18th World IMACS / MODSIM Congress, Cairns, Australia, 13-17 July 2009
http://msanz.org.au/modsim09

Boundary-Layer Hydrodynamics using Mesh-Free Modelling

J. S. Kapor[1], A. D. Lucey[1] and M. W. Pitman[1]

[1] *Fluid Dynamics Research Group, Curtin University of Technology,*
GPO Box 1987, Perth, WA 6845
Email: jarrad.kapor@postgrad.curtin.edu.au

Abstract: A significant component of drag on vehicles, especially ships, is due to skin friction. This results from the action of the viscous boundary layer between the principally inviscid outer mean flow and the shell of the vehicle. Numerical models that accurately predict boundary-layer dynamics can give new insights that will allow the design and/or critical evaluation of drag-reducing strategies such as the use compliant coatings, synthetic jets and viscosity modification. Of particular importance are the mechanisms that cause laminar-to-turbulent transition of the boundary-layer and thereby lead to a marked increase in skin-friction drag.

Boundary-layer modelling using numerical schemes is particularly challenging because the ratio of orthogonal length scales over which dynamical changes occur is extreme; the boundary-layer is very thin as compared with its stream-wise extent. Accordingly, if using conventional finite-volume or finite-element methods, over-resolution in the stream-wise direction is needed to maintain practicable element aspect ratios when mesh-based methods are used. This results in high computational costs that are then usually offset by inadequate resolution in the normal direction and thus a failure to capture properly the dynamics of disturbances within the boundary layer. The challenge for mesh-based methods is compounded when the wall bounding the flow is allowed to deform as is the case for compliant coatings. In such systems it is necessary to re-mesh at every step in the evolution of the walls geometry and correctly transfer nodal variables between old and new meshes.

Presented is a new mesh-free Lagrangian scheme that combines discrete-vortex and boundary-element methods to model a boundary layer over a surface that can be of arbitrary shape and at which non-standard boundary conditions can be applied. Surface modelling is achieved using the boundary-element method while discrete vortices are used within the boundary layer. Multiple layers of these vorticity elements are deployed in order to capture accurately, for example, Tollmien-Schlichting wave-like, responses to some form of external excitation. In our work, the well-known disadvantage of particle methods, namely the need to solve an NxN system, is overcome by the implementation of a novel Fast Multi-pole Method (FMM) that is extremely efficient.

To show relative speed-ups the FMM has been compared to the standard calculation algorithm that is used in the FMM's direct vortex-vortex interactions. Figure 1 shows both algorithms in single and multi(4)-threaded modes using randomly distributed point vortices on a standard shared memory desk-top computer (Intel Q9650 3.0GHz processor). For this case the FMM employed a $p = 13$ truncation value which offers a precision of the order of 10^{-4}. The FMM shows superior scalability over the direct method, with the velocity influence calculations of $\approx 65,000$ particles requiring less than 1 s using the multi-threaded FMM. The crossover where the direct method is faster occurs at an untested point of less than 500 particles, however with 1 s of computational effort the direct method can only calculate interactions $\approx 6,000$ particles.

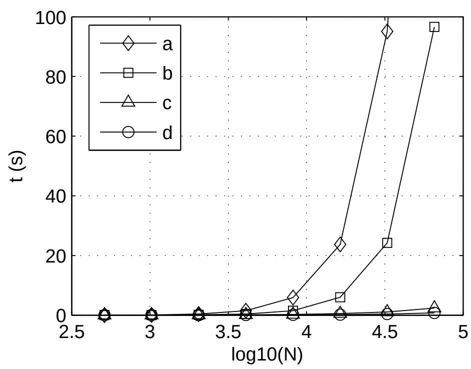


Figure 1: Variation of time to calculate particle interactions with problem size using: (a) single-threaded direct method, (b) multi(4)-threaded direct method, (c) single-threaded FMM, (d) multi(4)-threaded FMM.

