JOURNAL OF AIRCRAFT
Vol. 47, No. 2, March–April 2010

Computational Fluid Dynamics Validation Study of Wake-Capturing Capability for a Flat-Plate Wake

Akio Ochi,* M. K. Ibrahim,† and Yoshiaki Nakamura‡
Nagoya University, Nagoya 464-8603, Japan

DOI: 10.2514/1.33502

This paper studies a computational fluid dynamics validation study of wake-capturing capability for unsteady flow characteristics. The computational fluid dynamics code used in this study is based on the Cartesian grid system with octree mesh refinement and uses a weighted essentially nonoscillatory numerical scheme to achieve higher-order accuracy to preserve unsteady flow characteristics. Flow properties in the flat-plate wake flowfield, such as mean velocity, root mean square of perturbation velocity, and frequencywise power spectrum of flow velocity, were chosen to validate the computational fluid dynamics code. The comparison between computational fluid dynamics results and experimental data showed good agreement for mean velocity and low-frequency fluctuations.

I. Introduction

UNSTEADY computational fluid dynamics (CFD) analysis is getting feasible and important with the increase of computer performance, and steady-state CFD analysis has a long-term success in the field of aerodynamics [1]. Unsteady analysis has the substantial potential to ensure the reliability of aircraft aerodynamic design. Its advantages are obvious for nondesign point aerodynamics such as wake interaction, aerodynamic vibration, and buffet prediction. In addition, detailed unsteady computation leads to evaluate aero-acoustic sound sources. Unsteady CFD analysis is one of the most promising areas to increase CFD contribution to the field of aerodynamics. However, unsteady computation is still expensive compared with steady Reynolds-averaged Navier–Stokes analysis in spite of the rapid increase of computer performance and decrease of cost. Thus, an efficient CFD code for practical unsteady analysis is required.

The ultimate goal of the present research is the development of a CFD code and its validation for an unsteady high Reynolds number flow in realistic complex geometries. A high Reynolds number means several millions to several tens of millions to compute the whole aircraft. To achieve this goal, grid generation to treat complex geometries, a higher-order numerical scheme to well capture vortical flows, and scalable parallel processing for large-scale computation should be carried out at the same time.

The present CFD code has been developed to implement those three features. This code, which is referred to as CHARGE (Cartesian Higher-Order Adaptive Refinement Grid Experiment), uses a Cartesian grid with an octree adaptive refinement mesh (AMR) [2] to treat complex geometry and increase grid density in places of interest. In this study, the weighted essentially nonoscillatory (WENO) [3,4] scheme is employed as a higher-order numerical scheme. This scheme was chosen because of its capability to deal with a shock in transonic/supersonic flow. Regarding scalable computation, the present code was parallelized by using the OpenMP and the message-passing interface (MPI) library for data communication.

In this paper, a validation study for this newly developed CHARGE code is described. The validation was conducted for the rectangular flat-plate wake. Both mean and unsteady flow features were compared between numerical results by the present CHARGE code and experimental result by the wind-tunnel test.

II. Numerical Methodologies

A. Grid System

The Cartesian grid system with octree AMR is employed in the present code. A primary reason to use the Cartesian grid system is the orthogonality of a grid. When we implement a higher-order scheme, an orthogonal grid is more efficient to actually keep higher-order accuracy [5,6]. A higher-order method on an unstructured grid is still expensive for practical use. In addition, a Cartesian grid is easy to generate even for a complicated geometry and is suitable to implement solutions-adaptive refinement. Using a Cartesian system, grid cells keep a regular hexahedral shape even after the refinement process. A weak point of the Cartesian grid system is poor resolution at a thin boundary layer in high Reynolds number flow. Several remedies for this issue have been proposed by some researchers. However, the present version of this code has not yet employed such remedies. Other advantage is the use of less memory compared with an unstructured grid system, because of a simple data structure.

A JAVA code was developed that generates a Cartesian grid for an arbitrary shape. Stereolithography Interface Format is used as geometry input data format, and the input geometry does not need to be a watertight geometry, in which gaps less than half of a cell size and unlimited overlaps are allowed. The Bresenham algorithm was employed to find Cartesian volume cells that contain the surface mesh, and those cells are recursively refined until the prespecified refinement level. The Bresenham algorithm substantially accelerates the grid generation process. It takes less than a minute to generate one million volume cells and three minutes for ten millions. The seventh to ninth refinement levels are usually used.

Once a Cartesian grid is generated, it is reprocessed for parallel computation. The whole grid cells are decomposed into cell clusters; each cell cluster consists of 1000 to 10,000 cells, which depend on a total number of grid cells in a system. The total number of cell clusters is 1000 to 10,000, which is also dependent on the total number of grid cells. A cell cluster is used as a unit when distributing the grid system for parallel computation. Because the number of cell clusters is much less than that of grid cells, decomposition is much faster and so the number of parallel process can be specified as a runtime argument for the flow solver. Cell clusters are allocated to each process to balance the computational work load.

Presented as Paper 4195 at the 18th AIAA Computational Fluid Dynamics Conference, 25-28 June 2007, Miami FL; received 17 July 2007; revision received 31 July 2008; accepted for publication 10 August 2008. Copyright © 2008 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved. Copies of this paper may be made for personal or internal use, on condition that the copier pay the $10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0021-8669/10 $10.00 in correspondence with the CCC.

*Graduate Student, Department of Aerospace Engineering; akio@fluid.nuae.nagoya-u.ac.jp. Member AIAA.

†Research Associate, Department of Aerospace Engineering; khalil@fluid.nuae.nagoya-u.ac.jp.

‡Professor, Department of Aerospace Engineering; nakamura@nuae.nagoya-u.ac.jp. Member AIAA.

B. Numerical Schemes

The WENO scheme was employed for spatial discretization to achieve higher-order accuracy. Reducing numerical dissipation by higher-order schemes leads to reducing the number of grid points [3]. The WENO scheme was chosen because of its capability to handle a shock in transonic/supersonic flow. Basically, the numerical scheme of the present code is based on the reference [3,4].

The intensive effort in the present study is directed to a treatment of irregular-spacing-size cells. Because AMR grid is used in the present code, there is cell size gap at the boundary between different refinement-level regions, as shown in Fig. 1. The size of a larger cell has double that of a smaller cell at the boundary.

