TIME-ACCURATE SOLUTIONS OF INCOMPRESSIBLE NAVIER-STOKES EQUATIONS FOR POTENTIAL TURBOPUMP APPLICATIONS

Cetin Kiris
MCAT, Inc.
Dochan Kwak
NASA Ames Research Center
Mail Stop T27B-1
Moffett Field, CA 94035

Abstract

Two numerical procedures, one based on artificial compressibility method and the other pressure projection method, are outlined for obtaining time-accurate solutions of the incompressible Navier-Stokes equations. The performance of the two method are compared by obtaining unsteady solutions for the evolution of twin vortices behind a flat plate. Calulated results are compared with experimental and other numerical results. For an unsteady flow which requires small physical time step, pressure projection method was found to be computationally efficient since it does not require any subiterations procedure. It was observed that the artificial compressibility method requires a fast convergence scheme at each physical time step in order to satisfy incompressibility condition. This was obtained by using a GMRES-ILU(0) solver in our computations. When a line-relaxation scheme was used, the time accuracy was degraded and time-accurate computations became very expensive.

Introduction

The primary objective of this research is to support the design of liquid rocket systems for the Advanced Space Transportation System. Since the space launch systems in the near future are likely to rely on liquid rocket engines, increasing the efficiency and reliability of the engine components is an important task. One of the major problems in the liquid rocket engine is to understand fluid dynamics of fuel and oxidizer flows from the fuel tank to plume. Turbopumps in liquid rocket engines are one of the biggest source of vibrations. Understanding the flow through the entire turbopump geometry through numerical simulation will be of significant value toward design. This will help to improve safety of future space missions. One of the milestones of this effort is to develop, apply and demonstrate the capability and accuracy of 3D CFD methods as efficient design analysis tools on high performance computer platforms. In order to achieve flange-to-flange entire turbopump simulations, moving boundary capability and an efficient time-accurate integration method should be build in the numerical procedure. This paper, in particular, is concerned with the time integration procedure of incompressible Navier-Stokes equations.

The incompressible Navier-Stokes equations pose a special problem of satisfying the mass conservation equation because it is not coupled to the momentum equations. To satisfy incompressibility various procedures can be selected depending on the choice of formulations, variables, discretization and iterative schemes. In this paper, two formulations are considered, the first one based on an artificial compressibility method and the second one on a pressure projection method. The artificial compressibility method[1] takes advantage of the advances made in conjunction with compressible flow computations. This approach relaxes the requirement of enforcing mass conservation equation rigorously at each time iteration, however, at the expense of introducing an artificial wave phenomenon. This approach can be viewed as a special case of a preconditioned compressible flow formulation. However, the computational efficiency is in general better than that of compressible flow solvers at the incompressible limit. This approach has been shown to be very robust in a wide range of applications[2-4].

The first primitive variable method for incompressible flow was developed by Harlow and Welch[5] using pressure projection. Numerous variants have been developed since. In this method, the pressure is used as a mapping parameter to satisfy the continuity equation. The usual computational procedure involves choosing the pressure field at the current time step such that continuity is satisfied at the next time step. The time step is advanced in multiple steps (fractional step) which is computationally convenient. However, governing equations are not coupled as in an artificial compressibility approach. This will affect the robustness and limit the maximum allowable time step size. Since this approach is time accurate, there are cases where the fractional step solver is computationally more efficient compared to the artificial compressibility method[5-9].

Various numerical algorithms associated with these methods have been developed along with accompanying flow solvers. In the present paper, it is intended to outline the time integration procedures of the two methods discussed above. A new time integration scheme is also presented for pressure projection method. Numerical results from both formulations for the development of the twin vorticies[10,11] behind the flat plate are presented in computed results section.

