JOURNAL OF COMPUTATIONAL PHYSICS 110, 212-220 (1994)

The Alternating Direction Multi-zone Implicit Method

MOSHE ROSENFELD

Faculty of Engineering, Tel Aviv University, Tel Aviv, Israel

AND

YUVAL YASSOUR

Faculty of Aerospace Engineering, Technion, Haifa, Israel

Received August 17, 1992

Within the structured grid approach, numerical solution of partial differential equations (PDE) in complex regions requires the decomposition of the domain into several zones. Implicit solution methods of the discrete equations are preferred because of their superior numerical properties. Indeed, many existing multi-zone solution methods use implicit techniques, but the zonal boundaries are updated explicitly. The zonal boundaries are created between the zones in the process of domain decomposition, but otherwise they are regular interior field points. If only steady state solutions are sought, the explicit calculation of the zonal boundaries usually affects only the convergence rate but not the accuracy of the solution. However, in time-dependent cases this may degrade the time accuracy of the solution. In the present work, we propose a novel fully implicit method for solving sets of PDE using multi-zones and structured grids. The method combines the zonal approach with the alternating direction implicit (ADI) method, and hence the method is referred to as the alternating direction multi-zone implicit (ADMZI) method. The key idea is the generation of different sets of zones for each stage (factor) of the ADI method, rather than using the same set of zones for all the stages. Consequently, the ADI sweeps are performed between physical boundaries, so that zonal boundaries are avoided and the solution is fully implicit. The ADMZI method can be applied to any set of PDE that employs an ADI (or approximate factorization) method. Typical examples include the time-dependent compressible or incompressible Navier–Stokes equations. Several conceptual examples and actual numerical test cases (that solve the heat conduction equation) confirm the versatility, efficiency, and accuracy of the ADMZI method.

1. INTRODUCTION

Several reasons may require the decomposition of a computational domain into sub-domains (zones). Geometric complexity is probably the most common reason. In such complex situations it is impossible to generate a single zone grid with adequate control on the distribution of the mesh points using structured grids, see, for example, [1–5]. Three main types of domain decomposition techniques are in use: patched zones, overlapped zones, and overlaid zones. Patched zones have a common boundary line. The mesh lines across the boundaries may be continuous or discontinuous [1, 2]. In the second technique, an overlap region exists between the zones [3, 4]. The extent of that region may be from one up to several mesh points. In the third technique, which is also known as the Chimera method [5], smaller zones are defined on top of a base grid (which may have "holes" inside it). Inter-zone data transfer is accomplished by interpolation.

Defining the zones is one issue, while solving the partial differential equations (PDE) over a multi-zone region is yet another task. As long as explicit point-wise solution methods are used, the splitting of the solution domain into zones poses no difficulties. However, explicit solution methods may have severe stability restrictions, especially for stiff problems such as high Reynolds number flows. Implicit solution schemes are recommended in these cases. Indeed, in the majority of viscous flow calculations implicit schemes are used within each zone. The main difficulty in applying implicit schemes to structured grid multi-zone methods lies in the treatment of the boundaries separating the zones. These interior boundaries, referred to as the *zonal boundaries*, are a result of the decomposition of the solution domain into zones. Yet, the zonal boundary points are regular interior points where the solution is unknown and the PDE should be satisfied, as in any other point of the domain.

In previous works, the zonal boundary conditions were treated explicitly. Most methods are variants of the Schwarz procedure [6]. Typically, the boundary conditions at the zonal boundaries are first guessed and the PDE are solved implicitly within the zone. In the next iterations, the zonal boundary conditions are updated, based on the solution obtained from the neighboring zones. Difficulties may be encountered in the conservation of fluxes, unless special flux-conserving approximations are employed between the zones [2]. This problem might be especially severe in the calculation of incompressible flows, where the fictitious

0021-9991/94 $6.00

mass sources generated at the zonal boundaries have global effects not only on the convergence rate, but also on the accuracy of the solution, [7].

Although these *partially* implicit techniques result in faster convergence to steady state than fully explicit methods, the efficiency and stability of fully implicit schemes cannot be attained using the Schwarz approach. We shall refer to these techniques as the *standard* multi-zone approach.

In the case of time-dependent calculations, the explicit treatment of the zonal boundary conditions may have detrimental effects on the accuracy of the solution as well. The temporal accuracy may degrade, unless expensive sub-iterations are performed at each time-step. The sub-iterations are needed to update the explicit zonal boundary conditions, but significantly increase the computational resources. Time-dependent calculations are anyway computationally intensive; the additional computations required for the multi-zonal simulations might be prohibitive.

Rogers [8] used sub-iterations in a multi-zone calculation of an airfoil with an open flap, using the time-dependent incompressible Navier–Stokes equations. The solution is advanced in time using the artificial compressibility method with sub-iterations at each time step, resulting in an implicit treatment of the zonal boundaries. The convergence rate of the solution to the steady state has been significantly accelerated, demonstrating the possible gains that can be achieved using implicit zonal boundary conditions. A similar approach has been previously used by Rai [2] for compressible flows.

The alternating direction implicit (ADI) method, also known as the approximate factorization method, is a very common method of solving PDE equations, especially in the context of the Navier–Stokes equations, both for compressible and incompressible flows [9–12]. These methods, with all their variants, solve multi-dimensional differential equations by decoupling the coordinate directions. The original PDE is approximated in a factored form, such that each factor requires the simultaneous solution of the unknowns along a single coordinate line (resulting in a coefficient matrix with a narrow band). The implicit solution of the unknowns along a coordinate line will be referred to as a *sweep*. For example, in solving a Cartesian two-dimensional problem, the sweeps along all the x-coordinate lines are first performed (i.e., the algebraic equations along the x-coordinate lines are solved simultaneously), followed by the sweeps along the y-coordinate lines. The ADI methods were originally developed for rectangular-like regions. Yet, this is not a requirement; Mitchell and Griffiths [13] give examples of the extension of the ADI method to non-quadrilateral regions.

