Damping the fluctuating behaviour and improving the convergence rate of the axial induction factor in the BEMT based rotor aerodynamic codes

Alireza Maheri
University of Aberdeen

Siamak Noroozi
Bournemouth University

Chris Toomer
University of the West of England, Bristol

Damping the fluctuating behaviour and improving the convergence rate of the axial induction factor in the BEMT-based rotor aerodynamic codes

Alireza Maheri*, Siamak Noroozi, Chris Toomer, John Vinney
University of the West of England, Bristol, BS16 1QY, UK
*Ali2.Maheri@uwe.ac.uk

Abstract:

A common problem in the rotor aerodynamic codes which are based on the Blade Element Momentum Theory is the fluctuating behaviour of the axial induction factor in its iteration loop. These fluctuations are due to periodically switching of the loading state between light and heavy. Normally in such situations, after a pre-defined maximum number of iterations, the iteration process stops and the program skips that blade segment. This impacts both the accuracy of the predicted results and the code performance. This paper presents a method to avoid this behaviour by using a relaxation factor applicable in the iteration process. By this, no matter how high the amplitudes of the fluctuations, using a proper relaxation factor leads to a converged solution. This method also uses the first few iterated values to predict the neighbourhood of the final converged axial induction factor. This can be used to accelerate the convergence process. It has been shown that selection of a proper relaxation factor together with a simple modification to the predicted axial induction factor after a few iterations highly improves the convergence rate.

Keywords: BEMT, Relaxation factor, Wind turbine simulation, Rotor aerodynamic

1 Introduction

Blade Element Momentum Theory, BEMT is an analysis method applicable to the rotors. It is a combination of Blade Element or Strip Theory and Momentum or Actuator Disk Theory and gives basic insights into the rotor loading and performance. This theory is an intermediate theory between simple actuator disk theory and rigorous CFD based vortex theory. High computational costs and memory requirement for turbine wake calculation and also lack of a reliable turbulence model for predicting separated flow characteristics on the blades make CFD based models less practical in engineering use. BEMT-based codes are somewhat of an industry standard and almost all commercial codes dealing with the aerodynamics of the horizontal axis wind turbines are based on this theory.

BEMT is based on some assumptions which make it bounded within many limitations. For example the ground shear and yaw error contradict the basic assumption of axisymmetric flow. However in practice some of the limitations can be removed by applying some corrections to the original concepts. Reference [1] gives a comprehensive review of the BEMT and the corrections applicable to remove some of its limitations. Models based on BEMT are sensitive to the aerodynamic characteristics of the aerofoils used in the blade [2-3]. Therefore, having a reliable model for predicting the post stall aerodynamic coefficients and considering the effect of stall delay due to blade rotation [4] are also necessary for acceptable results.

BEMT postulates the effects of the presence and rotation of the rotor on the flow field around the rotor by introducing and calculating the field of the induced velocities. This evaluation is based on an iterative algorithm in which the induced velocities are initially assumed and then will be re-calculated by iteration.

2 Damping the fluctuations

When analysing each segment of the blade, BEMT deals with 8 unknown parameters. These unknowns are axial and rotational induction factors a and a', tip (and hub) loss correction factor F, inflow and attack angles φ and α, and lift, drag and thrust coefficients C_L, C_D and C_T. Six algebraic equations in accompany with a set of two discrete data equations (tables of $\alpha - C_L$ and $\alpha - C_D$) relate the 8 unknown to each other and form a pseudo system of equations. Having two tables instead of two ordinary algebraic equations make the nature of the BEMT analysis an iterative one. Forms of the six equations and roughly knowing the neighbourhoods of the induction factors a and a' makes them the best choices as iterative parameters.

Figures (1) and (2) show two different types of behaviour of the axial induction factor, a , in its iteration loop. The reason of fluctuating behaviour of the axial induction factor, as shown in Figure (2), can be explained as follows. Momentum theory predicts a parabolic variation for thrust coefficient with a maximum value of 1 at $a = 0.5$, while the experimental data show that C_T keeps increasing for $a > 0.5$. For small axial induction factors, $0 < a < a_c \cong 0.4$, known as light loading state, predicted thrust coefficient by the momentum theory is in a good agreement with the experimental data while in the heavy loading state, where $a > a_c$, predicted C_T departs dramatically from its actual value. For the heavy loading state the momentum-based equation is replaced by the Glauert's empirical formula. Separating light and heavy loading states, and as a result of that relating C_T and a through two different equations, imposes a singular point of a_c in the domain and therefore when two successive predicted axial induction factors lie in different sides of a_c a fluctuating behaviour, as shown in Figure (2), is expected.

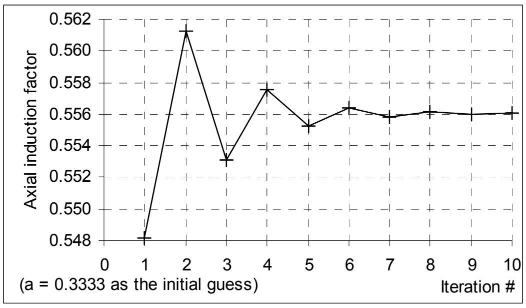

Figure 1: Converging oscillatory behaviour of the axial induction factor

The fact of having a pseudo system of 8 unknowns and 8 equations and an iterative algorithm for solving that, leads one to the idea of using the concepts applicable to a real system of equation, i.e. relaxation factor, to accelerate the convergence or avoid the divergence of the solution.