Artificial Compressibility Formulation

The artificial compressibility algorithm introduces a time-derivative of the pressure term into the continuity equation; the elliptic-parabolic type partial differential equations are transformed into the hyperbolic-parabolic type. The artificial compressibility method by Chorin (1967) can be written as

$$\frac{1}{\beta}\frac{\partial p}{\partial t} + \frac{\partial u_i}{\partial x_i} = 0 \tag{1}$$

$$\frac{\partial u_i}{\partial t} = -\frac{\partial p}{\partial x_i} + h_i \ = -r_i \tag{2}$$

where t is time, x_i the Cartesian coordinates, u_i the corresponding velocity components, p the pressure, β artificial compressibility, and h_i contains both convective and viscous terms. At steady state the pressure term in continuity equation drops out and thus incompressibility is recovered. For time accurate computations, this has to be repeated at each time level to maintain incompressibility at each time step.

In the present study, the time derivatives in the momentum equations are differenced using a second-order, three-point, backward-difference formula.

$$\frac{3\mathbf{u}^{n+1} - 4\mathbf{u}^{n} + \mathbf{u}^{n-1}}{2\Delta t} = -\mathbf{r}^{n+1} \tag{3}$$

where **u** and **r** denote the dependent variable vector and the right hand side vector for the momentum equations, respectively. After the discretization in time, the pseudocompressibilty term and pseudo-time level (m) are introduced to equations.

$$\begin{aligned}
&\frac{1}{\Delta\tau}(p^{n+1,m+1} - p^{n+1,m}) = -\beta\nabla\cdot\mathbf{u}^{n+1,m+1} \\
&\frac{1.5}{\Delta t}(\mathbf{u}^{n+1,m+1} - \mathbf{u}^{n+1,m}) = -\mathbf{r}^{n+1,m+1} - \\
&\qquad\qquad\qquad\qquad\qquad\frac{3\mathbf{u}^{n+1,m} - 4\mathbf{u}^{n} + \mathbf{u}^{n-1}}{2\Delta t}
\end{aligned} \tag{4}$$

Here Δt, $\Delta\tau$, n, and m denote physical time step, pseudo-time step, physical time level, and subiteration time level, respectively. The equations are iterated to convergence in pseudo-time for each physical time step until the divergence of the velocity field has been reduced below a specified tolerance value. This typically requires 10 to 30 subiterations.

The matrix equation is solved iteratively by using a nonfactored Gauss-Seidel type line-relaxation scheme,[12] which maintains stability and allows a large pseudo-time step to be taken. Details of the numerical method can be found in Refs. 2-3. GMRES scheme has also been utilized for the solution of the resulting matrix equation[13]. Computer memory requirement for the corresponding flow solver (INS3D-UP code) with line-relaxation is 35 times number of grid points in words, and with GMRES-ILU(0) scheme is 220 times number of grid points in words. Extensive memory requirement for GMRES scheme makes the code unpractical for three-dimensional applications. Writing a matrix-free GMRES solver renmains to be one of the items for future enhancemets.

The original version of the INS3D code[2] with pseudocompressibility approach utilized the Beam-Warming[14] approximate factorization algorithm and central differencing of the convective terms. Since the convective terms of the resulting equations are hyperbolic, upwind differencing can be applied to these terms. The current versions of the INS3D-UP code use flux-difference splitting based on the method of Roe.[15] Chakravarthy[16] outlines a class of high-accuracy flux-differencing schemes for the compressible flow equations. The third and fifth-order upwind differencing used here is an implementation of these schemes for the incompressible Navier-Stokes equations. The upwind differencing leads to a more diagonal dominant system than does central differencing and does not require a user-specified artificial dissipation. The viscous flux derivatives are computed by using central differencing.

Time-accurate artificial compressibility formulation has been used successfully for unsteady calculations. The only drawback of this formulation is the computational cost due to subiteration procedure.

Pressure Projection Method

The time integration scheme is based on operator splitting, which can be accomplished in several ways by combining the pressure, convective, and viscous terms in the momentum equations. The fractional step method is based on the decomposition of vector field into a divergence free component and gradient of a scalar field. The common application of this method is done in two steps. The first step is to solve for an auxiliary velocity field using the momentum equations. In the second step, the velocity field is corrected by using the pressure which can map the auxiliary velocity onto a divergence free velocity field. The momentum equations are discretized in time using a second-order implicit Runga-Kutta method (RK2) which can also be viewed as a predictor-corrector method.

