Development of a Navier-Stokes Flow Solver for All Speeds on Unstructured Grids

Zheng Li, Hongjun Xiang

***Abstract*—A finite volume, cell-centered, density-based flow solver on unstructured grids is developed. The Weiss & Smith precondition matrix is implemented for solving incompressible and compressible flows at all speeds. The AUSMDV (Advection Upstream Splitting Method) scheme with a second order reconstruction is given for the explicit Runge-Kutta and implicit Lower-Upper Symmetric Gauss-Seidel (LU-SGS) time integration methods. Numerical simulation of inviscid flows through a channel with a bump at various Mach numbers and driven flows in a square cavity are presented to demonstrate the performance of the solver. General solution enhancement and convergence acceleration for steady-state Navier-Stokes solutions are attained via the use of inviscid or viscous preconditioning. At last, the analysis of the internal flow in a model solid rocket motor and JPL nozzle are made by the use of this solver. The ability of the solver in providing accurate steady-state solutions for transonic and low-speed flow of variable density fluids is demonstrated.**

***Index Terms*—all speeds, density based, unstructured grids, flow solver**

I. Introduction

COMPUTATIONAL fluid dynamics (CFD) technologies are widely used in design and analysis process by industry, academia, and research community [1]. The numerical flow solvers are challenged with demand to provide answers to more complex and wide ranging problems from incompressible to high-speed compressible flows. The incompressible flows were first addressed by pressure-based solution algorithms which are solved in an uncoupled manner [2]. At the same time, density-based schemes were developed in the context of transonic and supersonic aerodynamic applications [3]. These methods employ time-marching procedures that use the physical time derivatives, both implicit and explicit, to solve the hyperbolic system of governing equations.

The potential applications in flows involving both low speeds (Ma<0.1) and high speeds (Ma>1.0), e.g., a cavitating flow in a rocket engine [4], [5], motivate the modifications to the existing (density-based) codes. However, when the compressible flow solvers are applied to very low Mach number flows, it is known that most of these solvers encounter degraded convergence speeds due to huge ratio of characteristic speeds and errors arising from excessive amount of numerical dissipation. Local preconditioning techniques have been first introduced to make up this drawback. They remedy the ill-conditioned matrix equations by rescaling the eigenvalues of the governing equations. The goal of preconditioning methods is to reduce the disparity between the particle and acoustic wave speeds so that good convergence properties may be obtained at all speeds.

Three main development groups have appeared in the CFD literature for the preconditioning methods. The first group is Chorin [6] and Turkel [7], [8] who built a preconditioning method based on the artificial compressibility. The Turkel system is derived using entropy as the dependent variable. The second group including Choi, Merkle [9], Weiss, Smith [10], Venkateswaran, and Merkle [11] developed a family of preconditioners whose derivation is based on the temperature as the dependent variable. Lastly, the third group led by van Leer [12], developed a symmetric preconditioner which is referred as optimal since it equalizes the eigenvalues of the system for all Mach numbers.

In this paper we describe a finite volume, Navier-Stokes flow solver based on an unstructured grid topology that employs time-marching algorithm with Weiss-Smith preconditioning method. We will present the governing equations and the derivation of the preconditioning matrix for variable density flows, followed by a description of the spatial and temporal discretization. Afterthat, we present two results to demonstrate this methodology, including inviscid flows through a channel with a bump and driven flows in a square cavity. These examples will provide a measure of the present algorithm's accuracy and performance. Finally, this solver is applied in the computation of solid rocket motor internal flow.

II. Governing Equations

The governing equations are the time-dependent compressible Navier-Stokes equations. This system of equations, written to describe the conservation of mass, momentum and energy of the flow field, is cast in integral, Cartesian form for an arbitrary control volume V with differential surface area d$\boldsymbol{A}$ as follows:

$$\frac{\partial}{\partial t}\iiint \boldsymbol{W} dV + \iint [\boldsymbol{F} - \boldsymbol{G}] \cdot d\boldsymbol{A} = 0 \qquad (1)$$

Where

$$\boldsymbol{W} = \begin{Bmatrix} \rho \\ \rho u_x \\ \rho u_y \\ \rho u_z \\ \rho E \end{Bmatrix}, \boldsymbol{F} = \begin{Bmatrix} \rho \boldsymbol{U} \\ \rho \boldsymbol{U} u_x + p\boldsymbol{i} \\ \rho \boldsymbol{U} u_y + p\boldsymbol{j} \\ \rho \boldsymbol{U} u_z + p\boldsymbol{k} \\ \rho \boldsymbol{U} E + p\boldsymbol{U} \end{Bmatrix}, \boldsymbol{G} = \begin{Bmatrix} 0 \\ \tau_{xi} \\ \tau_{yi} \\ \tau_{zi} \\ \tau_{ij} u_j + \boldsymbol{q} \end{Bmatrix}$$

and ρ, $\boldsymbol{U}$, E, and p are the density, velocity, total energy per unit mass, and pressure of the fluid, respectively.

