

Fully consistent CFD methods for incompressible flow computations

D. K. Kolmogorov¹, W. Z. Shen², N. N. Sørensen³, J. N. Sørensen⁴

^{1 2 4} Technical University of Denmark, Nils Koppels Alle, Building 403, 2800, Kgs. Lyngby Denmark

³ Technical University of Denmark, Frederiksborgvej 399, Building 114, 4000, Roskilde, Denmark

E-mail: ¹dkol@dtu.dk, ²wzsh@dtu.dk, ³nsqr@dtu.dk, ⁴jns@dtu.dk

Abstract.

Nowadays collocated grid based CFD methods are one of the most efficient tools for computations of the flows past wind turbines. To ensure the robustness of the methods they require special attention to the well-known problem of pressure-velocity coupling. Many commercial codes to ensure the pressure-velocity coupling on collocated grids use the so-called momentum interpolation method of Rhie and Chow [1]. As known, the method and some of its widely spread modifications result in solutions, which are dependent of time step at convergence. In this paper the magnitude of the dependence is shown to contribute about 0.5% into the total error in a typical turbulent flow computation. Nevertheless if coarse grids are used, the standard interpolation methods result in much higher non-consistent behavior. To overcome the problem, a recently developed interpolation method, which is independent of time step, is used. It is shown that in comparison to other time step independent method, the method may enhance the convergence rate of the SIMPLEC algorithm up to 25 %. The method is verified using turbulent flow computations around a NACA 64618 airfoil and the roll-up of a shear layer, which may appear in wind turbine wake.

1. Introduction

During the last decade a large effort has gone into developing CFD tools for prediction of wind turbine aerodynamics. In 1999, the flow over the NREL Phase II was computed using overset grid method by Duque [2]. Later, accurate predictions of the NREL Phase IV rotor were performed by Sørensen [3] and Johansen [4] and the stalled rotor prediction was also presented by Duque in [5]. To decrease the computational costs Xu and Sankar proposed in [6] an approach where small zones surrounding each blade were solved, whereas the rest of the domain was treated using a significantly less expensive full potential solver. Rather than modeling the entire rotor Pape in [7] modeled a single blade omitting the tower and nacelle. To predict the rotor-tower interactions, computations of the fully resolved rotors were performed by Zahle using incompressible overset grid method in [8]. Compressible sliding grid method was used by Gomez-Iradi in [9] for computations of the NREL Phase IV rotor. CFD modeling of laminar-turbulent transition for the wind turbine rotors was presented by Sørensen in [10]. In addition to the computations on structured grids computations using unstructured grid solvers were presented by Sezer-Udol in [11] and Potsdam in [12]. Wake simulations behind the wind turbines using CFD methods have been also an intensive field of study. Recent comparison of



CFD wake simulations with the MEXICO experimental data can be found e.g. in Bechmann [13]. To decrease the computational costs, actuator disk and actuator line models have been applied for the wake modeling as can be seen in [14–16].

Nowadays, most of the incompressible CFD tools for wind turbine computations are based on the Semi-Implicit Method for Pressure Linked Equations (SIMPLE) to link the velocity and pressure. To achieve a high robustness of the SIMPLE algorithm there is a common practice to employ it on collocated grids. One of the complexities of the collocated grid-based algorithms is the well known problem of the pressure-velocity decoupling. To overcome the problem, vast majority of the methods used today in commercial codes are based on the so-called momentum interpolation methods, initially proposed by Rhie and Chow [1].

Since the last two decades various modifications of the momentum interpolations have been presented in order to ensure an accurate solution on collocated grids. Originally the Rhie-Chow interpolation was developed for steady flow computations and is known to possess a dependence of velocity underrelaxation parameter at convergence. The problem with the dependence was solved in [17, 18], nevertheless it was later shown in [19, 20] that if the Rhie-Chow interpolation is used for unsteady flow computations, pressure wiggles appear for small time steps. An interpolation method for unsteady flow computations free from the pressure wiggles was later proposed independently by Choi [21] and Shen et al. [19]. Note that the method of Shen et al possesses the same properties as the method of Choi, but contrary to the Choi's method, it is based on second order scheme in time. As shown in [22] both methods of Shen et al and Choi possess a weak dependence of time step and relaxation parameter at convergence. To overcome the difficulty, several methods, which are independent of time step and relaxation parameter, were proposed in [20, 23–25]. Nowadays in spite of the existence of the time step independent methods, the standard methods of Choi and Shen et al are widely used as can be seen in example in [26–29].

