IChemE

0263–8762/98/$10.00+0.00

Trans IChemE, Vol 76, Part A, February 1998

DEVELOPMENTS IN INDUSTRIAL COMPUTATIONAL FLUID DYNAMICS

A. D. GOSMAN

Mechanical Engineering Department, Imperial College of Science Technology & Medicine, London, UK

An overview is presented of the progress over the past decade in the development of CFD as an industrial tool, with particular emphasis on its geometry-handling capabilities. It is demonstrated that the introduction of highly flexible unstructured meshes, together with the ability to dynamically distort, slide, insert and remove selected regions, has enabled CFD to be applied to virtually all types of industrial equipment. When allied with advances in mesh generation, numerical solvers, physics modelling, user interfaces and computer hardware, the end result is a powerful capability which is increasingly being exploited for design purposes.

Keywords: CFD; industrial; unstructured mesh; design

INTRODUCTION

The industrial utilization of Computational Fluid Dynamics has undergone a sea change during the period from the mid 1980s until now. A methodology previously regarded as suitable only for application in 'high-tech' industries by highly-trained specialists has been adopted by a whole range of industries, including automotive, built environment, chemical and food processing and many others. Moreover, it has steadily spread from the research groups into the design and development departments. This paper describes the main developments which have brought about this transformation, with particular emphasis on the geometry-handling aspect of the methodology, which is arguably one of the most important areas of advance.

Background

By way of background, it is useful briefly to recall the state of the art in industrial CFD at the beginning of the period in question. A few general-purpose commercial codes existed, which, with perhaps one exception, had grown out of the academic 'cylinders and boxes' school. By this is meant that their mesh structures were either Cartesian or cylindrical polar and therefore really only suitable for cylindrical or rectilinear geometries. This situation probably arose because these meshes were relatively simple to work with: also such geometries were widely used in academia, for perfectly good reasons, for experimental validation studies.

Unfortunately, industrial plant and equipment is infrequently so regular in shape, and this limitation thus acted as a deterrent to the use of the codes for practical applications. It was sometimes circumvented in simulations by altering the configuration to suit the code, but this was clearly unsatisfactory. Another approach was the use of castellated approximations to the real configuration, as in the example of Figure 1 showing the actual (a) and simulated (b) cooling passage network of an automobile engine, the design optimization of which is of considerable practical interest[1]. However, here serious concerns arise about the flow and heat transfer inaccuracies resulting from the artificially-introduced surface irregularities of the castellated mesh.

The numerical and physics modelling methodologies employed in the codes also had a number of common features: most used finite-volume discretization, with first-order upwind differing (UD) and solution by the iterative SIMPLE algorithm[2]. Turbulence effects were almost universally represented by the well-known $k - \varepsilon$ model in conjunction with wall functions[3]. The codes also catered for heat and mass transfer and chemical reaction, including combustion. In some cases a dispersed two-phase flow capability was provided, using Eulerian-Lagrangian methodology[4].

Calculations were usually performed on mini/mainframe computers, although engineering workstations were beginning to be used. The 'practical' upper limit on mesh size was in the mid to high tens of thousands; and this, in conjunction with the use of UD produced results which are now known often to have been adversely affected by numerical discretization errors, which compromised accuracy. This was another deterrent to more widespread use.

The commercial CFD codes came equipped with menu- or command-driven user interfaces, which endowed them with a degree of 'user friendliness' not usually available in special-purpose in-house codes. Nevertheless, they still required a considerable degree of expertise to learn and apply. This, along with their geometry and accuracy limitations described above, inhibited them from being used as design tools in the same way as computational structural analysis codes. Thus, whereas the latter were already incorporated into the Computer-Aided Engineering (CAE) environment, allowing them, for example, to import CAD geometry data, the CFD codes were almost invariably used in isolation and mainly for research purposes.

Contents

In what follows an overview will be provided of the

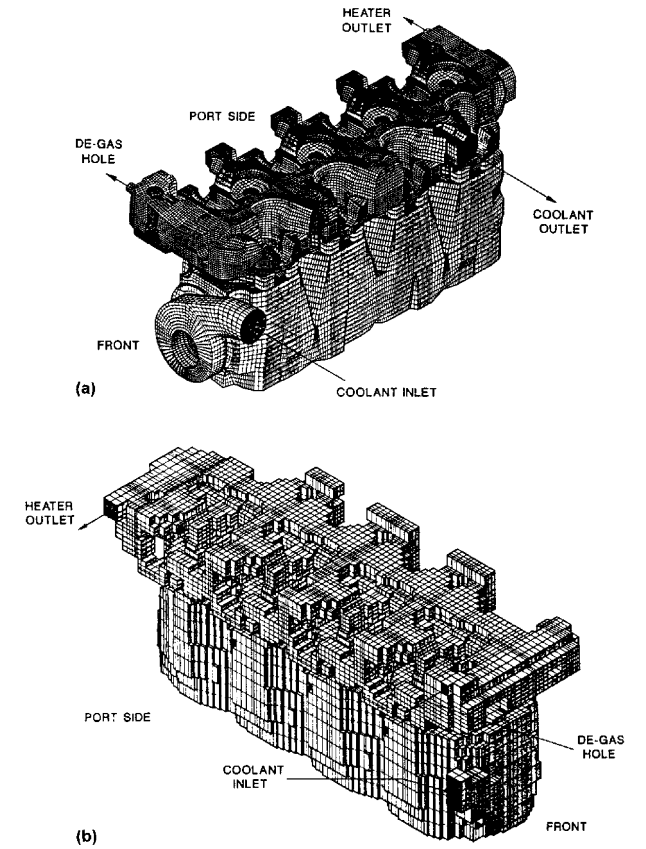

Figure 1. Automobile engine internal cooling circuit. (a) Actual geometry (with fitted multiblock hexahedral mesh); (b) Castellated-mesh approximation.

advances which have lead to the emergence of CFD as a real industrial tool. Pride of place has been given, in Section 2, to those aspects concerned with geometry handling, since these have been the key enablers. The advances here include flexible unstructured meshes with provision for local refinement and movement; and techniques for automatic mesh generation. Section 3 briefly highlights developments in other important areas, notably numerical methodology; turbulence and other physics modelling; and computer hardware, including parallel computers.

