

Scalability of the parallel CFD simulations of flow past a fluttering airfoil in OpenFOAM

Petr Šidlof^{1a}, Václav Řídký¹

¹Technical University of Liberec, NTI FM, Studentská 2, 461 17 Liberec, Czech Republic

Abstract. The paper is devoted to investigation of unsteady subsonic airflow past an elastically supported airfoil during onset of the flutter instability. Based on the geometry, boundary conditions and airfoil motion data identified from wind-tunnel measurements, a 3D CFD model has been set up in OpenFOAM. The model is based on incompressible Navier-Stokes equations. The turbulence is modelled by the Menter's k-omega shear stress transport turbulence model. The computational mesh was generated in GridPro, a mesh generator capable of producing highly orthogonal structured C-type meshes. The mesh totals 3.1 million elements. Parallel scalability was measured on a small shared-memory SGI Altix UV 100 supercomputer.

1 Introduction

In aerospace engineering, fluid-structure interaction can play a very important and potentially dangerous role: under certain circumstances, the coupling between flow and structure may lead to unstable exponentially increasing oscillations. The classical example is the flutter instability of airfoils, which occurs for systems with two degrees of freedom when the critical flow velocity is surpassed [1]. Flutter is a dynamic instability of an elastic structure coupled to airflow, caused by the interaction between elastic, inertial and aerodynamic forces. Flutter may occur at certain flow velocities and structural natural vibration frequencies, when the energy is transferred from the airflow to the structure and the internal damping is not able to absorb it. This can lead to vibration of an aircraft component with exponentially increasing amplitudes and catastrophic consequences. Thus, the aircraft components (wings, flaps and ailerions, stabilators, elevons and rudders) must be designed and tested not only to sustain dynamic loads induced by mechanical accelerations and air turbulence during take-off, cruise and landing, but also to ensure that within design flight conditions, the aircraft components may never encounter dynamic flow-induced instability.

There are more types of aeroelastic instabilities, e. g. coupled-mode flutter, panel flutter, stall flutter, buffeting or galloping. In general, flutter instability is a complex nonlinear phenomenon which is not yet fully understood in all the aspects. In current days, the flight flutter testing is performed at a number of discrete test points arranged in increasing order of dynamic pressure and airflow velocity. The number of these test points required to clear

the flutter envelope of an airplane is quite high - for example about 500 for the F-14 fighter plane, or 260 for the Gulfstream II civil airplane [2]. In between the test points, the data is interpolated, which might be questionable considering the high nonlinearity of the flutter phenomena.

In exceptionally rough weather conditions or in some cases of gross human error during aircraft service, the airplane may still encounter the instable regime during flight. This is why the aircraft disasters caused by flutter-induced structural disintegration are not limited to the early days of aviation (e.g. the Cody Floatplane Verona airliner crash in 1919, Lockheed L-188 Electra flight 542 crash in 1959, or Northwest Orient Airlines Flight 710 disaster in 1960, all of them killing everyone aboard), but also to recent history. A notable example is the 1997 Maryland airshow accident, where a F117 stealth jet fighter crashed before the eyes of the spectators after a part of its left wing broke off the fuselage due to elevon-wing flutter caused by missing fasteners.

In computational studies of airflow, a common approach is to simplify the problem by considering forced, prescribed vibrations of the airfoil (see e.g. [3-7]). This is the case of the current study, too. The CFD simulations of 3D flow past a vibrating airfoil are computationally demanding, and need to be run in parallel. The current paper summarizes the parallel scaling performance of the CFD simulations run in OpenFOAM on a SGI Altix-UV shared-memory supercomputer.

^a Corresponding author: petr.sidlof@tul.cz

2 CFD model

The geometry of the computational model was specified according to the experimental setup – a model of a NACA0015 airfoil vibrating in a test section of a suction-type wind tunnel. Details regarding the experimental setup are given in [8], the mathematical model for the CFD simulation is described in [9]. The airfoil has two degrees of freedom: vertical motion (plunge) and rotation (pitch) about the elastic axis, which is located in 1/3 of the airfoil chord. The computational domain and the mesh are shown in figure 1. The dimensions of the computational domain are 580x210x80 mm, the airfoil has a chord length of 65 mm. The 2D quadrilateral mesh consisting of 70k elements was generated in GridPro, software capable of producing structured, highly orthogonal C-type meshes. The 2D mesh was then extruded into 3D in 40 layers. The grid totals 3.1 million hexahedral elements, with a refined layer adjacent to the wing surface. The wall-normal distance of the first meshpoint off the wing is 0.1 mm.