To prevent accuracy loss at these irregular-spacing cells, an essentially nonoscillatory (ENO) reconstruction coefficient is calculated for each irregular-spacing stencil, as shown in Fig. 2. The essence of ENO reconstruction is how to obtain cell interface value v_i from cell-averaged values $\bar{v}_{i+1/2}$. The cell-averaged value is defined by Eq. (1), which can be expanded to Eq. (2) using Taylor expansion. Cell-averaged values for other cells are denoted in Eqs. (3–5) in the same manner. When obtaining the cell interface value, v_i and $dv/dx|_{x0}$ are unknown values. On the other hand $\bar{v}_{i+1/2}$ and x_i are known values. To obtain the cell interface value v_i, Eqs. (2–4) are solved as simultaneous linear equations. The number of equations that couple as simultaneous equations depends on the required order of accuracy. The rth order accuracy requires a number of r equations to be coupled. Equation (5) can be simplified to Eq. (6) for uniform-spacing grid, because $x_i - x_{i-1}$ can be reduced to Δx, which is independent of cell index i. Hence, the solution of the simultaneous linear equations for the simple uniform-cell case does not depend on index i, and it can be obtained by a plain linear combination using coefficient $c_k^r (k = 1, \ldots, r)$ as written in [3]. However, in the present study, it needs to solve linear equations for an irregular-cell-size grid.

$$\bar{v}_{1+1/2} = \frac{1}{x_1 - x_0}\int_{x0}^{x1} v(x)\mathrm{d}x = \frac{1}{x_1 - x_0}\int_0^{x1-x0} v(x_0 + x)\mathrm{d}x \tag{1}$$

$$\begin{aligned}\bar{v}_{1+1/2} &= \frac{1}{x_1 - x_0}\int_0^{x1-x0}\left(v(x_0) + x\frac{dv}{dx}\bigg|_{x_0} + \frac{x^2}{2}\frac{d^2v}{dx^2}\bigg|_{x_0}\right.\\ &\quad \left. + \frac{x^3}{6}\frac{d^3v}{dx^3}\bigg|_{x_0} \cdots + \frac{x^k}{k!}\frac{d^kv}{dx^k}\bigg|_{x_0} + \cdots\right)\mathrm{d}x\\ &= \frac{1}{x_1 - x_0}\left[v(x_0)x + \frac{x^2}{2}\frac{dv}{dx}\bigg|_{x_0} + \frac{x^3}{6}\frac{d^2v}{dx^2}\bigg|_{x_0} + \frac{x^4}{24}\frac{d^3v}{dx^3}\bigg|_{x_0}\cdots\right.\\ &\quad \left. + \frac{x^{k+1}}{(k+1)!}\frac{d^kv}{dx^k}\bigg|_{x_0} + \cdots\right]_0^{x1-x0} = \frac{1}{x_2 - x_1}\left((x_2 - x_1)v(x_0)\right.\\ &\quad + \frac{(x_2 - x_0)^2 - (x_1 - x_0)^2}{2}\frac{dv}{dx}\bigg|_{x_0} + \frac{(x_2 - x_0)^3 - (x_1 - x_0)^3}{6}\frac{d^2v}{dx^2}\bigg|_{x_0}\\ &\quad \left. + \cdots + \frac{(x_2 - x_0)^{k+1} - (x_1 - x_0)^{k+1}}{(k+1)!}\frac{d^kv}{dx^k}\bigg|_{x_0} + \cdots\right) = v(x_0)\\ &\quad + \frac{(x_2 - x_0)^2 - (x_1 - x_0)^2}{2(x_2 - x_1)}\frac{dv}{dx}\bigg|_{x_0} + \frac{(x_2 - x_0)^3 - (x_1 - x_0)^3}{6(x_2 - x_1)}\frac{d^2v}{dx^2}\bigg|_{x_0}\\ &\quad + \cdots + \frac{(x_2 - x_0)^{k+1} - (x_1 - x_0)^{k+1}}{(k+1)!(x_2 - x_1)}\frac{d^kv}{dx^k}\bigg|_{x_0} + \cdots\end{aligned} \tag{2}$$

$$\begin{aligned}\bar{v}_{2+1/2} &= v(x_0) + \frac{(x_2 - x_0)^2 - (x_1 - x_0)^2}{2(x_2 - x_1)}\frac{dv}{dx}\bigg|_{x_0}\\ &\quad + \frac{(x_2 - x_0)^3 - (x_1 - x_0)^3}{6(x_2 - x_1)}\frac{d^2v}{dx^2}\bigg|_{x_0} + \cdots\\ &\quad + \frac{(x_2 - x_0)^{k+1} - (x_1 - x_0)^{k+1}}{(k+1)!(x_2 - x_1)}\frac{d^kv}{dx^k}\bigg|_{x_0} + \cdots\end{aligned} \tag{3}$$

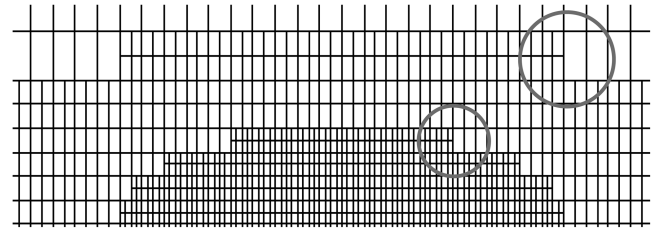


Fig. 1 Grid cell-size changes at the AMR boundary.

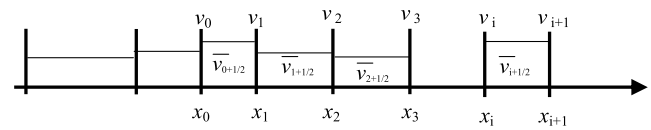


Fig. 2 Irregular-cell-spacing stencil.

$$\begin{aligned}\bar{v}_{3+1/2} &= v(x_0) + \frac{(x_3 - x_0)^2 - (x_2 - x_0)^2}{2(x_3 - x_2)}\frac{dv}{dx}\bigg|_{x_0}\\ &\quad + \frac{(x_3 - x_0)^3 - (x_2 - x_0)^3}{6(x_3 - x_2)}\frac{d^2v}{dx^2}\bigg|_{x_0} + \cdots\\ &\quad + \frac{(x_3 - x_0)^{k+1} - (x_2 - x_0)^{k+1}}{(k+1)!(x_3 - x_2)}\frac{d^kv}{dx^k}\bigg|_{x_0} + \cdots\end{aligned} \tag{4}$$

$$\begin{aligned}\bar{v}_{i+1/2} &= v(x_0) + \frac{(x_i - x_0)^2 - (x_{i-1} - x_0)^2}{2(x_i - x_{i-1})}\frac{dv}{dx}\bigg|_{x_0}\\ &\quad + \frac{(x_i - x_0)^3 - (x_{i-1} - x_0)^3}{6(x_i - x_{i-1})}\frac{d^2v}{dx^2}\bigg|_{x_0} + \cdots\\ &\quad + \frac{(x_i - x_0)^{k+1} - (x_{i-1} - x_0)^{k+1}}{(k+1)!(x_i - x_{i-1})}\frac{d^kv}{dx^k}\bigg|_{x_0} + \cdots\end{aligned} \tag{5}$$

