AIAA JOURNAL
Vol. 42, No. 9, September 2004

# Overrelaxation Applied to Lower–Upper Symmetric Gauss–Seidel Method for Hypersonic Flows

Satoru Yamamoto* and Daigo Sato†
*Tohoku University, Sendai 980-8579, Japan*

**An implicit time-marching method for acceleration of the convergence of solution for the compressible Navier–Stokes equations is presented. This is based on the lower–upper symmetric-Gauss–Seidel (LU-SGS) method proposed by Yoon and Jameson (Yoon, S., and Jameson, A., "Lower–Upper Symmetric-Gauss–Seidel Method for the Euler and Navier–Stokes Equations," *AIAA Journal*, Vol. 26, 1988, pp. 1025–1026). A very simple overrelaxation is additionally executed in the LU-SGS method coupled with diagonal flux splitting. Then, an overrelaxation parameter $\omega$, a parameter similar to that in the well-known successive-overrelaxation method is introduced. The calculated results for a hypersonic viscous flows around a cylinder and through a compression corner indicate that the convergence is accelerated by a proper value of $\omega(1 < \omega < 2)$. This method is also applied to a hypersonic thermochemical nonequilibrium flow around a sphere.**

## Introduction

THE lower–upper symmetric Gauss–Seidel (LU-SGS) method has been proposed by Yoon and Jameson.[1] This method is efficient and robust for solving high-speed flow problems. Because no matrix inversion is necessary in this algorithm and a large Courant–Friedrichs–Lewy (CFL) number can be taken for the time-marching calculation, the CPU time can be relatively shorter compared with conventional implicit methods. In this paper, an overrelaxation parameter $\omega$, originally implemented in the successive overrelaxation (SOR) method by Frankel,[2] is introduced in the LU-SGS algorithm to further save CPU time. A flux-splitting form derived as a diagonal form is also derived for calculation of nondiagonal implicit terms in the LU-SGS method. The present method may be easily applicable to more complicated flow equations, such as chemical reacting flows, condensate flows, and magnet–plasma–dynamic flows by changing only the set of subvectors in the flux-splitting form, as reported by Yamamoto.[3] Three typical two-dimensional hypersonic viscous flow problems, such as a flow around a cylinder, a flow through a compression corner, and a thermochemical nonequilibrium flow around a sphere are calculated by the use of the present method. Finally, the convergence of a solution derived by the use of the overrelaxation parameter $\omega(1 < \omega < 2)$ is compared with that obtained by $\omega = 1$.

## Fundamental Equations

The compressible Navier–Stokes equations can be written in the following vector form in curvilinear coordinates:

$$\frac{\partial Q}{\partial t} + \mathcal{F}(Q) = \frac{\partial Q}{\partial t} + \frac{\partial F_i}{\partial \xi_i} + S = 0$$

$$Q = J\begin{bmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ e \end{bmatrix}, \qquad F_i = J\begin{bmatrix} \rho U_i \\ \rho u_1 U_i + \partial \xi_i / \partial x_1 p \\ \rho u_2 U_i + \partial \xi_i / \partial x_2 p \\ (e+p)U_i \end{bmatrix}$$

$$S = -J\frac{\partial \xi_i}{\partial x_j}\frac{\partial}{\partial \xi_i}\begin{bmatrix} 0 \\ \tau_{1j} \\ \tau_{2j} \\ \tau_{kj}u_k + \kappa \partial T/\partial x_j \end{bmatrix} \tag{1}$$

$\boldsymbol{Q}$ is the vector of unknown variables; $\boldsymbol{F}_i$ and $S$ are the flux vector and the diffusion term; and $t$, $x_i$, $\xi_i$, $\rho$, $u_i$, and $e$ are the time, the components of the Cartesian coordinate, the components of the general curvilinear coordinate, the density, the Cartesian components of the physical velocity, and the total internal energy per unit volume, respectively. $U_i$, $p$, $T$, $\kappa$, $\tau_{ij}$, and $J$ are the components of the contravariant velocity, the static pressure, the static temperature, the heat conductivity, the components of the viscous stress, and the Jacobian for transformation, respectively.

