



# A Vortex Step Method for Nonlinear Airfoil Polar Data as Implemented in KiteAeroDyn

## Preprint

Rick Damiani,<sup>1</sup> Fabian Wendt,<sup>1</sup> Jason Jonkman,<sup>1</sup>  
and Jerome Sicard<sup>2</sup>

*1 National Renewable Energy Laboratory  
2 Makani*

*Presented at the American Institute for Aeronautics and Astronautics  
SciTech Forum 2019  
San Diego, California  
January 7-11, 2019*

*Produced under direction of Makani by the National Renewable Energy  
Laboratory (NREL) under Work for Others Agreement CRD-17-00659.*

**NREL is a national laboratory of the U.S. Department of Energy  
Office of Energy Efficiency & Renewable Energy  
Operated by the Alliance for Sustainable Energy, LLC**

This report is available at no cost from the National Renewable Energy  
Laboratory (NREL) at [www.nrel.gov/publications](http://www.nrel.gov/publications).

Contract No. DE-AC36-08GO28308

**Conference Paper**  
NREL/CP-5000-72777  
January 2019



# A Vortex Step Method for Nonlinear Airfoil Polar Data as Implemented in KiteAeroDyn

## Preprint

Rick Damiani,<sup>1</sup> Fabian Wendt,<sup>1</sup> Jason Jonkman,<sup>1</sup>  
and Jerome Sicard<sup>2</sup>

*1 National Renewable Energy Laboratory*

*2 Makani*

### Suggested Citation

Damiani, Rick, Fabian Wendt, Jason Jonkman, and Jerome Sicard. 2019. *A Vortex Step Method for Nonlinear Airfoil Polar Data as Implemented in KiteAeroDyn: Preprint*. Golden, CO: National Renewable Energy Laboratory. NREL/CP-5000-72777.

<https://www.nrel.gov/docs/fy19osti/72777.pdf>.

**NREL is a national laboratory of the U.S. Department of Energy  
Office of Energy Efficiency & Renewable Energy  
Operated by the Alliance for Sustainable Energy, LLC**

This report is available at no cost from the National Renewable Energy Laboratory (NREL) at [www.nrel.gov/publications](http://www.nrel.gov/publications).

Contract No. DE-AC36-08GO28308

**Conference Paper**  
NREL/CP-5000-72777  
January 2019

National Renewable Energy Laboratory  
15013 Denver West Parkway  
Golden, CO 80401  
303-275-3000 • [www.nrel.gov](http://www.nrel.gov)

## NOTICE

This work was authored by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding provided by Makani. The views expressed herein do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

This report is available at no cost from the National Renewable Energy Laboratory (NREL) at [www.nrel.gov/publications](http://www.nrel.gov/publications).

U.S. Department of Energy (DOE) reports produced after 1991 and a growing number of pre-1991 documents are available free via [www.OSTI.gov](http://www.OSTI.gov).

*Cover Photos by Dennis Schroeder: (clockwise, left to right) NREL 51934, NREL 45897, NREL 42160, NREL 45891, NREL 48097, NREL 46526.*

NREL prints on paper that contains recycled content.

# A Vortex Step Method for Nonlinear Airfoil Polar Data as Implemented in KiteAeroDyn

Rick Damiani\* Fabian Wendt† Jason Jonkman ‡  
National Renewable Energy Laboratory, Golden, CO 80401 - USA

Jerome Sicard§  
Makani, Alameda, CA 94501 - USA

**This paper summarizes the underlying theory in the recently developed KiteAeroDyn (KiteAD) module, which is part of KiteFAST (KiteFAST), a numerical simulation tool that is currently being developed at the NREL. KiteFAST is intended for the simulation of airborne wind energy systems (e.g., energy kites). KiteAD implements an innovative vortex step method for the calculation of aerodynamic loads on all of the lifting surfaces of the kite, which is capable of accounting for two-dimensional viscous effects through the use of nonlinear airfoil polars. Considering nonlinear airfoil polar data allows for the modeling of flap or other moving-surface deflections, as well as for the modelling of inflow conditions with larger angles of attack. From preliminary verification against other existing codes that can solve for lift distribution on lifting surfaces, the results of the newly implemented method proved to be numerically robust and computationally inexpensive, thus ideal for aeroelastic design and analysis applications.**

## I. Introduction

THE dynamics of semirigid kites can be simulated by combining models for aerodynamics and structures. This paper describes the theory developed and implemented in KiteAD, an aerodynamics module that is coupled to MBDyn [1] through the FAST framework [2] to simulate airborne wind energy kites.

The main aerodynamic problem for aircraft-like structures comes down to solving for the induction associated with the vorticity field generated by the presence of lifting surfaces. In this document, we denote any lifting surfaces as ‘wings’ regardless of the actual component function of the surfaces. Various methods exist to accomplish this task and with various levels of fidelity and computational demand. KiteVSM is based on Weissinger’s method [3], also known as Vortex Step Method (VSM), but with some modifications and extensions to improve the accuracy, computational efficiency, minimization of instabilities, and to consider nonlinear airfoil polar data (e.g., airfoil cross sections with flap deflections) in combination with multiple lifting surfaces.

In Section II, a brief review is offered on the existing induction models for lifting surfaces. Section III describes the theoretical development of the new VSM. The code organization and implementation within a prescribed modularization framework are summarized in Section IV.A. Sections V and VI present the results of preliminary verification conducted on the newly implemented method. The results demonstrate that the code is computationally efficient and appropriate for supporting the aeroelastic design and analysis of semirigid kites.

## II. Review of Existing Lift Models

The VSM belongs to the vortex-lattice class and it approximates Prandtl’s lifting-line theory [4], with contributions from the work of Munk [5], Pistolesi [6, 3/4-chord theorem], Wieghardt [7], and Mutterperl [8]. The lifting-line method can be applied to straight, planar wings of high aspect ratio ( $AR > 4$ ). In its original formulation, this method reaches a closed form solution for the induced angle of attack along the wingspan by assuming a simple flat-plate behavior where the lift coefficient is simply regarded as  $C_l = 2\pi\alpha$ .

