AN ENTROPY VARIABLE-BASED GRIDFREE SOLVER WITH LU-SGS ACCELERATOR

K. Anandhanarayanan*, M. Nagarathinam* and S.M. Deshpande**
***Scientist, CFD Division, DRDL, Kanchanbagh, Hyderabad-500058, India**
****Professor, Dept. of Aerospace Engg., IISc., Bangalore-560012, India**

Keywords: *Kinetic scheme, Gridfree, Implicit*

Abstract

Lower-Upper Symmetric Gauss Seidel (LU-SGS) implicit method has been implemented within the framework of gridfree Least Squares Kinetic Upwind Method (LSKUM). The LU-SGS method is modified to preserve the generality of the gridfree method for non-symmetric connectivity and yet retain matrix-free feature of the LU-SGS method. Moreover, in the present work kinetic boundary conditions are also implemented in the implicit framework. Implicit scheme for gridfree method is proved to be stable for all time steps and a speed-up 5 to 6 is obtained without any additional memory requirement. These aspects are demonstrated through specific examples.

1 Introduction

The Least Squares Kinetic Upwind Method (LSKUM) [1] is a gridfree method that requires only a cloud of points distributed in the computational domain and a set of neighbours around each point (connectivity). Therefore, this method makes solutions possible to geometrically complex configurations. The LSKUM is based on the Kinetic Flux Vector Splitting (KFVS) scheme [2], which exploits the connection between the Boltzmann equation of kinetic theory of gases, and the governing equations of fluid dynamics by using moment method strategy. Euler equations are obtained by taking ψ-moments of the Boltzmann equation with Maxwellian velocity distribution function. The upwinding is done at the molecular level and then taking ψ-moments lead to the KFVS scheme. In the LSKUM, the spatial derivatives of the Boltzmann equation are discretized using weighted least squares method. The upwinding is done by choosing appropriate stencils from the connectivity based on sign of the molecular velocity. Finally, taking ψ-moments lead to LSKUM. Higher order accuracy in space is achieved using the defect correction technique [3]. The q-LSKUM [4] is an improvement over LSKUM in the sense that the entropy variables, also called q-variables, are used in the defect correction step of LSKUM for obtaining second order accurate in space including boundary points. Both LSKUM and q-LSKUM have been applied to a number of 2-D and 3-D problems [5,6] and found to yield highly accurate results.

The solution to large-scale problems with explicit method requires large number of iterations. Convergence acceleration methods can be used to reduce the computation time. Among various convergence acceleration techniques, LU-SGS factorized implicit method is popular because it is a matrix-free method. The LU-SGS method proposed by Jameson and Yoon [7] on structured grid has been successfully generalized and extended to unstructured meshes to solve the Euler and Navier-Stokes equations by Sharov and Nakahashi [8]. The attractive feature of this approximate factorization method is that the evaluation and storage of the Jacobian matrix is eliminated by approximating the split flux Jacobians in the implicit operator. This approximation provides stability for all time steps and facilitates faster convergence. The resulting LU-SGS method has comparable CPU time per time step that of explicit method. LU-SGS method has been successfully implemented

and a speed-up 6 is obtained using KFVS based finite volume Euler and N.S. solvers [9, 10].

The above approximation of split-flux Jacobians does not fetch similar advantages for LSKUM, since it uses split stencils of arbitrary size for the spatial derivatives. Therefore, in the present work, a different approximation is used to retain all the advantages of original LU-SGS and yet retain the generality of the LSKUM.

This paper addresses the modification of LU-SGS method applied to LSKUM and its implementation in LSKUM based solvers. Numerical experiments are carried out to prove the efficacy of the extended LU-SGS method.