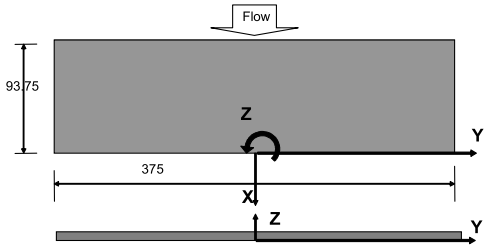
$$\begin{aligned}\bar{v}_{i+1/2} &= v(x_0) + \frac{\Delta x}{2}(2i - 1)\frac{dv}{dx}\bigg|_{x_0} + \frac{\Delta x}{6}(3i^2 - 3i + 1)\frac{d^2v}{dx^2}\bigg|_{x_0}\\ &\quad + \cdots + \frac{\Delta x}{(k+1)!}\{i^{k+1} - (i-1)^{k+1}\}\frac{d^kv}{dx^k}\bigg|_{x_0} + \cdots\end{aligned} \tag{6}$$

Although the ENO scheme achieves the rth order of accuracy, the WENO scheme achieves the $(2r - 1)$th order using the weighted average of ENO interpolation results. The weight function for this averaging is so complicated that the weight function cannot be evaluated for irregular-spacing cells. Hence, we can achieve the $(2r - 1)$th order by the WENO interpolation in a uniform-cell-spacing region and the rth order at an AMR-grid refinement-level boundary, which contains an irregular-spacing stencil. In this paper, rth order is used to describe the scheme accuracy instead of $(2r - 1)$ th from a conservative standpoint. Three cases of accuracy order $(r = 2, 3, 4)$ are compared to show the benefit of using a higher-order method, which correspond to $(2r - 1)$ equal to third, fifth, and seventh order in the uniform-spacing region, respectively.

A three-stage Runge–Kutta-type explicit method is used for time integration. Viscous flux is calculated by second-order central difference. No turbulence model was applied in the present computation. Only the molecular viscosity is calculated.

C. Parallel Computation

Parallel computation is one of the key features of the CHARGE code. Unsteady computation demands extensive computing resources because of a large number of grid points and time steps



thickness: t=8.0mm
unit: mm

Fig. 3 Flat-plate model geometry.

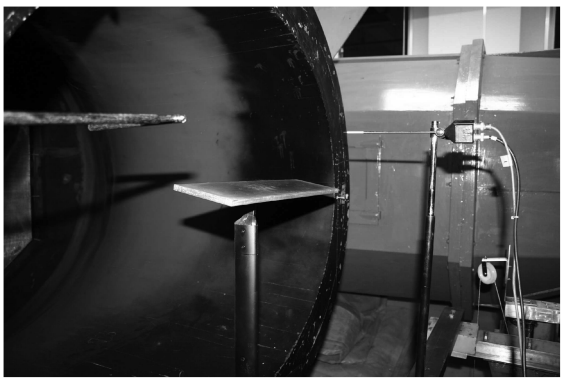


Fig. 4 Model set in the low-speed wind tunnel.

to preserve vortical flow physics. Two stages of parallelization are applied in this code. The first stage is a multithread approach for inside-node parallelization such as multicore/multi-CPU architecture; the OpenMP library is used for this purpose. About 20 OpenMP directives are inserted in the program code.

As a second stage, the MPI library is used for internode parallel processing, and a domain decomposition approach is employed to distribute the workload. As mentioned earlier, the volume grid is divided into cell clusters as a preprocess. Domain decomposition is conducted at the beginning of solver execution by distributing cell clusters to all processes.

Currently, parallel computation using 16 CPU cores achieves speedup of 14 times by using a gigabit Ethernet-based Linux PC cluster system. For the parallel case using 64 CPU cores, the speedup ratio was 50.

III. Wind-Tunnel Test

Wind-tunnel tests were conducted at the low-speed wind-tunnel facility at the Department of Aerospace Engineering at Nagoya University. The tunnel is a closed-circuit low-speed tunnel with an open-air-type test section. The cross section takes an elliptical shape, the size of which is approximately 900 mm (width) by 750 mm (height).

Table 1 Wind-tunnel test conditions

Freestream velocity	20 m/s (nominal)
Angle of attack	10 deg
Reference length	93.75 mm (chord length C)
Reynolds number	123,760, 20 m/s

An aluminum flat plate is used in the experiment. The model size is 375 mm (span) by 93.75 mm (chord) by 8 mm (thickness), as shown in Fig. 3. The aspect ratio is 4.0 and the thickness ratio 8.53%. The photograph of the model installed on the tunnel is shown in Fig. 4. The plate is supported at the lower surface. The spanwise location of the support is slightly offset to obtain a clean flow near the centerline. The angle of attack is fixed to 10 deg and the freestream velocity is 20 m/s. Experimental conditions are summarized in Table 1. Although the Reynolds number is not very high (about 0.1 million), it is enough to examine the fundamental behavior of the CHARGE code.

An X-probe hot-wire anemometer was used to measure the flow velocity. The mean velocity $(\bar{u}, \bar{v}, \bar{w})$, the root mean square (rms) of perturbation velocity (u', v', w'), and the frequencywise power spectral density of velocity were measured.

The mean velocity vectors and the rms values were measured at five downstream stations: $x/C = 0$ (just behind the trailing edge), 1, 2, 3, and 4, where C is the chord length. The spanwise and vertical data acquisition spacing are $0.05C$ for $x/C = 0$, 1 stations and $0.10C$ for $x/C = 2$, 3, 4 stations. Unsteady data were measured at nine locations at each x station, as shown in Fig. 5, and a 100 kHz sampling-rate acquisition was applied for unsteady hot-wire anemometry measurements.