On collocated grids both the solution accuracy and the convergence rate of the SIMPLE-like algorithms strictly depend on the choice of the interpolation method. Originally the SIMPLE-like algorithms, such as SIMPLE [30], SIMPLEC [31] and PISO [32], were developed for the staggered grids, where mass flux interpolation is not necessary. In most of the computational codes used in engineering application the choice of the SIMPLE-like algorithm and the choice of the mass flux interpolation method have been done independently [33, 34]. In the current paper, it is shown that using such the approach convergence rate is not always optimal and in order to achieve a high efficiency of the SIMPLE-like algorithms, the mass flux interpolation should be used consistently with the algorithms.

By taking an example of the SIMPLEC algorithm we will demonstrate that the usage of interpolation methods, which are fully compatible with the SIMPLEC algorithm, results in a convergence rate up to 25 % higher than the rate of the standard SIMPLEC algorithm. Theoretical justification of this fact is given in [35], whereas here by using a typical turbulent flow computations we will show that the proper choice of interpolation method may also increase the accuracy of the SIMPLEC algorithm.

Standard interpolation methods on collocated grids, such as the methods of Choi [21] and Shen et al [19], are known to result in solution, dependent of time step at convergence. In this work the magnitude of the dependence is estimated and is shown to be negligible for a typical turbulent flow computation. Nevertheless, when coarse grids are used the inconsistency of the solution becomes non-negligible. To overcome the problem the recently developed Modified Momentum Interpolation method (MMI) of Kolmogorov et al [35], will be used. The efficiency of the standard and the MMI interpolation methods will be tested in the roll-up case of a shear layer vortex, which appears in wind turbine wake. For the standard interpolation methods of Shen [19, 36], the magnitude of the solution dependence of time step will be measured in application to the turbulent flow field around a NACA 64618 airfoil. For the test case the MMI

method is proved to result in the solutions, which are independent of time step at convergence.

2. Discretization of the Navier-Stokes equations

The discretization of the incompressible Navier-Stokes equations for unsteady flow computations on collocated grids is presented below. First, the momentum equations are discretized. Second, the discrete continuity equation is presented with one of the recently developed interpolation methods of Kolmogorov et al in [35].

2.1. Momentum equations

Using the second order backward difference scheme in time and grouping two momentum equations in the x - and the y - directions together, the system of the underrelaxed momentum equations on collocated grids can be expressed for a control volume p in the following form:

$$\widetilde{A}_p \vec{v}_p^{m+1} + \sum_{E,W,N,S} A_{nb} \vec{v}_{nb}^{m+1} + \vec{\nabla} p_p^{m+1} dV_p = \vec{S}_p^m + A_p^V (2\vec{v}_p^m - 0.5\vec{v}_p^{m-1}) \quad (1)$$

where vector \vec{v}_p denotes the vector of the velocity flow field $(u_p, v_p)^T$, whereas vector \vec{S}_p indicates the explicitly treated source terms $(S_p^x, S_p^y)^T$. Terms A_p , A_{nb} are the diagonal and non-diagonal matrix terms accounting for the discrete convective and diffusion terms, the term A_p^V is the coefficient of the time derivative and equal to $\frac{\rho dV_p}{\tau}$, where τ is the time step and ρdV_p is the control volume mass. The term \widetilde{A}_p accounts for the diagonal term of the matrix of the momentum equations, which are underrelaxed using the spatial term A_p similarly to [19, 24, 36] as below:

$$\widetilde{A}_p = A_p / \alpha + 1.5 A_p^V \quad (2)$$

where α is the velocity underrelaxation parameter.

The superscripts m and n are the subiteration- and time step- counters respectively, such that the solution at time step $n + 1$ is obtained at convergence. In order to compute the flow field at the subiteration step $m + 1$, the coefficients A_p , A_{nb} and \widetilde{A}_p are taken from the former subiteration step m . In the notations of the coefficients the superscript counter m is dropped in the following section.