For a Newtonian fluid, the viscous stress is given by :

Manuscript received April 30, 2013.

Zheng Li is a Ph. D. candidate at School of Astronautics, Beijing University of Aeronautics and Astronautics (BUAA), Beijing, China. (phone: 86-010-82339476; e-mail: roy_lizheng@hotmail.com).

Hongjun Xiang is a vice Professor at School of Astronautics, Beijing University of Aeronautics and Astronautics (BUAA), Beijing, China. (e-mail: hjxiang@263.net).

$$\tau_{ij} = 2\mu S^*_{ij}$$

Where

$$S^*_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}$$

The heat flux vector, $\boldsymbol{q}$, is given by Fourier's Law:

$$q_j = -\lambda\frac{\partial T}{\partial x_j}$$

where T is the fluid temperature.

And $\boldsymbol{x} = x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k}$ is the position vector. To close these equations it is also necessary to specify an equation of state, typically of the form $\rho = \rho(p, T)$.

III. Preconditioning

A. Choice of Preconditioning Methods

In this section, comparisons have been made between the three low speed preconditioning methods introduced by Weiss-Smith, van Leer-Lee-Roe (VLR) and Turkle, respectively. The VLR preconditioner is better in convergence due to yielding the most optimal condition number (Fig. 1). In accuracy issues, all these preconditioners have the same impact. Although they are different in formula structure, they have a common point linked to the coefficient as the Mach number approach to zero. The most robust was found to be that proposed by Weiss and Smith which suffers only from the stagnation point singularity. Thus the Weiss–Smith preconditioning method is implemented in our solver.

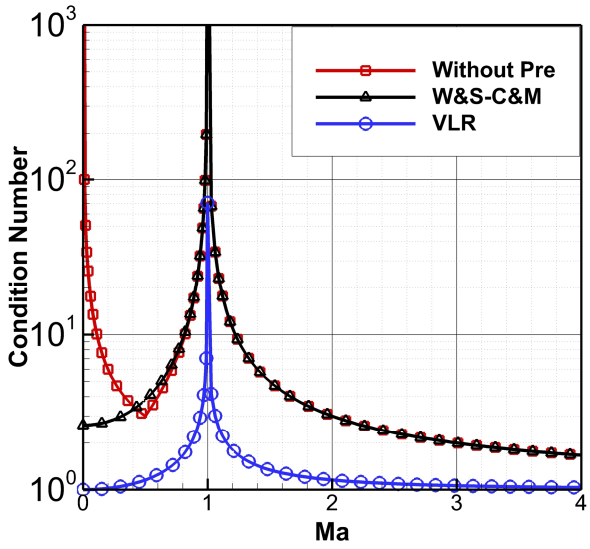


Fig. 1 Condition number variation with Mach number

B. Weiss-Smith Precondition Matrix

Preconditioning techniques involve the alteration of the time-derivatives used in time-marching CFD methods with the primary objective of enhancing their convergence. According to [10], we transform the original system of equations from the conservative variables $\boldsymbol{W}$ to the primitive variables $\boldsymbol{Q} = [p, u_x, u_y, u_z, T]^T$ as follows:

$$\frac{\partial \boldsymbol{W}}{\partial \boldsymbol{Q}}\frac{\partial}{\partial t}\iiint \boldsymbol{Q}dV + \iint[\boldsymbol{F} - \boldsymbol{G}]\cdot d\boldsymbol{A} = 0 \qquad (2)$$

The original non preconditioned Jacobian $\partial \boldsymbol{W}/\partial \boldsymbol{Q}$ matrix can be replaced to a preconditioning matrix $\boldsymbol{\Gamma}$:

$$\boldsymbol{\Gamma}\frac{\partial}{\partial t}\iiint \boldsymbol{Q}dV + \iint[\boldsymbol{F} - \boldsymbol{G}]\cdot d\boldsymbol{A} = 0 \qquad (3)$$

$$\Gamma = \begin{pmatrix} \Theta & 0 & 0 & 0 & \rho_T \\ \Theta u_x & \rho & 0 & 0 & \rho_T u_x \\ \Theta u_y & 0 & \rho & 0 & \rho_T u_y \\ \Theta u_z & 0 & 0 & \rho & \rho_T u_z \\ \Theta H - 1 & \rho u_x & \rho u_y & \rho u_z & \rho_T H + \rho C_p \end{pmatrix}$$