2 Implicit q-LSKUM

Consider 1-D unsteady compressible Euler equations

$$\frac{\partial U}{\partial t}+\frac{\partial}{\partial x}G(U^{n+1})=0 \qquad (1)$$

where U is vector of conserved variables and G is the inviscid flux vector. The flux $G(U^{n+1})$ is linearized in time and can be written in terms of flux Jacobian A and the difference between conserved variables at time levels n and $n+1$, $\delta U^{n+1}=U^{n+1}-U^{n}$, as

$$G(U^{n+1}) = G(U^{n})+\left(\frac{\partial G}{\partial U}\right)^{n}\delta U^{n}$$
$$= G(U^{n})+A^{n}\,\delta U^{n}$$

and substituting in Eq. (1), we get

$$\frac{\partial U}{\partial t}+\frac{\partial}{\partial x}(A\delta U)^{n}=-\frac{\partial}{\partial x}(G(U^{n})) \qquad (2)$$

Consider the residue, $R(U^n)$, the spatial derivatives in the RHS of Eq. (2). The above residue can be written in KFVS split-flux [2] form as

$$R(U^{n})=\frac{\partial}{\partial x}(G(U^{n}))$$
$$=\frac{\partial}{\partial x}(G^{+}(U^{n}))+\frac{\partial}{\partial x}(G^{-}(U^{n})) \qquad (3)$$

The residual vector $R(U^n)$ is discretized using q-LSKUM [3]. Consider a point P_o and its connectivity $C(Po)$. The connectivity is split into two sub stencils $C^{+}(P_o)$ and $C^{-}(P_o)$ (Fig.1), such that

$$\left.\begin{aligned} C^{+}(P_o)&=[P_i\in C(P_o): x(P_i)<x(P_o)]\\ C^{-}(P_o)&=[P_i\in C(P_o): x(P_i)>x(P_o)]\end{aligned}\right\} \qquad (4)$$

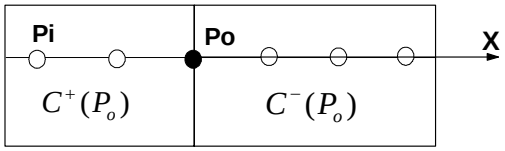

Fig. 1 Sub-stencil for point P_o

The split fluxes derivatives in Eq. (3) are evaluated using least squares method with above split-stencil such that upwind property is satisfied [1]. Then the residue vector at P_o can be written in discretized form as

$$R(U_o^{n})=\frac{\sum_{i\in C^{+}(P_o)}\Delta x_i\Delta G_i^{+}}{\sum_{i\in C^{+}(P_o)}(\Delta x_i)^2}+\frac{\sum_{i\in C^{-}(P_o)}\Delta x_i\Delta G_i^{-}}{\sum_{i\in C^{-}(P_o)}(\Delta x_i)^2} \qquad (5)$$

where $\Delta(.)_i = (.)_i - (.)_o$. In order to obtain a steady-state solution, the spatially discretized Euler equations must be integrated in time. Eq. (2) can be written in discretized form using Euler implicit time-integration and least squares method for spatial derivatives at Po as

$$\left(\frac{\delta U_o}{\delta t}\right)^{n}+\frac{\sum_{i\in C^{+}(P_o)}(A_i^{+}\delta U_i-A_o^{+}\delta U_o)^{n}\Delta x_i}{\sum_{i\in C^{+}(P_o)}(\Delta x_i)^2}$$
$$+\frac{\sum_{i\in C^{-}(P_o)}(A_i^{-}\delta U_i-A_o^{-}\delta U_o)^{n}\Delta x_i}{\sum_{i\in C^{-}(P_o)}(\Delta x_i)^2}=-R(U_o^{n}) \qquad (6)$$

where δt is the time increment. The LHS terms are grouped based on node number, i.e., corresponding to node P_o whose index o and its neighbor P_i, such that index $i < o$ and $i > o$,

