Some recent topics of CFD activity in JAXA

Toshiyuki Iwamiya,* Satoru Ogawa*

1 Introduction

Computational Fluid Dynamics (CFD) technology has greatly developed with the progress of computers. In the development of the advanced aerospace technology such as a next generation SST, it is expected that CFD makes an important contribution. Without the computational result of highly precise elucidation of complicated fluid phenomenon, such as turbulent flow, separation, and noise, it is thought that development cost becomes more and more huge.

In JAXA-ITA, research and development of numerical simulation have been positively done since the early time of the 1980's, while top-level computers have been constantly introduced. Present CFD research is done about several research subjects in several research divisions auch as the Computational Sciences Research Group, the Aerodynamics Research Group, the Aircraft Engine Center, etc. One of the roles of ITA is to carry out a positive contribution to other development departments, and an application research which will put design development into the field of view is required.

Hereafter, we introduce some recent topics among the CFD researches being done in JAXA. From the basic reseaches we take two topics. One is a numerical method of high accuracy and resolution The other is a numerical simulation of combustion dynamics in a lean premixed combustor. From the application field we pick up the virtual flight test for helicopter. From the tool development we take an automatic grid generation for component-based surface geometry.

2 Numerical method of high accuracy and resolution

Recently, high-accuracy and high-resolution finite different schemes become widely used in many applications such as problems of aeroacaustics and stall of an airfoil. Analysis of these flows often requires accurate

unsteady treatments with turbulence, flow separation and flow compressibility. Direct numerical simulations seem to be impossible except for simple geometries at low Reynolds numbers in the near future. Therefore, researches of less expensive methods such as large eddy simulations (LES) are active for high Reynolds numbers and more complex geometries. However, getting quantitatively acceptable results for these flows are still now challenging.

In a series of papers ([1], [2], [3]), the team lead by Dr. T. Kurotaki in the Aerodynamic Research Group have proposed a new LES approach to simulate the flow containing turbulence and transition. The approach essentially combines ADM (approximate deconvolution model) with with the generalized characteristic interface conditions for singular lines in a grid. In the ADM approach, which was originally developed by Stolz et. al. [5], an approximation to the unfiltered solution is obtained from the filtered solution by the series expansion of deconvolution operator. The effect of scales smaller than the filter width is modeled by a relaxation regularization including a dynamically estimated relaxation parameter. So, it is no need to explicitly compute subgrid scale closures and construct extended models for the compressible flow. In practical computations with structured grid, singular points can be frequently found where an abrupt grid change exists. These singularities pose troublesome problems especially when high order and high resolution scheme is applied. So generalized characteristic interface conditions are used.

The main features of the numerical technique can be summarized as follows.

1. High-accuracy and high-resolution finite different schemes for spatial derivatives and time integration. 8.0pt plus 2.0pt minus 4.0pt0.0pt1.0pt plus 1.0pt minus 1.0pt

   - Optimized 6th order tri-diagonal type compact scheme
   - Low storage type of 2-step 4th order low-dissipation and low-dispersion Runge-Kutta (LDDRK) scheme

2. Compressible Large Eddy Simulations (LES) approach for turbulent flows with ADM

---

*Japanese Aerospace Exploration Agency, Institute of Aerospace Technology, Computational Science Research Group

3. Boundary conditions
   - the Navier-Stokes characteristic boundary conditions extended to the generalized coordinate system by Kim et. al. [6].
4. Treatments of singular lines for generalized coordinates.
   - Single grids with the generalized characteristic interface conditions along the singular line behind thetrailing edge of the wing.

The more detail of the computational technique can be seen in [1].

Some examples of this method are shown below.

The first example is flows around the airfoil NACA0012 with chord length of 1.0m. The angle of attack is 0 degree and the uniform flow velocity is taken as 15, 20, 25, and 30 m/s. The number of grid points ia about 3.5 million.

(a) $U_\infty = 15$m/s   (b) $U_\infty = 20$m/s
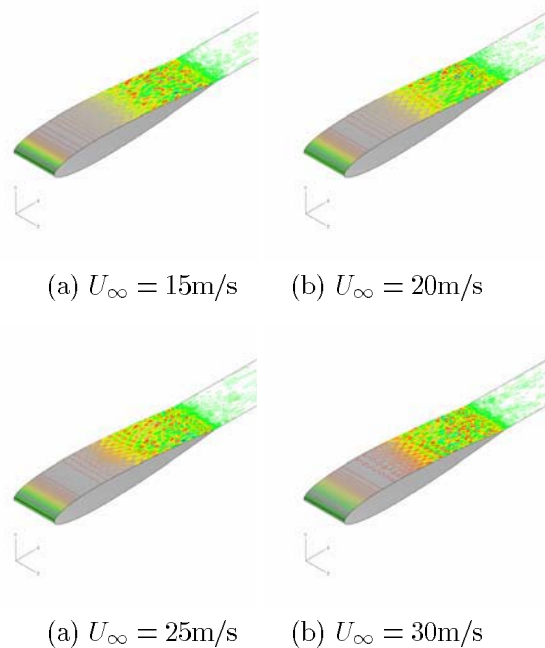

(a) $U_\infty = 25$m/s   (b) $U_\infty = 30$m/s

Fig. 1 The contour of instantaneous vorticity

Fig. 1 shows the contour of instantaneous vorticity of the spanwise component just away from the wall. As can be seen from this figure, the so-called 2-D T-S wave is growing at the beginning of the boundary-layer transition and then is deformed into the peak-valley structure due to the secondary instability, resulting in amplifying oblique T-S waves. This process agrees with the scenario of 2-D boundary-layer transition observed in the low turbulence uniform flow. From this figure, it can be supposed that the transition occurs at the location of about 60% code length.

