Importance of Algorithm Simplicity for Complex Flow Simulations on Peta/Exa-Flops Computers

Kazuhiro Nakahashi

Department of Aerospace Engineering, Tohoku University, Sendai 980-8579, JAPAN
(*Tel: +81-22-795-6978; e-mail:naka@ad.mech.tohoku.ac.jp*)

1. WILL CFD TAKE OVER WIND TUNNELS?

More than 20 years ago, I heard an elderly physicist in fluid dynamics say that it was as if CFD were just surging in. Other scientists of the day said that with the development of CFD, wind tunnels would eventually become redundant.

Impressive progress in CFD has been made during the last three decades. In the early stage, one of the main targets of CFD for aeronautical fields was to compute flow around airfoils and wings accurately and quickly. Body-fitted-coordinate grids, commonly known as structured grids, were used in those days (Fig. 1).

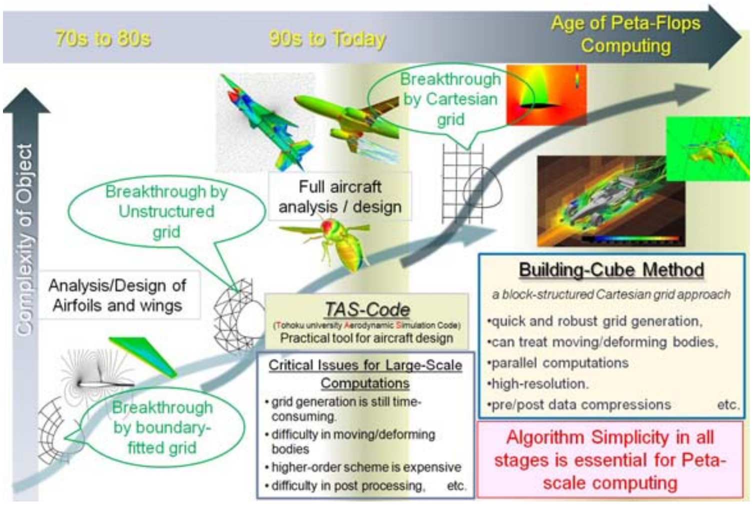

Fig.1: *Progress of Aeronautical CFD*

From the late eighty's, the target was moved to analyzing full aircraft configurations [1]. This spawned a surge of activities in the area of unstructured grid CFDs. Unstructured grids provide considerable flexibility in tackling complex geometries [2]. CFD has become an indispensable tool for analyzing and designing aircrafts.

So, is CFD taking over the wind tunnels as predicted twenty years ago?

Today, Reynolds-averaged Navier-Stokes (RANS) computations can accurately predict lift and drag coefficients of a full aircraft configuration [3]. It is, however, still quantitatively not reliable for high-alpha conditions where flow separates. Boundary layer transition is another cause of inaccuracy. These are mainly due to the incompleteness of physical models used in RANS simulations. Large Eddy Simulation (LES) and Direct Numerical Simulation (DNS) are expected to reduce the physical model dependencies. But we have to wait for the further progress of computers for the use of those large-scale computations in engineering purposes.

For the time being, the wind tunnel is the central player and CFD plays a subordinate part in aircraft developments.

2. RAPID PROGRESS OF COMPUTERS

The past CFD progress has been highly supported by the improvements of computer performance. The latest Top500 Supercomputers Sites [4] tell us that the performance improvement of computers has reached a factor of 1000 in the last 10 years as shown in Fig. 2. Increase in the number of processors in a system in addition to the degree of integration contributes to this rapid progress.

With a simple extrapolation of Fig. 2, we can expect to use PetaFlops computers soon and ExaFlops ones in near future. This will accelerate the use of 3D RANS computations for the aerodynamic analysis and design of entire airplanes. DNS which does not use any physical models may also be used for engineering analysis of wings. In the not very far future, CFD could take over wind tunnels.

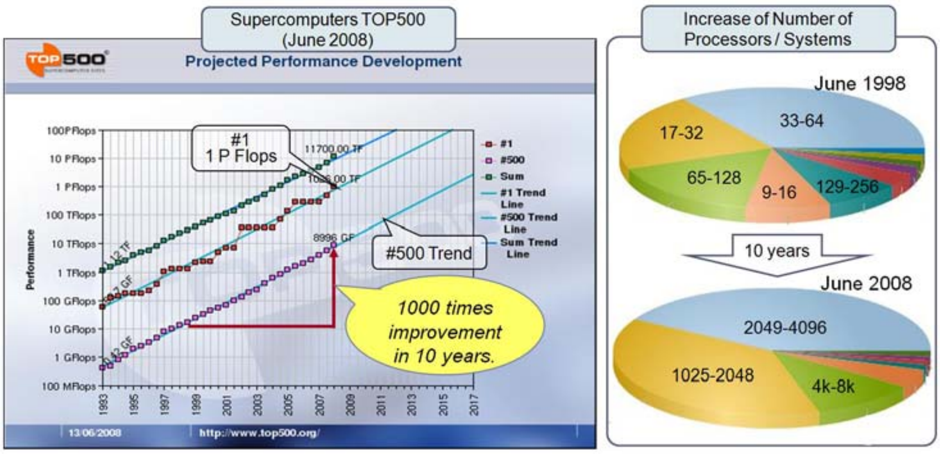

Fig.2: *Performance development in Top500 Super-computers* [4].