$$\left(\frac{I}{\delta t}-\frac{\sum_{i\in C^+(P_o)} A_o^{+n}\Delta x_i}{\sum_{i\in C^+(P_o)}(\Delta x_i)^2}-\frac{\sum_{i\in C^-(P_o)} A_o^{-n}\Delta x_i}{\sum_{i\in C^-(P_o)}(\Delta x_i)^2}\right)\delta U_o{}^n + \left(\frac{\sum_{i\in C^+(P_o),i<o}(A_i^+\delta U_i)^n\Delta x_i}{\sum_{i\in C^+(P_o)}(\Delta x_i)^2}+\frac{\sum_{i\in C^-(P_o),i<o}(A_i^-\delta U_i)^n\Delta x_i}{\sum_{i\in C^-(P_o)}(\Delta x_i)^2}\right) + \left(\frac{\sum_{i\in C^+(P_o),i>o}(A_i^+\delta U_i)^n\Delta x_i}{\sum_{i\in C^+(P_o)}(\Delta x_i)^2}+\frac{\sum_{i\in C^-(P_o),i>o}(A_i^-\delta U_i)^n\Delta x_i}{\sum_{i\in C^-(P_o)}(\Delta x_i)^2}\right) = -R(U_o^n) \quad (7)$$

The above system of equations can be written in the matrix form in terms of strict lower $\boldsymbol{L}$, upper $\boldsymbol{U}$ and diagonal $\boldsymbol{D}$ block matrices

$$(\boldsymbol{L}+\boldsymbol{D}+\boldsymbol{U})\,\delta U^n = -R(U^n)$$

and the system of equations can be solved using LU-SGS in two steps as

Forward sweep:

$$\delta U^* = \boldsymbol{D}^{-1}(-R(U^n) - \boldsymbol{L}\,\delta U^*)$$

Backward sweep:

$$\delta U^n = \delta U^* - \boldsymbol{D}^{-1}(-\boldsymbol{U}\,\delta U^*)$$

The solution involves split flux Jacobian evaluation and inversion which is computationally expensive. Consider the diagonal term

$$\mathrm{D} = \left(\frac{I}{\delta t}-\frac{\sum_{i\in C^+(P_o)} A_o^{+n}\Delta x_i}{\sum_{i\in C^+(P_o)}(\Delta x_i)^2}-\frac{\sum_{i\in C^-(P_o)} A_o^{-n}\Delta x_i}{\sum_{i\in C^-(P_o)}(\Delta x_i)^2}\right)$$

In FVM and FDM methods, the split-flux Jacobians are approximated with

$$A^{\pm}=\frac{1}{2}\left(A\pm\rho(A)I\right)$$

where $\rho(A)$ is the spectral radius of flux Jacobian A and the approximation makes diagonal block matrix as a scalar matrix. In LSKUM, with the above approximation, the diagonal term becomes

$$\boldsymbol{D} = \left(\frac{I}{\delta t} - A\left\{\frac{\sum_{i\in C^+(P_o)}\Delta x_i}{\sum_{i\in C^+(P_o)}(\Delta x_i)^2}+\frac{\sum_{i\in C^-(P_o)}\Delta x_i}{\sum_{i\in C^-(P_o)}(\Delta x_i)^2}\right\} - \rho(A)\left\{\frac{\sum_{i\in C^+(P_o)}\Delta x_i}{\sum_{i\in C^+(P_o)}(\Delta x_i)^2}-\frac{\sum_{i\in C^-(P_o)}\Delta x_i}{\sum_{i\in C^-(P_o)}(\Delta x_i)^2}\right\}\right)$$

The above diagonal block matrix becomes scalar diagonal matrix only if the second term in RHS vanishes. This is possible only for the symmetric point distributions and cannot be ensured for an arbitrary point distribution. Therefore, in the present work the split-flux Jacobian is approximated as

$$A^{\pm}=\pm\rho(A^{\pm})I$$

and the diagonal block matrix becomes a scalar diagonal matrix as

$$\boldsymbol{D} = \left(\frac{1}{\delta t}-\rho(A^+)\left\{\frac{\sum_{i\in C^+(P_o)}\Delta x_i}{\sum_{i\in C^+(P_o)}(\Delta x_i)^2}\right\}+\rho(A^-)\left\{\frac{\sum_{i\in C^-(P_o)}\Delta x_i}{\sum_{i\in C^-(P_o)}(\Delta x_i)^2}\right\}\right)I$$