IV. Numerical and Experimental Results and Discussion

A flat-plate wake was used to examine the capability of the CHARGE code to capture unsteady flowfield characteristics. The values of the scheme accuracy and grid parameters were varied to observe the effect on CFD results. The primary objective is the comparison of scheme accuracy. As a secondary objective, grid density (baseline and coarse) and symmetry-plane treatment (half-span with a symmetry boundary condition and full-span calculation) were investigated. The grid spacing of the coarse grid has a 20% larger spacing than that of the baseline grid in all directions. Table 2 shows the conditions used in this analysis. The cross section of the baseline half-span grid is shown in Figs. 6 and 7. The grid is refined in the wake and shear flow region by user input to capture more accurate

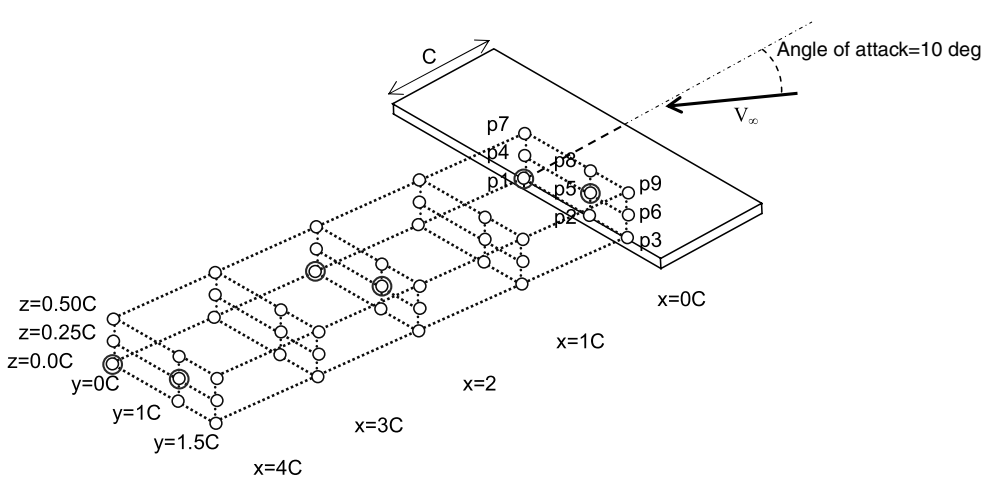


Fig. 5 Measurement points for unsteady data; data at the double circle points are used in Figs. 15 and 16.

Table 2 CFD analysis condition

Freestream velocity	20 m/s (nominal)
Reynolds number	123,760 (20 m/s), 61,880 (10 m/s)
Angle of attack	10 deg
Reference length	93.75 mm (chord length)
Time step	1.0e–4 (nondimensional), approximately equivalent to a Courant–Friedrichs–Lewy number of 0.25, corresponding to real wall-clock time of 1e–6 s
Grid	
Baseline	Baseline grid density, half-span domain, 2.1 million cells
Coarse	Coarse grid density, half-span domain, 1.6 million cells
Baseline full span	Baseline grid density, full-span domain, 4.2 million cells
Scheme accuracy	
Spatial	WENO scheme; second (third) order, third (fifth) order, fourth (seventh) order; ( ) in uniform-cell region
Time	Three-stage Runge–Kutta; third-order

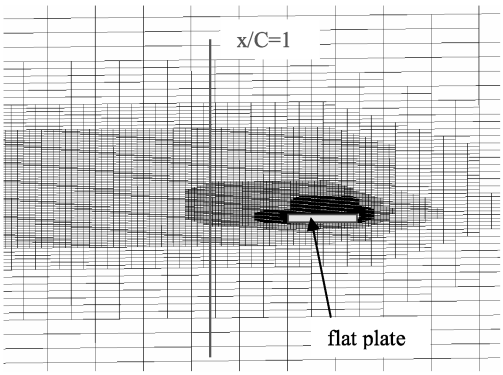


Fig. 6 Grid symmetry plane at $y/C = 0$.

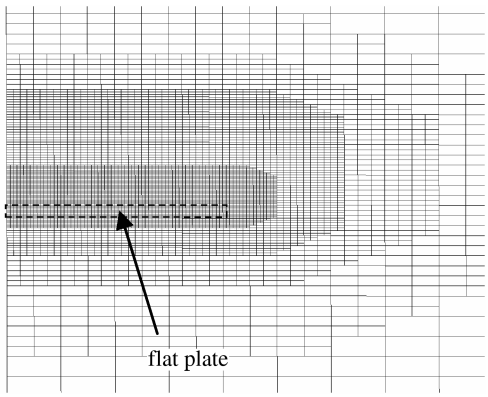


Fig. 7 Grid cross section at $x/C = 1$.

flow characteristics using the octree AMR method. The instantaneous streamwise velocity component u in the computed flowfield using the WENO third-order scheme is shown in Fig. 8. The wake field is well captured by the present code. Details are described in the next section.

A. Mean Flow Characteristics of the Wake

Figures 9–11 show comparisons of mean velocity $(\bar{u}, \bar{v}, \bar{w})$ between CFD and experiment at three x stations ($x/C = 0$, 2, 4). Contour lines show mean streamwise velocity $\bar{u}$. Vectors show $\bar{v}$ and $\bar{w}$. The averaging time period for CFD results, converted to real-time scale, is about 50 ms, which corresponds to 1 m ($10C$) convection by the freestream. The averaging time period for experiment is about 1 s. The overall agreement between CFD and experiment is quantitatively good. In Fig. 9 ($x/C = 0$), all velocity profiles are almost identical. In Fig. 10 ($x/C = 2$), CFD well captures the wake shape such as tip vortex roll-up, whereas the second-order result shows a slightly dissipated profile. In Fig. 11 ($x/C = 4$), the tip vortex by the CFD looks larger than that of the experiment due to the dissipation caused by numerical viscosity. The second-order result is the most dissipated result, and the third and fourth orders showed similar results. In spite of a 20% larger grid spacing for all directions, the coarse grid result is almost the same result as the baseline grid result. The difference between the half-span grid result and the full-span grid result is negligible.

B. Unsteady Flow Characteristics of the Wake

The rms values of the fluctuation velocity $u' = u - \bar{u}$ are compared in Figs. 12–14. Although the CFD and the wind-tunnel test data show similar distributions near the trailing edge ($x/C = 0$), the far-field CFD results ($x/C = 2$, 4) show much smaller perturbations

