

The New Incompressible Flow Capabilities in LS-DYNA

Mark A. Christon

Livermore Software Technology Corp.

7374 Las Positas Road

Livermore, California 94550

(925) 449-2500

christon@lstc.com

Keywords:

Div-Free, Incompressible, Large-Eddy Simulation, Projection

ABSTRACT

Over the past year, efforts have been underway to extend LS-DYNA's extensive list of capabilities to include an incompressible CFD solver. The focus of this paper is on the second-order approximate projection method used in LS-DYNA to solve the time-dependent Navier-Stokes equations. In order to address the computational demands of the implicit pressure field, LS-DYNA relies, in part, upon an A-conjugate projection technique coupled with the preconditioned conjugate gradient method and sub-domain preconditioners. Results of time-dependent laminar and large-eddy simulations are used to illustrate the effectiveness of the projection-based preconditioned conjugate gradient method coupled with the second-order approximate projection flow solver. As a new physics option in LS-DYNA, the incompressible flow solver complements the existing compressible fluid/ALE simulation capabilities.

INTRODUCTION

LS-DYNA provides a comprehensive set of multi-physics simulation capabilities for problems ranging from crashworthiness, occupant safety and metal forming to fluid-structure interaction, heat transfer and now incompressible flow. Incompressible flows are some of the most frequently encountered flow regimes encompassing problems that range from atmospheric dispersal to food processing, aerodynamic design of automobiles, and manufacturing processes such as chemical vapor deposition, mold filling and casting. The need for scalable time-accurate solution algorithms is growing due to emerging applications that are inherently time-dependent, e.g., mold filling, fluid-structure interaction, and internal combustion engine manifolds. The expanding use of large-eddy simulation (LES) and time-dependent Reynolds Averaged Navier-Stokes (RANS) calculations is also driving the need for scalable transient flow solution methods.

The algorithmic challenges involved in solving time-dependent, incompressible flow problems hinge upon the div-free constraint, efficient treatment of the concomitant implicit pressure and scalable, parallel solution algorithms. Over the past 5-10 years, second-order projection methods have emerged as the most computationally efficient algorithms for performing time-accurate incompressible flow simulations. A detailed review of projection methods is beyond the scope of this paper, but a partial list of relevant work is provided. Projection methods, also commonly referred to as fractional-step, pressure correction methods, or Chorin's method [1] have grown in popularity due to the relative ease of implementation and computational performance of these methods. This is reflected by the volume of work published on the development of second-order accurate projection methods, see for example van Kan [2], Bell, et al. [3], Gresho, et al. [4-7], Almgren, et al. [8-11], Rider [12-15], Minion [16], Guermond and Quartapelle [17], Puckett, et al. [18], Sussman, et al. [19], and Knio, et al. [20]. In addition, the numerical performance of projection methods has been investigated by Brown and Minion [21,22], Wetton [23], Guermond [24,25], Guermond and Quartapelle [26,27], and Almgren, et al. [11].

This paper is intended to introduce the projection methods used in LS-DYNA for time-dependent incompressible flow simulations and outline the type of problems that may be addressed using the incompressible flow solver. The ensuing discussion begins with an overview of the underlying finite element formulation and the basic time integration method. A brief summary of the A-conjugate projection technique used for the pressure solution is presented with a series of sample computations.

FORMULATION

A brief review of the incompressible Navier-Stokes equations and the spatial discretization is presented before proceeding with a description of the second-order projection method. To begin, the incompressible Navier-Stokes equations are

$$\nabla \cdot \vec{u} = 0, \text{ and} \quad (1)$$

$$\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} = -\nabla P + \nu \nabla^2 \vec{u} + \vec{f}, \quad (2)$$

where $\vec{u} = (u, v, w)$ is the velocity, ν is the kinematic viscosity, \vec{f} is the body force, ρ is the mass density, $P = p/\rho$, and p is the pressure. Note that the simplified form of the Navier-Stokes equations, i.e., with constant density and viscosity, is used in order to streamline the presentation.

The system of equations above are subject to boundary conditions that consist of specified velocity on Γ_1 as in Eq. (3), or pseudo-traction boundary conditions on Γ_2 as in Eq. (4) and (5).

$$\vec{u} = \vec{u} \text{ on } \Gamma_1 \quad (3)$$

$$-P + \nu \frac{\partial \vec{u}}{\partial n} = f_n \text{ on } \Gamma_2 \quad (4)$$

$$\nu \frac{\partial \vec{u}}{\partial \tau} = f_\tau \text{ on } \Gamma_2 \quad (5)$$

Here, $\Gamma = \Gamma_1 + \Gamma_2$ is the boundary of the domain. Here $\partial \vec{u} / \partial \vec{n}$ and $\partial \vec{u} / \partial \vec{\tau}$ represent the derivative of \vec{u} in the normal (\vec{n}) and tangential ($\vec{\tau}$) directions respectively. Similarly, f_n and f_τ represent the normal and tangential components of the boundary traction. Homogeneous traction boundary conditions correspond to the well-known natural boundary conditions that are typically applied at outflow boundaries.