which can be inverted easily. The product of split-flux Jacobians with increment in conserved variables appearing in Eq. (7) are approximated as increment in split-fluxes as

$$A^{\pm}\delta U = G^{\pm}(U+\delta U)-G^{\pm}(U) \equiv \delta G^{\pm}$$

Then, the two-step sweep procedure can be written as

Forward sweep:

$$\delta U_o^* = D^{-1}\left[R(U_o^n)-\left\{\frac{\sum_{i\in C^+(Po),i<o}\Delta x_i\,\delta G_i^{+*}}{\sum_{i\in C^+(Po)}\Delta x_i^2}+\frac{\sum_{i\in C^-(Po),i<o}\Delta x_i\,\delta G_i^{-*}}{\sum_{i\in C^-(Po)}\Delta x_i^2}\right\}\right]$$

Backward sweep:

$$\delta U_o^n = \delta U_o^* - D^{-1}\left[\frac{\sum_{i\in C^+(Po),i>o}\Delta x_i\,\delta G_i^{+n}}{\sum_{i\in C^+(Po)}\Delta x_i^2}+\frac{\sum_{i\in C^-(Po),i>o}\Delta x_i\,\delta G_i^{-n}}{\sum_{i\in C^-(Po)}\Delta x_i^2}\right]$$

It can be observed that in the above two-step procedure, the split-flux Jacobian evaluation, storage and inversions are completely eliminated. Also, the present approach is applicable to non-symmetric point distribution.

3 Stability Analysis

The spectral radii approximation to split-flux Jacobian makes each term in Eq. (15) individually positive. Presence of time-step term ensures the system of equations is always diagonally dominant. Therefore, the solution to the system is unconditionally stable and allows a larger time-step. In the present work, the spectral radii is chosen as

$$\rho(A^{\pm}) = \frac{1}{2}\left|(u \pm a) \pm |u \pm a|\right|$$

Ghosh[1] has made a detailed stability analysis for the explicit method and derived the time step requirement as

$$\delta t = CFL \;\; Min\left|\frac{\Delta s_i}{|u| + 3\sqrt{RT}}\right|$$

where Δsi is the distance between points *Po* and its neighbour *Pi*, *R* is the gas constant, *T* is the absolute temperature and *u* is the fluid velocity.

4 Code Development

The above procedure has been implemented in 2-D q-LSKUM based Euler and Navier-Stokes codes and 3-D q-LSKUM based Euler code. In the implicit method, first explicit method is used to obtain the residue and then two-sweep procedure is adopted to update the state variables. Kinetic Characteristic Boundary Condition (KCBC) and Kinetic Outer Boundary Conditions KOBC)[11] are implemented to satisfy the slip wall boundary and farfield boundary conditions respectively. The code has been validated for various inviscid test cases to verify the modified implicit scheme. In the present work, local time stepping is used for time marching. CFL number is chosen smaller than 1.0 for the explicit method. In the current implicit calculation, a large time step is chosen and the CFL number for the implicit scheme is as high as 10,000.

4 Results and Discussions

4.1 Flow past 2-D bump

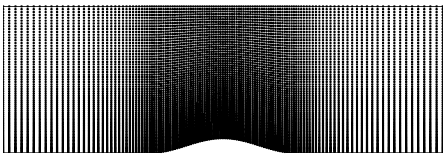

Fig.2 Point distribution for bump problem

The subsonic flow past 2-D bump problem has been studied with two sets of cloud of points with 2275 points and 8901 points (Fig. 2). Both explicit and implicit methods are used to obtain the solution with CFL numbers 0.5 and 10,000 respectively. The convergence histories are shown in Fig. 3. Implicit method converges rapidly even for the finer cloud of points when compared with explicit method. The pressure contours obtained using both explicit and implicit methods are presented in Fig. 4. It can be observed that the steady-state solution does not depend on the time-step chosen. Also contours are smooth and symmetric about vertical line passing through mid-chord.