## Numerical Method

### LU-SGS Algorithm

Yoon and Jameson[1] have proposed a new LU-SGS method, in which no inverse of matrices must be calculated. The algorithm is explained by the use of the following two-dimensional implicit $\Delta$ form:

$$[I + \theta \Delta t(\partial/\partial \xi A + \partial/\partial \eta B)]\Delta Q^n = \text{RHS}$$

$$\text{RHS} = -\Delta t(F_\xi + G_\eta + S)^n \tag{2}$$

with RHS indicating right-hand side. Here, $\xi = \xi_1$ and $\eta = \xi_2$; $\theta = 1$ or $\frac{1}{2}$; $F = AQ = F_1$ and $G = BQ = F_2$. $A$ and $B$ are the Jacobian matrices in $\xi$ and $\eta$ directions. The upwinding according to the sign of characteristic speeds is applied into the left-hand side of Eq. (2). Then,

$$\left[I + \theta \Delta t\left(\nabla_\xi A^+ + \Delta_\xi A^- + \nabla_\eta B^+ + \Delta_\eta B^-\right)\right]\Delta Q^n = \text{RHS} \tag{3}$$

where, the following relations are defined:

$$\nabla_\xi A^+ \Delta Q^n = A^+_{i,j}\Delta Q^n_{i,j} - A^+_{i-1,j}\Delta Q^n_{i-1,j}$$

$$\Delta_\xi A^- \Delta Q^n = A^-_{i+1,j}\Delta Q^n_{i+1,j} - A^-_{i,j}\Delta Q^n_{i,j}$$

$$\nabla_\eta B^+ \Delta Q^n = B^+_{i,j}\Delta Q^n_{i,j} - B^+_{i,j-1}\Delta Q^n_{i,j-1}$$

$$\Delta_\eta B^- \Delta Q^n = B^-_{i,j+1}\Delta Q^n_{i,j+1} - B^-_{i,j}\Delta Q^n_{i,j} \tag{4}$$

The subscripts $i$ and $j$ indicate a grid point in the $\xi$ and $\eta$ directions. These equations are substituted into Eq. (3). Then, Eq. (3) is

Received 18 July 2001; accepted for publication 9 February 2004. 

*Associate Professor, Department of Aeronautics and Space Engineering. Member AIAA.
†Graduate Student, Department of Aeronautics and Space Engineering.

rewritten as

$$\left[I+\theta\Delta t\left(A^+_{i,j}-A^-_{i,j}+B^+_{i,j}-B^-_{i,j}\right)\right]\Delta Q^n+\theta\Delta tG(\Delta Q^n)=\mathrm{RHS} \tag{5}$$

$$G(\Delta Q^n)=-(A^+\Delta Q^n)_{i-1,j}+(A^-\Delta Q^n)_{i+1,j} -(B^+\Delta Q^n)_{i,j-1}+(B^-\Delta Q^n)_{i,j+1} \tag{6}$$

In the LU-SGS method, $A^{\pm}_{i,j}$ and $B^{\pm}_{i,j}$ in Eq. (5) are approximately defined by

$$A^{\pm}_{i,j}=[A_{i,j}\pm r(A_{i,j})I]/2, \qquad B^{\pm}_{i,j}=[B_{i,j}\pm r(B_{i,j})I]/2 \tag{7}$$

where $A_{i,j}$ and $B_{i,j}$ are the Jabcobian matrices at the grid point $(i, j)$. The function $r(A)$ is given by

$$r(A)=\alpha\max[\lambda(A)] \tag{8}$$

where $\alpha\geq 1$. Here, $\lambda(A)$ indicates the spectral radius of the Jabcobian matrix $A$. Actually, this is given by the eigenvalues of $A$. Because the relations

$$A^+_{i,j}-A^-_{i,j}=r(A_{i,j}), \qquad B^+_{i,j}-B^-_{i,j}=r(B_{i,j}) \tag{9}$$

are found, Eq. (5) is further rewritten by

$$\{I+\theta\Delta t[r(A_{i,j})+r(B_{i,j})]\}\Delta Q^n+\theta\Delta tG(\Delta Q^n)=\mathrm{RHS} \tag{10}$$

Note that the first term of the left-hand side in Eq. (10) is reduced to four sets of individual diagonal matrices. If the diagonal operator is defined by

$$D=I+\theta\Delta t[r(A_{i,j})+r(B_{i,j})] \tag{11}$$

Eq. (10) is rewritten as

$$D\Delta Q^n+\theta\Delta tG(\Delta Q^n)=\mathrm{RHS} \tag{12}$$

This equation calculated with the Gauss–Seidel relaxation method is divided into two sweeps in each time step, as follows:

$$D\Delta Q^*=\mathrm{RHS}+\theta\Delta tG^+(\Delta Q^*) \tag{13}$$

$$\Delta Q^n=\Delta Q^*-D^{-1}\theta\Delta tG^-(\Delta Q^n) \tag{14}$$

$$G^+(\Delta Q^*)=(A^+\Delta Q^*)_{i-1,j}+(B^+\Delta Q^*)_{i,j-1} \tag{15}$$

$$G^-(\Delta Q^n)=(A^-\Delta Q^n)_{i+1,j}+(B^-\Delta Q^n)_{i,j+1} \tag{16}$$

**Overrelaxation Algorithm**

The RHS of Eq. (2) calculates the fundamental equations of Eq. (1) explicitly. This calculation has a physical meaning. On the other hand, the calculation in the left-hand side of Eq. (2) is executed implicitly. This calculation does not have a physical meaning. Therefore, the residue in the left-hand side must be zero when the residue of the RHS reaches zero. In contrast, the algorithm in the left-hand side could be arbitrarily constructed if the residue of the RHS only needed to decrease.