The present study suggests a novel method for overcoming some of the shortcomings of existing multi-zone methods, the alternating direction multi-zone implicit (ADMZI) method. The method combines a zonal approach for decomposing complex domains and the ADI solution method into a single efficient, stable, and accurate method for solving *implicitly* PDE in complex multi-zone domains.

2. LAYOUT OF THE ADMZI METHOD

The idea of the ADMZI method is the use of a different set of zones for each stage of the ADI method. Thus, in two-dimensional cases two sets of different zones are used, one set for each sweep direction. Each zone is chosen to reside between two physical boundaries of the problem, i.e., between two boundaries where boundary conditions are specified. The edges of the sweep lines of each ADI stage reside on these physical boundaries. This eliminates the need for specifying explicit zonal boundary conditions, with the accompanying numerical difficulties. The other two boundaries of the quadrilateral zones (the boundaries that are parallel to the sweep direction) may be physical boundaries, regular interior points (zonal boundaries), or a combination of the two.

Three examples of decomposing domains into sets of zones appropriate for the implementation of the ADMZI method will be elaborated. The examples are mainly related to fluid dynamics problem, although the same methodology can be applied to many other PDE. The first two examples consider simple cases to establish the principles of the ADMZI method and the terminology to be used. The third example exploits better the full capabilities of the ADMZI method. Numerical results and more complicated examples will be given in the Results section.

2.1. Example 1: H-Grid

The first example considers an H-type grid over an airfoil, as shown in Fig. 1a. This case can be solved implicitly using standard ADI schemes with a few modifications to account for the presence of the airfoil. In the ADMZI method, this problem is solved by splitting the H-grid into two sets of zones, as shown in Figs. 1b and c. The physical boundaries of the problem are indicated by the thick lines; the thin lines are mesh lines, while the medium-thick lines show the zonal boundaries. Figure 1b shows the set of zones used for performing the ξ-sweeps (i.e., sweeps with constant η in the present terminology). Two zones are generated, one that includes the domain above the airfoil, while the other zone contains the domain below the airfoil. The ξ-sweeps are performed separately in each zone.

The different set of four zones, used to execute the η-sweeps, are shown in Fig. 1c. Each zone of the ADMZI method is a rectangular zone in the computational domain (although of a different number of mesh points), and the sweep lines span two physical boundaries, where boundary conditions are specified. It should be noted that in the

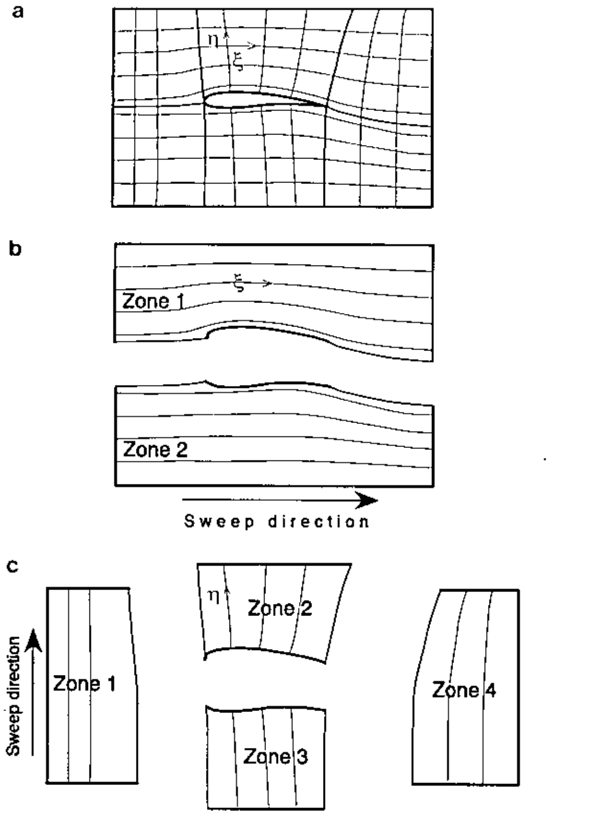

FIG. 1. Domain decomposition of an H-grid over an airfoil: (a) geometry; (b) ξ-set of zones; (c) η-set of zones.

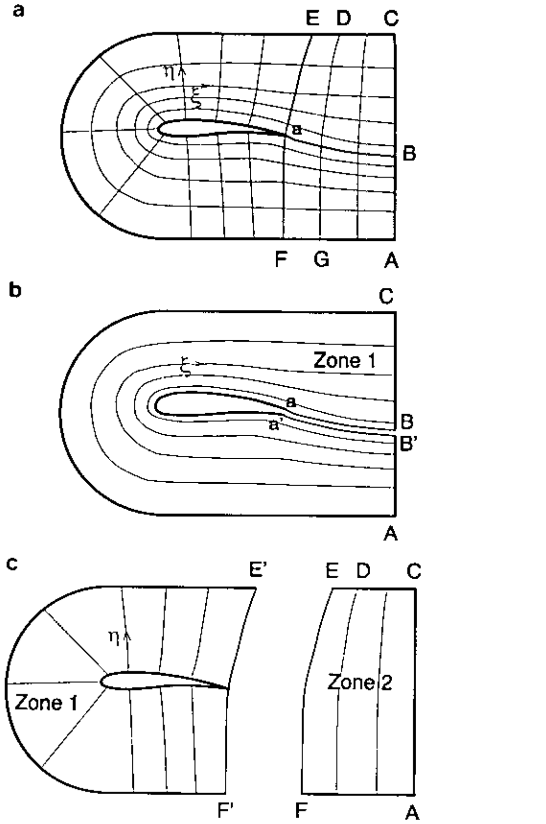

FIG. 2. Domain decomposition of a C-grid over an airfoil: (a) geometry; (b) ξ-set of zones; (c) η-set of zones.