Keywords: *Fast Multipole Method, Discrete Vortex Method, Boundary Element Method, Boundary Layer*

1. INTRODUCTION AND BACKGROUND

The Discrete Vortex Method (DVM) is a numerical modelling scheme that is essentially grid free and based on using the interactions of discrete vortices to approximate solutions to Euler's equations. The advantages of the DVM are its grid free nature and efficiency when compared to grid based methods. The foundations of the DVM were laid out in 1858 by Helmholtz who was the first to show that in an inviscid fluid, the vortex lines consisted of the same fluid elements and that flows with vorticity could be modelled with an approximate circulation and infinitely small cross section. However the first serious attempt at vortex modelling is often cited as being the work by Rosenhead (1931) who studied the Kelvin-Helmholtz instabilities of vortex sheets. Rosenhead used a distribution of finite elemental vortices along the length of the vortex sheet and allowed the movements of the vortices to be determined by numerical time stepping. Since then much progress has been made using discrete vortex methods with excellent review articles being published by Clements & Maull (1975), Saffman & Baker (1979), Leonard (1980, 1985), Aref (1983) and Sarpkaya (1989). What is evidenced by these reviews is that the DVM lends itself very well to modelling applications in aerodynamics, turbulent mixing, combustion and of particular interest in this study, for boundary-layers.

While Lagrangian schemes such as the DVM have the advantage of being essentially grid-free they have the generic disadvantage of the "N-Body" problem. The simplest evaluation of any N-body problem requires calculations in the order of $O(N^2)$, i.e. the square of the number of particles present. This property of the DVM and other N-Body schemes means that increasing the resolution in a system can quickly become prohibitively expensive. Early solutions to this problem for the DVM came in the form of vortex-in-cell methods first presented by Christiansen (1973) with later work from others such as Baker (1979).

As an N-body problem, the DVM is similar to other areas in classical physics such as celestial mechanics involving point masses and plasma physics involving point charges. Consequent demand for a fast and efficient method for solving N-Body problems resulted in the development of various tree-base algorithms and solvers. Early techniques included monopole calculations and divide-and-conquer strategies such as that developed by Barnes & Hut (1986) and Appel (1985). Shortly after, Greengard & Rokhlin (1987*a*) developed a similar algorithm titled the Fast Multipole Method (FMM) that has since been cited as one of the top ten algorithms of the 20th century (Dongarra & Sullivan 2000).

The basis of the FMM is the conversion of the $\log z$ potential function into an equivalent infinite Taylor series. For "well separated" particles and within reasonable error bounds, the Taylor series of relatively close particles can be summed and shifted to a collocation point. As the truncation of an infinite Taylor series in itself can be cumbersome, Greengard proposed using the natural hierarchy of a Quad tree to calculate these far field interactions. Lastly, any particle interactions that could not be refined down to a resolution such that it could be classified as far field, the conventional $O(N^2)$ methods would be applied. This resulted in an algorithm of $O(N \log N)$.

While the FMM was very popular and its use widespread in classical particle physics, its published adoption in fluid mechanics has thus far been limited. Work by Pringle (1994) involved applying the two-dimensional FMM to the DVM, introducing optimisations such as the "Dynamic-P" principles and investigating its use for parallel computation. Hamilton & Majda (1995) also applied the FMM to the DVM using vortex "blobs" and analysed its accuracy and efficiency concluding that care must be taken to ensure errors are not of significant magnitude.

Shortly after publishing their 2-D FMM algorithm, Greengard & Rokhlin (1987*b*) published a FMM algorithm for 3-D problems. However, its complexity and the lack of computing power at the time resulted in its practical adoption being limited. Cheng et al. (1999) later published optimisations to their algorithm using new mathematical compression techniques that achieved considerable speed-ups for little cost in accuracy. An example of its adoption for the purpose of the DVM was provided by Lindsay & Krasny (2001) who used the 3-D FMM to simulate the roll-up of circular-disk vortex sheets into vortex rings.

