N89-19828

KNOWLEDGE-BASED ZONAL GRID GENERATION FOR COMPUTATIONAL FLUID DYNAMICS

Alison E. Andrews
Applied Computational Fluids Branch
Mail Stop 258-1
NASA Ames Research Center
Moffett Field, California 94035

ABSTRACT

Automation of flow field zoning in two dimensions is an important step towards reducing the difficulty of three-dimensional grid generation in computational fluid dynamics. Using a knowledge-based approach makes sense, but problems arise which are caused by aspects of zoning involving perception, lack of expert consensus, and design processes. These obstacles are overcome by means of a simple shape and configuration language, a tunable zoning archetype, and a method of assembling plans from selected, predefined subplans. A demonstration system for knowledge-based two-dimensional flow field zoning has been successfully implemented and tested on representative aerodynamic configurations. The results show that this approach can produce flow field zonings which are acceptable to experts with differing evaluation criteria.

INTRODUCTION

Computational fluid dynamics (CFD) is becoming an essential tool in the understanding of fluid physics and in the design of aerospace vehicles. The long range goal is to be able to quickly and accurately simulate the viscous flow about realistic configurations[1]. Ideally, the time required for each computation should be small enough to permit incorporation of such a solution technique into the design cycle. Several aspects of CFD have been identified as pacing items, or areas which require significant advances before the goals of CFD can be realized[2,3]. Three-dimensional grid generation is prominent among them. Although grid generation methods have become fairly sophisticated, it is often extremely difficult to generate a reasonable single grid about a general, three-dimensional configuration[4-6]. Factors which are principally responsible for this difficulty are: 1) complex geometries, 2) the need for selective refinement to resolve fluid physics phenomena efficiently, and 3) limitations on the size of computer physical memory. Domain decomposition, or flow field zoning, is an effective solution. Partitioning a flow field into zones can reduce the topological complexity of the problem. It allows grid refinement to be limited to just those regions of the flow field where it is required in order to resolve high gradients. Also, if each zonal grid is small enough to fit into the physical memory of the computer, problems of any size can be tackled.

The full benefits of flow field zoning can be gained only if it is done well. A user must have the following qualifications: experience with zonal methods, familiarity with grid generation capabilities, fluid dynamics knowledge, knowledge about the flow solver being used, criteria for evaluating zonings, and imagination. Unfortunately, the following conditions exist which prevent the widespread use of a zonal approach: 1) those qualifications are possessed by few, and are not easily taught; 2) tedium and frequent error are inherent in zonal boundary condition specification, regardless of the user's expertise; 3) it is difficult even for an expert to visualize and specify general zonal boundary surfaces in three dimensions; and 4) zoning evaluation criteria are not well established. Clearly, automating flow field zoning is an important step towards achieving routine application of CFD to aerospace problems[4].

The goal of the present research is to lay the foundation for an automated zonal grid (composite grid) generation capability in three dimensions by developing a knowledge-based demonstration program capable of automated flow field zoning in two dimensions. There are aspects of the zoning problem which make a knowledge-based programming approach a natural choice. There are other aspects, however, which tend to discourage the use of knowledge-based techniques as they currently exist. The purpose of this paper is to briefly describe the demonstration system which has been implemented, focusing on the methods used to overcome the difficulties arising from these latter aspects.

EZGRID

A knowledge-based system called EZGrid (Expert Zonal Grid generator) has been developed to demonstrate the feasibility of using a knowledge-based approach to automate flow field zoning for typical two-dimensional CFD applications. The program was written in MRS[7,8] (a logic programming language developed at Stanford University), Franz Lisp, and C. It contains over 400 rules which enable it to automatically design flow field zonings and generate the necessary zonal boundary curves for representative two-dimensional aerodynamic configurations. The input consists of one or more geometry coordinate data files and the desired inflow conditions (Mach number, angle of attack, etc.). Zonal boundary coordinate data files are output in a form which can be accepted directly by a grid generator. Approximately three person-years were required for the development of EZGrid.

Figure 1 shows a simple zonal grid (i.e., that has only one zone) for a single airfoil in a subsonic, inviscid fluid at 4° angle of attack . In simple cases such as this, EZGrid can automatically set the grid generation parameters as well, so this grid was generated without human intervention. In general, EZGrid provides the boundaries and topologies for each zonal grid, and the user sets the grid generation parameters for the final step of grid generation. Figure 2 shows the seven-zone (or seven-block) grid designed by EZGrid for a four-element airfoil in a transonic, viscous fluid at 4° angle of attack. All of the grids in this paper were generated using the General Dynamics grid generator GRIDGEN2D[9]. Note that the airfoil in Figure 1 is identical to the downstream-most airfoil in Figure 2, demonstrating that in addition to shape and inflow conditions, the configuration plays an important role in determining the zoning.