ADMZI approach different sets of zones are used in each sweep direction, unlike in regular multi-zone methods where one set of zones is used. The decomposition of the zones is dictated by the factorization used by the ADI solution method.

2.2. Example 2: C-Grid

The next example considers the case of a C-grid over the same airfoil, see Fig. 2a. This case can be solved by other multi-zone approaches, although most of them treat explicitly the branch line cut (aB).

The C (ξ)-sweep is simple to perform in all the methods, see Fig. 2b. The sweep lines span the boundaries AB and $B'C$. The different approach of the ADMZI method is manifested in the η-sweeps. In standard multi-zone solution methods, the η-sweeps in the wake region are completed in two parts. First, the sweeps from the branch line cut to the upper outer boundary are completed, followed by the sweeps from the branch line cut to the lower outer boundary (or vice versa). Thus, the branch line cut is treated as an explicit zonal boundary. Yet, lines such as GD (Fig. 2a) are regular interior lines in the physical space. They are "special" only because of the data structure of the C-grid topology (each line in the wake region is stored in two non-contiguous segments).

In the ADMZI method, two η-zones are defined, Fig. 2c. Zone 1 is similar to other solvers. The benefits of using the ADMZI method are clearly demonstrated in zone 2. The ADI sweeps are done over entire lines (such as line GD), although each line is composed of two different sections of the data set. To perform the η-sweeps of zone 2, the two separate sections of the zone ($aBCE$ and $a'B'AF$) are copied from the storage arrays into a temporary work area to form a contiguous region and to permit an implicit sweep. Stanaway *et al.* [14] used recently a similar approach and solved implicitly for the entire η-lines in the wake region of airfoils with blunt trailing edges.

In both the ξ- and η-sweeps, all the zones are quadrilateral and the sweeps are done between two physical boundaries. It should be noted that the boundary lines parallel to the sweep lines in each region, such as lines $F'E'$ and FE, are treated exactly the same way as the interior sweep lines, i.e., the equations are solved implicitly.

2.3. Example 3: Branched Channel

The last example in this section considers the domain given in Fig. 3a by the thick lines. This case may describe the

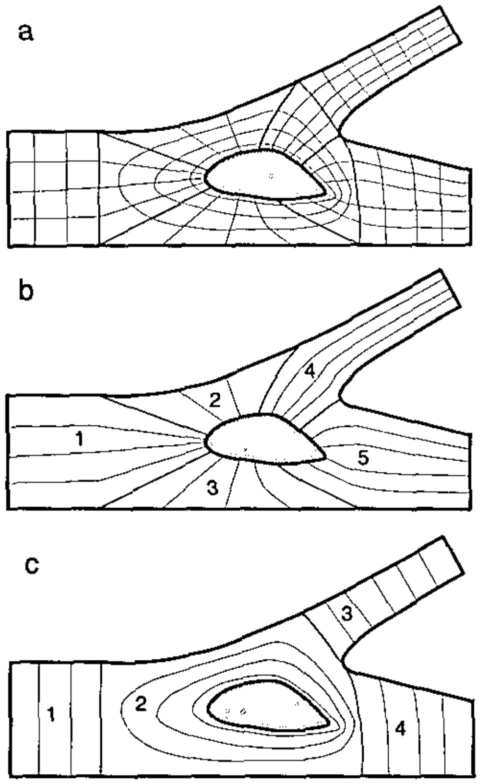

FIG. 3. Domain decomposition of the branched channel: (a) geometry; (b) first set of zones; (c) second set of zones.

flow in a branched channel with a body upstream of the branch. Figure 3a also shows a possible grid system; an O-type mesh is created around the body, while in the upstream and in the two downstream channel branches, H-type grids are generated. The radial lines that emerge from the body segments blend smoothly with the coordinate lines of the corresponding channel parts.

The two sets of zones required for solving the problem using the ADMZI method are shown in Figs. 3b and c. The numbers refer to the zone number of each set. Five zones are generated for executing the "radial" sweeps, Fig. 3b. The mesh lines shown are the sweep lines along that direction. All the lines emerge from the inner body and end at different sections of the channel. Four zones are required for performing the second stage of the ADI method, see Fig. 3c. In zone 2, O-sweeps are performed, while in the other three channel segments the sweeps span the two relevant walls of the channel.

It should be noted that the sweep lines of both ADI stages are done between physical boundaries, where boundary conditions are specified (in the case of the O-sweep, periodic boundary conditions are used). Consequently, no zonal boundary conditions should be specified and the solution method is fully implicit. The separate domain decomposition in each direction permits the implicit solution of the discrete equations even for complex regions.

2.4. Summary of the ADMZI Method

The essence of the ADMZI method is the proper decomposition of the computation domain into sets of zones that conform with the stages of the ADI method (or any other approximate factorization method). It means that all the edges of the coordinate lines along which the discrete equations are solved simultaneously should reside on physical boundaries where boundary conditions are specified. This approach eliminates the need of using zonal boundary conditions, which necessarily impose explicitness into the scheme. The satisfaction of this requirement is straightforward; most available grid generators fulfill this condition anyway.

