Implicit Lambda Methods for Three-Dimensional Compressible Flow

M. Napolitano* and A. Dadone†
Universitá di Bari, Bari, Italy

This paper derives the three-dimensional lambda-formulation equations for a general orthogonal curvilinear coordinate system and provides various block-explicit and block-implicit methods for solving them numerically. Three model problems, characterized by subsonic, supersonic, and transonic flow conditions, are used to assess the reliability and compare the efficiency of the proposed methods.

Introduction

AMONG the many theoretical models employed in the numerical simulation of compressible inviscid flows, the so-called lambda-formulation has received considerable interest.[1-8] In this formulation, the time-dependent Euler equations are recast into compatibility conditions of bicharacteristic variables along corresponding bicharacteristic lines and discretized using upwind differences, in order to correctly account for the direction of wave propagation phenomena. Such an approach has many good properties: it provides very accurate numerical results, even with rather coarse meshes[2,3,6]; it requires only the physical boundary conditions, so that there is no need for any additional numerical boundary treatments, which are frequently the cause of numerical instability[9]; it handles in a most automatic and physically sound way mixed supersonic-subsonic flowfields; and finally, it has a well documented, although controversial, capability of capturing shocks without any additional artificial dissipation.[2,6]

Because of these properties, in spite of the fact that the "captured shocks" are only isentropic approximations to correct weak solutions of the Euler equations and do not correctly move within the flowfield unless properly fitted,[4] the lambda-formulation is considered to be a very useful and reliable tool for predicting compressible flowfields and therefore very worthy of further studies and improvement. In fact, in the last two years, for the cases of quasi-one-dimensional and two-dimensional flows, the development of various kinds of implicit integration schemes[5-8] has removed the only major limitation of previous lambda methods, namely, the Courant-Friedrichs-Lewy (CFL) stability restriction associated with their explicit integration procedures.

It now appears very timely and worthwhile to develop efficient numerical methods based on the lambda-formulation for three-dimensional flows. In the present paper, the "most appropriate" three-dimensional lambda-formulation equations are first derived for the case of a general orthogonal curvilinear coordinate system. The governing equations are then discretized and linearized in time using a delta approach,[11] and various block-explicit as well as block-implicit numerical techniques are proposed to solve the resulting discrete equations approximately at every time step. Finally, all of the proposed methods are applied to solve three model problems, characterized by subsonic, supersonic, and transonic flow conditions, respectively, in order to assess their reliability and efficiency.

Received April 7, 1984; revision received Aug. 20, 1984.

*Professor, Istituto di Macchine. Member AIAA.
†Professor, Istituto di Macchine.

Three-Dimensional Lambda-Formulation Equations

The nondimensional continuity and momentum (Euler) equations for the homentropic flow of a perfect gas are given in vector form as[3,5]

$$\delta(a_t + \boldsymbol{v}\cdot\nabla a) + a\nabla\cdot\boldsymbol{v} = 0 \tag{1}$$

$$\boldsymbol{v}_t + (\boldsymbol{v}\cdot\nabla)\boldsymbol{v} + \delta a\nabla a = 0 \tag{2}$$

where a is the speed of sound, $\boldsymbol{v}$ the velocity vector, ∇ the gradient operator, subscript t indicates partial derivatives with respect to time, and $\delta = 2/(\gamma-1)$, where γ is the specific heat ratio.

In a general orthogonal curvilinear coordinate system we have

$$\boldsymbol{v} = v_1\boldsymbol{e}_1 + v_2\boldsymbol{e}_2 + v_3\boldsymbol{e}_3 \tag{3}$$

$$\nabla = \boldsymbol{e}_1\frac{\partial}{\partial s_1} + \boldsymbol{e}_2\frac{\partial}{\partial s_2} + \boldsymbol{e}_3\frac{\partial}{\partial s_3} = \frac{\boldsymbol{e}_1}{h_1}\frac{\partial}{\partial q_1} + \frac{\boldsymbol{e}_2}{h_2}\frac{\partial}{\partial q_2} + \frac{\boldsymbol{e}_3}{h_3}\frac{\partial}{\partial q_3} \tag{4}$$