In this paper, we introduce an overrelaxation technique into the LU-SGS method to accelerate the convergence of solution in the RHS. It is known that overrelaxation was proposed by Frankel[2] to accelerate the convergence of the Gauss–Seidel method. Then, an overrelaxation parameter $\omega(1<\omega<2)$ is introduced. The optimum value of $\omega$ can be obtained theoretically in a simple problem, such as a boundary value problem in rectangular coordinates. However, this values should be determined by trial and error in more complicated problems.

A very simple overrelaxation is further applied to Eq. (12) by

$$D\Delta Q^n+D\Delta Q^n+\omega[\theta\Delta tG(\Delta Q^n)-D\Delta Q^n]=\mathrm{RHS} \tag{17}$$

where $\omega$ is an overrelaxation parameter. This equation is calculated by the use of the same procedure with Eqs. (13) and (14) in each time step as

$$(2-\omega)D\Delta Q^*=\mathrm{RHS}+\omega\theta\Delta tG^+(\Delta Q^*) \tag{18}$$

$$\Delta Q^n=\Delta Q^*-[(2-\omega)D]^{-1}\omega\theta\Delta tG^-(\Delta Q^n) \tag{19}$$

where $\omega$ should be at least $1<\omega<2$. The present method, as written in Eqs. (18) and (19) is called the LU-SGS-$\omega$ method hereafter. Another description can be introduced by

$$[D+(1-\omega)I]\Delta Q^*=\mathrm{RHS}+\omega\theta\Delta tG^+(\Delta Q^*) \tag{20}$$

$$\Delta Q^n=\Delta Q^*-[D+(1-\omega)I]^{-1}\omega\theta\Delta tG^-(\Delta Q^n) \tag{21}$$

The LU-SGS-$\omega$ method can be easily extended to a maximum second-order time-accurate method by the introduction of the Newton iteration and the Crank–Nicolson method (see Ref. 4) as

$$(2-\omega)D\Delta Q^{*m}=\mathrm{RHS}^m+\omega\theta\Delta tG^+(\Delta Q^{*m}) \tag{22}$$

$$\Delta Q^m=\Delta Q^{*m}-[(2-\omega)D]^{-1}\omega\theta\Delta tG^-(\Delta Q^m) \tag{23}$$

where

$$\Delta Q^m=Q^{m+1}_{i,j}-Q^m_{i,j}$$

$$\mathrm{RHS}^m=-\left(Q^m_{i,j}-Q^n_{i,j}\right)-\Delta t\left[\mathcal{F}\left(Q^m_{i,j}\right)+\mathcal{F}\left(Q^n_{i,j}\right)\right]/2$$

where $m$ is the Newton iteration number. If $\Delta Q^m\to 0$ as $m\to\infty$, then a maximum second-order time-accurate solution can be obtained.

**Calculation of $G^{\pm}$**

Now, $A^+$, $A^-$, $B^+$, and $B^-$ in Eqs. (15) and (16) are transformed into $A^+_1$, $A^-_1$, $A^+_2$, and $A^-_2$. In the original LU-SGS method by Yoon and Jameson, $A^{\pm}_k$ is approximately calculated by the following equation:

$$A^{\pm}_k=[A_k\pm r(A_k)I]/2 \tag{24}$$

On the other hand, in this paper, $A^{\pm}_k\Delta Q$ is derived as a set of subvectors of the diagonal flux-splitting form written by

$$A^{\pm}_k\Delta Q=\lambda^{\pm}_{k1}\Delta Q+\left(\lambda^{\pm}_{ka}\Big/c\sqrt{g_{kk}}\right)Q_{ka}+\left(\lambda^{\pm}_{kb}\Big/c^2\right)Q_{kb}$$

$$\lambda_{k1}=U_k, \qquad \lambda_{k3}=U_k+c\sqrt{g_{kk}}, \qquad \lambda_{k4}=U_k-c\sqrt{g_{kk}}$$