EZGrid has both automatic and interactive modes of operation. Though an automated design is possible for a variety of zoning problems, a user may choose to design a zoning interactively (to test out a particular idea, for example).

A KNOWLEDGE-BASED APPROACH

There are several useful rules for identifying tractable task domains for knowledge-based system development[10], which are: 1) a closed-form or algorithmic solution cannot be found, 2) domain expertise exists, 3) the task can be performed by an expert within a reasonable amount of time (hours, days), 4) the task is cognitive (as opposed to perceptual), 5) the skill is routinely taught to nonexperts, and 6) the task is worth doing, and has a high payoff. Furthermore, a distinction is often made between problems which are solved by means of classification, or selection of predefined solutions, and problems whose solutions must be constructed or designed (analysis versus synthesis)[11]. Problems that require construction of a solution usually respond to knowledge-based techniques less readily than those which can be solved using selection or classification procedures.

Several aspects of flow field zoning obey these rules. Zoning is an ill-structured problem, and no satisfactory conventional solution has been found. Expertise is required to perform the task well, and an expert can design and generate a zoning in several days or weeks, depending on the complexity of the configuration. Finally, flow field zoning is an important element in the drive to make three-dimensional grid generation faster and easier, and is thus definitely worth doing.

Unfortunately, several aspects of flow field zoning break these rules. The art of flow field zoning is not easily taught, perhaps because there is no good language to describe the process. The task has an unmistakable perceptual element, involving qualitative shape and position information. The process of flow field zoning has been modelled as one in which a solution is designed as opposed to selected. Lastly, while there are recognized experts, ideas differ (and are even still evolving) as to what constitutes a good zoning, so the solution preferred by one expert may be unacceptable to another. This implicit "user bias" affects the design and evaluation of flow field zonings. These aspects of flow field zoning serve to make the application of knowledge-based techniques challenging at best.

The plan that was adopted to successfully develop a system for this domain included the following:

1. Develop a model and language to describe the fundamentals of two-dimensional flow field zoning.
2. Debug the model and language through the implementation of an *interactive* knowledge-based system (in which the mechanics and bookkeeping of designing a zoning and generating the boundary curves are automated, but the user supplies the necessary perception, zoning design knowledge, and implicit bias, i.e., the aspects of the problem which are difficult to automate).
3. Increase the level of system automation *incrementally* by replacing the parts previously supplied by the user, one at a time, as described in the sections below.
4. Use existing grid generation capabilities.

In the sections which follow, pragmatic solutions to the problems of perception, design knowledge, and user bias in flow field zoning are described.

SHAPE – A MATTER OF INTERPRETATION

To automatically design a flow field zoning for a given problem, the system must have available qualitative shape and configuration information. This information may be obtained in one of two ways: interactively from the user or automatically through extensive processing of the raw geometric data input by the user. Interactive input was selected for EZGrid for the following reasons: 1) data processing can be time-consuming, even for cases which are simple and obvious to a user; 2) the shape distinctions typically resulting from such processing[12] are finer than necessary for this application; 3) the way an object is described by a user may reflect some bias, and can radically affect a zoning design; and 4) having the user describe the configuration permits the system to share the user's focus on object groupings as a way to decouple portions of the problem where possible (objects which are far apart or separated by one or more other objects may have little influence on each other in terms of how the zoning is designed). The user still provides the perceptual information, but explicitly, in a consistent manner, only for the input geometry, and only during the set-up phase at the beginning of an EZGrid run.

To make the interactive input of shape and configuration descriptions as consistent and as painless as possible, a simple shape and configuration language was developed based on Brady's hypothesis[12] that all shapes have identifiable subshapes. In EZGrid, object shapes are composed of one or more primitive parts which are described by various attributes (orientation, length, width, etc.). Each part has a front end, a back end, a top side, and a bottom side. Ends can be blunt, sharp, or base. Sides can be straight, convex, or concave. Common end/side combinations are given names, such as ellipse (both ends blunt, both sides convex), teardrop (one end blunt, one end sharp, sides any value), and bullet (one end blunt, one end base, both sides convex). Parts fit together via "joins", a simplified version of Brady's join concept. Configurations are described by grouping objects that the user feels have a direct influence on each other, and then providing nearest distances and relations among objects within a grouping and among groupings. Object grouping permits a decomposition of the problem into simpler subproblems, as is discussed in the section on encoding zoning design knowledge.