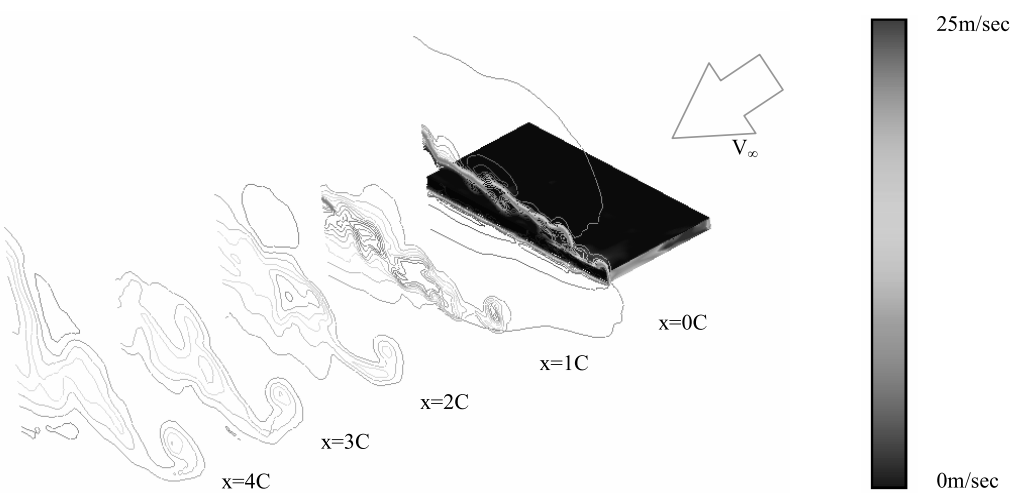
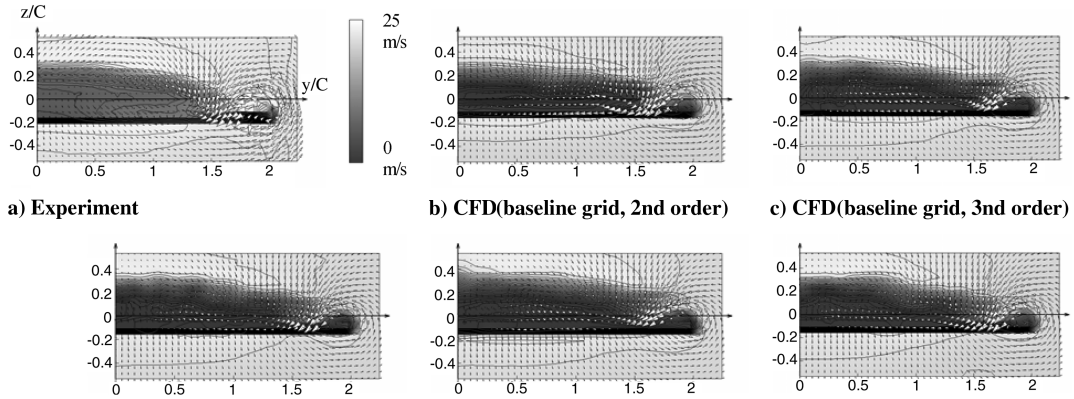


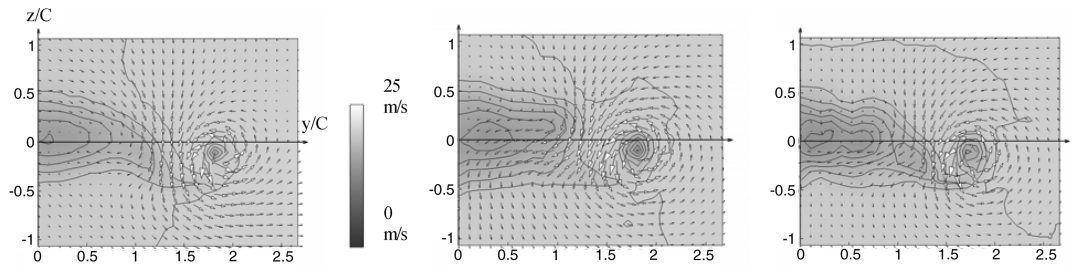
Fig. 8 Instantaneous U velocity distributions (WENO third-order scheme, baseline half-span grid).



a) Experiment b) CFD(baseline grid, 2nd order) c) CFD(baseline grid, 3nd order)

d) CFD(baseline grid, 4th order) e) CFD(coarse grid, 3rd order) f) CFD(full span grid, 3rd order)

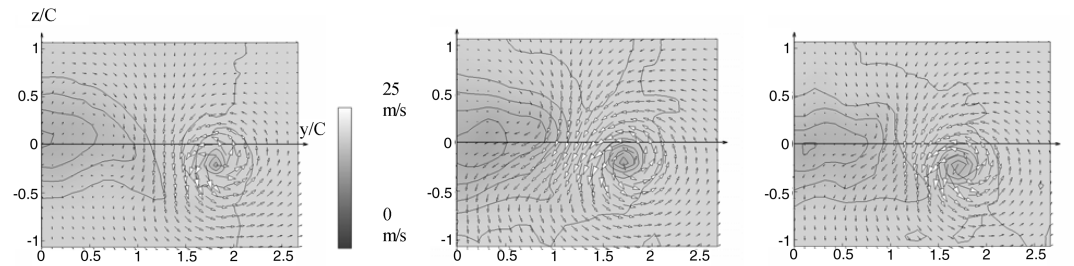
Fig. 9 Mean streamwise velocity contours and in-plane mean velocity vectors at $x/C = 0$.



a) Experiment b) CFD(baseline grid, 2nd order) c) CFD(baseline grid, 3rd order)

d) CFD(baseline grid, 4th order) e) CFD(coarse grid, 3rd order) f) CFD(full span grid, 3rd order)

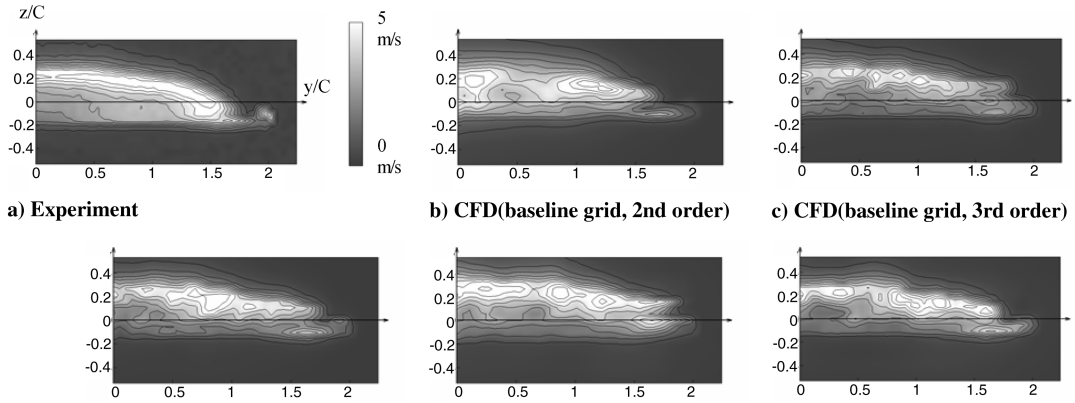
Fig. 10 Mean streamwise velocity contours and in-plane mean velocity vectors at $x/C = 2$.