In order to discuss about transition location, they used the C_f distribution. Firstly the C_f line is extrapolated by three different lines. Two intersections usually appear along the C_f curve. Then one of the intersection close to the leading edge is defined as "the on set of transition." Whereas, the intersection close to the trailing edge is defined as "the end of transition." Finally, the transition location is defined as the middle point of those two intersections. The procedure of this is shown in Fig. 2.

(a) $U_\infty = 15$m/s   (b) $U_\infty = 20$m/s

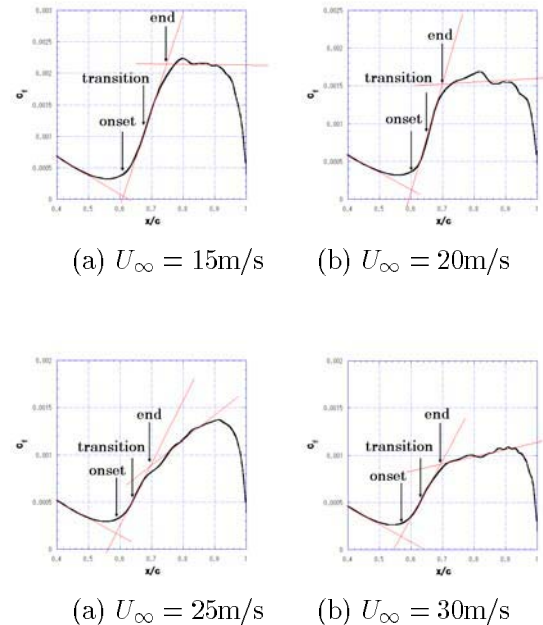
(a) $U_\infty = 25$m/s   (b) $U_\infty = 30$m/s

Fig. 2 The definition of the transition location

Fig. 3 shows the top view of the contour map for four uniform flow cases corresponding to Fig. 1. The uniform flow comes from the left hand side and the indices on the top of the figures denote the spanwise location under the normalization with the chord length. Three arrows put on the each case denote location of the onset, the transition, and the end, respectively. In this figure, it is shown that each arrow gradually goes upstream as the uniform flow velocity increases. In the present case, since the increase of the uniform velocity means the increase of the Reynolds number, the movement of the transition location is quite reasonable.
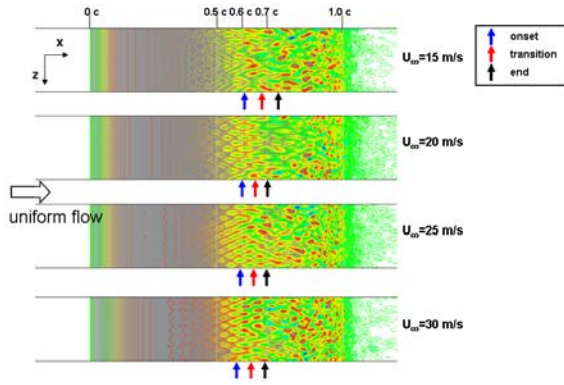
Fig. 3 The transition location for each case

Then the results plotted on the experimental results obtained by Tokugawa et.al. ([7]) is shown in Fig. 4. The experiments was carried out using two different wind tunnels indicated triangle and circle symbols in the figure. The discrepancy between the experimental results seems mainly caused by the turbulent level. This level for the wind tunnel A is about 0.05%, whereas the wind tunnel B is about 0.15%. The transition location is generally moved forward when the turbulent level increases. In this circumstance, it seems reasonable that the CFD delays the transition because the turbulent level is quit small.
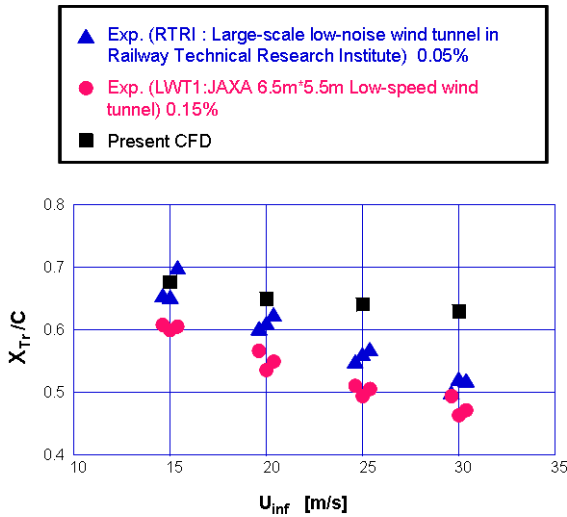
Fig. 4 The transition location

Fig. 5 shows the velocity fluctuation for some X/C locations. From this figure, the sinusoidal wave appears after X/C = 0.2 and the amplitude increase toward the trailing edge (the scale of vertical axes are different at each location). Moreover, the frequency of the wave is roughly estimated about 200 Hz. These facts imply that the T-S wave appears at least after 20% chord location and its frequency is about 200 Hz.
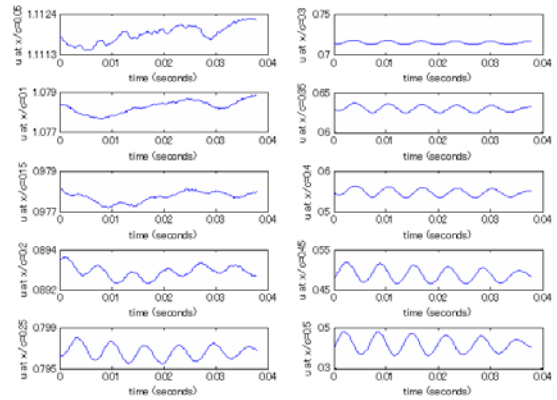