$$\frac{1}{\Delta t}(u_i^* - u_i^n) = -\frac{\partial p^n}{\partial x_i} + h(u_i^*) \tag{5}$$

and a corrector step

$$\frac{2}{\Delta t}(u_i^{n+1} - u_i^n) = -\frac{\partial p^{n+1}}{\partial x_i} + h(u_i^{n+1}) - \frac{\partial p^n}{\partial x_i} + h(u_i^*)\,) \tag{6}$$

where u_i^* denotes the auxiliary velocity field. The **h** term in the momentum equations includes the convective and viscous terms. By using equation (5), equation (6) can be written as

$$\frac{2}{\Delta t}(u_i^{n+1} - u_i^*) = -\frac{\partial p^{n+1}}{\partial x_i} + h(u_i^{n+1}) - \frac{1}{\Delta t}(u_i^* - u_i^n) \tag{7}$$

By subtracting equation (5) from equation (7), we obtain

$$\frac{2}{\Delta t}(u_i^{n+1} - u_i^*) = -\nabla p^{'} + h(u_i^{n+1}) - h(u_i^*) \tag{8}$$

where $p^{'} = p^{n+1} - p^n$. At $n+1$ time level, the velocity field has to satisfy the incompressibility condition which is the continuity equation.

$$\nabla \cdot \mathbf{u}^{n+1} = 0 \tag{9}$$

This incompressiblity condition is enforced by using a Poisson equation for pressure.

$$\nabla^2 p^{'} = \frac{2}{\Delta t}\nabla \cdot \mathbf{u}^* \tag{10}$$

The Poisson equation for pressure is obtained by taking the divergence of equation (8) and using equation (9). The only assumption is made in this procedure is that $h(u_i^{n+1}) - h(u_i^*)$ term in equation 8 is considered small. If the corrector step was explicit, this term would vanish.

In equations (5) and (7), both convective and viscous terms are treated implicitly. The residual term at the $(*)$ and $(n+1)$ level is linearized giving the following equations in delta form

$$\left[\frac{I}{\Delta t} + \left(\frac{\partial h}{\partial u_i}\right)^n\right](u^* - u^n) = -\frac{\partial p^n}{\partial x_i} + h(u_i^n) \tag{11}$$

$$\left[\frac{2I}{\Delta t} + \left(\frac{\partial h}{\partial u_i}\right)^*\right](u^{n+1} - u^*) = -\frac{\partial p^{n+1}}{\partial x_i} + h(u_i^*) - \frac{1}{\Delta t}(u_i^* - u_i^n) \tag{12}$$

where I is the idendity matrix. Equation (12) can also be written in more familiar form of fractional-step method by substituting equation (5) in equation (12).

$$\left[\frac{2I}{\Delta t}+\left(\frac{\partial h}{\partial u_i}\right)^*\right](u^{n+1}-u^*)=-\frac{\partial p^{'}}{\partial x_i} \tag{13}$$

Equations (11), (10), and (12) give the proposed time integration procedure of the pressure projection method.

The algorithm for the pressure projection method is based on a finite-volume formulation and uses the pressure in the cell center and the mass fluxes across the faces of each cell as dependent variables. The discretization of the mass conservation equation in finite volume formulation gives

$$\begin{aligned}&(\mathbf{S}^\xi\cdot\mathbf{u})_{j+\frac{1}{2},k,l}-(\mathbf{S}^\xi\cdot\mathbf{u})_{j-\frac{1}{2},k,l}+\\&(\mathbf{S}^\eta\cdot\mathbf{u})_{j,k+\frac{1}{2},l}-(\mathbf{S}^\eta\cdot\mathbf{u})_{j,k-\frac{1}{2},l}+\\&(\mathbf{S}^\zeta\cdot\mathbf{u})_{j,k,l+\frac{1}{2}}-(\mathbf{S}^\zeta\cdot\mathbf{u})_{j,k,l-\frac{1}{2}}=0\end{aligned} \tag{14}$$

where $\mathbf{S}$ is the surface area vector. The mass conservation equation is evaluated over the faces of a computational Each term in equation (14) approximates the mass flux over the corresponding cell face. If the mass fluxes are chosen as unknowns, the continuity equation is satisfied automatically in generalized coordinate systems. The mass fluxes over the ξ, η , and ζ faces of the computational cell are

$$\begin{aligned}U^\xi&=\mathbf{S}^\xi\cdot\mathbf{u}\\U^\eta&=\mathbf{S}^\eta\cdot\mathbf{u}\\U^\zeta&=\mathbf{S}^\zeta\cdot\mathbf{u}\end{aligned} \tag{15}$$