The VSM, on the other hand, promises accurate solutions for low and high  $AR$  wings of different shapes, including swept and dihedral wings. Furthermore, the VSM can be modified to account for the nonlinear polar curve of real airfoils.

---

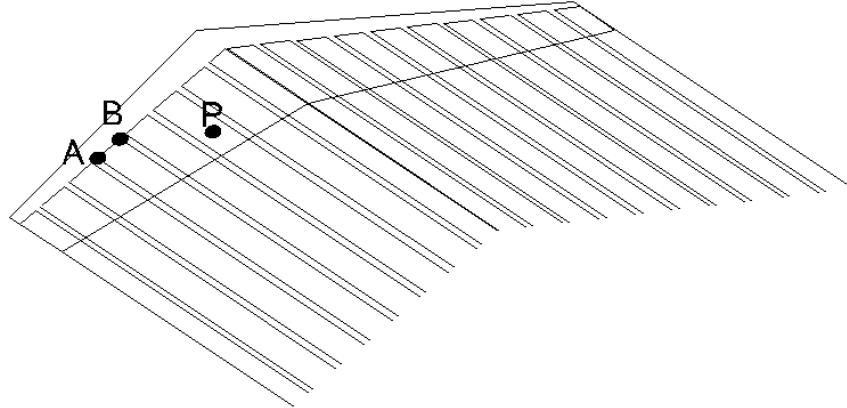
\*Sr. Engineer, National Renewable Energy Laboratory, AIAA Professional Member. rick.damiani@nrel.gov

†Engineer, National Renewable Energy Laboratory.

‡Sr. Engineer, National Renewable Energy Laboratory, AIAA Professional Member.

§Sr. Engineer, Makani, a project of X

Whereas Prandtl's lifting line sheds a continuous trailing vorticity from the wing's  $1/4$ -chord ( $1/4$ -c) line the VSM approximates the resulting vortex sheet with a finite number of horseshoe vortices. The bound portion of the horseshoe vortex lies along the  $1/4$ -c line with the trailing arms aligned with the freestream direction. One other difference is in the control-point locations. The lifting-line method calculates induction along control points at the  $1/4$ -c whereas the VSM places control points at the  $3/4$ -c but in the direction of the freestream.

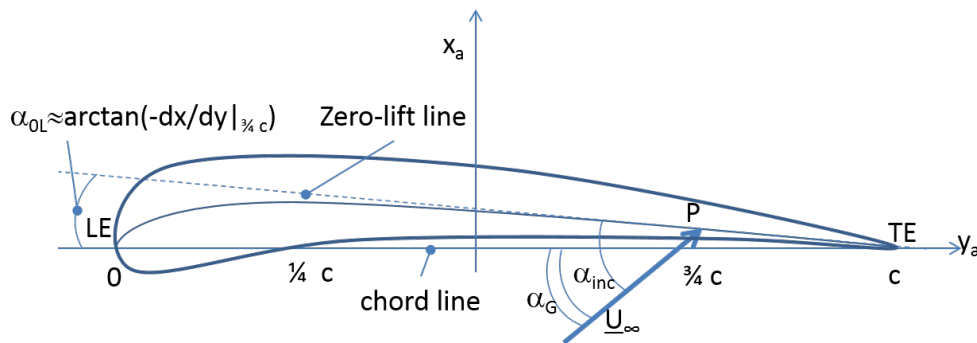


**Fig. 1 Horseshoe vortices distributed following the adopted convention for the VSM.  $A_i$ ,  $B_i$  are the generic start and end points of the  $1/4$ -c-bound segment of the generic horseshoe vortex.  $P$  is a generic control point located at the  $3/4$ -c.**

### III. KiteVSM: A New Vortex Step Method

In this section, the fundamental theory of KiteVSM is discussed.

Normally, techniques based on Weissinger's method assume either a simple array of unknown values or some Fourier-modal representation with unknown coefficients for the circulation ( $\Gamma(y)=\Gamma(\theta)$ , where  $y$  is the spanwise coordinate and  $\theta$  is its transformed value from  $y = -\frac{span}{2} \cos \theta$ ). To attain the unknowns, those methods impose a slip, nonpenetrating wall condition at the  $3/4$ -c control points that translates into solving the resulting linear system of equations. The choice of the  $3/4$ -c condition derives from Pistolesi's theorem, which states that the zero-lift angle of attack is approximated by the tangent to the camber line at the  $3/4$ -c point is equal to  $\alpha_{inc} = \alpha_G - \arctan\left(\left.\frac{dx}{dy}\right|_{3/4-c}\right)$  (assuming an airfoil local-reference frame, where  $\alpha_G$  is the angle of attack of the generic wing section).



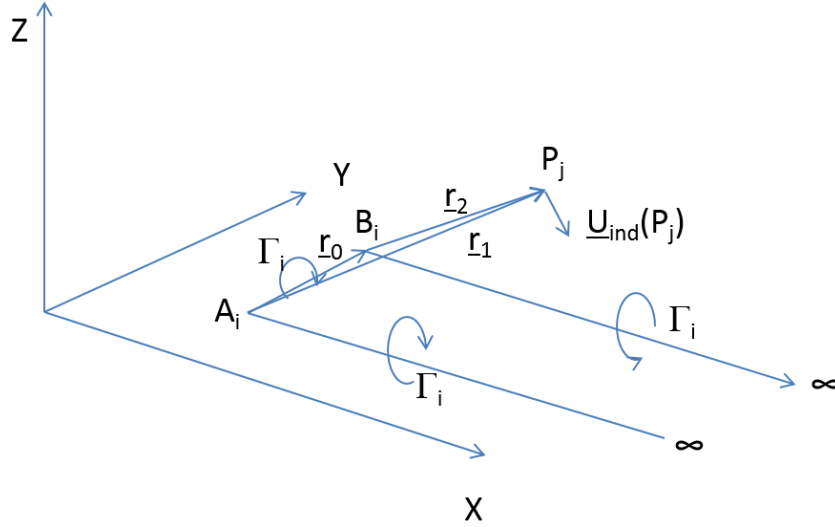
**Fig. 2 Representation of Pistolesi's theorem at the generic spanwise wing section. Note the local airfoil coordinate system ( $\hat{x}_{airf}$ ,  $\hat{y}_{airf}$ ) is displayed.**

