

THE FREE-ALE METHOD FOR FLOW IN DEFORMING GEOMETRIES

Chris Were and Gordon Mallinson
Department of Mechanical Engineering
University of Auckland
New Zealand

ABSTRACT

A technique is being developed for the modelling of incompressible unsteady fluid flow within grossly deforming geometries, using Voronoi finite volumes. The method is second order accurate in space, first order in time, and is based on an explicit marching technique. A major goal of this research is to model blood flow within Ventricular Assist Devices (VAD's). This paper will briefly describe the motivation for the development of the method, the method itself, and present some initial results. These results are for the Von Karman vortex street behind a 2D cylinder, a 2D pumping chamber, and flow through a stationary 3D VAD.

INTRODUCTION

The technique is designed for use with grossly deforming and/or complex geometries. Many applications exist in industry and science where the domain shape is changing, due either to physical processes (such as pumps, free surface flows, freezing/melting problems) or chemical ones (reactive fronts).

The motivation to develop the method was provided by a desire to model blood flow within Ventricular Assist Devices (VADs).

VAD's are mechanical pumps used to temporarily bypass and assume the functionality of a heart ventricle while, for instance, a donor heart is located. Problems exist with the bio-compatibility of the VADs, particularly with mechanical damage to blood cells near the valves, but also with cell damage due to the flow patterns and shear rates within the VAD itself. The Spiral Vortex (SV) VAD is currently being developed at the University of New South Wales[1] in an attempt to provide a smoother blood flow, and thus reduce cell damage.

Due to the high deformation of the VAD boundary geometry and consequent extreme mesh distortion, traditional finite volume/finite element methods have considerable difficulty maintaining accuracy and stability. It is thus a problem well suited to the new technique.

THEORY

Navier Stokes Equations and their Discretisation

We assume that the blood flow is incompressible and Newtonian, and thus the non-dimensional governing equations are :

$$\mathbf{u}_t + \nabla(\mathbf{u}\mathbf{u}^T) + \nabla p - \frac{1}{Re} \nabla^2 \mathbf{u} = \mathbf{0} \quad (1a)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (1b)$$

Delaunay Triangulations and Voronoi Diagrams

A Delaunay triangulation is a particular triangulation of a set of points (seed nodes) in n dimensions. In 3D it may be considered a "tetrahedralisation". The triangulation has certain properties which have made it popular with finite element modellers. In this paper we are less concerned with the triangulation itself, and more concerned with a dual construction, the Voronoi diagram. The Voronoi, or "Thiessen", regions will be used as the finite volumes.

Figure 1 below shows the relationship between the two dual structures in the two dimensional case. By definition the circumcircle of a triangle within a Delaunay Triangulation contains no other nodes. This property extends to three dimensions by considering "circumballs", and can be used to generate the Delaunay triangulation.

- Seed Nodes
- Voronoi Edges
- Delaunay Edges
- Circumcircle

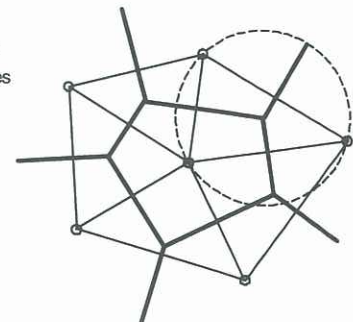


FIGURE 1 VORONOI-DELAUNAY DUALITY

The Voronoi regions have several properties which make them suitable candidates for finite volumes:

- The Voronoi regions represent the loci of points in space nearer to a particular seed node than to any other.
- The faces of the Voronoi regions perpendicularly bisect the edges of the Delaunay triangulation and thus the lines linking adjacent nodes.
- The Voronoi regions are always convex.

As the geometry deforms, the seed nodes move in relation to each other and the connectivity of the Delaunay triangulation changes. That is, nodes which were linked separate and vice versa.