a) Experiment b) CFD(baseline grid, 2nd order) c) CFD(baseline grid, 3rd order)

d) CFD(baseline grid, 4th order) e) CFD(coarse grid, 3rd order) f) CFD(full span grid, 3rd order)

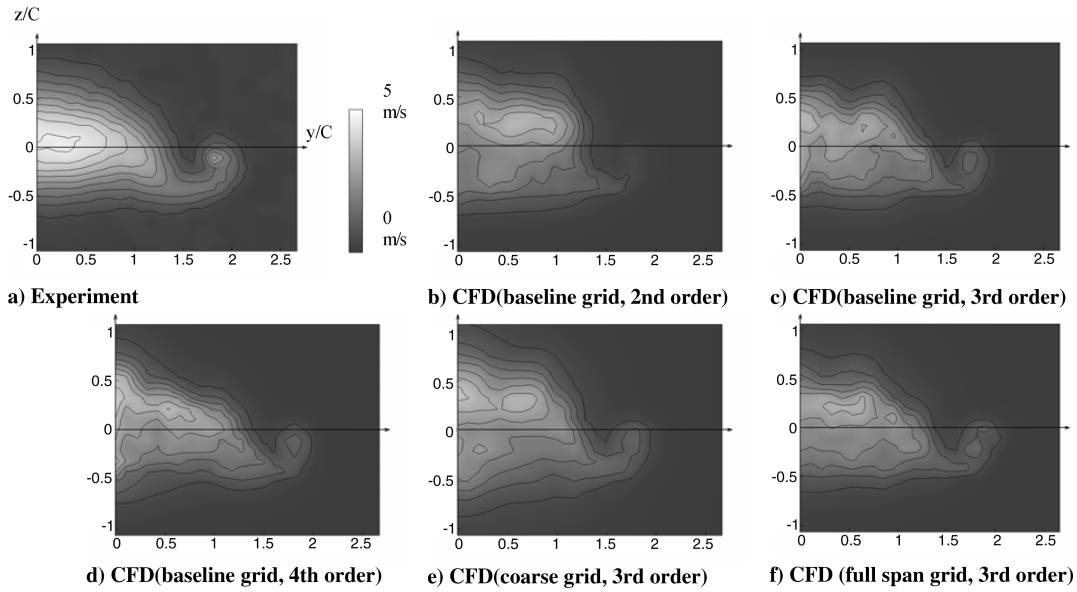
Fig. 11 Mean streamwise velocity contours and in-plane mean velocity vectors at $x/C = 4$.



a) Experiment b) CFD(baseline grid, 2nd order) c) CFD(baseline grid, 3rd order)

d) CFD (baseline grid, 4th order) e) CFD(coarse grid, 3rd order) f) CFD(full span grid, 3rd order)

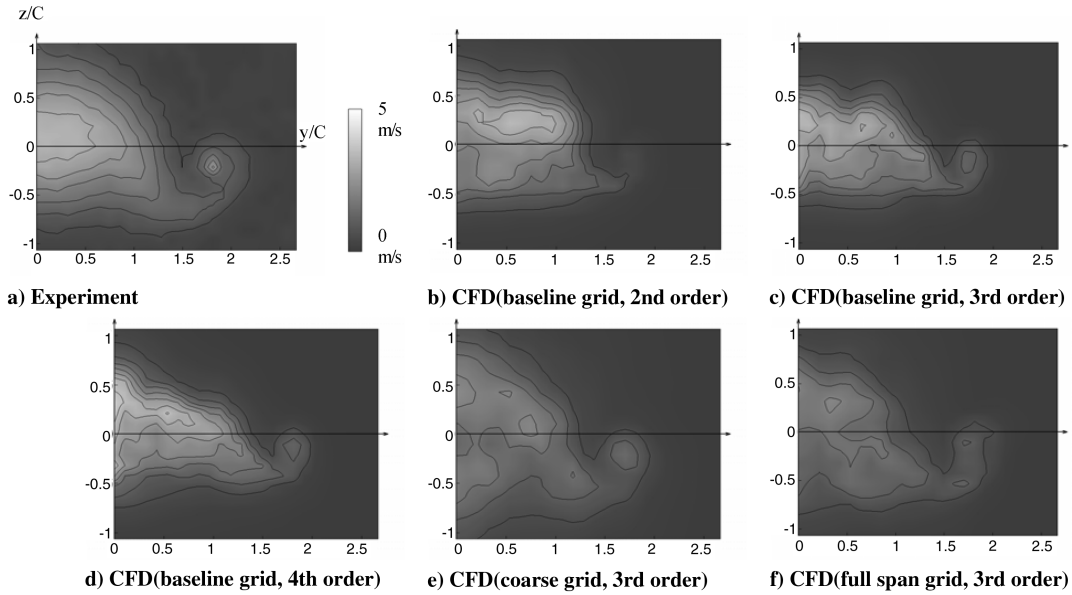
Fig. 12 RMS of fluctuation velocity $u' = u - \bar{u}$ at $x/C = 0$.



a) Experiment b) CFD(baseline grid, 2nd order) c) CFD(baseline grid, 3rd order)

d) CFD(baseline grid, 4th order) e) CFD(coarse grid, 3rd order) f) CFD (full span grid, 3rd order)

Fig. 13 RMS of fluctuation velocity $u' = u - \bar{u}$ at $x/C = 2$.



a) Experiment b) CFD(baseline grid, 2nd order) c) CFD(baseline grid, 3rd order)

d) CFD(baseline grid, 4th order) e) CFD(coarse grid, 3rd order) f) CFD(full span grid, 3rd order)