The induced velocity at the control point location ( $P_j$ ) can be obtained by summing the contributions from all

horseshoe vortices with associated circulation  $\Gamma_i$ . The velocity induced by the generic  $i$ -th horseshoe vortex ( $A_i B_i$ ,  $A_\infty$ ,  $B_\infty$ ) of circulation  $\Gamma_i$  on the generic  $P_j$  can be written, with reference to Figs. 1 and 3, as in Eq. (1):

$$\underline{U}_{ind,i}(P_j) = \underline{U}_{A_i B_i}(P_j) + \underline{U}_{A_i \infty}(P_j) + \underline{U}_{B_i \infty}(P_j) \quad (1)$$

where  $\underline{U}_{A_i B_i}(P_j)$  is the contribution to the induced velocity by the  $i$ -th bound vorticity segment  $A_i B_i$ ;  $\underline{U}_{A_i \infty}(P_j)$  is the contribution to the induced velocity by the first trailing-vorticity, semi-infinite line of the generic horseshoe vortex  $\Gamma_i$ ;  $\underline{U}_{B_i \infty}(P_j)$  is the contribution to the induced velocity by the second trailing-vorticity, semi-infinite line of the generic horseshoe vortex  $\Gamma_i$ .



**Fig. 3** Contribution of the generic  $i$ -th horseshoe vortex to the induced velocity at the generic  $j$ -th control point,  $P_j$ .

#### A. Computation of Induced Velocities and Vortex Core Correction

The various terms in Eq. (1) can be found by using the Biot-Savart law and can be expressed as in Eq. (2) [ref. 9]:

$$\underline{U}_{A_i B_i}(P_j) = \begin{cases} \frac{\Gamma_i}{4\pi} \frac{r_1 \times r_2}{|r_1 \times r_2|^2} \left[ r_0 \cdot \left( \frac{r_1}{r_1} - \frac{r_2}{r_2} \right) \right] & \text{if } \frac{|r_1 \times r_0|}{r_0} > \epsilon_2 \\ \frac{|r_1 \times r_0|}{r_0 \epsilon_2} \underline{U}_{A_i B_i}(P'_j) & \text{otherwise} \end{cases}$$

$$\underline{U}_{A_i \infty}(P_j) = \begin{cases} \frac{\Gamma_i}{4\pi} \frac{1 + \frac{r_1 \cdot \underline{\xi}}{r_1}}{|r_1 \times \underline{\xi}|^2} r_1 \times \underline{\xi} & \text{if } |r_1 \times \underline{\xi}| > \epsilon_1 \\ \frac{|r_1 \times \underline{\xi}|}{\epsilon_1} \underline{U}_{A_i \infty}(P'_j) & \text{otherwise} \end{cases}$$

$$\underline{U}_{B_i \infty}(P_j) = \begin{cases} -\frac{\Gamma_i}{4\pi} \frac{1 + \frac{r_2 \cdot \underline{\xi}}{r_2}}{|r_2 \times \underline{\xi}|^2} r_2 \times \underline{\xi} & \text{if } |r_2 \times \underline{\xi}| > \epsilon_1 \\ \frac{|r_2 \times \underline{\xi}|}{\epsilon_1} \underline{U}_{B_i \infty}(P'_j) & \text{otherwise} \end{cases} \quad (2)$$

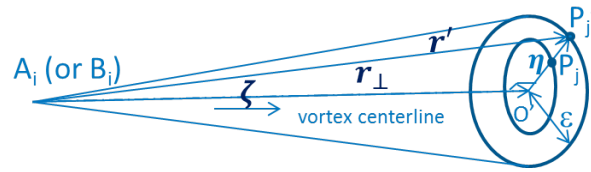
where  $\underline{r}_0$  is the position vector from  $A_i$  to  $B_i$ ;  $\underline{r}_1$  is the position vector from  $A_i$  to  $P_j$ ;  $\underline{r}_2$  is the position vector from  $B_i$  to  $P_j$ ; and  $\underline{\xi}$  is the unit vector along either the relative freestream velocity or the local chord direction (as selected by the user in the KiteAD input file); the vortex core radius  $\epsilon_1$  for the trailing vorticity is given by Eq. (3) [10] (see also Fig. 4):

$$\epsilon_1 = \sqrt{4\alpha_o \nu \frac{|\underline{r}_\perp|}{U_\infty}} \quad (3)$$

where  $\nu$  is the air kinematic viscosity;  $\alpha_o$  is the Oseen parameter equal to 1.25643; and  $\underline{r}_\perp$  is the position vector from either  $A_i$ ,  $B_i$ , or  $C_i$  to the projection of  $P_j'$  onto the respective vortex centerline as shown in Eq. (4):

$$\underline{r}_\perp = (\underline{r} \cdot \underline{\xi}) \underline{\xi} \quad (4)$$

where  $\underline{r}$  is the generic position vector from  $A_i$ ,  $B_i$ , or  $C_i$  to  $P_j'$ .