3. DEMANDS FOR NEXT-GENERATION CFD

So, is it enough for us as CFD researchers to just wait for the progress of computers? Probably it is not. Let's consider demands for next-generation CFD on PetaFlops computers.

1. *Easy and quick grid generation around complex geometries,*
2. *Easy adaptation of local resolution to local flow characteristic length,*
3. *Easy implementation of spatially higher-order schemes,*
4. *Easy massively-parallel computations,*
5. *Easy post processing for huge data output,*
6. *Algorithm simplicity for software maintenance and update.*

Unstructured grid CFD is a qualified candidate for the demands 1 and 2 as compared to structured grid CFD. However, an implementation of higher-order schemes on unstructured grids is not easy. Post processing of huge data output may also become another bottleneck due to irregularity of the data structure.

Recently, studies of Cartesian grid method were renewed in the CFD community, because of the several advantages such as rapid grid generation, easy higher-order extension, and simple data structure for easy post processing. This is another candidate for the next-generation CFD.

4. BUILDING-CUBE METHOD

A drawback of uniform Cartesian grid is the difficulty of changing the mesh size locally. This is critical, especially for airfoil/wing computations, where an extremely large difference in characteristic flow lengths exists between boundary layer regions and far fields. Accurate representation of curved boundaries by Cartesian meshes is another issue.

A variant of the Cartesian grid method is to use the adaptive mesh refinement in space and cut cells or the immersed boundary method on the wall boundaries. However, introduction of irregular subdivisions and cells into Cartesian grids complicate the algorithm for higher-order schemes. The advantages of the Cartesian mesh over the unstructured grid, such as simplicity and less memory requirement, disappear.

The present author proposed a Cartesian grid based approach, named Building-Cube method [5]. Basic strategies employed here are; (a) zoning of a flow field by cubes of various sizes to adapt the mesh size to local flow characteristic length, (b) uniform Cartesian mesh in each cube for easy implementation of higher-order schemes, (c) same grid size in all cubes for easy parallel computations, (d) staircase representation of wall boundaries for algorithm simplicity.

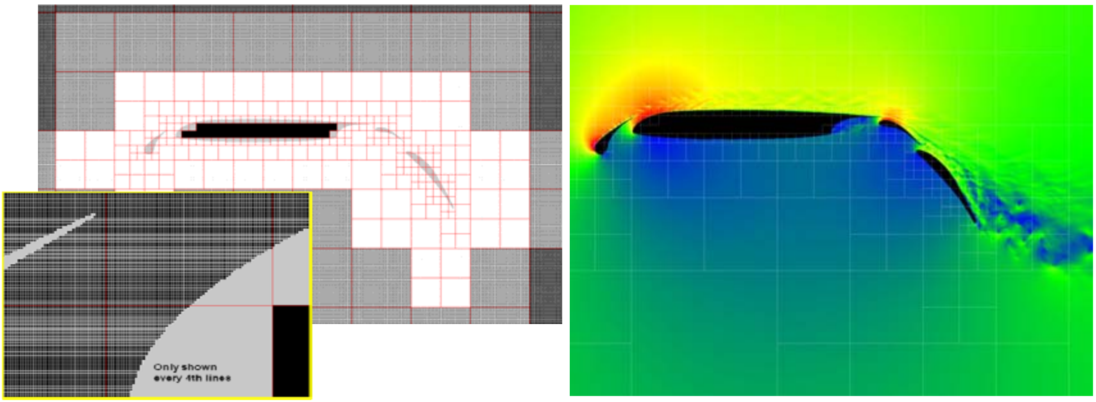

Fig. 3: *BCM mesh (481 cubes with 256x256 Cartesian mesh in each cube) and the instantaneous Mach distributions for a four-element airfoil; Re*=2.83x10^6, M_∞=0.201, $\alpha = 8.16°$

It is similar to a block-structured uniform Cartesian mesh approach [6], but unifying the block shape to a cube simplifies the mesh generation [7] and the domain decomposition of a computational field around complex geometry. Equality of computational cost among all cubes significantly simplifies the massively parallel computations as shown in Fig. 4. It also enables us to introduce data compression techniques for pre and post processing of huge data [5].

A staircase representation of curved wall boundaries requires very small grid spacing to keep the geometrical accuracy as shown in Fig. 3. But the flexibility of geometrical treatments obtained by it will be a strong advantage for complex geometries and their shape optimizations. Although computation using high-density Cartesian mesh is still far from practical use because of the computational time, it will be resolved with a progress of computers.