In addition to the boundary conditions, initial conditions are required.

$$\vec{u}(\vec{x}, 0) = \vec{u}^0(\vec{x}) \quad (6)$$

For a well-posed time-dependent incompressible flow problem, the prescribed initial velocity field in Eq. (6) must satisfy Eq. (7) and (8) below (see Gresho and Sani [28]). If $\Gamma_2 = 0$ (i.e., an enclosure flow with $\vec{u} \cdot \vec{n}$ prescribed on all surfaces), then global mass conservation enters as an additional solvability constraint as shown in Eq. (9).

$$\nabla \cdot \vec{u} = 0 \text{ in } \Omega \quad (7)$$

$$\vec{n} \cdot \vec{u}(\vec{x}, 0) = \vec{n} \cdot \vec{u}^0(\vec{x}) \quad (8)$$

$$\int_{\Gamma} \vec{n} \cdot \vec{u}^0 d\Gamma = 0 \quad (9)$$

Before proceeding with a description of the time integration algorithms, the semi-discrete Navier-Stokes equations are presented. The basic spatial discretization of the conservation equations is achieved using the Q1Q0 element [29] with bilinear support for velocity and piecewise constant support for the pressure in two dimensions. In three dimensions, the velocity support is tri-linear with piecewise constant support for pressure. The methods for obtaining the weak form of the conservation equations are well known and will not be repeated here (see for example, Gresho and Sani [29]).

The spatially discrete forms of Eq. (1) and (2) are

$$C^T \vec{u} = 0, \text{ and} \quad (10)$$

$$M \dot{\vec{u}} + A(\vec{u})\vec{u} + K\vec{u} + CP = \vec{F}, \quad (11)$$

where M is the unit mass matrix, $A(\bar{u})\bar{u}$ and K are the advection and the viscous diffusion operators respectively, and \bar{F} is the body force. C is the gradient operator, and C^T is the divergence operator. In order to simplify the nomenclature in the subsequent discussion, \bar{u} and P are understood to be discrete approximations to the continuous velocity, and pressure.

FEM PROJECTION METHOD

Although fully coupled solution strategies are available, the cost of such methods is currently considered prohibitive for time-accurate three-dimensional simulations – particularly where high-resolution grids are required. The philosophy behind projection algorithms is to achieve a legitimate decoupling of the pressure and velocity fields in the hope of providing an efficient computational method for transient, incompressible flow simulations. Projection methods decouple the solution of the velocity and pressure fields by first computing an intermediate velocity field, and then performing an orthogonal projection to a div-free subspace.

In LS-DYNA, the *optimal* Projection-2 (P2) method identified by Gresho [4,5] provides the basic underlying solution strategy.

Semi-Implicit Projection-2 (P2)

1. Given a div-free velocity, \bar{u}^n , and its corresponding pressure field, P^n , solve for an intermediate velocity field, $\tilde{\bar{u}}$, at time t^{n+1} . Here, the advection is treated explicitly using a centroid advection velocity combined with a predictor-corrector step with an “operator limiting” procedure to preserve monotonicity.

$$[M + \theta\Delta t\hat{K}]\tilde{\bar{u}} = [M - (1-\theta)\Delta t\hat{K}]u^n + \Delta t\{\theta F^{n+1} + (1-\theta)F^n - A(\bar{u})\bar{u}^n - MM_L^{-1}CP^n\} \quad (12)$$

2. Given the approximate velocity, $\tilde{\bar{u}}$, solve a pressure Poisson equation (PPE) for the Lagrange multiplier, λ .

$$[C^T M_L^{-1}C + S]\lambda = C^T \tilde{\bar{u}} - C^T \bar{u}^n \quad (13)$$

3. Perform the projection step to obtain the final div-free velocity field, \bar{u}^{n+1} .

$$\bar{u}^{n+1} = \tilde{\bar{u}} - M_L^{-1}C\lambda \quad (14)$$

4. After the velocity update, a new pressure at time t^{n+1} is obtained via

$$P^{n+1} = P^n + \frac{\alpha}{\Delta t} \lambda. \quad (15)$$

5. Repeat steps 1-4 until a maximum simulation time limit or a maximum number of time steps is reached.

Remarks

1. In Eq. (12), \hat{K} is the viscous operator obtained in the weak formulation and augmented by a balancing tensor diffusivity (BTD) that derives from the second-order, explicit time integrator applied to the advective terms. See Gresho, et al. [30] or Christon [31] for additional details on BTD. The semi-implicit treatment of BTD permits stable computations for CFL numbers from 5 to 10. By default $\theta = 1/2$ is chosen corresponding to a second-order trapezoid method applied to the viscous diffusion terms.
2. Eq. (13) represents an algebraic system of equations that is solved for the element-centered Lagrange multiplier, λ , during the time-marching procedure. The modified

PPE in Eq. (13) incorporates the effect of the essential velocity boundary conditions from Eq. (3), and automatically builds in the boundary conditions from Eq. (4) and (5) - see Gresho, et al. [5]. Figure 1 shows the primary and dual staggered grids and the location of the cell-centered where λ and P variables.

3. The use of a stabilized PPE, $[C^T M_L^{-1} C + S]$, yields an approximate projection method. In Eq. (13), the consistent PPE is recovered when $S = 0$ resulting an exact projection method.