**Fig. 4 Diagram showing the relative location of  $A_i$ ,  $B_i$ ,  $P_j$ ,  $P_j'$  for the trailing-vortex core correction. Other symbols defined in the text.**

$\underline{U}_{A_i\infty}(P_j')$  is the  $\underline{U}_{A_i\infty}(P_j)$  calculated at the radial projection of  $P_j$  on the vortex core edge while  $\underline{U}_{B_i\infty}(P_j')$  is the  $\underline{U}_{B_i\infty}(P_j)$  calculated at the radial projection of  $P_j$  on the vortex core edge, respectively, which can be calculated as shown in Eq. (5):

$$\begin{aligned} \underline{U}_{A_i\infty}(P_j') &= \frac{\Gamma_i}{4\pi} \frac{1 + \frac{\underline{r}'_1 \cdot \underline{\xi}}{r'_1}}{\left| \underline{r}'_1 \times \underline{\xi} \right|^2} \underline{r}'_1 \times \underline{\xi} \\ \underline{U}_{B_i\infty}(P_j') &= -\frac{\Gamma_i}{4\pi} \frac{1 + \frac{\underline{r}'_2 \cdot \underline{\xi}}{r'_2}}{\left| \underline{r}'_2 \times \underline{\xi} \right|^2} \underline{r}'_2 \times \underline{\xi} \end{aligned} \quad (5)$$

$$\text{with} \quad \underline{r}' = \underline{r}_\perp + \epsilon_1 \underline{\eta}$$

where the  $\underline{r}'$  is the position vector from either  $A_i$  or  $B_i$  to  $P_j'$  (i.e., either  $\underline{r}'_1$  or  $\underline{r}'_2$ ) and the radial unit vector  $\underline{\eta}$  can be calculated as in Eq. (6):

$$\underline{\eta} = \frac{\frac{\underline{r}}{|\underline{r}|} - \underline{\xi}}{\left| \frac{\underline{r}}{|\underline{r}|} - \underline{\xi} \right|} \quad (6)$$

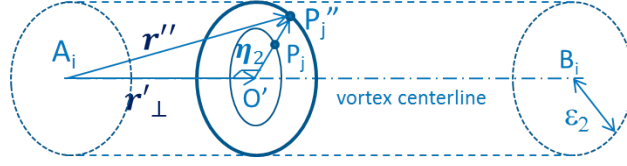
For the bound vorticity, the core radius is fixed as shown in Eq. (7) [11] and Fig. 5:

$$\epsilon_2 = 0.05 * r_0 \quad (7)$$

$\underline{U}_{A_i B_i}(P_j')$  is the  $\underline{U}_{A_i B_i}(P_j)$  calculated at the radial projection of  $P_j$  on the vortex core edge, calculated according to Eq. (8):

$$\underline{U}_{A_i B_i}(P_j') = \frac{\Gamma_i}{4\pi} \frac{\underline{r}'_1 \times \underline{r}'_2}{\left| \underline{r}'_1 \times \underline{r}'_2 \right|^2} \left[ \underline{r}_0 \cdot \left( \frac{\underline{r}'_1}{r'_1} - \frac{\underline{r}'_2}{r'_2} \right) \right] \quad (8)$$

$$\text{with} \quad \underline{r}'' = \underline{r}' + \epsilon_2 \underline{\eta}_2$$



**Fig. 5 Diagram showing the relative location of  $A_i$ ,  $B_i$ ,  $P_j$ , and  $P_j''$  for the bound-vortex core correction. Other symbols defined in the text.**

where the  $\underline{r}''$  is the position vector from either  $A_i$  or  $B_i$  to  $P_j''$  (i.e., either  $\underline{r}''_1$  or  $\underline{r}''_2$ ) and the radial unit vector  $\underline{\eta}_2$  can be calculated as in Eq. (9):

$$\underline{\eta}_2 = \frac{\underline{r}_1 \times \underline{r}_0}{|\underline{r}_1 \times \underline{r}_0|} \quad (9)$$

The  $\underline{r}'_{\perp}$  is the position vector from either  $A_i$ ,  $B_i$  to the projection of  $P_j'$  onto the respective vortex centerline as shown in Eq. (10):

$$\underline{r}'_{\perp} = (\underline{r} \cdot \underline{r}_0) \frac{\underline{r}_0}{r_0^2} \quad (10)$$

### B. Solving for the Circulation Distribution

The typical VSM enforces the slip condition  $\underline{U}_{rel} \cdot \underline{n}|_{3/4-c} = 0$  at the  $3/4$ -c point location on the camber line (i.e., the thickness is reduced to zero in the approximation of thin airfoils).

In our method, however, we make use of the more generic lifting-line fundamental equation as in Eq. (11) [e.g., 12] to create a constraint ( $\mathbf{f}=\mathbf{0}$ ) for the circulation distribution  $\underline{\Gamma}(y)$ :

$$\mathbf{f} = \rho |\underline{U}_{\infty} \times \underline{\Gamma}(y)| - \frac{1}{2} \rho |\underline{U}_{rel} \times \hat{z}_{airf}|^2 c C_l(\alpha, \delta_f) = \mathbf{0} \quad (11)$$

where  $\rho$  is the air density,  $\underline{U}_{\infty}$  is the free-stream air velocity vector,  $\underline{\Gamma}(y)$  is the circulation vector,  $\underline{U}_{rel}$  is the relative air velocity,  $\hat{z}_{airf}$  is the unit vector along the airfoil z-axis,  $c$  is the chord length,  $C_l(\alpha, \delta_f)$  is the 2D lift coefficient as a function of  $\alpha$  and  $\delta_f$ ,  $\alpha$  is the effective angle of attack seen by the airfoil,  $\delta_f$  is the airfoil's flap or aileron deflection. The nonlinearity stems from  $C_l(\alpha, \delta_f)$  and from the  $U_{rel}^2$  term. Eq. (11) represents an array of constraints, one per  $j$ -th element in the VSM model ( $j = 1..N_{elms}$ ).