$$\nabla\cdot\boldsymbol{v} = \frac{1}{h_1h_2h_3}\left[\frac{\partial}{\partial q_1}(h_2h_3v_1) + \frac{\partial}{\partial q_2}(h_1h_3v_2) + \frac{\partial}{\partial q_3}(h_1h_2v_3)\right] \tag{5}$$

where $\boldsymbol{e}_i$ $(i=1,2,3)$ are the unit vectors of the (right-handed) orthogonal curvilinear coordinate system, q_i and h_i the corresponding coordinates and scale factors, $ds_i = h_i\,dq_i$ the elementary arc lengths along the coordinate lines,[10] and v_i the components of $\boldsymbol{v}$. Equations (1) and (2) can be written in the general orthogonal curvilinear coordinate system by means of Eqs. (3-5) and some lengthy but straightforward algebra. The only difficulty is the evaluation of the derivatives of the unit vectors $\boldsymbol{e}_i$ with respect to the coordinates q_i. These expressions are therefore given as

$$\frac{\partial \boldsymbol{e}_i}{\partial q_j} = \frac{\boldsymbol{e}_j}{h_i}\frac{\partial h_j}{\partial q_i} \qquad (i \neq j) \tag{6}$$

$$\frac{\partial \boldsymbol{e}_i}{\partial q_i} = -\frac{\boldsymbol{e}_j}{h_j}\frac{\partial h_i}{\partial q_j} - \frac{\boldsymbol{e}_k}{h_k}\frac{\partial h_i}{\partial q_k} \qquad (i \neq j \neq k) \tag{7}$$

The six lambda compatibility equations can now be obtained by summing and subtracting from the continuity equation (1) each component of the momentum equation (2), to give

$$C_t+(v_1+a)\frac{\partial C}{\partial s_1}+v_2\frac{\partial C}{\partial s_2}+v_3\frac{\partial C}{\partial s_3}=-\zeta-\alpha_1+\beta_1-\gamma_1 \tag{8a}$$

$$D_t+(v_1-a)\frac{\partial D}{\partial s_1}+v_2\frac{\partial D}{\partial s_2}+v_3\frac{\partial D}{\partial s_3}=\zeta-\alpha_1+\beta_1+\gamma_1 \tag{8b}$$

$$E_t+v_1\frac{\partial E}{\partial s_1}+(v_2+a)\frac{\partial E}{\partial s_2}+v_3\frac{\partial E}{\partial s_3}=-\zeta-\alpha_2+\beta_2-\gamma_2 \tag{8c}$$

$$F_t+v_1\frac{\partial F}{\partial s_1}+(v_2-a)\frac{\partial F}{\partial s_2}+v_3\frac{\partial F}{\partial s_3}=\zeta-\alpha_2+\beta_2+\gamma_2 \tag{8d}$$

$$G_t+v_1\frac{\partial G}{\partial s_1}+v_2\frac{\partial G}{\partial s_2}+(v_3+a)\frac{\partial G}{\partial s_3}=-\zeta-\alpha_3+\beta_3-\gamma_3 \tag{8e}$$

$$H_t+v_1\frac{\partial H}{\partial s_1}+v_2\frac{\partial H}{\partial s_2}+(v_3-a)\frac{\partial H}{\partial s_3}=\zeta-\alpha_3+\beta_3+\gamma_3 \tag{8f}$$

where C, D, E, F, G, and H are the six bicharacteristic variables given as

$$C=v_1+\delta a \tag{9a}$$

$$D=v_1-\delta a \tag{9b}$$

$$E=v_2+\delta a \tag{9c}$$

$$F=v_2-\delta a \tag{9d}$$

$$G=v_3+\delta a \tag{9e}$$

$$H=v_3-\delta a \tag{9f}$$

$$\zeta=\frac{a}{h_1h_2h_3}\left[v_1\frac{\partial}{\partial q_1}(h_2h_3)+v_2\frac{\partial}{\partial q_2}(h_1h_3)+v_3\frac{\partial}{\partial q_3}(h_1h_2)\right] \tag{10}$$

$$\alpha_1=\frac{v_1}{h_1}\left(\frac{v_2}{h_2}\frac{\partial h_1}{\partial q_2}+\frac{v_3}{h_3}\frac{\partial h_1}{\partial q_3}\right) \tag{11a}$$

