A Laplacian-based grid manipulator for ALE calculations in screw compressors

John Vande Voorde*, Jan Vierendeels, Erik Dick

Department of Flow, Heat and Combustion Mechanics, Ghent University, Sint Pietersnieuwstraat 41, B-9000 Ghent, Belgium

Abstract

A grid manipulation algorithm is presented that allows CFD calculations in complex moving geometries using block-structured grids. An Arbitrary Lagrangian–Eulerian formulation of the flow problem is used. In terms of grid manipulation, this means that grid nodes are moved in-between time steps without changing the grid topology. The focus of the paper lies on the development of the grid manipulator.

The block-structured grids are constructed from the solution of the Laplace equation $\nabla^2(\Phi) = 0$ obtained on an unstructured grid in the same geometry. High quality meshes are constructed from the gradient lines and the potential lines. As an illustration, a calculation of compressible flow in a screw compressor is set up with the constructed meshes.

Keywords: CFD; Grid manipulation; Arbitrary Lagrangian–Eulerian; Structured grid; Tooth compressor; Internal flow; Moving boundaries

1. Introduction

Calculating the flow through a screw compressor is very difficult. The main obstacle for simulations in screw compressors is the movement of the rotors in the flow domain. As the compression is volumetric, only a time-dependent simulation of (at least) an entire compression cycle is meaningful. Therefore the grid in the flow domain has to move. Due to the specific geometry of these compressors, the movement is very complex. In this paper, the flow through a screw compressor is used as a representative example of the grid manipulation algorithm.

To perform the flow simulation, the time-dependent Arbitrary Lagrangian–Eulerian (ALE) formulation of the flow equations is used. The ALE method [1,2] allows the grid to move with a velocity that is independent from the flow solution. The only limitations are the need to have a valid grid at each time step, and the restriction that the grid topology must be maintained during time stepping. The latter means that at each time step each cell must be defined by the same cell faces and vertices. The main advantage of this method is that the flow equations can be solved time-dependently without need for remapping or flow data interpolation between time steps.

The grid manipulation algorithm presented in this paper manipulates the nodes of the grid based on the solution of the Laplace potential equation in the grid domain. The use of the favorable characteristics of the solution of a potential problem dates back to Winslow in [3] and is a well-known technique [4,5]. The potential solution is not explicitly obtained in Winslow's method, but the position of the mesh nodes reflects this solution.

The difference between the method described in this paper and the methods based on Winslow's idea, is that in the present method the solution of the potential equation is explicitly obtained on an unstructured grid in the flow domain to allow more control over the use of the potential solution. This unstructured grid has this sole purpose and is in no other way related to the flow calculations.

2. The grid manipulation algorithm

The grid manipulation algorithm is based on the generation of 2D structured grids in sections of the compressor house as illustrated in Fig. 1. A 3D grid is constructed from these 2D slices and inlet and outlet ports (with stationary grid) are added.

* Corresponding author. Tel.: +32 926 43312; Fax: +32 926 43586; E-mail: John.Vandevoorde@Ugent.be

Computational Fluid and Solid Mechanics 2005
K.J. Bathe (Editor)