The qualitative shape and configuration information input by the user has a great impact on the design of a zoning for any given problem. An example of the effect of shape description is shown in Figures 3a-d, in which the geometry, inflow conditions, and user bias are all identical. The only difference lies in how the shape has been described qualitatively. The objects in Figures 3a-d have been described, respectively, as an ellipse, a bullet, a wedge (one end sharp, one end base), and a teardrop. The "ellipse" is surrounded by a single zonal grid with what is known as a C-type topology. A two-zone grid was designed for the "bullet" case. The "wedge" flow field was partitioned into three zones, and the "teardrop" shape was surrounded by two zones, each with an H-type topology. The four zonings are quite different, and demonstrate the large effect that shape description has on flow field zoning design.

INCORPORATION OF USER BIAS

One of the problems associated with flow field zoning is that the experts do not agree on what makes a good zoning. One way to deal with this problem in the development of a knowledge-based system is to establish a standard, or archetypal, set of guidelines. The drawback of this approach for zoning is the possibility that the standard could be totally unacceptable to some experts, and totally acceptable to none. The approach which was chosen, and which has worked well, is to establish a *tunable* zoning archetype. A user can tune the archetype to reflect her or his own bias.

The criteria for designing and evaluating flow field zonings can be categorized as either objective or subjective. Objective criteria include basic guidelines for the type of zoning being used, such as: zones are empty and topologically four-sided, zones abut without gaps rather than overlap, zonal boundaries do not cross each other or the boundaries of the input geometry, and the outer boundary location depends on the physical conditions of the problem. Subjective criteria comprise what is commonly referred to in expert systems parlance as "standard practice." As noted above, zoning practice is not standard. These subjective criteria, which depend on a user's bias, form the basis for the tunable zoning archetype.

The bias an expert user brings to a zoning design problem involves a variety of factors: 1) the particular capabilities of the user's flow solver code, for example, how boundary conditions and singularities are handled, what sort of turbulence model is used, and what effect grid skewness has on the robustness of the code; 2) the user's own experience (often based on a specific flow solver), which determines the user's threshold of tolerance of inaccuracies caused by grid skewness, discontinuities, singularities at body surfaces, and zonal boundary intersections with body surfaces; 3) the user's objectives for the problem at hand – for example, the finest resolution is needed at body surfaces and in the wake in order to calculate drag accurately, or high grid point efficiency is needed to get the greatest accuracy with a small, fixed number of grid points; and 4) aesthetics – "I don't like the look of that discontinuity in the boundary curve."

To incorporate a user's bias into the design and evaluation processes, it must first be parameterized. The archetype is defined as the collection of parameters chosen to capture zoning user bias, and is tuned by the assignment of weights to each parameter. A list of zoning parameters and their possible weight values is found in Table I. Note that the values are all qualitative. The archetype is intended both to guide the design of zonings and to evaluate completed zonings. These qualitative values are used by the design rules to influence a zoning design, as evidenced by the results in Figures 4a-c. The geometry, inflow conditions, and shape description are identical for each case in Figure 4. Different zonings result from different archetype parameter values. The zoning in Figure 4a was generated for a user who places more importance on surface quantities (such as lift and surface drag) than on field quantities or the wake. Figure 4b results when those priorities are reversed. The zonings differ mainly in topology an O-type toplology was automatically selected for 4a and a C-type topology for 4b. The two-zone zoning in Figure 4c shows the EZGrid method of compromising when both surface quantities and wake resolution are considered important.

Numerical values are more convenient than qualitative ones for evaluating and comparing zonings. To translate the qualitative values into numerical ones, the archetype was calibrated in the following manner. Three different configurations of NACA0012 airfoil pairs were selected as test cases: horizontally aligned, vertically aligned, and staggered. For the first two cases, three candidate zonings were generated using EZGrid in interactive mode, and four candidate zonings were generated for the staggered configuration. These three test cases were shown to five flow field zoning experts. The experts were asked first to select weights for the archetype parameters consistent with 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 by preference, presumably consistent with the archetype as they had tuned it. Their orderings for each case are represented by the first number in each column in Table II. For example, expert 1 rated zoning A as second best, zoning B as worst, and zoning C as best for the first case.