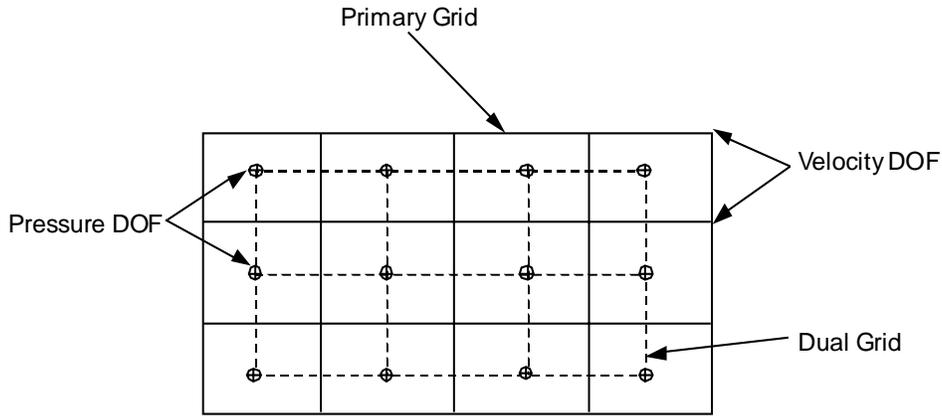


Figure 1. Mesh showing two velocity degrees-of-freedom (DOF) per node, and the PPE dual grid with one DOF per element.

Pressure Stabilization

Although the Q1Q0 element has been condemned by *theoreticians* for its weakly singular modes, this element has been the workhorse for incompressible flow and continues to be widely used. The unstable modes of the Q1Q0 element have been investigated by Sani, et al. [32,33] and more recently by Griffiths and Silvester [34]. Griffiths and Silvester have demonstrated that for problems of physical interest, the Q1Q0 element converges to the true solution in the limit as $h \rightarrow 0$. Additionally, a new convergence proof for the Q1Q0 element may be found in Gresho and Sani [29].

In LS-DYNA, two pressure stabilization methods are used to circumvent the well-known div-stability condition. The local jump stabilization techniques developed by Silvester [35,36] and by Norburn and Silvester [37] provide the foundation for the local methods that are implemented in LS-DYNA. For completeness, the so-called global jump stabilization (first proposed by Hughes and Franca [38]) is also available. In effect, both jump stabilization techniques provide an a priori filter for the weakly unstable pressure modes associated with the Q1Q0 element. However, the use of pressure stabilization results in an approximate projection method since the stabilized PPE is no longer constructed using only the discrete div and grad operators.

The global jump stabilization formulation introduces a pressure diffusion operator that perturbs the incompressibility constraint. The global jump formulation insures mass conservation in a global sense since the null space of the stabilizing matrix, S , contains the constant-pressure vector. However, the global jump stabilization smears the div-free constraint locally over a small region, i.e., the divergence is not zero at the element level.

In contrast to the global method, the local jump stabilization procedure relies on the construction of macro-elements that contain at least one velocity node per edge of the macro element in two dimensions and one velocity node per face in three dimensions. This is shown schematically for two-dimensions in Figure 2. In order to use the local jump stabilization formulation, a pre-processing step that identifies the macro-elements is required. Unlike the global jump stabilization, mass is conserved globally and at the macro-element level. The beneficial effect of the jump stabilization formulations on the convergence rate for ICCG(0) has been presented in Gresho and Sani [29] where a factor of two reduction was observed. Similar results have been obtained by Christon [39] for the SSOR preconditioned conjugate gradient method.

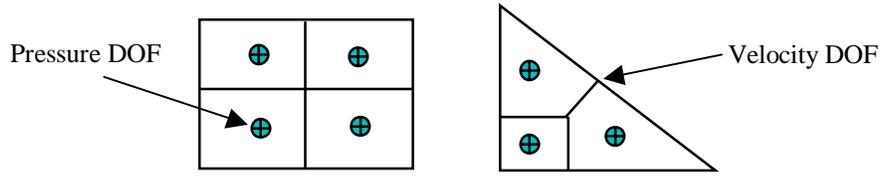


Figure 2. Reference macro-elements showing velocity DOF on inter-element boundaries.

The Projection CG Method

The solution of the time-dependent incompressible Navier-Stokes requires the repeated solution of the PPE problem where the coefficient matrix is fixed and the right-hand side changes each time step. To address this aspect of solving the PPE, an A-conjugate projection technique is integrated with the iterative solution of the PPE in order to use solution information from the previous time steps. In the ensuing discussion, the PPE problem is cast as $Ax = b$ where $A = C^T M_L^{-1} C + S$, $x = \lambda$ and $b = C^T \tilde{u} - C^T \bar{u}^n$.

The use of an A-conjugate projection as a pre-processing step for the solution of the linear system, $Ax = b$, follows the development presented by Fischer [40] with extensions that permit the treatment of either the consistent or stabilized PPE and seeding the A-conjugate projection vectors. Related work on solving linear systems with multiple right-hand sides may be found in Saad [41] and Chan and Wan [42].

To begin the development, the idea of a pre-processing A-conjugate projection step relies on minimizing the distance between the solution at a given time step, x , and the base vectors Φ in the A-norm. Here, Φ is a set of A-conjugate vectors derived from N prior solutions to $Ax = b$ where $\Phi = \{\phi_1, \phi_2, \dots, \phi_N\}$. As shown in Christon [39] $N = 5$ to $N = 10$ provides a nearly optimal trade-off between memory requirements and performance. The memory cost associated with this number of projection vectors is slightly less than the cost of one preconditioning matrix.

Following Christon [39], given a set of A-conjugate vectors, Φ , the best approximation to x^n at time level n is the one that minimizes the error in the A-norm. This is obtained by projecting the right-hand-side, b^n , at time-step n onto the set of base vectors, Φ . This suggests the following solution procedure.