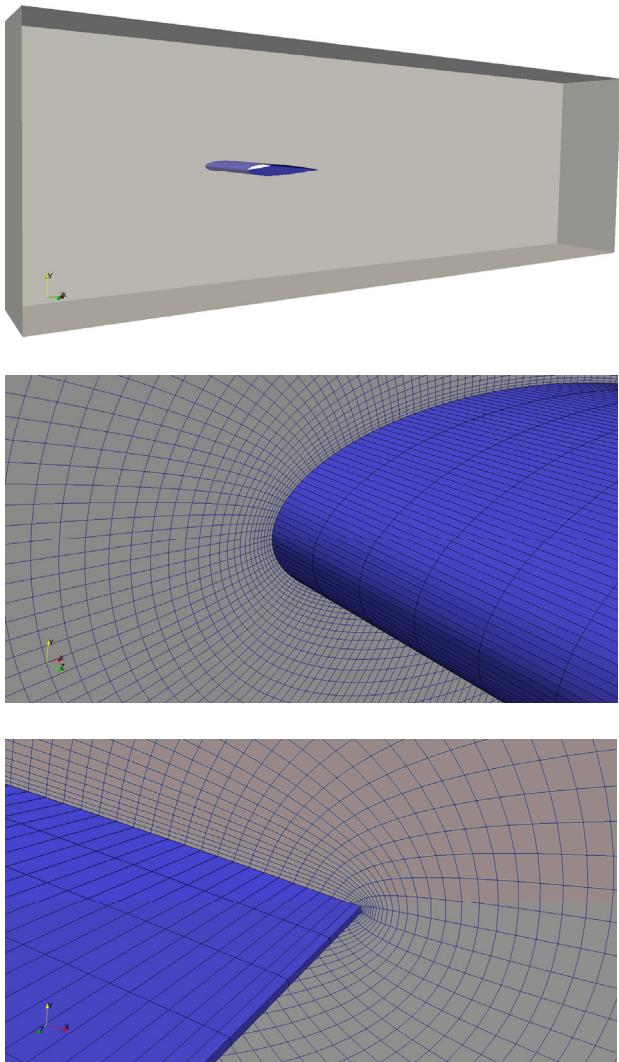


Figure 1. The computational domain (top) with the airfoil in blue, detail of the computational mesh near the airfoil leading edge (middle) and trailing edge (bottom).

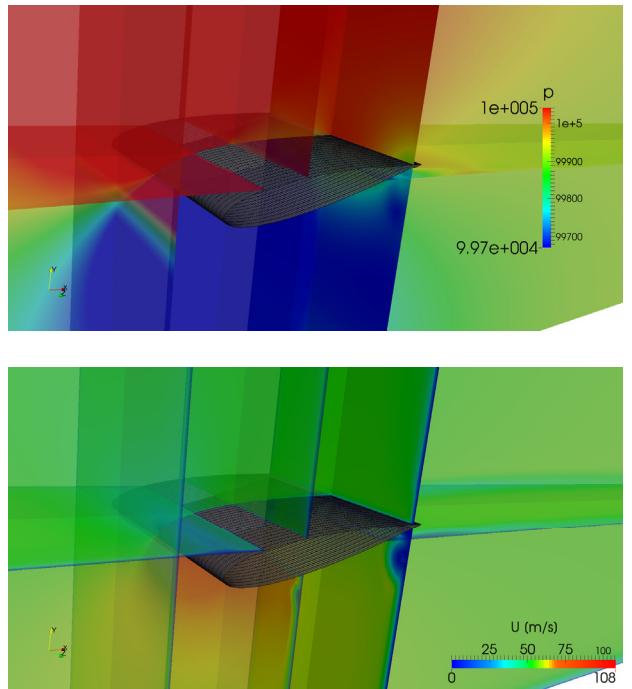


Figure 2. Pressure and velocity field in the vicinity of the oscillating airfoil. Time 0.117 s.

