\triangle t}\delta V_p + \sum_{nn} \boldsymbol{J}\cdot\boldsymbol{A} \mid_{nn} = S_p \tag{2.5}$$

where nn denotes the cell face index, p is the cell-center index, $\boldsymbol{A}$ area vector and S_p is the total source in the control volume. Because the grid is non-orthogonal the derivatives that occur in the viscous and pressure terms must be evaluated in the transformed curvilinear coordinate system (ξ_1, ξ_2, ξ_3). If we use the chain rule for differentiation, we get

$$\frac{\partial\phi}{\partial x_j} = \sum_l \frac{\partial\xi_l}{\partial x_j}\frac{\partial\phi}{\partial\xi_l} = \sum_l e^l \frac{\partial\phi}{\partial\xi_l} \tag{2.6}$$

where e^l are the contravariant basis vectors of the curvilinear coordinates. Details of the derivation can be found in [Mel90, BW87]. In order to alleviate checkerboard oscillations in pressure (due to the use of a co-located grid arrangement) we use a pressure weighted interpolation of the cell-face velocities in the discretised continuity equation. The idea goes back to Rhie and Chow [RC83].

The coupled momentum and continuity equations are solved in a sequential manner using the SIMPLE method of Patankar and Spalding [PS72]. Notice that the arising linear system for each component of the velocity is nonsymmetric. The SIMPLE solution method belongs to the well-known class of pressure correction methods popular in the primitive variable computation of complex incompressible flows [Dec92]. The methods use a predictor-corrector approach to the solution of the Navier-Stokes equations. The SIMPLE method is widely used in general purpose CFD codes. It is a robust methodology applicable for complex, turbulent recirculating flows and it is potentially applicable to all regimes from incompressible laminar flows to supersonic flows. The basic SIMPLE solution strategy consists of iteratively solving the momentum and the pressure correction equations. The solution of scalar equations such as equations for the turbulence production rate and turbulence dissipation rate are then solved using essentially the same basic form of the equations (2.1). The complete main loop in each time step is given by

Algorithm 1 (Main loop)

```
Update primary and derived fields, time,
boundary conditions etc.
Iteration Loop:
   Solve u-velocities for each phase
   Solve v-velocities for each phase
   Solve w-velocities for each phase
   Calculate advective fluxes by Rhie and Chow interpolation
   Solve pressure correction
      Update pressure, velocity, advective fluxes and density
   Solve volume fractions (if number of phases > 1)
repeat (if necessary)
Solve scalar fields
```

When only the steady-state solution is of interest, the time step, $\triangle t$, is used as a parameter through which the convergence rate may be optimized. The viscous flux (see (2.5)) is split in a primary flux that contains the orthogonal terms (relating to grid) which are treated implicitly, while the other terms are treated explicitly (i.e. lumped into the source terms). Thus the arising system of linear equations takes the form

$$a^p \phi^p = \sum_{nb} a^{nb} \phi^{nb} + S_p \tag{2.7}$$

where each point p is coupled to its six neighbours (nb) and S_p denotes the discretised source term. The source term is split such that the resulting matrix associated with the linear system (2.7) is an M-matrix with constant bandwidth. In the approximate inversion of the subdomain matrices on each block an Incomplete Factorization method is being used here. The M-matrix property is important for the existence of the Incomplete Factorization method used as preconditioner in connection with the Krylov subspace methods [Hac94].

3 Multiblock Linear Solver

In this section we present the multiblock linear solvers based upon acceleration of the basic Schwarz alternating method. A thorough discussion and derivation of various domain decomposition algorithms can be found in [SBG96].

The alternating Schwarz method consists of dividing the computational domain into overlapping domains and using efficient solvers on the subdomains. It must be emphasized that the solution of the subproblems can never solve the complete problem on the composite domain, but only represent a partial step of the algorithm. The presentation and derivation of the additive Schwarz method can be cast using the linear algebra framework, see e.g. [SBG96] or [Hac94].