A-conjugate projection CG method

1. Compute the dot-products to obtain $\alpha_i = \phi_i^T b^n$ for all available base vectors.
2. Estimate the initial solution, $\bar{x} = \sum_{i=1}^M \alpha_i \phi_i$ where M is the number of A-conjugate base vectors available in the set of N possible base vectors, i.e., $M \leq N$.
3. Solve $A\Delta x^b = r^b$ using the conjugate gradient method where $r^n = b^n - A\bar{x}$.
4. Update the solution, $x^n = \bar{x} + \Delta x^n$, where $\bar{x} = \sum_{i=1}^M \alpha_i \phi_i$
5. Update the base vectors, Φ , to include new information from the current solution.

Initializing Φ for the first solution, or when the number of existing base vectors exceeds N , is achieved by normalizing the solution as $\phi_1 = x^1 / \|x^1\|_A$ where $\|\cdot\|_A = \sqrt{\{\cdot\}^T A [\cdot]}$. For all other cases, a solution vector is a candidate for addition to Φ only when it contains non-trivial information that is not already present in the basis. Here, the basic idea is that Δx is A-conjugate to \bar{x} as well as to the individual base vectors ϕ_i . Therefore, the addition of a new base vector should be based on criteria that guarantees only new information is being added to the existing base vectors. Addition of a solution vector (or a part of a solution vector) proceeds by first computing that part of the solution that is not contained in the basis as

$$\psi_{M+1} = x^n - \sum_i^M \alpha_i \phi_i. \quad (16)$$

Here, it is assumed that the basis already contains M vectors. Now, each ϕ_i is A-conjugate to ψ_{M+1} by construction and is added to the basis as

$$\phi_{M+1} = \frac{\psi_{M+1}}{\|\psi_{M+1}\|_A}. \quad (17)$$

To illustrate the use of the A-conjugate projection algorithm, Figure 3a) shows a snapshot of the pressure field for a $Re = 100$ vortex shedding computation. In addition, Figures 3b) – 3f) show snapshots of the Φ vectors from the previous five time steps. It is clear that ϕ_1 provides primarily long wavelength information while the other 4 base vectors provide detailed information about the wake. Thus, the vectors $\{\phi_2, \dots, \phi_5\}$ may be viewed as providing primarily short wavelength corrections to ϕ_1 that yield the best approximation to the current pressure field. The A-conjugate projection procedure, in effect, selects the appropriate information from each ϕ vector in order to minimize the residual in the A-norm before performing any CG iterations.

The A-conjugate projection procedure retains both long and short wavelength information, and in this sense, the procedure may be viewed as an approximate means of deflating the eigenvalue spectrum for the PPE. The combination of the A-conjugate projection method, PPE stabilization and SSOR preconditioning has proved to be the most computationally efficient method for solving the PPE.

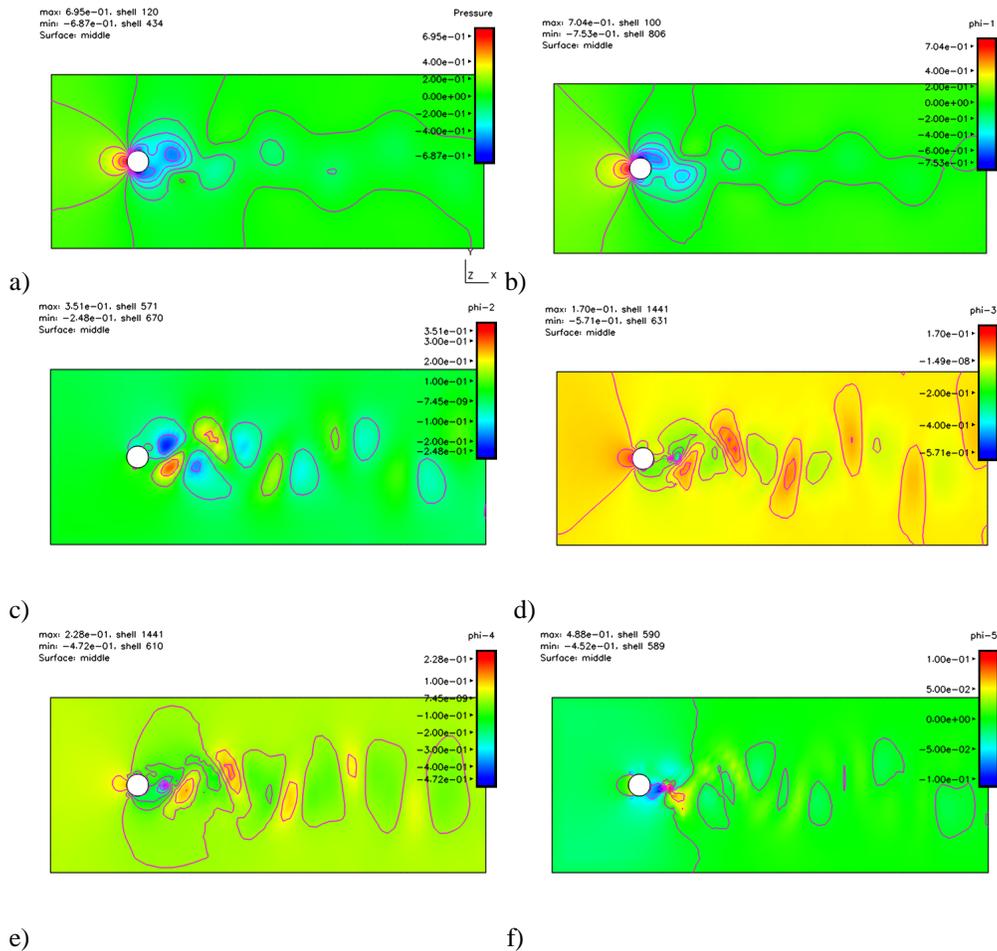


Figure 3. Snapshots of a) the pressure field during $Re = 100$ vortex shedding, and five A-conjugate ϕ fields b) - f) based on pressure solution at the five prior time steps.