This paper aims to show that high resolution 2D-DVM simulations are now possible on a common desk-top computer using the FMM, and this combination of FMM and DVM can be useful for boundary layer modelling.

2. DVM FORMULATION

The formulation of the DVM begins with the Euler's equations for inviscid, incompressible, unforced, two-dimensional flow given in Equation 1. This is coupled with the two-dimensional mass conservation equation given in Equation 2.

$$\dot{\vec{u}} + (\vec{u} \cdot \nabla)\,\vec{u} = -\frac{1}{\rho}\nabla p \tag{1}$$

$$\nabla \cdot \vec{u} = 0 \tag{2}$$

By taking the curl of these equations we are left with,

$$\frac{\partial \omega}{\partial t} + (\vec{u} \cdot \nabla)\,\omega = 0 \tag{3}$$

and by tracking the vorticity in a Lagrangian reference frame we have,

$$\frac{D\omega}{Dt} = 0 \tag{4}$$

These results show that vorticity is always conserved along particle trajectories and that numerically we can define a system by a series of discrete packets of constant vorticity. Calculation of the velocity field in any discrete vortex system is a result of the $O(N^2)$ summation of the influences of all other vortices. This is a significant problem as increased model resolution requires more discrete vortices.

When a boundary is present, such as the wall-bounded boundary-layer model in Section 4.2, the no-flux condition must be satisfied. Instead of using image vortices, we enforce no-flux using a Boundary-Element Method (BEM), an approach that is more versatile (see Lewis (1991) and Katz & Plotkin (1991)). Unlike vortex-vortex interactions, calculating the influence of M boundary elements on the N free vortices is of the order $N \times M$. However due to the relatively complex nature of the typical panel equations the panel-vortex computations can become significant. To reduce this impact we can approximate the panel elements to an acceptable tolerance in the far field as simple point elements. The benefit of this is that the FMM now can be employed for these far-field interactions, while the standard direct calculation is applied to locations that fall in the near-field region of the panel.

3. THE FAST MULTIPOLE ALGORITHM

The full FMM algorithm is a process which is best understood graphically or through visualisations such as those produced by Greengard & Rokhlin (1987*b*), Pringle (1994) or Wang (2005). A very brief summary of the process is included here. However additional texts (Greengard & Rokhlin 1987*b*) should be consulted for its theoretical formulation and proof.

The basic premise behind the FMM as discussed by Greengard & Rokhlin (1987*b*) is to use multi-pole expansions to evaluate interactions of "clusters" (boxes) of particles at well separated distances. The FMM broadly involves three steps - the tree-building, the upward pass and the downward pass. In the tree-building step a suitably dimensioned quad tree is constructed such that it encapsulates the parent-child and neighbouring relations of all boxes at each level. The tree's lowest level contains the individual source and target particles in question where the multi-pole expansions are evaluated to and from. Next the upward-pass begins at the lowest level of the tree where the "influence" of the all m particles of individual strength and position q_i and z_i in a box are converted into a multi-pole about the box centre by the equation,

$$\phi(z) = Q \log z + \sum_{k=1}^{\infty} \frac{a_k}{z^k} \tag{5}$$

where,

$$Q = \sum_{i=1}^{m} q_i, \quad a_k = \sum_{i=1}^{m} \frac{-q_i z_i^k}{k} \tag{6}$$

The remaining process of the FMM is as follows:

1. The lowest level multi-pole expansions are translated up through the parents to the top of the tree.
2. Next in the downward pass, the upward expansions are translated onto well separated nearest neighbours and translated down the tree through children to the lowest target boxes.
3. At the lowest boxes the translated expansions are evaluated back onto the real targets.
4. Any local and neighbouring source particles that have not had their influence counted in a previous step are dealt with using direct methods.

In the process, a neighbouring box does not classify as well separated unless its centre is located a distance $3r$ away from the current box encapsulated in radius r. This condition ensures that the error on the translation process is bound to 2^{-p} where p is an arbitrary value to which the infinite-series is truncated.