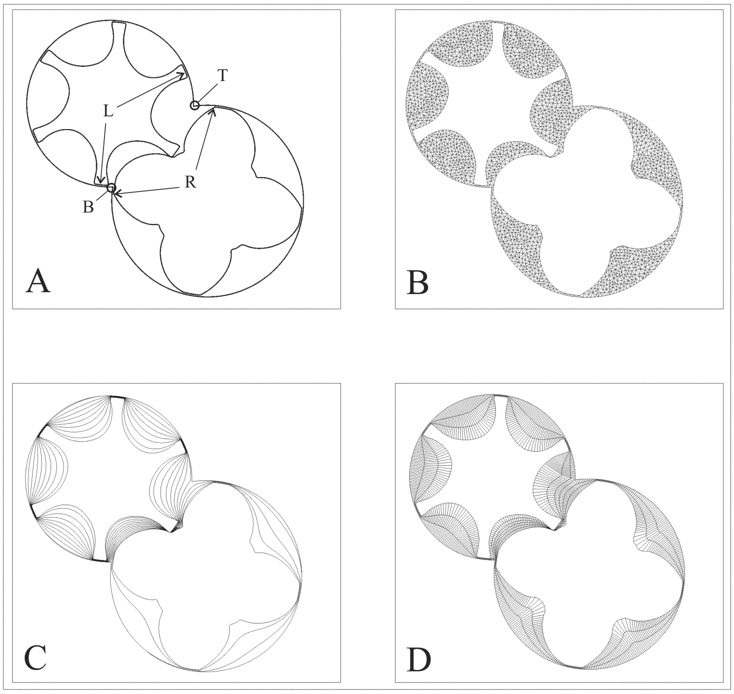

Fig. 1. Three steps in the generation of structured grids: (a) domain; (b) unstructured grid; (c) potential solution obtained on the unstructured grid; (d) final structured grid based on the potential solution.

The 2D structured grid in the compressor house is constructed based on the solution of the potential equation $\nabla^2 (\Phi) = 0$ in the flow domain. To obtain this solution, the flow domain is meshed with an unstructured grid (Fig. 1(b)). The unstructured grid generator is an in-house algorithm [6]. On this unstructured grid, the Laplace equation $\nabla^2 (\Phi) = 0$ is solved with well-chosen boundary conditions (Fig. 1(c)).

A valuable characteristic of the potential solution is that the potential lines (lines of constant potential Φ) never cross. Furthermore, the direction of the gradient $\nabla\Phi$ is perpendicular to the potential lines. The lines tangent to $\nabla\Phi$ are called gradient lines.

These characteristics are used in the generation of the structured grid.

First, the domain is split into 2 blocks by defining a line connecting the CUSPs T and B (Fig. 1(a)). We call this line the division line. The middle part of the division line follows a potential line $\Phi = \Phi_s$. Φ_s is thus chosen as to avoid jumps in cell volume in the final grid. The upper and lower parts of the division line are transitional lines, where the potential Φ changes smoothly from Φ_s to the CUSP values.

A structured grid can now be constructed in each block using potential lines and gradient lines of the solution. The basic structured grid is defined as follows:

- Nodes on the outer wall are equidistant.
- Nodes on the division line are equidistant.
- Gradient lines are lines of constant tangential index (radial lines).
- Nodes on the radial lines are equidistantly distributed.
- Connecting nodes with constant radial index creates tangential lines.

With constant potential boundary conditions, this would mean that the grid lines are potential and gradient lines (in accordance with Winslow's method [3]). Such a grid obtains high quality almost everywhere, except in the vicinity of the CUSPs T and B (Fig. 2). This is remedied by applying appropriate non-constant boundary conditions varying linearly between CUSPs T and B [7].

The final grid quality is strongly determined by the spacing of the radial lines and the tangential lines. If flow calculations are performed on a grid with neighboring cells with strongly differing cell volumes, the accuracy of the solution and even the convergence of the problem are impaired. Such cells can be found mainly around the upper and the lower part of the division line (around the CUSPs T and B). In order to improve grid quality, the originally equidistant spacing of nodes on

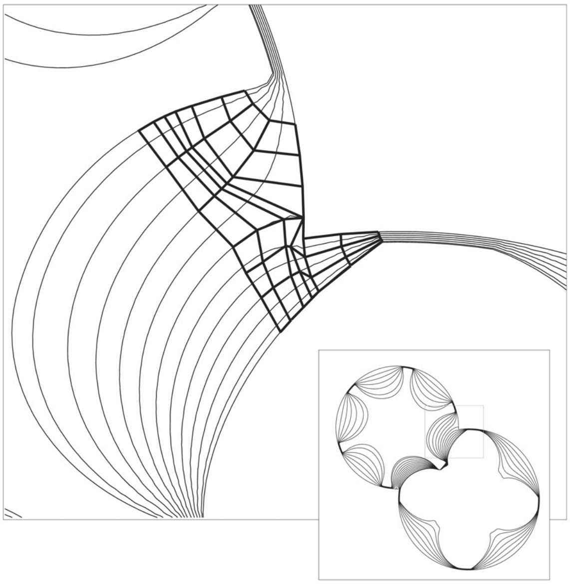

Fig. 2. Poor quality grid around CUSP T due to constant potential boundary condition.

radial lines is abandoned, except for the nodes associated with the middle part of the division line.