Discrete Approximations on Voronoi Meshes

Before describing the actual flow solution algorithm it will be useful to define the discrete cell centred approximations to the differential operators of Divergence ($\mathbf{D} \approx \nabla \cdot$), Gradient ($\mathbf{G} \approx \nabla$) and Laplacian ($\mathbf{L} \approx \nabla^2$). These discrete operators have the form of matrices:

$$\mathbf{D}(\mathbf{u}_i) \equiv \left(\sum_{neibs} \mathbf{u}_{face} \cdot \mathbf{a}_{face} \right) / Vol_{cell} \approx \nabla \cdot \mathbf{u} \quad (2a)$$

$$\mathbf{G}(\phi_i) \equiv \left(\sum_{neibs} \phi_{face} \mathbf{a}_{face} \right) / Vol_{cell} \approx \nabla \phi \quad (2b)$$

$$\mathbf{L}(\phi_i) \equiv \left(\sum_{neibs} \frac{\phi_{neib} - \phi_i}{d_{i_neib}} \mathbf{a}_{face} \right) / Vol_{cell} \approx \nabla^2(\phi) \quad (2c)$$

The summations in the above expressions all refer to summing over the faces (and hence neighbours) of the Voronoi cell i . The "face" subscripted quantities required in the above need not be explicitly stored, but can be simply interpolated as the mean of the value at node i and its corresponding neighbour, $neib$. Also, \mathbf{a}_{face} represents the outward directed area vector of the face, and d_{i_neib} is the distance from node i to node $neib$. In the code integral forms (without the division by cell volume) are used for efficiency.

Flow Solution Algorithm

The Navier Stokes equations are solved in their primitive variables form, using an explicit predictor-corrector method, loosely based on the MAC technique[2]. All variables (velocity components, pressure) are collocated at the Voronoi seed nodes. A secondary "flux" field is also stored on the Voronoi faces.

The algorithm will be briefly outlined for the static mesh case, followed by the modifications necessary to accommodate mesh motion.

Each time step begins by explicitly "marching" a discrete approximation to (1a), ignoring for the moment the effect of the pressure gradient:

$$\mathbf{u}'_i = \mathbf{u}_i^n - \Delta t \left[\langle \text{Advection} \rangle - \frac{1}{Re} \mathbf{L}(\mathbf{u}_i^n) \right] \quad (3)$$

The term $\langle \text{Advection} \rangle$ refers to the discrete approximation to the advective flux of momentum. The exact form of this is an active area of research, and a detailed description of the form of this discretisation is beyond the scope of this paper. It will suffice to say that the current approximation is based around the well

known, 1st order "hybrid" scheme, and that a second order non-diffusive scheme[3] is being developed.

The resulting \mathbf{u}'_i are a non-solenoidal approximation to the new time step velocity field, and must be corrected to satisfy (1b) in some approximate sense. This is done by generating a pressure field, so that when the effect of this pressure field is applied to the \mathbf{u}'_i the resulting \mathbf{u}_i^{n+1} are approximately solenoidal, in a sense described below. This is known as a Pressure Poisson Equation (PPE) method, and is closely related to the Projection method[4]. The discrete equation for the pressure is given by:

$$\mathbf{L}(p_i) = \frac{1}{\Delta t} \mathbf{D}(\mathbf{u}'_i) \quad (4)$$

The linear system of equations for p is symmetric, and positive definite, and thus amenable to efficient iterative solution methods. The corrected velocities are given by:

$$\mathbf{u}_i^{n+1} = \mathbf{u}'_i - \Delta t \mathbf{G}(p_i) \quad (5)$$

The resulting velocities are approximately solenoidal only, that is $\mathbf{D}(\mathbf{u}_i^{n+1}) \neq 0$. However a set of fluxes may be defined at the Voronoi faces which are exactly solenoidal in the sense of (2a). These fluxes are used in the predictor phase of the next time step when computing the discrete approximation to the advection terms and play a similar role[5] to the staggering of velocities and pressures in the original MAC method. The solenoidal flux through $face$ is given by:

$$flux_{face} = \mathbf{u}'_{face} \cdot \mathbf{a}_{face} - \Delta t \left[\mathbf{a}_{face} \cdot \frac{p_{neib} - p_i}{d_{i_neib}} \right] \quad (6)$$

Note that this is based on the uncorrected velocities and the new pressures.