The specific FMM algorithm applied in this paper employs the Dynamic-P optimisations shown by Pringle (1994) and is multi-levelled to ensure memory and computational efficiency. Additionally it has been designed to deal with Gaussian vortices, similar to those presented by Hamilton & Majda (1995). The code is fully implemented in C++ and is built using the C++ Standard Template Libraries (STLs) to store efficiently the adaptive-tree data structures. Due to the modern desktop computers advancing along the multi-cored, shared memory path, the code was also multi-threaded using C-Pthreads.

4. ILLUSTRATIVE EXAMPLES

To test our FMM algorithm and its effect on the usefulness of the DVM, we present two illustrative cases.

4.1 Model A - Heaving Plate

This model focuses on an infinitely thin plate, heaving, in the presence of a high Reynolds (inviscid) uniform flow. The model is constructed as shown in Figure 2. To model the free field, discrete vortices with Gaussian cores are shed from the trailing edge of the plate and are allowed to convect naturally in the domain. To model the heaving plate a series of first-order vortex panels are used to enforce the no-flux boundary condition with newly shed discrete vortices of sufficient strength to ensure vorticity is conserved. For simplicity a first order Eulerian time-stepping scheme is employed.

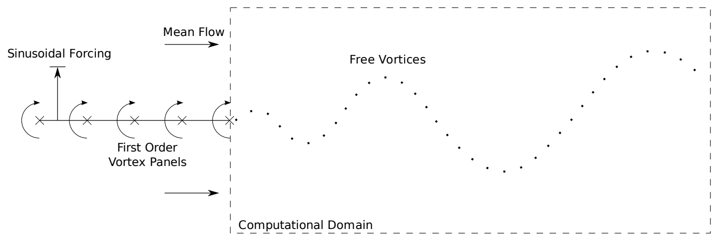


Figure 2: Schematic of the heaving plate model.

Shown in Figure 3 are the results of the heaving plate simulation. For all cases the forcing amplitude $A_f = 0.01\mathrm{m}$ and Gaussian core-size is $\sigma = 0.005$. All FMM cases used $dt = 2 \times 10^{-4}$ s and $N = 12,500$ free vortices, with the direct method cases using $dt = 1 \times 10^{-3}$ s and $N = 2,500$. The parameters chosen require similar computational effort (≈ 0.3 s per time-step) showing that the FMM allows the model to be run with much higher resolution. This is essential to capture the behaviour of the model at higher forcing amplitudes and frequencies.

4.2 Model B - Inviscid Boundary Layer

This model, as shown in Figure 4, involves stacking many discretised shear layers on top of one another to model the effects of a quasi-boundary-layer flow bounded by a rigid wall. Each shear layer is modelled by three components. In the flow field there are discrete vortices with Gaussian cores that are free to convect