Fig. 5 The velocity fluctuation at some streamwise location

On the other hand, the disturbance intensity growth is usually estimated by e^N method in the linear stability analysis. The variation of the N factor with frequency at the location of X/C = 0.4 is shown in Fig. 6. It is clearly shown that the peak exists at almost 700Hz. A square symbol denotes the result from the SALLY code for the most amplified disturbance. Although the magnitude of the N factor shows difference a little, the frequency of the most amplified disturbance coincides with each other. This quantitative discrepancy may come from the insufficiency of accuracy pf the numerical algorithms. Recent study shows that more accurate scheme is able to reproduce about 700Hz of the most amplified frequency.
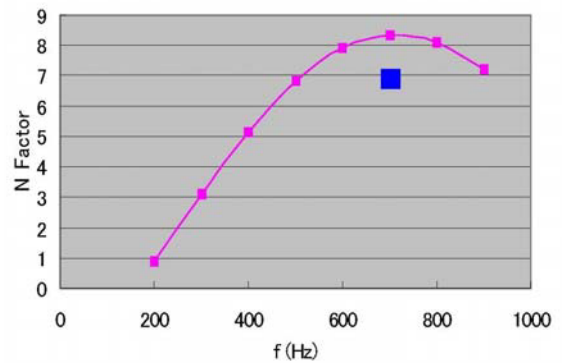

Fig. 6 Variation of the N factor with frequency at X/C = 0.4

3   Numerical Simulation of Combustion Dynamics in a Lean Premixed Combustor

Emission regulations for gas turbines are becoming more and more stringent due to environmental con-

cerns. Lean premixed combustion is a promising way to reduce thermal NOx emissions because the flame temperature is relatively low. However, this combustion method tends to be unstable. Combustion oscillations may occur and cause mechanical damage to the combustor system. Blow-off may also be induced. Thus, suppressing the instabilities is necessary to realize lean premixed combustors.

Dr. Shinjo and his team in the computational science group are performing numerical simulations to understand the underlying physical processes of oscillation suppression by secondary fuel injection.

The experiment has been carried out by Tachibana et al. ([9], [10]). The features of the experiment are as follows (See Fig. 7). The main flow equivalence ratio is low around 0.5, which means low NOx and CO. The active control is done by secondary fuel injection at the harmonic frequency with a high-speed valve. The secondary fuel changes the heat release field in the combustor and the coupling in the oscillation cycle. The control algorithm is based on robust H_2/H_∞ mixed control. In the experiment, measurements were done to obtain data such as temporal pressure traces, time-resolved chemiluminescence images, NOx gas sampling.
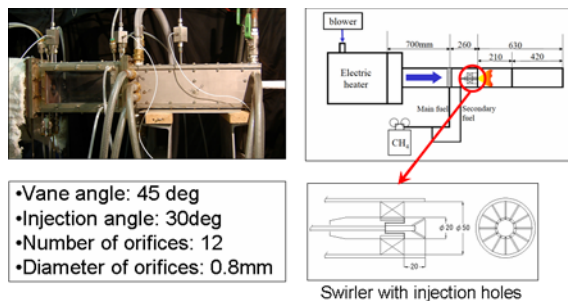

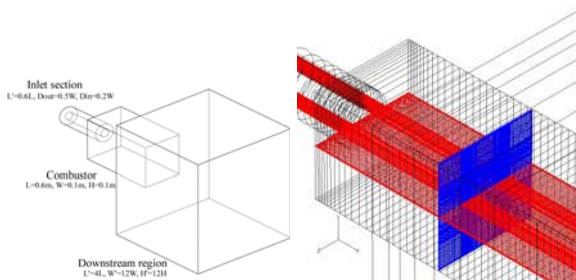Fig. 7 Experiment of model gas turbine combustor

Fig. 8 Computational domain and grid system

The computational domain consists of combustor, inlet section and downstream region. The downstream region is attached to avoid the ambiguity of boundary conditions at the combustor exit. The number of grid point is about 10 million in total. The grid points are clustered in the flame region. The minimum grid

spacing is 0.15mm. Fig. 8 shows the computational domain and grid system used here.

The governing equations of flow field are three-dimensional compressible Navier-Stokes equations. The conservation of mass, momentum and energy are solved. For the LES subgrid scale modeling, the conventional dynamic Smagorinsky model is used.

Unstable combustion dynamics and control methods have been simulated by LES. The present numerical formulation has succeeded in reproducing the phenomena inside the combustor such as acoustic resonance and flame/vortex interaction. Acoustic resonance in the combustor causes the flow rate into the combustor to oscillate. This fluctuation produces periodical vortex shedding from the dump plane. The interaction of these vortices with the flame generates heat release rate fluctuation. If the pressure and heat release fluctuations are in phase, this will generate the acoustic energy. This cycle is repeated during the unstable behavior. ([11], [12])
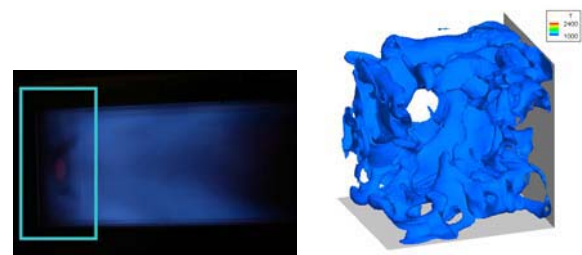