2.2. Mass flux interpolation methods

To ensure flow field continuity the momentum equations in Eq. (1) have to be solved together with the continuity equation:

$$\sum_{e,w,n,s} f_k^{m+1} = 0 \quad (3)$$

which essentially represents the fact that the sum of mass fluxes f_k^{m+1} through the control volume faces k equals to zero.

On collocated grids the cell face fluxes are not available. To identify the fluxes one of the widely spread methods is the momentum interpolation method originally proposed by Rhie and Chow [1]. Nowadays, for unsteady flow computations there is a common practice [26–29] to employ the modified Rhie-Chow interpolations, proposed by Choi or Shen et al in [19, 21]. It is known that, the methods are weakly dependent of time step at convergence, but the magnitude of the dependence in real life applications is not known yet.

Nowadays, there exist several interpolation methods on collocated grids, resulting in solutions, independent of time step at convergence [20, 22–25]. Most of the methods are aimed for

applications in the SIMPLE-like algorithms. To enhance the convergence rate of the SIMPLE-like algorithms on collocated grids we consider the Modified Momentum Interpolation method (MMI), recently proposed in [35]. From one hand the MMI method results in the solution, which is independent of time step at convergence. From the other hand, as will be later seen in the paper, it may increase the convergence rate of the SIMPLEC algorithm up to 25% in comparison to other interpolation method existing in literature.

According to the MMI method the mass flux at some cell face k is defined as below:

$$f_k^{m+1} = \chi_k (\tilde{f}_k^n - \frac{\tau}{\rho} \vec{\nabla} p_k^{m+1} \cdot d\vec{S}_k + \left[\frac{\vec{h}_p^{m+1}}{AV} \right]_k \cdot d\vec{S}_k + (1/\alpha - 1 - \beta) \left[\frac{A}{AV} \right]_k f_k^m) \quad (4)$$

where $\left[\frac{A}{AV} \right]_k$ denotes linear interpolation from cell centers to cell face k , the term χ_k and the vector \vec{h}_p^{m+1} are defined respectively as:

$$\chi_k = 1 / \left(1.5 + (1/\alpha - \gamma) \left[\frac{A}{AV} \right]_k \right) \quad (5)$$

$$\vec{h}_p^{m+1} = - \sum_{E,W,N,S} A_{nb} \vec{v}_{nb}^{m+1} - \gamma A_p \vec{v}_p^{m+1} + \beta A_p \vec{v}_p^m + \vec{S}_p^m \quad (6)$$

where the new parameters γ and β are the constant parameters described below.

In Eq. (4) the term \tilde{f}_k^n indicates $2f_k^n - 0.5f_k^{n-1}$, where f_k^n , f_k^{n-1} , f_k^m are the fluxes available from the former time steps n , $n-1$ and subiteration m . To ensure the pressure-velocity coupling on collocated grids the pressure force term $\vec{\nabla} p_k^{m+1} \cdot d\vec{S}_k$ in Eq. (4) has to be calculated directly from the pressure at the cell centers as in the original Rhie-Chow method [1].

The MMI method above results in solution independent of time step and relaxation parameter, as will be seen in the result section. The method has two forms, namely the first form with $\gamma = 0$, $\beta = 0$ and the second form with $\gamma = 1$, $\beta > 0$. The two forms of the method possess different properties as described below:

1) The MMI method with $\gamma = 0$ and $\beta = 0$ is similar to an existing method of Pascau in [25], but contrary to Pascau's method, it is based on momentum equations, which are discretized using second order backward difference scheme in time. As shown in [35], the MMI method in this form is fully compatible with SIMPLE algorithm.

2) The MMI method with $\gamma = 1$ and $\beta > 0$ is another interpolation method, independent of time step and relaxation parameter at convergence. According to [35], for the robust performance of the method, the parameter β has to be equal 0.04.

Contrary to other time step independent methods, the MMI method in this form is fully compatible with SIMPLEC algorithm. As will be later seen in the result section, if the MMI method is applied with SIMPLEC algorithm, it becomes advantageous in the convergence speed over other time step independent interpolation method.