$$\beta_1=\frac{v_2^2}{h_1h_2}\frac{\partial h_2}{\partial q_1}+\frac{v_3^2}{h_1h_3}\frac{\partial h_3}{\partial q_1} \tag{11b}$$

$$\gamma_1=a\left(\frac{1}{h_2}\frac{\partial v_2}{\partial q_2}+\frac{1}{h_3}\frac{\partial v_3}{\partial q_3}\right) \tag{11c}$$

and $\alpha_2,\beta_2,\ldots,\gamma_3$ have very similar expressions which can be obtained by appropriate subscript rotation and are thus omitted for the sake of conciseness.

Equations (8) are the compatibility conditions of the bicharacteristic variables along their bicharacteristic lines (in the four-dimensional q_1,q_2,q_3,t space), the left-hand sides of Eqs. (8) clearly being total derivatives along such lines. Therefore, they could be integrated by means of any numerical method using upwind differences according to the direction of wave propagation along the bicharacteristic lines, thus providing a three-dimensional lambda scheme. However, as in the two-dimensional case,[6] there are two major difficulties associated with solving Eqs. (8) numerically. First, the six bicharacteristic variables are not independent, insofar as their very definitions, Eqs. (9), imply that:

$$v_1=(C+D)/2 \tag{12a}$$

$$v_2=(E+F)/2 \tag{12b}$$

$$v_3=(G+H)/2 \tag{12c}$$

$$2\delta a=C-D=E-F=G-H \tag{12d,e,f}$$

so that

$$F=-C+D+E \tag{13}$$

$$H=-C+D+G \tag{14}$$

Therefore, any numerical solution obtained by integrating Eqs. (8) directly would lead to a "nonuniqueness" in the value of the speed of sound a. Furthermore, the right-hand sides of Eqs. (8), namely the γ_i coefficients, contain spatial derivatives of the velocity components, which are not associated with the convection of physical disturbances and are therefore likely to reduce the accuracy of the spatial discretization, if not the stability of the integration process. For these reasons, as in Refs. 5 and 6 for the two-dimensional case, the following equivalent system is obtained by taking a complete set of appropriate linear combinations of Eqs. (8).

$$C_t+D_t+(v_1+a)\frac{\partial C}{\partial s_1}+(v_1-a)\frac{\partial D}{\partial s_1}+v_2\frac{\partial}{\partial s_2}(C+D)+v_3\frac{\partial}{\partial s_3}(C+D)=-2\alpha_1+2\beta_1 \tag{15}$$

$$E_t+F_t+v_1\frac{\partial}{\partial s_1}(E+F)+(v_2+a)\frac{\partial E}{\partial s_2}+(v_2-a)\frac{\partial F}{\partial s_2}+v_3\frac{\partial}{\partial s_3}(E+F)=-2\alpha_2+2\beta_2 \tag{16}$$

$$G_t+H_t+v_1\frac{\partial}{\partial s_1}(G+H)+v_2\frac{\partial}{\partial s_2}(G+H)+(v_3+a)\frac{\partial G}{\partial s_3}+(v_3-a)\frac{\partial H}{\partial s_3}=-2\alpha_3+2\beta_3 \tag{17}$$

$$\frac{1}{3}\{C_t-D_t+E_t-F_t+G_t-H_t\}+(v_1+a)\frac{\partial C}{\partial s_1}-(v_1-a)\frac{\partial D}{\partial s_1}+(v_2+a)\frac{\partial E}{\partial s_2}-(v_2-a)\frac{\partial F}{\partial s_2}+(v_3+a)\frac{\partial G}{\partial s_3}-(v_3-a)\frac{\partial H}{\partial s_3}=-2\zeta \tag{18}$$

$$C_t-D_t-E_t+F_t=0 \tag{19}$$

$$C_t-D_t-G_t+H_t=0 \tag{20}$$

It is noteworthy that Eqs. (15-17) are simply the three components of the momentum equation (2), expressed as the sums of two compatibility conditions of two bicharacteristic variables along their bicharacteristic lines, whereas Eqs. (18-20) all coincide with the continuity equation (1). Also, after a straightforward integration with respect to time, Eqs. (19) and (20) reproduce identically Eqs. (13) and (14), so that they effectively reduce the number of dependent variables from six to four and any numerical integration of Eqs. (13-18) will guarantee a unique solution for the physical variable a. Finally, the right-hand sides of Eqs. (15-18) are seen to contain only source-like terms which do not involve spatial derivatives of the dependent variables and, therefore, are not likely to deteriorate the accuracy of any numerical method using upwind differences for the "total" derivatives of the bicharacteristics variables.

