NASA Technical Memorandum 101023

Knowledge-Based Flow Field Zoning

Alison E. Andrews

(NASA-TM-101023) KNOWLEDGE-BASED FLOW FIELD ZONING (NASA) 13 p CSCL 09B N88-30314

Unclas
G3/61 0159383

August 1988

NASA
National Aeronautics and
Space Administration

NASA Technical Memorandum 101023

Knowledge-Based Flow Field Zoning

Alison E. Andrews, Ames Research Center, Moffett Field, California

August 1988

NASA
National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

KNOWLEDGE-BASED FLOW FIELD ZONING

Alison E. Andrews [1]

ABSTRACT

Automation of flow field zoning in two dimensions is an important step towards easing the three-dimensional grid generation bottleneck in computational fluid dynamics. A knowledge-based approach works well, but certain aspects of flow field zoning make the use of such an approach challenging. A knowledge-based flow field zoner, called EZGrid, has been implemented and tested on representative two-dimensional aerodynamic configurations. Results are shown which illustrate the way in which EZ-Grid incorporates the effects of physics, shape description, position, and user bias in a flow field zoning.

1. INTRODUCTION

At the beginning of this decade, Chapman [1] identified three-dimensional (3-D) grid generation as a pacing item in computational fluid dynamics (CFD). In a recent survey of three-dimensional grid generation capabilities, Thompson and Steger [2] contend that grid generation continues to impede progress towards achieving quick, realistic fluid flow simulation. It is difficult to generate a reasonable, single grid about a general, 3-D configuration. The factors primarily responsible for this difficulty [3] are complex geometries, the need for selective grid refinement, and computer memory and speed limitations. Decomposition of the physical domain into simpler subdomains, or zones, can be used effectively to solve these difficulties. Domain decomposition, or flow field zoning, can reduce topological complexity, permit local grid refinement (if grid lines are allowed to be discontinuous across zonal interfaces), and provide a convenient mechanism for splitting a problem into smaller chunks for either sequential or parallel computation on portions of the domain.

Many CFD researchers and practitioners have adopted a composite zonal grid approach. The predominant version is the nonoverlapping, or composite block approach [2]. As experience is gained with these methods, it has become apparent that: (1) flow field zoning must be performed *well* in order to reap its potential benefits, (2) flow field zoning must be performed *quickly* in order to significantly ease the grid generation bottleneck, and (3) the first two requirements are difficult to satisfy.

To perform flow field zoning well, the following expertise is required: experience with composite zonal grid methods, familiarity with available grid generation capabilities, knowledge about the behavior of the zonal flow solver to be used, fluid dynamics knowledge (including the ability to predict the important physical features of the flow solution), and evaluation criteria. To perform zoning quickly, the user must have both expertise and fast, easy-to-use graphics tools. The problem lies primarily with the expertise requirement. Flow field zoning expertise is not widespread and is not easily taught. In three dimensions, even expert users find it difficult to visualize and specify general zonal interfaces. Finally, zoning experts do not agree on evaluation criteria. The growing consensus [2,4,5,6] is that flow field zoning must be automated.

---

[1] Research Scientist, NASA Ames Research Center, Moffett Field, California

The goals of the present research are to: (1) lay the foundation for an automated 3-D zonal grid generation capability by developing a demonstration computer program which can automatically zone representative 2-D aerodynamic configurations, and (2) determine the applicability of a knowledge-based programming approach to flow field zoning. This paper contains a (necessarily) cursory introduction to knowledge-based programming (Section 2), a brief description of the flow field zoning model and language (Section 3), an overview of the program implementation (Section 4), and zoning results generated for representative two-dimensional test cases (Section 5).

2. A KNOWLEDGE-BASED PROGRAMMING APPROACH

Knowledge-based programming is an artificial intelligence problem-solving approach that has met with some success in the solution of real-world problems in a variety of fields. Like any computer program, a knowledge-based system has the three basic elements required for computer-based problem solving: a representation of the objects of the problem, a means of operating on those objects to transform them to different objects, and a strategy for applying those operators so that a solution is obtained. In a knowledge-based system, those elements are called, respectively, the database, the knowledge base, and the control structure or inference engine. Unlike conventional computer programs, however, these elements are kept separate, and emphasis is placed on the amount and quality of the domain knowledge contained by the system rather than on the methods used to apply it.