Fig. 9 Experimental flame photo and instantaneous flame shape by simulation

Fig. 9 shows the experimental direct flame image and the flame shape obtained by the simulation. The flame is anchored near the combustor entrance due to the recirculation zones created by the main flow swirling motion (See Fig. 10). In the no-control case, the flame attaches to both inner and outer swirler rims and its behavior is very unstable.
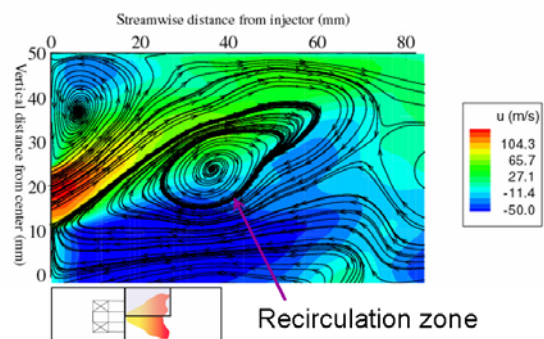

Fig. 10 Recirculation zone

The combustion control method based on secondary fuel injection is then applied to this flow field. First,

constant injection of secondary fuel is tested. As observed in the experiment, the change in the flame shape has an effect on the overall stability. The main flame shape modification leads to reduced flame/vortex interaction and reduced heat release/pressure coupling. The secondary flame serves to stabilize the central recirculation zone.

Next, secondary fuel modulation is examined. The phase-shift approach is examined to simulate the experiment. Here, the phase is defined as 0deg when the pressure (or injection velocity) starts to increase from the mean value. The secondary fuel flow rate is harmonically varied with the phase angle as a parameter. Two injection velocity phase angles of 0deg and 180deg are tested. The injection velocity is determined so that the secondary fuel is 3% of the main fuel and the variation is set at 10% of the total secondary fuel at the maximum pressure oscillation amplitude. Fig. 11 shows the temporal evolution of pressure, injection velocity and secondary heat release for both cases. The amplitude reduces for the phase of 0deg and increases for 180deg. Note that the 0deg and 180deg are for the injection velocity at the injection hole. The heat release phase is shifted from the angle of this velocity. For both cases, the secondary heat release is in a reverse phase with the injection velocity. In the 0deg case, the secondary heat release and the combustor pressure are in a reverse phase, resulting in reduced oscillation. On the other hand, for the 180deg case, the secondary heat release is in phase with the combustor pressure and, as a result, the oscillation grows. The secondary heat modulation has a strong effect on the stability. The heat release fluctuation produces pressure fluctuation, so the phase angle between the pressure and heat release is important in determining the stability. The actively changed secondary flame has a significant effect when the phase is reversed.
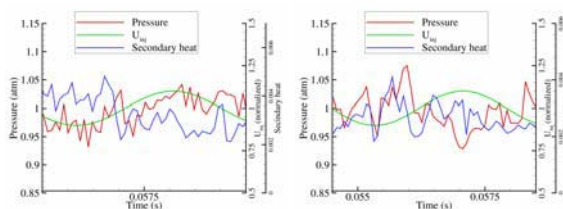
0deg case (left) and 180deg case (right)
Fig. 11 Temporal profiles of combustor pressure, injection velocity and secondary heat release

4 Virtual Flight Test for Helicopter

JAXA has been developing its own CFD code to solve full helicopter simulation using accurate flow solver and acoustic solver as is shown in Fig. 12.
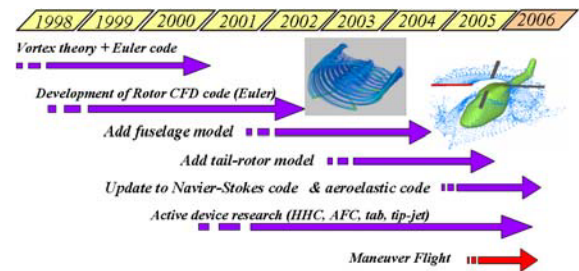

Fig. 12 Analytical tools of helicopter aerodynamics and acoustics developed in JAXA(NAL)

The researches have shown its ability to capture the distinct peak of BVI for several problems including active control analysis. The code expanded its ability to solve the flow-field including tail-rotor and fuselage configuration including interaction noise analysis. The noise generated by a maneuvering rotorcraft would be the next important challenge on the way to full helicopter simulation. The lead by Dr. Aoyama in the Computational Science Group is applying a CFD code using moving overlapped grid method, which is one of the most advanced techniques for tip-vortex capturing at present, to solve the noise of maneuvering rotorcraft. Although it is difficult to fully validate the noise prediction capability in short of experimental data, the numerical simulations are supposed to provide understanding of the characteristics for maneuvering rotorcraft, which can be valuable in full helicopter design.
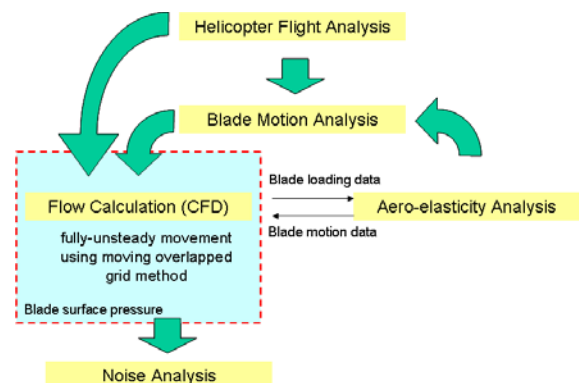