DISCUSSION OF RESULTS

This section presents several computations that demonstrate the capability of the transient incompressible flow solver in LS-DYNA. The first computation is shown in Figure 4 and consists of a slot jet with $Re = 4000$ based on a 15 mm slot width and a jet velocity of 4 m/s . In this computation, an energy equation was solved in conjunction with the Navier-Stokes equations using the Boussinesq approximation. The initial conditions consisted of a div-free velocity field with a free-field temperature of 300 K and an inlet jet temperature of 400 K . The snapshots of temperature and vorticity illustrate the inherently unsteady shear driven Kelvin-Helmholtz instability.

Figure 5 shows snapshots of the pressure and vorticity field for the MIT waterfoil. The waterfoil is a NACA 16 thickness form with maximum thickness $t_0/c = 8.84\%$ and maximum camber $f_0/c = 2.576\%$ with a beveled anti-singing trailing edge. This computation was performed as a laminar-flow simulation for $Re = 10,000$ based on chord length. At this Reynolds number, the wake is inherently unsteady and exhibits periodic vortex

shedding. The proximity of the outflow boundary to the trailing edge of the waterfoil was intentional to illustrate the performance of the “natural” outflow boundary conditions.

Figure 6 shows pressure isosurfaces for a steady-state solution for the $Re = 10,000$ flow past a $1/24^{\text{th}}$ scale streamline hull. In this computation, quarter-symmetry was used with tow-tank conditions prescribed at the far field of the computational domain. The transient solution algorithm was used in a pseudo-time marching mode to obtain a steady-state solution.

Figure 7 shows a snapshot of the x-vorticity field from a large-eddy simulation of a $Re = 10,000$ lid-driven cavity [43]. The computation used a $48 \times 48 \times 48$ grid and a Smagorinsky model. In this calculation, the pressure solver used 5 A-conjugate vectors, SSOR preconditioning and required approximately 11 iterations per time step on average to solve for the 110,592 unknown pressures. In this computation, the pressure stabilization permits the prescription of non-leaky velocity boundary conditions without any deleterious effects from spurious pressure modes.

CONCLUSIONS

The current incompressible flow simulation capabilities in LS-DYNA include the following.

- 2-D/3-D time-accurate incompressible Navier-Stokes based on second-order accurate approximate projection methods.
- A unique A-conjugate projection-based preconditioned CG pressure solver that minimizes the computational cost associated with the implicit pressure.
- Robust pressure-stabilization with mass-conserving macro-elements.
- A fully-parallel implementation based on a domain-decomposition message-passing paradigm (MPI based).
- A monotonicity-preserving advection scheme.
- Thermal convection and up to 10 species transport equations.
- Turbulence models.
 - Baseline Smagorinsky
 - Dynamic Smagorinsky¹
 - Spallart-Allmaras¹
- Boundary conditions based on nodesets and sidesets.

The time-accurate incompressible flow solver in LS-DYNA will provide the basis for a viscous fluid-structure interaction capability that will complement the existing boundary-element fluid-structure and ALE capabilities. In the future, the incompressible flow solver will be integrated with the existing interface reconstruction algorithms for mold-filling and casting simulations. In addition, the segregated incompressible solution methodology will be extended to provide a native steady-state flow solver.

REFERENCES

1. Chorin, A. J., (1968). “Numerical solution of the Navier-Stokes equations”, *Mathematics of Computations*, 22, pp. 745-762.

¹ The dynamic Smagorinsky and Spallart-Allmaras models are under development at the time of this writing.