In Eq. (11), the unknowns are  $\underline{\Gamma}(y)$ ,  $U_{rel}$ , and  $\alpha$ . The latter two can be expressed as a function of the induced velocity and thus  $\underline{\Gamma}(y)$ . By specifying a control point  $P_j$ , one can write:

$$\begin{aligned} \underline{U}_{rel}(P_j) &= \underline{U}_{\infty}(P_j) + \underline{U}_{ind}(P_j) \\ \underline{U}_{ind}(P_j) &= \sum_i \underline{U}_{ind,i}(P_j) = \sum_i \left[ \underline{U}_{A_i B_i}(P_j) + \underline{U}_{A_i \infty}(P_j) + \underline{U}_{B_i \infty}(P_j) \right] \\ \alpha_j &= \arctan \frac{\underline{U}_{rel,j} \cdot \hat{x}_{airf,j}}{\underline{U}_{rel,j} \cdot \hat{y}_{airf,j}} \end{aligned} \quad (12)$$

where  $\underline{U}_{rel}$  is the relative air velocity,  $\underline{U}_{ind}$  is the induced velocity,  $\hat{x}_{airf}$  is the unit vector along the airfoil x-axis,  $\hat{y}_{airf}$  is the unit vector along the airfoil y-axis. All these quantities are calculated at the local airfoil station identified by the control point,  $P_j$ .

Eq. (11) has the great advantage of incorporating the generic polar curve of an airfoil through  $C_l(\alpha, \delta_f)$ , thus accounting for nonlinear effects of actual airfoils (beyond the simple flat plate of the lifting-line method) and for the presence of flap or moving-surface deflections. Other methods exist in the literature to account for nonlinear polars, but they introduce more complications with multiple solving loops and with mixed success [13, 14].

Our innovative method, however, does not converge to the correct solution as it is written. The reason is that Eq. (11) in the lifting-line sense should be enforced at the  $1/4$ -c, whereas we are using the  $3/4$ -c control point location



to account for the effects of camber. In order to bring the solution back on track, we must account for the effect of a two-dimensional (2D) contribution to the induction. An approach that is based on and further described in [15], which builds upon the work described in [16]. The new induced velocity is calculated as in Eq. (13):

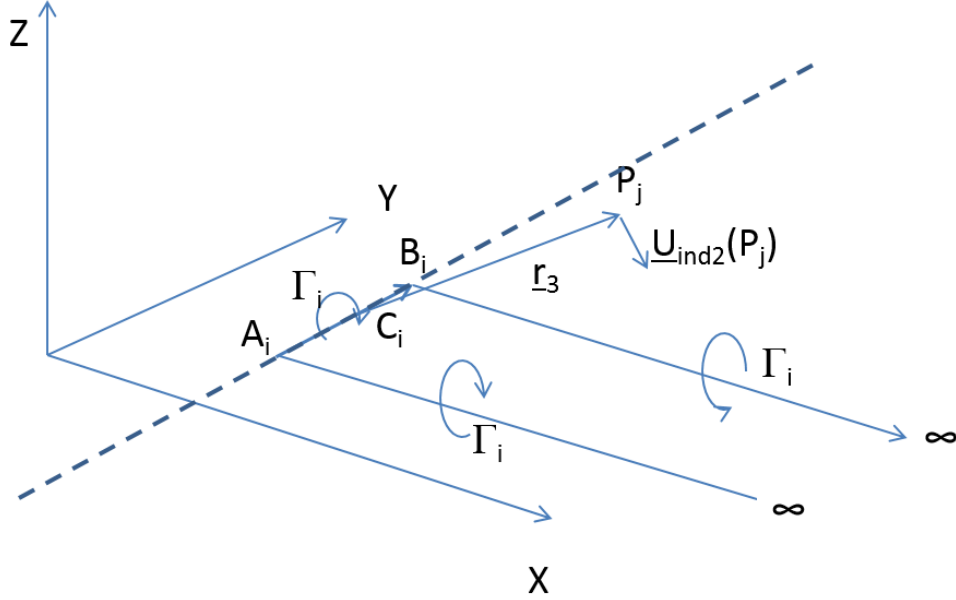


Fig. 6 Contribution of the generic  $i$ -th 2D bound vorticity to the induced velocity at the generic  $P_j$ .

$$\underline{U}_{ind,i}(P_j) = \underline{U}_{A_i B_i}(P_j) + \underline{U}_{A_i \infty}(P_j) + \underline{U}_{B_i \infty}(P_j) - \underline{U}_{A_i B_i 2D} \quad (13a)$$

$$\underline{U}_{A_i B_i 2D} = \begin{cases} \frac{\Gamma_i}{2\pi} \frac{\underline{r}_0 \times \underline{r}_3}{|\underline{r}_0 \times \underline{r}_3|^2} r_0 \delta_{ij} \end{cases} \quad (13b)$$

where  $\underline{r}_3$  is the position vector from  $C_i$  (midpoint of  $A_i B_i$ ) to  $P_j$ ,  $\underline{U}_{A_i B_i 2D}$  is the contribution to the induced velocity by the 2D bound vorticity aligned with the segment  $A_i B_i$  and it is considered only for control point within the local vortex element,  $\delta_{ij}$  is the Kronecker's delta, equal to 0 unless  $i = j$ , in which case it is equal to 1.