GEOMETRY AND MESH

General Body-Fitted Mesh Structures

The first significant step towards dealing with practical configurations in commercial CFD codes using the FVM was the introduction of block-structured, body-fitting curvilinear meshes. Initially only a single block was used, but later codes allowed multiple blocks, arranged in a more or less arbitrary fashion, apart from requiring mesh continuity at block interfaces. Each block consists of a topologically-rectangular mesh, whose elements can be distorted (thereby producing general hexahedral shapes), within practical limits, to fit the physical boundaries of the application. This is illustrated in the examples of Figures 1(a) and 2, the latter showing the internal cooling passages in a radial turbine blade. In both cases, the multiple flow passages are much better fitted by a multiblock configuration than they would with a single block: indeed this would require the presence of inactive cells in those regions of the mesh lying within the solid structure.

However, although the increased flexibility was a significant improvement over what was previously available, multiblock meshes still have two important drawbacks. Firstly, the block structure requirement limits the degree of control available to the user over local mesh fineness and hence in optimizing numerical accuracy; and secondly, the hexahedron is not necessarily always the most suitable shape for geometry-fitting. (A third consideration, to be discussed later, is suitability for automated mesh generation).

It should also be noted that these mesh developments and the additional ones described below were only made

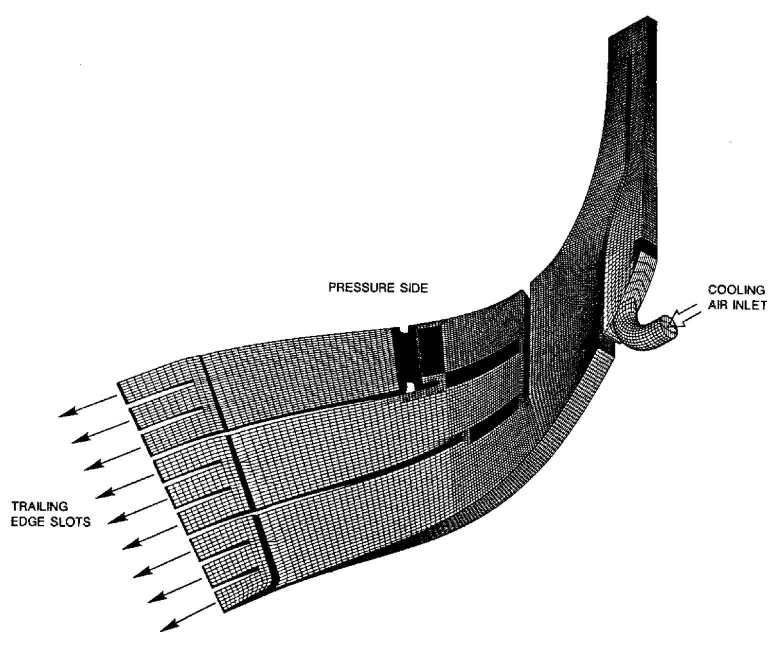

Figure 2. Internal cooling passage of radial rotor blade, fitted with multiblock mesh.

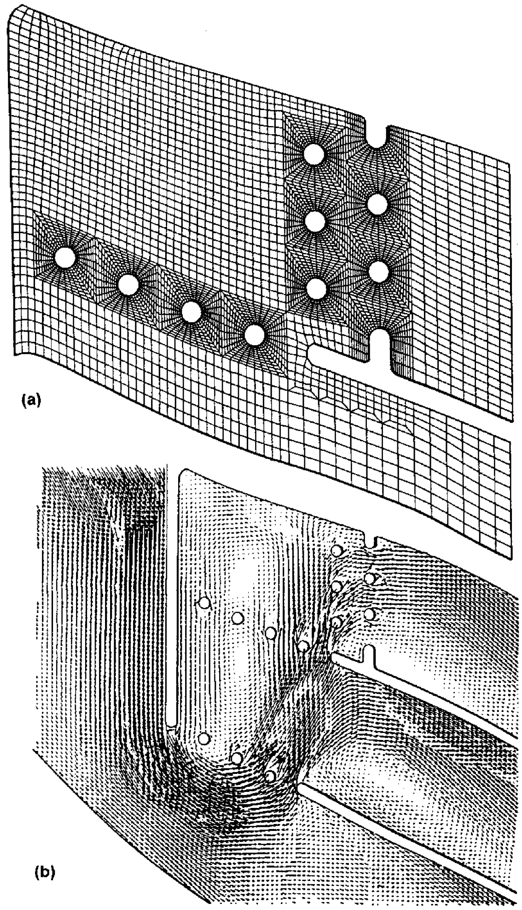

Figure 3. Section of rotor blade passage hexahedral mesh. (a) Unstructured mesh refinement around pin fins and ribs; (b) Predicted flow field.

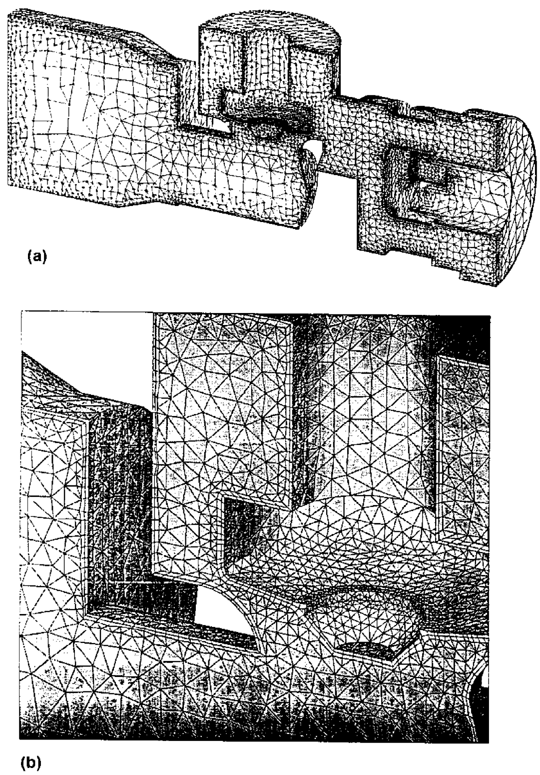

Figure 5. Thermostatic control valve. (a) Overall view of predominantly-tetrahedral hybrid mesh; (b) Close-up showing layered near-wall prismatic cells.