The continuity equation with this choice of the dependent variables takes a form identical to the Cartesian case. Therefore, the mass fluxes are considered as the 'natural' dependent variables for projection methods in curvilinear coordinates. The mass conservation equation with new dependent variables in a generalized coordinate system becomes

$$\begin{aligned}U^\xi_{j+\frac{1}{2},k,l}-U^\xi_{j-\frac{1}{2},k,l}+U^\eta_{j,k+\frac{1}{2},l}-U^\eta_{j,k-\frac{1}{2},l}+&\\U^\zeta_{j,k,l+\frac{1}{2}}-U^\zeta_{j,k,l-\frac{1}{2}}=0&\end{aligned} \tag{16}$$

Treating the mass fluxes as dependent variables in finite volume formulation is equivalent to using contravariant velocity components, scaled by the inverse of the transformation Jacobian, in a finite-difference formulation. This choice of mass fluxes as dependent variables complicates the discretization of the momentum equations. In order to replace $\mathbf{u}$ by the new dependent variables U^l, the corresponding area vectors are dotted with the momentum equations. Then the integral momentum equation is evaluated on different computational cells for each unknown U^l. Each cell has the dimensions of $\Delta\xi\times\Delta\eta\times\Delta\zeta$, but the centers are located at $(j+\frac{1}{2},k,l)$, $(j,k+\frac{1}{2},l)$, and $(j,k,l+\frac{1}{2})$ for U^ξ, U^η, and

U^{ζ} momentum equations, respectively. The staggered grid orientation eliminates pressure checker-board-like oscillations in pressure and provides more compact stencils. The derivation of momentum equations and the solution procedure is outlined in reference 9. Since each equation is solved in a segregated fashion, memory requirements for GMRES solver in INS3D-FS is not as big as INS3D-UP code. Required memory for INS3D-FS is 70 times number of grid poins in words.

Computed Results

In this section, numerical results for the time evolution of twin vortices behind a flat plate are presented in order to verify the time-dependent features of the two algorithms. In order to investigate different features of the algorithms, several cases are needed to run with various code related parameters. To speed up this process, a two dimensional test case is selected here. It should be noted that associated flow solvers, INS3D-UP for artificial compressibility method and INS3D-FS for pressure projection method, are written for three-dimensional applications. With this numerical experiment, it is intended to give some basis for selecting a method for large three-dimensional unsteady applications where computing resources become a critical issue.

Computed results from both methods are compared with the experimental data by Taneda and Honji[10]. The experiment has carried out in a water tank 40 cm wide. A thin test plate of size $H = 3cm$ immersed in the water was started from the rest impulsively at the velocity $U = 0.495cm/s$. Reynolds number for this case is 126 based on $U = 0.495cm/s$ velocity and the plate height H. Computational grid with with size of 181x81x3 is presented in figure 1. Since INS3D-FS is written in finite volume staggered grid formulation, it requires one additional ghost cell in each direction. Figure 2 shows calculated velocity vectors obtained from INS3D-FS at various times. The flow separates the plate at each edge and forms a vortex pair. The twin vortices become longer in the flow direction with time.

The calculated time history of the stagnation point is compared with experimental results and other numerical results in figure 3. Symbols represent experimental measurements, solid line and dashed line represent results from INS3D-UP and INS3D-FS, respectively. Dotted line show the numerical results from finite element formulations of Yoshida and Nomura[11]. The interval for time integration was 0.5 sec, which corresponds to nondimensional value of 0.0825, for all computations in figure 3. Eventhough the plate started impulsively in the experiment, the computations presented in figure 3 have a slow start procedure. Figure 4 shows prescribed velocity for an impulsive start (4a) and for a slow start (4b) used in INS3D-UP and INS3D-FS calculations. Reference 11 also used same slow start procedure in their calculations. When nondimensional time step of 0.0825 was used with an impulsive start, large discrepancies were observed between numerical results and the experimental mesurements. This can be seen in figure 5a. When the time step is decreased, fairly good agreement was observed between numerical results and the measurements as seen in figure 5b. For the slow start case, the velocity profile shown in figure 4b is prescribed and the origin of time of calculation is appropriately shifted from the time of experiment. This unsteady computations with INS3D-FS ($\Delta t = 0.0825$) was completed in two hours of CPU time on single processor Cray-J90.