$$\lambda^{\pm}_{k\ell}=(\lambda_{k\ell}\pm|\lambda_{k\ell}|)/2, \qquad \lambda^{\pm}_{ka}=\left(\lambda^{\pm}_{k3}-\lambda^{\pm}_{k4}\right)\Big/2$$

$$\lambda^{\pm}_{kb}=\left(\lambda^{\pm}_{k3}+\lambda^{\pm}_{k4}\right)\Big/2-\lambda^{\pm}_{k1}$$

$$Q_{ka}=\bar{p}Q_{kc}+\Delta\bar{m}_kQ_d, \qquad Q_{kb}=\Delta\bar{m}_kc^2Q_{kc}/g_{ii}+\bar{p}Q_d$$

$$\bar{p}=Q_p\cdot\Delta Q, \qquad \Delta\bar{m}_k=Q_{km}\cdot\Delta Q \tag{25}$$

where $g_{kk}=\nabla\xi_k\cdot\nabla\xi_k$. Equation (25) is accurate for upwinding according to the sign of characteristic speeds. Because Eq. (25) is written in a general form, Eqs. (18) and (19) can be easily applied to a system of more complicated flow equations only if the subvectors, $Q_{ic}$, $Q_p$, $Q_{im}$, and $Q_d$ are introduced. The subvectors for equations of thermochemical nonequilibrium flows, condensate flows, and the magnet–plasma–dynamics flows, have been already derived by Yamamoto.[3] In this paper, the following subvectors, $Q_{ic}$, $Q_p$, $Q_{im}$, and $Q_d$, are used for the compressible Navier–Stokes equations:

$$Q_{kc}=[0 \quad \partial\xi_k/\partial x_1 \quad \partial\xi_k/\partial x_2 \quad U_k]^T \tag{26}$$

$$Q_p=[-\tilde{\gamma}u_\ell u_\ell/2 \quad -\tilde{\gamma}u_1 \quad -\tilde{\gamma}u_2 \quad \tilde{\gamma}]^T \tag{27}$$

$$Q_{km}=[-U_k \quad \partial\xi_k/\partial x_1 \quad \partial\xi_k/\partial x_2 \quad 0]^T \tag{28}$$

$$Q_d=[1 \quad u_1 \quad u_2 \quad (e+p)/\rho]^T \tag{29}$$

where $\tilde{\gamma} = \gamma - 1$ and $\gamma$ is the specific heat. Except for $\Delta Q$, values are evaluated at the midpoint between the grid point $(i, j)$ and the next point, such as $(i+1, j)$, $(i-1, j)$, $(i, j+1)$, $(i, j-1)$.

## Results

A hypersonic viscous flow around a cylinder is first calculated with the LU-SGS-$\omega$ method to present the convergence of solution for the compressible Navier–Stokes equations. The flow conditions are specified as uniform Mach number 4.0 and Reynolds number $1.0 \times 10^5$. The flow is supposed to be laminar, without chemical reaction. The first-order accurate AUSM[5] algorithm is used for the space difference. The computational grid used in this case has $81 \times 71$ grid points. The local time step is used together with this method. Then, the CFL number at all grid points is fixed to 1 and 100. Figure 1 shows the convergence histories of solution ($L2$-norm). The cases of $\omega = 1.0$ and 1.35 at CFL $= 1.0$ and $\omega = 1.0$ and 1.2 at CFL $= 100$ in Eqs. (18) and (19) are calculated, and the convergence of solution is compared in Fig. 1. The convergence of solution is clearly accelerated by the increase in the parameter $\omega$. The convergence in the case of $\omega = 1.35$ at CFL $= 1$ is about 1.5 times faster than that of $\omega = 1.0$. The maximum value in this case, was $\omega = 1.35$. Also, the convergence in the case of $\omega = 1.2$ at CFL $= 100$ is about 1.2 times faster than that of $\omega = 1.0$. In this case, $\omega = 1.2$ was the maximum value. As a result, the convergence of solution in the case of $\omega = 1.2$ at CFL $= 100$ is highly accelerated in this flow problem. Note that the present method could enlarge the CFL number to much more than 100, but that the convergence rate and the maximum limit of the value $\omega$ was exactly the same with the case of $\omega = 1.2$ at CFL $= 100$. Figure 2 shows the calculated Mach number contours in the case of $\omega = 1.2$ at CFL $= 100$. A monotone bow shock is captured ahead of the cylinder. The calculated results were identical to each other in these cases.