2. van Kan, J., (1986), "A second-order accurate pressure-correction scheme for viscous incompressible flow", *SIAM Journal for Scientific and Statistical Computing*, 7, pp. 870-891.
3. Bell, J. B., Colella, P., and Glaz, H. M., (1989). "A second-order projection method for the incompressible Navier-Stokes equations, *Journal of Computational Physics*, 85, pp. 257-283.
4. Gresho, P. M., (1990). "On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. part 1: Theory", *International Journal for Numerical Methods in Fluids*, 11, pp. 587-620.
5. Gresho, P. M. and Chan, S. T., (1990). "On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. part 2: Implementation", *International Journal for Numerical Methods in Fluids*, 11, pp. 621-659.
6. Gresho, P.M., Chan, S. T., Christon, M. A., and Hindmarsh, A. C., (1995), "A little more on stabilized Q1Q1 for transient viscous incompressible flow", *International Journal for Numerical Methods in Fluids*, 21, pp. 837-856.
7. Gresho, P.M., and Chan, S. T., (1996). "Projection 2 goes turbulent -- and fully implicit", preprint *International Journal for Computational Fluid Dynamics* (also see LLNL UCRL-JC-123727).
8. Almgren, A. S., Bell, J. B., Colella, P. and Howell, L. H. (1993). "An adaptive projection method for the incompressible Euler equations, in Eleventh AIAA Computational Fluid Dynamics Conference, AIAA, pp. 530-539.
9. Almgren, A. S., Bell, J. S., and Szymczyk, W. G., (1996). "A numerical method for the incompressible Navier-Stokes equations based on an approximate projection", *SIAM Journal for Scientific Computing*, 17, pp.358-369.
10. Almgren, A. S., Bell, J. B., Colella, P., Howell, L. H., and Welcome, M. L., (1998). "A conservative adaptive projection method for the variable density incompressible Navier-Stokes equations", *Journal of Computational Physics*, 142, pp. 1-46.
11. Almgren, A. S., Bell, J. B. and Crutchfield, W. S., (1999). "Approximate projection methods: Part I. Inviscid analysis", preprint submitted to *SIAM Journal on Scientific Computing*.
12. Rider, W. J., (1994). "Filtering non-solenoidal modes in numerical solutions of incompressible flows", Tech. Rep. LA-UR-3014, Los Alamos National Laboratory, Los Alamos, New Mexico.
13. Rider, W. J., (1994). "The robust formulation of approximate projection methods for incompressible flows", Tech. Rep. LA-UR-3015, Los Alamos National Laboratory, Los Alamos, New Mexico.
14. Rider, W. J., (1995). "Approximate projection methods for incompressible flow: implementation, variants and robustness", Tech. Rep. LA-UR-2000, Los Alamos National Laboratory, Los Alamos, New Mexico.
15. Rider, W. J., Kothe, D. B., Mosso, S. J., Cerutti, J. H., and Hochstein, J. I., (1995). "Accurate solution algorithms for incompressible multiphase flows", Tech. Rep. AIAA-95-0699, AIAA, Reno, Nevada, January.
16. Minion, M. L. (1996). "A projection method for locally refined grids", *Journal of Computational Physics*, 127, pp. 158-178.
17. Guermond, J. L. and Quartapelle, L., (1997). "Calculation of incompressible viscous flow by an unconditionally stable projection fem", *Journal of Computational Physics*, 132, pp. 12-23.
18. Puckett, E. G., Almgren, A. S., Bell, J. B., Marcus, D. L. and Rider, W. J., (1997). "A high-order projection method for tracking fluid interfaces in variable density incompressible flows", *Journal of Computational Physics*, 130, pp. 269 -282.

19. Sussman, M., Almgren, A. S., Bell, J. B., Colella, P., Howell, L. H. and Welcome, M. L., (1999). "An adaptive level set approach for two-phase flows", *Journal of Computational Physics*, 148, pp. 81-124.
20. Knio, O. M., Najm, H. N., and Wyckoff, P. S., (1999). "A semi-implicit numerical scheme for reacting flow. II. stiff operator-split formulation", pre-print submitted to *Journal of Computational Physics*.
21. Brown, D. L., and Minion, M. L., (1995). "Performance of under-resolved two-dimensional incompressible flow simulations", *Journal of Computational Physics*, 122 pp.165-183.
22. Minion, M. L., and Brown, D. L., (1997). "Performance of under-resolved two-dimensional incompressible flow simulations, II", *Journal of Computational Physics*, 138 pp. 734-765.
23. Wetton, B. B., (1998). "Error analysis of pressure increment schemes", submitted to *SINUM*.
24. Guermond, J.-L., (1996). "Some implementations of projection methods for Navier-Stokes equations", *Mathematical Modelling and Numerical Analysis*, 30, pp. 637-667.
25. Guermond, J.-L., (1997). "A convergence result for the approximation of the Navier-Stokes equations by an incremental projection method", *C. R. Acad. Sci. Paris*, 325, pp. 1329-1332.
26. Guermond, J.-L., and Quartapelle, L., (1998). "On stability and convergence of projection methods based on pressure Poisson equation", *International Journal for Numerical Methods in Fluids*, 26 , pp. 1039-1053.
27. Guermond, J.-L., and Quartapelle, L., (1998). "On the approximation of the unsteady Navier-Stokes equations by finite element projection methods", *Numerische Mathematik*, 80, pp. 207-238.
28. Gresho, P. M., and Sani, R. L., (1987). "On pressure boundary conditions for the incompressible Navier-Stokes equations", *International Journal for Numerical Methods in Fluids*, 7, pp. 1111-1145.
29. Gresho, P. M., and Sani, R. L., (1998). "Incompressible flow and the finite element method, Advection-diffusion and isothermal laminar flow", John Wiley & Sons, Chicester, England.
30. Gresho, P. M., and Chan, S. T., and Lee, R. L., and Upson, C. D. (1984). "A modified finite element method for solving the time-dependent, incompressible Navier-Stokes equations. Part 1: Theory", *International Journal for Numerical Methods in Fluids*, 4, pp. 557-598.
31. Christon, M. A., (1997). "A domain-decomposition message-passing approach to transient viscous incompressible flow using explicit time integration", *Computer Methods in Applied Mechanics and Engineering*, 148, pp. 329-352.
32. Sani, R. L., Gresho, P. M., Lee, R. L., and Griffiths, D. F., (1981). "The cause and cure (?) of the spurious pressures generated by certain FEM solutions of the incompressible Navier-Stokes equations: Part 1", *International Journal for Numerical Methods in Fluids*, 1, pp. 17-43.
33. Sani, R. L., Gresho, P. M., Lee, R. L., Griffiths, D. F., and Engelman, M., (1981). "The cause and cure (!) of the spurious pressures generated by certain FEM solutions of the incompressible Navier-Stokes equations: Part 2", *International Journal for Numerical Methods in Fluids*, 1 , pp. 171-204.
34. Griffiths, D. and Silvester, D., (1994). "Unstable modes of the Q1Q0 element," Tech. Report NA-257, University of Manchester/UMIST, Manchester Centre for Computational Mathematics.
35. Silvester, D. J., and Kechkar, N., (1990). "Stabilised bilinear-constant velocity-pressure finite elements for the conjugate gradient solution of the stokes problem", *Computer Methods in Applied Mechanics and Engineering*, 79, pp. 71-86.

