The Discretization of the Full Potential Equation

Shlomy Shitrit and David Sidilkover [1]

Abstract

The discretization process of the full potential equation (FPE) both in the quasi-linear and in the conservation form, is addressed. This work introduce the first stage toward a development of a fast and efficient FPE solver, which is based on the algebraic multigrid (AMG) method. The mathematical difficulties of the problem are associated with the fact that the governing equation changes its type from elliptic (subsonic flow) to hyperbolic (supersonic flow). A pointwise relaxation method when applied directly to the upwind discrete operator, in the supersonic flow regime, is unstable. Resolving this difficulty is the main achievement of this work. A stable pointwise direction independent relaxation was developed for the supersonic and subsonic flow regimes. This stable relaxation is obtained by post-multiplying the original operator by a certain simple first order downwind operator. This new operator is designed in such a way that makes the pointwise relaxation applied to the product operator to become stable. The discretization of the FPE in the conservation form is based on the body-fitted structured grid approach. In addition the 2D stable operator in the supersonic flow regime was extended to 3D case. We present a 3D pointwise relaxation procedure that is stable both in the subsonic and supersonic flow regimes. This was verified by the Von-Neumann stability analysis.

Keywords: transonic flow, full potential equation, algebraic multigrid.

1. Introduction

The potential flow model is equivalent to the Euler equations for continuous, irrotational flows. For subsonic external and internal flows, the solution to the Euler and potential equations are in many cases almost identical. The difference between the solutions of the two models become more evident for supersonic flows with shock waves. The main advantage in the potential flow model, whenever adequate, is that there is only one equation to solve, instead of set of five equations for the 3D Euler system. In spite of the limitations associated with the potential flow model, it is still useful in engineering applications. A robust and efficient solver for the FPE may still be preferable in many cases over a more complicated Euler equation solver, provided its computational efficiency is substantially higher. One of such cases is an aerodynamic design problem, when the flow field solution should be computed repeatedly many times with variation of the body geometry. An efficient solver, even if for a simplistic model, can be highly desirable in such a situation. This fact makes the potential flow model very valuable in the design process since the basic physics of an inviscid flow field is still captured.

This work presents the first stage toward a development of a FPE solver, which is based on the algebraic multigrid (AMG) method [1, 2, 3, 4]. Since the 1980s, the AMG method has become a workhorse of large-scale computations due to its robustness and scalability. The main focus of this paper is on the discretization process of the FPE for solving transonic flow problems. The mathematical difficulties of the problem are associated primarily with the fact that the governing equation changes its type, being elliptic in the subsonic and hyperbolic in the supersonic regions of the flow [5]. Since these two cases differ in their properties, a suitable numerical approximation should be devised for each one of these two regions. It was first shown by Murman and Cole in 1971 [6] that the stable solution to the transonic small disturbances equation can be obtained by switching from central differencing in the subsonic region to upwind differencing in the supersonic region and applying the line implicit relaxation scheme. In 1973, Murman presented a solution of the Transonic Small Disturbance (TSD) equation by using central differencing for the subsonic regions and upwind differencing for the supersonic region [7]. After the work of Murman and Cole other researchers kept developing this idea. Steger and Lomax [8] presented the successive over-relaxation scheme (SOR) to solve the potential

equation. They presented transonic airfoil solutions. Garabedian and Korn [9] increased the order of accuracy for the Murman and Cole's scheme and also solved the FPE in nonconservative form. In 1972 Ballhaus and Bailey [10] and Bailey and Steger [11] both solved transonic flows around wings using the TSD equation. The approach by Murman and Cole was generalized by Jameson to the FPE and general flow direction [12]. The iterative process developed there was based upon the concept of artificial time, and in order to maintain stability it involved three levels. As can be seen, the time between the 1970s and 1980s was a period of rapid development of potential flow solvers. Contributions were made by numerous researchers. Following the pioneering work by Jameson and his co-workers [13, 14, 15] that opened a way for solving the Euler and Navier-Stokes equations, further research on development of an efficient FPE solver was abandoned. The industry adopted the new methodology, that has been used widely until now, while undergoing continuously an evolutionary process. However, potential flow analysis still plays an important role in aircraft design and optimization. Because of the speed that potential solutions can be attained, potential solvers fit well into the design process. Although the accuracy of the potential flow model is less than solving the Euler or Navier-Stokes equations, the basic physics of an inviscid flow field is still captured.

Usually, the relaxation used as an ingredient of the AMG algorithm is a pointwise one. One of the central contributions of this work is a development of a stable pointwise direction-independent relaxation for the entire range of the flow speed, from low Mach number flow to transonic and supersonic regions. This development was a prerequisite for considering an application of AMG to the transonic flow problem. However, a pointwise relaxation method when applied directly to the upwind discrete operator, in the supersonic flow regime, is unstable. Resolving this difficulty is the main achievement of this work. A stable pointwise direction-independent relaxation was developed for the supersonic and subsonic flow regimes. This stable relaxation is obtained by post-multiplying the original operator by a certain simple first order downwind operator. This new operator is designed in such a way that makes the pointwise relaxation applied to the product operator to become stable. In the present work we insist on employing a pointwise relaxation (Jacobi or symmetric Gauss–Seidel) for the following reasons: first, it is relatively simple and can be parallelized effectively. Second, pointwise relaxation is desirable in the AMG context since (unlike the line relaxation) it releases us from the reliance on the grid geometry. Using line relaxation within the geometric multigrid context, while the flow is grid aligned is quite acceptable, but it is not clear how to extend it to the general flow direction case. Third, Jacobi or symmetric Gauss-Siedel methods are independent of the flow direction, and, therefore, of the problem's geometry.

The purpose of this research is to introduce the discretization process of the FPE in the conservation form. Simulation results of elementary and advanced applications were reported in [16, 17], while using the same mathematical basis for solving elementary and advanced applications (such as flow around a cylinder, channel with a bump, nonsymmetric airfoil). This paper is organized as follows: The discretization of the FPE in the quasi-linear form is presented in Section 2. The transonic flow problem is introduced in this Section and our approach to solve the supersonic region is presented. In Section 3, the discretization of the FPE in the conservation form is addressed. Section 4 demonstrates a proof of feasibility to get a stable 3D discrete operator in the supersonic flow regime, while applying the same rationale as is applied in the 2D case. This approach was verified by numeral stability analysis for various flow directions.

2. Discretization of the FPE in the quasi-linear form

Transonic flow is a flow that subsonic and supersonic regions coexist. Usually the supersonic region of the flow is bounded by sonic lines with smooth gradual acceleration of the flow from subsonic to supersonic, and by shock wave through, which the flow slows down to subsonic speed. This type of flow occurs in a variety of applications such as flow over aircraft wings, helicopter rotor blades, compressors, turbines, inlets, etc.

Transonic flow can be described by the FPE in a quasi-linear form,

$$(c^2 - u^2)\phi_{xx} - 2uv\phi_{xy} + (c^2 - v^2)\phi_{yy} = 0 \tag{1}$$

where ϕ is the potential, u and v are the velocity components, and c is the speed of sound. This equation can be expressed (see [12]) by writing the equation in a new Cartesian

coordinate system $s - n$ where the s-axis is aligned with the flow direction and n is normal to it. The angle between the x-axis, and s-axis equals to the flow angle θ. Equation (1) becomes

$$(1 - M^2)\phi_{ss} + \phi_{nn} = 0, \tag{2}$$

where ϕ_{ss} and ϕ_{nn} are

$$\phi_{ss} = \cos^2(\theta)\phi_{xx} + 2\sin(\theta)\cos(\theta)\phi_{xy} + \sin^2(\theta)\phi_{yy}, \tag{3}$$

$$\phi_{nn} = \sin^2(\theta)\phi_{xx} - 2\sin(\theta)\cos(\theta)\phi_{xy} + \cos^2(\theta)\phi_{yy}. \tag{4}$$

The transonic flow problem is a very difficult one from the computational standpoint since Eq. 2 changes its type. It is elliptic in the subsonic and hyperbolic in the supersonic flow region. Until the year of 1971 the attempts of scientists to solve numerically the transonic flow problem were successful only in the subsonic flow regime. Earl Murman and Julian Cole [6] were the first to recognize the fact that since the equation changes its character from elliptic to hyperbolic, different discrete operators should be applied for each of these two cases. Murman and Cole observed that the dependence of the point (i, j) on the entire neighborhood is physically correct only in the subsonic case where the PDE is elliptic. In the supersonic regions the PDE is hyperbolic and point (i, j) is influenced by points that are bounded by the characteristic lines. Murman and Cole recognized that in the supersonic regions, since the domain of dependence of a certain grid-point (i, j) is bounded by the characteristics, the numerical scheme should reflect this upstream dependency. Therefore, they proposed to use the upwind scheme.

The small disturbance equation was solved in this way, where the flow was aligned with the x-axis. The solution was applied as follows — a line Gauss–Seidel relaxation along the y-axis grid lines in the order that follows the flow direction. In the present work we insist in using a pointwise relaxation (for the reasons stated previously). If the flow is subsonic (and central differencing is used), a simple pointwise relaxation (Gauss–Seidel or damped Jacobi) is stable. In the supersonic case, however, (when an upwind difference approximation is used), such a relaxation is unstable (the amplification factor is greater than unity for certain components). The main achievement of the work presented in this paper is the development of a relaxation procedure that is stable when applied in the pointwise manner.

2.1. Subsonic flow – the discrete approximation

Assume the equation is to be discretized on a uniform Cartesian grid while $\Delta x = \Delta y = h$. Also recall that the aim of this work is to construct not only a stand-alone FPE solver but also a building block for a future algorithm for solving the full flow equations based on the factorizable discretization methods (see [18]). Therefore, when constructing a finite difference approximation to ((2)) we apply the following rationale: the discrete form of this equation is expected to follow from the form of the full potential co-factor when the system of Euler equations is discretized by the means of the factorizable scheme. Therefore, the concepts of "narrow" and "wide" approximations to derivatives (see [18]) are applied here as well. One way to augment the five-point stencil to nine points is by approximating the streamwise derivative by six nodes instead of three. Practical experience has shown that AMG's performance with "wide" approximation derivatives in the streamwise direction, for subsonic as well as transonic and supersonic flow, is much better, in terms of convergence properties, compared to "narrow" approximations. The reader is referred to [19] for some typical examples demonstrating this.