The following guidelines for identifying appropriate, tractable problems for the application of knowledge-based techniques have evolved over the last several years:

1. The problem has no closed form or algorithmic solution.
2. Expertise is required to solve the problem.
3. An expert can solve the problem fairly quickly.
4. The problem is important to solve.
5. The skill is routinely taught to nonexperts.
6. Solution of the problem does not involve perception.
7. Experts agree on how to solve the problem.
8. Solution by analysis is easier than by synthesis.

Several aspects of flow field zoning follow these guidelines. Zoning is an ill-structured problem to which no general solution has been found. Expertise is required to perform the task quickly and well. An expert can design and generate a flow field zoning in several days or weeks, depending on the complexity of the configuration. Finally, because zoning is an integral part of the effort to make 3-D grid generation faster and easier, it is an important problem to solve. Unfortunately, other aspects of zoning run counter to these guidelines. The art of flow field zoning is not easily taught. It has an unmistakable perceptual element, involving qualitative shape and position information. While there are recognized zoning experts, ideas differ (and are even still evolving) as to what constitutes a good zoning, so the solution preferred by one expert may be less acceptable to another. Lastly, the process of flow field zoning has been modeled as one in which a solution is designed (i.e., a synthesis procedure) as opposed to selected (i.e., an analysis procedure). These latter aspects preclude a rapid and straightforward application of knowledge-based techniques to flow field zoning, necessitating the following system development approach.

1. Develop a model and language to describe the fundamentals of 2-D flow field zoning.
2. Debug the basic components of the model and language (concerning zoning objects and processes) by implementing an *interactive* knowledge-based system, in which the mechanics and bookkeeping associated with zoning (i.e., interface

curve generation, connectivity, containment, and adjacency) are automated, but the user supplies the essential elements of perception, individual bias, and zoning design knowledge (i.e., the aspects of the problem which are more difficult to automate).

3. Increase the level of system automation incrementally by replacing the elements previously supplied by the user, one at a time, with automated versions based on the remaining components of the zoning model and language.
4. Use existing interactive grid generation capabilities.

By using this approach, a 2-D flow-field-zoner demonstration system has been successfully developed, as described in the following sections.

3. A MODEL AND LANGUAGE FOR ZONING

The key to automating any process lies in the ability to describe the process unambiguously, which is possible only if the nature of the process is understood, and if there is a language that can express the concepts involved. There is no theory which governs the way in which a flow field should be partitioned into zones. In the absence of theory, it is necessary to formulate a model based on observation of how experts perform the task. Here, zoning is modeled as a design problem; a zoning is designed by applying a sequence of actions to zoning objects, modifying the initial situation by stages until an acceptable zoning is achieved. This idea is illustrated by the sketch in Fig. 1. Factors which influence the choice of a zoning action at any stage of the design process include: (1) basic zoning criteria and guidelines, (2) geometry (both quantitative and qualitative information), (3) fluid physics, and (4) individual bias (involving flow solver capabilities, personal experience, computational objectives, and aesthetics).

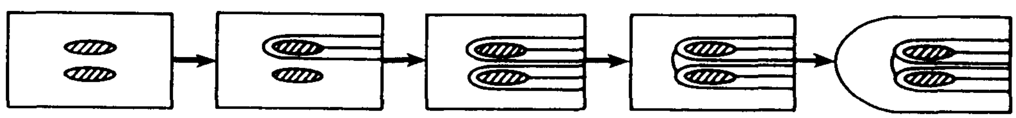


Fig. 1. Zoning as a sequence of actions.

The language which was developed to describe and automate zoning consists of five major elements: object terms, action terms, qualitative shape and configuration description, a zoning archetype (a collection of parameters which can be adjusted to characterize a user's bias in zoning design), and zoning design knowledge encoded in the form of plans (sequences of zoning actions). See Ref. 7 for a detailed discussion of this model and language, which is not possible here. So that the reader can understand the significance of the results shown later, however, some explanation of the last three language elements is included.