For these reasons Eqs. (13-18) are considered the "most appropriate" three-dimensional lambda-formulation equations for a general orthogonal coordinate system and will be the basis for all of the numerical methods proposed in this study.

Numerical Methods

The governing equations (13-18) are discretized and linearized in time using the delta form[11] to give

$$\frac{\Delta C}{\Delta t}+\frac{\Delta D}{\Delta t}+(u+a)^n\Delta C_x+(u-a)^n\Delta D_x + v^n(\Delta C_y+\Delta D_y)+w^n(\Delta C_z+\Delta D_z)=-(u+a)^n C_x^n -(u-a)^n D_x^n - v^n(C_y+D_y)^n - w^n(C_z+D_z)^n \tag{21}$$

$$\frac{\Delta E}{\Delta t}+\frac{\Delta F}{\Delta t}+u^n(\Delta E_x+\Delta F_x)+(v+a)^n\Delta E_y +(v-a)^n\Delta F_y+w^n(\Delta E_z+\Delta F_z)=-u^n(E_x+F_x)^n -(v+a)^n E_y^n-(v-a)^n F_y^n - w^n(E_z+F_z)^n \tag{22}$$

$$\frac{\Delta G}{\Delta t}+\frac{\Delta H}{\Delta t}+u^n(\Delta G_x+\Delta H_x)+v^n(\Delta G_y+\Delta H_y) +(w+a)^n\Delta G_z+(w-a)^n\Delta H_z=-u^n(G_x+H_x)^n -v^n(G_y+H_y)^n-(w+a)^n G_z^n-(w-a)^n H_z^n \tag{23}$$

$$\frac{1}{3}\left\{\frac{\Delta C}{\Delta t}-\frac{\Delta D}{\Delta t}+\frac{\Delta E}{\Delta t}-\frac{\Delta F}{\Delta t}+\frac{\Delta G}{\Delta t}-\frac{\Delta H}{\Delta t}\right\}+(u+a)^n\Delta C_x -(u-a)^n\Delta D_x+(v+a)^n\Delta E_y-(v-a)^n\Delta F_y +(w+a)^n\Delta G_z-(w-a)^n\Delta H_z=-(u+a)^n C_x^n +(u-a)^n D_x^n-(v+a)^n E_y^n+(v-a)^n F_y^n -(w+a)^n G_z^n+(w-a)^n H_z^n \tag{24}$$

$$\Delta F=-\Delta C+\Delta D+\Delta E \tag{25}$$

$$\Delta H=-\Delta C+\Delta F+\Delta G \tag{26}$$

where Cartesian coordinates (x,y,z) have been used for simplicity, u, v, and w are the Cartesian components of $\boldsymbol{v}$, subscripts indicate partial derivatives, Δt is the time step, $\Delta C = C^{n+1} - C^n$ (the superscripts $n+1$ and n indicating the new and old time levels $t^{n+1} = t^n + \Delta t$ and t^n), etc. Equations (21-26) constitute a first-order-accurate implicit time discretization of the corresponding differential problem. Equations (21-24) are then discretized in space, using two-point first-order-accurate upwind differences which properly account for the direction of wave propagation, and the ΔF and ΔH unknowns are eliminated in favor of ΔC, ΔD, ΔE, and ΔG by means of Eqs. (25) and (26) to produce, together with appropriate boundary conditions, a large 4×4 block-sparse linear system of the type