Where Θ is given by

$$\Theta = \left(\frac{1}{{U_r}^2} - \frac{\rho_T}{\rho C_p}\right)$$

Here U_r is a reference velocity.

$$\boldsymbol{U}_r = \begin{cases} \epsilon c, if\ |\boldsymbol{U}| < \epsilon c \\ |\boldsymbol{U}|, if\ \epsilon c < |\boldsymbol{U}| < c \\ c, if\ |\boldsymbol{U}| > c \end{cases}$$

For the viscous low Reynolds number flows the reference velocity should not be smaller than the local diffusion velocity $\boldsymbol{\nu}/\Delta x$. Thus,

$$\boldsymbol{U}_r = \max(\boldsymbol{U}_r, \frac{\boldsymbol{\nu}}{\Delta x})$$

The resultant eigenvalues of the preconditioned system are given by

$$\lambda\left(\Gamma^{-1}\frac{\partial \boldsymbol{F}}{\partial \boldsymbol{Q}}\right) = \boldsymbol{U}, \boldsymbol{U}, \boldsymbol{U}, \boldsymbol{U}' + c', \boldsymbol{U}' - c'$$

where

$$\boldsymbol{U}' = \boldsymbol{U}(1-\alpha)$$
$$c' = \sqrt{\alpha^2 u^2 + U_r^2}$$
$$\alpha = (1 - \beta U_r^2)/2$$
$$\beta = \left(\rho_p + \frac{\rho_T}{\rho C_p}\right)$$

we can see that all eigenvalues remain of the order of u as long as the reference velocity is of the same order as the local velocity.

IV. Solution Precedure

A. Data Structure and Grid Entry

Contrary to structured solvers, the unstructured flow solver is much complex in data structure and grid entry due to its indirect data addressing. The procedure here is written in C++ with a set of classes to describe vertex, face, cell and their link. As a cell-centered solver, the system stores the flow field data in cell object. The face object contains the pointer of its two adjoining cells (Fig. 2), while the two cells store the face pointer. This is a fast and efficient way for cell objects to store the connection to its neighbor cell object. The data members of cell class are shown in Table I. The solver can input the mesh generated by Gambit which is easier in partitioning grid over geometrically complex domains.

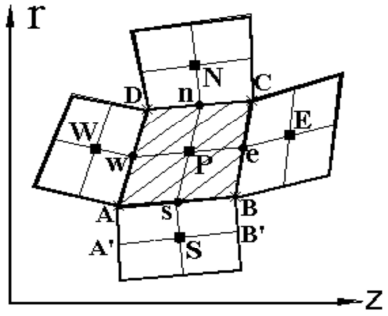


Fig. 2 Control volume

TABLE I
DATA MEMBERS OF CELL CLASS

Type	Name	Description
int	ID	The ID of the cell.
double	Geometry	Geometry information of the cell.
List<Vertex*>	Vertex_ptr	The list of Vertex pointers to the vertex.
List<Face*>	Face_ptr	The list of Face pointers to the face.
double	Parameters	Flow field information of the cell.

B. Spatial Discretization

The preconditioned governing equations are discretized spatially using a finite volume scheme wherein the physical domain is subdivided into small (nondeforming) hexahedron volumes and integral equations are applied to each cell. The discrete, inviscid flux vectors appearing in (1) are evaluated by AUSMDV (Advection Upstream Splitting Method) scheme with a modification to operate effectively at low Mach numbers. The schemes introduced by Liou have been discussed more thoroughly in [13] and [14].

C. Reconstruction

The solution vector $\boldsymbol{Q}$ used to evaluate the fluxes at cell faces is computed using a multidimensional linear reconstruction approach. In this approach, higher order accuracy is achieved at cell faces through a Taylor series expansion of the cell-averaged solution vector about the cell centroid:

$$Q_f = Q_c + \psi \nabla Q \cdot r \tag{4}$$

where r is the displacement vector from the cell centroid to the face centroid. This formulation requires determining the solution gradient ∇Q in each cell.

The gradients are computed in the solver according to the Green-Gauss and Least Squares method. [1] describes the Green-Gauss method, so this paper only presents the Least Squares method. In this method the solution is assumed to vary linearly.