The computational domain is deformed in time due to airfoil motion, which was identified from the wind-tunnel measurement No. 2810-2-45 [8]. The motion in both modes is approximately sinusoidal, with a frequency $f = 15.5$ Hz, semi-amplitude of the plunging mode $A_y = 3.17$ mm, amplitude of the pitching mode $A_\phi = 9.9^\circ$ and phase shift between the pitch and plunge $\psi = -1.178 \pi$:

$$\begin{aligned} y(t) &= A_y \sin(2\pi f t) \\ \phi(t) &= A_\phi \sin(2\pi f t + \psi) \end{aligned} \quad (1)$$

The numerical solution of the fluid flow is performed in OpenFOAM, an open-source CFD library based on cell-centred finite volume method. For the medium-speed flow, the incompressible Navier-Stokes equations were used in conjunction with the $k-\omega$ SST turbulence model [10]. The boundary conditions for the turbulence model were calculated from the inflow velocity 78.5 m/s and turbulence intensity at the inlet to the wind-tunnel, which was estimated from experience to 5%. The boundary conditions for the velocity are $U = 147$ m/s at inlet and no-slip condition on the upper and lateral walls. On the moving airfoil, the flow velocity is equal to the velocity of the structure. The pressure is set to atmospheric at inlet. On the walls and outlet of the domain, Neumann condition for the pressure is prescribed.

Figure 2 shows the results of the transient simulation at time $t = 0.117$ s, which corresponds to the third period of vibration and the time instant, where the pitching angle reaches maximum. The high-pressure zone is on the upper airfoil surface, where stagnation occurs. The lower surface, on the contrary, accelerates the airflow and creates a zone of low pressure. The mean flow field is not separated, which might be attributed to the turbulence

model. Laminar simulations suggest that for higher pitching angles the airflow separates, especially in the phases where the airfoil moves towards the high-pressure zone.

3 Parallel scaling tests

For parallelization of the CFD simulations, OpenFOAM employs domain decomposition method. The mesh was decomposed using the Scotch algorithm [11], see figure 3.

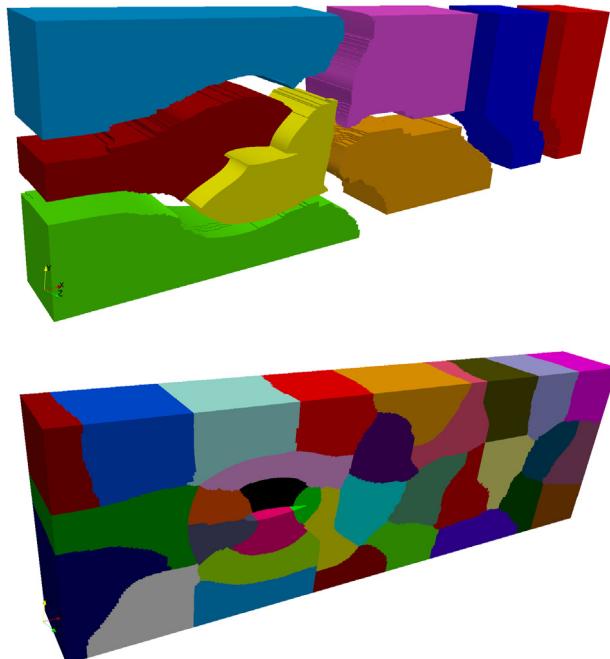


Figure 3. Decomposition of the computational domain into 8 and 32 subdomains using the scotch method

The simulations were run on a parallel computer SGI Altix UV 100, located at the Supercomputing Centre of the Czech Technical University in Prague. This shared-memory machine is built on the cache coherent non-uniform memory access (cc-NUMA) architecture and offers 12 6-core Intel Xeon Nehalem processors, with 8 GB per core RAM. All the 576 GB of the memory can be allocated by any processor. The nodes are interconnected by the SGI NUMalink 5 interconnects, providing low latency and 15GB/s bandwidth through two 7.5 GB/s unidirectional links. The OpenFOAM library is compiled using standard gcc, and the parallel solvers run via SGI MPT implementation of the MPI standard.