The second number in each column of Table II is the ordering given by EZGrid for the same cases. The EZGrid preferences were arrived at by comparing the scores calculated for the candidate zonings. 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 raw penalties for each case did not vary from expert to expert, of course, but the weights which multiply those values come from each expert's tuned archetype, so when the results are summed and compared, the EZGrid preference in each case does vary from expert to expert. The assignment of numerical values to the qualitative weights chosen by the user was adjusted so as to maximize the number of matches between expert and EZGrid. 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, based on this study, that user bias has a measurable effect on flow field zoning design, and that the the proposed zoning archetype is a promising method of evaluating zoning results in the absence of universally accepted criteria.

ENCODING ZONING DESIGN KNOWLEDGE

There are many approaches to design. The process of flow field zoning is modeled as being of the design-by-composition variety. A zoning design can be described by a composition of primitive zoning actions. To automate zoning, it is necessary either to automate the choice of an action at each stage of the design, or to follow a plan (a sequence of actions) constructed or selected at the outset.

Consider zoning design as a search problem in which zoning knowledge is used to restrict the search for an acceptable solution. The space of possible solutions (or search space) consists of all possible combinations of zoning actions. Weak heuristics are bits of knowledge that help to prune the number of possibilities, but do not narrow the range of possibilities sufficiently to eliminate the need for search. The nature of the zoning search space is such that if only weak heuristics are used (involving number of objects, prior actions, containment information, and inflow conditions), the number of possible solutions remains large, and much redundancy in the final zoning design candidates is unavoidable. If strong heuristics are used (involving qualitative shape and configuration information, user bias, zoning design knowledge, and fluid dynamics/CFD knowledge), the search space is narrowed to one or possibly several action sequences. In fact, search is eliminated. For such a search space, the plan option was adopted.

Zoning plans are constructed by assembling predefined subplans. Subplans are sequences of zoning actions applicable to a single grouping of objects. The bulk of the zoning design knowledge in EZGrid is contained directly by these subplans. If certain preconditions about an object grouping are true, a complete subplan is asserted into the database. If the configuration is described by only one grouping, the subplan becomes the plan. If there is more than one grouping, more than one subplan is selected and assembled together to produce a plan. The rules which determine the way in which the subplans are combined contain the remainder of the zoning design knowledge. Figure 5a shows a three-zone viscous zonal grid for a single rotor blade cross-section. Its plan was composed of a single subplan. For the case of a cascade of rotor blades, represented here by just two blades, one could either develop a subplan for a grouping consisting of two vertically-aligned blades, or describe the configuration as having two groupings, each consisting of one blade. In the case of two groupings, the subplan of Figure 5a would be selected twice and combined. The five-zone grid resulting from this assembly is shown in Figure 5b. The result of a somewhat different combination of the same subplan is shown for the rotor-stator pair in Figure 5c. The final example of subplan assembly is the seven-zone zoning of Figure 2, which was generated automatically following a plan assembled from three simpler subplans. The assembly of plans from subplans increases the system's efficiency (it is not always necessary to add new subplans to handle new configurations) and generality (designs for complex problems may be composed of designs for simpler problems). This approach has proved successful for the test problems selected.

CONCLUSIONS

Flow field zoning is an effective solution to the three-dimensional grid generation problem of computational fluid dynamics. To further exploit the potential of zonal approaches, zonal grid generation must be automated. As a stepping stone to an automated three-dimensional zonal grid generation capability, a two-dimensional flow field zoning demonstration system, EZGrid, has been implemented using knowledge-based techniques. This paper focuses on those aspects of flow field zoning which make the use of such techniques challenging: the element of perception, lack of expert consensus, and the modelling of zoning as a design process. In the case of perception, the solution involves the use of a simple shape and configuration language to facilitate the interactive input of such information by the user. The lack of expert consensus is overcome not by imposing a rigid standard, but by the development of a tunable zoning archetype, which affects the zoning design and provides a means of evaluation. The design knowledge essential to the construction of a flow field zoning is encoded in the form of subplans, which, when selected, are assembled into a plan for designing and generating a zoning. A design problem is thus transformed into a simpler selection and assembly problem.

It is clear, then, that the guidelines for choosing a problem to which knowledge-based techniques can be applied are not hard-and-fast rules. The successful implementation of EZGrid demonstrates that even if some aspects of a problem prevent it from being a perfect application for a knowledge-based approach, time and persistence can overcome the problems which arise.