Once the zones of the ADI stages are defined, the mesh can be completed. The simplicity of the ADMZI method originates from the definition of the zones. All the zones are single-connected quadrilaterals with all the boundary conditions at the edges of the zones, so that special treatment of interior points is avoided. The steps necessary to set up and solve the equations implicitly are identical in each set of zones.

The steps required to execute a stage of the ADMZI method (i.e., implicit solution of the discrete equations of a single ADI stage) are summarized below. In two-dimensional cases, two stages are performed, while in three-dimensional cases, three such stages are done to advance the solution one time-step:

(a) Decompose the domain for performing one stage of the ADI method. Both edges of the sweep lines (along which the equations are solved simultaneously) should reside on physical boundaries.

(b) Copy the relevant data from the global storage to a contiguous single zone. This step may require to assemble the zonal data from different sections of the global storage space.

(c) Solve the equations along all the lines of the ADI stage in the zone.

(d) Copy the solution of the zone back to the global storage space.

(e) Repeat these steps for all the zones of the set.

3. RESULTS

The present section considers three numerical test cases, as well as an additional example that is intended to demonstrate conceptually the capabilities of the ADMZI method to handle complex geometry.

3.1. Numerical Test Problems

The proposed ADMZI method is a general method for solving partial differential equations using the ADI (or approximate factorization) method in multi-zone regions. The particular form of the PDE is unimportant for studying the characteristics of the ADMZI method that is essentially an external driver. In this paper, the two-dimensional heat conduction equation is chosen to demonstrate the properties of the ADMZI method. The application of the method to more complex PDE, such as the Navier–Stokes equations, follows a very similar path. Moreover, whether the coordinate system in each region is curvilinear or not is irrelevant, since the ADMZI method is a technique for handling multi-zones. The type of the coordinate system (as well as the form of the PDE) is relevant only in the set-up and solution of the discrete equations. These operations are local to each zone and do not interfere directly with the ADMZI method.

Consequently, the numerical results given in this section consider the heat conduction equation in a Cartesian coordinates system. Yet, the results apply equally well to more complex cases, as far as only the principles of the ADMZI technique are concerned.

3.1.1. *Formulation*

The time-dependent two-dimensional heat conduction equation in a Cartesian coordinate system $\{x, y\}$ is used to study the ADMZI method,

$$\frac{\partial T}{\partial t} = \kappa \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right), \tag{1}$$

where T is the dependent variable, such as temperature, κ is the conductivity, and t is the time.

The PDE were discretized using the Crank–Nicolson scheme. The solution is advanced in time using the Douglas–Gunn ADI scheme in "delta" form. The scheme is unconditionally stable and the solution is spatially and temporarily second-order accurate. Three test cases, out of numerous cases that have been solved, are presented to demonstrate the complex problems the ADMZI method can handle.

3.1.2. *Multi-body*

The first test case considers the solution of the heat conduction equation in a rectangular region with four isothermal bodies at a temperature of $T = 100°$ inside it, as shown in Fig. 4a. The outer closed boundary is held at a constant temperature of $T = 0°$. The zonal decompositions in the x- and y-directions are shown in Figs. 4b and c (the arrows are parallel to the sweep lines). Note that here, as well as in the remaining figures, mesh lines are not drawn. Only physical boundaries (thick lines) and the zonal boundaries (thin lines) are shown.

Although a relatively large number of zones are used (11 in each direction), all the zones have similar structure and properties: they are single-connected rectangular regions with all the boundaries (whether physical or zonal) at the edges of the individual zones. The sweep lines are bounded by two walls where the temperature is specified and consequently the equations can be solved simultaneously.