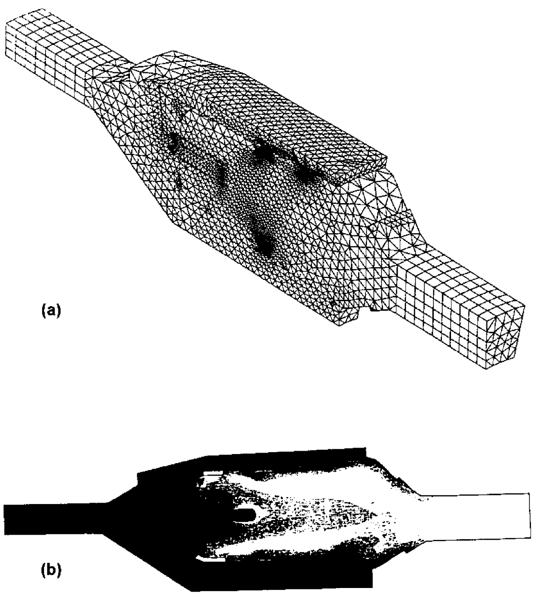

Figure 4. Annular gas turbine combustor sector. (a) View of all-tetrahedral mesh; (b) Predicted flame structure in longitudinal section.

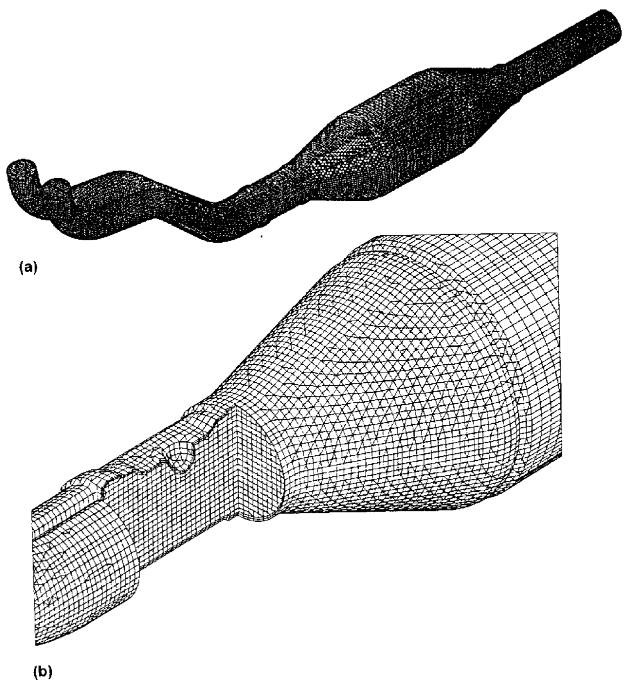

Figure 6. Catalytic converter with trimmed hexahedral hybrid mesh. (a) Overall view of predominantly-hexahedral mesh; (b) Close-up showing trimmed hexahedral ('polyhedral') cells and prismatic near-wall layers.

possible by extensive research into more general numerical solution algorithms. Some indications of what was done in this area will be given Section 3.

Unstructured meshes

The next major step in hexahedral meshing was relaxation of the block structure requirement, by allowing cells to be attached to each other face-to-face in an otherwise arbitrary topology. The benefits of this are illustrated in Figure 3(a), which shows the mesh in a section of the internal flow network of the previous example. Here local refinement has been possible around the fins and ribs inserted to enhance cooling, thus enabling important flow details to be better captured, as seen in Figure 3(b). The resulting 'unstructured' mesh forms require somewhat different programming and solution techniques for the FVM equations (in particular, for the linear equation systems) but these are now well-developed, as will be discussed later.

It should also be noted that at least one commercial code offered at this stage the flexibility to locally collapse edges or faces of the hexahedral cells to form tetrahedra, prisms and other degenerate shapes, which could be advantageous in particularly awkward regions of the geometry such as narrow corners. This capability, only possible with unstructured methodology, anticipated a wider generalization described below.

Recognition that the finite volume method is not inherently restricted to hexahedral cells, nor indeed even to one uniform shape of cell, has been the most recent and important impetus to commercial CFD development. The first-named freedom was first exploited to produce tetrahedral mesh methodology and codes. Figure 4 shows an industrial application utilizing such a mesh, consisting of a sector of an annular gas turbine combustor. Figure 4(a) is a longitudinal section through the mesh and Figure 4(b) shows the predicted flame structure in this plane.

Although the main initial motivation for the move to tetrahedra was ease of mesh generation (see later), this type of mesh is also arguably more geometry-fitting and offers a greater degree of local resolution control than all-hexahedral meshes. Against this, they are less suitable than the latter for near-wall flow calculations, which favour long, thin cells, because tetrahedra have poor numerical properties when stretched in a particular direction. Also, it is often claimed that on an equal cell number basis tetrahedral meshes give lower accuracy than hexahedral ones, although this is controversial. One counter argument is that already mentioned regarding greater flexibility of distribution.

Hybrid meshes

The currently-favoured practice in commercial CFD is to cater for meshes containing a variety of cell shapes, also termed 'hybrid' meshes, so that the topology can be locally chosen to suit the requirements.

One way in which this freedom can be exploited to advantage is illustrated in Figure 5, which depicts a thermostatic flow control valve, meshed mainly with tetrahedra. However, as shown more clearly in the local detail plot (b), in this instance layers of prismatic cells have been introduced adjacent to the walls in order to overcome the above-described drawbacks of tetrahedral meshes in calculations of boundary layers.

A second type of hybrid mesh, shown fitted to the catalytic converter example of Figure 6, consists mainly of hexahedra but, also has additional 'polyhedral' cells produced by trimming off one or more edges or corners of a hexahedron. The ability to work with these more general cell shapes facilitates boundary-fitting (and also, as will be explained later, mesh generation), while retaining the favourable properties of the hexahedral structure for the bulk of the mesh. In addition, as shown in the local expanded view (b), here too it is possible to introduce a layered prismatic mesh (prisms being part of the library of allowable cell shapes) adjacent to the walls, for better resolution of the boundary layers.