For nodes associated with the upper and lower transitional parts of the division line, the equidistant radial spacing is abandoned in favor of spacing according to a geometrical series with factor q_i. The factor q_i is thus determined that the radial sides of the two cells bordering the division line are equal in length. The nodes on radial lines attached to nodes on the outer wall are also redistributed according to a geometric series, whereby the factor q_i evolves linearly from q_T and q_B to 1 (equidistant) in the points L and R (Fig. 1(a)) determined by the position of the rotor tips.

The grid obtained so far is of good quality throughout the domain, and especially in the gaps between rotors and between rotors and casing. Only in the vicinity of the CUSPs T and B does the quality sometimes deteriorate.

For certain positions, the angle between the division line and the casing can be sharp. This sharp angle is propagated in the structured grid. The result is the emergence of very small cells and cells that are at a sharp angle to each other.

If the aspect ratio is too large, tangential smoothing is applied to a small region of the grid around the CUSP. In this region the nodes are repositioned through interpolation on the tangential lines. If necessary, radial smoothing is applied to the nodes on the radial line attached to the CUSP T or B to avoid sharp angles in cells.

With the algorithm described, high-quality 2D block-structured grids can be generated (Fig. 1(d)). To use these grids in a calculation, several 2D slices are connected to form a 3D grid (Fig. 3). To allow freedom in the construction of the 3D grid, and in particular the axial spacing of it, the 2D slices used can be constructed for any time step by interpolating between two generated 2D grids. If the difference between the two generated grids is small (typically 1 degree rotor rotation), it is safe to assume little degradation in quality of the interpolated grids.

A rotation of the compressor rotors is in this way translated in a rotation of the screw profile in the 2D

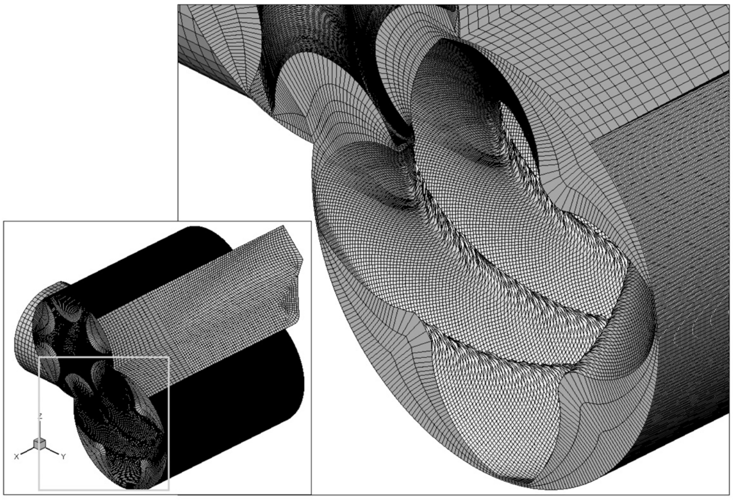

Fig. 3. The 3D grid used for the flow calculation.