$$\boldsymbol{A}\boldsymbol{f}=\boldsymbol{b} \tag{27}$$

For the case of a cubic integration domain having N gridpoints in every spatial direction, $\boldsymbol{A}$ is a square matrix of order N^3 having only seven nonzero diagonals of 4×4-matrix-elements, $\boldsymbol{f}$ is the unknown vector having N^3 four-element-vector-components, and $\boldsymbol{b}$ is the known coefficient vector. It is noteworthy that in previous works[5,6] a second-order-accurate time linearization was employed. However, due to the use of a backward Euler time discretization, Eq. (27) is only first-order-accurate in time anyway. Moreover, the present linearization, coupled with two-point upwind difference approximations for the left-hand sides of Eqs. (21-24), leads to a diagonally dominant matrix $\boldsymbol{A}$ and has been verified to increase the stability of all the implicit methods to be proposed later in this study. It is also noteworthy that second-order-accurate, three-point upwind differences can be used to approximate the right-hand sides of Eqs. (21-24) (without altering the matrix $\boldsymbol{A}$) so that, if the flow reaches a steady state, the final solution is second-order-accurate.[5,6]

The main reason to employ an implicit method is to remove the CFL stability restriction, thus improving the efficiency of the calculations. However, a direct solution of Eq. (27), even if feasible, is certainly impractical. Therefore, matrix $\boldsymbol{A}$ will be replaced by a matrix $\boldsymbol{B}$ which is easily invertible and is a first-order-accurate approximation (in time) of $\boldsymbol{A}$.

A Block-Explicit (BE) Method

The simplest first-order-accurate approximation to $\boldsymbol{A}$ can be obtained by dropping all but the time-derivative terms in the left-hand sides of Eqs. (21-24). The resulting matrix $\boldsymbol{B}$ is diagonal, and a simple 4×4 linear system needs to be solved at every gridpoint to provide the local ΔC, ΔD, ΔE, and ΔG values. Furthermore, Eqs. (21-24) can be rearranged to give

$$\frac{2\Delta C}{\Delta t} = \mathrm{RHS}(21)+\mathrm{RHS}(24) \tag{28}$$

$$\frac{\Delta C}{\Delta t}+\frac{\Delta D}{\Delta t} = \mathrm{RHS}(21) \tag{29}$$

$$-\frac{\Delta C}{\Delta t}+\frac{\Delta D}{\Delta t}+\frac{2\Delta E}{\Delta t} = \mathrm{RHS}(22) \tag{30}$$

$$-\frac{\Delta C}{\Delta t}+\frac{\Delta D}{\Delta t}+\frac{2\Delta G}{\Delta t} = \mathrm{RHS}(23) \tag{31}$$

[where RHS(21) is a shorthand notation for the right-hand side of Eq. (21), etc.] so that every element of $\boldsymbol{B}$ is a 4×4 lower triangular matrix which can be inverted directly. The present BE method has been developed mainly for assessing the efficiency of various implicit methods; however, due to its extreme coding simplicity, it could very well be a useful tool by itself, especially if implemented on a vector computer.

A Block-Alternating-Direction-Implicit (BADI) Method

An alternating-direction-implicit (ADI) technique has been developed which is a direct extension of the three-dimensional problems of the method of Refs. 5 and 6. A three-sweep ADI process is used to solve Eq. (27) approximately. At the first sweep the t and x derivatives in the left-hand sides of Eqs. (21-24) are evaluated implicitly, whereas the y and z derivatives are evaluated explicitly. At the second and third sweeps the t and y and t and z derivatives are evaluated implicitly so that $\boldsymbol{A}$ is approximated by the product of three 4×4 block-tridiagonal matrices. In practice, at every sweep of the BADI method a 4×4 block-tridiagonal system of order N has to be solved along each line of the computational grid, so that $3N^2$ such systems need to be solved at every time step [i.e., to solve Eq. (27) approximately]. With respect to two-dimensional flow problems, the present ADI method is less competitive as compared to a standard explicit method for two reasons: the block size of the tridiagonal systems increases from three to four, and, more importantly, the number of tridiagonal systems to be solved at every time step grows from $2N$ to $3N^2$. Actually, for the simple problems later considered in this study the computer time per step for an 11^3 mesh was found to be about 30 times greater than that required by the BE method. More efficient implicit methods need therefore to be devised for the three-dimensional lambda-formulation equations.