The perceptual information needed to automatically zone a flow field is provided interactively by the user by means of a simple qualitative shape and configuration description language. The shape of an object is described as the composition of primitive parts, each of which is described in terms of several attributes, including primitive shape (primitive shapes are ellipse, teardrop, bullet, eye, wedge, and rectangle). The way in which an object is divided into parts and the way those part shapes are described are matters of individual interpretation, which is one of the reasons shape description is not automated. A configuration of objects is described by first grouping objects according to their probable influence on each other, and then specifying qualitative relationships and approximate separation distances between objects within the same grouping and between groupings.

One solution to the problems arising from a lack of expert consensus in zoning evaluation is to establish a zoning archetype, or standard, which can be *tuned* to reflect a user's bias. The archetype is defined as the collection of parameters which best characterize user bias and are easily measurable. It is tuned by the assignment of qualitative weights to each parameter, representing the subjective importance or acceptability of that parameter. The parameters and their possible qualitative weights are listed in Table I. These qualitative weights are used to guide the automatic design of flow field zonings.

Table I. Zoning archetype parameters

ARCHETYPE PARAMETER	POSSIBLE VALUES
SIMPLICITY ZONE CORNER SKEWNESS ZONE SIDE SMOOTHNESS ZONE SIDE MAPPING DISPARITY GRID POINT EFFICIENCY ORTHOGONALITY AT BODY SURFACES SURFACE vs. FIELD QUANTITIES WAKE RESOLUTION	NO LOW MEDIUM HIGH } IMPORTANCE
ZONE TUPLE POINTS SINGULARITIES AT BODY SURFACES ZONE/BODY INTERSECTIONS VISCOSITY IN MORE THAN ONE DIRECTION	ALLOWED BUT NOT IMPORTANT ALLOWED SOMEWHAT DISCOURAGED DISCOURAGED STRONGLY DISCOURAGED NOT ALLOWED

A zoning is designed through the application of a sequence of actions to zoning objects. It is in the determination of this sequence of actions that much of zoning expertise lies. Stored in the knowledge base are subplans, which are sequences of zoning actions that govern the zoning of a single grouping. If there is only one grouping, the subplan which is selected becomes the overall plan. If there are multiple groupings, the selected subplans are assembled to obtain the final plan. The design of a zoning is thus transformed into an analytic process in which the configuration is broken down into simpler groupings when possible, and subplans are selected and assembled into a coherent plan.

4. A KNOWLEDGE-BASED FLOW FIELD ZONER

A knowledge-based system called EZGrid (Expert Zonal Grid generator) has been developed using the approach outlined in the section on knowledge-based systems, and based on the model and language just described. The program was implemented in C, Franz Lisp, and MRS [8]. MRS is a logic programming language which processes symbolic propositions (facts and if-then rules) using pattern matching and logical deduction (for example, given the fact (man Socrates) and the rule (if (man x) then (mortal x)), MRS could deduce that Socrates is mortal).

Over 400 rules comprise the EZGrid knowledge base, enabling it to operate in either interactive or automatic mode. Operation of EZGrid is divided into two parts: the set-up phase and the stage-by-stage zoning design loop. The first step within the set-up phase is the tuning of the zoning archetype. The user may set the archetype parameter values interactively, or by reading in a file, or by keeping the default settings. Geometry input can be interactive or accomplished by reading in a data file of (x,y) coordinates. In automatic mode, the user is asked to provide a qualitative shape description (using menus) for each body defined. The outer boundary of the computational domain may then be specified by the user with the same options as offered in geometry input, or may be left to EZGrid to determine. If left to EZGrid, the user is asked to supply inflow conditions such as freestream Mach number, angle of attack, and flow steadiness. These and other physical parameters may also be needed later during the zoning design. EZGrid asks for their values only once, and only when and if needed. The final step of the set-up phase (in automatic mode only) is the input of the qualitative configuration description.