The induced velocity is therefore calculated as the difference between a three-dimensional (3D) and a 2D contribution. By doing so, Eq. (11) can be enforced at each control point where  $U_{rel}$ ,  $\alpha$ , and  $C_l(\alpha, \delta_f)$  are expressed in terms of  $\Gamma_i$ . The resulting nonlinear system of equations can then be solved with a numerical method (e.g., Newton's solver).

$$\underline{U}_{ind}(P_j) = \sum_{i=1}^{N_{elems}} \underline{U}_{ind,i}(P_j) = \sum_{i=1}^{N_{elems}} \left[ \underline{U}_{A_i B_i}(P_j) + \underline{U}_{A_i \infty}(P_j) + \underline{U}_{B_i \infty}(P_j) - \underline{U}_{A_i B_i 2D} \right] \quad (14)$$

The presence of multiple lifting surfaces is simply given by the superposition of the induced velocities (see Eq. (14), which replaces the second in Eq. (12)), i.e., the induced velocity at the generic  $P_j$  is calculated the same way as in Eq. (13), where the contribution of each  $i$ -th horseshoe vortex must be taken into account from all lifting surfaces.

Enforcing Eq. (11) at each control point (one per horseshoe vortex) renders a system of nonlinear equations where the various terms can be calculated as a function of  $\underline{\Gamma}(y)$  as shown in Eq. (12) and Eq. (13), with  $C_l(\alpha, \delta_f)$  calculated from an interpolation of the airfoil polar data. The system of equations can be solved via a Gaussian method (e.g., as done in KiteAD).

#### IV. KiteVSM and KiteFAST's Modularization Framework

The integration of KiteAD in KiteFAST was performed following the modularization framework as described in [2]. KiteAD plays the analogous role in KiteFAST as AeroDyn version 15 (AeroDyn15) in the turbine aero-servo-hydro-elastic code OpenFAST (<https://github.com/openfast>) (OpenFAST). KiteVSM is a submodule of KiteAD that replaces

the blade element momentum theory component used within AeroDyn15 [17]. Here, only a few key aspects of the implementation are discussed, including KiteVSM's inputs, outputs, and states.

### A. Inputs, Outputs, Parameters, States

The parameters ( $p$ ) for KiteVSM are:

- $VSM_{mod}$ : flag indicating whether control points and trailing vortices are aligned along the chord (1) or the local free-stream direction (2)
- $\delta_y$ : finite length of the generic bound vortex segment per wing element
- $c$ : chord length per wing element
- $\theta_a$ : aerotwist per wing element
- $AFIP_{params}$ : airfoil static tables of  $C_l$ ,  $C_d$ ,  $C_m$  and dynamic stall (DS) parameters
- $AFidx$ : index pointing to the correct airfoil file/table/database
- $\nu$ : air kinematic viscosity

The *inputs* ( $u$ ) to KiteVSM are:

- $\underline{U}_\infty$ : free-stream air velocity vector. More specifically, KiteVSM requires the components of  $\underline{U}_\infty$  (including contributions from structural body motions and wind velocity) in the plane of the local element airfoil ( $\hat{x}_{airf}$ ,  $\hat{y}_{airf}$ )
- $A, B$  coordinates in the aircraft reference frame ( $O, x_b, y_b, z_b$ ) for every ( $i = 1..N_{elms}$ ) vortex element. From these coordinates, the local chord length, and the  $\underline{U}_\infty$ , the  $C$  and  $P$  coordinates are calculated.
- $\theta$ : structural twist deflection plus aerotwist per element
- $\delta_f$ : flap or aileron deflection (control setting per element)

Note that these inputs are for a given node (within a given lifting surface) and chord. Furthermore, the air density value is used to calculate the dimensional load at each spanwise station.

The KiteVSM *states* are constraint states ( $x_z$ ):

- $\Gamma(y)$ : circulation

The *outputs* ( $y$ ) from the KiteVSM are:

- $\hat{F}_x$ : shear force per unit length along the airfoil's x-axis
- $\hat{F}_y$ : shear force per unit length along the airfoil's y-axis
- $\hat{M}_z$ : torque moment per unit length about the airfoil z-axis

Strictly speaking, these outputs are forces and moments per unit length. In order to transfer loads to MBDyn, these will be converted into actual forces and moments ( $F_x$ ,  $F_y$ ,  $M_z$ ) applied at the various nodes ( $C_i$ ) along the wings. With reference to Figure 2, Eqs. (15)–(17) show how these outputs are calculated when plugging in the local values of  $U_{rel}$ ,  $\alpha$ ,  $c$ ,  $\delta_y$  (finite length of the generic bound vortex segment) and by using Eq. (12) with the calculated state  $\Gamma_i$ .

$$F_x = 0.5 \cdot \rho U_{rel}^2 c \delta_y [C_l(\alpha, \delta_f) \cdot \cos \alpha + C_d(\alpha, \delta_f) \cdot \sin \alpha] \quad (15)$$

$$F_y = 0.5 \cdot \rho U_{rel}^2 c \delta_y [-C_l(\alpha, \delta_f) \cdot \sin \alpha + C_d(\alpha, \delta_f) \cdot \cos \alpha] \quad (16)$$

$$M_z = 0.5 \cdot \rho U_{rel}^2 c^2 \delta_y \cdot C_m(\alpha, \delta_f) \quad (17)$$

where  $C_l(\alpha, \delta_f)$ ,  $C_d(\alpha, \delta_f)$ ,  $C_m(\alpha, \delta_f)$  are given by interpolation of the airfoil tables ( $AFIP_{params}$ ) and  $\delta_y$  is the finite length of the generic bound vortex segment. Note that these forces and moments are considered applied at the  $1/4$ - $c$  points.

### B. KiteVSM Implementation

KiteVSM makes use of the standard framework *Init*, *UpdateStates*, *CalcOutput*, and *End* routines (refer to [18]). In addition to local I/O routines that also handle the outputs to files, a few additional local routines solve the VSM system of equations within *UpdateStates*. The solution of the constraint system of equations is done by means of LAPACK library functions (<http://www.netlib.org/lapack>) that implement an iterative, Gaussian method.

### 1. *KiteVSM\_Init Routine*

This routine allocates the module's data structures, initializes the module's states, and sets the nontime-varying parameters at initialization (see also Section IV.A).