For cell C, cell 1,2,... J is its neighbor, so:

$$Md = \Delta Q \tag{5}$$

where

$$M = \begin{bmatrix} x_1 - x_c & y_1 - y_c & z_1 - z_c \\ x_2 - x_c & y_2 - y_c & z_2 - z_c \\ \vdots & \vdots & \vdots \\ x_J - x_c & y_J - y_c & z_J - z_c \end{bmatrix}$$

$$d = \begin{bmatrix} \partial Q/\partial x \\ \partial Q/\partial y \\ \partial Q/\partial z \end{bmatrix}_C \quad \Delta Q = \begin{bmatrix} Q_1 - Q_C \\ Q_2 - Q_C \\ \vdots \\ Q_J - Q_C \end{bmatrix}$$

The cell gradient $\nabla Q = \frac{\partial Q}{\partial x}\boldsymbol{i} + \frac{\partial Q}{\partial y}\boldsymbol{j} + \frac{\partial Q}{\partial z}\boldsymbol{k}$ is determined by solving the minimization problem for the system of the non-square coefficient matrix in a least-squares sense:

$$M^T M d = M^T \Delta Q \tag{6}$$

Finally, the gradients ∇Q are limited by the coefficient ψ so that they do not introduce new maxima or minima into the reconstructed data. The Venkatakrishnan limiter [15] is implemented on this code.

D. Temporal Discretization

An explicit multistage time-stepping scheme and an implicit scheme are implemented on the preconditioned system using the conservative variables of unknowns.

1) Explicit Scheme

The explicit m-stage Runge-Kutta (R-K) scheme which solves the equation from time t to time $t + \Delta t$ is given by

$$Q^0 = Q_t \tag{7}$$

$$Q^i = Q^0 - \alpha_i \Delta t \Gamma^{-1} R^{(i-1)} \tag{8}$$

$$Q_{t+\Delta t} = Q^{(m)} \tag{9}$$

where $i = 1,2,3,\ldots m$ is the stage counter for the m-stage scheme and α_i is the multistage coefficient for the ith stage.

2) Implicit Scheme

The implicit Lower-Upper Symmetric Gauss-Seidel (LU-SGS) scheme for (1), see [16], can be described as:
Forward sweep:

$$\Gamma \Delta Q_i^* = D^{-1}\left[Res_i - 0.5\sum_{j:j\in \mathrm{L(i)}}\left(s_{ij}\cdot \Delta F_j^* - \left|s_{ij}\right|\rho_{A_j}\Gamma \Delta Q_j^*\right)\right] \tag{10}$$

Backward sweep:

$$\Gamma \Delta Q_i = \Gamma \Delta Q_i^* - 0.5 D^{-1}\sum_{j:j\in \mathrm{U(i)}}\left(s_{ij}\cdot \Delta F_j - \left|s_{ij}\right|\rho_{A_j}\Gamma \Delta Q_j\right) \tag{11}$$

$$\mathrm{Res}_i = -\sum_{j(i)} s_{ij}\cdot F_{ij}$$

Where D is the diagonal matrix

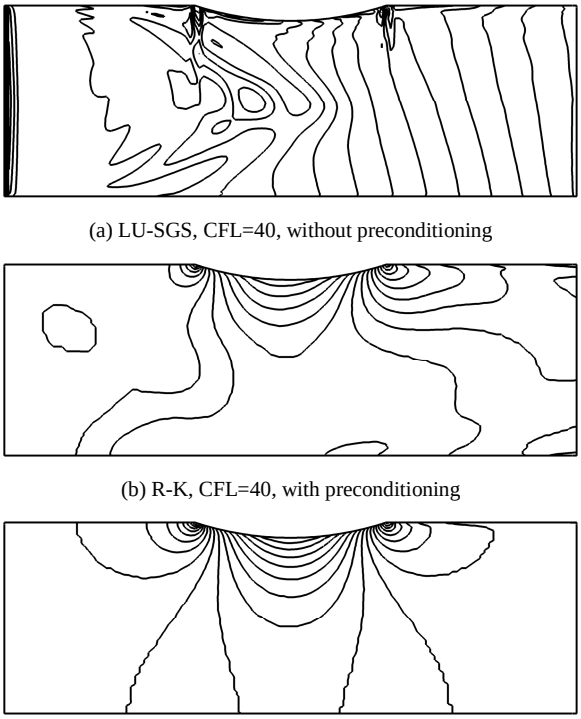
$$\mathrm{D} = \left(\frac{Vol_i}{\Delta \mathrm{t}} + 0.5\sum_{j(i)}\rho_{A_i}\left|s_{ij}\right|\right) I$$

V. RESULT

In this section, the preconditioning implementation for the unstructured flow solver has been assessed using some representative cases. The test cases chosen for this purpose include inviscid flow past a bump in a channel, laminar driven flow in square cavity and planar supersonic nozzle. The two dimensional test cases have been geometrically represented as three-dimensional cases by simply extending the geometry in the spanwise direction. The first result will show the performance of the present method for computing inviscid flows at various speeds. The second case will demonstrate the importance of preconditioning for solving low speed flows of variable density fluids. And in the last two cases we demonstrate the performance of the solver in the internal flows of solid rocket motor.