36. Silvester, D. J., (1995). "Stabilised mixed finite element methods", draft preprint.
37. Norburn, S., and Silvester, D. J., (1997). "Stabilised vs. stable mixed methods for incompressible flow," Tech. Report NA-307, University of Manchester/UMIST, Manchester Centre for Computational Mathematics.
38. Hughes, T. J. R., and Franca, L. P., (1987)., "A new finite element formulation for computational fluid dynamics: VII. the Stokes problem with various well-posed boundary conditions: symmetric formulations that converge for all velocity/pressure spaces", *Computer Methods in Applied Mechanics and Engineering*, 65, pp. 85-96.
39. Christon, M., (1999). "Dealing with pressure: FEM solution strategies for the pressure in the time-dependent Navier-Stokes equations," submitted to *International Journal for Numerical Methods in Fluids*.
40. Fischer, P. F., (1993). "Projection techniques for iterative solution of $Ax = b$ with successive right-hand sides," Tech. Rep. 93-90, ICASE.
41. Saad, Y., (1987). "On the Lanczos method for solving symmetric linear systems with several right-hand sides", *Mathematics of Computation*, 48, pp. 651-662.
42. Chan, T. F., and Wan, W. L., (1997). "Analysis of projection methods for solving linear systems with multiple right-hand sides", *SIAM Journal on Scientific Computing*, 18, pp.1698-1721.
43. Zang, Y., Street, R. L., and Koseff, J. R., (1993). "A dynamic mixed subgrid-scale model and its application to turbulent recirculating flows," *Physics of Fluids A*, 5, pp.3186-3196.

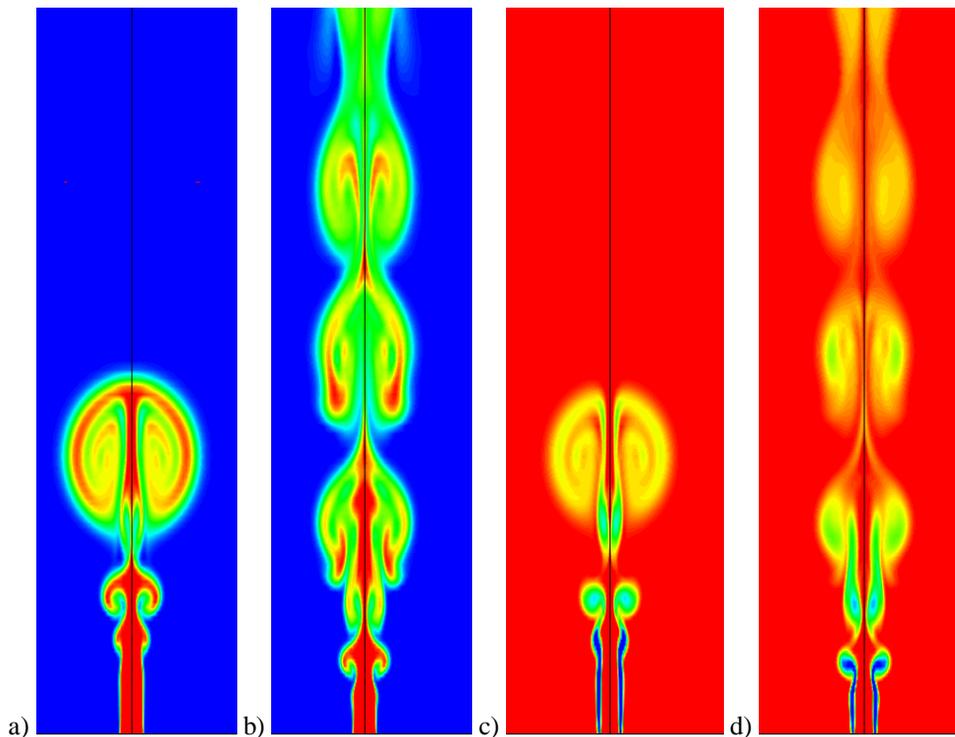


Figure 4. Snapshots of the $Re = 4000$ momentum driven slot jet showing the temperature field a) – b), and the vorticity field c) – d). (The temperature and vorticity fields have been reflected about the vertical centerline.)