In particular,  $\Gamma(y)$  is set equal to an elliptical distribution at the first time step.

### 2. *KiteVSM\_UpdateStates Routine*

In this routine, for a given set of *inputs* ( $u$ ), and at the current step in time ( $t_i$ ),  $\Gamma(y)_i$  is used as initial guess to solve for  $\Gamma(y)_{i+1}$  by numerically solving Eq. (11), together with Eqs. (2), (13), and (12). *Inputs* are at  $t_{i+1}$ .

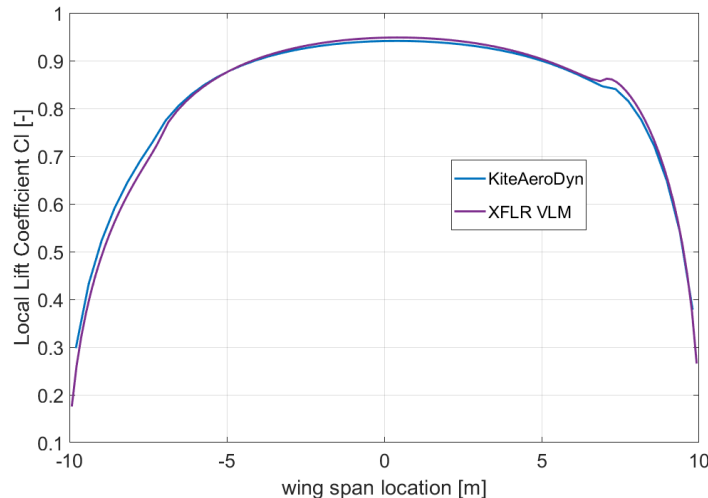
### 3. *KiteVSM\_CalcOutput*

Here, the *outputs* are calculated by using Eqs. (15)–(17), and using Eq. (12) with the calculated state  $\Gamma_i$  through Eqs. (2), (5), (8), (13), and (14).

## V. Preliminary Verification

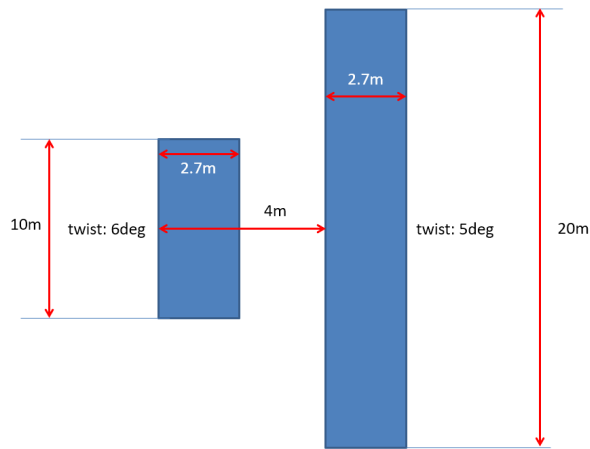
KiteAD was verified against XFLR version 5 (<http://www.xflr5.com/xflr5.htm>) (XFLR5) for several different lifting surface configurations. XFLR5 is an analysis tool for airfoils, wings and planes operating at low Reynolds numbers. Among its capabilities is the prediction of lift distributions for finite wing sections using a vortex lattice method. More details on XFLR5 and its vortex lattice method can be found in [19].

Results for two verification cases are shown below. Fig. 7 shows the lift distribution for a wing with 20 m span and opposite flap deflections on each end of the wing. This case is particularly telling as it considers three different airfoil cross sections (NACA1410 with 3° downward flap deflection, clean NACA1410, and NACA1410 with 4° upward flap deflection) distributed along the wing. The nonsymmetric features in the lift distribution because of the flap deflections are predicted in a similar fashion by XFLR5 and KiteAD. This verifies the correct consideration of varying airfoil polar data by KiteVSM ( $C_l(\alpha, \delta_f)$  in Eq. 11).

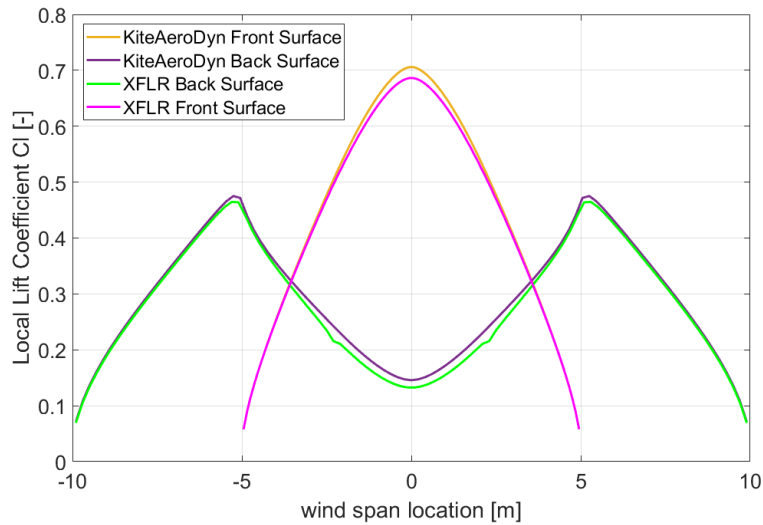


**Fig. 7 Comparison of lift distribution between KiteAeroDyn and XFLR for the first verification case: A straight wing (NACA1410 airfoil) with 2.7 m chord length and flap deflection of  $-3^\circ$  on the left and  $4^\circ$  on the right side.**

The second verification case shown here involves two lifting surfaces that are arranged in a canard-type configuration as shown in Fig. 8. This specific case is aimed at verifying KiteAD's capability to predict the aerodynamic interaction of multiple lifting surfaces. A canard configuration is also very useful for verification as the strong trailing vortices from the forward surface interact with the aft surface. Except for minor differences in the predicted lift distribution between KiteAD and XFLR5 in the middle section of the wing, KiteAD and XFLR5 show very good agreement as illustrated in Fig. 9.