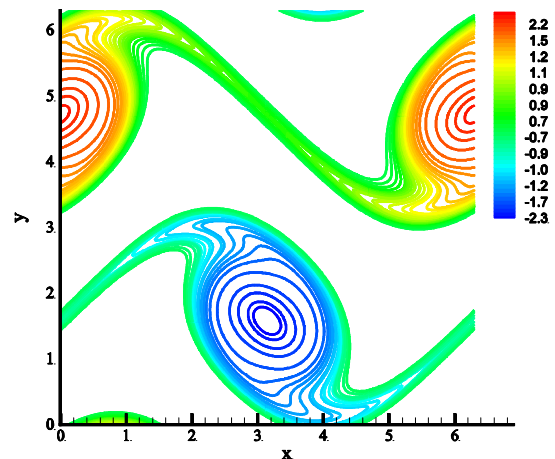


Figure 1. Vorticity field of a shear layer roll-up at $Re = 100$ and $t = 8$.

3. Results

Two test cases are computed, namely an idealized case of a shear layer [37], which appears in wind turbine wake, and the turbulent flow around a NACA 64618 airfoil.

The steady and unsteady state solutions are achieved when the residuals, computed in L_1 norm, are reduced by a factor of 10^8 and 10^6 , respectively. For the tests the velocity underrelaxation parameter $\alpha = 0.8$ is used.

3.1. Roll-up of a shear layer vortex

An idealized case of a shear layer, which appears in wind turbine wake, is computed at $Re = 100$ using SIMPLEC algorithm and the MMI interpolation method in Eq. (4). To achieve the high efficiency of the SIMPLEC algorithm, the MMI method is used with $\gamma = 1$ and $\beta = 0.04$ (see Section 2.2). To verify the efficiency of the MMI method, the method is compared with the interpolation method of Pascau [25].

The flow is initialized in the domain of a square of $[0, \pi]$ with periodic boundary conditions as following:

$$u = \begin{cases} \tanh(\frac{y-\pi/2}{\delta}), y \leq \pi & v = \epsilon \sin(x) \\ \tanh(\frac{3\pi/2-y}{\delta}), y > \pi & \delta = \pi/15, \epsilon = 0.05 \end{cases}$$

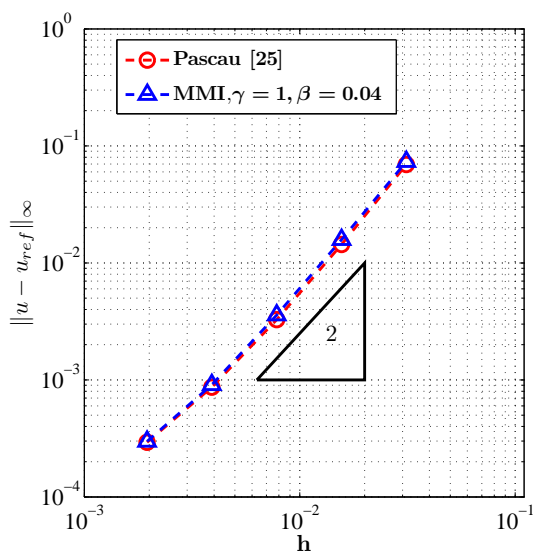


Figure 2. Spatial error convergence.

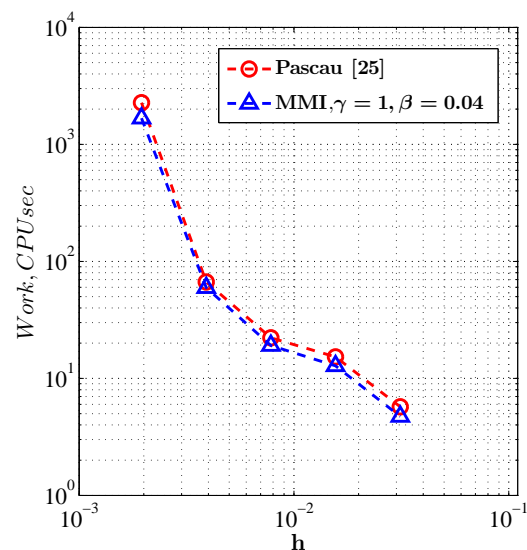


Figure 3. Work load dependence on grid resolution.

An example of the flow field at a dimensionless time $t = 8$ is shown in Fig. 1. The solutions of the two methods are compared at the dimensionless time $t = 4$ and the errors are measured in comparison to the reference solution of the MMI method obtained on the fine grid of 512^2 cells and with a small time step of 0.005. The computations are performed on a sequence of the successively coarsened grids. For each coarse grid, the error is computed by interpolating the reference solution on the coarse grid and subtracting the solution from that grid. As seen from Fig. 2, the second order of the spatial accuracy is obtained and the error tolerances of both methods are nearly identical.