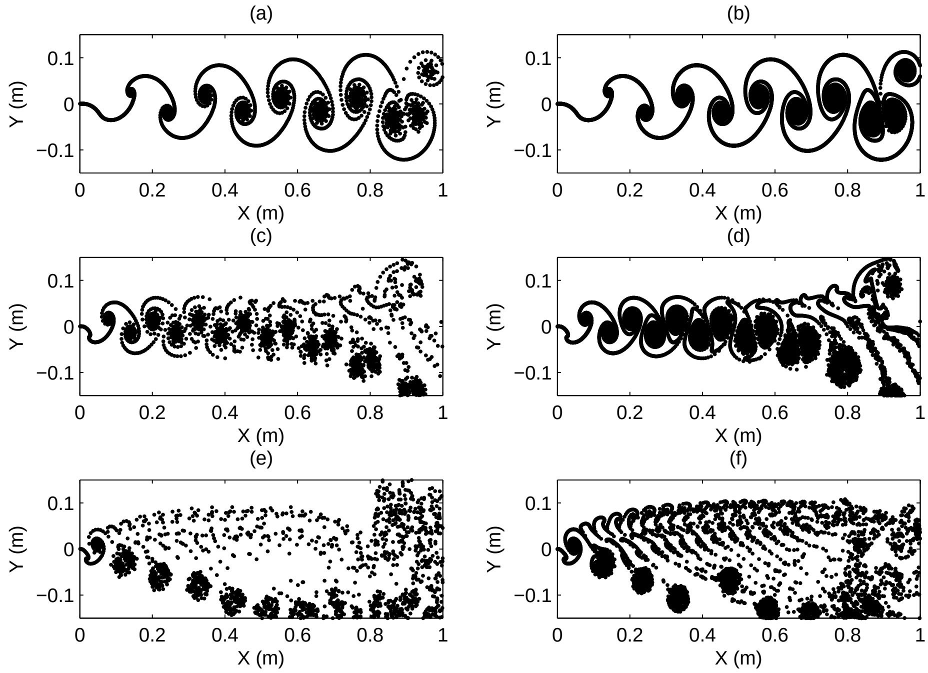


Figure 3: Results of heaving plate model for: (a) direct method, $f_f = 1$Hz, (b) FMM, $f_f = 1$Hz, (c) direct method, $f_f = 2$Hz, (d) FMM, $f_f = 2$Hz, (e) direct method, $f_f = 4$Hz, (f) FMM, $f_f = 4$Hz.

according to the resulting flow field. Vortex injection occurs at the leading edge of the domain to conserve vorticity and vortex removal occurs at the trailing edge of the domain. Semi-infinite vortex sheets positioned on the outside of the domain account for far-field effects. A small section of discrete vortices are fixed in place in the region of the leading edge of the domain to aid in the transition from the semi-infinite vortex sheets to the free vortices.

When stacking each layer the boundary-layer velocity profile is achieved by assigning each shear-layer with a predefined strength and vertical position to approximate the velocity profile defined by the Polhausen boundary layer velocity profile. To model the rigid boundary of the lower wall, zero-order source/sink panel elements are used that strictly enforce the no-flux condition.

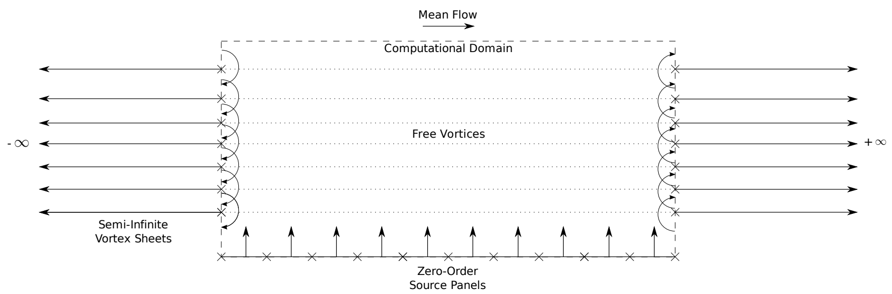
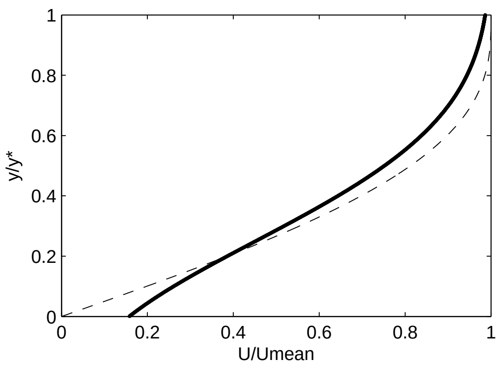


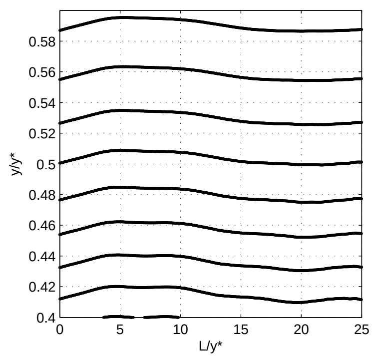
Figure 4: Schematic of the inviscid boundary-layer model.