INS3D-UP computations with line-relaxation scheme is presented in figure 6. Various artificial compressibility parameters and number of subiterations were used. Figure 6 shows the effects of number of subiterations and the effects of using two different artificial compressibility parameters β. When the incompressibility conditions is not satisfied at each physical time step, numerical results can be erronous in time-accurate computations. With line-relaxation scheme, INS3D-UP calculations required between 4 hours of CPU time (10 subiterations at each physical time step) and 14 hours (40 subiterations) on a single processor Cray-J90 computer. Our observation for the time-accurate computations from this numerical example is that the artificial compressibility method requires a fast convergence scheme at each physical time step in order to satisfy incompressibility condition. If this is not satisfied as seen in line-relaxation scheme, the time accuracy is degraded (see figure 6). In addition, artificial compressibility method with line relaxation scheme can be expensive for 3D time-accurate computations. In figure 7, INS3D-UP results with GMRES-ILU(0) solver are presented. These results were obtained less than 4 hours on a Cray-J90 computer. Fairly good agreement was obtained between the computed results and experimental data. With GMRES-ILU(0) solver, the mass flow ratio between inflow and exit was always satisfied. In addition, the discrepancies between numerical results are very small when two different values of artificial compressibility parameter were used. Figure 8 shows the results from artificial compressibility method with and without Poisson equation correction for the pressure. In artifical compressibility method, after the first sub-iteration, the Poisson equation is employed for the pressure correction. Chain-dashed line in figure 8 represents the results from this new procedure. With the Poisson equation correction, the line relaxation results compare well with experimental data and the GMRES results with 10 subiterations. With this new procedure, both computing time and memory requirement are substantially reduced (at least three times).

Concluding Remarks

Unsteady computations were performed using two different solution algorithms, which are artificial compressibility method and pressure projection method. When a fast converging scheme, such as GMRES-ILU(0) solver, was incorparated in artificial compressibility method, fairly good agreement was obtained between computed results and experimental data. Our numerical experiment showed that incompressiblity condition was satisfied in 10 subiterations at each physical time step. Memory requirement of this scheme is the major drawback for three-dimensional large applications. However, memory requirement may not be an issue on the paralel platforms, such as SGI Origin 2000. The line-relaxation scheme in artificial compressibility method becomes very expensive and results in erronous solution for time-accurate computations. For an unsteady flow which requires small physical time step, pressure projection method was found to be computationally efficient since it does not require any subiterations procedure. However, governing equations are not fully coupled as in the artificial compressibility approach. This may affect the robustness and limit the maximum allowable time step size. A new method is developed by combining pressure projection method with artificial compressibility method. With Poisson solver correction, the number of subiteration was reduced to two iterations at each physical time step.

Acknowledgments

Computer time was provided by the Numerical Aerodynamic Simulation (NAS) Facility and the Central Computing Facility at NASA Ames Research Center.

References

1. Chorin, A., J., "A Numerical Method for Solving Incompressible Viscous Flow Problems," *Journal of Computational Physics,* Vol. 2, pp. 12-26, 1967.

2. Kwak, D., Chang, J. L C., Shanks, S. P., and Chakravarthy, S., "A Three-Dimensional Incompressible Navier-Stokes Flow Solver Using Primitive Variables," *AIAA Journal* , Vol. 24, No. 3, pp. 390-396, 1977.

3. Rogers, S. E., Kwak, D. "Upwind Differencing for the Time-Accurate Incompressible Navier-Stokes Equations," *AIAA Journal* , Vol. 28, No. 2, pp. 253-262, 1990.