REFERENCES

1. Kutler, P., Steger, J.L., and Bailey, F.R., "Status of Computational Fluid Dynamics in the United States," AIAA paper 87-1135-CP, AIAA 8th Computational Fluid Dynamics Conference, Honolulu, Hawaii, June 9-11, 1987.
2. Chapman, D.R., "Trends and Pacing Items in Computational Aerodynamics," *Lecture Notes in Physics*, Vol. 141, Ed. Reynolds, W.C. and MacCormack, R.W., Springer-Verlag, Berlin, Heidelberg, Germany, 1981.
3. Kutler, P., "A Perspective of Theoretical and Applied Computational Fluid Dynamics," *AIAA Journal*, Vol. 23, 1985.
4. Thompson, J.F. and Steger, J.L., "Three-Dimensional Grid Generation for Complex Configurations – Recent Progress," AGARDograph No. 309, 1988, Yoshihara, H., Ed., North Atlantic Treaty Organization Advisory Group for Aerospace Research and Development.
5. Rai, M.M., "A Conservative Treatment of Zonal Boundaries for Euler Equation Calculations," AIAA Paper 84-0164, AIAA 22nd Aerospace Sciences Meeting, Reno, Nevada, Jan. 9-12, 1984.
6. Hessenius, K.A., and Rai, M.M., "Applications of a Conservative Zonal Scheme to Transient and Geometrically Complex Problems", AIAA paper 84-1532, AIAA 17th Fluid Dynamics, Plasma Dynamics, and Lasers Conference, Snowmass, Colorado, June 25-27, 1984.
7. Genesereth, M.R., Greiner, R., Grinberg, M.R., and Smith, D.E., "The MRS Dictionary," Stanford Heuristic Programming Project Report No. HPP-80-24, Stanford, California, Jan. 1984.
8. Russell, S., "The Compleat Guide to MRS," Stanford University Department of Computer Science Report No. STAN-CS-85-1080, Stanford, California, June 1985.
9. Steinbrenner, J.P., "GRIDGEN2D – Interactive Elliptic Surface Grid Generation – User's Manual," General Dynamics Report No. CFD 063-4-8601, Fort Worth, Texas, June 1986.
10. Hayes-Roth, F., Waterman, D.A., and Lenat, D.B., Eds., *Building Expert Systems*, Addison-Wesley, 1983.
11. Clancey, W.J., "Classification Problem Solving," Proceedings of the AAAI National Conference on Artificial Intelligence, Austin, Texas, Aug. 1984, pp. 49-55.
12. Brady, M. and Haruo, A., "Smoothed Local Symmetries and Their Implementation," *The International Journal of Robotics Research*, Vol.3, No. 3, Fall 1984, pp. 36-61.

Table 1 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

Table 2 Zoning Archetype Calibration Results

GEOMETRY	CANDIDATE ZONING	CANDIDATE ORDERING BY PREFERENCE (HUMAN EXPERT/EZGRID)				
		EXPERT 1	EXPERT 2	EXPERT 3	EXPERT 4	EXPERT 5
HORIZONTAL NACA0012 PAIR	A	2/2	1/1	2/2	1/1	1/1
	B	3/3	2/3	3/3	2/3	3/3
	C	1/1	3/2	1/1	3/2	2/2
STAGGERED NACA0012 PAIR	A	3/3	4/4	4/4	2/2	4/4
	B	1/1	3/3	1/1	1/1	1/3
	C	4/4	2/2	3/3	4/4	3/2
	D	2/2	1/1	2/2	3/3	2/1
VERTICAL NACA0012 PAIR	A	2/2	1/1	1/1	3/2	1/1
	B	1/1	2/2	2/2	2/1	2/2
	C	3/3	3/3	3/3	1/3	3/3

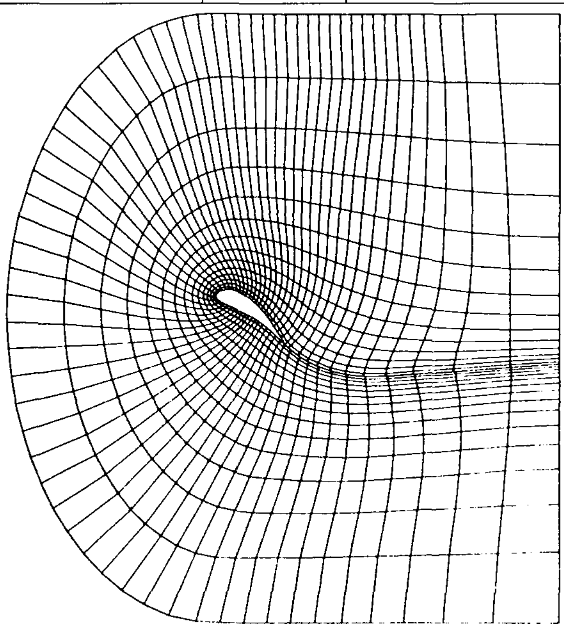


Figure 1 A single-zone grid.