Embedded refinement

The meshes shown so far, although differing appreciably in form, share the feature that each cell face has a one to one correspondence with a face of a neighbouring cell. This constraint can be relaxed, to advantage, by the technique of local embedded refinement or cell subdivision, now employed extensively by commercial CFD codes and illustrated in Figure 7. It is most commonly applied to hexahedral meshes (but is not restricted to them) and consists of passing cutting planes or surfaces through pairs of opposing faces of the cells in a selected region to produce two or more smaller cells in each 'parent' cell. The result is a step change in resolution at the boundaries of the refined region.

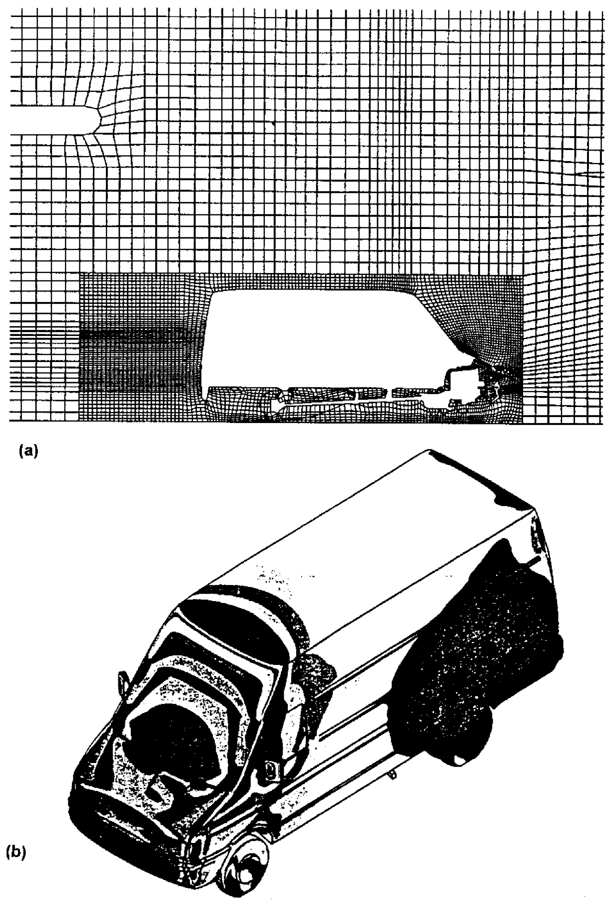

Figure 7. Simulation of van in wind tunnel. (a) Hexahedral mesh with embedded refinement around vehicle; (b) Predicted vehicle surface pressure distribution.

The illustrative example involves simulation of a commercial vehicle in a wind tunnel, where embedded refinement is used to enhance resolution around the vehicle itself, so as to improve the accuracy of prediction of features like the pressure distribution on the surfaces (Figure 7(b)). In fact, it is possible, and now common, to further locally subdivide the mesh in critical regions of the flow which require it.

Embedded refinement is often viewed as producing so-called 'hanging nodes' where the cutting planes intersect the grid lines at the refinement boundaries, requiring special treatment. In fact, the result of such refinement is simply another form of unstructured mesh, which can be routinely handled by unstructured mesh methodology[5].

Arbitrary interfacing

In the mesh embedment treatment, the underlying parent mesh is continuous at the refinement boundaries. A few commercial codes now offer the possibility of interfacing two different regions with *no mesh continuity requirement whatsoever* at their boundaries, i.e. none of the cell vertices on one side need to be coincident with those on the other. This freedom, which allows changes across the interface not only in resolution but also in topology, is achieved by an extension of the practices used for unstructured and embedded meshes.

The 'arbitrary' or 'non-conforming' interface treatment offers substantial benefits, including the possibility of meshing different regions of a problem with the most suitable grid structure and then joining them together. It also facilitates prototyping studies, in which particular parts of equipment and their associated mesh can be interchanged at will. These capabilities are illustrated by the example of Figure 8, showing part of a mesh used in the simulation of the flow in and around an automobile disc brake assembly (the complete mesh covers the entire vehicle). The individual mesh zones, which have been separately generated, can be seen, as can the discontinuities where they meet with their neighbours. This arrangement allows alternative brake designs with their own fitted meshes to be easily inserted.

Dynamic Mesh Features

A significant number of industrial problems involve moving boundaries. The motion may be one of translation or rotation or combinations thereof. Examples include wheeled vehicles in motion, stirred mixing vessels, pumps and compressors. These pose particular challenges to CFD analysis. In the simplest cases, when the motion is tangential to the grid, all that is required is the appropriate boundary condition; but in more general problems, some form of dynamic mesh adjustment is required. Some current practices employed in commercial CFD codes will now be outlined.

Moving meshes

Motion which gives rise to significant changes in the solution domain volume can be catered for by allowing the computational mesh to dynamically distort, ideally in an 'arbitrary' fashion: this is sometimes referred to as the Eulerian-Langrangian approach. The mesh motion can either be prescribed beforehand or calculated during the simulation, as appropriate to the application. This capability is often used in simulations of the in-cylinder flow and combustion processes in reciprocating engines[6,7], as

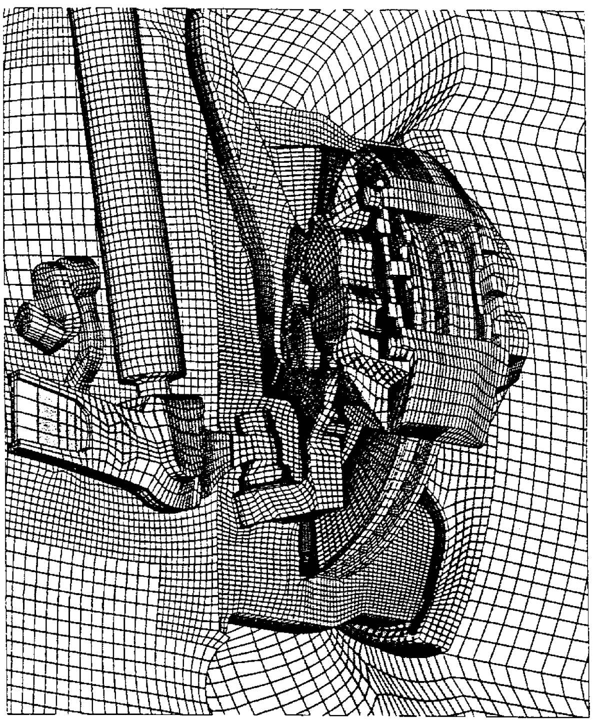

Figure 8. Block-structured hexahedral mesh around automobile disk brake assembly, with arbitrary interfacing between blocks, and embedded refinement.