4. Rogers, S. E., Kwak, D. and Kiris, C., "Numerical Solution of the Incompressible Navier-Stokes Equations for Steady and Time-Dependent Problems," *AIAA Journal* , Vol. 29, No. 4, pp. 603-610, 1991.

5. Harlow, F. H., and, Welch, J. E., "Numerical Calculation of Time-Dependent Viscous Incompressible Flow with Free Surface," *Phys. Fluids* , Vol. 8, No. 12, pp. 2182-2189, 1965.

6. Kim, J. and Moin, P., "Application of a Fractional-Step Method to Incompressible Navier-Stokes Equations," *J. Comp. Phys.* , Vol. 59, pp. 308-323, 1985.

7. Choi, H., and Moin, P., "Effects of the Computational Time Step on Numerical Solutions of Turbulent Flow," *J. Comp. Phys.* , Vol. 113, pp. 1-4, 1994.

8. Rosenfeld, M., Kwak, D., and Vinokur, M., "A Fractional-Step Method for the Unsteady Incompressible Navier-Stokes Equations in Generalized Coordinate Systems," AIAA Paper No. 88-0718, 1988.

J. Comp. Physics , Vol. 94, No. 1, pp. 102-137, 1991.

9. Kiris, C., and Kwak, D., "Numerical Solution of Incompressible Navier-Stokes Equations Using a Fractional-Step Approach,"AIAA Paper No. 96-2089, 1996.

10. Taneda, S., and Honji, H., "Unsteady Flow Past a Flat Plate Normal to the Direction of Motion," *J. Phys. Soc. Japan* , Vol. 30, pp. 262-273, 1971.

11. Yoshida, Y., and Nomura, T., "A Transient Solution Method for the Finite Element Incompressible Navier-Stokes Equations," *Int. J. Num. Methods in Fluids* , Vol. 5, pp. 873-890, 1985.

12 MacCormack, R., W., "Current Status of Numerical Solutions of the Navier-Stokes Equations,"
AIAA Paper No. 85-0032, 1985.

13. Rogers, S. E., "A Comparison of Implicit Schemes for the Incompressible Navier-Stokes Equations and Artificial Compressibility," *AIAA Journal* , Vol. 33, No. 10, Oct. 1995

14 Beam, R. M., Warming, R. F., " An Implicit Factored Scheme for the Compressible Navier-Stokes Equations," *AIAA Journal* , Vol 16, pp. 393-401, 1978.

15 Roe, P.L., “Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes,” *Journal of Computational Physics,* Vol. 43, pp. 357-372 1981

16 Chakravarthy, S. R., Osher, S., “A New Class of High Accuracy TVD Schemes for Hyperbolic Conservation Laws,” AIAA Paper No. 85-0363, 1985.

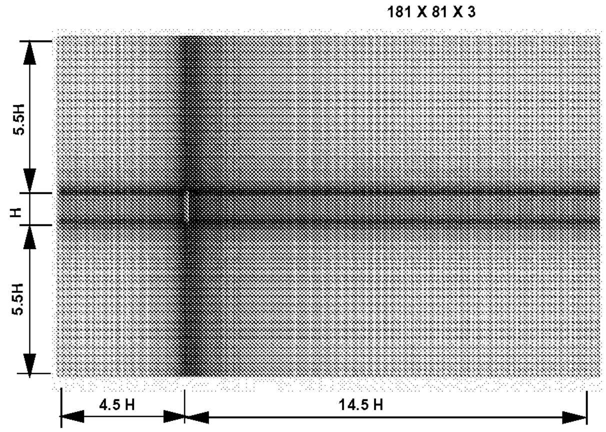

Figure 1 : Computational grid for the flow past a 90-degree flat plate. (plate tickness = 0.03H)

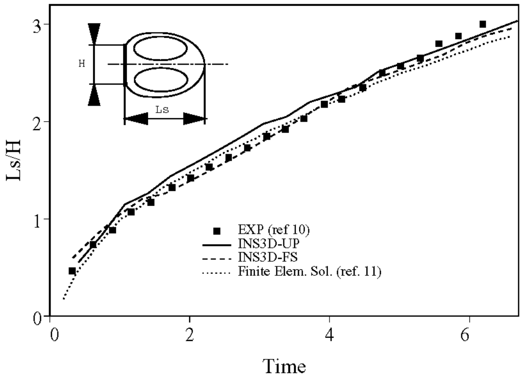

Figure 3 : Calculated time history of the stagnation point.