The explicit nature of the predictor step requires the Courant-Friedrich-Lewy restriction on the time step; essentially that fluid must not cross more than one cell per step.

Flow Solution with Mesh Motion

Mesh motion is accommodated via an Arbitrary Lagrangian Eulerian[6] (ALE) formulation, which allows arbitrary mesh motion. This involves just two essential modifications to the above description.

- The advection terms are modified, so that the fluxes crossing the Voronoi faces are relative to the faces, rather than relative to a fixed coordinate system.
- Equation (3) is modified to account for cell volume variation:

$$\mathbf{u}'_i = \frac{Vol^n}{Vol^{n+1}} \mathbf{u}_i^n - \Delta t \left[\langle \text{Advection} \rangle - \frac{1}{Re} \mathbf{L}(\mathbf{u}_i^n) \right] \quad (7)$$

This requires the additional storage of the old time step volumes. Face velocities are taken to be the mean of the adjacent nodal mesh velocities.

While the motion of the boundary nodes can usually be specified explicitly, such a specification for the internal nodes is complex and unwieldy. Currently two internal node motion schemes have been successfully implemented. The first involves solving a second Poisson equation at each time step for a potential, with source terms based upon the deviation of the current cell volume from a "desired cell volume". The second scheme is more computationally efficient. It involves

iteratively setting each new node position to the Voronoi-face-area weighted average of the connected neighbour positions.

Mesh Reconnection

The unstructured methodology used here requires a reconnection step when the mesh is deforming. This is the process whereby the integrity of the Voronoi diagram is maintained; it enables the Voronoi cells to freely move in relation to each other, and for mesh connectivity to alter. This is achieved by maintaining a dual Delaunay triangulation, and using efficient local transformation ("flipping") algorithms which avoid the necessity to regenerate the entire mesh.

The flipping procedures used are based on a local circumball test of "Delaunayhood". In 2D, if an edge between 2 triangles is not Delaunay it is swapped[7]. In 3D the situation is more complex[8], with flipping between a conglomerate of 3 tetrahedra joined at an edge, and 2 tetrahedra joined at a face.

RESULTS

The code is being tested and evaluated in both 2D and 3D. Brief results will be given here for 3 examples: 2D Vortex shedding behind a circular cylinder, 2D pumping chamber, and 3D VAD "wash-through".

2D Vortex Shedding from a Circular Cylinder

The classic fluid dynamics problem of the Von Karman vortex street behind a circular cylinder provides an excellent test problem for unsteady CFD codes, allowing assessment of both temporal and spatial accuracy. The case presented here is at a Reynolds number (Re) of 100, based upon the free stream velocity and the cylinder diameter. A portion of the mesh used is shown in Figure 2, and it may be seen that a "structured" region has been used around the cylinder to resolve the boundary layer. This is easily accomplished with a Voronoi mesh simply by specifying a regular distribution of seed nodes.

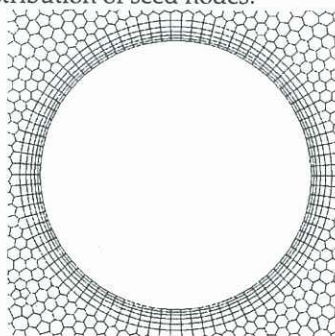


FIGURE 2 MESH IN THE VICINITY OF THE CYLINDER

This simulation used hybrid differencing, which was second order accurate close to the cylinder and in the wake, but only first order further away, where the mesh was coarser. The oscillations were initially excited by rotating the cylinder for the first 20 time steps, to cause a perturbation. The domain was approximately 18 diameters high and wide.

Streaklines generated by the model may be seen in Figure 3, while Figure 4 shows the time history of the total (viscous+pressure) lift and drag coefficients.