The second phase is then entered, in which a loop is executed once for each stage of the zoning design. First, the situation is assessed (i.e., object relations and properties are examined and updated if necessary). In automatic mode, a plan is constructed the first time through the loop, followed by a search for all possible zoning actions for that situation. If the action at the top of the list of pending actions of the plan is one of the possible actions, it is selected for execution. In interactive mode, no plan is constructed, and the user must choose an action from among the possible ones. The user must also specify any control points and angles needed by the system to generate the zonal interface curves (cubic splines) for the chosen action. In automatic mode, curve generation proceeds without input from the user. New zoning objects are constructed from these curves, and new object properties and relations are established. Control returns to the beginning of the loop where the new situation is assessed. If is discovered that the outer boundary domain has been zoned, the design is complete and output files containing interface position information (and, implicitly, zone topologies) are prepared and written. For simple cases, there are rules in the knowledge base which determine the number and distribution of grid points for each zone. In general, however, no attempt is made to automate the grid generation task.

In order to quantitatively evaluate and compare flow field zonings, the possible qualitative weights which can be assigned to the zoning archetype parameters were translated into numerical weights through a calibration of the archetype. Three different configurations of NACA 0012 airfoil pairs were selected as test cases. Several candidate zonings were generated for each case using EZGrid in interactive mode, and were shown to five flow field zoning experts. The experts were asked first to assign qualitative weights to the archetype parameters representative of their own views and appropriate for the test cases. No two of the resulting archetypes were identical. They were then asked to order the candidate zonings for each test case according to their preference, consistent with the archetype as they had tuned it. Each parameter has a measurement function which, when applied to a zoning (prior to grid generation), yields a number that denotes a penalty for that aspect of the zoning. The penalties are each multiplied by the numerical equivalents of the qualitative weights of the archetype, and are summed to produce a score for that zoning. Comparison of scores for the candidate zonings yields an ordering reflecting EZGrid's "preference." The translation of qualitative to numerical values was adjusted so as to maximize the number of matches between expert preferences and EZGrid preferences. Out of fifteen orderings, in only two does EZGrid fail to choose the same "best" candidate as the expert. It would be misleading to state that these results are statistically significant, but it is reasonable to claim that user bias has a measurable effect on flow field zoning design, and that the proposed zoning archetype is a promising method of evaluating zoning results in the absence of universally accepted criteria.

5. COMPUTED RESULTS

The interactive mode of EZGrid is completely general, and can be used to zone any two-dimensional problem. The automatic mode is more limited in scope due to the finite number of subplans in the knowledge base. It does provide the capability of zoning *representative* aerodynamic configurations, including single bodies, multiple-body single-grouping configurations, and multiple-body multiple-grouping configurations. Figure 2 shows two different zonings for a single body (an axisymmetric AOTV cross-section) with a shape description composed of two primitive parts. The upstream part is described as a half-bullet in the first case, and as a half-ellipse in the second. Geometry, inflow conditions, and user bias are identical for both cases, thereby illustrating the effect shape has on zoning design. In all of the results shown, Steinbrenner's GRIDGEN2D [9] was used to interactively generate the grids.

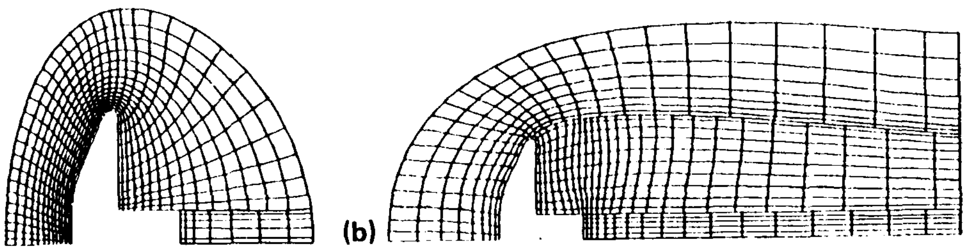


Fig. 2. Effect of shape description.