Let the corresponding matrices for the local subproblems be denoted by A_i. We define a (rectangular) restriction matrix R_i such that when applied to a vector x, it gives back the vector of smaller dimension x_i having components that refers to grid points in subdomain Ω_i. Similarly the prolongation operator is defined as the adjoint of the restriction operator (not necessary in general but used here in the derivation). The submatrices A_i corresponding to the local subproblems can either be formed using the same discretization procedure as for the composite problem (this is the route we have chosen) or they can be computed automatically via a so-called Galerkin formulation, see e.g. [Wes92]. The Galerkin (or variational) method of constructing these submatrices is

$$A_i = R_i A R_i^T \tag{3.8}$$

If we define the operator

$$B_i = R_i^T \left(R_i A R_i^T \right)^{-1} R_i \tag{3.9}$$

the Additive Schwarz method can be written as

$$x^{n+1} = x^n + B(b - Ax^n) \tag{3.10}$$

where b denotes the right hand side and A and x are composite grid matrix and solution vector respectively, and n is the iteration number. The preconditioner B is given by

$$B = \sum_i B_i \tag{3.11}$$

The iteration procedure can now be accelerated by using some appropriate Krylov subspace method. The main goal is the solution of the composite grid problem and the coupling between grid blocks has to be accounted for. The coupling is via the preconditioner and the overlap between grid blocks. In the case that meshing in the subregions is identical in the overlapping region the solution procedure simplifies. Interpolation of the variables is then avoided since the overlapping control volumes are made identical to the corresponding internal control volumes. Where the grid blocks are not connected, the extra control volumes are collapsed. The variables will then be located on the sides of the control volumes as before and will define boundary values. In the algorithm below $(\cdot,\cdot)$ denotes the inner product between two vectors.

Algorithm 2 (Multiblock Bi-CGSTAB linear solver)

x_0 *initial guess*, $r_0 = B(b - Ax_0)$;
($\bar{r_0}$ *arbitrary such that* $(\bar{r_0}, r_0) \neq 0$)
$\omega_0 = \rho_0 = \alpha_0 = 1$ *and* $\nu_0 = p_0 = 0$;
for $i = 1, 2, 3, \ldots$
    $\rho_i = (\bar{r_0}, r_{i-1})$;
    $\beta_{i-1} = (\rho_i/\rho_{i-1})(\alpha_{i-1}/\omega_{i-1})$;
    $p_i = r_i + \beta_{i-1}(p_{i-1} - \omega_{i-1}\nu_{i-1})$;
    $\hat{p} = Bp_i$; $\nu_i = A\hat{p}$;
(*add contributions + overlap, i.e. after* $\nu_i = A\hat{p}$ *for* ν_i *variable*).
    $\alpha_i = \rho_i/(\bar{r_0}, \nu_i)$; $s = r_i - \alpha_i\nu_i$;
    if $\|s\|$ *small?* ***then*** $x_{i+1} = x_i + \alpha_i\hat{p}$; *exit loop*;
    $z = Bs$; $t = Az$;
(*add contributions + overlap, i.e. after* $t = Az$ *for* t *variable*).
    $\omega_i = (t, s)/(t, t)$;
    $x_{i+1} = x_i + \alpha_i\hat{p} + \omega_i z$;
    if x_i *accurate enough?* ***then*** *exit loop*;
    $r_{i+1} = s - \omega_i t$;
end for

4 Multiblock Strategies Combined into the SIMPLE Solution Method

The flow field in all grid blocks is solved using the multiblock linear solvers described in the previous section. Some modifications to the SIMPLE algorithm have to be done.

The following algorithm describes what we call the SIMPLE-Schwarz algorithm, i.e. it uses the multiblock linear solvers described above.