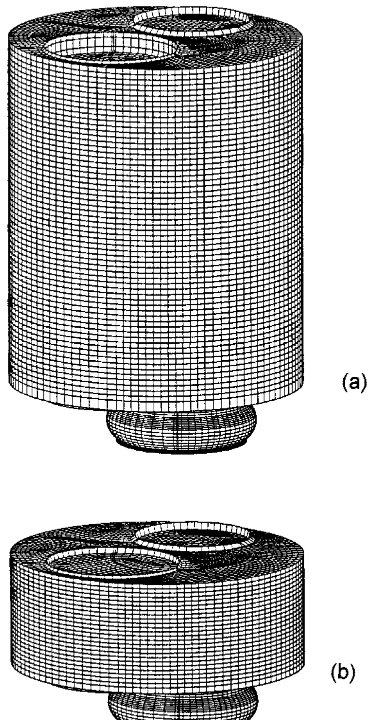

Figure 9. Moving mesh (hexahedral) simulation of flow in automobile Diesel engine, including cell-layer removal/addition during compression/expansion strokes. (a) Bottom stroke; (b) Mid-stroke.

illustrated by the example of Figure 9. This shows the mesh at two different stages in the engine cycle, during which it is compressed by the upwards-moving piston.

A pitfall of mesh motion is the possibility of creating badly-distorted cells. In engine applications, the distortion can take the form of excessively high aspect ratios, due to the unidirectional compression. This can be avoided by providing for dynamic deletion of individual cells or groups thereof at appropriate stages (the possibility of dynamic insertion during the expansion stroke is also catered for). Inspection of the two meshes in Figure 9 will reveal that this practice was followed in this simulation.

Sliding meshes

Some moving-boundary problems cannot be handled by mesh distortion alone. Rotating machinery problems are of this kind, as exemplified by the stirred mixing vessel shown in Figure 10. The rotation of the impeller blades relative to the fixed walls and baffles of the vessel would rapidly cause a moving mesh attached at these surfaces to be distorted beyond acceptable limits.

A solution to this problem is to allow selected portions of the mesh in Figure 10(a) to slide relative to each other at a common interface[8–10]: in this case it is a cylindrical surface lying between the tips of the blades and baffles. The inner darker-shaded mesh simply rotates with the impeller, while the outer one remains stationary. Views of the predicted time-varying flow field at a particular instant are shown in plots (b) and (c). The CFD methodology required to accommodate the sliding interface is an extension of the arbitrary interfacing procedure for static meshes described earlier.

Multiple rotational frames (MRF)

Sliding-mesh calculations are inherently unsteady, as are the flows in the applications they are used for. However such problems sometimes contain regions of nearly-stationary flow. This may occur for example in the mixing vessel, in the region between the blades and baffles, when they are sufficiently separated. This behaviour can be exploited to advantage, to reduce complication and expense, by introducing the concept of multiple rotational reference frames[11]. In this, local rotation of the mesh is simulated, rather than directly invoked, by inserting appropriate rotational body force terms in the momentum equations for that part of the mesh; and making suitable transformations in the CFD calculations at the interface between it and the stationary region. The entire calculation can then proceed in a steady-state fashion on a totally static mesh, with considerably reduced computing overheads, which is very beneficial for design studies[12].