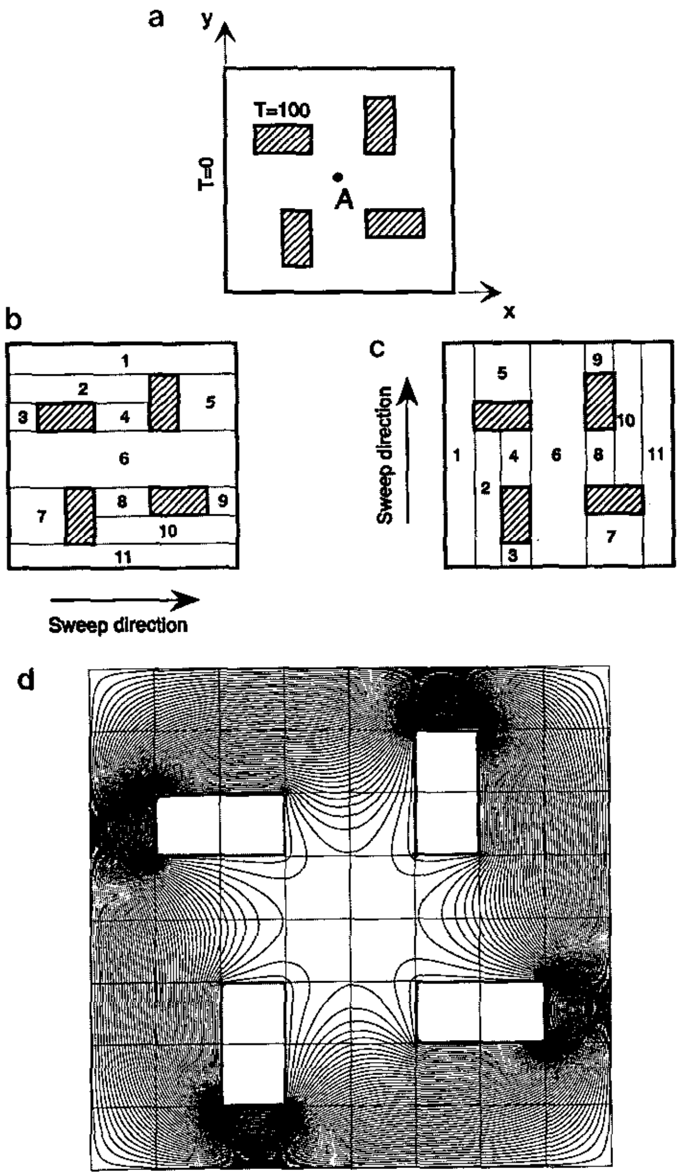

FIG. 4. The multi-body test case: (a) geometry; (b) first set of zones; (c) second set of zones; (d) solution.

A uniform grid is generated with 193×193 mesh points along the x- and y-directions, respectively. An arbitrary initial condition was specified and the solution was advanced in time until a steady solution was reached. The solution of the temperature field is given in Fig. 4d (the increment between the contour lines is 2°). Very smooth lines are obtained, even across the zonal boundaries. This is, of course, no surprise, bearing in mind that zonal boundaries do not exist in the ADMZI method.

For comparison reasons, the same problem was also solved by an overlap multi-zonal approach. Eleven rectangular zones, similar to the ADMZI method zones along one sweep direction (see Fig. 4b) have been used. An overlapping of one mesh point is used to update the zonal boundary conditions. Each zone was advanced in time by the ADI method (i.e., in each zone, the two ADI stages were performed sequentially, before moving on to the next zone), using zonal boundary conditions lagged in time. The converged (steady state) solutions obtained from the explicit zonal method and the ADMZI method are identical.

The lagging of the zonal boundary conditions reduces the temporal accuracy of the solution from second- to first-order accuracy, as Fig. 5 demonstrates. This figure shows the dependence of the solution on the time-step at the center point of the domain (point A in Fig. 4a). The solution is given for $t = 0.32$, when a steady state has not been yet attained. The vertical axis gives the difference between the solution of a particular time-step and the solution obtained at the finest time-step of $\Delta t_0 = 3.125 \times 10^{-4}$. Not only the formal accuracy of the explicit method is reduced, but also the magnitude of the error is significantly larger. The temporal accuracy of the explicit zonal boundary method can be restored and the convergence can be accelerated by performing sub-iterations at each time-step. However, this requires a considerably larger CPU time than the ADMZI method.

The convergence history at the center of the computational domain is shown in Fig. 1 both for the explicit multi-zone and the ADMZI solutions. The mesh consisted of 33×33 points and a time-step of $\Delta t = 0.02$ was used. The superior convergence properties of the ADMZI method are very obvious. After about 65 time-steps, the error in the ADMZI method is almost two orders of magnitudes less than the error obtained in the explicit method.

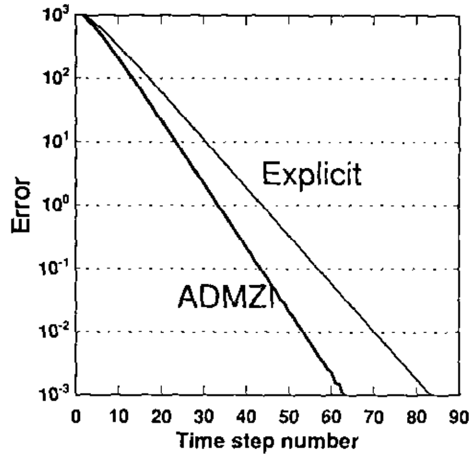

FIG. 6. Convergence history of the ADMZI and the explicit zonal boundary methods.

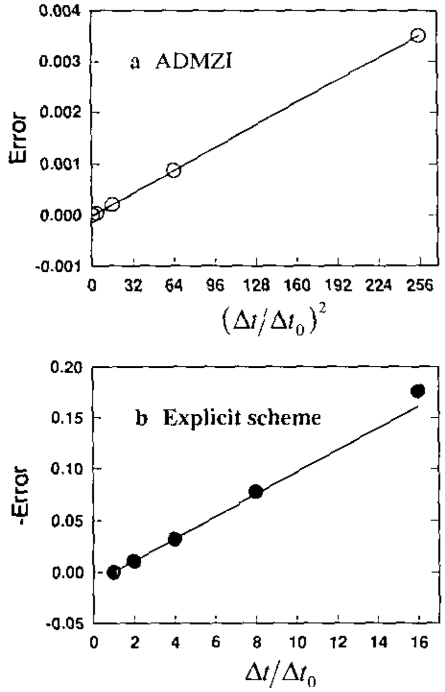

FIG. 5. Time-step refinement study of the (a) ADMZI method and (b) explicit zonal boundary method.

3.1.3. *Labyrinth*

The second test case considers the solution of the heat conduction equation (1) in a domain resembling a labyrinth. The labyrinth is built of two walls that are kept at constant temperatures of $T = 100°$ and $T = 0°$, as shown in Fig. 7a. An additional complexity, besides the geometric complexity, is introduced through the specification of the boundary conditions: on the open boundaries AB and CD, periodic boundary conditions are specified.