Algorithm 3 (SIMPLE-Schwarz method)

```
Time step loop:
    Update primary and derived fields, time, boundary conditions etc.
    Iteration Loop:
        Solve u-velocities (for all blocks (algorithm 2)) for each phase
        Solve v-velocities (for all blocks (algorithm 2)) for each phase
        Solve w-velocities (for all blocks (algorithm 2)) for each phase
        Calculate advective fluxes by Rhie and Chow interpolation
        Solve pressure correction (for all blocks (algorithm 2))
        Update pressure, velocity, advective fluxes and density
        Solve volume fractions (for all blocks (algorithm 2))
    repeat (if necessary)
    Solve scalar fields (for all blocks (algorithm 2))
Repeat
```

In the steps above all equations are solved in turn over the entire composite region. The updating of variables as well as the calculation of advective fluxes is done for all blocks. An alternative would be to perform the interblock coupling once every time step, i.e. as in the algorithm below

Algorithm 4 (Alternative time stepping loop)

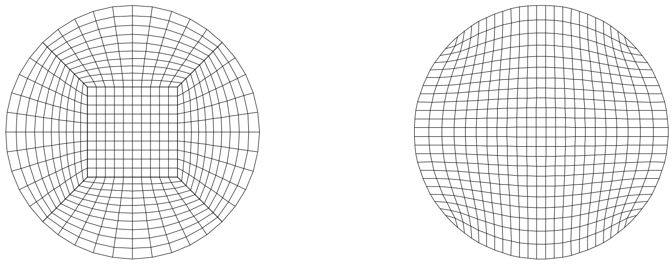
```
Time step loop:
do i = 1,..., max. number of blocks
    SIMPLE loop given by algorithm 1
    Update boundary conditions (inner and outer)
end do
```

Most of the routines used need no modification to handle the multiblock situation. Pointers to the different block variables being accessed have to be set up properly. A couple of situations warrant extra precautions. The Rhie and Chow algorithm involves addressing pressure variables at two neighboring points in each direction, and the other situation is when extrapolating variables to boundaries. In situations involving transient incompressible flows or flows involving elliptic regions algorithm 3 is more suitable than algorithm 4 since in those situations the coupling is stronger across block boundaries. Furthermore, in incompressible flows pressure disturbances are felt instantly over the entire region and thus using algorithm 3 seems more appropriate in that case.

5 Numerical Experiments

In this section we present results of a few numerical experiments involving incompressible turbulent flow in a straight pipe section as well as an example involving gravitational sedimentation. Simulation of turbulent flow in pipes is an important field of research, e.g. in the development of multiphase flow metering devices. Preliminary

Figure 1 Cross section of grid, left: 5 block configuration, right: H-type grid.



results with our multiblock approach on some simple laminar flows has been presented previously, see [Tei95].

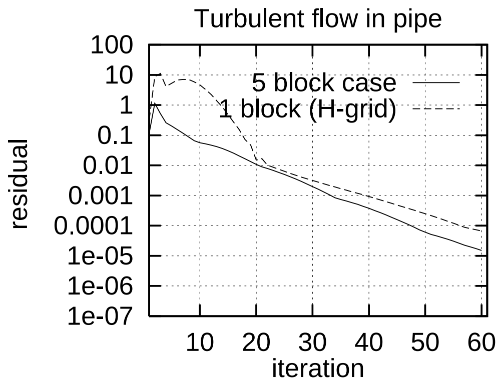
The first set of problems considered here involve 3D turbulent flow in a cylindrical pipe using a five-block configuration and a one block configuration using an H-type grid. In both cases a uniform flow is given at the inlet, and at the outlet variables are extrapolated. The Reynolds number is 32300 and the total number of cells is 10000. The length of the pipe is $20m$ and the radius of the pipe is $1.4m$. The prescribed inlet velocity was $9.9m/s$. This problem was run for a one block H-type grid and a five block grid, see Figure 1.

We notice that in the H-type grid there are 4 corner cells degenerating into triangles. In the figure below we show results of the convergence history of mass residuals versus time iteration with the 5-block and 1-block configuration (H-type grid). We notice that the 5-block configuration is more efficient. H-type grids are widely used in simulation of flow in pipes using a single block approach. We remark that the multiblock approach (5-block case) can easily be parallelized and therefore a significant speedup can potentially be achieved compared to the use of a single block grid (H-type grid).