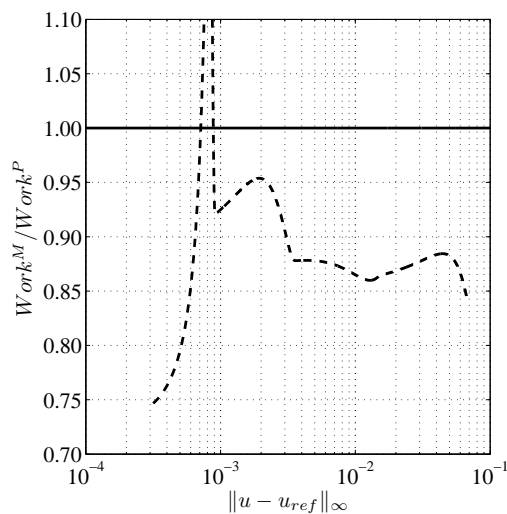


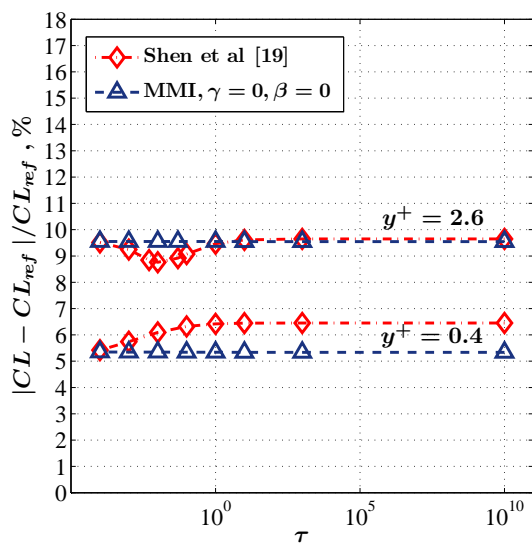
Figure 4. Dependence of work load on accuracy in a shear layer roll-up test case.

The corresponding work load of the methods, measured in CPU seconds, is plotted in Fig. 3. The efficiencies of the methods can be compared if the work loads are compared for same accuracy levels. Therefore, the work load for each of the method is found first as a function of tolerance error. Then the ratio of the work loads of the MMI method, $Work^M$, and Pascau's method, $Work^P$, is computed and plotted in Fig. 4.

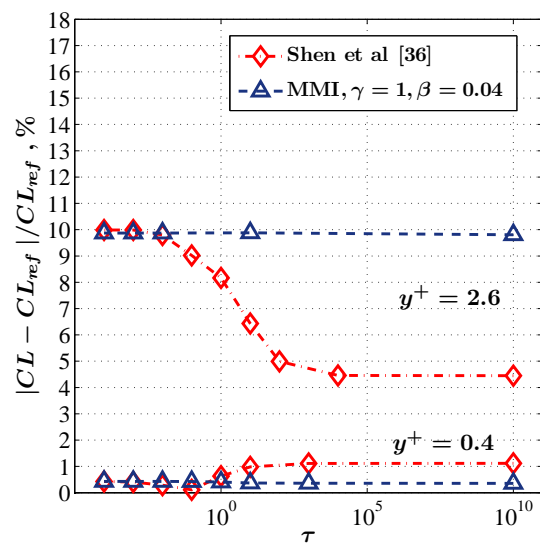
As seen in Fig. 4, the work load of the MMI method is lower than the work load of Pascau's method for nearly all accuracy levels. For the highest accuracy level, the efficiency of the MMI method is 25% higher in comparison to Pascau's method. As seen in Fig. 4 there is a narrow accuracy region where the MMI method is less efficient. This is explained by the fact that the optimal β parameter of the MMI method depends on grid resolution. In spite of the fact that the optimal β is not known in advance, the results of the test case and the deliberate analysis of the optimal β presented in [35] show that for general use $\beta = 0.04$ can be employed to ensure higher efficiency of the MMI method.