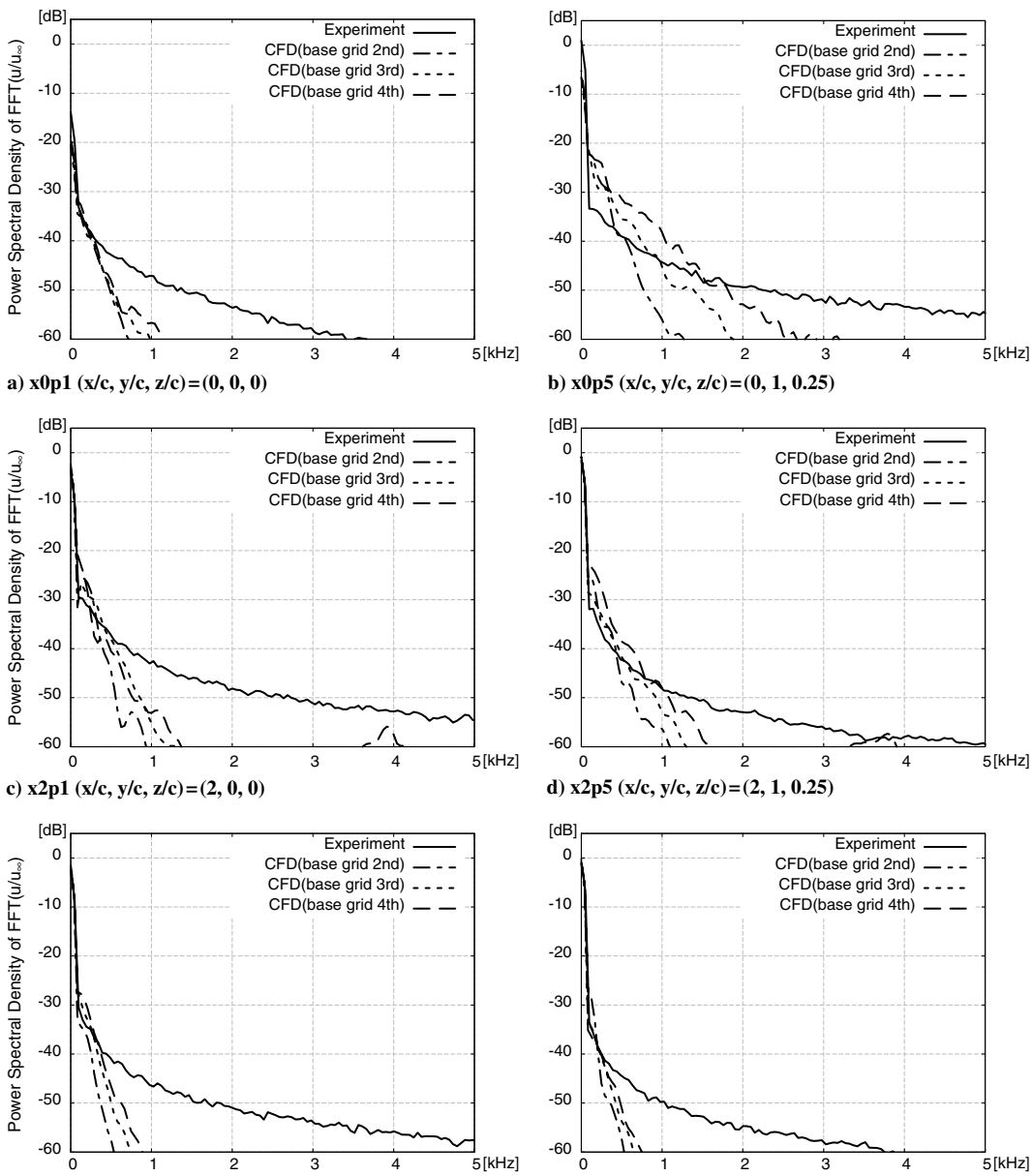
Fig. 14 RMS of fluctuation velocity $u' = u - \bar{u}$ at $x/C = 4$.

because high-frequency perturbations decayed at the far field. However, an outline of the perturbation area is fairly captured by the third- and fourth-order-accuracy schemes. In terms of the wake shape and velocity fluctuation, third- and fourth-order schemes showed comparable results. The grid influence is not obviously similar to the average velocity result.

Figures 15 and 16 show the frequency analysis of the streamwise velocity component. The horizontal axis is a frequency in hertz. The vertical axis is a power spectral density of the streamwise velocity normalized by freestream velocity. Accordingly, power = 0 [dB] at frequency = 0 [Hz] indicates that mean velocity is the same as freestream velocity. For instance, power = −20 [dB] indicates that velocity is a tenth of freestream velocity. Two locations (p1 and p5) and three chordwise stations ($x/C = 0, 2, 4$), indicated in Fig. 5, are selected as typical examples.

The frequency analysis for the three different orders of scheme accuracy is plotted in Fig. 15. The CFD results calculated by the baseline grid are shown in Fig. 15. The CFD results captured unsteady frequency properties up to 500 or 1000 Hz, depending on the x station and probe position. The high-frequency components of the CFD result are faded away with an increase of the distance from the trailing edge. Second-order results (short dashed line) show the most attenuated distributions in general. Fourth-order results (solid line) maintain the largest unsteady component. The difference between the third- (long dashed line) and fourth-order schemes is not very large.

In spite of loss at the high-frequency region, low-frequency characteristics are well captured. The frequencies of 500 and 1000 Hz correspond to the vortex sizes of 40 and 20 mm. The grid spacing is 8 mm ($x/C > 1.5$), 4 mm ($0.1 < x/C < 1.5$), and 2 mm ($x/C < 0.1$). Thus, the current CHARGE code needs five grid points to capture the vortex. Although an ordinary fourth-order scheme needs more than five grid points to capture a vortex, the fourth order in the present code has seventh-order accuracy in the uniform-cell-size region, as mentioned in Sec. II.B. Therefore, the present code achieved five grid points to capture the vortex, which is a minimum number of points to represent a wave. Hence, the grid resolution determines the upper-side limit of the unsteady frequency. Usually, 500 Hz will be enough for an aerodynamic force estimation, whereas aeroacoustic computations need much higher-frequency resolution,



a) x0p1 (x/c, y/c, z/c) = (0, 0, 0)

b) x0p5 (x/c, y/c, z/c) = (0, 1, 0.25)