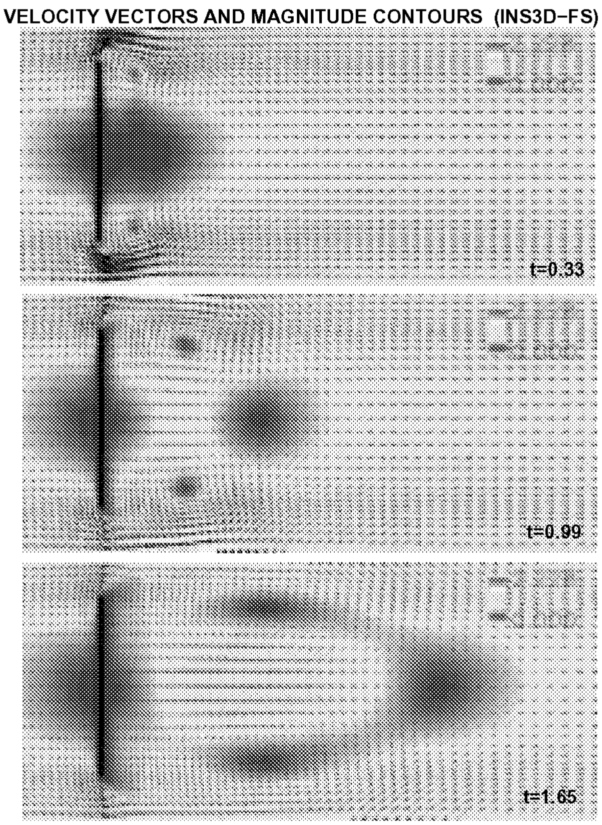

Figure 2a : Velocity vectors at various non-dimensional times (INS3D-FS).

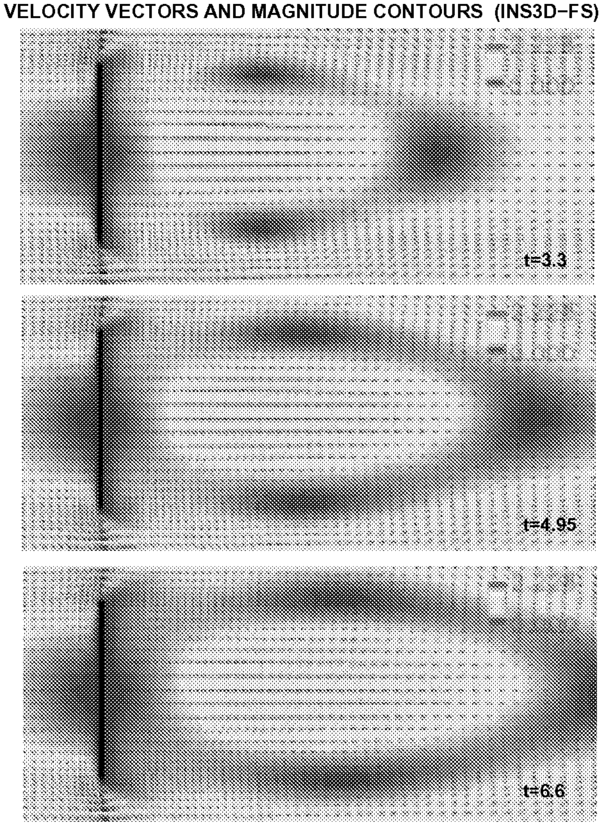

Figure 2b : Velocity vectors at various non-dimensional times (INS3D-FS).

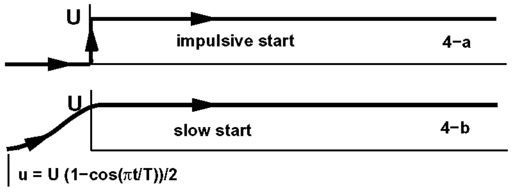

Figure 4 : Prescribed velocity for an impulsive start (a) and for a slow start(b).