The MRF treatment is formally exact when the rotational

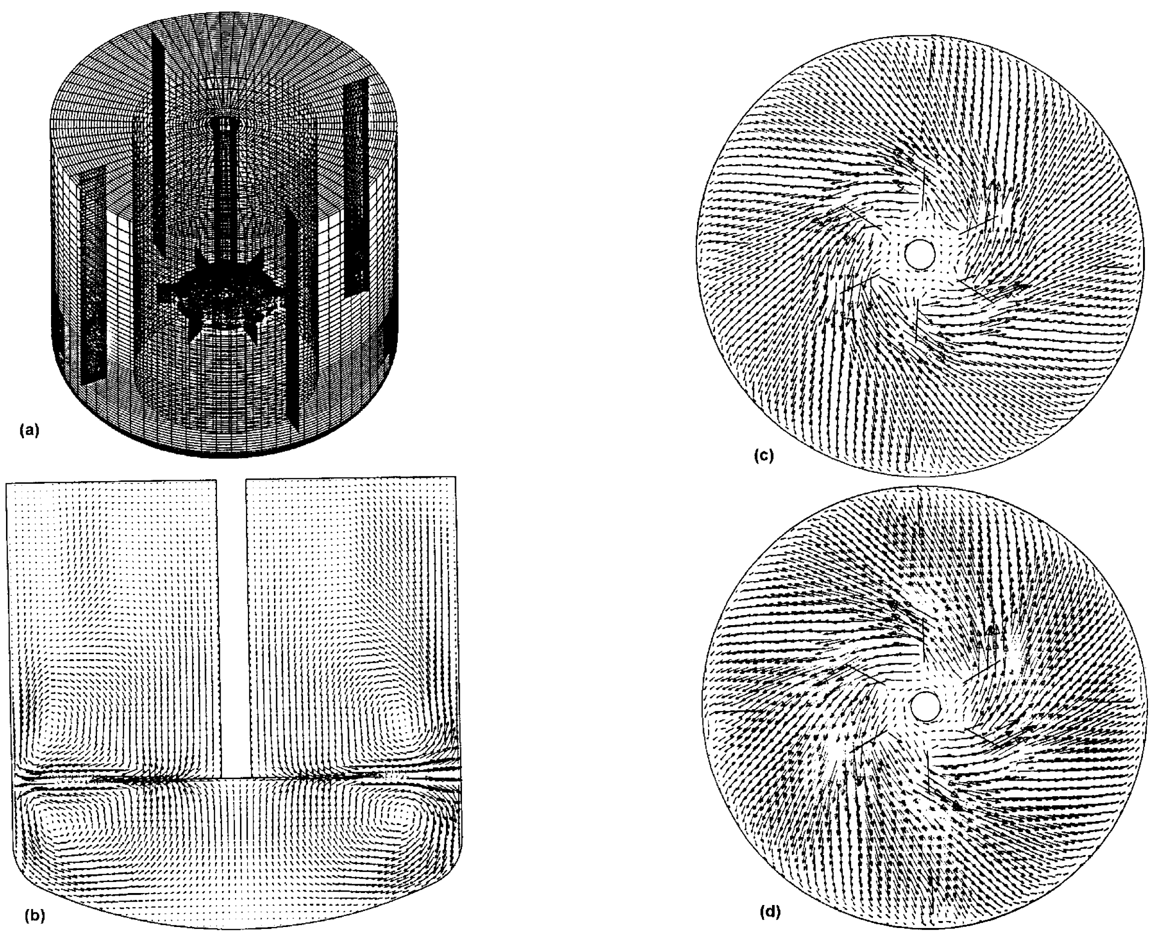

Figure 10. Impeller-stirred mixing vessel. (a) Geometry and sliding mesh—inner part rotates with impeller, outer portion remains stationary; (b) Predicted instantaneous flow pattern in vertical section; (c) Predicted instantaneous flow pattern in horizontal section through impeller; (d) Multiple Reference Plane prediction corresponding to (c).

frame interface is indeed in a region of steady-state flow. This is the case in the present mixing vessel example, as is shown by the close correspondence between the sliding-mesh velocity field of Figure 10(c) and the corresponding MRF velocity predictions of Figure 10(d). However, even when this requirement is not met, it has been found that the MRF treatment can give useful results. An example is the application in Figure 11 to the simulation of the flow in a centrifugal pump, where the spacing between the impeller and casing is small. Of course, in such situations the sliding mesh treatment can generally be expected to give greater accuracy.

Automated Mesh Generation

Hand in hand with the work on mesh flexibility have been efforts to automate the mesh-generation process. Indeed, the latter activity has tended to be driven by the former, because as the capability to simulate complex industrial configurations has developed, so has the requirement to quickly fit them with the appropriate meshes. Even with the increased mesh flexibility this can be a time-consuming activity if done manually; hence the incentive for automation. (In fact, a substantial proportion of the total mesh preparation time is currently often spent in obtaining and cleaning up the surface geometry data prior to mesh generation, but these are separate matters which will not be discussed here).

The often-stated 'ultimate' goal is fully-automatic meshing, in which, given an appropriate closed-surface model of the geometry, a 'good quality' mesh is automatically generated within it. It is true that certain geometrical quality measures like aspect ratio and degree of non-orthogonality can be imposed as part of the generation process, but it should be noted that no a priori automated procedure possesses the fluid dynamics knowledge that humans often drawn on when producing meshes manually. Thus, for this reason alone, even 'push-button' procedures may need a degree of manual intervention to obtain more optimal meshes. Automatic mesh optimization requires solution adaptivity, an exciting upcoming development in industrial CFD[5].

Various automated mesh generation procedures have been developed, differing according to the mesh structures they produce, the methodology they use and the degree of automation they achieve. The earliest procedures were for block-structured hexahedral meshes, as in the examples of Figures 1 and 2. These tend to be semi-automatic, for two main reasons. Firstly, human intervention is usually required to decide on the block layout, especially in complex geometries; and secondly, it is often also necessary to locally 'repair' cells which are excessively distorted, usually due to the difficulties of fitting this type of mesh to the geometry.

It is not difficult to see that the necessity for the above manual actions could have been reduced or eliminated if the rigid requirements of block-structuring and/or uniform cell type had been relaxed, as is now the case with unstructured hybrid-mesh CFD methodology. This is one of the motives for developing the trimmed-hexahedral hybrid mesh capability illustrated in Figure 6. This mesh has been fitted to the geometry simply by allowing the surfaces of the latter to trim the corners and/or edges of those hexahedra which overlay them, thereby producing the (allowable) polyhedral cells. Added to this is the ability to introduce layered meshes near walls by first meshing to a shrunken virtual surface and then filling the gap between it and the real surface with prismatic cells.

Prior to the development of hybrid meshes, the closest approaches to fully-automated mesh generation have been for all-tetrahedral meshes. A number of different techniques for these have been developed and implemented in commercial packages. However, their use in commercial CFD has until recently been inhibited by concerns about the accuracy achievable relative to hexahedral meshes, especially in near-wall regions. The desire for quick turnaround has often been allowed to over-ride the accuracy concerns (which can in any event be addressed to some extent by employing finer meshes). Fortunately, they have now been allayed to a considerable extent by the introduction of near-wall prismatic layers (Figure 5), once again made possible by the development of hybrid-mesh CFD methodology.

This completes the review of geometry-handling and mesh generation developments. As a final comment, it will be noted that the most advanced commercial codes now offer most, if not all of the features which have been described, which represents a very large improvement in geometry-handling capability over the situation at the beginning of period reviewed. Indeed, it can now be claimed that virtually all the geometrical complexities of industrial applications can be addressed with the current methodology.

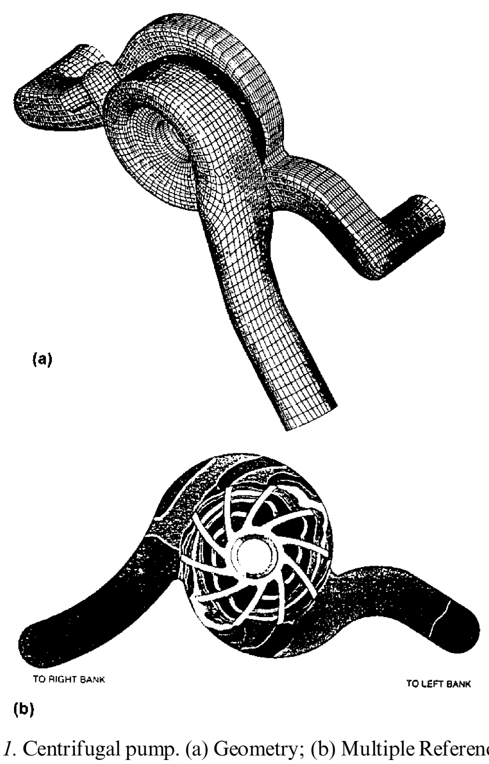

Figure 11. Centrifugal pump. (a) Geometry; (b) Multiple Reference Frame prediction of surface pressure distribution.