3.2. Turbulent flow around a NACA 64618 airfoil

The turbulent flow around NACA 64618 airfoil at zero angle of attack is computed at $Re = 1.6 \cdot 10^6$ using SIMPLEC algorithm. The grid of O-type is used with boundaries placed in a distance of 20 chords from the airfoil.



(a) Shen's method [19] and the MMI method.



(b) Shen's method [36] and the MMI method.

Figure 5. Dependence of relative errors of the methods of Shen et al [19, 36] in the turbulent flow computations around a NACA 64618 airfoil on grids with 64×32 and 128×64 cells.

Two standard interpolation methods of Shen et al in [19] and [36] are used as the representatives of interpolation methods, resulting to the solutions, dependent of time step at convergence. To compare with the standard methods, the MMI method in Eq. (4) is used.

The $k-\omega$ SST turbulence model is used on two relatively coarse grids with 64×32 and 128×64 cells. For the two grids the maximum y^+ at points one cell away from the airfoil equals to 2.6 and 0.4, respectively. To measure the time step dependence of the standard methods, the lift coefficient is compared against an experimental value in [38], which is equal to 0.44.

As seen from Figs. 5(a) and 5(b), contrary to the MMI method, the solutions of the standard methods of Shen, are dependent of time step at convergence. For the standard methods, the change of the error due to the time step dependence is about 1-2% on the grid with $y^+ = 0.4$, whereas for the grid with $y^+ = 2.6$ the error variations may achieve up to 5.5%, as seen from Fig. 5(b).

It should be noted, that a typical grid set-up for the turbulent flow computations is based on a grid with 128 cells in the normal direction. Results of computations on a grid with 256×128 cells using the standard and the MMI methods are shown in Fig. 6. It is seen from the figure that on such fine grid, the time step dependence of the error contributes about 0.5% into the total error.

The solution, obtained using the MMI method with $\gamma = 0$, $\beta = 0$ on the grid with 256×128 cells is seen to be about 4% different from the experimental value. The solution is less accurate than the solution obtained on the coarser grid using the MMI method with $\gamma = 1$, $\beta = 0.04$. This is explained by the fact that for the steady state problems, the MMI method with $\gamma = 1$ and $\beta = 0.04$ may result in superconvergence, as was reported in [35].

4. Conclusions

It is concluded, that for a typical turbulent flow computation the standard momentum interpolation methods still can be used. For the methods the solution dependence of the time step is negligible. However, in optimization tasks, where coarse grids are used, the standard methods may result in non consistent solution behavior. The inconsistency may become crucial in 3D computations, where employing the fine grids becomes computationally demanding. For the turbulent flow computations, the MMI method of Kolmogorov et al [35] was proved to result in solutions, independent of time step at convergence. The results of the unsteady flow computations have also shown, that when SIMPLEC algorithm is used, the MMI method, results in up to 25% higher convergence rate, than an existing method. In general, the MMI method can be used on both coarse and fine grids to ensure both high convergence rate and high accuracy of the SIMPLE-like algorithms.

Acknowledgments

The present work was funded by the Danish Council for Strategic Research (DSF), under contract 2104-09-0026, Center for Computational Wind Turbine Aerodynamics and Atmospheric Turbulence.

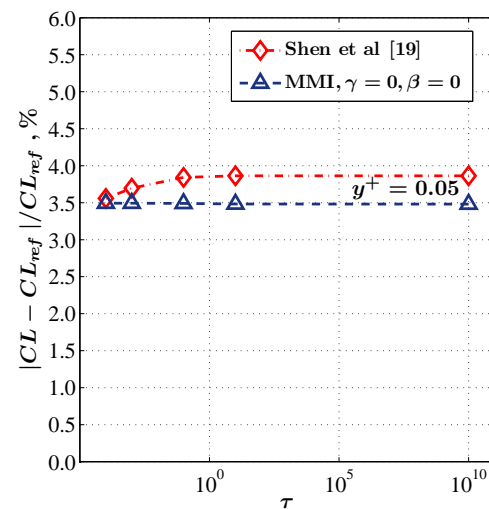


Figure 6. Dependence of relative error in the flow computations around NACA 64618 airfoil on a grid with 256×128 cells.