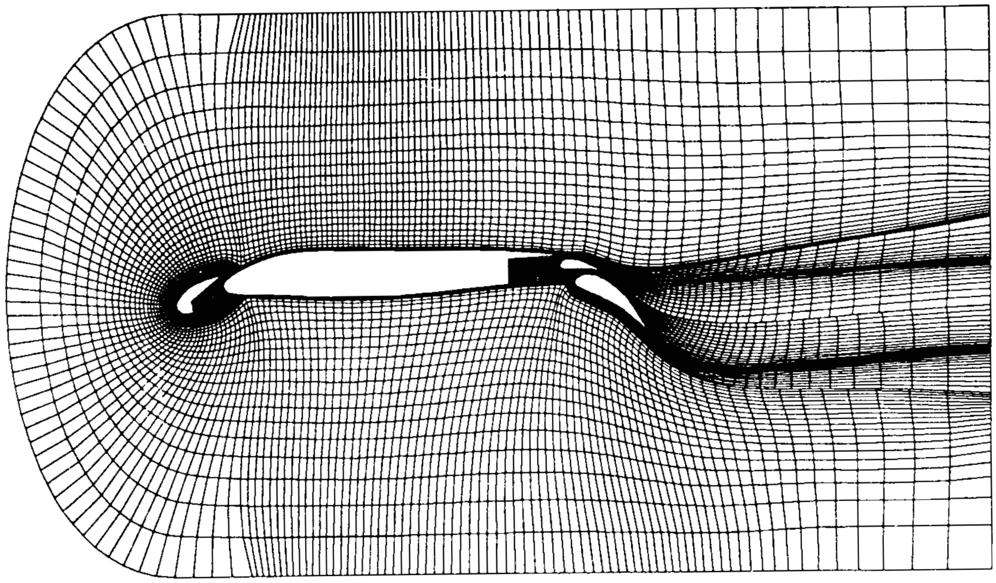
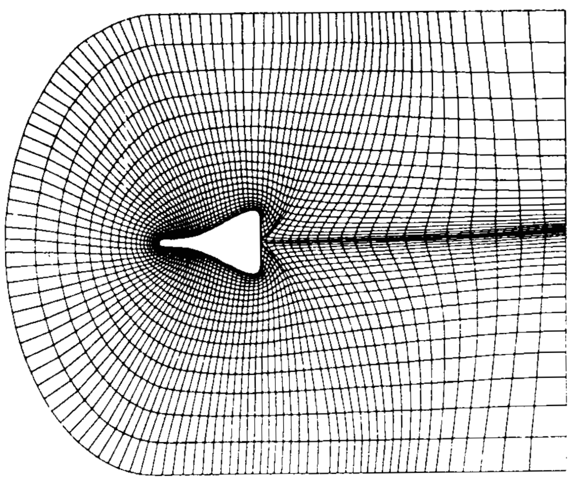
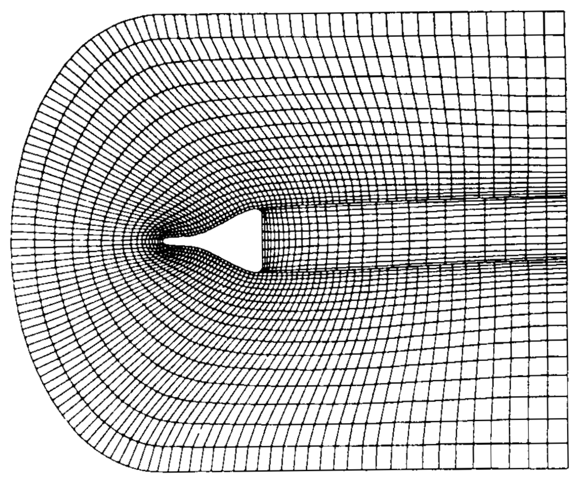