To test the model 40 shear layers were employed with a total 64,000 free Gaussian vortices that have a core size of $\sigma = 9 \times 10^{-4}$. The layout at time $t = 0$ s generates a boundary layer velocity profile as shown in Figure 5(a). It is noted that the velocity profile deviates away from the theoretical Polhausen profile due to the use of the discrete Gaussian vortices. By reducing the core-size σ the profile would offer a better approximation, however, this would cause noisy results due to the numerically unstable nature of point-vortices.

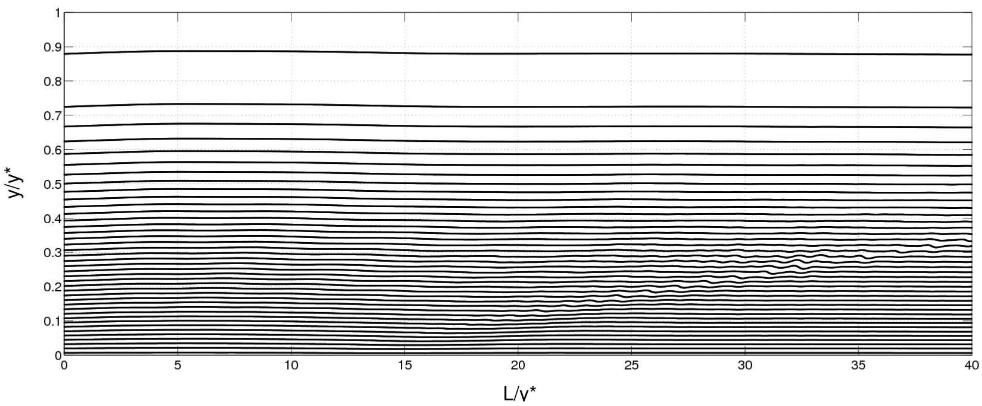
Figure 5(c) shows the $x-y$ positions of the free vortices at time $t = 14$ s responding to a very small sinusoidal, hyperbolic perturbation to the vortex injection positions at the leading edge. The zoomed results, as shown in Figure 5(b), highlight the presence of a Tollmien-Schlichting wave-like response propagating downstream. The small-scale waves shown near the trailing edge of the domain in Figure 5(c) are examples numerical noise that can be reduced by using larger Gaussian cores. In this case, with 64,000 free particles and 6,400 boundary elements, the FMM allows a full time-step to be completed in ≈ 13 s where using the direct calculation methods takes up 190 s.



(a) Velocity Profile



(b) Zoomed



(c) Full Domain

Figure 5: (a) Velocity profiles generated for the inviscid boundary layer model for the theoretical Polhausen profile (dashed line) and generated profile from model (solid line). {(b), (c)} Snapshots of x-y positions of the free vortices after running the inviscid boundary layer model. Note that a non-square aspect ratio has been used for these figures.

5. CONCLUSIONS

The Discrete Vortex Method (DVM) as a Lagrangian technique offers a promising alternative to Eulerian grid-based techniques, especially for flows involving complex boundary conditions. However its wide adoption has largely been hampered by the poor scaling associated with increasing resolution of models. Fast particle algorithms, such as the Fast Multipole Method (FMM) utilised here allow these disadvantages to be overcome. With additional algorithm improvements such as multi-leveling and dynamic-p optimisations the DVM and Boundary Element Method (BEM) combination have much potential in future for complicated fluid-structure interaction problems.

Shown in our results is the ability of the FMM to synergise the BEM and DVM combination into a powerful tool that can be used on a modern desktop computer to run inviscid DVM simulations previously practicable only on super-computers of the recent past. Achieved at the cost of a small loss in accuracy, the FMM offers faster simulations at the same or higher level of resolution for the same computational cost as the direct methods.