References

- [1] Rhie C and Chow W 1983 *AIAA journal* **21** 1525–1532
- [2] Duque E P, Van Dam C and Hughes S C 1999 *EWEC-CONFERENCE* pp 79–84
- [3] Sørensen N N, Michelsen J and Schreck S 2002 *Wind Energy* **5** 151–169
- [4] Johansen J, Sørensen N, Michelsen J and Schreck S 2002 *ASME 2002 Wind Energy Symposium* (American Society of Mechanical Engineers) pp 106–114
- [5] Duque E P, Burklund M D and Johnson W 2003 *Journal of Solar Energy Engineering* **125** 457–467
- [6] Xu G and Sankar L N 2002 *Wind Energy* **5** 171–183
- [7] Pape A L and Lecanu J 2004 *Wind Energy* **7** 309–324
- [8] Zahle F, Sørensen N N and Johansen J 2009 *Wind Energy* **12** 594–619
- [9] Gomez-Iradi S, Steijl R and Barakos G 2009 *Journal of Solar Energy Engineering* **131** 031009
- [10] Sørensen N N 2009 *Wind Energy* **12** 715–733
- [11] Sezer-Uzol N and Long L N 2006 *AIAA paper* **394** 2006
- [12] Potsdam M A and Mavriplis D J 2009 *AIAA paper* **1221** 2009
- [13] Bechmann A, Sørensen N N and Zahle F 2011 *Wind Energy* **14** 677–689
- [14] Sørensen J, Shen W and Munduate X 1998 *Wind Energy* **1** 73–88
- [15] Troldborg N, Sørensen J N and Mikkelsen R 2010 *Wind Energy* **13** 86–99
- [16] Mikkelsen R 2003 *Actuator disc methods applied to wind turbines* Ph.D. thesis Technical University of Denmark
- [17] Majumdar S 1988 *Numerical Heat Transfer* **13** 125–132
- [18] Miller T and Schmidt F 1988 *Numerical Heat Transfer, Part A: Applications* **14** 213–233
- [19] Shen W Z, Michelsen J A and Nørkær Sørensen J 2001 *AIAA journal* **39** 2406–2409
- [20] Yu B, Tao W Q, Wei J J, Kawaguchi Y, Tagawa T and Ozoe H 2002 *Numerical Heat Transfer: Part B: Fundamentals* **42** 141–166
- [21] Choi S K 1999 *Numerical Heat Transfer: Part A: Applications* **36** 545–550
- [22] Kawaguchi Y, Tao W Q and Ozoe H 2002 *Numerical Heat Transfer: Part B: Fundamentals* **41** 85–94
- [23] Lien F and Leschziner M 1994 *Computer methods in applied mechanics and engineering* **114** 123–148
- [24] Cubero A and Fuego N 2007 *Numerical Heat Transfer, Part B: Fundamentals* **52** 507–529
- [25] Pascau A 2011 *International Journal for Numerical Methods in Fluids* **65** 812–833
- [26] Kamakoti R, Shyy W, Thakur S and Sankar B 2004 *AIAA* **886** 2004
- [27] Mencinger J and Žun I 2007 *Journal of Computational Physics* **221** 524–538
- [28] Kamakoti R and Shyy W 2004 *Progress in Aerospace Sciences* **40** 535–558
- [29] Abbasi R, Ashrafzadeh A and Shadaram A 2013 *Computers & Fluids*
- [30] Patankar S V 1980 *Numerical heat transfer and fluid flow* (Taylor & Francis)
- [31] Van Doormaal J and Raithby G 1984 *Numerical heat transfer* **7** 147–163
- [32] Issa R I 1986 *Journal of Computational physics* **62** 40–65
- [33] Ferziger J H and Perić M 2002 *Computational methods for fluid dynamics* vol 3 (Springer Berlin)
- [34] Wesseling P 2009 *Principles of computational fluid dynamics* vol 29 (Springer)
- [35] Kolmogorov D K, Shen W Z, Sørensen N N and Nørkær Sørensen J 2014 *Numerical Heat Transfer, Part B: Fundamentals* (submitted)
- [36] Shen W Z, Michelsen J A, Sørensen N N and Nørkær Sørensen J 2003 *Numerical Heat Transfer: Part B: Fundamentals* **43** 221–239
- [37] Minion M L and Brown D L 1997 *Journal of Computational Physics* **138** 734–765
- [38] Cheng J, Zhu W, Fischer A, García N R, Madsen J, Chen J and Shen W Z 2013 *Wind Energy* **16**