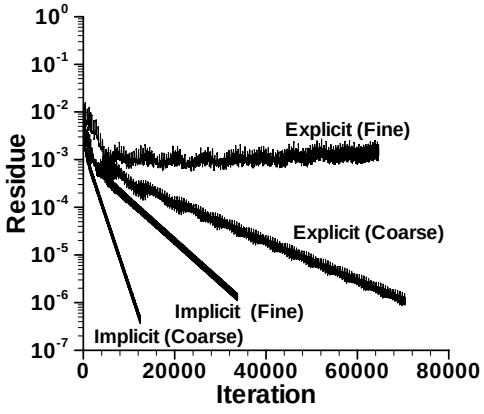

Fig.3 Residue history for flow past bump

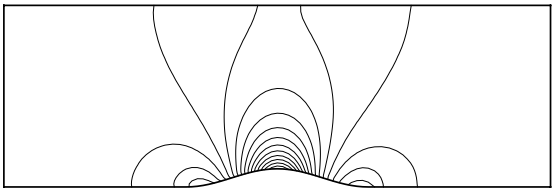

Fig.4 Pressure contours for flow past bump

4.2 Subsonic viscous flow past NACA0012 airfoil

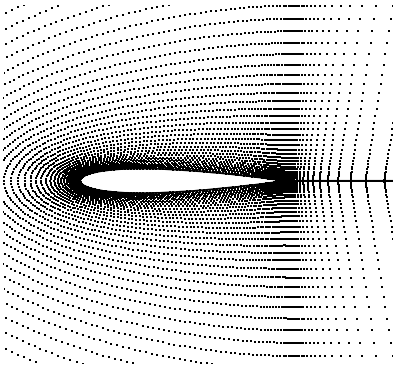

Fig. 5 Point distribution for viscous flow past NACA0012 airfoil

The 2-D q-LSKUM based Navier-Stokes code has been applied to solve laminar flow past NACA 0012 airfoil. A simple cloud with 257 x 97 points including 165 points on the airfoil is used (Fig. 5). The freestream flow conditions are M_∞ = 0.5, α = 3^o and Re = 5000. Comparison of residue for both implicit and explicit methods is presented in Fig. 6. A speed-up 5 is achieved using LU-SGS method. Mach contours are shown in Fig. 7 which clearly shows the separation on the upper surface of airfoil. The flow separates at 55% of the chord. The length of flow separation region is slightly under predicted.

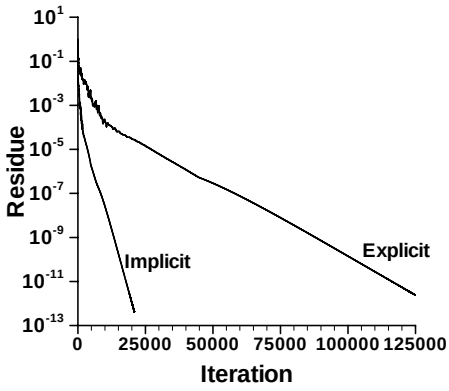

Fig. 6 Residue history for viscous flow past NACA0012 airfoil

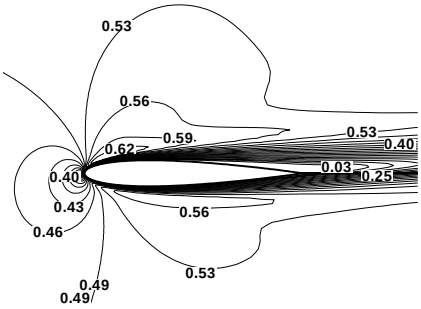

Fig. 7 Mach contours for viscous flow past NACA0012 airfoil

4.3 Supersonic flow past hemisphere

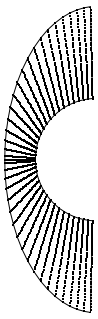

Fig. 8 Point distribution for flow past hemisphere

Next a supersonic flow past hemisphere is considered. The freestream Mach number is 2.0. The point distribution in the meridian plane is shown in Fig. 8 and there are 9261 points in the domain. The residue plot for both explicit and implicit methods is shown in Fig. 9. The residue plot shows a speed-up 6 is obtained with the extended LU-SGS method. Even in supersonic flow on relatively coarse grid, very good speed-up is obtained using implicit method without any extra storage. The pressure contours are plotted in Fig. 10. The detached shock is captured very well and contours are smooth. Stagnation-to-freestream pressure ratio of 5.86 is obtained as compared to theoretical value 5.64.