As a second example, a hypersonic viscous flow through the 15-deg compression corner is calculated. This flow problem is well known as a benchmark test to check hypersonic flow codes. The uniform mach number is 14.1, the uniform temperature is 72.7 K and the Reynolds number is $1.04 \times 10^5$. The flow is supposed to be laminar, without chemical reaction. The AUSM[5] with the second-order MUSCL extrapolation is used for the space difference to check the performance of the LU-SGS-$\omega$ method for high-order calculations in space. An H-type computational grid is generated, and it has $201 \times 101$ grid points. The CFL number at all grid points is fixed to 1 and 100. The cases of $\omega = 1.0$ and 1.20 at CFL $= 1.0$, and $\omega = 1.0$ and 1.15 at CFL $= 100$ in Eqs. (18) and (19) are calculated. Figure 3 shows the convergence histories of solution ($L2$-norm). Although the value of $\omega$ must be set at a lower value than that of the preceding cylinder problem, the acceleration of convergence of solution can certainly be achieved. Figure 4 shows the calculated Mach number contours obtained by the case of $\omega = 1.15$ at CFL $= 100$. An oblique shock is generated from the front edge of the compression corner, and it interacts with another oblique shock generated by the flow compression on the ramp region of the compression corner. A boundary layer without separation grows from the front edge of the compression corner. The calculated pressure coefficient distributions on the compression corner can be compared with the