Figures 3a-3d are examples of the effect of viscosity, object separation distance, and user bias on the zoning of a single grouping containing two NACA 0012 airfoils. Each case has inflow conditions of 5° angle of attack and $M_\infty = 0.8$, and steady flow. EZGrid uses this information to determine the location of the outer boundary. Even in automatic mode, EZGrid presents the outer boundary location in the form of a suggestion which the user may follow or ignore. In the cases shown, the suggested distances (which were much greater than the distances actually used) were rejected in favor of closer boundaries simply for presentation purposes. Figure 3a shows a zoning automatically generated for a case in which an inviscid solution is sought, and where the archetype parameter settings allow both singularities and zone/body intersections. The remaining cases were all generated for viscous computations, and therefore contain C-type zones around each body. The zoning in Fig. 3b results when the two objects are considered to be "close" to each other. The zoning in Fig. 3c results when the airfoils are considered to be "far apart," and there is room for an intervening zone between the two C-type zones. In both cases, the archetype was tuned to *allow* zone tuple points (points where more than two zones meet). The zoning in Fig. 3d differs from that of Fig. 3c only because the archetype has been tuned so that tuple points are *strongly discouraged* and mapping disparity (the ratio of the maximum separation

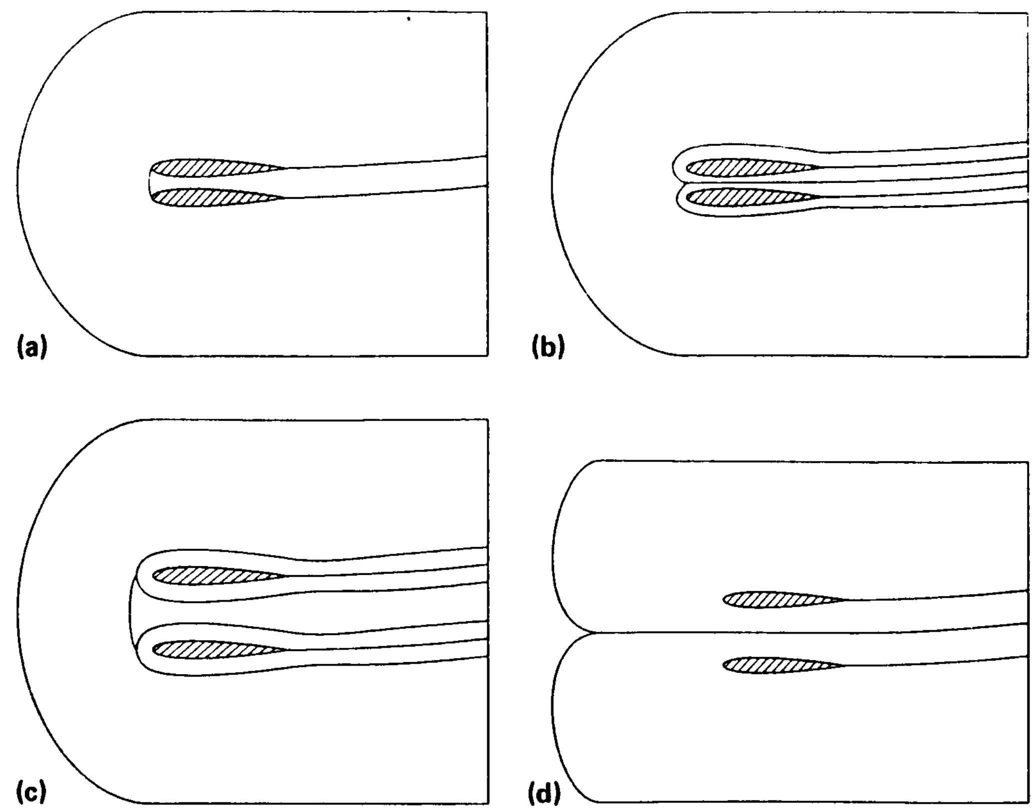


Fig. 3. Different zonings for an NACA 0012 airfoil pair.

distance between opposite sides to the minimum separation distance) is set to *low* or *medium* importance. The result is a simpler zoning, but one which may lead to grid generation difficulties.