A Block-Line-Gauss-Seidel (BLGS) Method

Classical relaxation methods have been recently employed with considerable success in connection with "upwind schemes" for the one- and two-dimensional Euler equations.[7,8,12] Here an obvious choice, which would lead to a reduction of the computer time per step to about one third, is to employ a single step 4×4 block-line-Gauss-Seidel method: all of the time and x derivatives in the left-hand sides of Eqs. (21-24) are evaluated implicitly together with the diagonal contributions of the y and z derivatives, so that only N^2 4×4 block-tridiagonal systems (of order N) have to be solved at every time level. By accounting for the previously evaluated nontridiagonal entries explicitly, the matrix A is effectively replaced by its three main diagonals plus its two additional nonzero lower diagonals. Furthermore, the ordering of the solution process is changed at every time step so as to account for the two additional nonzero upper or lower diagonals, alternately.

A Block-Point-Gauss-Seidel (BPGS) Method

By taking the logic behind the previous method to its extreme, an obvious choice presents itself: to replace matrix A with its lower or upper triangular part. In Eqs. (21-24) the diagonal contributions are accounted for implicitly, and the previously evaluated off-diagonal contributions are brought to the right-hand sides of the equations and accounted for explicitly. At every gridpoint location, a 4×4 linear system needs to be solved as in the BE method. However, due to its variable coefficients, the local 4×4 matrix cannot be triangularized, and a complete Gauss-Jordan elimination procedure, using diagonal pivot strategy, has been employed (here, as well as to solve the local linear systems within a general block-tridiagonal inversion routine in all of the present implicit methods).

A Simplified-Line-Gauss-Seidel (SLGS) Method

From their very definitions [Eqs. (9)] and their compatibility conditions [Eqs. (8)], it appears that the waves associated with the bicharacteristic variables C and D mainly propagate in the x direction, whereas the E and F waves and the G and H waves mainly propagate in the y and z directions, respectively. Therefore, it would seem appropriate to devise a numerical method exploiting such a property of the compatibility Eqs. (8), as done in Refs. 7 and 8 for the case of one- and two-dimensional flows. However, whereas Moretti[7,8] integrates the compatibility conditions directly, here Eqs. (13-18) are preferred for the two reasons previously discussed. In conclusion, the following simplified line-Gauss-Seidel method is proposed here: Eqs. (21) and (24) are solved coupled together for the ΔC and ΔD variables by means of a line-Gauss-Seidel method, implicit in the x direction, so that a 2×2 block-tridiagonal system of order N has to be solved at every y_j and z_k grid point location. Equations (22) and (23) are then solved by means of line-Gauss-Seidel methods implicit in the y and z direction, respectively, so that $2N^2$ additional scalar tridiagonal systems need to be solved. Obviously, Eqs. (25) and (26) are used to eliminate ΔF and ΔH from Eqs. (21-24) and all of the $\Delta C, \ldots, \Delta H$ terms already evaluated at any level of the computation process are accounted for in the right-hand sides of the equations. Furthermore, since the continuity Eq. (24) does not have a main direction of propagation, it is coupled to Eq. (22) to evaluate ΔE and ΔF implicitly in the y direction, and Eq. (23) to evaluate ΔG and ΔH implicitly in the z direction, at successive time steps.

As far as the boundary conditions are concerned, they can be implemented, in general, as suggested in Ref. 6 and are contained in both the matrix A and the vector $\boldsymbol{b}$, so that they are accounted for with the level of implicitness typical of every single method. However, for simplicity, in the present applications the exact solution of the continuum problem is enforced at all boundaries to provide homogeneous boundary conditions for the incremental bicharacteristic variables.

Results

In order to test the proposed methods, a simple, steady, one-dimensional spherical source flow of air ($\gamma = 1.4$) has been considered. For such a flowfield the continuity and energy equations are given as

$$a^5 v_r r^2 = c_1 \tag{32}$$

$$0.2 v_r^2 + a^2 = c_2 \tag{33}$$

v_r being the radial velocity component and r the radial distance from the origin, so that an "exact" solution can be easily computed for v_r and a. Such a "one-dimensional" flowfield has, however, a three-dimensional nature in a Cartesian coordinate system and can be a very suitable model problem if solved inside an appropriate computational domain, such as the unit cube with $2 \le x \le 3$, $-0.5 \le y \le 0.5$, and $-0.5 \le z \le 0.5$, used in the present calculations. In such a region, in fact, the flowfield under investigation is characterized by a dominant flow direction (the velocity vector $\boldsymbol{v}$ being at most at a 20 deg angle with respect to the x axis) as is the case of most compressible flows of practical interest (if a body-oriented coordinate system is used) and by both positive and negative values for the transversal velocity components v and w.