Fig. 13 Flowchart for helicopter noise analysis

Fig. 13 shows general and present flowcharts for helicopter noise analysis. The noise prediction system for maneuvering flight problems should integrate several

analysis tools: flight simulation, blade motion analysis, aero-elasticity coupling, noise analysis, and flow simulation (CFD) code. Each step is important to get an accurate noise prediction, but the step to get exact flow solution including tip vortex trace would be most difficult and time-consuming part.
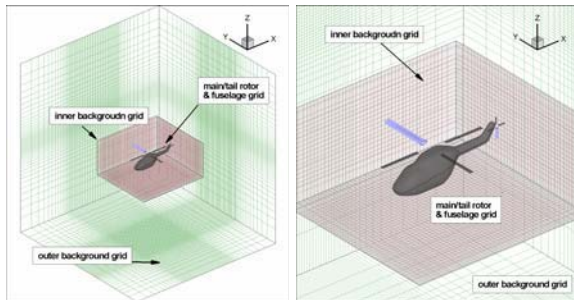
(a) Perspective view   (b) Close-up
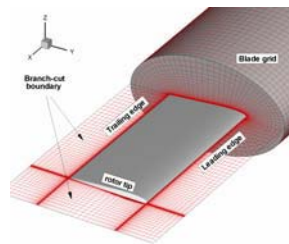
(c) Blade grid

Fig. 14 Moving overlapped grid system

A moving overlapped grid system with three different types of grids (blade grid, inner and outer background grids) shown in this figure is used. The inner background grid is placed around the rotor disk. The outer background grid covers the whole computation region with a sparse grid density. The flow data are exchanged between the inner and outer background grids, and between the blade grid and the inner-background grid. The body-fitted blade grid in O-H topology moves along with the blade motion including rotation, flapping, feathering, and lagging. The elastic motion of blade is neglected because blades with rigid body are considered here.

Table 1 shows the specification of each grid. Most of the grid points is concentrated in inner-background grid, which captures the trace of tip vortex during several rotations. The number of grid points in spanwise direction of blade grid is considerably increased to model the active flap. The grid spacing of inner background grid corresponds to $0.1c$, where c is the blade chord length.
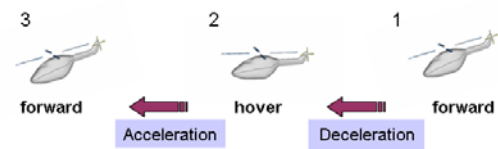

Fig. 15 Transient flight

A transient flight between forward flight and hovering flight is simulated to demonstrate the unsteady flow characteristics with acceleration and deceleration of maneuvering helicopter. A list of flight conditions along with relevant flight state information is shown in Table 2(a).
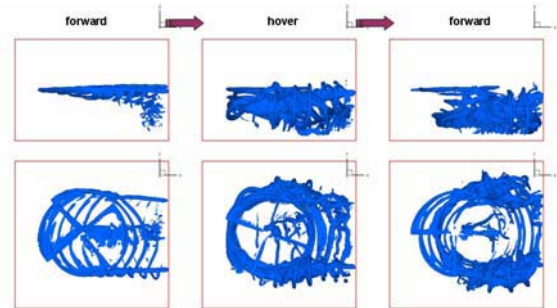

Fig. 16 Vorticity iso-surface in case 1

Fig. 16 shows the side view and top view of vorticity iso-surface at each stage. The side views show well the difference of tip vortex downwash according to the flight conditions. The top views show the effect of flight speed with the distance change between each tip vortex according to each stage.
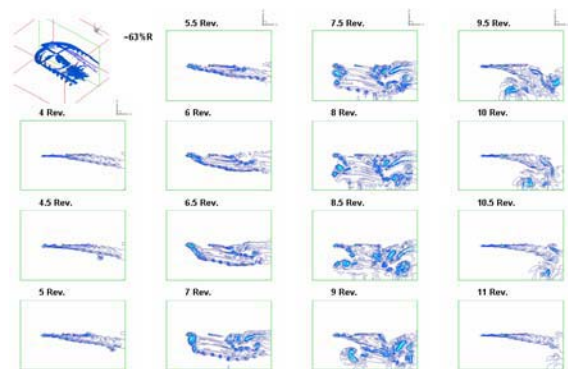

Fig. 17 Side view vorticity contour at sectional plane about 63% of rotor radius in advancing side

Fig. 17 shows vorticity contours at sectional plane about 63% of rotor radius in advancing side, where the BVI is supposed to originate. The contours show the trace of tip vortex according to each stage well. As deceleration begins at 4th revolution, the downwash grows dominant relatively because the flight speed (i.e.