The last example consists of gravitational sedimentation of an initially homogeneous mixture of a gas and a liquid phase. The densities of the gas and liquid phases are $\rho_{\text{gas}} = 1.2kg/m^3$ and $\rho_{\text{liquid}} = 980kg/m^3$, respectively. The viscosity of the phases are chosen to be in the range of air and water respectively. The computational domain is a channel of dimensions $1m \times 10m$, with free slip boundaries.

The results from computations with the multiblock version of our code is given in the figure below. The domain was divided into a 2×2 grid block system, each block of size $0.5m \times 5m$ each. In the figure below a vertical cross section (crossing blocks 2 and 4) showing the volume fraction of each phase at different times is shown. These results are identical to results of simulations using a one block configuration [GST93].

Figure 2 Convergence history for 3D turbulent flow in a cylindrical pipe.



6 Conclusions

The multiblock solver presented in this paper has been shown to work well in situations involving various block configurations as well as for different flow situations involving both single- and multi-phase flows. We have focused here on some fairly simple examples involving incompressible flows since in that situation pressure corrections are felt instantaneously across the entire domain. The SIMPLE solution methodology is widely used in multi purpose CFD simulators and is potentially applicable to all Mach number regimes ranging from incompressible laminar flows to supersonic flows. The multiblock approach is well suited for implementation on high performance machines.

Acknowledgement

This research was supported by the Research Council of Norway through programme number STP-30074.

REFERENCES

[BW87] Burns A. and Wilkes N. (July 1987) A finite difference method for the computation of fluid flows in complex three dimensional geometries. Technical Report HL87 1327 (07), Harwell Laboratory.

[Dec92] Deconinck H. (ed) (March 1992) *VKI Lecture Series*, Rhode St. Genese, Belgium. von Karman Institute.

[GST93] Gjesdal T., Storvik I., and Teigland R. (April 1993) MUSIC version 1.0, implementation issues and test examples. CMR Report CMR-93-A20001, Christian Michelsen Research AS, Fantoftvegen 38, N-5036 Fantoft, NORWAY.

[Hac94] Hackbusch W. (1994) *Iterative Solution of Large Sparse Systems of Equations*,

volume 95 of *Applied Mathematical Sciences*. Springer Verlag.
[KG93] Kuerten H. and Geurts B. (1993) Compressible turbulent flow simulation with a multigrid multiblock method. In Melson N. D., Manteuffel T. A., and McCormick S. F. (eds) *Sixth Copper Mountain Conference on Multigrid Methods*, volume CP 3224, pages 305–315. NASA, Hampton, VA.
[Mel90] Melaaen M. (1990) *Analysis of curvilinear non-orthogonal coordinates for numerical calculation of fluid flow in complex geometries.* Dr. ing. thesis, NTH, Trondheim, Norway.
[PS72] Patankar S. and Spalding D. (1972) A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *Int. J. Heat Mass Transfer* 15: 1787–1806.
[RC83] Rhie C. and Chow W. (November 1983) A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation. *AIAA Journal* 21: 1525–1532.
[SBG96] Smith B., Bjørstad P., and Gropp W. (1996) *Domain Decomposition, Parallel Multilevel Methods for Elliptic Partial Differential Equations.* Cambridge University Press.
[Tei95] Teigland R. (1995) An additive Schwarz procedure for complex flows. In Thibault P. A. and Bergeron D. M. (eds) *Proc. Third Annual Conference of the CFD Society of Canada*, pages 145–152.
[TF92] Tu J. Y. and Fuchs L. (1992) Overlapping grids and multigrid methods for three dimensional unsteady flow calculations in IC engines. *Int. J. Numer. Meth. Fluids* 15: 693–714.
[VdV92] Van der Vorst H. (1992) Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of non-symmetric linear systems. *SIAM J. Sci. Stat. Comput.* 13: 631–644.
[Wes92] Wesseling P. (1992) *An Introduction to Multigrid Methods.* Pure and Applied Mathematics. Wiley & Sons.

Figure 3 Results for the sedimentation example using 3 SIMPLE iterations per timestep. Volume fraction, left: Liquid phase, right: Gas phase. Solid line block 2, dotted line block 4.