The composite zonal grids shown in Figs. 4a-e resulted from a study aimed at comparing the performance of EZGrid to that of a human expert. A zoning expert was given two test cases to zone, and was asked to tune the zoning archetype for each and sketch the desired zoning. Zonings based on the expert's sketches were generated using EZGrid in interactive mode, and are shown in Figs. 4a and 4c. The same geometry, inflow conditions, and zoning archetype parameter weights were input to EZGrid in automatic mode, and the zonings shown in Figs. 4b and 4d resulted. When the zonings were evaluated by EZGrid and compared, EZGrid preferred its own two zonings to those of the expert. For the first case (the one with the flap at a positive angle of attack), the zonings are comparable. The primary differences are that the EZGrid zoning is simpler (fewer zones and zonal interfaces), has one fewer tuple points, and has C-type grids around each body. These aspects are intrinsically neither advantageous nor disadvantageous – what matters is how they are viewed by the user as reflected in the zoning archetype. For that case, the expert had set the simplicity parameter to *high importance* and the tuples parameter to *discouraged*, so those aspects were responsible for the better score for the EZGrid result. In the second case, although the zoning generated by EZGrid appears to be less desirable than that generated by the expert, the evaluation again resulted in a better score for the EZGrid zoning. The archetype for case 2 was identical to that of case 1 except for one parameter, which does not affect the outcome for this case. The emphasis on simplicity and the discouragement of tuple points again made the difference since the EZGrid zoning has only two zones (to the expert's four) and no tuple points (to the expert's two). Figure 4e shows the zoning generated automatically by EZGrid for case 2 using the default zoning archetype weights. The significant difference in the archetypes lies in the importance placed on zone side mapping disparity – in the previous zonings it had a *medium importance* value, whereas in the default archetype, it has a *high importance* value, which effectively discourages the sort of zoning found in Fig. 4d.