A. Inviscid Flow Past a Bump in a Channel

This test case validates the implementation of preconditioning for the inviscid flow over a 10% circular arc bump at various speeds. The symmetrical computation domain is 3×1 with a 129×65 grid partition. Fig. 3 shows the results in different methods at 0.01 Mach number over 4000 iteration cycles. Without preconditioning, the LU-SGS result is completely unphysical (Fig. 3a). The solution provided by R-K method with preconditioning (Fig. 3b) shows marked improvement but not well enough because of the limitation of CFL number. This drawback is corrected by using the LU-SGS implicit scheme (Fig. 3c). Fig. 4 provides an indication of the behavior of implementations of LU-SGS with preconditioning method for a range of Mach number. Fig. 5 and 6 highlights the convergence history in various conditions. Good efficiency across the Mach number range is obtained with the preconditioning method.

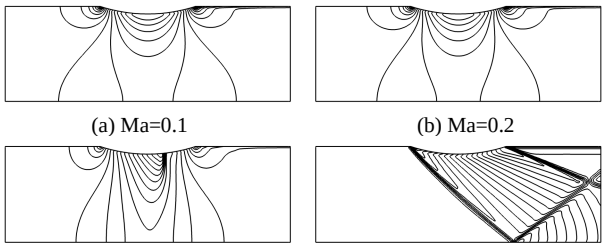


(a) LU-SGS, CFL=40, without preconditioning

(b) R-K, CFL=40, with preconditioning

(c) LU-SGS, CFL=40, with preconditioning

Fig. 3 Density contour at Ma=0.01



(a) Ma=0.1 (b) Ma=0.2