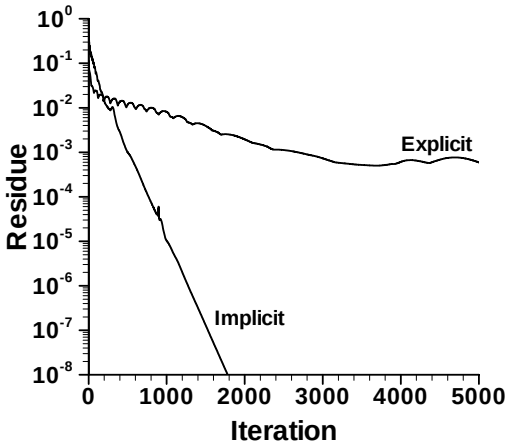

Fig. 9 Residue history for flow past hemisphere

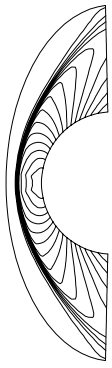

Fig. 10 Pressure contours for flow past hemisphere

4.4 Supersonic Flow Past Scout Vehicle

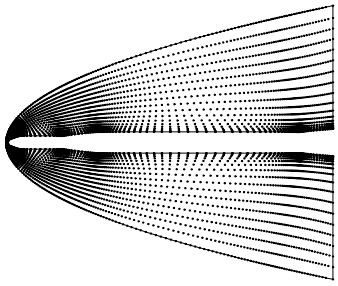

Fig. 11 Point distribution for flow past Scout vehicle

The 3-D q-LSKUM based Euler solver has been applied to solve the flow past a three-stage Scout vehicle for which experimental results are available [12]. The points in the computational domain are obtained using an elliptic grid generator. The point distribution in the meridian plane is shown in Fig. 11. The computed wall pressure distribution for $M_\infty = 4.86$ and $\alpha = 8^o$ is compared with the experimental results [12] and is presented in Fig. 12. Good agreement is apparent. Comparison of residue history for the explicit and implicit schemes is given in Fig. 13, which indicates the faster convergence capability of LU-SGS method. Pressure contours near nose are shown in Fig. 14. Detached shock and followed by expansions are captured.