OTHER DEVELOPMENTS

Some brief comments will here be made about progress in other important areas of industrial CFD development.

Numerical Algorithms

As already noted, the introduction of the mesh flexibilities described in the previous section was made possible by developments in numerical solution methodology. One of the key advances was the ability to work with Cartesian velocity components, using techniques originally proposed by Rhie and Chow[13] and later improved and elaborated by others, e.g.[14]. Another important contributor was the emergence of efficient iterative solvers[15] for the sparse unstructured matrices characteristic of the discretized flow equations for these meshes. In recent times, multigrid techniques[16] have found favour due to the additional advantages that they sometimes offer. Last but not least, the development of higher-order discretization practices like 'Total Variation Diminishing' (TVD) schemes, e.g.[17], which provide higher accuracy without generating unphysical 'wiggles' have reduced the degree of mesh sensitivity as compared with the much-used UD discretization practice.

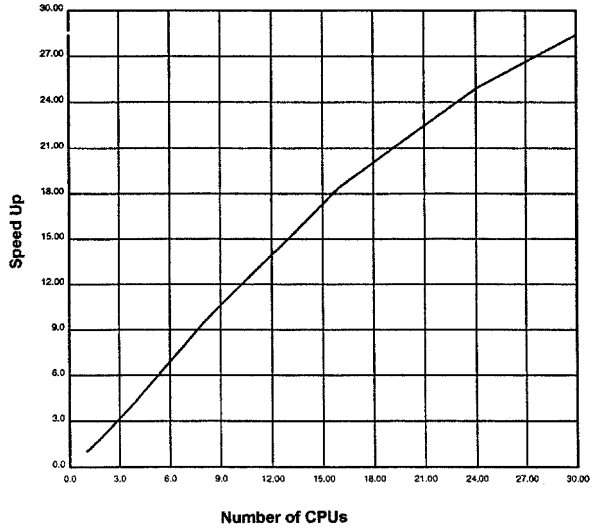

Figure 12. Engine coolant flow simulation speedup using parallel computer.

Parallel Computing

While the aforementioned numerical methodology developments have contributed to increased speed and accuracy of calculation, the speed improvements have tended to be overshadowed by advances in the performance and cost/performance ratios of computers. These have been particularly dramatic in the case of parallel computers, due to several factors. Firstly, most machines are now based on relatively low-cost, high-performance 'commodity' RISC processors and standard programming languages and operating systems. Secondly, the latter software changes have enabled CFD code developers to more readily adapt and port their codes onto these machines. Finally, these machines are much more affordable than vector supercomputers, which were previously the only high-performance option.

The advantages of parallelism are illustrated in Figure 12, showing the speedup (defined as the ratio of single to multiple processor computing times) obtained for an engine cooling application akin to that illustrated in Figure 1(a) as a function of the number of processors used. Evidently it is possible to obtain near-ideal linear behaviour in this case for the number of processors investigated, implying that turn-around time can be reduced by up to nearly thirty-fold. Lesser, but still useful, gains can also be obtained by using workstation networks as virtual parallel machines. In either case, the dramatic reductions in computing time[18,19], allow CFD to have a greater impact on the design/development process, especially when allied with automated mesh generation.

Physics Modelling

There has been a substantial increase in the range of thermofluids phenomena which can be modelled with industrial CFD codes: non-Newtonian fluids; chemical reaction, including combustion (various modes); multiphase (gas/liquid/solid) flows with phase transformation: free surface flows; thermal radiation in transparent and participating media; fluid/solid interactions (thermal, structural) . . . the list is long and continues to grow. The accuracy of the modelling is variable and needs substantial improvements in many areas, but at least the framework is broadly in place to accept them.

In many applications, however, turbulence and its modelling continue to be the principal concern, on the basis that sufficiently accurate prediction of the flow field is nearly always a prerequisite for good predictions of other thermofluids processes. The good news is that as the accuracy of the numerical methodology has improved, in general so has the overall quality of prediction of many turbulent flows, even with the much-used and frequently-maligned $k - \varepsilon$ turbulence model. However, although the latter has proven to be adequate in many instances, this is not true for certain important applications, notably those involving, and being highly sensitive to, flow separation. The bad news is that despite intensive research, leading to many 'improved' and new models over the past few decades, progress in resolving the deficiencies has been slow. Other approaches, notably Large Eddy Simulation (LES) are now becoming practically feasible, due to computer advances, and it will be interesting to see whether these will displace the current models based on Reynolds-averaging.

Integration into CAE

As already noted, at the beginning of the period covered CFD was used primarily as a research tool. For this and other reasons cited, CFD codes tended to be used in isolation from other design and analysis software. By contrast, structural analysis codes had by this stage already begun to be integrated into company CAE environments, importing their geometry and other data directly from CAD systems and exporting results in standard forms for various uses.

CAE integration of CFD is now well under way (see, for example, references 20–22), thanks to the developments already described, and also to improvements to the pre/post-processing facilities in the commercial CFD codes. The latter include the introduction of graphical user interfaces; improved data visualisation facilities; and provision for import and export of data in recognised standard formats to and from CAD systems, visualization packages, and other analysis software. CFD is thus at last becoming part of the repertoire of analysis tools available to the designer. However, it will for some time to come still require a degree of specialized knowledge and expertise to use it properly.