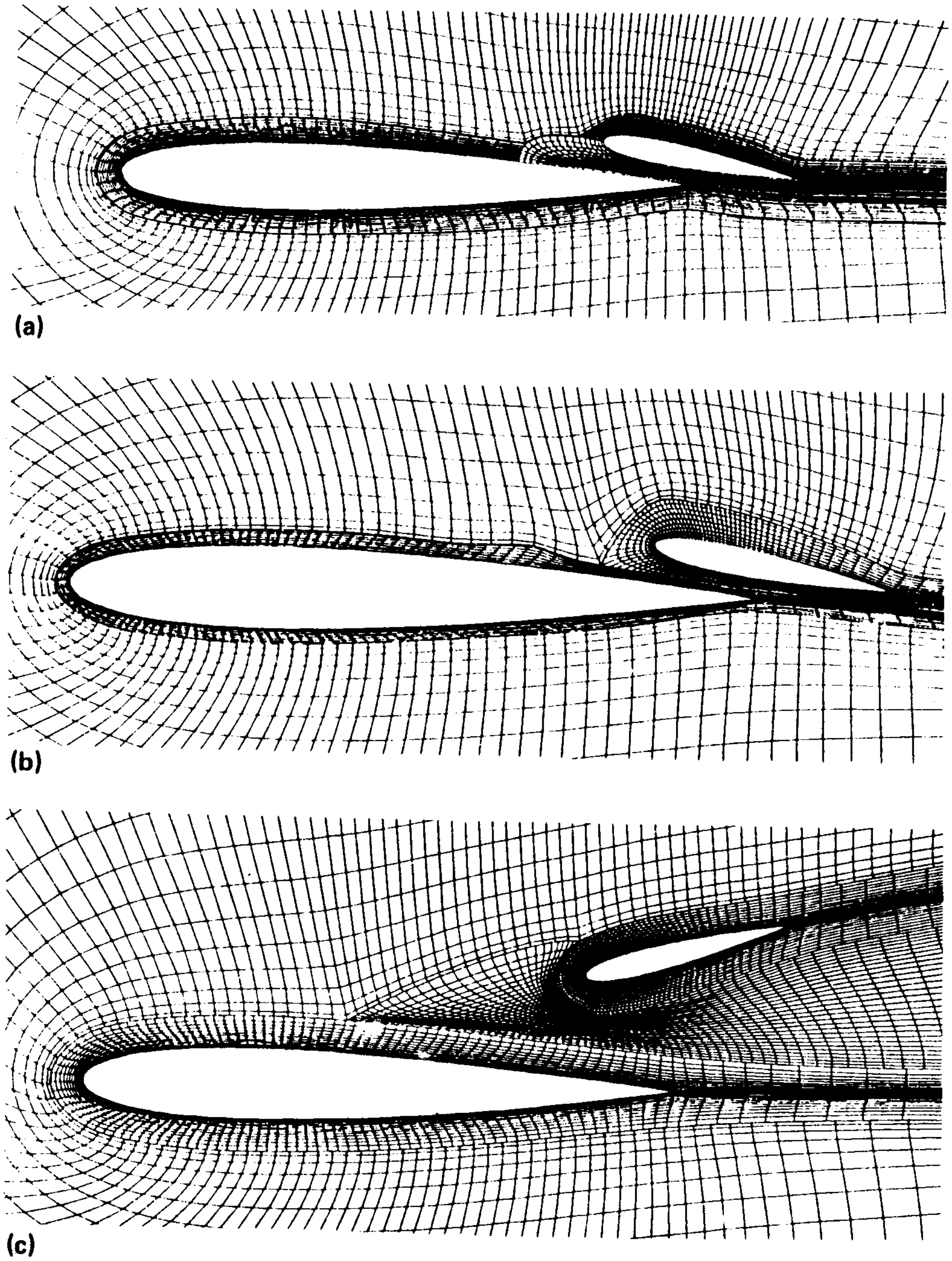
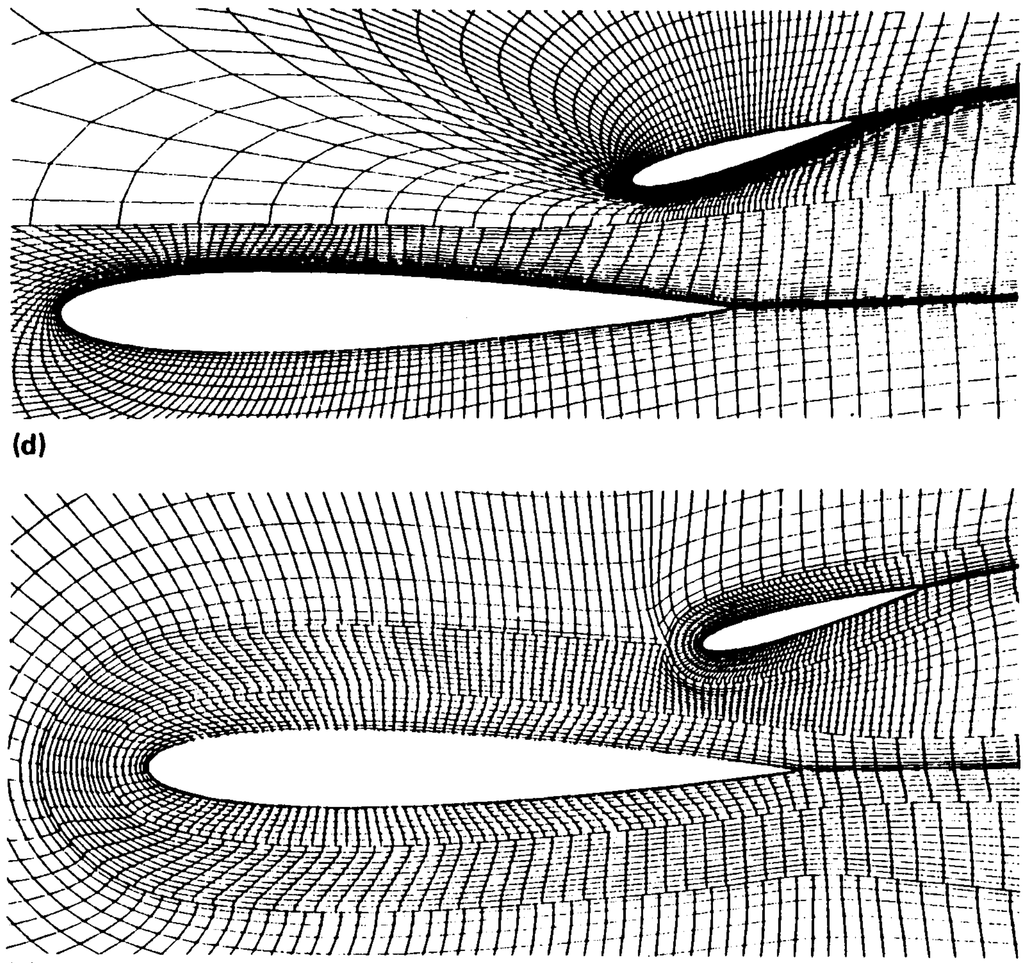


Fig. 4. EZGrid/human expert comparison study.