(c) Ma=0.7 (d) Ma=2.0

Fig. 4 Mach number contour at various speeds

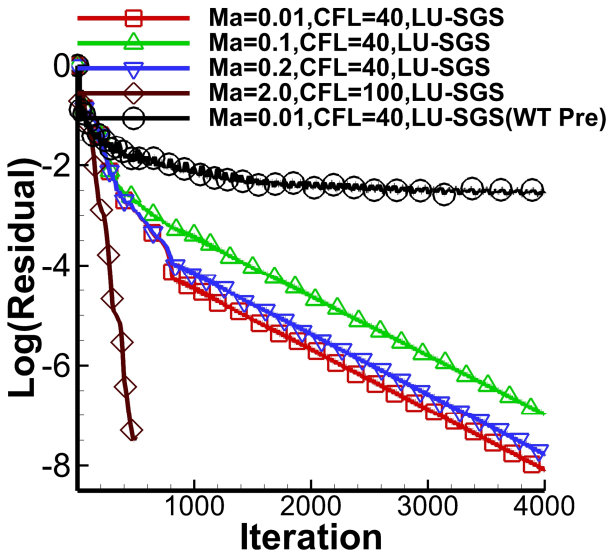


Fig. 5 Convergence histories for LU-SGS at various speeds

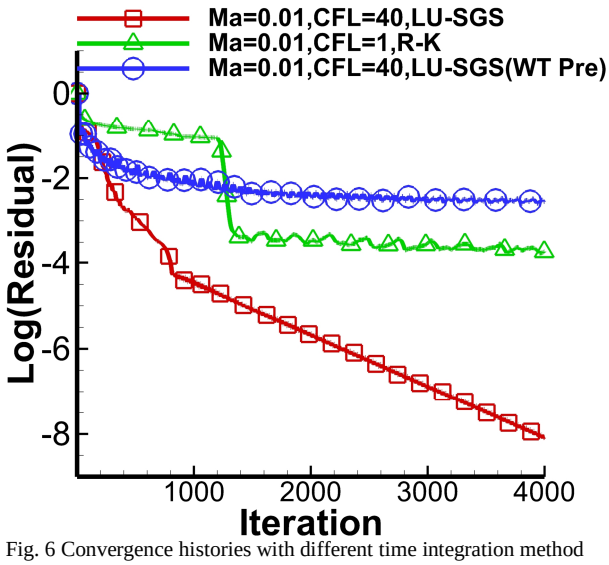


Fig. 6 Convergence histories with different time integration method

B. Driven Flows in Square Cavity

This test case, representing a 2-dimensional laminar incompressible flow in a square cavity, has been numerical investigated in detail by Ghia [17] et al. The computation has been made with a uniform velocity at the top wall of the cavity. A set of Re numbers (100, 400, and 1000) are considered in this case. Fig. 7 and 8 shows the distribution of velocity through geometric center of the cavity. Fig. 9 gives the streamline pattern at 100 and 1000 Re number. The results show great agreement with the numerical simulations of Ghia. The convergence history is highlighted in Fig. 10. The ability in computation of laminar incompressible flow is demonstrated.

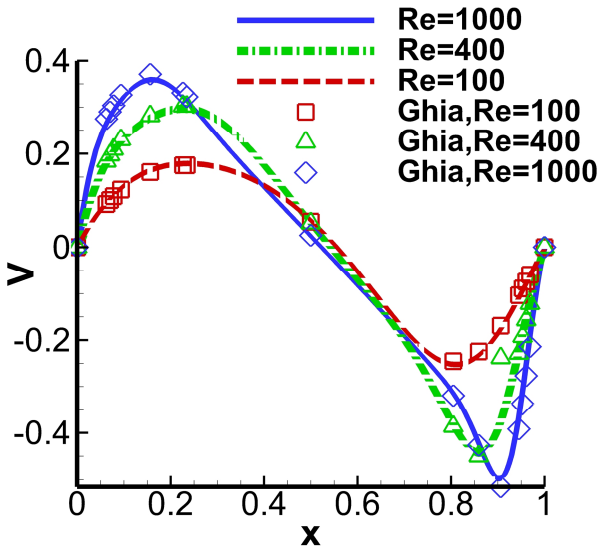


Fig. 7 v-velocity along horizontal line through geometric center of cavity

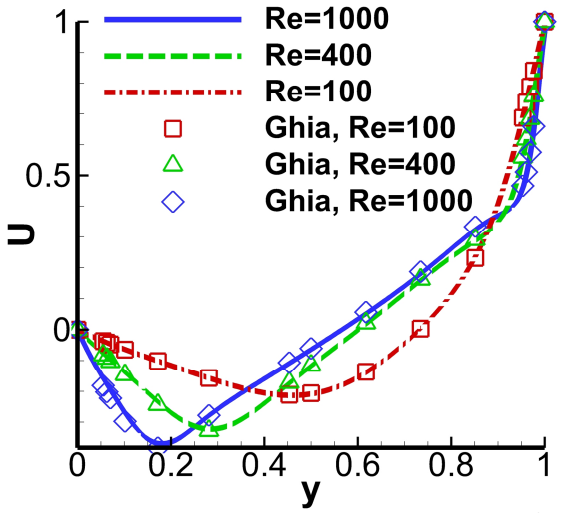
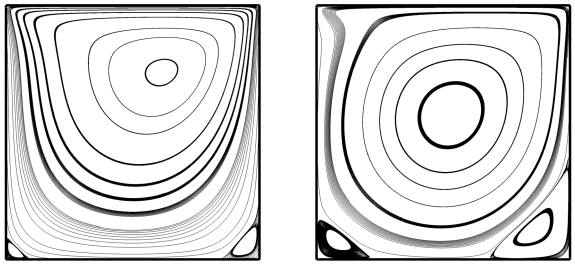