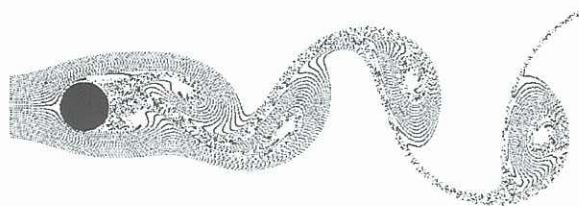


FIGURE 3 STREAKLINES, Re 100

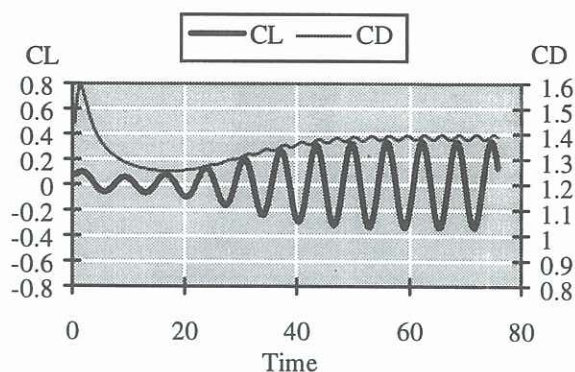


FIGURE 4 Re 100 LIFT AND DRAG

From the above data the Strouhal Number ($St=fD/U_0$, f is the frequency of shedding) was computed as 0.161. This is in good agreement with published data (Kim and Benson[9] give 0.157).

2D Pumping Chamber

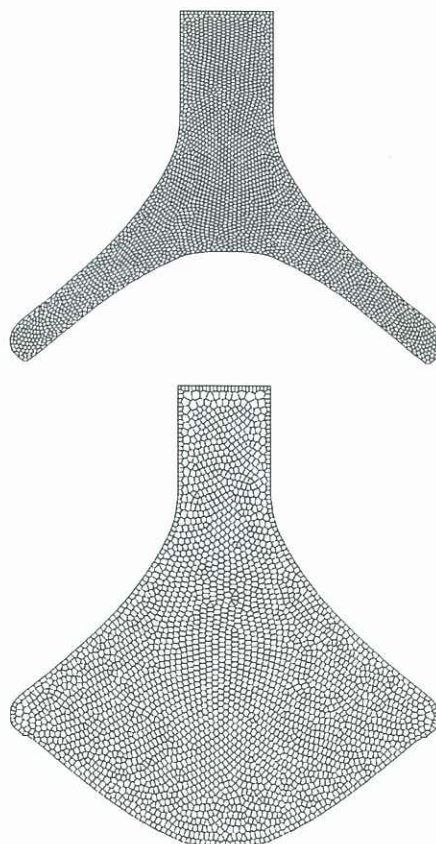


FIGURE 5 2D CHAMBER MESH, COMPRESSED AND EXPANDED

The second example was created mainly as a 2D test of the mesh motion/reconnection algorithms. It is loosely based upon the VAD geometry, but a single inlet/outlet is used for simplicity. The diaphragm moves with a sinusoidal velocity profile. Figure 5 shows the mesh at the two extreme cycles of the pumping cycle ("end-systole" and "end-diastole"). Figure 6 shows velocity vectors during early systole (compression), showing vortices formed from separated flow during diastole.

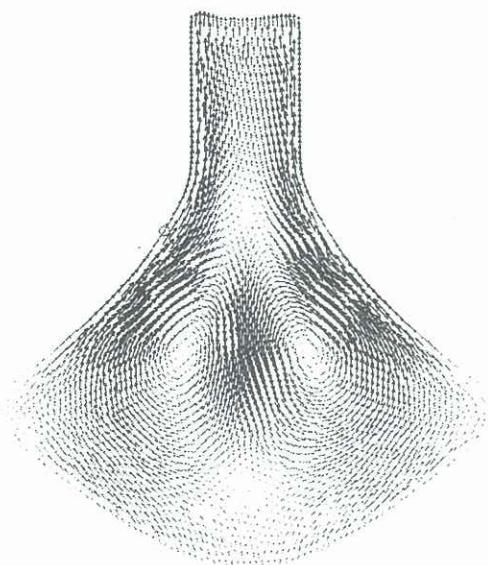


FIGURE 6 VELOCITY VECTORS, EARLY SYSTOLE

3D VAD "Wash Through"