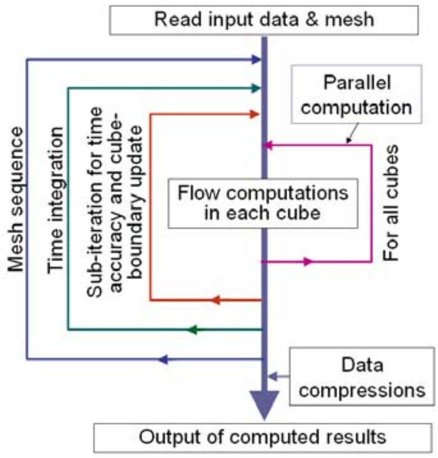

Fig.4: *Overall flow-solution procedure*

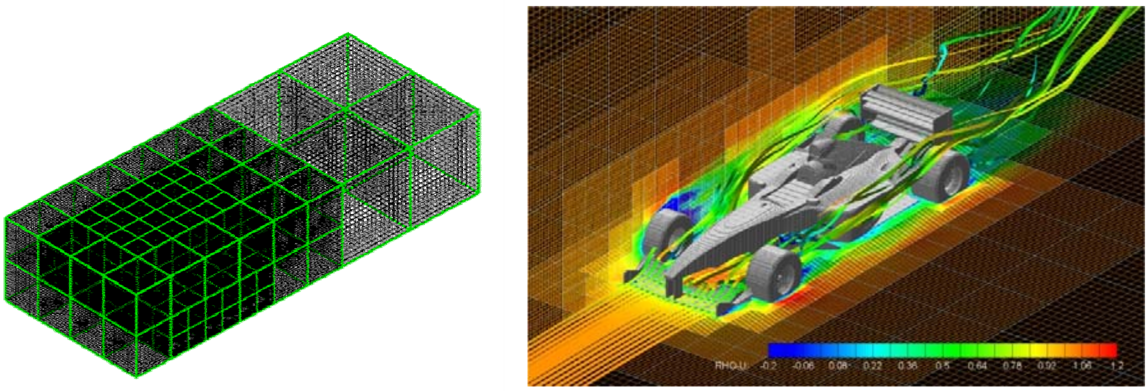

Fig. 5: BCM mesh (5930 Cubes with 32x32x32 mesh in each cube, totally about 200 million cells) and the time-averaged velocity distribution of the incompressible flow computation [8].

5. SIMPLICITY IS ESSENTIAL FOR NEXT-GERATION CFD

CFD, using a high-density Cartesian mesh, is still limited in its application due to the computational cost. The predictions about Cartesian mesh CFD and computer progress in this paper may be too optimistic. However, it is probably correct to say that the simplicity of the algorithm from grid generation to post processing of Cartesian mesh CFD will be a big advantage in the days of PetaFlops computers.

The flow is always wonder. CFD as a tool to see the complex flow physics, however, should be as simple as possible. This is what I learned from Professor Kunio Kuwahara.

REFERENCES

1. Jameson, A. and Baker, T. J. (1987). Improvements to the Aircraft Euler Method, *AIAA Paper 1987-452*.
2. Nakahashi, K., Ito, Y. and Togashi, F. (2003). Some Challenges of Realistic Flow Simulations by Unstructured Grid CFD, *Int. J. for Numerical Methods in Fluids,* Vol.43, pp.769-783.
3. Yamazaki, W., Matsushima, K., Nakahashi, K., (2008). Drag Prediction, Decomposition and Visualization in Unstructured Mesh CFD Solver of TAS-code, *Int. J. for Numerical Methods in Fluids*, Vol. 57, 417-436.
4. Top500 Supercomputers Sites (2008). http://www.top500.org/.
5. Nakahashi, K. (2005). High-Density Mesh Flow Computations with Pre-/Post-Data Compressions, *AIAA 2005-4876, Proc. AIAA 17th CFD Conf.*.
6. Meakin, R. L. and Wissink, A. M. (1999). Unsteady Aerodynamic Simulation of Static and Moving Bodies Using Scalable Computers, *AIAA-99-3302, Proc. AIAA 14th CFD Conf.*
7. Ishida, T., Takahashi, S., Nakahashi, K. (2008). Efficient and Robust Cartesian Mesh Generation for Building-Cube Method, *J. of Computational Science and Technology*, Vol.2, No.4, pp.435-446.
8. Takahashi, S., Ishida, T., Nakahashi, K., Kobayashi, H., Okabe, K., Shimomura, Y., Soga, T., and Musa, A. (2008). Large Scaled Computation of Incompressible Flows on Cartesian Mesh Using a Vector-Parallel Supercomputer, *Int. Conf. on Parallel Computational Fluid Dynamics*, Lyon, France.