Table 1 Specification of each grid

Inner background grid	(X×Y×Z) 290×270×180 = 14,094,000
Outer background grid	(X×Y×Z) 83×79×49 = 321,293
Blade grid	(chord×normal×span)×blade (141×25×125)×4 = 1,706,100
Total	16,000,000 points
Inner background spacing	0.08c (=0.0105R)

Table 2: Specification of helicopter motions and flow conditions

(a) Case 1: Accelerating/decelerating flight

case 1	condition	M_∞	α_{tpp}	Collective (deg)	Lateral (deg)	Longitudinal (deg)	Yaw (deg)	Roll (deg)
Forward	periodic	0.109	3	5.2	-1.33	2.72	0	0
Transient flight	aperiodic	curvilinear interpolation using flight data						
Hovering flight	periodic	0	0	9.92	0	0	0	0
Transient flight	aperiodic	curvilinear interpolation using flight data						
Forward	periodic	0.109	3	5.2	-1.33	2.72	0	0

(b) Case 2: Right-turn flight

case 2	condition	M_∞	α_{tpp}	Collective (deg)	Lateral (deg)	Longitudinal (deg)	Yaw (deg)	Roll (deg)
Forward	periodic	0.109	3	5.2	-1.33	2.72	0	0
Transient flight	aperiodic	curvilinear interpolation						
Right Turn	periodic	0.109	2.21	14.03	-3.31	10.23	-90	45
Transient flight	aperiodic	curvilinear interpolation						
Forward	periodic	0.109	3	5.2	-1.33	2.72	-90	0

inflow velocity) decreases. As re-acceleration begins at 7th revolution, the down-going tip vortex starts moving again along the stream, and the old tip vortex under the rotor disk move together with the newly-generated tip vortex as the flight speed increases. Finally the trace of tip vortex recovers its shape to be initial level flight condition.
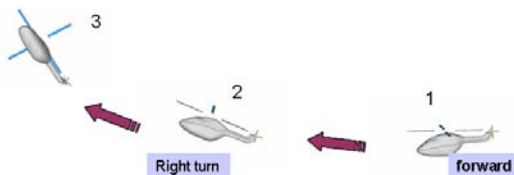

Fig. 18 Right-turn flight

A transient flight from level flight to a coordinate right-turn transient flight from hovering flight to forward flight was simulated to demonstrate the unsteady. A list of flight conditions along with relevant flight state information is shown in Table 2(b).
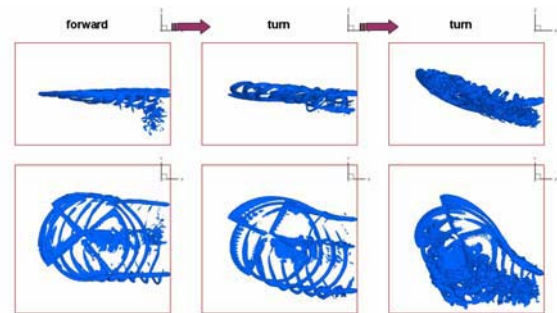

Fig. 19 Side view and top view of typical iso-surface of vorticity

Fig. 19 shows side view and top view of vorticity iso-surface at each stage. The side views show well the difference of tip vortex downwash according to roll angle during right-turn. The top views show the effect of flight speed (in other words, inflow velocity) and flight direction (in other words, yaw angle) with the change of directivity in the trace of tip vortex according to each stage. The main characteristics of turning process, as shown in these figures, can be explained that the tip vortex of advancing side are overlapped to be accumulated according to the turning blade it-

self. The blade is expected to experience stronger interactions during passing through the accumulated tip vortex when compared to ordinary blade-vortex interaction.

Fig. 20 shows unsteady pressure histories at a surface point of 3% chord and 90% span for several revolutions during right-turn flight. The time derivative of this unsteady pressure has good correlation with noise intensity. The shape of pressure history changes a lot according to different flight condition, and after entering right-turn, the gradient of pressure become quite steep similar to typical BVI case. This enlarged pressure gradient, which is directly related to noise, are supposed to be originated from the strong interaction between proceeding blade and accumulation of tip vortex by right-turn.
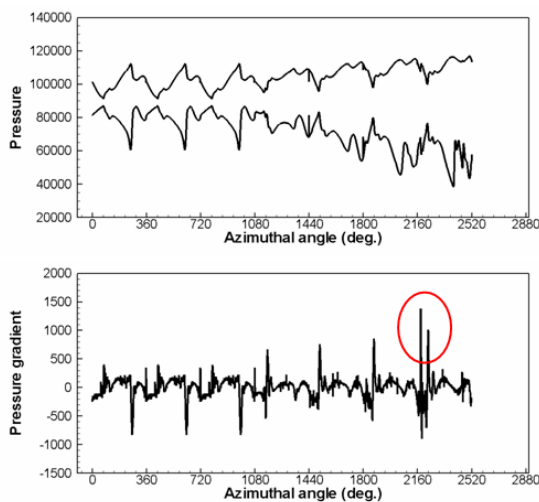

Fig. 20 Pressure and pressure gradient history at a
point on upper/lower surface of 1st blade

These effects of sudden pressure change are well shown in this figure, pressure gradient history at a point on lower surface of 1st blade. Compared to the first three peaks which are supposed to come from BVI in level flight, the following peaks are growing up to show the BVI of right-turn flight. The final peak shows relatively large magnitude, which is supposed to come from so-called 'super-BVI'.

5   Automatic Grid Generation for Component-based Surface Geometry

Grid generation is still a major bottleneck in CFD analysis. Dr. Lahur has proposed an approach based on the Cartesian grid because of its fast and automatic

nature, as well as its preferred shape. Since a method that can generate grid for both inviscid and viscous flows is desirable, a hybrid with prismatic grid is considered to obtain a body-fitted boundary.