A convenient way to express graphically a local discrete operator is through its stencil. A standard way to represent a stencil is by a matrix (see, for instance, [20]).

1. When the flow is grid-aligned (the s-axis coincides with the x-axis), the streamwise derivative is approximated by the "wide" central second difference

$$\phi_{ss}^h = \frac{1}{4h^2} \begin{bmatrix} 1 & -2 & 1 \\ 2 & -4 & 2 \\ 1 & -2 & 1 \end{bmatrix}. \tag{5}$$

The cross-flow second derivative is approximated by the standard "narrow" difference

$$\phi_{nn}^{h} = \frac{1}{h^2} \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}. \tag{6}$$

2. In the case of zero Mach number flow an approximation to the entire equation is a discrete Laplacian of the following form:

$$\Delta^{h}\phi = \frac{1}{4h^2} \begin{bmatrix} 1 & 2 & 1 \\ 2 & -12 & 2 \\ 1 & 2 & 1 \end{bmatrix}. \tag{7}$$

3. It follows from here that the concepts of "wide" and "narrow" second finite differences have to be generalized to the arbitrary direction.

First, we generalize a "narrow" second finite difference to a general direction. Consider the following finite difference stencils:

$$
\begin{aligned}
\phi_{xx}^{s} = \phi_{xx}^{n} &= \frac{1}{h^2} \begin{bmatrix} 1 & -2 & 1 \end{bmatrix}, \\
\phi_{yy}^{s} = \phi_{yy}^{n} &= \frac{1}{h^2} \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}, \\
\phi_{xy}^{s} &= \frac{1}{2h^2} \begin{bmatrix} 0 & -1 & 1 \\ -1 & 2 & -1 \\ 1 & -1 & 0 \end{bmatrix}, \\
\phi_{xy}^{n} &= \frac{1}{2h^2} \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}.
\end{aligned}
\tag{8}
$$

Then the "narrow" second difference in direction n can be given by the following expression:

$$\phi_{nn}^{h} = sin^2(\theta)\phi_{xx} + 2sin(\theta)cos(\theta)\phi_{xy}^{n} + cos^2(\theta)\phi_{yy}. \tag{9}$$

Note that this difference approximates the cross-flow second derivative. Also it is sufficient for the purpose of this work to consider the case where the flow direction forms with, say, the x-axis an angle

$$0^o \leq \theta \leq 45^o. \tag{10}$$

Therefore, the expression (9) covers all the relevant situations. The "wide" difference in general flow direction is defined simply by subtracting (9) from the "wide" approximation to Laplacian (7) (exactly as described by Jameson's rotated difference scheme [12]). In this approach the operator discretization is obtained by using the already defined derivatives, with no need to construct an approximation for each flow's direction.

2.2. Supersonic flow

In the supersonic region the equation changes type from elliptic to hyperbolic, and, therefore upwind differencing should be used to approximate the second derivative in the flow's direction ϕ_{ss} (see 8). Again, in the case of a grid-aligned flow, the second derivative in the flow direction is going to be approximated by *one-sided* "wide" second difference. In the case of a general flow direction the following approximations to the derivatives are employed:

$$\partial_{xx}^{s} = \frac{1}{h^2} \begin{bmatrix} \left(\frac{1}{4} - \frac{1}{2}sin^2(\theta)\right) & \left(-\frac{1}{2} + sin^2(\theta)\right) & \left(\frac{1}{4} - \frac{1}{2}sin^2(\theta)\right) & \left(\frac{1}{4} - \frac{1}{2}sin^2(\theta)\right) & 0 & 0 \\ \frac{1}{2} & -1 & \frac{1}{2} & 0 & 0 \\ \left(\frac{1}{4} - \frac{1}{2}sin^2(\theta)\right) & \left(-\frac{1}{2} + 2sin^2(\theta)\right) & \left(\frac{1}{4} - \frac{5}{2}sin^2(\theta)\right) & sin^2(\theta) & 0 \end{bmatrix}, \tag{11}$$

$$\partial_{yy}^{s} = \frac{1}{h^2} \begin{bmatrix} 0 & sin^2(\theta) & 0 & 0 & 0 \\ 0 & \left(\frac{1}{4} - \frac{5}{2}sin^2(\theta)\right) & \frac{1}{2} & \left(\frac{1}{4} - \frac{1}{2}sin^2(\theta)\right) & 0 \\ 0 & \left(-\frac{1}{2} + 2sin^2(\theta)\right) & -1 & \left(-\frac{1}{2} + sin^2(\theta)\right) & 0 \\ 0 & \left(\frac{1}{4} - \frac{1}{2}sin^2(\theta)\right) & \frac{1}{2} & \left(\frac{1}{4} - \frac{1}{2}sin^2(\theta)\right) & 0 \end{bmatrix}, \tag{12}$$

$$\partial_{xy}^s = \frac{1}{h^2} \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 1 & -1 & 0 \end{bmatrix}. \tag{13}$$

The approximations to ϕ_{nn} remain the same as in the subsonic case.

2.3. Devising a stable relaxation procedure

As it was stated above, we restrict ourselves in this work to usage of a pointwise relaxation. Therefore, we have to make sure there is a variant of such a relaxation at our disposal that not only is stable for all the cases of interest but also provides a good smoothing. While simple damped Jacobi and symmetric Gauss–Seidel relaxation schemes are suitable for the subsonic case, both of them are unstable in the supersonic case. This can be easily verified by Von–Neumann analysis (see [19]).

A well-known approach that can help in the situation described above is the Kaczmarz relaxation [21, 20]. According to the theory, this relaxation always gives smoothing (provided the operator is not semi-definite), but it is rather inefficient. Instead of solving the algebraic linear system $Ax = b$ directly, it suggests to solve

$$AA^T y = b, \tag{14}$$

where

$$x = A^T y. \tag{15}$$

The implementation of the Kaczmarz relaxation in our case turns out to be quite expensive: since the matrix A has to reflect the upwind (second-order) difference operator, matrix A^T will have to reflect downwind differencing. Their product, therefore, is going to be rather cumbersome. Therefore, our approach was to find a simpler matrix $\tilde{A}$ to replace A^T and to solve a system $A\tilde{A}y = b$ where $x = \tilde{A}y$. The difference operator $\tilde{L}$ resulting in matrix $\tilde{A}$, for the cases of $0^o \leq \theta \leq 45^o$ was chosen to be of the following form:

$$
\begin{aligned}
\tilde{L} = & \left(\frac{1}{4} sin^2(\theta) + \frac{1}{4} sin(\theta)cos(\theta) \right) \phi_{i-1,j+1} \\
& + \left(\frac{1}{4} - \frac{1}{2} sin(\theta)cos(\theta) \right) \phi_{i+1,j+1} \\
& + \frac{1}{2} sin^2(\theta) \phi_{i,j+1} \\
& + \left(\frac{1}{2} - \frac{1}{2} sin^2(\theta) \right) \phi_{i+1,j} \\
& + \left(\frac{1}{4} cos^2(\theta) + \frac{1}{4} sin(\theta)cos(\theta) \right) \phi_{i+1,j-1} - \phi_{i,j}.
\end{aligned}
\tag{16}
$$

For an illustration, we present here this operator in two special cases. When the flow is grid aligned, $\tilde{L}$ is given as follows:

$$\tilde{L} = \frac{1}{h} \begin{bmatrix} 0 & 0 & \frac{1}{4} \\ 0 & -1 & \frac{1}{2} \\ 0 & 0 & \frac{1}{4} \end{bmatrix}, \tag{17}$$

while for the diagonal direction flow, the discrete operator is

$$\tilde{L} = \frac{1}{h} \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & -1 & \frac{1}{4} \\ 0 & 0 & \frac{1}{4} \end{bmatrix}. \tag{18}$$

It is important to note that the performance of the overall AMG algorithm depends strongly on the choice of operator $\tilde{L}$. The particular structure presented in (16) seems to lead to the best results. AMG was applied to the product operator, $L\tilde{L}$, which is h-elliptic, according to the concept of h-ellipticity introduced by Brandt [22]. Figure (1) presents the amplification factor (the function that describes how the error amplitudes evolve - for convergence of the method, we must have amplification factor lower than unity) for the Gauss–Seidel relaxation method, as a surface over the frequencies $[-\pi, \pi] \times [-\pi, \pi]$. The flow conditions are $M = 1.1$ and $\theta = 0°$. One can see that the whole surface is lower than unity, which indicates that the method is stable. Clearly, the amplification factor decreases as the modes become more oscillatory. The smoothing properties of such a relaxation are much better than that of a classical Kaczmarz. To the best of our knowledge, there are no other known results of an application of a simple pointwise relaxation and of the AMG method based on it for the transonic flow problem.
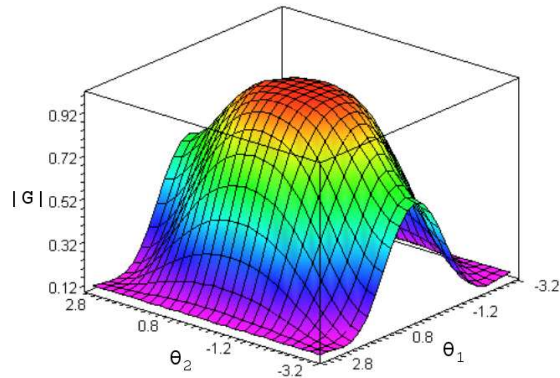

Figure 1:   Amplification factor,$|G(\theta_1, \theta_2)|$, for the damped Gauss–Siedel method, applied to the discrete upwind operator in two dimensions, shown as a surface over the region $[-\pi, \pi] \times [-\pi, \pi]$.