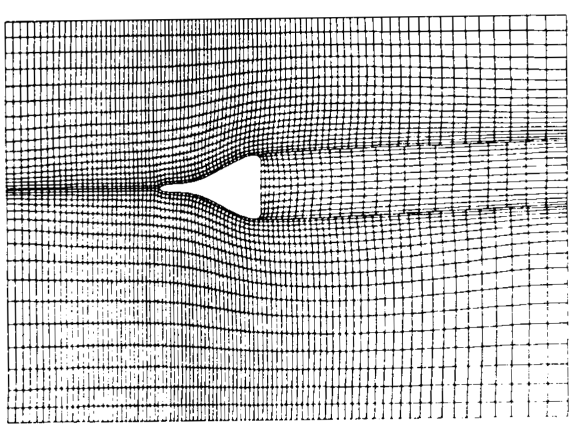
Figure 2 A 7-zone grid.



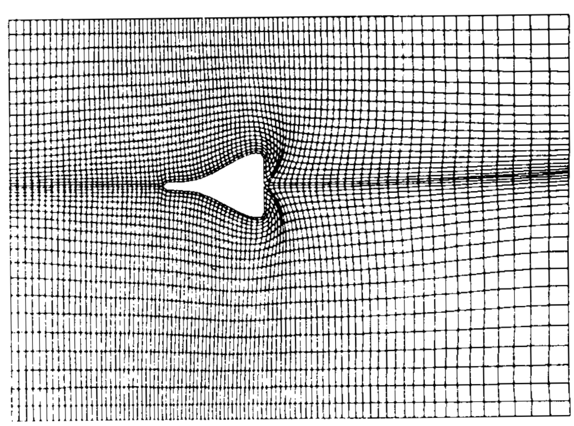
(a) ELLIPSE



(b) BULLET



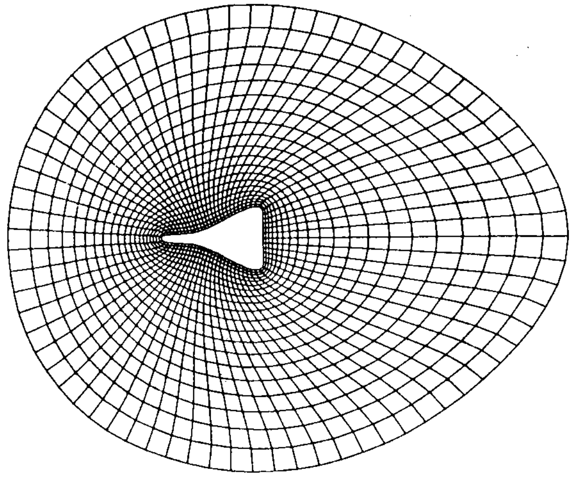
(c) WEDGE



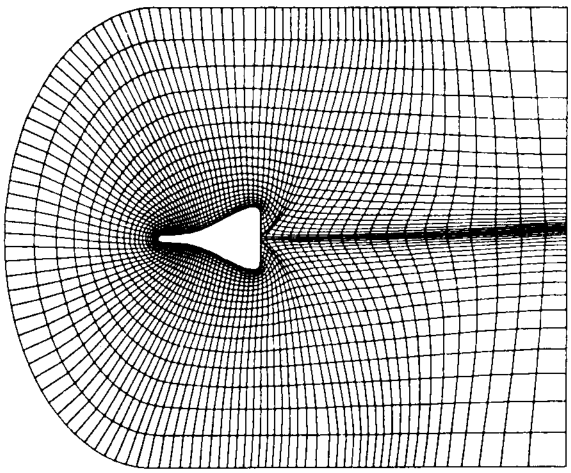
(d) TEARDROP

Figure 3 Effect of qualitative shape description.