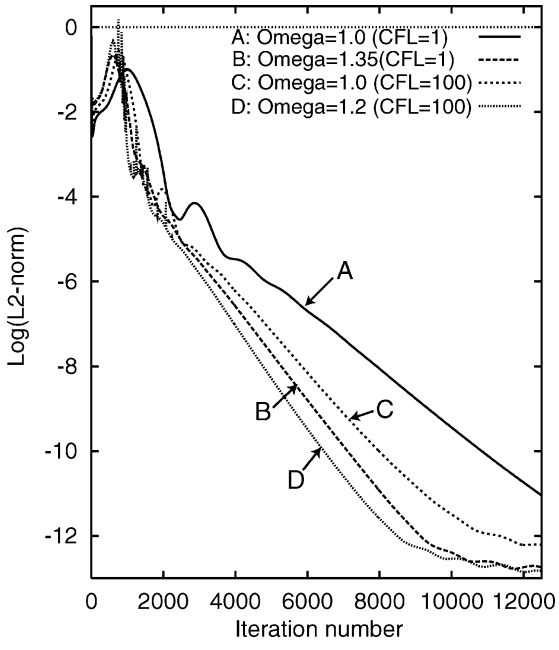


**Fig. 1 Convergence histories of solution for the hypersonic viscous flow around a cylinder.**

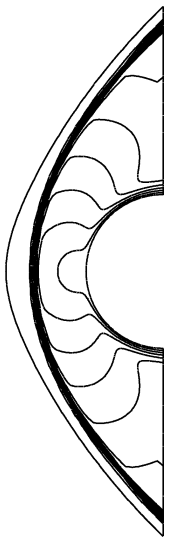

**Fig. 2 Mach number contours.**

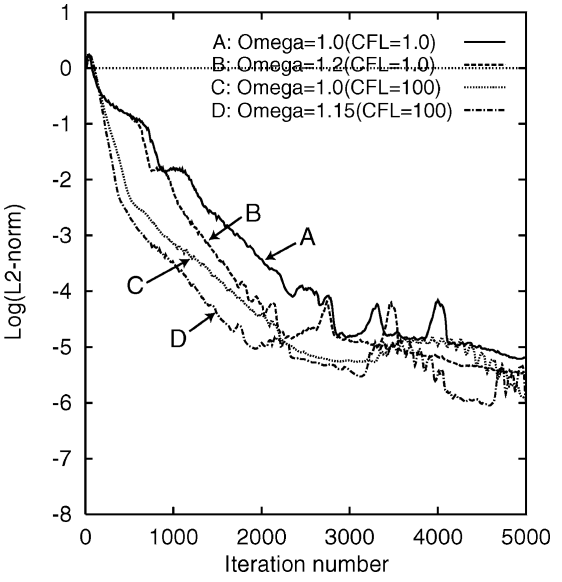


**Fig. 3 Convergence histories of solution for the hypersonic viscous flow through a compression corner.**

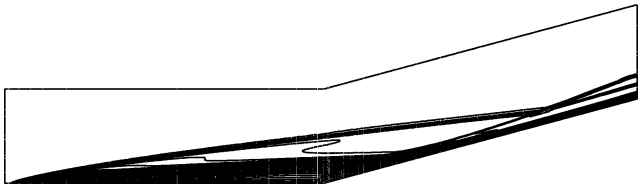

**Fig. 4 Mach number contours.**

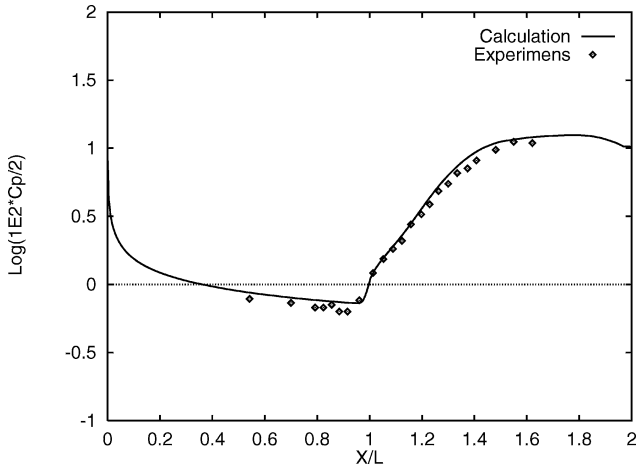


**Fig. 5 Pressure distributions on the compression corner compared with the experiments.**

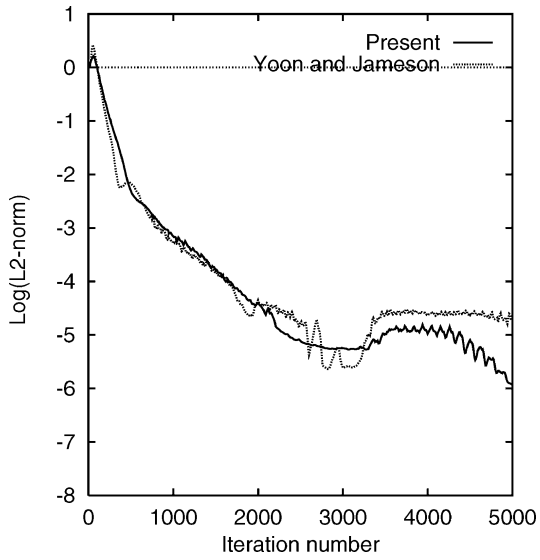


**Fig. 6 Comparison of convergence histories of solution between the present method and the original method.**

experiments[6] in Fig. 5. The calculated results are in good agreement with the experiments.

Note that the diagonal flux-splitting form in Eq. (25) is used for the calculation of Eqs. (15) and (16) instead of the original equation in Eq. (24). The flux-splitting form has been derived in our previous study[4] and also has been extended to more complicated flow problems.[3] The primary reason why we use the flux-splitting form for the LU-SGS algorithm is its extensity to a complicated system of equations. We have only to derive subvectors such as Eqs. (26–29) in each system. The algorithm, except for the subvectors in the present form, are fundamentally all the same in each system. Therefore, we need no information about elements of the Jacobian matrices for each system. In addition to this, the present overrelaxation does not work well in the code with the original method using Eq. (24). The correct reason has been yet to be resolved. Consequently, Eq. (25) was only used in this paper for the overrelaxation cases. Here, in Fig. 6, the convergence of solution obtained by the use of the original LU-SGS method can be compared with that by the present method in the case of $\omega = 1.0$ at CFL $= 100$. The convergence rates were almost identical. This indicates that the present flux-splitting form works at least as well as the original method when overrelaxation is not used. The difference of computational costs between the present form and the original form was within a few percent when the supercomputer SX-7/4CPU with parallel and vector optimizations was used for the calculations.