Future developments of the FMM will provide further improvements to the DVM modelling by assisting in the modelling of viscous effects. The deterministic Corrected-Core-Spreading method involves greatly increased vortex numbers and thus vortex-merging. The FMM will be of great advantage for this modelling due to its inherent requirement of recognising particle near-neighbours, which is a prohibitively expensive component of current core merging algorithms.

ACKNOWLEDGEMENTS

The main author would like to acknowledge the support of the Australian Postgraduate Awards (APA) and the Curtin University Postgraduate Scholarship (CUPS) schemes.

REFERENCES

Appel, A. W. (1985), 'An efficient program for many-body simulation', *SIAM Journal on Scientific and Statistical Computing* **6**(1), 85–103.

Aref, H. (1983), 'Integrable, chaotic and turbulent vortex motion in two dimensional flows', *Annual Review of Fluid Mechanics* **15**, 345–389.

Baker, G. (1979), 'The'cloud-in-cell' technique applied to the roll up of vortex sheets', *Journal of Computational Physics* **31**(1), 76–95.

Barnes, J. & Hut, P. (1986), 'A Hierarchical O(NlogN) Force-Calculation Algorithm', *Nature* **324**, 446–449.

Cheng, H., Rokhlin, V. & Greengard, L. (1999), 'A fast adaptive multipole algorithm in three dimensions', *Journal of Computational Physics* **155**(2), 468–498.

Christiansen, J. P. (1973), 'Numerical simulation of Hydrodynamics by the Method of Point Vortices', *Journal of Computational Physics* **13**, 363–379.

Clements, R. R. & Maull, D. J. (1975), 'The representation of vortex streets by discrete vortices', *Prog. Aerospace Sci.* **16,2**, 129–146.

Dongarra, J. & Sullivan, F. (2000), 'The top 10 algorithms', *Computing in Science & Engineering* **2**(1), 22–23.

Greengard, L. & Rokhlin, V. (1987*a*), 'A Fast Algorithm for Particle Simulations', *Journal of Computational Physics* **73**, 325–348.

Greengard, L. & Rokhlin, V. (1987*b*), 'The rapid evaluation of potential fields in three dimensions', *Lecture Notes in Mathematics* **1360**, 121–41.

Hamilton, J. & Majda, G. (1995), 'On the Rokhlin-Greengard Method with Vortex Blobs for Problems Posed in All Space or Periodic in One Direction', *Journal of Computational Physics* **121**(1), 29–50.

Katz, J. & Plotkin, A. (1991), *Low Speed Aerodynamics: From Wing Theory to Panel Methods*, number 0-07-050446-6, McGraw-Hill.

Leonard, A. (1980), 'Vortex methods for flow simulation', *Journal of Comutational Physics* **37**(3), 289–335.

Leonard, A. (1985), 'Computing three-dimensional incompressible flows with vortex elements', *Annual Review of Fluid Mechanics* **17**, 523–559.

Lewis, R. I. (1991), *Vortex Element Methods for Fluid Dynamic Analysis of Engineering Systems*, Cambridge University Press.

Lindsay, K. & Krasny, R. (2001), 'A particle method and adaptive treecode for vortex sheet motion in three-dimensional flow', *Journal of Computational Physics* **172**(2), 879–907.

Pringle, G. (1994), Numerical Study of Three-Dimensional Flow using Fast Parallel Particle Algorithms, PhD thesis, Ph. d. thesis, Napier University, Edinburgh, Scotland.

Rosenhead, L. (1931), 'The formation of vorticies from a surface of discontinuity', *Proc. Royal Society* **134**, 170–192.

Saffman, P. G. & Baker, G. R. (1979), 'Vortex interactions', *Annual Review of Fluid Mechanics* **11**, 95–122.

Sarpkaya, T. (1989), 'Computational methods with vortices-the 1988 freeman scholar lecture', *Journal of Fluids Engineering* **111**, 5–52.

Wang, Y. (2005), The fast multipole method for two-dimensional coulombic problems: Analysis, implementation and visualization, Master's thesis, University of Maryland, College Park.