Now, when we have the stable and efficient relaxation schemes for cases of subsonic, sonic, and supersonic flow, it is necessary to combine them together. When the flow is subsonic, a central difference is used for the second derivative in the flow direction, while pointwise relaxation can be applied directly in conjunction with the matrix A (or operator L). If the flow is supersonic, we apply relaxation directly with the product of downwind and upwind operators $L\tilde{L}$ (or matrix $A\tilde{A}$). Since the operator $\tilde{L}$ cannot suddenly appear in the supersonic flow regime, it must exist also in the subsonic flow regime. Therefore, these two schemes have to be combined. One way of doing this is still to keep $\tilde{A}$ in the subsonic case while modifying it so that it gradually becomes a unity matrix as the flow speed decreases.

Once we have a stable relaxation procedure based upon the combined operator, we can proceed to devising the overall solver. However, when operator L is nonlinear it becomes complicated to construct and apply such a combined operator. For the purpose of overcoming this difficulty, we devised and applied the distributive relaxation technique such that it is identical to the simple pointwise relaxation on the operator $L\tilde{L}$ in the linear case.

2.4. Residual distributive relaxation

Denote the residual of the discrete equation at point

$$r = f - L\phi^c, \tag{19}$$

where ϕ^c stands for a current approximation to ϕ. Then (in the case of a linear operator L), the equation for correction can be written as follows

$$L(\delta\phi) = r. \tag{20}$$

However, as discussed previously, a pointwise relaxation procedure applied directly to the above equation is unstable in the hyperbolic case. Therefore, it was suggested to apply relaxation directly to the "combined" operator

$$L\tilde{L}(\delta y) = r, \tag{21}$$

and, after computing the correction δy, to evaluate the correction $\delta\phi = \tilde{L}(\delta y)$. The entire AMG solution algorithm can address the combined operator $L\tilde{L}$ (or matrix $A\tilde{A}$) and the corresponding unknown y (or correction δy). However, when operating on the finest level, we need to deal with the problem in terms of the unknown ϕ. One way of doing this can be to translate the correction δy into the correction $\delta\phi$ at the end of a relaxation sweep or a multigrid cycle. An alternative way is to perform this translation immediately after computing the correction at each and every point

$$(\delta\phi)_{i,j} = \tilde{L}\,(\delta y)_{i,j}\,. \tag{22}$$

Note, that due to the structure of operator $\tilde{L}$, at grid point i,j we have

$$(\delta\phi)_{i,j} = (\delta y)_{i,j}\,. \tag{23}$$

Therefore, this procedure can be described as evaluating correction $(\delta\phi)_{i,j}$ at a point based on the original operator L. However, in addition to introducing it at point i,j and in order to avoid the instability of such a "direct" relaxation procedure, we also distribute its fractions at certain downstream points according to the operator $\tilde{L}$.

The residual distribution method consists of distributing fractions of the flux balance in a cell to the adjacent cells in the upwind direction, with weights summing up to one for consistency. The idea is to build a procedure that accurately mimics the structure of the operator $\tilde{L}$ in all the flow conditions that we have tested. The distribution of the residual in every point in the field is done in a systematic procedure, exactly according to the structure of the operator $\tilde{L}$.

Consider a subsonic grid aligned flow, in a cell center far from the boundaries. After computing the residual, it is distributed to three adjacent cells according to the operator $\tilde{L}$,

$$
\begin{aligned}
\phi_{i-1,j+1} &= \phi_{i-1,j+1} + M_{i,j}^2 \frac{1}{4} r_{i,j}, \\
\phi_{i-1,j} &= \phi_{i-1,j} + M_{i,j}^2 \frac{1}{2} r_{i,j}, \\
\phi_{i-1,j-1} &= \phi_{i-1,j-1} + M_{i,j}^2 \frac{1}{4} r_{i,j},
\end{aligned}
\tag{24}
$$

where $r_{i,j}$ is the residual computed in the cell (i,j) and $M_{i,j}$ is the local Mach number through the face (i,j). As one can see, the the additional weight to each cell is modified so it gradually becomes zero as the flow speed decreases.

3. Discretization of the FPE in the conservation form

Transonic flow can be described by the FPE that is derived from the Euler equations by assuming that the flow is inviscid, isentropic, and irrotational. This potential flow will be treated in the conservation form:

$$\frac{\partial}{\partial x}\,(\rho u) + \frac{\partial}{\partial y}\,(\rho v) = 0, \tag{25}$$

where u and v are the velocity components in the Cartesian coordinates x and y, respectively, and ρ is the density. The velocity components are the gradient of the potential ϕ,We give details about complexity issues and the advantages and difficulties in using a pointwise relaxation method.

$$u = \frac{\partial\phi}{\partial x}, \qquad v = \frac{\partial\phi}{\partial y}. \tag{26}$$

The density ρ is computed from the isentropic formula:

$$\frac{\rho}{\rho_\infty} = \left(1 + \frac{\gamma - 1}{2} \left(V_\infty^2 - \phi_x^2 - \phi_y^2 \right) \right)^{\frac{1}{\gamma - 1}}, \qquad (27)$$

where γ is the ratio of specific heats and V_∞ is the free-stream velocity and ρ_∞ is the free-stream density. The relation between the local speed of sound a and the flow speed is defined by Bernoulli's equation:

$$a = \left(a_\infty^2 - \frac{\gamma - 1}{2} \left(V_\infty^2 - \phi_x^2 - \phi_y^2 \right) \right). \qquad (28)$$

The discretization of the FPE in the conservation form is based on the same rationale that was applied in the quasi-linear case.

The purpose of the current section is to describe a procedure for constructing stable finite volume approximations to the conservation form of the full potential equation. The strategy of discertizing the FPE in the conservation form is based on the rotated difference approach introduced by Jameson [12] and was used in the quasi-linear form. However, this approach is not made directly. Instead it is accomplished indirectly by following the same rationale. We start from the (2) that can be formulated as,

$$\nabla^2 \phi - M^2 \frac{\partial^2}{\partial s^2} \phi = 0. \qquad (29)$$

Let us look at both terms (2) from a numerical standpoint. Note that when the Mach number is close to zero (incompressible flow) the second term can be neglected; thus we are left with $\nabla^2 \phi$ that is discretized by a certain type central differencing, according to (7). As the Mach number increases the second term that describes the second derivative in the streamwise direction, actually determines the "dynamics" of the flow. When the flow is subsonic, a central difference is used for the second derivative in the flow direction (ϕ_{ss}), while pointwise relaxation is applied directly in conjunction with the matrix A (or operator L). If the flow is supersonic, we apply relaxation directly with the product of downwind and upwind operators $L\tilde{L}$ (or matrix $A\tilde{A}$). We would like to apply the same rationale to the discretization of the FPE in the conservation form, while the advantages that were obtained in the quasi-linear case discretization, would be implemented. The relation between the discretization approach applied in the quasi-linear form and the FPE in the conservation form will now be covered. Expanding the FPE and rearranging terms, Eq. (29) can be reformulated as,

$$\rho \nabla^2 \phi + \left(\phi_x \frac{\partial}{\partial x} + \phi_y \frac{\partial}{\partial y} \right) \rho = 0, \qquad (30)$$

where the density ρ is given in Eq. (27). Note that Eq. (29) and (30) have a similar structure. The density parameter ρ plays two roles. In the first term it serves as a constant. In the second term it serves as an unknown variable. As one can see, the second terms in both of the above equations are identical. Therefore, the same rationale applied in the quasi-linear case can be applied to the conservation form. The description of the discretization technique is as follows: For the first term $\nabla^2 \phi$ the fluxes are computed by a central discretization independent of the flow direction and speed. The dynamics of the flow is reflected in the second term that is discretized in such a way that the result is a "wide" approximation in the streamwise direction.

The criterion for selecting the best discretization consists mainly of how accurate the discretization is in computing the gradients and how generally applicable the algorithm can be. An initial test to verify the discretization's accuracy is to check that the method can reproduce a free-stream velocity applied to an arbitrary mesh. If not, then the discretization will not be acceptable. The second criterion deals with how general the discretization is. It is desired to attain solutions on highly stretched irregular structured grids, and flow in various speeds and directions. The flux through a given cell's face is a product of the velocity vector and the density, which are both functions of the potential ϕ. The type of the discretization approach, central fluxes, or upwind fluxes, is determined by the local Mach number across the cell's face. So, the first obstacle in forming the flux is therefore calculating the velocity vector at each cell's face.