Fig. 8 u-velocity along vertical line through geometric center of cavity



(a) Re=100 (b) Re=1000

Fig. 9 Streamline pattern

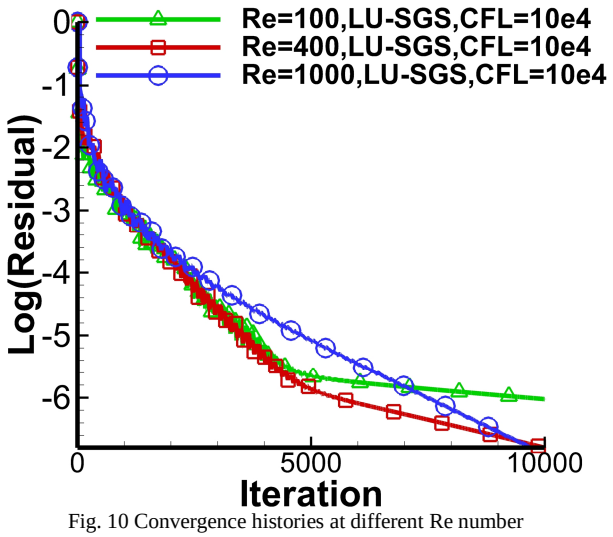


Fig. 10 Convergence histories at different Re number

C. Model Solid Rocket Motor

In this section, numerical simulation has been made for the internal flow field of a model solid rocket motor, including the combustion and nozzle. The solid propellant boundary is mass flow inlet. The nozzle exit pressure is 1 atm and the combustion pressure is 4 MPa. The geometry is shown in Fig. 11. The grid is all-triangles, consisting of 4968 elements. The global flow structure of the Model SRM is illustrated by Mach number in Fig. 12. Both incompressible flow (in combustion) and compressible flow (at nozzle) are obtained in this result. Fig. 13 indicates the streamline of the internal flow field. We can see that there is a low speed vortex in the head of the combustion.

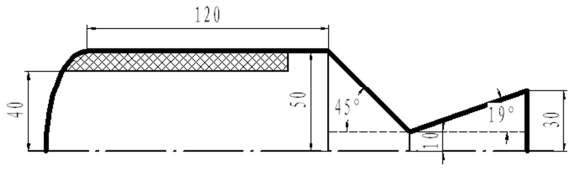


Fig. 11 Model solid rocket motor geometry (mm)

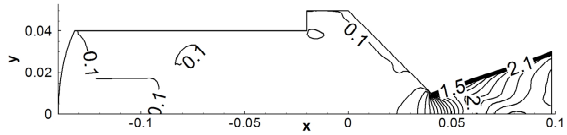


Fig12 Mach number contour

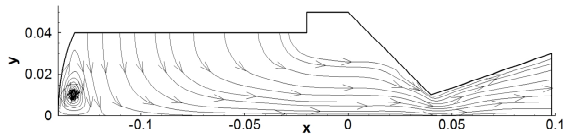


Fig 13 Streamline pattern in SRM

D. JPL Nozzle

The compressible flow inside the JPL axisymmetric convergent-divergent nozzle, shown in Fig. 14, with a 45˚ entrance and a 15˚ exit straight wall tangent to a circular throat is a classic nozzle flow problem, which has been is experimentally investigated in [18]. The tests are conducted with air at a stagnation pressure of 70 psia and a stagnation temperature of 540 R. The structure of the flow field is represented by temperature in Fig.15. Fig. 16 indicates the pressure distribution at wall and the axis. Comparisons with the experimental data show good agreement.