The scaling tests were performed on the same machine on 1 – 36 CPU cores. Each test was run for 12 fixed timesteps, and the computational time was evaluated between the second and 11th timestep to eliminate the influence of initialization routines and I/O during loading of the mesh from the storage device. The results of the strong scaling tests, plotted in figure 4, show that the parallel speedup increases up to about 32 computational cores, where the empirically known limit 100k elements per core is surpassed. Further, the speedup

stalls due to the fact that interprocessor communication on the processor boundaries exceeds the actual communication.

4 Discussion and conclusions

A parallelized 3D numerical simulation of flow past an oscillating airfoil has been developed. The motion of the airfoil and the boundary conditions match the values from experiments on a physical model of a self-oscillating airfoil with two degrees of freedom in a wind tunnel. The airflow is modelled by incompressible Navier-Stokes

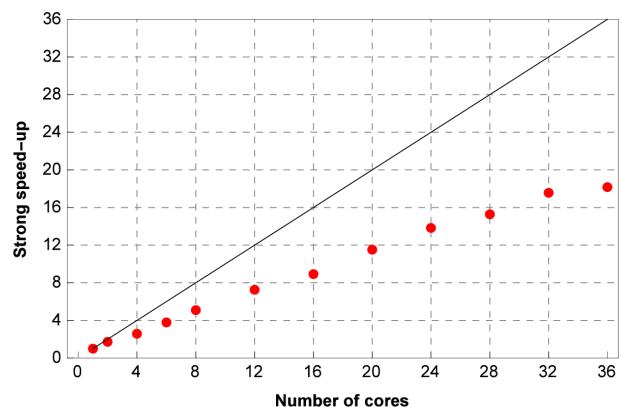


Figure 4. Strong speedup on the SGI Altix UV shared-memory supercomputer. Fixed job size – mesh with 3.1 million elements.

equations, which are on the verge of validity for the given airflow velocities. The compressible CFD model is currently under development.

Before proceeding to large transient simulations, parallel speedup tests have been performed on a small shared-memory supercomputer. As far as the number of elements per core does not drop below about 100 000, the code scales well, although the scaling is quite far from linear speedup. In previous parallel simulations of the authors in OpenFOAM on a different application, the speedup was nearly linear. Before proceeding to massively parallel simulations on large 3D meshes, it would be appropriate to locate the parallel bottleneck and improve the scalability of the parallel speedup.

Acknowledgements

The research has been supported by the Czech Science Foundation, project 13-10527S "Subsonic flutter analysis of elastically supported airfoils using interferometry and CFD". The parallel scaling tests were run on the facilities of the Supercomputing centre at the Czech Technical University in Prague.

References

1. E.H. Dowell, Kluwer Academic Publishers, Dordrecht (1995)
2. M. W. Kehoe, NASA technical memorandum 4720, National Aeronautics and Space Administration (1995)
3. T. O'Neil, T.W. Strganac, Journal of Aircraft **35**, 4 (1998)
4. Y. Lian, M.V. Ol, W. Shy, AIAA paper 2008-652, (2008)
5. M. R. Soltani, F.R. Marzabadi, Transaction B: Mechanical Engineering **17**, 6 (2010)
6. T. Prangemeier, D. Rival, C. Tropea, Journal of Fluids and Structures **26** (2010)
7. I. Fenercioglu, O. Cetiner, Journal of Fluids and Structures **31** (2012)
8. P. Šidlof, V. Vlček, M. Štěpán, J. Horáček, M. Luxa, D. Šimurda, J. Kozánek, Proceedings of the ASME PVP 2014 conference, ASME, 2014
9. P. Šidlof, M. Štěpán, V. Vlček, V. Řídký, D. Šimurda, J. Horáček, EPJ Web of Conferences **67**, 02108 (2014)
10. F. R. Menter, AIAA Journal **32**, 8 (1994)
11. C. Chevailer, F. Pellegrini, Parallel Computing **34** (2008)