The nine zones used for performing the x-coordinate sweeps are shown in Fig. 7b. The y-direction splitting into zones needs more attention due to the periodic boundary conditions given on AB and CD. To permit an implicit solution, zone 1a is copied on top of zone 1b, as shown in Fig. 7c, to form a contiguous single zone (shown by the shaded region). In the combined zone, the y-sweep lines are defined between two Dirichlet-type boundaries (shown by the thick solid lines). This procedure, which is straightforward to perform in the ADMZI method, eliminated the periodic boundary lines by replacing them with a regular interior line. The elegant way of combining zonal splitting and unconventional boundary conditions within the ADMZI approach is remarkable. Standard multi-zone methods can implement this boundary condition explicitly only, leading to further numerical complications and degradation of the convergence properties.

A uniform Cartesian mesh of 161×161 points has been employed. The time-accurate solution method has been

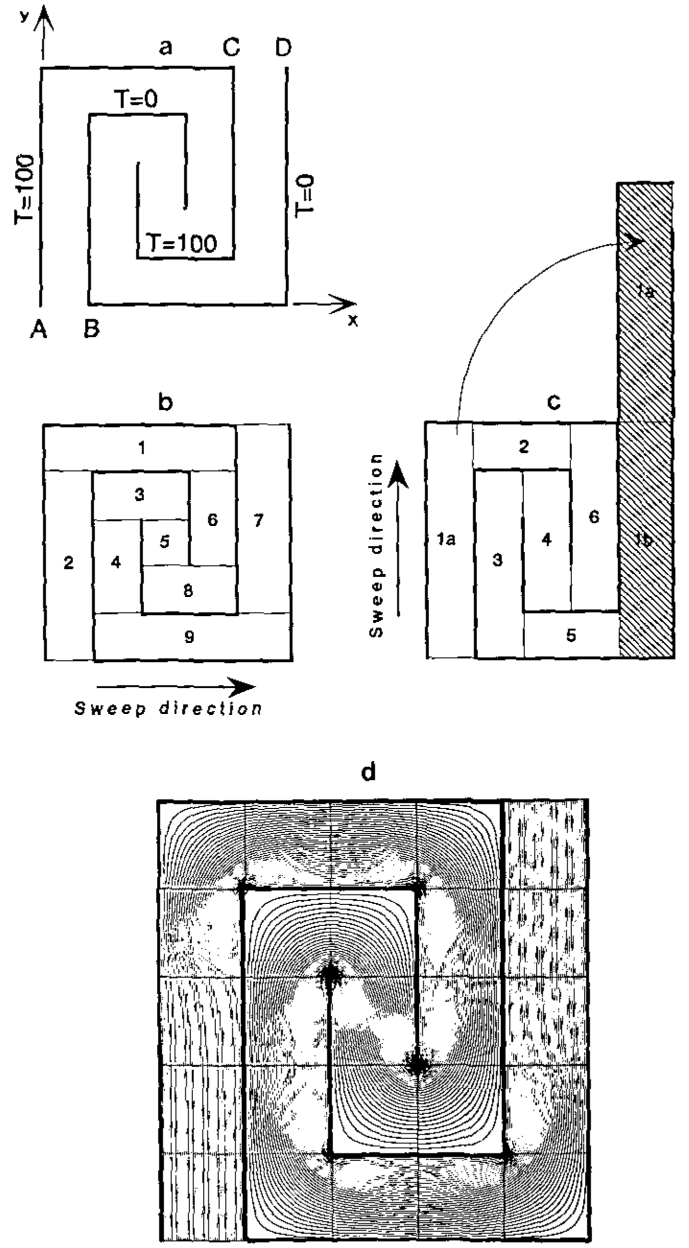

FIG. 7. The labyrinth test case.

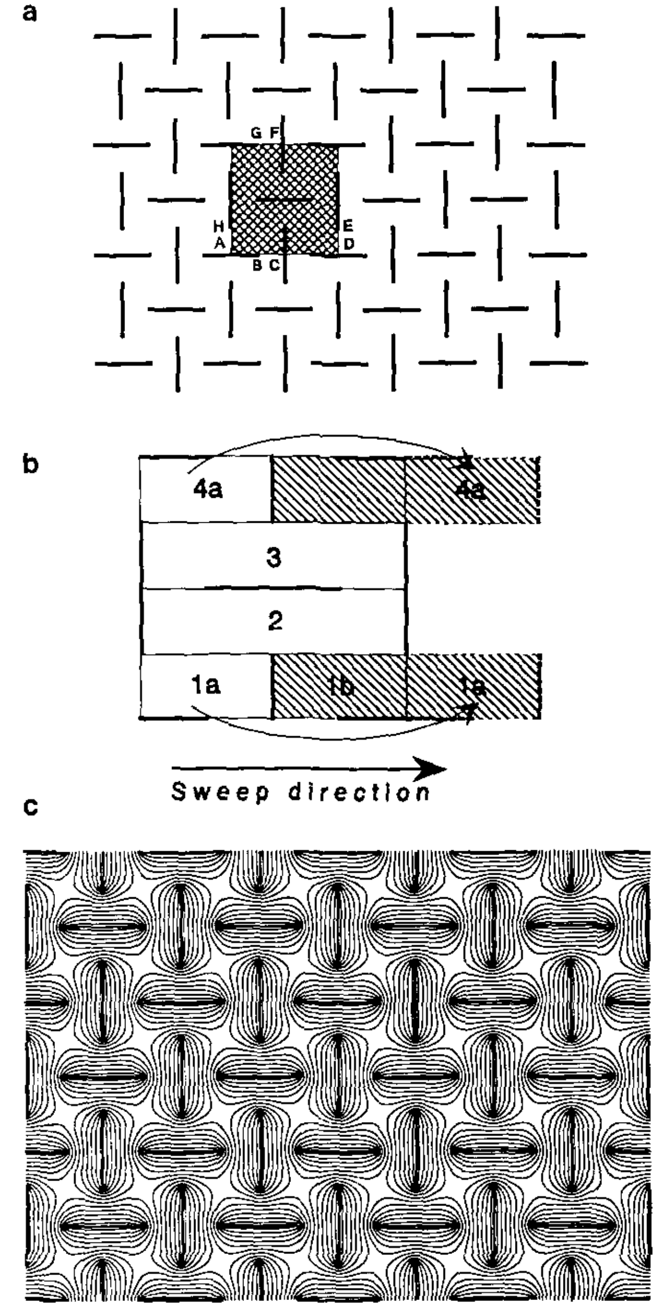

FIG. 8. The cooling fins test case: (a) geometry; (b) first set of zones; (c) solution.