By using a relaxation factor, as represented in Equation (1), the new results are partly due to the effect of the history of the iteration. This dependence to the old values reduces the oscillation amplitudes or over-damps the oscillatory behaviour.

$$a_{k+1} := \omega a_{k+1} + (1-\omega) a_k ; \quad 0 < \omega \le 1 \qquad (1)$$

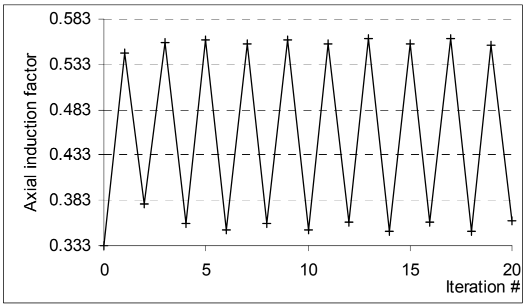

Figure 2: Fluctuating behaviour of the axial induction factor due to change of loading state

Relaxation factors greater than 1 amplify the amplitude of the oscillations and will impact the convergence rate or may cause divergence. Very small values of relaxation factors also reduce the rate of convergence. The optimal relaxation factor is a problem dependent variable. In our case, since the calculations in each iteration are partly through the tabulated data, there is no analytical way to find the optimum relaxation factor.

Figures (3) and (4) show the results of iterations when using different relaxation factors. Experiments show that the minimum relaxation factor that permits the oscillatory behaviour is about 0.56. Therefore selecting a relaxation factor of $\omega \cong 0.5$ can be a safe selection that substitutes an oscillatory behaviour with a smoothly increasing or decreasing behaviour.

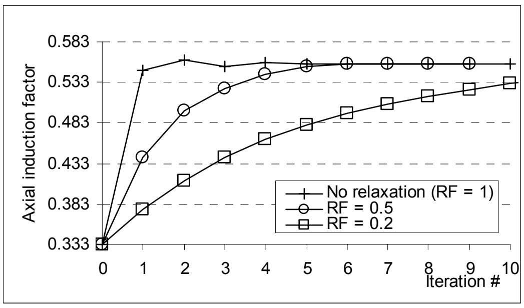

Figure 3: Effect of different values of relaxation factor (RF) on the oscillatory behaviour of the axial induction factor

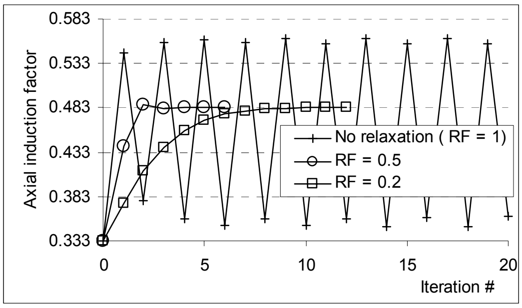

Figure 4: Effect of different values of relaxation factor (RF) on the fluctuating behaviour of the axial induction factor

Figure (3) shows that using a relaxation factor of 0.5 not only damps the oscillatory behaviours but also slightly improves the convergence rate. Figure (4) shows that using a relaxation factor of about 0.5 or less replaces a fluctuating divergent solution by a converged one.

3 Accelerating the convergence

Convergence rate strongly depends on the value of the initial guess of the axial induction factor. When starting the iteration there is no knowledge about the neighbourhood of the converged solution. After a few iterations there is enough knowledge to find a reasonable narrow neighbourhood for the final result. The dashed lines in Figure (5) show the boundaries of the oscillation domain if no relaxation factor is used. Since the final result lies between these lines, they can be used as a measure of the neighbourhood of the converged solution. Sooner entering to this neighbourhood yields a faster convergence. It is popular to use $a_0 = 1/3$ as initial guess. But using $a_0 = 1/3$ as initial guess and applying Equation (1) from the first iteration prevents the calculated a_1 from entering into the neighbourhood of the converged solution unless $a_0 = 1/3$ has been already in the neighbourhood. This delay in entrance lowers the convergence rate. See the curve with the circular signs in Figure (5).

An option to avoid this situation is applying a relaxation factor of $\omega = 1$ for the first few iterations to let the first few oscillations happen in order to make sure that the axial induction factor has been lied on the boundary of the neighbourhood. The curve with the triangular signs in Figure (5), shows the results when the relaxation factor of 0.5 has been applied after the 2nd iteration. A good improvement in the convergence can be seen due to this modification.