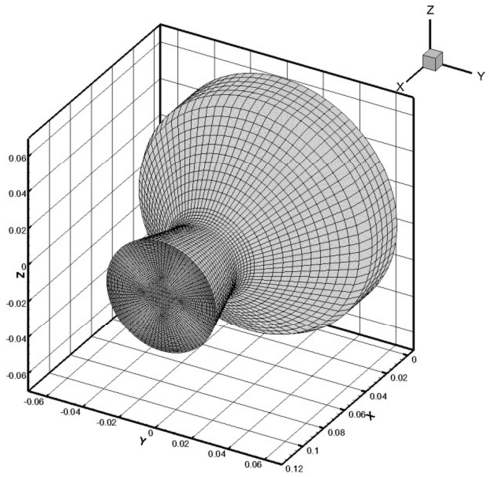


Fig. 14 JPL Nozzle and computation mesh

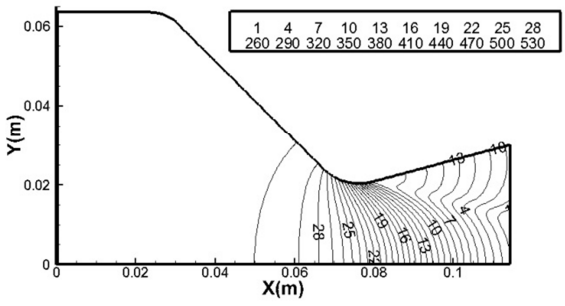


Fig. 15 Temperature contour

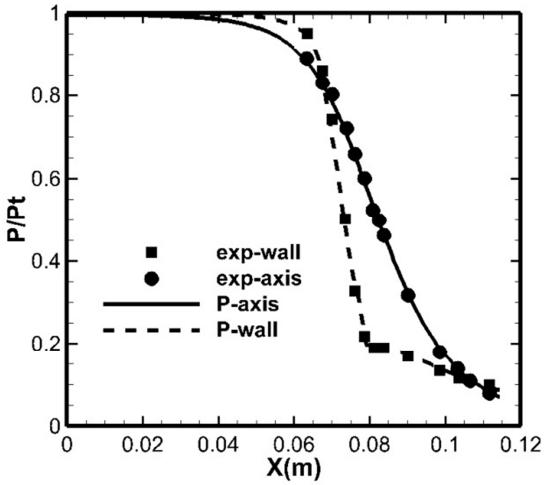


Fig. 16 Pressure distribution

VI. Conclusion

In this paper, time-derivative low-speed preconditioning of the Navier-Stokes equations, suitable for flows at all speeds has been successfully implemented on an unstructured flow solver. The AUSMDV scheme with a second order upwind-biased reconstruction is presented to accommodate the preconditioned eigenvalues and eigenvectors. Both explicit and implicit schemes are developed to march the solution of the preconditioned system to steady-state. To demonstrate the accuracy and efficiency of the present algorithm two test cases were presented. General convergence enhancement is demonstrated with the use of preconditioning on the Navier-Stokes equations at various speeds. Finally, the analysis of rocket internal flow is implemented by the solver.

References

[1] Z. Li and H. Xiang, "The Development of a Navier-Stokes Flow Solver with Preconditioning Method on Unstructured Grids", Lecture Notes in Engineering and Computer Science: Proceedings of The International MultiConference of Engineers and Computer Scientists 2013, 13-15 March, 2013, Hong Kong, pp1168-1172.

[2] M. ur Rehman, C.Vuik and G. Segal, "Preconditioners for the Steady Incompressible Navier-Stokes Problem", IAENG International Journal of Applied Mathematics, 38:4.

[3] Ong J. Chit, Ashraf A. Omar and C. V. Rorst, "Reynolds Averaged Navier-Stokes Flow Computation of RAE2822 Airfoil Using Gas-kinetic BGK Scheme", Lecture Notes in Engineering and Computer Science: Proceedings of The International MultiConference of Engineers and Computer Scientists 2009, 18-20 March, 2009, Hong Kong.

[4] C. C. Kiris, D. Kwak, W. Chan, and J. A. Housman, High-fidelity simulations of unsteady flow through tubopumps and flowliners, Comput. Fluids., 37 (2008), 536–546.

[5] E. Shima and K. Kitamura, On New Simple Low-Dissipation Scheme of AUSM-Family for All Speeds, 47tn AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition 5-8 January, 2009 Orlando, Florida.

[6] Chorin A. A numerical method for solving incompressible viscous flows problems [J]. J Comput Phys 1967:2:12-26.

[7] Turkel E. Preconditioning techniques in computational fluid dynamics [J]. Ann Rev Fluid Mech 1999:31:385-416.

[8] Turkel E, Radespiel R, Kroll N. Assessment of preconditioning methods for multidimensional aerodynamics [J]. Comput Fluids 1997:26:613-34.

[9] Choi Y, Merkle C. The application of preconditioning in viscous flows [J]. J Comput Phys 1993:105:207-23.

[10] Weiss J. Smith W. Preconditioning applied to variable and constant density flows [J]. AIAA J 1995:33:2050-6.

[11] Venkateswaran S. Merkle L. Analysis of preconditioning methods for the Euler and Navier-Stokes equations. Von Karman Institute for Fluid Dynamics, 1999.

[12] van Leer B, Lee W, Roe P. Characteristic time-stepping or local preconditioning of the Euler equations [J]. AIAA paper 91-1552:1991.

[13] Liou, M.-S. Further Progress in Numerical Flux Scheme, Proceedings of the 15th International Confer-ence on Numerical Methods in Fluid Dynamics, June 1996.

[14] Edwards J, Liou M. Low-diffusion flux-splitting methods for flows at all speeds [J]. AIAA-97-1862.

[15] Venkatakrishnan V. On the accuracy of limiters and convergence to steady state solutions[C]. AIAA 93-0880.

[16] Sharov D, Nakahashi K. Low speed preconditioning and LU-SGS scheme for 3-D viscous flow computations on unstructured grids[C]. AIAA 98-0614.

[17] Ghia U. High-Re Solutions for Incompressible Flow Using the Navier-Stokes Equations and a Multigrid Method [J]. J Comput Phys，1982:48:387-411.

[18] R. F. Cuffel, L. H. Back and P. F. Massier, Transonic Flowfield in a Supersonic Nozzle with Small Throat Radius of Curvature, AIAA Journal, 1969.