advanced in time until a steady solution has been obtained. The steady state solution is given in Fig. 7d. As expected, very smooth contour lines are obtained in the whole domain, including through the zonal boundaries and the periodic boundary.

3.1.4. *Cooling Fins*

The last numerical example considers an infinite array of cooling fins, part of which is shown in Fig. 8a. Flow through this geometrical arrangement can lead to significant heat transfer augmentation and reduction of the pressure losses [15]. A significant simplification can be done by exploiting the symmetry properties of the problem. The shaded region in Fig. 8a defines a basic cell of the problem. The complete (infinite) fin configuration is composed of these cells. Consequently, the problem should be solved in the simpler region shown by the dashed domain in Fig. 8a, but with unconventional boundary conditions on the "open" boundaries between the fins. On the fins, Dirichlet-type boundary conditions are specified; the horizontal and vertical fins are kept at constant temperatures of $T=100°$ and $T=0°$, respectively. To account for the infiniteness, periodic boundary conditions are specified on the open boundaries, e.g., for the boundaries *BC* and *GF* or *AH* and *DE*.

The set of zones required for performing the x-sweeps of the ADMZI method are shown in Figs. 8b. The ease of implementing implicitly complex boundary conditions with the ADMZI approach is exhibited in the treatment of the periodic boundary conditions. The approach is similar to the previous example (the labyrinth case), and it will be elaborated only for zones 1a and 1b. An implicit sweep in zone 1a cannot be completed because of the periodic boundary condition specified on *AH*. By copying zone 1a to the right of zone 1b (see Fig. 8b), such that *AH* coincides with *DE*, the zonal boundary is converted into a regular interior line. A single contiguous zone (as shown by the shaded region in Fig. 8b) is created and implicit sweeps between two Dirichlet boundaries can be performed. Zones 4a and 4b are treated in a similar manner. The y-sweeps are performed in an equivalent way and therefore will not be further elaborated.

The temperature contours obtained by employing a mesh of 161×161 points (in a basic cell), are shown in Fig. 8c for the region shown in Fig. 8a.

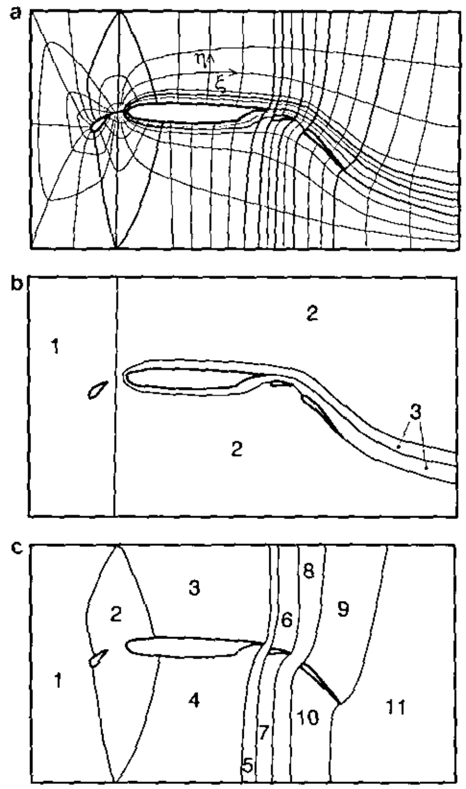

FIG. 9. Domain decomposition of the multi-element airfoil case: (a) geometry; (b) first set of zones; (c) second set of zones.

3.2. Multi-element Wing

In this example, the domain decomposition required to solve by the ADMZI method the viscous flow over a multi-element two-dimensional wing will be outlined. The geometric configuration is similar to a case presented by Stewart [16], who considered the four-element airfoil of the A310 Airbus in a landing configuration. The purpose of this example is to show conceptually how the ADMZI method can handle problems with complex geometry. The separate definition of the zones for each stage of the ADI method is the novelty introduced by the ADMZI method, and therefore only that aspect is considered. Once this decomposition is determined, the generation of the discrete equations and the solution of the implicit set of algebraic equations within each zone is a straightforward task.

A possible mesh is shown in Fig. 9a. The mesh consists of all three common grid-topologies. An O-type mesh is created around the leading-edge flap, a C-type mesh is used over the main wing section, while an H-type mesh is generated over the two trailing edge flaps. Yet, the spatial topology of the mesh sections is irrelevant in the case of the ADMZI method. In this method, the topology of multi-dimensional grids has no significance, because only the one-dimensional sweep lines of the ADI stages have a meaning.

One set of zones is shown in Fig. 9b. The sweep lines (the mesh lines along which the algebraic equations are solved simultaneously) are not shown, but their shape in each zone may be determined from Fig. 9a. Zone 1 includes the O-type sweeps (with periodic boundary conditions at the edges). Zone 3 is a regular C-sweep zone. Zone 2 also includes C-type sweep lines, but a branch-cut line is absent. The second set of zones is shown in Fig. 9c. A total of 11 zones is used in this direction.