3.1. Velocity components

The velocity field can be found by calculating the gradients of ϕ in the x, y, and z directions. For an orthogonal structured grid, this calculation is straight forward due to the Cartesian ordering of the cells. For a nonorthogonal grid, it is less clear how to formulate and compute the gradients of ϕ. Before obtaining the divergence, however, one needs to define the physical components of the velocity vector for each face in the grid. The velocity components are derived in the covariant coordinate system (ξ, η). Consider for example the face at the half node $(i - 1/2, j)$ as is sketched in Figure 2. The velocity vector of the flow through this face has two covariant components as follows:

$$V_{cov} = V_\xi \hat{\xi} + V_\eta \hat{\eta}, \tag{31}$$

where V_ξ is approximated by a "narrow" derivative in ξ-direction and V_η is derived by splitting a central difference between both sides of the face, in η-direction. It is done as follows:

$$\begin{aligned}
\vec{V_\xi} &= \frac{(\phi_{i,j} - \phi_{i-1,j})}{\triangle \xi_{i,j}}, \\
V_\eta &= \frac{1}{2}\left(\frac{\phi_{i,j+1} - \phi_{i,j-1}}{\triangle \eta_{i,j+1} + \triangle \eta_{i,j}}\right) + \frac{1}{2}\left(\frac{\phi_{i-1,j+1} - \phi_{i-1,j-1}}{\triangle \eta_{i-1,j+1} + \triangle \eta_{i-1,j}}\right).
\end{aligned} \tag{32}$$

Now when we have the covariant velocity components we need to express the velocity in the Cartesian coordinate system in order to compute the local speed of sound $a_{i,j}$ and then the Mach number through this face. The transformation from the covariant coordinate system to the Cartesian coordinate system is done by multiplying the covariant vector by the inverse of the matrix M,

$$[u, \ v]^T = M^{-1}\left[V_\xi, \ V_\eta\right]^T, \tag{33}$$

and the velocity absolute value is,

$$q^2 = \left(u^2 + v^2\right). \tag{34}$$

The local speed of sound is then

$$a^2 = a_\infty^2 + \frac{(\gamma - 1)}{2}\left(V_\infty^2 - u^2 - v^2\right), \tag{35}$$

where $a_\infty = V_\infty/M_\infty = 1$. Finally, the Mach number (the ratio of the flow speed to the local speed of sound) M at the half point $(i - 1/2, j)$ is $M = \frac{q}{a}$.
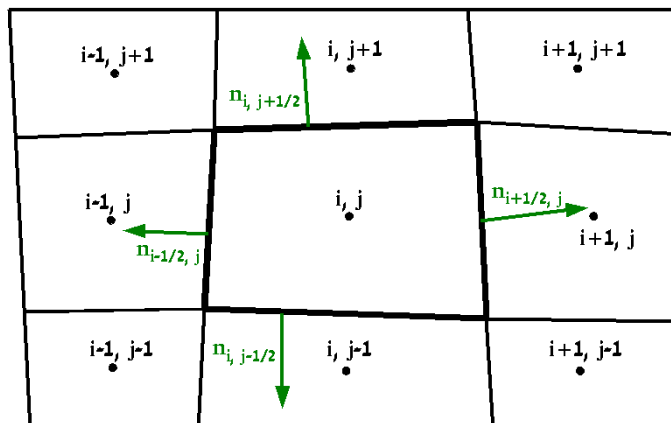

Figure 2: Sub-domain or control volume surrounding a node (i, j).

3.2. Flux calculation — incompressible flow

The computational space is divided into quadrilateral cells where the FPE is solved in each cell separately. Using Cartesian coordinate(x, y, z) the FPE is written as follows:

$$\frac{d}{dx}\left(\rho \cdot u\right) + \frac{d}{dy}\left(\rho \cdot v\right) = 0. \tag{36}$$

The velocity components u and v are calculated as the gradient of the potential ϕ.

$$u = \phi_x, \qquad v = \phi_y. \tag{37}$$

The flow is assumed to be uniform in the far field with a Mach number M_∞. At the body surface, "no penetration" boundary condition is applied,

$$\overrightarrow{V} \cdot \overrightarrow{n} = 0, \tag{38}$$

where the product $\overrightarrow{V} \cdot \overrightarrow{n}$ is the velocity component normal to the surface. The density is computed from the isentropic formula (27). The space discretized form of (36) in the covariant coordinate system (ξ, η) can be written as follows:

$$\frac{d}{d\xi}\left(F\right) + \frac{d}{d\eta}\left(G\right) = 0, \tag{39}$$

where $F = \rho\left(\phi_\xi, \phi_\eta\right)\phi_\xi$ and $G = \rho\left(\phi_\xi, \phi_\eta\right)\phi_\eta$. From the notation above, the fluxes labeled F and G are constructed of terms that contribute to the flux in the ξ and η directions.

The conservatively requirement on (39) will be satisfied if the scheme can be written under the form:

$$\frac{\left(f_{\left(i+\frac{1}{2},j\right)} - f_{\left(i-\frac{1}{2},j\right)}\right)}{\triangle\xi} + \frac{\left(f_{\left(i,j+\frac{1}{2}\right)} - f_{\left(i,j-\frac{1}{2}\right)}\right)}{\triangle\eta} = 0, \tag{40}$$

where f is the numerical flux at the cell's faces mid points $\left(i - \frac{1}{2}, j\right)$, $\left(i + \frac{1}{2}, j\right)$, $\left(i, j + \frac{1}{2}\right)$ and $\left(i, j - \frac{1}{2}\right)$.

In order to simplify the derivation, let introduce first the incompressible flow equation that is characterized by a constant density, referenced to a uniform free-stream density ρ_∞, $\rho = \rho_\infty = 1$, this results in

$$\frac{d}{d\xi}\left(\phi_\xi\right) + \frac{d}{d\eta}\left(\phi_\eta\right) = 0. \tag{41}$$

In a case of a low Mach number flow an approximation to the entire equation is a discrete nine-point Laplacian, exactly as the quasi-linear. Now, let express ϕ_ξ and ϕ_η to their covariant components,

$$\begin{aligned} \phi_\xi &= F_\xi\hat{\xi} + F_\eta\hat{\eta}, \\ \phi_\eta &= G_\xi\hat{\xi} + G_\eta\hat{\eta}, \end{aligned} \tag{42}$$

where the covariant velocity components, F_ξ, F_η, G_ξ, and G_η, which are calculated at the face $\left(i - \frac{1}{2}, j\right)$ are discretized as follows:

$$\begin{aligned} F_\xi &= \frac{3}{4}\frac{\left(\phi_{i,j} - \phi_{i-1,j}\right)}{\triangle\xi_{i,j}} + \frac{1}{8}\frac{\left(\phi_{i,j+1} - \phi_{i-1,j+1}\right)}{\triangle\xi_{i,j+1}} + \frac{1}{8}\frac{\left(\phi_{i,j-1} - \phi_{i-1,j-1}\right)}{\triangle\xi_{i,j-1}}, \\ F_\eta &= \frac{1}{2}\frac{\left(\phi_{i,j+1} - \phi_{i,j-1}\right)}{\triangle\eta_{i,j+1} + \triangle\eta_{i,j}} + \frac{1}{2}\frac{\left(\phi_{i-1,j+1} - \phi_{i-1,j-1}\right)}{\triangle\eta_{i-1,j+1} + \triangle\eta_{i-1,j}}, \\ G_\xi &= \frac{3}{4}\frac{\left(\phi_{i,j} - \phi_{i,j-1}\right)}{\triangle\eta_{i,j}} + \frac{1}{8}\frac{\left(\phi_{i+1,j} - \phi_{i+1,j-1}\right)}{\triangle\eta_{i+1,j}} + \frac{1}{8}\frac{\left(\phi_{i-1,j} - \phi_{i-1,j-1}\right)}{\triangle\eta_{i-1,j}}, \\ G_\eta &= \frac{1}{2}\frac{\left(\phi_{i+1,j} - \phi_{i-1,j}\right)}{\triangle\eta_{i+1,j} + \triangle\eta_{i,j}} + \frac{1}{2}\frac{\left(\phi_{i+1,j-1} - \phi_{i-1,j-1}\right)}{\triangle\eta_{i+1,j-1} + \triangle\eta_{i+1,j}}. \end{aligned} \tag{43}$$

The parameters $\triangle\xi$ and $\triangle\eta$ are the grid spacing in ξ and η directions, respectively. The distance separating the cell's centers (i, j) and $(i - 1, j)$ is $\triangle\xi_{i,j}$. The horizontal distance separating the cells (i, j) and $(i + 1, j)$ is $\triangle\xi_{i+1,j}$. Likewise, the vertical distances that separate

(i, j) from $(i-1, j)$ and $(i, j+1)$ are $\triangle\eta_{i,j}$ and $\triangle\eta_{i,j+1}$, respectively. Spacings are computed during the grid generation phase, and are stored as one-dimensional arrays, for later use during the discretization. Since the grid is not uniform we have to take each relevant grid spacing in order to get an accurate discretization. The approximations for the above covariant velocities at the faces $(i + 1/2, j)$ and $(i, j + 1/2)$ are done in the same way.

In order to get the flux that crosses each face in the cell (i, j), the velocity components must be projected normal to the face through which the flux is computed. Hence, the fluxes are approximated as follows,

$$
\begin{aligned}
F_{i-\frac{1}{2},j} &= \left[\begin{array}{c} F_\xi \\ F_\eta \end{array}\right]_{i-\frac{1}{2},j} \cdot \vec{n}_{i-\frac{1}{2},j}, \\
F_{i+\frac{1}{2},j} &= \left[\begin{array}{c} F_\xi \\ F_\eta \end{array}\right]_{i+\frac{1}{2},j} \cdot \vec{n}_{i+\frac{1}{2},j}, \\
G_{i,j-\frac{1}{2}} &= \left[\begin{array}{c} G_\xi \\ G_\eta \end{array}\right]_{i,j-\frac{1}{2}} \cdot \vec{n}_{i,j-\frac{1}{2}}, \\
G_{i,j+\frac{1}{2}} &= \left[\begin{array}{c} G_\xi \\ G_\eta \end{array}\right]_{i,j+\frac{1}{2}} \cdot \vec{n}_{i,j+\frac{1}{2}}.
\end{aligned}
\tag{44}
$$

Computation of the remaining terms is done in a similar fashion, using the four-step process described above.