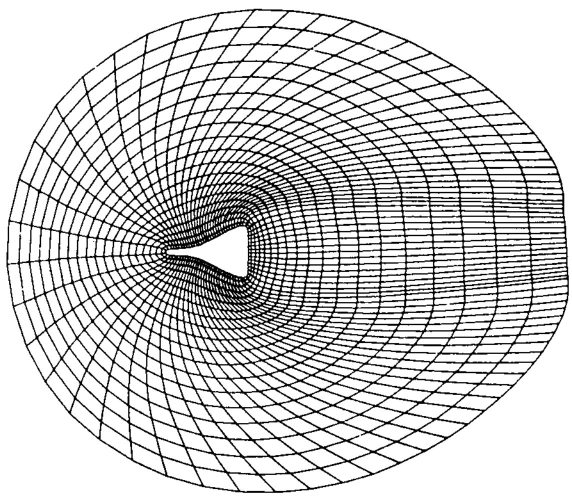
ORIGINAL PAGE IS
OF POOR QUALITY



(a) WAKE – LOW IMPORTANCE
SURFACE QUANTITIES – HIGH IMPORTANCE

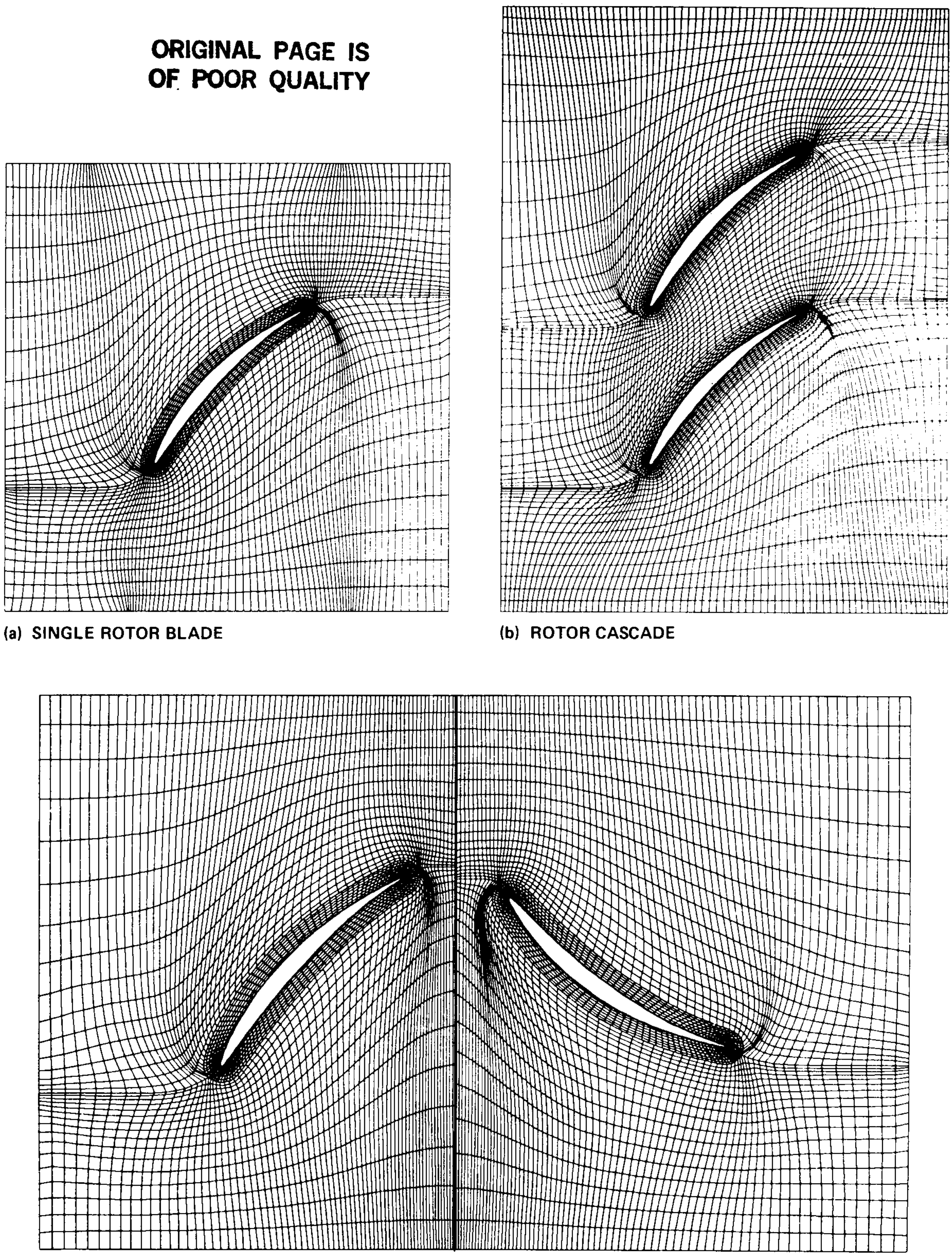


(b) WAKE – HIGH
SURFACE QUANTITIES – LOW



(c) WAKE – HIGH
SURFACE QUANTITIES – HIGH

Figure 4 Effect of user bias.



(a) SINGLE ROTOR BLADE

(b) ROTOR CASCADE

(c) ROTOR-STATOR PAIR

Figure 5 Effect of assembling subplans.