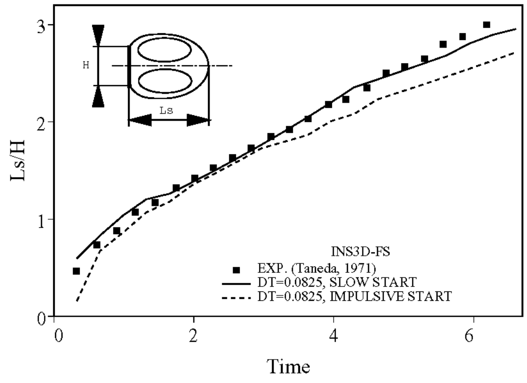

Figure 5a : Effects of starting procedure.

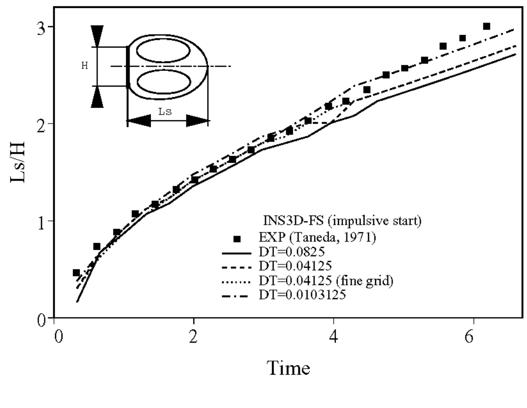

Figure 5b: Effects of time-step size for impulsive start.

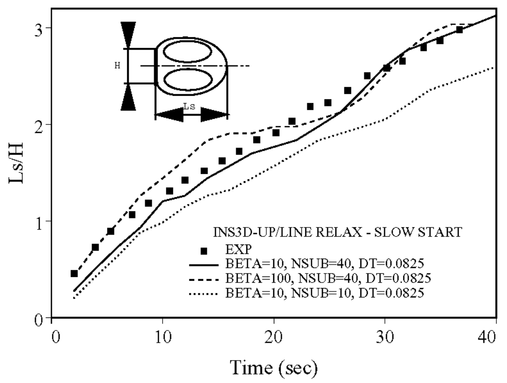

Figure 6 : Evaluation movement of stagnation point from INS3D-UP calculation with line-relaxation scheme.

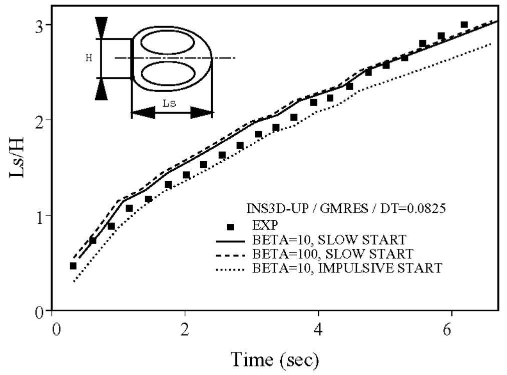

Figure 7 : Evaluation movement of stagnation point from INS3D-UP calculation with GMRES-ILU(0) scheme.

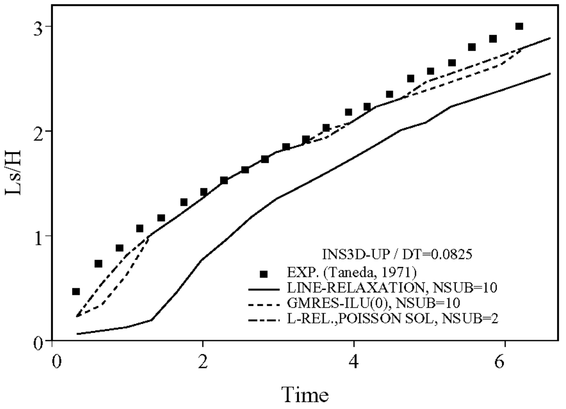

Figure 8 : Artificial compressibility results with and without Poisson equation correction.