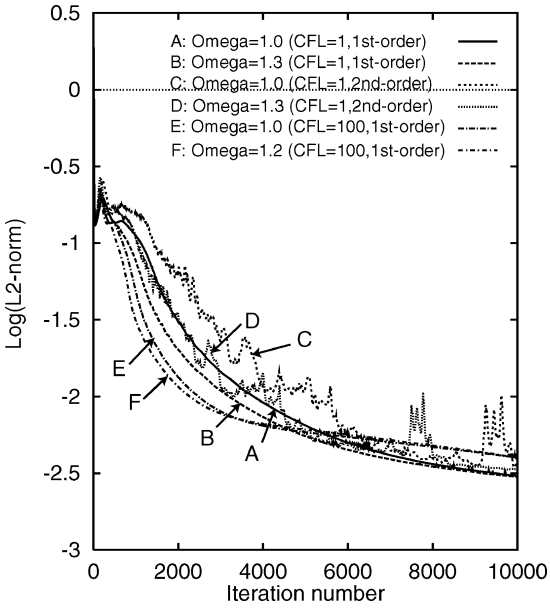


**Fig. 7 Convergence histories of solution for the hypersonic thermochemical nonequilibrium flow around a sphere.**

Finally, the present method is applied to a hypersonic thermochemical nonequilibrium flow around a sphere. The employed test case is well known as Lobb's test problem.[7] The continuity equations for chemical species $N_2$, $O_2$, NO, N, and O and a vibrational energy equation are also solved with the momentum equations and the total energy equation. The two-temperature chemical reaction model by Park[8] is used for thermochemical nonequilibrium effects. The details are reported in Ref. 9. The uniform Mach number is 15.3, the uniform velocity is 5280 m/s, the uniform pressure is 664 Pa, the uniform translational–rotational temperature is 293 K, and the Reynolds number is $1.46 \times 10^4$. The flow is supposed to be laminar. The first-order and second-order MUSCL extrapolations are applied to AUSM_DV[10] to determine the space difference. The computational grid used in this case has $65 \times 65$ grid points. The local time step is used together with these schemes. Then, the CFL number at all grid points is fixed to 1 and 100. The cases of $\omega = 1.0$ and 1.30 at CFL $= 1.0$, and $\omega = 1.0$ and 1.2 at CFL $= 100$ in Eqs. (18) and (19) are calculated with the first-order scheme in space. The cases of $\omega = 1.0$ and 1.30 at CFL $= 1.0$ in Eqs. (18) and (19) are calculated with the second-order scheme in space. The convergences of solution are compared in Fig. 7. The convergence of solution for the present thermochemical nonequilibrium flow is relatively slower than that of the preceding nonreacting flow. Although the convergence may not always be accelerated as much as the preceding nonreaction cases, the improvement of convergence can be found in Fig. 7 if the overrelaxation parameter $\omega$ increases. The CFL number can also be enlarged much more in this case. However, the convergence rate is completely the same with the rate at CFL $= 100$. In addition to this, the final $L2$-norm for the case of CFL $= 100$ was ultimately larger than that of CFL $= 1$ in this case. Figures 8a and 8b show the calculated Mach number contours determined with the first- and second-order-accurate schemes in space, respectively. The shock standoff distance in the second-order-accurate solution is slightly shorter than that of the first-order-accurate solution. Figure 9 shows the calculated translational–rotational temperatures and vibrational temperatures at the stagnation line. The second-order-accurate solution is compared with the first-order-accurate solution in Fig. 9. The maximum value of the translational–rotational temperature in the second-order-accurate solution is 1.1 times larger than that in first-order-accurate solution. Figure 10 shows the calculated mole fractions of $N_2$, $O_2$, NO, N, and O at the stagnation line. Finite rate dissociations for $N_2$ and $O_2$ can be found in Fig. 10 behind the bow shock. $O_2$ is almost dissociated close to the body surface.