CONCLUSIONS

The main conclusions to be drawn from this review of industrial CFD development over the past decade are as follows:

1. Progress has been substantial, to the extent that CFD is now a recognized tool for analysis and design.
2. The key area of development has been geometry-handling, now greatly improved with unstructured-mesh methodology, allied with the abilities to dynamically distort, slide, insert and remove selected regions and rapidly generate meshes.
3. Important advances have also been made in numerical solvers, physics modelling, user interfaces, CAE integration and computer hardware. Concerning the last-named, the ability of some commercial CFD codes to effectively harness the power of the new generation of parallel computers is of particular significance.
4. There is nevertheless still need and scope for further improvement, especially in the accuracy and capabilities of the flow physics modelling, starting with turbulence and the phenomena which it influences.

REFERENCES

1. Bauer, W., Ehrenreich, H. and Reister, H., 1995. Design of cooling systems with computer simulation and underhood flow analysis using CFD, SAE/IMechE Second Vehicle Thermal Management Systems Conf (VTMS2), London, England, Paper C496/042, pp. 499–511.
2. Patankar, S. V. and Spalding, D. B., 1972, A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows, *Int J Heat Mass Transfer* 15: 1787.
3. Launder, B. E. and Spalding, D. B., 1974, The numerical computation of turbulent flows, *Comp Meth Appl Mech & Eng*, 3: 269.
4. Gosman, A. D. and Ioannides, S. I., 1983, Aspects of computer simulation of liquid-fuelled combustors, *AIAA J of Energy*, 7(6): 482–490.
5. Muzaferija, S. and Gosman, A. D., 1997, An adaptive finite-volume discretisation technique for unstructured grids, (to appear in *J Computational Physics*).
6. Adamson, B., Gosman, A. D., Maroney, C. H., Nasseri, B. and Theodoropoulos, T., 1990, A new unstructured-mesh method for flow prediction in internal combustion engines, *Proc COMODIA '90, Japan*, September 1990.
7. Bo, T., Clerides, D., Gosman, A. D. and Theodossopoulos, P., 1997, Prediction of the flow and spray processes in an automobile DI diesel engine, *SAE Int Congress, Detroit*, Paper No 978802.
8. Luo, J. Y., Gosman, A. D., Issa, R. I., Middleton, J. C. and Fitzgerald, M. K., 1993, Full flow field computation of mixing in baffled stirred vessels, *The 1993 IChemE Research Event, Birmingham, January 1993.*
9. Tabor, G., Gosman, A. D. and Issa, R. I., 1996, Numerical simulation of the flow in a mixing vessel stirred by a Rushton turbine, *Proc Fluid Mixing V Conf, Bradford. 1996 IChem E Symp Series N. 140*, pp 25–34.
10. Lee, K. C., Ng, K., Yianneskis, M., Lange, F. and Sanatian, R., 1996, Sliding mesh predictions of the flows around Rushton impellers. *Proc Fluid Mixing Conf, Bradford*, 1996.
11. Luo, J. Y., Gosman, A. D. and Issa, R. I., *1994, I ChemE Symp Series No. 140*, pp. 47–58. Prediction of impeller induced flows in mixing vessels using multiple frames of reference *8th Europ Conf on Mixing, Mixing 8, Cambridge, UK, IChemE Symp Series No. 136*, pp. 549–556.
12. van der Geest, R. S. H., Klahn, J. K. and Agterof, W. G. M., 1995, CFD simulation of the influence of impeller spacings on flow fields and mixing times of stirred vessels, *Chemical Engineering Chemputers Europe II Conf, Noordwijk, The Netherlands, 10–12 October 1995.*
13. Rhie, C. M. and Chow, W. L., 1982, A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation, *AIAA-82-0998.*
14. Demirdzic, I., Gosman, A. D., Issa, R. I. and Peric, M., 1987, A calculation procedure for turbulent flow in complex geometries *Comput & Fluids*, 15(3): 251–273.
15. Meijerink, J. A. and Van der Horst, H. A., 1981, Guidelines for the usage of incomplete decompositions in solving sets of linear equations as they occur in practical problems, *J Comp Phys*, 44: 134.
16. Peric, M., Ruger, M. and Scheuerer, G., 1989, A finite volume multigrid method for calculating turbulent flows, *7th Symp Turbulent Shear Flows, Standford University*, p 7.3.1.
17. Zhu, J., 1992, On the higher-order bounded discretisation schemes for finite volume computations of incompressible flows, *Comp Methods Appl Mech Eng*, 98: 345–360.
18. Jones, D., Demirdzic, I., Krishna, R. and Robinson, D., 1995, Use of parallel CFD for demanding chemical process applications, *Chemical Engineering Chemputers Europe II Conf, Noordwijk, The Netherlands, 10–12 October 1995.*
19. Behling, S., Robinson, D., Bauer, W., 1996, Recent experience with STAR-HPC on the CRAY T3E, in *High Performance Computing in Automotive Design, Engineering, and Manufacturing* (Ed. M. Sheh), *Proc 3rd Int Conf on High Performance Computing in the Automotive Industry, Cray Research, Paris, France, 7–10 October 1996*, pp. 377–384.
20. Lisbona, M. G. and Vafidis, C., 1997, Adoption of CFD in the engine design process: present and future, *5th Int ATA Conf on The Virtual Automobile and the Role of Experimentation, Florence, Italy, 26–28 February 1997*, Paper 97A1012, pp 85–106.
21. Novara, A. and Zucchelli, A., 1997, Computational fluid dynamics analysis current applications at design stage in IVECO *5th Int ATA Conf on The Virtual Automobile and the Role of Experimentation, Florence, Italy, 26–28 February 1997*, Paper 97A1014, pp. 119–134.
22. Blümcke, E. and Nefischer, P., 1995, Improved engine cooling systems using calculation methods, *SAE/IMechE 2nd Vehicle Thermal Management Systems Conf (VTMS2), London, England*, Paper C496/013.

ACKNOWLEDGEMENTS

The helpful provision of industrial examples by Computational Dynamics Ltd and Adapco and their client companies is acknowledged.

ADDRESS

Correspondence concerning this paper should be addressed to Professor A. D. Gosman, Mechanical Engineering Department, Imperial College, London SW7 2BX, UK.

This paper was presented at the 5th UK National Heat Transfer Conference, held at Imperial College London 17–18 September 1997.