There are basically two options in generating such hybrid as in Fig. 21. One is a hybrid with cutcell approach, where prismatic grid is generated first from the solid surface (Boundary to Interior approach), followed by Cartesian grid and cutcells. The other is a hexahedra grid, where prismatic grid is generated toward solid surface (Interior to Boundary approach). In this case, the shape of the prismatic cells is hexahedral.
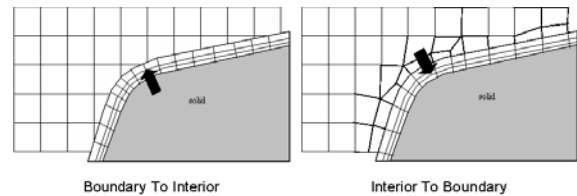

Fig. 21 Two approaches of a hybrid grid system

We follows the second approach because it does not necessarily require high quality of the surface grid. The solid surface does not have to be water-tight, thus gaps and overlaps of surface elements are allowed. This advantage can be exploit further, that is, to generate volume grid directly from intersecting solid components. Thus, each component can be defined, constructed, and modified independently without having to regenerate the whole surface grid prior to volume grid generation. This is particularly advantageous in preliminary design, where major configuration change often takes place.
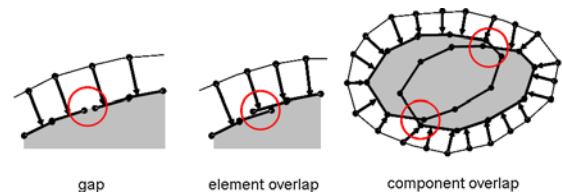

Fig. 22 Examples of low-quality surface grid

At first, Cartesian grid is generated by means of successive local refinements. This step starts with one cell that covers the whole computational domain. In three-dimensional space, each refinement divides a cell isotropically into eight child cells of equal size and shape. This type of refinement is supported naturally by octree data structure, where child cells are represented as branches of the tree. As the grid becomes more refined, the tree becomes taller and acquires more branches. Only the cells that do not have children participate in fluid computation. This data structure provides an efficient way to find neighboring cell by means

of traversing the tree. The strategy of grid refinement is as follows: At the beginning, the grid is locally refined until the size of cells intersecting solid surface is smaller than a maximum allowable value. Then the grid is further refined until the size of cells intersecting the features of solid surface reaches either a satisfactory level, or a user-defined minimum value.

The next step is removal of cells near solid surface and construction of "quad surface". The quad surface is a surface grid that approximates the wetted surface of solid body. The quad surface contains quadrilateral faces, with the exception of faces whose neighbor belong to higher refinement level (faces that contain "hanging node"). The construction is carried out by removing cells near solid surface. The exposed faces of the remaining Cartesian grid cells are then interconnected to form the quad surface. It is then smoothed to improve the approximation to surface in order to ease the next task. A new quad surface is constructed, which initially has identical topology and geometry to quad surface.

Then the new surface is then snapped onto the solid surface by moving each node of the new surface to the closest location on the solid surface. The algorithm works even when the solid surface contains gap, overlapping faces and even intersecting components. However, severely sharp concave features still can not be captured with snapping. The same is also true for intersection region, because it tends to form sharp concave features. Thus, these features need to be explicitly captured if necessary.

Depending on the complexity of the geometry, the features may form a simple line, or they may form a complicated arrangement. It is found that the key to successful feature capturing is to make use of the fact that only a valid concave feature needs to be explicitly captured. A concave feature appears either within a component, or as the result of intersection between components. one, two, or three node(s) of quad face may be moved onto the line. The result is shown in the last figure.
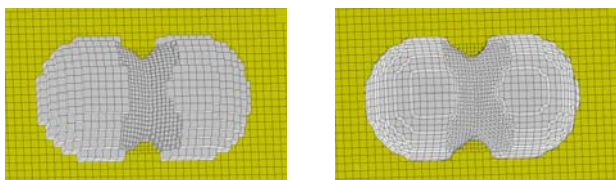
(a) Initial quad surface   (b) Smoothed quad surface
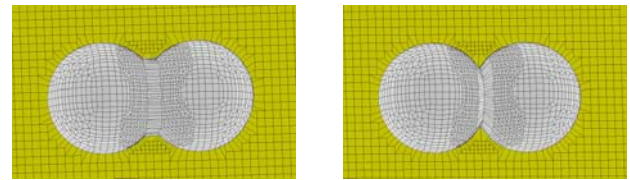
(c) Snapped quad surface   (d) Feature captured

Fig. 23 Grid generation process

The model in the test case as shown in Fig. 24 consists of body and wing. The final hexahedra grid contains about 200,000 cells, 700,000 faces, and 250,000 nodes. The time to generate is around 7 minutes on a PC with modest specification (single Pentium 4 processor running at 1.9 GHz). The grid is quite coarse, especially in the region around the wing's leading edge. It is indeed the purpose for this case to demonstrate that the method works even when the grid is coarse. As in other methods of grid generation, increasing the resolution of the grid makes the task easier to perform. Feature capturing reduces the maximum value of approximation error of the quad surface almost one order of magnitude. As for the cells on the solid surface, their faces are very close to flat. This is especially true for the quad faces on the Cartesian grid side and those on solid surface, whose flatness values are 100% within 0.9 and 1.0. For side faces that connect the two quad surfaces, 97% are within the same range of values. Reduction in flatness of side faces has to be accepted in order to capture the sharp concave and convex features.
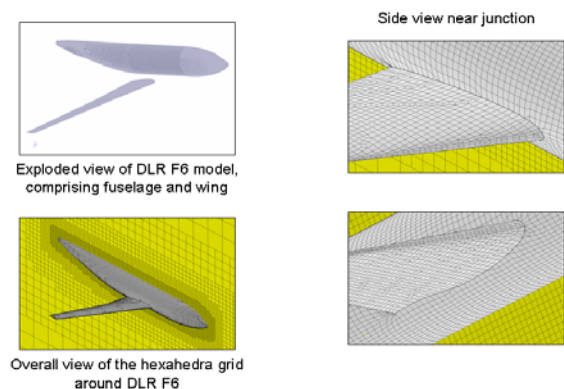