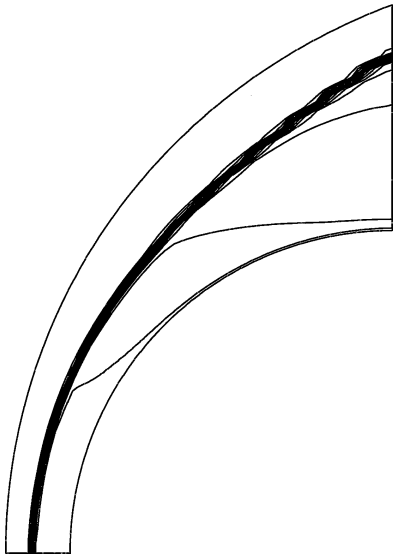

**Fig. 8a Mach number contours (first-order accuracy in space).**

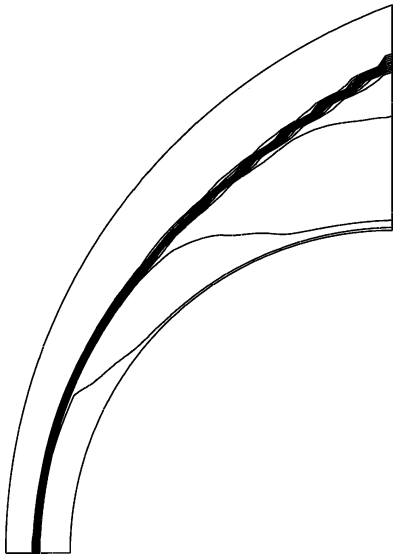

**Fig. 8b Mach number contours (second-order accuracy in space).**

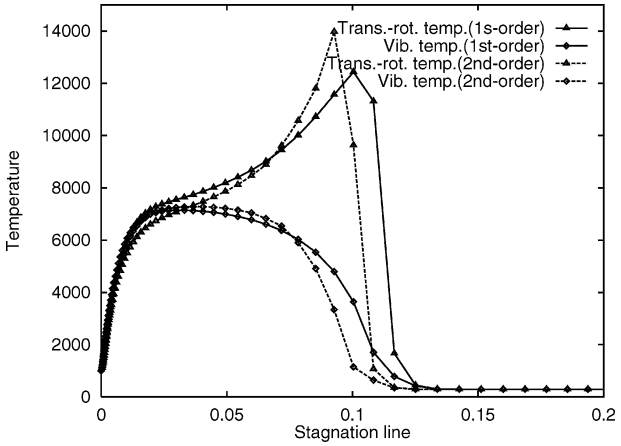


**Fig. 9 Translational–rotational and vibrational temperature distributions along the stagnation line.**

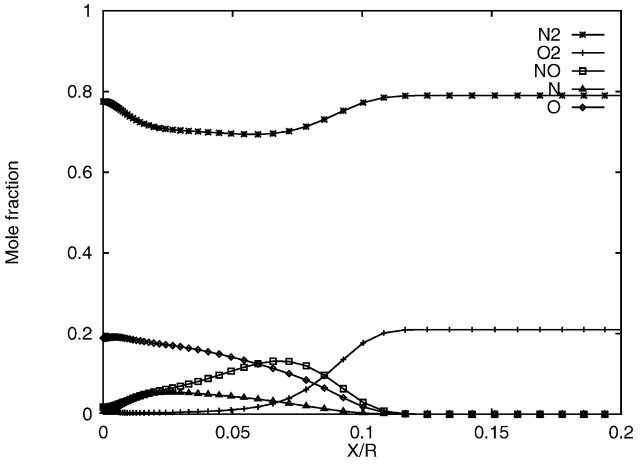


**Fig. 10 Mole fraction distributions along the stagnation line.**

## Conclusions

A very simple overrelaxation method applied to the LU-SGS method was proposed. Because the algorithm is quite simple, all of the existing computational codes based on the LU-SGS method can be easily modified by this algorithm. Three typical hypersonic flow problems were calculated without special optimization to the overrelaxation parameter $\omega$. The calculated results indicate that the convergence of solution can be accelerated by the use of a proper value of the overrelaxation parameter $\omega$. This method will also be extended to other, already developed complicated flow codes, such as multiphase flows and magnetoplasma flows, in the near future.

## References

[1]Yoon, S., and Jameson, A., "Lower–Upper Symmetric-Gauss–Seidel Method for the Euler and Navier–Stokes Equations," *AIAA Journal*, Vol. 26, 1988, pp. 1025–1026.

[2]Frankel, S. P., "Convergence Rates of Iterative Treatments of Partial Differential Equations," *Mathematical Tables and Other Aids to Computation*, Vol. 4, 1950, pp. 65–75.

[3]Yamamoto, S., "Time-Marching Method for Complicated Compressible Flow Equations with Source Term," *Proceedings of 1st International Conference of Computational Fluid Dynamics, Computational Fluid Dynamics 2000*, Springer, Berlin, 2000, pp. 133–138.

[4]Yamamoto, S., and Kano, S., "An Efficient CFD Approach for Simulating Unsteady Hypersonic Shock–Shock Interference Flows," *Computers and Fluids*, Vol. 27, 1998, pp. 571–580.

[5]Liou, M.-S., and Steffen, C. J., "A New Flux Splitting Scheme," *Journal of Computational Physics*, Vol. 107, 1993, pp. 23–39.

[6]Holden, M. S., and Moselle, J. R., "Theoretical and Experimental Studies of the Shock Wave–Boundary Layer Interaction on Compression Surfaces in Hypersonic Flow," U.S. Air Force Research Lab., Rept. ARL 70-0002, Wright–Patterson AFB, OH, Jan. 1970.

[7]Lobb, R. K., "Experimental Measurement of Shock Detachment Distance on Spheres Fired in Air at Hypervelocities," *The High Temperature Aspects of Hypersonic Flow*, Pergamon, New York, 1964, pp. 519–527.

[8]Park, C., "Two-Temperature Interpretation of Dissociation Rate Data for $N_2$ and $O_2$," AIAA Paper 88-0458, Jan. 1988.

[9]Yamamoto, S., Takasu, N., and Nagatomo, H., "Numerical Investigation of Shock/Vortex Interaction in Hypersonic Thermochemical Nonequilibrium Flow," *Journal of Spacecraft and Rockets*, Vol. 36, No. 2, 1999, pp. 240–246.

[10]Wada, Y., and Liou, M.-S., "A Flux Splitting Scheme with High Resolution and Robustness for Discontinuities," AIAA Paper 94-0083, Jan. 1994.

M. Sichel
*Associate Editor*