(a) Static VAD Flow:
Streamlines

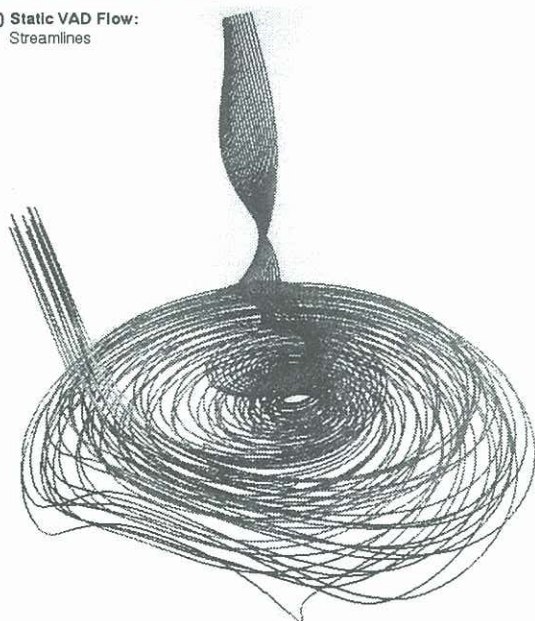


FIGURE 7 3D VAD FLOW: STREAMLINES

The final results presented here are from a preliminary 3D simulation involving the actual SV VAD geometry. For this simulation the diaphragm, and hence mesh, are static, and "blood" is pumped in through the inlet vessel at constant velocity (about 10m/s), to observe the steady state behaviour of the flow. The mesh consists of

approximately 10,000 Voronoi cells. It is expected that larger meshes will be necessary for an accurate simulation.

The streamlines in Figure 7 were tracked using software developed by Dr David Knight, who is now looking at developing a mass conservative tracking algorithm for Voronoi meshes.

CONCLUSIONS

A flexible computational method has been developed in two and three dimensions for modelling fluid flow within grossly deforming geometries. Future work will concentrate on applying the method to the real world problem of the SV-VAD. Improvement of the numerical accuracy, particularly with regard to advection is also a priority. It is hoped eventually to parallelize the code and apply it to the simulation of living human hearts.

ACKNOWLEDGEMENTS

This work has been supported by the NZVCC and Maurice-Paykel Scholarships. Thanks to Prof. Chris Bertram and Allen Nugent of the Graduate School of Biomedical Engineering, UNSW, for assistance with the VAD modelling.

REFERENCES

- [1] Nugent, AH, Bertram CD, Ye C, Umezumi M, *Flow Visualisation in a Cardiac Assist Device*, 11th Australasian Fluid Mechanics Conference, Hobart, 1992.
- [2] Harlow FH and Welch JE, *Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with a Free-Surface*, Physics of Fluids, Vol 8 No 12, pp 2182-2189, 1965.
- [3] Colella P, *Multidimensional upwind Methods for Hyperbolic Conservation Laws*, J. Comp. Phys., Vol 87, pp171-200, 1990.
- [4] Gresho PM, *Some Current CFD Issues Relevant to the Incompressible Navier-Stokes Equations*, Comp Meth Appl Mech Eng, Vol 87, pp 201-252, 1991.
- [5] Peric M, Kessler R and Scheurer G, *Comparison of Finite-Volume Numerical Methods with Staggered and Colocated Grids*, Computers & Fluids, Vol 16, No 4 pp389-403, 1988.
- [6] Hughes TJR, Liu WK, and Zimmermann TK, *Lagrangian-Eulerian Finite Element Formulation for Incompressible Viscous flows*, Comp Meth Appl Mech Eng, Vol 29, pp 329-349, 1981.
- [7] Lawson CL, *Software for C¹ Surface Interpolation*, Mathematical Software III, JR Rice ed., Academic Press NY, pp161-194, 1977.
- [8] Joe B, *Three-Dimensional Triangulations from Local Transformations*, SIAM J Sci Stat Comput, Vol 10 No 4, pp 718-741, 1989.
- [9] Kim SW and Benson TJ, *Comparison of the SMAC, PISO and Iterative Time-Advancing Schemes for Unsteady Flows*, Comp Fluids, Vol 21 No 3, pp 435-454, 1992.