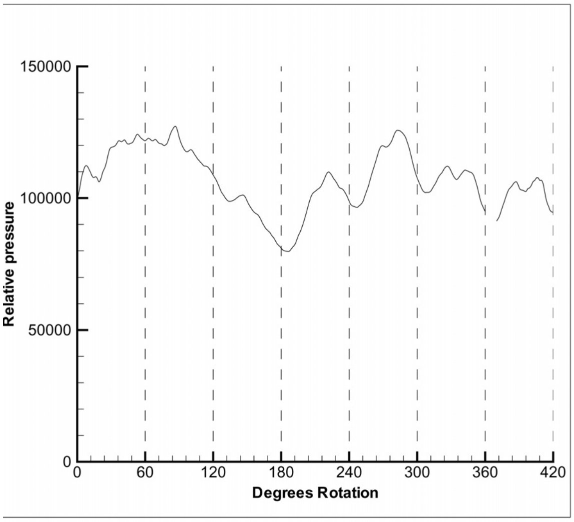

Fig. 4. Relative pressure vs degrees rotation.

slices. Through interpolation between generated grids, the 3D grid can be moved for each time step while maintaining grid topology.

3. Flow calculations in a screw compressor

The actual flow calculations are performed by a commercial package capable of performing ALE calculations. An interface is constructed that moves the grid for each time step, based on the grids supplied by the grid generator.

For the illustration of the grid manipulation algorithm, a screw compressor is chosen (Fig. 3). The structured grid in the casing of the compressor exists of 2 blocks each with 360 cells tangentially, 3 cells radially and 399 cells axially. The total number of cells (including inlet and outlet port) is 973,252. A grid with 3 cells radially will capture all physical phenomena correctly, except fluid–wall friction, as the boundary layer is ill-captured. The losses from friction are small versus the leakage flows, but the reduction in grid size is significant.

As compressed fluid, air is used. The compression is governed by the ideal-gas law. All walls are considered adiabatic. The rotational speed of the rotors is 15750 rpm left and 10500 rpm right. At inlet and outlet, a pressure boundary condition of 1 bar absolute is applied. Due to this boundary condition, over-compression will occur (the pressure rise from the internal compression is up to 2.3 bar absolute) and high-pressure air will be purged into the outlet. Figure 4 shows a plot of the purge pressure during start-up until periodic operation.

There are no experimental data available for the simulated compressor. The results shown here are only meant as an illustration of the capacity of the grid manipulation algorithm.

4. Conclusion

A grid manipulation algorithm is presented, which, linked to a commercial flow solving package, is capable of performing Arbitrary Lagrangian–Eulerian calculations on block-structured grids.

The structured grid is constructed based on the solution of a potential equation with appropriate boundary conditions. This solution is explicitly obtained in the flow domain on an unstructured grid. This allows for extended control over the grid generation.

With this method, it is possible to perform ALE calculations in geometries that have very complex moving boundaries. To illustrate the possibilities of the grid manipulator, an ALE calculation was presented in a screw compressor.

Acknowledgment

The research reported here was funded with a fellowship granted by the Flemish Institute for the Promotion of Scientific and Technological Research in Industry (IWT).

References

[1] Hirt CW, Amsden AA, Cook JL. An Arbitrary Lagrangian–Eulerian computing method for all flow speeds. Int J Comput Phys 1974;14:227.

[2] Farhat C, Geuzaine P, Grandmont C. The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids. Int J Comput Phys 2001;174:669–694.

[3] Winslow A. Numerical solution of the quasi-linear Poisson equation in a nonuniform triangle mesh. J Comput Phys 1967;1:149–172.

[4] Brackbill JU, Saltzman JS. Adaptive zoning for singular problems in two dimensions. J Comput Phys 1982;46: 342–368.

[5] Thompson JF, Thames FC, Mastin CW. Automatic numerical generation of body-fitted coordinate systems for fields containing any number of arbitrary two-dimensional bodies. J Comput Phys 1974;15:299–319.

[6] Riemslagh K, Vierendeels J, Dick E. Two-dimensional incompressible Navier-Stokes calculations in complex-shaped moving domains. J Eng Math 1998;34:57–73.

[7] Vande Voorde J, Vierendeels J, Dick E. Development of a Laplacian-based mesh generator for ALE calculations in rotary volumetric pumps and compressors. J Comput Methods Appl Mech Eng 2004;193:4401–4415.