Three flow conditions have been considered: the subsonic flow corresponding to $c_1 = 3.2$ and $c_2 = 1.128$ and the supersonic and transonic flows corresponding to $c_1 = 4.2$ and $c_2 = 1.2205$. In the last case, an isentropic shock at $r = 2.15$ separates a supersonic region (for $r < 2.15$) from a subsonic one (for $r > 2.15$). All of the calculations were performed using two-point first-order-accurate upwind differences throughout and a uniform, rather coarse mesh with $\Delta x = \Delta y = \Delta z = 0.1$. A flowfield having the exact values for u and a and zero v and w has been used as a suitable initial condition. The solution was advanced in time by means of any of the proposed methods using a constant (in time) and uniform (in space) value of Δt, until the average absolute value of ΔC at all interior points was less than 10^{-6}. The final steady-state solution is obviously the same for all of the methods; the computed Mach number distribution along the x axis is plotted in Fig. 1 for each of the three flow cases vs the exact solution. It appears that the present numerical results are fairly good for the subsonic and supersonic case and qualitatively correct for the transonic one. In particular, the shock is captured in the correct mesh interval and no wiggles are present, in spite of the absence of any additional dissipation. However, for shocks as strong as that given in Fig. 1, a shock-fitting procedure is warranted.

The main purpose of this paper was to devise "efficient" implicit methods for the three-dimensional lambda-formulation equations. Therefore the performances of all of the present methods are given in Table 1 as the values of Δt leading to the fastest convergence and the corresponding number of

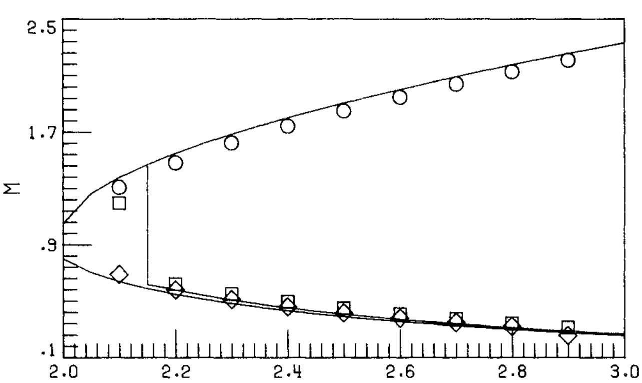

Fig. 1 Numerical (symbols) vs. exact (solid lines) solutions for spherical source flows.

Table 1 Performance of the various numerical methods

	Subsonic flow			Supersonic flow			Transonic flow		
Method	Δt	K	CPU, s	Δt	K	CPU, s	Δt	K	CPU, s
BE	0.03	178	57	0.03	62	21	0.03	403	128
BADI	0.3	35	361	0.3	38	398	0.3	86	887
BLGS	≥ 5	18	75	≥ 2	11	46	≥ 10	71	290
BPGS	≥ 5	29	49	≥ 2	11	20	≥ 10	76	126
SLGS	2.0	22	34	0.4	27	41	≥ 10	99	147

time steps (K) and CPU seconds (on a CDC Cyber 175 computer).

The following conclusions can be drawn from Table 1. For the supersonic flow case the BE and BPGS methods are clearly superior. This is obvious insofar as there is no upstream propagation in the x direction and thus the implicit methods use most of the CPU time accounting for zero entries. In terms of the number of iterations, the performance of the BLGS and BPGS methods are identical, as they should be (all of the x derivatives being approximated with backward differences). For the more relevant transonic and subsonic flow cases the BLGS method always requires the fewest iterations to converge. However, the BPGS, SLGS, and BE methods are the most efficient ones, whereas, the BADI method is consistently the least competitive one. It is noteworthy that all of the Gauss-Seidel methods are very robust insofar as they maintain a quasi-optimal convergence rate over a wide range of Δt values.