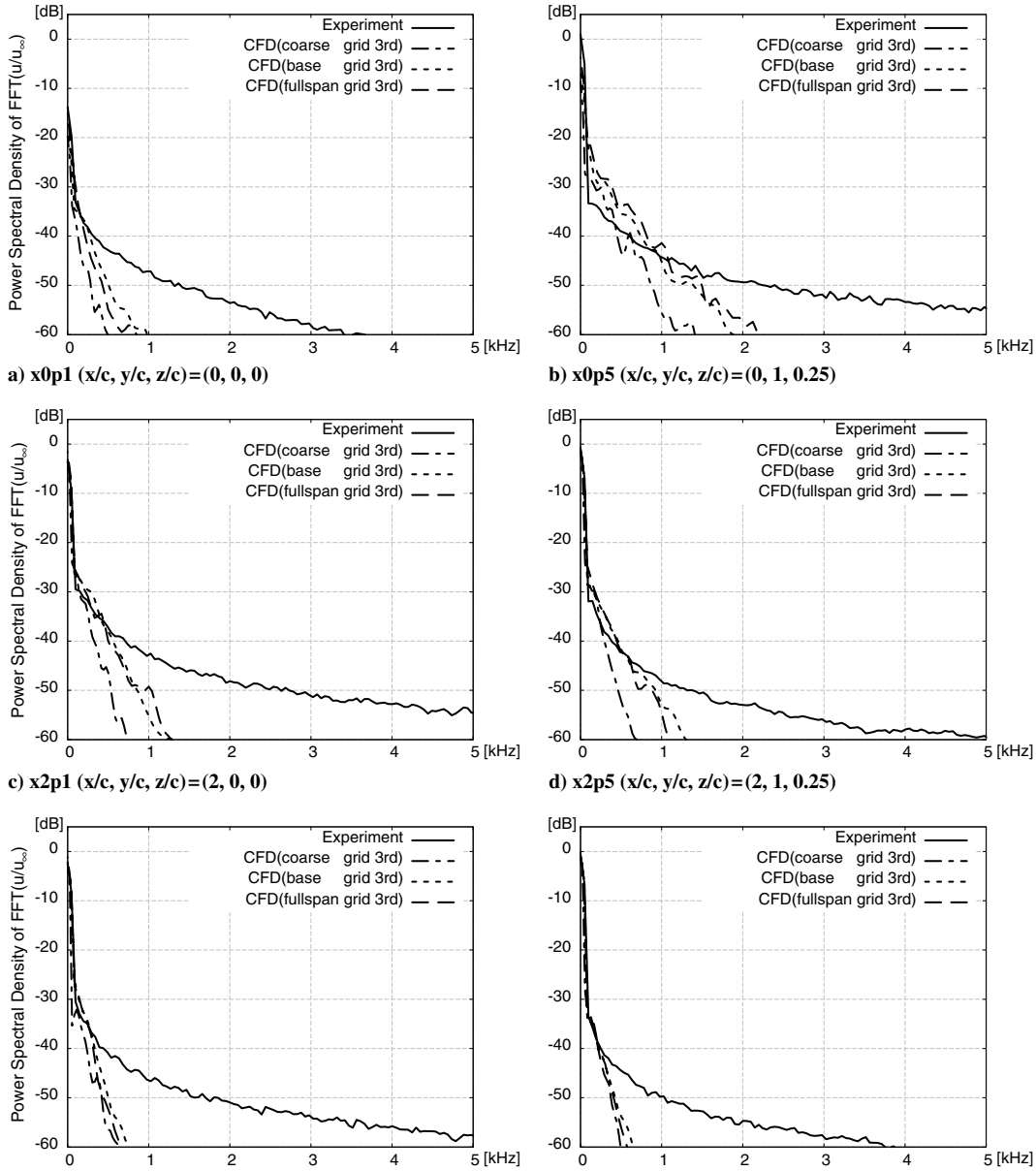
c) x2p1 (x/c, y/c, z/c) = (2, 0, 0)

d) x2p5 (x/c, y/c, z/c) = (2, 1, 0.25)

e) x4p1 (x/c, y/c, z/c) = (4, 0, 0)

f) x4p5 (x/c, y/c, z/c) = (4, 1, 0.25)

Fig. 15 Frequency analysis of streamwise velocity for three orders of scheme accuracy (second, third, fourth) at $x/C = 0, 2, 4$.



a) x0p1 (x/c, y/c, z/c) = (0, 0, 0)

b) x0p5 (x/c, y/c, z/c) = (0, 1, 0.25)

c) x2p1 (x/c, y/c, z/c) = (2, 0, 0)

d) x2p5 (x/c, y/c, z/c) = (2, 1, 0.25)

e) x4p1 (x/c, y/c, z/c) = (4, 0, 0)

f) x4p5 (x/c, y/c, z/c) = (4, 1, 0.25)

Fig. 16 Frequency analysis of streamwise velocity for three grid systems (coarse, base, full span) at $x/C = 0, 2, 4$.

which means that much more grid is required. In some positions, CFD shows larger fluctuation components than the experiment. A possible cause is that the wake region is spread by the lack of grid resolution.

Figure 16 is a comparison among the grid systems. Three types of grid are compared: coarse half-span, baseline half-span, and baseline full-span. The difference of grid density is distinct in Figs. 16b–16d, in which fluctuation component is large. It is interesting that grid spacing substantially affects the unsteady component, in contrast to the average wake velocity. The half-and full-span results are almost identical, considering the scatter of data.

V. Conclusions

A validation study was conducted for the CHARGE code, which uses the higher-order WENO scheme on the Cartesian AMR grid, by using wake flowfield of the simple flat plate. The mean and unsteady velocity computed by three different orders of scheme accuracy and grid system were compared with experimental results. The mean velocities of wake are very well captured by the present code among all orders of accuracy and grids. It was found that the current CHARGE code with fourth- and seventh-order accuracy needs five grid points to capture the vortex through the far field. Although grid density was not very sensitive to mean flow characteristics, the unsteady component was substantially affected. The capability of the present code to maintain unsteady flow characteristics is well demonstrated, whereas high-frequency perturbation was dissipated because of the lack of grid resolution.

References

[1] Johnson, F. T., Tinoco, E. N., and Yu, N. J., "Thirty Years of Development and Application of CFD at Boeing Commercial Airplanes," *Computers and Fluids*, Vol. 34, No. 10, 2005, pp. 1115–1151.
doi:10.1016/j.compfluid.2004.06.005

[2] Aftosmis, M. J., *Solution Adaptive Cartesian Grid Methods for Aerodynamic Flows with Complex Geometries*, VKI Lecture Series 1997–02, Von Karman Institute, Rhode-Saint-Genèse, Belgium, 1997.

[3] Jiang, G., and Shu, C., "Efficient Implementation of Weighted ENO Schemes," *Journal of Computational Physics*, Vol. 126, No. 1, 1996,

pp. 202–228.
doi:10.1006/jcph.1996.0130

[4] Balsara, D., and Shu, C., "Monotonicity Preserving Weighted Essentially Non-Oscillatory Schemes with Increasingly High Order of Accuracy," *Journal of Computational Physics*, Vol. 160, No. 2, 2000, pp. 405–452.
doi:10.1006/jcph.2000.6443

[5] Cockburn, B., and Shu, C. W., "The Runge-Kutta Discontinuous Galerkin Method for Conservation Laws 5: Multidimensional Systems," *Journal of Computational Physics*, Vol. 141, No. 2, April 1998, pp. 199–224.
doi:10.1006/jcph.1998.5892

[6] Wang, Z. J., "Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids: Basic Formulation," *Journal of Computational Physics*, Vol. 178, No. 1, May 2002, pp. 210–251.
doi:10.1006/jcph.2002.7041