A further step for improvement of the convergence rate can be still carried out by entering the axial induction factor into the neighbourhood of convergence after few iterations rather than putting that on the boundary of the neighbourhood. To do this a three-point modification is necessary while the middle point must be a peak point. Again, similar to the first modification, a relaxation factor of $\omega = 1$ applies to the first few iterations to make sure of having a peak point. Then the following three-point equation can be used to modify the calculated new axial induction factor. And finally equation (1) with a relaxation factor of $\omega \cong 0.5$ applies to calculate the new axial induction factors until the final convergence.

$$a_{k+1} := \frac{1}{4} a_{k+1} + \frac{1}{2} a_k + \frac{1}{4} a_{k-1} \qquad (2)$$

Equation (2) locates the centre of a triangle constructed on three successive points of iterations. The curve with the square signs in Figure (5), shows the results when the modification of Equation (2) has been applied for the 3rd iteration. This modification gives a faster convergence than the previous one.

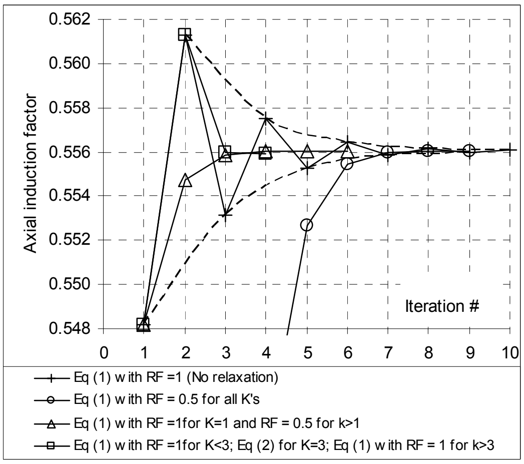

Figure 5: Accelerating the convergence

A combination of the results of the accelerating the convergence and damping fluctuations can be summarised in the following algorithm.

- Use Equation (1) with $\omega = 1$ until passing the first peak point. In the case of detecting no peak point, there is no fluctuation and therefore $\omega = 1$ gives the highest convergence rate.

- Apply Equation (2) for the first next iteration to shift the iterated axial induction factor into the convergence neighbourhood.
- Use Equation (1) with $\omega = 0.5$ for next iterations to prevent fluctuation until the final convergence.

Figure (6) shows the average number of iterations per segment versus the wind velocity for a typical AWT 27 wind turbine. The blade is divided into 40 segments and the convergence tolerance has taken equal to 0.0001. The results have been obtained by using WTAero, a BEMT-based code developed in the University of the West of England. According to this figure the average number of iterations per segment varies with the wind velocity. The mean values of these averages over a 15 m/s range of wind velocities are also calculated and shown on Figure (6).

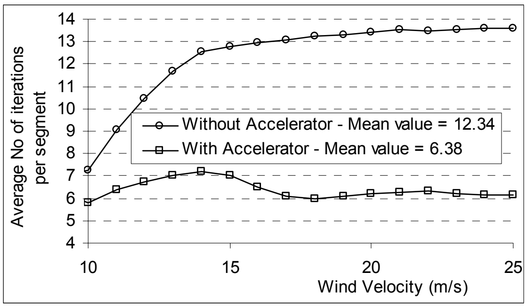

Figure 6: Improvement in a BEMT-based code performance

Comparing these mean values show a reduction of about 48.3% in the number of iterations if the above accelerator algorithm is used. Since in a BEMT code most of the calculations happen inside the iteration loop, by a 48.3% reduction of the number of iterations, the overall performance of the code improves almost by the same factor. This amount of improvements in the performance of the code in a single run, which only takes a fraction of second, is not noticeable. But it becomes important when the code operates as the evaluator of an optimisation or design procedure. To gain a better understanding, assume a genetic algorithm optimisation procedure with a population of size of 10 and total number of generations of 1000. In this problem the code must run 10,000 times for every single wind velocity over a range of cut-in to cut-off wind velocities. In cases like this, such an improvement in the convergence rate is noticeable.

4 Conclusion

Fluctuating behaviour of the axial induction factor reduces the accuracy of the BEMT-based codes. A simple way to avoid this behaviour is introducing a relaxation factor as a damper. Experiments show that the relaxation factors greater than of about 0.56 can not damp all the possible fluctuations. Lowering the relaxation factor to make sure of not having fluctuation, impacts the convergence rate. To improve the convergence rate an algorithm has been presented to use the knowledge of the first few iterations to accelerate the convergence. It is shown that using this simple algorithm prevents the fluctuations and therefore improves the accuracy of the BEMT codes and at the same time can improve the convergence rate up to about 50%.

References

[1] Spera D A. Wind Turbine Technology, Fundamental Concepts of Wind Turbine Engineering. American Society of Mechanical Engineers,1994.

[2] Tangler J L. The Nebulous Art of Using Wind-Tunnel Airfoil Data for Predicting Rotor Performance. National Renewable Energy Laboratory, Report NREL/CP-500-31243, January 2002.

[3] Tangler J L, Kocurek J D. Wind Turbine Post-Stall Airfoil Performance Characteristics Guidelines for Blade-Element Momentum Methods. National Renewable Energy Laboratory, Report NREL/CP-500-36900, October 2004.

[4] Du Z, Selig M S. A 3-D Stall-Delay Model for Horizontal Axis Wind Turbine Performance Prediction. Proceedings of the 1998 ASME Wind Energy Symposium, Reno, NV 1998 p. 9–19.