Among the three "best methods," the BPGS and the BE methods are considerably simpler to code and require less computer memory, a very critical resource when dealing with three-dimensional problems. Therefore, preliminary studies have been conducted to assess the influence on their convergence rate of refining the mesh and using three-point second-order-accurate upwind differences for the nonincremental terms in Eqs. (21-24). Both methods converge in a number of iterations which is roughly inversely proportional to the step size (e.g., for the subsonic flow problem, convergence is reached after 261 and 52 iterations for a 17^3 mesh and after 309 and 59 iterations for a 21^3 mesh, for the BE and the BPGS methods, respectively. However, it is noteworthy that for these calculations (performed on a VAX 11/750 computer), the BE method required about 2.5 more CPU time than the BPGS method. This indicates that the solution routine for the local 4×4 linear systems used in this study works much less efficiently on the Cyber 175 than on the VAX 11/750, and that the superiority of the BPGS method over the BE one is greater than it actually appears from Table 1. Also, the use of second-order-accurate spatial differencing for the nonincremental derivatives deteriorates the convergence rate of the BPGS method less than that of the BE method. Finally, the superiority of the BPGS method (with respect to the BE method) is expected to increase even further by using a variable Δt[5,6,12] and when more general boundary conditions are employed because the additional work will be relatively greater for the simpler BE method.

In conclusion, the BPGS method, by itself or as a robust smoother within a more general multigrid procedure, appears to be the most promising technique for solving three-dimensional compressible flow problems. Furthermore both the BLGS and SLGS methods proposed herein appear to be very promising alternatives to the ADI method of Refs. 5 and 6 for solving two-dimensional steady flows. In this process they are likely to outperform even the present BPGS method.

Acknowledgments

This research was supported in part by the Consiglio Nazionale delle Richerche and in part by the National Aeronautics and Space Administration under NASA Contract NAS1-17070 while the first author was in residence at the Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, Hampton, Va. Drs. D. Dwoyer and B. Van Leer have provided valuable suggestions and comments during most of this project. The ICASE environment was vital to the success of this effort.

References

[1] Butler, D.S., "The Numerical Solution of Hyperbolic System of Partial Differential Equations in Three Independent Variables," *Proceedings of the Royal Society of London*, *Series A*, Vol. 255, April 1960, pp. 232-252.

[2] Moretti, G., "The λ-Scheme," *Computers and Fluids*, Vol. 7, Sept. 1979, pp. 191-205.

[3] Zannetti, L. and Colasurdo, G., "Unsteady Compressible Flow: A Computational Method Consistent with the Physical Phenomena," *AIAA Journal*, Vol. 19, July 1981, pp. 851-856.

[4] Chakravarti, S.R., Anderson, D.A., and Salas, M.D., "The Split Coefficient Matrix Method for Hyperbolic Systems of Gas Dynamic Equations," AIAA Paper 80-0268, Jan. 1980.

[5] Dadone, A. and Napolitano, M., "An Implicit Lambda Scheme," *AIAA Journal*, Vol. 21, Oct. 1983, pp. 1391-1399.

[6] Dadone, A. and Napolitano, M., "Efficient Transonic Flow Solutions to the Euler Equations," AIAA Paper 83-0258, Jan. 1983; see also, "An Efficient ADI Lambda Formulation," *Computers and Fluids*, to appear.

[7] Moretti, G., "Fast Euler Solver for Steady One-Dimensional Flows," NASA CR-3689, June 1983.

[8] Moretti, G., "A Fast Euler Solver for Steady Flows," AIAA Paper 83-1940, July 1983.

[9] Moretti, G., "A Physical Approach to the Numerical Treatment of Boundaries in Gas Dynamics," *Numerical Boundary Condition Procedures*, NASA CP-2201, pp. 73-95.

[10] Karamcheti, K., *Principles of Ideal-Fluid Aerodynamics*, John Wiley & Sons, New York, 1966, pp. 140-146.

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

[12] Van Leer, B. and Mulder, W.A., "Relaxation Methods for Hyperbolic Equations," Technische Hogeschool Delft, The Netherlands, Tech. Rept., 1983.