3.3. Compressible flow – subsonic flow ($M \leq 1$)

When the Mach number is increased, the density of the fluid changes with respect to the pressure. This case of compressible flow is distinguished from the previous incompressible flow in that the density can no longer be considered constant. Consider the term $\frac{d}{d\xi}\left[\rho\left(\phi_\xi, \phi_\eta\right)\phi_\xi\right]$ in (39). The terms ϕ_x and ϕ_y in (27) reflect most of the flow's "dynamic". The discretization approach of these two terms must take into account the value of the Mach number through the face and the flow's direction. Consider for example the face $(i - 1/2, j)$. The term $\frac{d}{d\xi}\left(\rho\left(\phi_\xi, \phi_\eta\right)\phi_\xi\right)$ in (39) may be computed at the half node $(i - 1/2, j)$ as follows:

1. First, the discretization of ϕ_ξ (fluxes of the Laplacian) that were presented in the previous subsection, holds.
2. Next, the Cartesian velocity components u and v, the contravariant velocity components V_ξ and V_η, the local speed of sound, $a_{i,j}$, and the local Mach number $M_{i,j}$, are computed at the half node $(i - 1/2, j)$.
3. Now, when the Local Mach number is available across the face $(i - 1/2, j)$, and the flow direction is known we can decide how to discretized the velocity ϕ_ξ (recall that we are dealing with the flux through the face $\left(i - \frac{1}{2}, j\right)$ in the equation for the density

$$
\rho\left(\phi\right) = \rho_\infty\left[\frac{(\gamma-1)}{2}\left(V_\infty^2 - \phi_\xi^2 - \phi_\eta^2\right)\right]^{\frac{1}{(\gamma-1)}},
\tag{45}
$$

where the velocity vector of the flow through this face has two covariant components as follows:

$$
V_{cov} = V_\xi\hat{\xi} + V_\eta\hat{\eta}.
\tag{46}
$$

The discretization of ϕ_ξ depends on the direction of the flow relative to the vector $\hat{n}$, normal to the face, as is sketched in Figure 3.
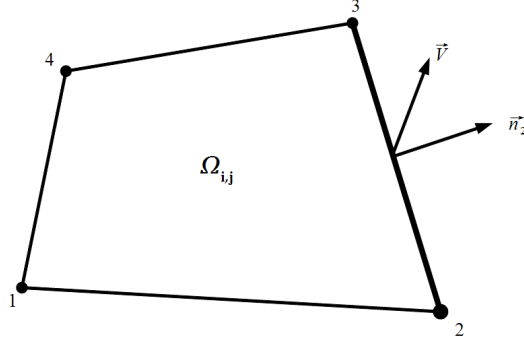
Figure 3: Definition of the parameter $\delta = \hat{V} \cdot \hat{n}$

The parameter delta is defined as,

$$\delta = \hat{V} \cdot \hat{n}, \tag{47}$$

where $\hat{V}$ and $\hat{n}$ are unit vectors of the flow's velocity and the face's normal, respectively. The way we choose to discretized the velocity terms ϕ_x and ϕ_y is determined by the parameter δ. For example, when $\delta = 1$ the flow direction is normal to the face 2-3, and it results in a "wide" approximation in ξ-direction and "narrow" approximation in η-direction. The case of $\delta < 0$ indicates that the flow direction is opposite to the way that the grid's indexes are defined. It is a typical situation when solving the flow field through a cylinder, airfoil or a turbine blade. Further details and examples are obtained in Section 13.

The covariant velocity components through face $2 - 3$, V_ξ, and V_η are derived by central differencing for a flow in general direction as follows:

$$\vec{V_\xi} = \left(1 - \frac{1}{2}\delta^2\right)\frac{(\phi_{i,j} - \phi_{i-1,j})}{\triangle\xi_{i,j}} + \frac{1}{4}\delta^2\frac{(\phi_{i,j+1} - \phi_{i-1,j+1})}{\triangle\xi_{i,j+1}} + \frac{1}{4}\delta^2\frac{(\phi_{i,j-1} - \phi_{i-1,j-1})}{\triangle\xi_{i,j-1}},$$

$$\vec{V_\eta} = \frac{1}{4}\left(\frac{\phi_{i,j+1} - \phi_{i,j}}{\triangle\eta_{i,j+1}}\right) + \frac{1}{4}\left(\frac{\phi_{i-1,j} - \phi_{i-1,j-1}}{\triangle\eta_{i-1,j}}\right) \qquad if \qquad \hat{V}_y \geq 0, \tag{48}$$

$$\vec{V_\eta} = \frac{1}{4}\left(\frac{\phi_{i,-1j+1} - \phi_{i-1,j}}{\triangle\eta_{i-1,j+1}}\right) + \frac{1}{4}\left(\frac{\phi_{i,j} - \phi_{i,j-1}}{\triangle\eta_{i,j}}\right) \qquad if \qquad \hat{V}_y < 0,$$

where the velocity vector in Cartesian coordinate system is defined by $\vec{V} = V_x\hat{x} + V_y\hat{y}$. The terms $\hat{x}$ and $\hat{y}$ are unit vectors in x and y directions, respectively.

4. When the density ρ and the covariant velocity ϕ_ξ are available at the half node $(i - 1/2, j)$, the flux can be computed by

$$f_{i-1/2.j} = \rho\left(\phi_\xi, \phi_\eta\right)\phi_\xi. \tag{49}$$

5. Computation of the remaining fluxes at $(i + 1/2, j)$, $(i, j + 1/2)$, and $(i, j - 1/2)$ is done in a similar fashion, using the four-step process described above.

3.4. Compressible flow − supersonic flow $(M > 1)$

In the supersonic region the equation changes type from elliptic to hyperbolic, and therefore, a "wide" upwind difference should be used to approximate the derivatives in the streamwise direction. In the construction of the discrete approximation to the conservation form, it is necessary to switch to upwind differencing. Consider again the flux in the ξ-direction through the face $(i - 1/2, j)$ is approximated as follows:

1. A central discretization of the Laplacian holds also in this case.
2. Next, according to the relations (48) in the Cartesian components of velocity u and v, the contravariant component of velocity $V_{\xi n}$ and $V_{\eta n}$, the local speed of sound $a_{i,j}$, and the Mach number $M_{i,j}$, are computed at the half node $(i - 1/2, j)$.
3. The discretization of the velocities ϕ_x and ϕ_y in (27) is done in the streamwise direction while taking into account an upwind approximation as the flow exceeds $M = 1$. The final formula for ϕ_x and ϕ_y is written as follows:

$$\phi_x^2 = \frac{1}{M_{i,j}^2} V_\xi \cdot V_{\xi n} + \left(1 - \frac{1}{M_{i,j}^2}\right) V_\xi^u \cdot V_{\xi n}^u,$$

$$\phi_y^2 = \frac{1}{M_{i,j}^2} V_\eta \cdot V_{\eta n} + \left(1 - \frac{1}{M_{i,j}^2}\right) V_\eta^u \cdot V_{\eta n}^u. \tag{50}$$

The superscript u denotes an upwind approximation. Since the equation changes type from elliptic to hyperbolic, therefore, a "wide" upwind difference should be used. As an illustration, when the flow is grid aligned, the fluxes at the half node $(i - 1/2, j)$ are discretized, in a stencil notation, as follows:

$$V_\xi = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 \\ -\frac{1}{2} & \frac{1}{2} & i,j & 0 & 0 \\ -\frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 \end{bmatrix}, \qquad V_\eta = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & -\frac{1}{2} & 0 & 0 \end{bmatrix}. \tag{51}$$

As one can see, the upwind differencing is introduced smoothly since $\frac{1}{M^2} \to 1$ as $M \to 1$. If the upwind differencing were introduced directly as $M_\infty > 1$, there would be a sudden change in the representation of the Laplacian term $\nabla^2 \phi$, which does not vanish when $M_\infty = 1$. A scheme of this type was tested and found to be much less efficient.

Any numerical scheme used to solve the FPE for the potential ϕ must satisfy the domain of dependence. Since the characteristics are symmetric about the velocity vector, both the ξ and η derivative terms must be properly shifted when solving the FPE. The discretization of V_ξ and V_η is already presented in the previous subsection. However, the upwind approximations V_ξ^u and V_η^u in a general direction is done as follows: In case of $\delta \geq 0$,

$$V_\xi^u = \left(\frac{1}{2}\delta^2\right) \frac{\phi_{i-1,j} - \phi_{i-2,j}}{\triangle\xi_{i-1,j}} + \left(\frac{1}{4}\delta^2\right) \frac{\phi_{i-1,j-1} - \phi_{i-2,j-1}}{\triangle\xi_{i-1,j-1}},$$

$$+ \left(\frac{1}{4}\delta^2\right) \frac{\phi_{i-1,j+1} - \phi_{i-2,j+1}}{\triangle\xi_{i-1,j+1}} + \frac{1}{2}\left(1 - \delta^2\right) \frac{\phi_{i,j-1} - \phi_{i-1,j-1}}{\triangle\xi_{i,j-1}},$$

$$+ \frac{1}{2}\left(1 - \delta^2\right) \frac{\phi_{i,j} - \phi_{i-1,j}}{\triangle\xi_{i,j}}, \tag{52}$$

$$V_\eta^u = \frac{1}{2}\frac{\phi_{i-1,j} - \phi_{i-1,j-1}}{\triangle\eta_{i-1,j}} \qquad if \qquad \hat{V}_y \geq 0,$$

$$V_\eta^u = \frac{1}{2}\frac{\phi_{i-1,j+1} - \phi_{i-1,j}}{\triangle\eta_{i-1,j+1}} \qquad if \qquad \hat{V}_y < 0.$$

If $\delta < 0$, then

$$V_\xi^u = \left(\frac{1}{2}\delta^2\right) \frac{\phi_{i+1,j} - \phi_{i,j}}{\triangle\xi_{i+1,j}} + \left(\frac{1}{4}\delta^2\right) \frac{\phi_{i+1,j-1} - \phi_{i,j-1}}{\triangle\xi_{i+1,j-1}},$$

$$+ \left(\frac{1}{4}\delta^2\right) \frac{\phi_{i+1,j+1} - \phi_{i,j+1}}{\triangle\xi_{i+1,j+1}} + \frac{1}{2}\left(1 - \delta^2\right) \frac{\phi_{i,j+1} - \phi_{i-1,j+1}}{\triangle\xi_{i,j+1}},$$

$$+ \frac{1}{2}\left(1 - \delta^2\right) \frac{\phi_{i,j} - \phi_{i-1,j}}{\triangle\xi_{i,j}}, \tag{53}$$

$$V_\eta^u = \frac{1}{2}\frac{\phi_{i,j+1} - \phi_{i,j}}{\triangle\eta_{i,j+1}} \qquad if \qquad \hat{V}_y \geq 0,$$

$$V_\eta^u = \frac{1}{2}\frac{\phi_{i,j+1} - \phi_{i,j}}{\triangle\eta_{i,j+1}} \qquad if \qquad \hat{V}_y < 0.$$

In these approximations the velocity is assumed to be from left to right, namely $\hat{V}_x > 0$.