(d)

(e)

Fig. 4. Concluded.

5. CONCLUSIONS

Many researchers have found flow field zoning to be an effective solution to the 3-D grid generation bottleneck of CFD. However, automation of flow field zoning is necessary to promote its widespread use. A knowledge-based approach to automating flow field zoning in two dimensions has been investigated, and a demonstration system has been implemented which is capable of automatically zoning the flow field about representative 2-D aerodynamic configurations. Several examples are shown which illustrate the effect of physics, configuration, shape description, and user bias on the zoning design. A knowledge-based approach is reasonable for automating flow field zoning, but several aspects of zoning prevent the straightforward application of such techniques. The difficulties which arise are not insurmountable; they merely necessitate a more lengthy system development process.

REFERENCES

1. CHAPMAN, D.R. – Trends and Pacing Items in Computational Aerodynamics, Lecture Notes in Physics, Ed. Reynolds, W.C. and MacCormack, R.W., Springer-Verlag, 1980.
2. THOMPSON, J.F., and STEGER, J.L. – Three-Dimensional Grid Generation for Complex Configurations - Recent Progress, AGARDograph No. 309, Ed. Yoshihara, H., North Atlantic Treaty Organization Advisory Group for Aerospace Research and Development, 1988.
3. RAI, M.M. – A Conservative Treatment of Zonal Boundaries for Euler Equation Calculations. J. Comp. Physics, vol. 62, no. 2, pp. 472-503, 1986.
4. KUTLER, P. – A Perspective of Computational Fluid Dynamics. NASA TM-88256, May, 1986.
5. STEINBRENNER, J.P., KARMEN, S.L., and CHAWNER, J.R. – Generation of Multiple Block Grids for Arbitrary 3D Geometries, AGARDograph No. 309, Ed. Yoshihara, H., NATO Advisory Group for Aerospace Research and Development, 1988.
6. WEATHERILL, N.P., and SHAW, J.A. – Component Adaptive Grid Generation for Aircraft Configurations, AGARDograph No. 309, Ed. Yoshihara, H., NATO Advisory Group for Aerospace Research and Development, 1988.
7. ANDREWS, A.E. – A Knowledge-Based Approach to Automated Flow Field Zoning for Computational Fluid Dynamics. Ph.D. Dissertation, Department of Mechanical Engineering, Stanford University, 1988.
8. GENESERETH, M.R., GREINER, R., GRINBERG, M.R., and SMITH, D.E., – The MRS Dictionary. Stanford Heuristic Programming Project, Report No. HPP-80-24, January, 1984.
9. STEINBRENNER, J.P. – GRIDGEN2D User's Manual. General Dynamics, Fort Worth Division, CFD Report No. 063-4-8601, June, 1986.

NASA
National Aeronautics and Space Administration

Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TM-101023		

4. Title and Subtitle	5. Report Date
Knowledge-Based Flow Field Zoning	August 1988
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
Alison E. Andrews	A-88274
	10. Work Unit No. 505-60
9. Performing Organization Name and Address	11. Contract or Grant No.
Ames Research Center Moffett Field, CA 94035	13. Type of Report and Period Covered Technical Memorandum
12. Sponsoring Agency Name and Address	14. Sponsoring Agency Code
National Aeronautics and Space Administration Washington, DC 20546-0001	

15. Supplementary Notes

Point of Contact: Alison Andrews, Ames Research Center, MS 258-1
Moffett Field, CA 94035 (415) 694-6741 or FTS 464-6741

16. Abstract

Automation flow field zoning in two dimensions is an important step towards easing the three-dimensional grid generation bottleneck in computational fluid dynamics. A knowledge-based approach works well, but certain aspects of flow field zoning make the use of such an approach challenging. A knowledge-based flow field zoner, called EZGrid, has been implemented and tested on representative two-dimensional aerodynamic configurations. Results are shown which illustrate the way in which EZGrid incorporates the effects of physics, shape description, position, and user bias in a flow field zoning.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Computational fluid dynamics Knowledge-based systems Zonal grid generation	Unclassified-Unlimited Subject Category - 61

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	12	A02