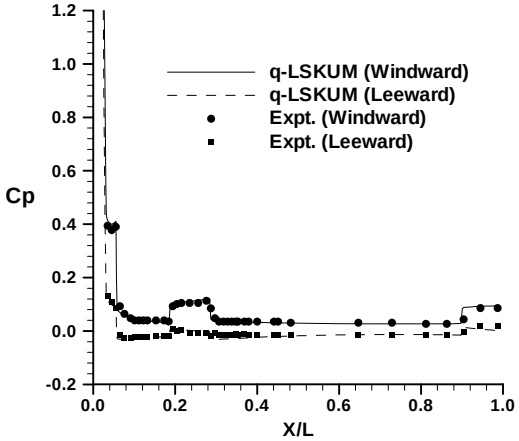

Fig. 12 Cp distribution for flow past Scout vehicle

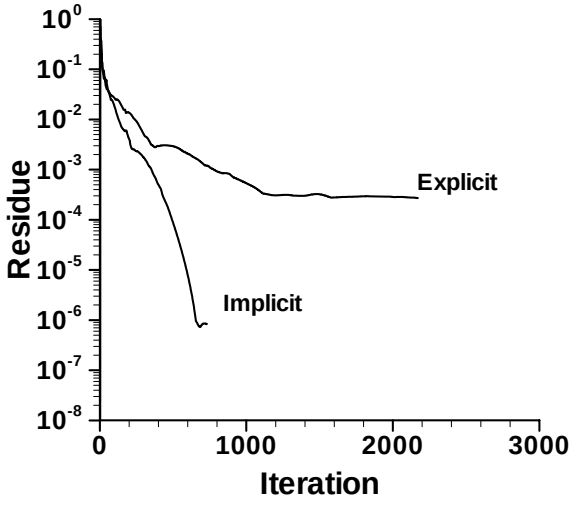

Fig. 13 Residue history for flow past Scout vehicle

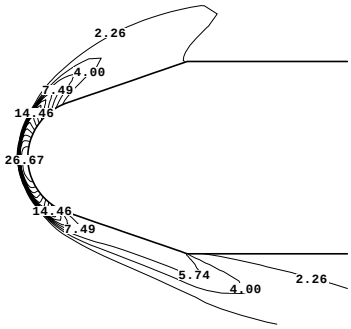

Fig. 14 Pressure contours near nose region of Scout vehicle

5. Conclusions

LU-SGS implicit method has been extended to LSKUM-based gridfree solvers. A different approximation is used for the split-flux Jacobian to retain all the advantages of original LU-SGS method like matrix-free, no-Jacobian evaluation, unconditionally stable with comparable CPU time per iteration of explicit method. In addition, the present method is applicable to gridfree methods employing arbitrary point distribution. Two and three-dimensional numerical simulations with explicit and implicit methods have been performed. The numerical simulations show a speed-up of 5 to 6 with implicit method over the explicit method.

References

[1] Ghosh AK. Robust least squares kinetic upwind method for inviscid compressible flows. *PhD thesis*, Department of Aerospace Engineering, Indian Institute of Science, Bangalore, 1996.

[2] Mandal JC and Deshpande SM. Kinetic flux vector splitting for Euler equations. *Computers and Fluids*, Vol. 23, No. 2, pp. 447-478, 1994.

[3] Deshpande SM. Meshless method, accuracy, symmetry breaking, upwinding and LSKUM. *FM report no. 2003 FM 1*, Dept. of Aerospace Engg., Indian Institute of Science, Bangalore, Jan., 2003.

[4] Dauhoo MZ, Ghosh AK, Ramesh V and Deshpande SM. q-LSKUM - A new higher order kinetic method for Euler equations using entropy variables. *Proceedings ISCFD 99*, Bremen, Germany,1999.

[5] Ramesh V. Least squares grid-free kinetic upwind method. *PhD thesis*, Dept. of Aerospace Engg., Indian Institute of Science, Bangalore, 2001.

[6] Deshpande SM. Anandhanarayanan K, Praveen C and Ramesh V. Theory and applications of 3-d LSKUM based on entropy variables. *Int Jnl of Numerical Methods in Fluids*,2002.

[7] Jameson A and Yoon S. Lower-upper implicit schemes with multiple grids for the Euler equations. *AIAA Journal*, Vol. 25, No. 7,1987.

[8] Sharov D and Nakahashi K. Low speed preconditioning and LU-SGS scheme for 3-d viscous flow computations on unstructured grids. *AIAA Paper 98-0614*, 1998.

[9] Balasubramanian R, Anandhanarayanan K and Nagarathinam M. LU-SGS convergence accelerator for KFVS based Euler solver. *Proceedings 4th Ae.S.I. CFD symposium*, Bangalore, India, 2001.

[10] Anandhanarayanan K, Nagarathinam M and Deshpande SM. A KFVS-based N.S. solver. *Proceedings 3rd Ae.S.I. CFD symposium*, Bangalore, India, 2000.

[11] Ramesh V, Mathur JS and Deshpande SM. Kinetic treatment of the farfield boundary condition. *FM report No. 97 FM2*, CFD center, Dept. of Aerospace Engg., IISc., April 1997.

[12] Jernell LS. Aerodynamic loading characteristics of a 1/10-scale model of the three-stage vehicle at Mach numbers from 1.57 to 4.65. *NASA TND-1930*, Langley research center, 1963.