4. When ϕ_x^2 and ϕ_y^2 are calculated, the density ρ is available at the half node $(i - 1/2, j)$. Then, the flux (in ξ-direction) can be calculated by (49).

5. Computation of the remaining fluxes at $(i + 1/2, j)$, $(i, j + 1/2)$, and $(i, j - 1/2)$ is done in a similar fashion, using the four-step process described above.

4. Extending the 2D upwind scheme to 3D

The 2D upwind numerical scheme for the supersonic flow regime was extended to 3D discretization. This was done while applying the same approach as is applied in 2D. As is already stated, the equation changes type when the flow becomes supersonic. This switching changes the diffusive character of the elliptic flow field to the propagation dominated behavior associated with the hyperbolic equation. Therefore, the discretization in the supersonic region must be inside the domain of dependence, between the characteristic lines. A common approach to implement this change is by using the rotated difference scheme that introduced by Jameson [23]. In this part of the work, the difference scheme for the FPE in the quasi-linear form is designed to deal with problems characterized by flow in an arbitrary direction. The idea is to rearrange the equation as if it were locally expressed in a coordinate system aligned with the flow. The derivation of the three dimensional FPE in the rotated difference scheme is based on the results introduced by Jameson [23]. First the 3D FPE in the quasi-linear can be written as follows:

$$\left(a^2 - q^2\right)\phi_{ss} + a^2\left(\nabla^2\phi - \phi_{ss}\right) = 0, \tag{54}$$

where q is the flow speed determined by $q^2 = u^2 + v^2 + w^2$ and $\nabla^2\phi = \phi_{xx} + \phi_{yy} + \phi_{zz}$. The direction of the flow is calculated as $\frac{u}{q} = cos(\theta)$, $\frac{v}{q} = sin(\theta)$, $\frac{w}{q} = cos(\psi)$ where θ is the azimuthal angle in the $x - y$ plane from the x-axis and ψ is the polar angle, measured from the $x - y$ plane to the velocity vector. The streamswise second derivative can be expressed as

$$\phi_{ss} = \frac{1}{q^2}\left(u^2\phi_{xx} + v^2\phi_{yy} + w^2\phi_{zz} + 2uv\phi_{xy} + 2vw\phi_{yz} + 2uw\phi_{xz}\right). \tag{55}$$

When the flow is subsonic all the derivatives are approximated by central difference formulas. If the flow is supersonic $(q > a)$, then the terms appearing within ϕ_{ss} must be shifted in the direction of the flow while all contributions to the remaining terms are approximated by central difference formulas. The one sided difference operators biased in the upstream sense in all three coordinate directions. For simplification I will use only "narrow" approximation in the flow direction. The discretized operators inside ϕ_{ss} are as follows:

$$\begin{aligned}
\phi_{xx} &= \frac{(\phi_{i,j,k} - 2\phi_{i-1,j,k} + \phi_{i-2,j,k})}{\triangle x^2}, \\
\phi_{yy} &= \frac{\phi_{i,j,k} - 2\phi_{i,j-1,k} + \phi_{i,j-2,k}}{\triangle y^2}, \\
\phi_{zz} &= \frac{\phi_{i,j,k} - 2\phi_{i,j,k-1} + \phi_{i,j,k-2}}{\triangle z^2}, \\
\phi_{xy} &= \frac{\phi_{i,j,k} - \phi_{i-1,j,k} - \phi_{i,j-1,k} + \phi_{i-1,j-1,k}}{\triangle x\triangle y}, \\
\phi_{xz} &= \frac{\phi_{i,j,k} - \phi_{i-1,j,k} - \phi_{i,j,k-1} + \phi_{i-1,j,k-1}}{\triangle x\triangle z}, \\
\phi_{yz} &= \frac{\phi_{i,j,k} - \phi_{i,j-1,k} - \phi_{i,j,k-1} + \phi_{i,j-1,k-1}}{\triangle y\triangle z}.
\end{aligned} \tag{56}$$

All the remaining terms are approximated by central differencing:

$$\phi_{xx} = \frac{\phi_{i+1,j,k} - 2\phi_{i,j,k} + \phi_{i-1,j,k}}{\triangle x^2},$$

$$\phi_{yy} = \frac{\phi_{i,j+1,k} - 2\phi_{i,j,k} + \phi_{i,j-1,k}}{\triangle y^2},$$

$$\phi_{zz} = \frac{\phi_{i,j,k+1} - 2\phi_{i,j,k} + \phi_{i,j,k-1}}{\triangle z^2},$$

$$\phi_{xy} = \frac{\phi_{i+1,j+1,k} - \phi_{i-1,j+1,k} - \phi_{i+1,j-1,k} + \phi_{i-1,j-1,k}}{4\triangle x \triangle y}, \quad (57)$$

$$\phi_{xz} = \frac{\phi_{i+1,j,k+1} - \phi_{i-1,j,k+1} - \phi_{i+1,j,k-1} + \phi_{i-1,j,k-1}}{4\triangle x \triangle z},$$

$$\phi_{yz} = \frac{\phi_{i,j+1,k+1} - \phi_{i,j-1,k+1} - \phi_{i,j+1,k-1} + \phi_{i,j-1,k-1}}{4\triangle y \triangle z}.$$

Since we restrict ourselves in this work to a usage of a pointwise relaxation, while applying Gauss–Seidel and using Von-Neumann stability analysis, we get, as expected, an unstable scheme. Therefore the product operator $L\tilde{L}$ must be applied in order to achieve stability. From several numerical experiments we see that the difference operator resulting in the matrix $\bar{A}$ (see section), for the cases of $0^o \leq \theta \leq 45^o$ was chosen to be the same as was applied in the 2D case but written in 3D form. Compared to the 2D case, there are much more possibilities to construct the operator $\tilde{L}$ that results in a stable operator $L\tilde{L}$. For simplicity the operator $\tilde{L}$ is discretized on the $x - y$ plane with no derivation in z-direction.

$$\tilde{L} = \left(\frac{1}{4}sin^2(\theta) + \frac{1}{4}sin(\theta)cos(\theta)\right)\phi_{i-1,j+1,k}$$

$$+ \left(\frac{1}{4} - \frac{1}{2}sin(\theta)cos(\theta)\right)\phi_{i+1,j+1,k}$$

$$+ \frac{1}{2}sin^2(\theta)\phi_{i,j+1,k} \quad (58)$$

$$+ \left(\frac{1}{2} - \frac{1}{2}sin^2(\theta)\right)\phi_{i+1,j,k}$$

$$+ \left(\frac{1}{4}cos^2(\theta) + \frac{1}{4}sin(\theta)cos(\theta)\right)\phi_{i+1,j-1,k} - \phi_{i,j,k}.$$