**Fig. 8** Geometric layout for the second verification case: Two straight wings (flat plates with 10 m and 5 m span and 2.7 m chord length). Wing 1 ( $6^\circ$  constant twist) is located 4 m forward of wing 2 ( $5^\circ$  constant twist).



**Fig. 9** Comparison of lift distribution between KiteAD and XFLR5 for the second verification case: Two straight wings (flat plates with 10 m and 5 m span and 2.7 m chord length). Wing 1 ( $6^\circ$  constant twist) is located 4 m forward of wing 2 ( $5^\circ$  constant twist).

## VI. Conclusion

This paper presented a brief overview on different lift models for finite surfaces and outlined the theoretical foundation of a novel Vortex Step Method implemented in KiteAD. Results from a verification study against XFRL5 were presented and discussed. The verification results are encouraging and an upcoming publication will cover a comprehensive verification campaign of KiteFAST. Especially for larger angles of attack where the airfoil polar becomes increasingly nonlinear, the outlined vortex step method could potentially show improved performance as it considers these nonlinearities in the airfoil polar data. Further model verification studies are currently being conducted at NREL.

## VII. Acknowledgment

Thanks to Makani, a project of X, for providing support for this research. This work was authored by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding provided by the U.S. Department of Energy Office of Energy Efficiency and Renewable Energy Wind Energy Technologies Office. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

## References

- [1] Masarati, P., “MBDyn Input File Format Version 1.7.3,” Tech. rep., Dipartimento Di Ingegneria Aerospaziale Politecnico Di Milano, 2017. URL <https://www.mbdyn.org/userfiles/documents/mbdyn-input-1.7.3.pdf>.
- [2] Jonkman, J., “New Modularization Framework for the FAST Wind Turbine CAE Tool,” *Proceedings of the 51st AIAA Aerospace Sciences Meeting*, AIAA, Dallas, TX, 2013.
- [3] Weissinger, J., “The Lift Distribution of Swept-Back Wings,” Tech. Rep. NACA-TM-1120, National Advisory Committee for Aeronautics, Langley Aeronautical Lab., Langley Field, VA, March 1947.
- [4] Prandtl, L., “Theory of Lifting Surfaces,” Tech. Rep. NACA-TR-116, National Advisory Committee for Aeronautics, Washington, DC, 1918.
- [5] Munk, M. M., “The Minimum Induced Drag of Aerofoils,” Tech. Rep. NACA-TR-121, National Advisory Committee for Aeronautics, Washington, DC, 1921.
- [6] Pistolesi, E., “Alcune considerazioni sul problema del biplano indefinito,” Tech. rep., Rome University, 1929.
- [7] Wieghardt, K., “Chordwise Load Distribution of a Simple Rectangular Wing/WinLoad,” Tech. Rep. NACA-TM-963, National Advisory Committee for Aeronautics, Washington, DC, 1940.
- [8] Mutterperl, W., “The Calculation of Span Load Distributions of Swept-back Wings,” Tech. Rep. NACA-TN-834, National Advisory Committee for Aeronautics, Washington, DC, 1941.
- [9] Phillips, W. F., and Snyder, D. O., “Modern Adaption of Prandtl’s Classic Lift-Line Theory,” *J. of Aircraft*, Vol. 37, No. 4, 2000, pp. 600–608.
- [10] Bhagwat, M. J., and Leishman, J. G., “Generalized Viscous Vortex Model for Applications to Free-Vortex Wake Aeroacoustic Calculations,” *58th Annual Forum and Technology Display of the American Helicopter Society International*, Montreal, Canada, 2002.
- [11] van Garrel, A., “Development of a Wind Turbine Aerodynamics Simulation Module,” Tech. Rep. ECN-C-03-079, ECN, The Netherlands, August 2003.
- [12] Anderson, J. D. J., *Fundamentals of Aerodynamics*, 3<sup>rd</sup> ed., McGraw-Hill Series in Aeronautical and Aerospace Engineering, McGraw-Hill, 2001.
- [13] van Dam, C. P., Vander Kam, J. C., and Paris, J. K., “Design-Oriented High-Lift Methodology for General Aviation and Civil Transport Aircraft,” *J. of Aircraft*, Vol. 38, No. 6, 2001, pp. 1076–1084.

- [14] Ortega, M. A., M., G. R., and Komatsu, P. S., “A Numerical Method to Predict the Lift of Aircraft Wings at Stall Conditions,” *ENCIT 2004 – ABCM*, Rio de Janeiro, Brazil, 2004.
- [15] Ranneberg, M., “Direct Wing Design and Inverse Airfoil Identification with the Nonlinear Weissinger Method,” arXiv:1501.04983 [physics.flu-dyn], June 2015.
- [16] Piszkin, S. T., and Levinsky, E. S., “Nonlinear Lifting Line Theory for Predicting Stalling Instabilities on Wings of Moderate Aspect Ratio,” Tech. Rep. CASD-NSC-76-001, General Dynamics Convair Division, 5001 Kearny Villa Rd., San Diego, CA 92138, June 1976.
- [17] Jonkman, J. M., Hayman, G. J., Jonkman, B. J., and Damiani, R. R., “AeroDyn v15 User’s Guide and Theory Manual,” Tech. rep., National Renewable Energy Laboratory, Golden, CO, 2015.
- [18] Jonkman, B., Michalakes, J., Jonkman, J., Buhl, M., Platt, A., and Sprague, M., “NWTC Programmer’s Handbook: A Guide for Software Development Within the FAST Computer-Aided Engineering Tool,” Tech. Rep. NREL/TP-XXXX-XXXXX, NREL, Golden, CO, April 2013.
- [19] “XFLR5 - Project Page,” available online, June 2018. URL <http://www.xflr5.com/xflr5.htm>.