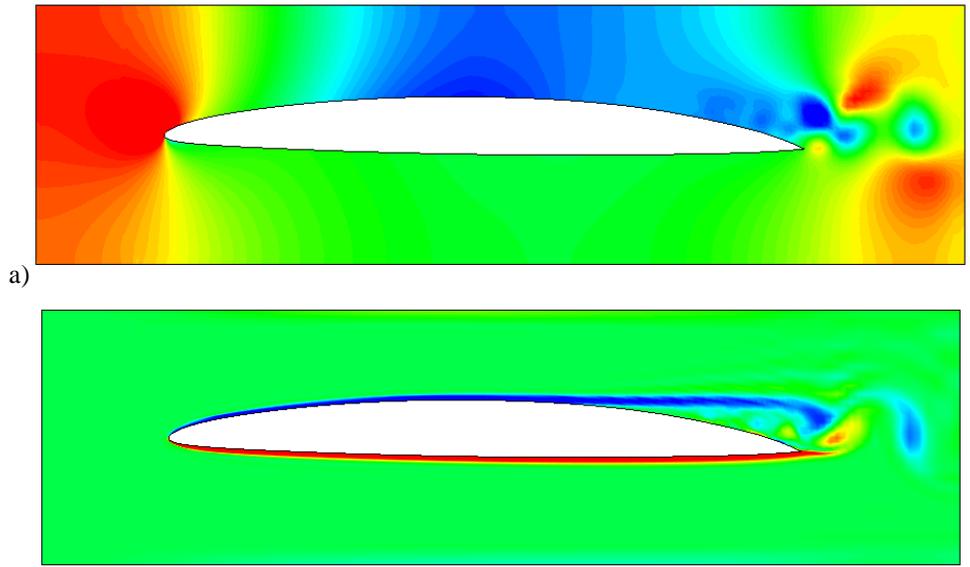


Figure 5. MIT waterfoil – snapshots showing a) pressure and b) vorticity during a vortex shedding cycle for $Re = 10,000$.

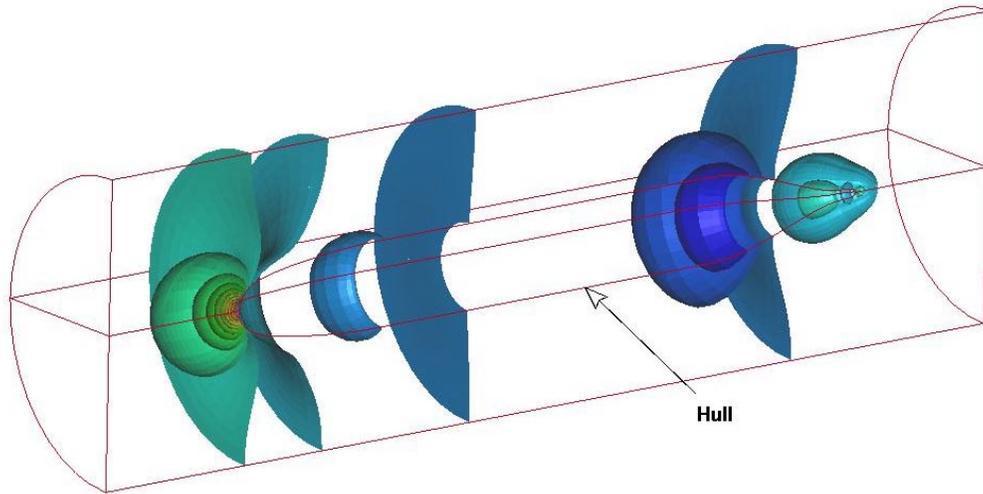


Figure 6. Steady pressure field for 1/24 scale “suboff” configuration at $Re = 10,000$. (The quarter-symmetry pressure field has been reflected about the x-y plane.)

Re=10,000 VORTEX SHEDDING

Time = 61.5
Isosurfaces of x-vorticity
min=-166.936, at node# 84646
max=166.879, at node# 113614

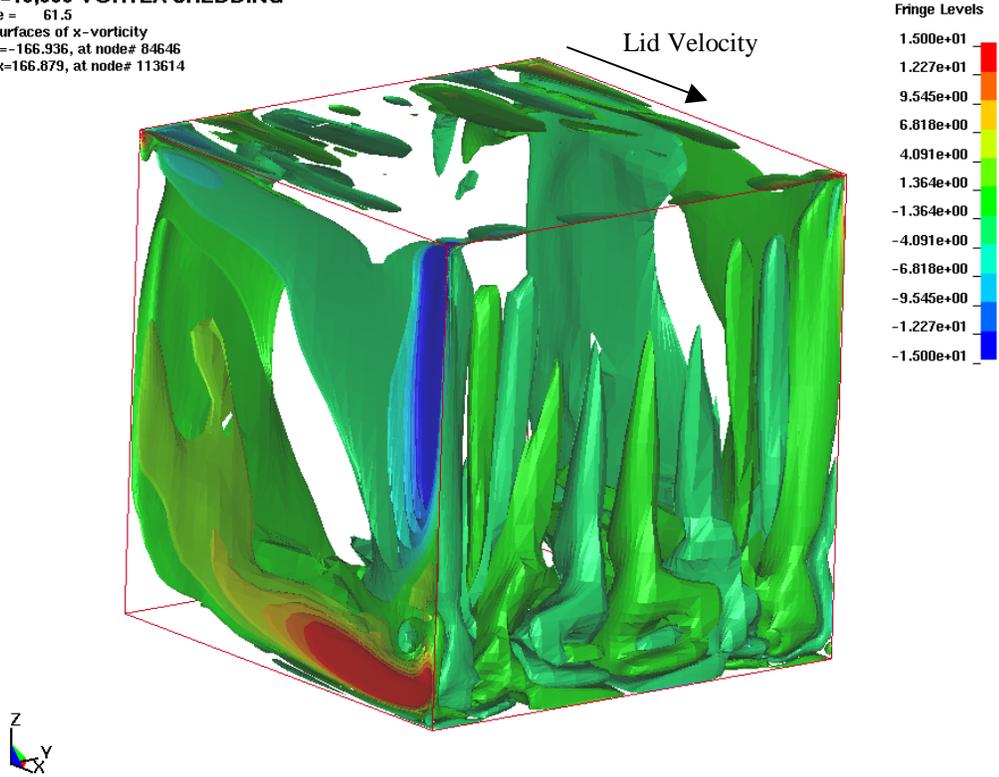


Figure 7. Snapshot showing the x-vorticity field for a large eddy simulation of a $Re = 10,000$ lid driven cavity.