Although the geometric complexity of the present problem is significantly greater than in the previous cases, the properties of the zones of both ADI stages are unchanged. All the zones are quadrilateral regions with specified boundary conditions at both edges of the sweep lines. In the other direction (parallel to the sweep direction), the boundary lines may be composed of any combination of physical or zonal boundaries.

4. CONCLUDING REMARKS

The ADMZI method introduces a novel approach for solving differential equations in complex domains. The essence of the ADMZI method is the proper decomposition of the computation domain into sets of zones that conform with the stages of the ADI method (or any other approximate factorization method). Different sets of zones are generated for each stage, in contrary to available multi-zone methods where only one set of zones is employed. Each zone of the ADMZI method has very simple properties; in the two-dimensional case, they are always single-connected quadrilateral regions with boundary conditions specified on the edges of the zone. All the interior points are regular field points. This arrangement considerably simplifies the setup and solution of the discrete equations in each zone.

The boundaries at the edges of the sweep lines are always physical boundaries, and therefore the sweeps can be completed implicitly. The implicit solution of the PDE enhances the stability and convergence properties of the solution method. Sub-iterations for updating explicit zonal boundary conditions are not required and hence the method offers enhanced efficiency. Moreover, problems arising from nonconservative approximations between different zones are avoided in the ADMZI method. These properties are crucial in the case of time-dependent problems, where implicit solution is necessary for feasible calculations of cases with very complex geometry. The ADMZI method maintains the temporal accuracy of the scheme, while in standard multi-zone methods the accuracy degrades, if the zonal boundary conditions are lagged in time.

The suggested ADMZI method can treat implicitly not only complex geometry, but also complex or unconventional boundary conditions with unusual grid topologies. Cases 3.1.3 and 3.1.4 of the Results section demonstrated how the ADMZI method can tackle such problems by assembling sub-zones with hard to implement boundary conditions to form a zone with conventional boundary conditions at the edges.

Domain decomposition is usually implemented for solving PDE in complex domains. Yet, other reasons may require the use of the multi-zone approach. Such is the case if memory limitations inhibit the simultaneous solution of the whole region. In this case, the domain of calculation is decomposed into several zones, so that each zone can fit inside the allocated memory. Contrary to available solution methods, in the ADMZI method the solution is still fully implicit and identical with the solution that would have been obtained from the in-core implicit solution of the whole domain.

The method can, of course, be applied to other line methods such as successive line overrelaxation methods, but here the gain in efficiency is less, although only one set of zones (along the direction of the "lines") should be generated. Although the ADMZI method has been elaborated and tested in the present study for two-dimensional cases only, the same methodology can be used to extend it to three-dimensional cases. The addition of the third direction requires the definition of three sets of zones, one set for each stage of the ADI method. Otherwise, no changes are required; each zone is a simple hexahedron and the sweep lines start and end on physical boundaries.

Still, the most time-consuming part of the ADMZI method is the setup and the implicit solution of the equations. These standard steps are performed within each zone separately. Consequently, standard vectorization and parallelization techniques can be used and most of the approaches developed recently can be equally well applied to the ADMZI method. In particular, the solutions of the zones belonging to a single stage of the ADMZI method are uncoupled, permitting parallel solution of the zones.

The ADMZI approach can be used in a wide range of applications that require the solution of PDE in complex regions. Moreover, the method can be applied to existing solution methods (based on ADI or other factorization methods) without major revisions because the ADMZI method is essentially a driver for solving any set of PDE. Many available compressible and incompressible viscous flow solvers fall in this category.

ACKNOWLEDGMENTS

This work was supported in part by a grant from the Defense Ministry of Israel. The technical monitor was A. Kuritzki.

REFERENCES

1. T. L. Holst, S. D. Thomas, U. Kaynak, K. L. Gundy, J. Flores, and N. M. Chaderjian, NASA Technical Memorandum 86774, 1985 (unpublished).
2. M. M. Rai, *J. Comput. Phys.* **66**, 99 (1986).
3. E. H. Atta and J. Vadyak, *AIAA J.* **21**, 1271 (1983).
4. M. M. Rai, *J. Propulsion* 3, 387 (1987).
5. F. C. Dougherty, J. A. Benek, and J. L. Steger, NASA Technical Memorandum 88193, 1985 (unpublished).
6. H. A. Schwarz, *J. Reine Angew. Math.* **70**, 105 (1869).
7. C. Y. Perng and R. L. Street, *Int. J. Numer. Methods Fluids* **13**, 269 (1991).
8. S. E. Rogers, AIAA Paper 91-1611, 1991 (unpublished).
9. R. Beam and R. F. Warming, AIAA Paper 77-645, 1977 (unpublished).
10. W. R. Briley and H. McDonald, *J. Comput. Phys.* **34**, 54 (1980).
11. D. Kwak, J. L. Chang, S. P. Shanks, and S. Chakravarthy, *AIAA J.* **24**, 390 (1986).
12. M. Rosenfeld, D. Kwak, and M. Vinokur, *J. Comput. Phys.* **94**, 102 (1991).
13. A. R. Mitchell and D. F. Griffiths, *The Finite Difference Method in Partial Differential Equations* (Wiley, New York, 1980), p. 65.
14. S. Stanaway, W. McCroskey, and I. Kroo, AIAA Paper 92-0024, 1992 (unpublished).
15. A. Degani, Y. Yassour, and M. Wolfshtein, TAE Report No. 512, Technion, Haifa, Israel, 1983 (unpublished).
16. M. Stewart, AIAA Paper 90-1606, 1990 (unpublished).