Fig. 24 Model with body and wing

The model in Fig. 25 case consists of four components: fuselage, wing, engine nacelle and pylon. They contain 180,000 triangles and 92,000 nodes in total. Figures show the final grid, which contains about 830,000 cells, 2,200,000 faces and 520,000 nodes. The number of faces on solid surface is 64,000, and the nodes are about the same. The time to generate is 11 minutes

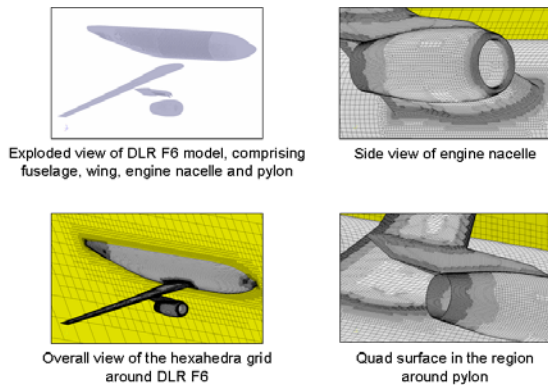and 52 seconds on the same machine as in the previous test case.

Fig. 25 Model with nacelle

The last test case in Fig. 26 is a generic wind tunnel model. Generic vertical and horizontal tails, as well as a sting support system are added to the model in the previous test case. Now the surface grid contains 185,000 triangles and 93,000 nodes. Note again that adding new parts requires nothing more than including them in the existing geometry file. There is no need to go back to CAD and rebuild the surface grid. The grid generated around the model consists of 2.4 million cells, 6.5 million faces and 1.6 million nodes. There are about 120,000 grid faces and 120,000 nodes on the solid surface. The time taken to generate the grid is 22 minutes 45 seconds (CPU time) on the same machine as in the previous test cases.
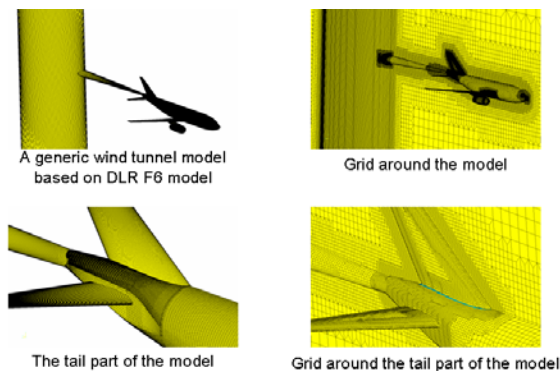

Fig. 26 Wind tunnel model

References

[1] Kurotaki, T. and Sumi, T., Proc. Fifth International Symposium on Advanced Fluid Information AFI-2005, 2005, pp.14,15.

[2] Sumi, T., Kurotaki, T. and Hiyama, J., AIAA 2006-1272, 2006.

[3] Atobe, T., Kurotaki, T. , Sumi, T. and Takagi, S., AIAA 2006-3533, 2006.

[4] J. Kim and D. Lee, AIAA Journal, Vol. 41, 2003, pp. 2341-2348.

[5] Stolz, S., Adams, N. A., and Kleiser, L, Phys. Fluids, Vol. 13, 2001, pp. 2985, 3001.

[6] Kim, J., and Lee, D., " AIAA J. Vol. 38, 2000, pp. 2040, 2049.

[7] Tokugawa, N., Takagi, S, et al., J. Jpn Soc. Fluid Mech., Vol. 22, No. 6, 2003, pp. 485, 497, (in Japanese).

[8] Shinjo, J., Matsuyama, S., Mizobuchi, Y. and Ogawa, S., to appear in Computational Fluid Dynamics JOURNAL.

[9] Tachibana, S., Zimmer, L., Kurosawa, Y., Suzuki, K., Shinjo, J., Mizobuchi, Y., and Ogawa, S., Proc. 6th Symp. Smart Control of Turb., Center for Smart Control of Turbulence, Tokyo, 2005, pp181-190

[10] Tachibana, S., Zimmer, L., Kurosawa, Y., Suzuki, K., Sato, H., Hayashi, A.K., Nishidome, C., and Kajiwara, I., Int. J. Vehicle Des., to appear

[11] Shinjo, J., Mizobuchi, Y., and Ogawa, S. Proc. 4th Symp. Smart Control of Turb., Center for Smart Control of Turbulence, Tokyo, 2003, pp165-172

[12] Shinjo, J., Mizobuchi, Y., and Ogawa, S., CFD J. 13(2) (2004) 348-354

[13] Ochi, A., Aoyama, T., Saito, S., Shima, E., and Yamakawa, E., AHS 55th Annual Forum, Montreal, Canada, May 1999.

[14] Yang, C., Aoyama, T., and Saito, S., 31st ERF, No. 24, Florence, Italy, September, 2005

[15] Aoyama, T., Yang, C., and Saito, S., AHS 61st Annual Forum, Grapevine, TX, June, 2005

[16] Aoyama, T., Yang, C., Kondo, N., and Saito, S., Theoretical and Applied Mechanics Japan, Vol.53, pp.215-220, October, 2004 3.

[17] Yang, C., Aoyama, T., and Saito, S., 24th International Congress of the Aeronautical Science (ICAS), Yokohama, Japan, 29 Aug.-3 Sep.,2004