The numerical stability of the product operator while applying the damped Gauss-Seidel relaxation, with an under relaxation parameter of $\omega = 0.8$, was tested by Von-Neumann analysis. The stability condition requires that the modulus of the amplification factor should be lower or equal to one. It is interesting to note that in the supersonic region, no additional terms were needed to the operator $\tilde{L}$ written in 2D form, for the stability of the scheme, in all flow directions. Three views of the amplification factor as a surface over the regions $[-\pi, \pi] \times [-\pi, \pi] \times [0]$, $[-\pi, \pi] \times [-\pi, \pi] \times [-\pi]$ and $[-\pi, \pi] \times [-\pi, \pi] \times [\pi]$, are given in Figures 8-7 for different flow conditions. The computations were done by using the Maple mathematical commercial software (Maple version 11). If the operator is not stable, it is likely a result of the slower modes ($\theta_1 \sim 0$, $\theta_2 \sim 0$, $\theta_3 \sim 0$). Although, the operator was checked for $-\pi \leq z \leq \pi$, for presentation three cases of $z = [-\pi, 0, \pi]$ are enough to present stability. Each surface corresponds to the variation of $|G(\theta_1, \theta_2, \theta_3)|$ over $-\pi \leq \theta_1 \leq \pi$ and $-\pi \leq \theta_2 \leq \pi$ for a fixed value of θ_3.
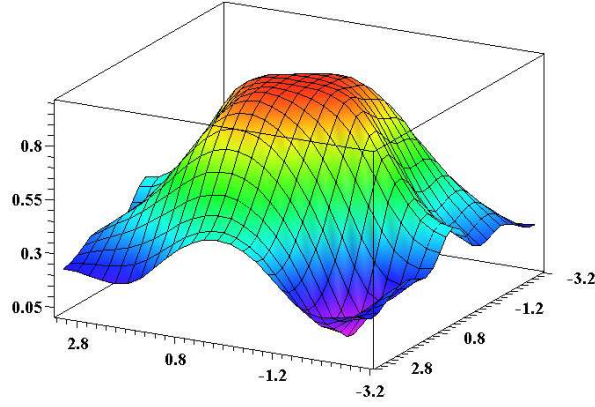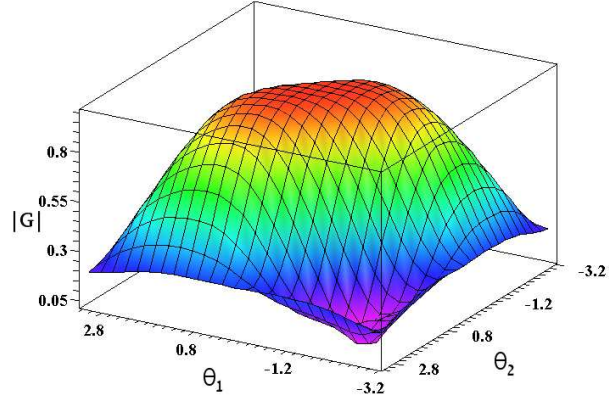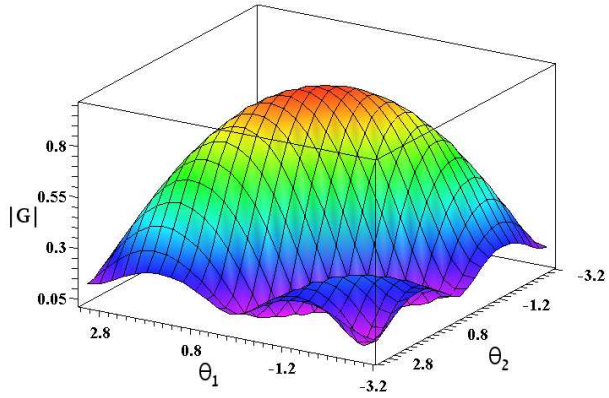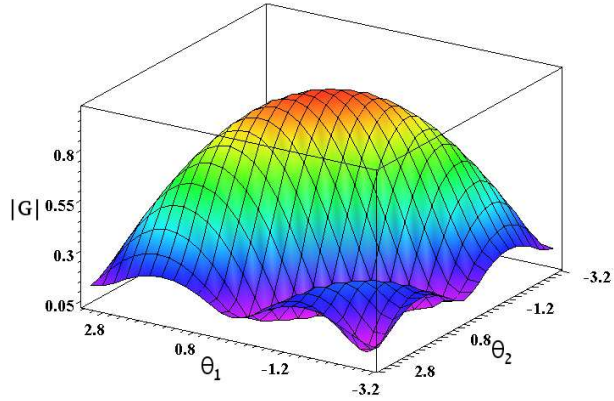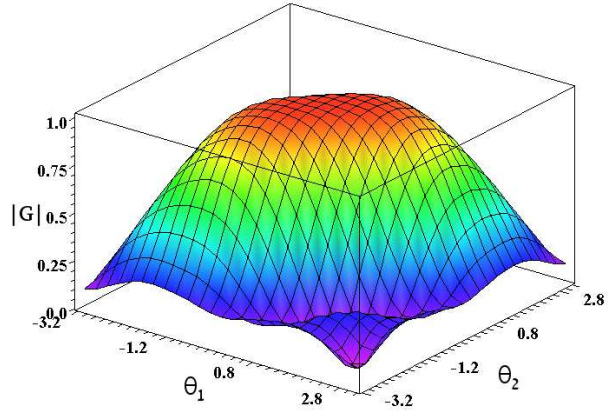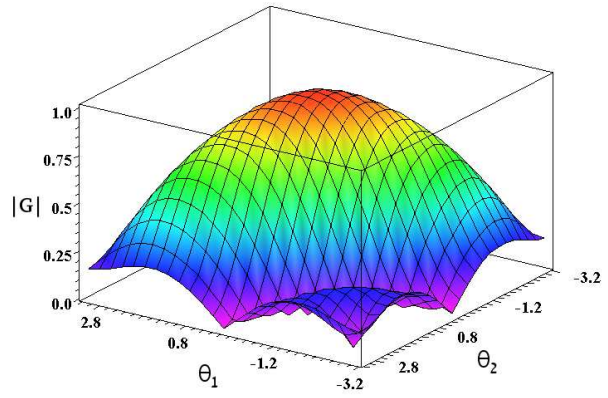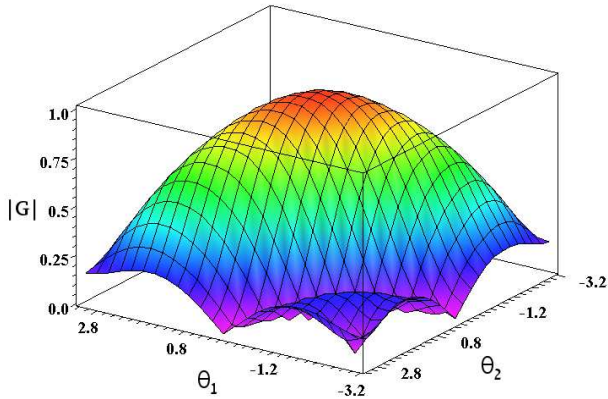
(a)
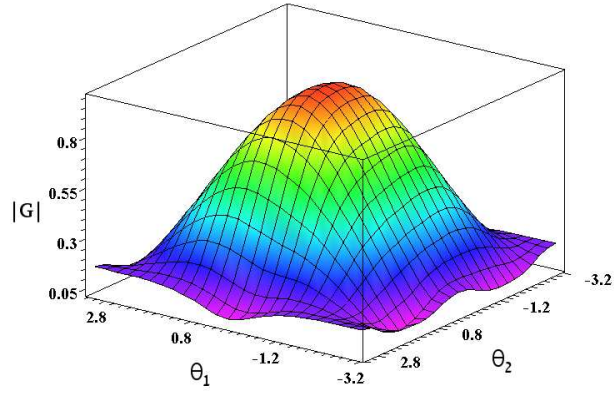
(b)
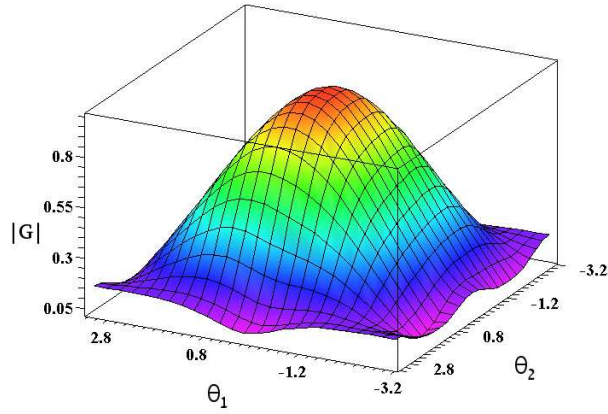
(c)

Figure 4: Amplification factor, $|G(\theta_1, \theta_2, \theta_3)|$, for the Gauss–Seidel method applied to the model problem in three dimensions, shown as a surface over the regions: a.) $[-\pi, \pi] \times [-\pi, \pi] \times [0]$, b.) $[-\pi, \pi] \times [-\pi, \pi] \times [-\pi]$, c.) $[-\pi, \pi] \times [-\pi, \pi] \times [\pi]$. The flow conditions are: $M_\infty = 1.1$, $\theta = 0°$, $\psi = 0°$.
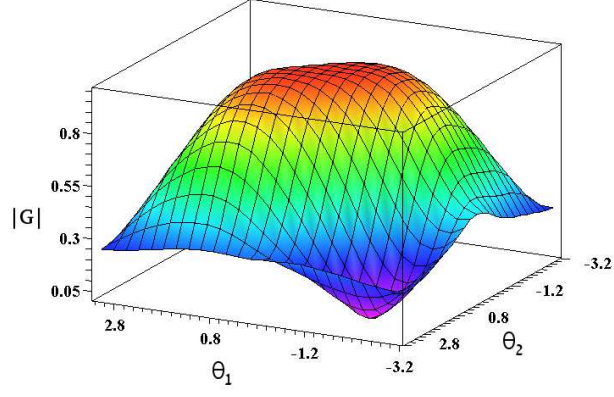
(a)

(b)

(c)

Figure 5: Amplification factor, $|G(\theta_1, \theta_2, \theta_3)|$, for the Gauss-Seidel method applied to the model problem in three dimensions, shown as a surface over the regions: a.) $[-\pi, \pi] \times [-\pi, \pi] \times [0]$, b.) $[-\pi, \pi] \times [-\pi, \pi] \times [-\pi]$, c.) $[-\pi, \pi] \times [-\pi, \pi] \times [\pi]$. The flow conditions are: $M_\infty = 1.1$, $\theta = 45°$, $\psi = 45°$.

(a)

(b)

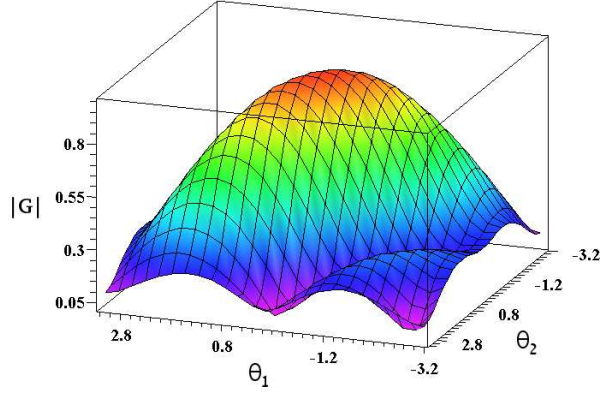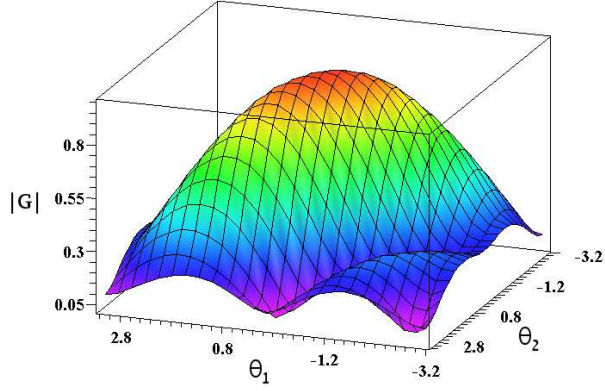

(c)

Figure 6: Amplification factor, $|G(\theta_1, \theta_2, \theta_3)|$, for the Gauss-Seidel method applied to the model problem in three dimensions, shown as a surface over the regions: a.) $[-\pi, \pi] \times [-\pi, \pi] \times [0]$, b.) $[-\pi, \pi] \times [-\pi, \pi] \times [-\pi]$, c.) $[-\pi, \pi] \times [-\pi, \pi] \times [\pi]$. The flow conditions are: $M_\infty = 1.1$, $\theta = 45°$, $\psi = 22.5°$.

(a)

(b)

(c)

Figure 7: Amplification factor, $|G(\theta_1, \theta_2, \theta_3)|$, for the Gauss-Seidel method applied to the model problem in three dimensions, shown as a surface over the regions: a.) $[-\pi, \pi] \times [-\pi, \pi] \times [0]$, b.) $[-\pi, \pi] \times [-\pi, \pi] \times [-\pi]$, c.) $[-\pi, \pi] \times [-\pi, \pi] \times [\pi]$. The flow conditions are: $M_\infty = 1.1$, $\theta = 30^0$, $\psi = 60°$.

Some analysis and experimentation reveals that a better smoothing factor is obtained when the the operator $\widetilde{L}$ includes points in the z-direction. As an example, the discretization of $\widetilde{L}$ in the streamwise direction is done as follows:

$$
\begin{aligned}
\widetilde{L} &= \phi_{i+1,j,k} + \phi_{i+1,j+1,k} + \phi_{i+1,j-1,k} \\
&+ \phi_{i+1,j,k+1} + \phi_{i+1,j,k-1} - \phi_{i,j,k}.
\end{aligned}
\tag{59}
$$

The amplification factor as a surface over the regions $[-\pi, \pi] \times [-\pi, \pi] \times [0]$, $[-\pi, \pi] \times [-\pi, \pi] \times [-\pi]$ and $[-\pi, \pi] \times [-\pi, \pi] \times [\pi]$, is given in Figures 8 for $M_\infty = 1.2$ and $(\theta = 0°, \psi = 0°)$.
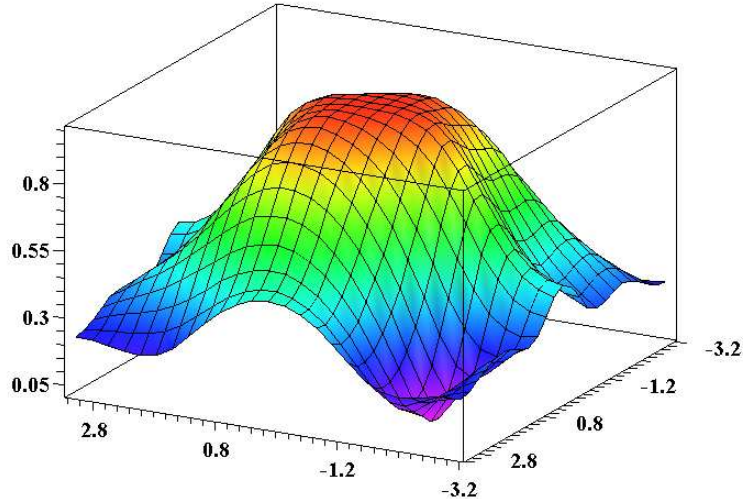
Figure 8: Amplification factor, $|G(\theta_1, \theta_2, \theta_3)|$, for the Gauss-Seidel method applied to the model problem in three dimensions, shown as a surface over the regions: a.) $[-\pi, \pi] \times [-\pi, \pi] \times [0]$, b.) $[-\pi, \pi] \times [-\pi, \pi] \times [-\pi]$, c.) $[-\pi, \pi] \times [-\pi, \pi] \times [\pi]$. The flow conditions are: $M_\infty = 1.1$, $\theta = 30°$, $\psi = 60°$.

It is evident that the scheme is numerically stable whenever the velocity coincides with one of the three coordinate directions. These examples indicate that the approach of solving the supersonic flow regime while applying a pointwise relaxation method is promising, both in the subsonic and supersonic flow regimes.

5. Conclusions

The FPE is obtained by assuming an inviscid, irrotational flow, and the Navier-Stokes equations are reduced down to a single equation. The FPE is useful for design and analysis of airfoil, wings, diffusers etc.. The computations are usually less resource-consuming than those solving the Euler or Navier-Stokes equations. The FPE can be used for transonic flows, where a lot of design issues are of interest. Transonic flow problem is a rather complex one from the computational point of view. One of the main difficulties is the fact that the differential operator changes its type between elliptic for subsonic flow regime and hyperbolic (with respect to the flow direction) in the supersonic flow regime. Another (sub-)difficulty is that the subsonic flow regime itself presents two extremities (and all the possible cases in between): nearly isotropic operator for the flow speed case and highly anisotropic operator for a nearly sonic flow speed. While the simple damped Jacobi and Gauss–Seidel relaxation schemes are suitable for the subsonic case, both of them are unstable in the supersonic case. Resolving this difficulty is the main achievement of this work.

A stable pointwise direction independent relaxation was developed for the supersonic and subsonic flow regimes. This stable relaxation is obtained by post-multiplying the original operator by a certain simple first order downwind operator. This new operator is designed in such a way that makes the pointwise relaxation applied to the product operator to become stable (and constitutes a good smoother − in the algebraic sense). First, the discretization is presented for the FPE in the quasi-linear form while applying the finite difference approach. Second, the procedure for constructing a stable finite volume approximation to the conservation form of the FPE under body-fitted structured grid approach is presented. The strategy of discertizing the FPE in the conservation form is based on the rotated difference approach introduced by Jameson [12] and was used in the quasi-linear form.

Finally, it was demonstrated that the 2D upwind numerical scheme for the supersonic flow regime can be extended to 3D case, while applying the same operator $\tilde{L}$ that was used in the 2D case. Since the discretization process results in a discrete operator (matrix A) with a "stronger"

20

diagonal, this in turn contributes to the stabilization of the 3D discrete operator. This approach was verified by numerical stability analysis in various flow directions.

References

[1] A. Brandt, S. F. McCormick, and J. W. Ruge. Algebraic Multigrid (AMG) for sparse matrix equations. In *Sparsity and Its Applications*. Cambridge University Press, Cambridge, 1984.

[2] A. Brandt. Algebraic multigrid theory: The symetric case. *Appl. Math. Comput.*, 19:23–56, 1986.

[3] K. Stueben. An introduction to Algebraic Multigrid. Technical report, German National Research Center for Information Technology (GMD), San Diego, 2001.

[4] J. W. Ruge and K. Stueben. Algebraic Multigrid. In *Multigrid Methods*, volume 3, pages 73–130. S. F. McCormick, ed., SIAM, Philadelphia, 1987.

[5] C. Hirsch. *Numerical Computation of Internal and External Flows*, volume 1. John Wiley & Sons, 1988.

[6] E.M. Murman and J.D. Cole. Calculation of plane steady transonic flows. *AIAA*, 9(1):114–121, Jan 1971.

[7] E. M. Murman. Analysis of Embedded Shock Waves Calculated by Relaxation Methods. In *Proceeding First AIAA Computational Fluid Dynamics Conference*, Palm Springs, July 1973.

[8] J. L. Steger and H. Lomax. Transonic Flow About Two-Dimensional Airfoils by Relaxations Procedures. *AIAA*, 10:49–54, 1972.

[9] P. R. Garabedian and D. Korn. Analysis of Transonic Airfoils. *Communications on Pure and Applied Mathematics*, 24:841–851, 1971.

[10] W. F. Ballhaus and F. R. Bailey. Numerical Calculation of Transonic Flow About Swept Wings. *AIAA*, 72:677, 1972.

[11] F. R. Bailey and J. L. Steger. Relaxation Techniques for Three-Dimensional Transonic Flow About Wings. *AIAA*, 72:189, 1972.

[12] A. Jameson. Iterative Solution of Transonic Flows Over Airfoils and Wings, Including Flows at Mach 1. *Pure. Appl. Math*, 27:283–309, 1974.

[13] A. Jameson, W. Schmidt, and E. Turkel. Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping Schemes. In *AIAA 14th Fluid and Plasma Dynamics*, number 81-1259, Reston, VA 1981. AIAA.

[14] A. Jameson. Transonic Airfoil Calculations Using the Euler Equations. In P. L. Roe, editor, *Numerical Methods in Aeronautical Fluid Dynamics*. Academic Press, New York, 1982, New York, 1982.

[15] A. Jameson. Solution of the Euler Equations for Two-Dimensional Transonic Flow by a Multigrid Method. *Applied Mathematics and Computation*, 13:327–355, 1983.

[16] S. Shitrit and D. Sidilkover. A Full Potential Equation Solver Based on the Algebraic Multigrid Method: Elementary Applications. *to be published at SIAM Journal of Scientific Computing*, -:–, 2010.

[17] D. Sidilkover S. Shitrit and A. Gelfgat. An Algebraic Multigrid Solver For Transonic Flow Problems. *to be published at JCP Journal of Scientific Computing*, -:–, 2010.

[18] D. Sidilkover. A factorizable scheme for the equations of fluid flow. *Appl. Numer. Math.*, 41:423–426, 2002.

[19] S. Shitrit and D. Sidilkover. Toward applying algebraic multigrid to transonic flow problems. *SIAM Journal of Scientific Computing*, 32:2007–2028, July 2010.

[20] U. Trottenberg, A. Schuller, and C. W. Oosterlee. *Multigrid*. Elsevier Academic Press, 2001.

[21] S. Kaczmarz. Angenaherte Auflosung von Systemen linearer Gleichungen. *Bulle. Acad. Pol. Sci. Lett. A*, 15:355–357, 1937.

[22] A. Brandt. Stages in developing multigrid solutions. In *Proceedings of the 2nd International Congress on Numerical Methods for Engineers*, 1980.

[23] A. Jameson. Numerical Calculations of the Three-Dimensional Transonic Flow Over Yawed Wing. In *AIAA Computational Fluid Dynamics Conference*, pages 18–26, Palm Springs, July 1973.