Abstract: In this paper we apply a system identification algorithm and an adaptive controller to a simple kite system model to simulate crosswind flight maneuvers for airborne wind energy harvesting. The purpose of the system identification algorithm is to handle uncertainties related to a fluctuating wind speed and shape deformations of the tethered membrane wing. Using a pole-placement controller we determine the required locations of the closed-loop poles and then enforce them by adapting the control gains in real time. We compare the path-following performance of the proposed approach with a classical PID controller using the same system model. The capability of the system identification algorithm to recognize sudden changes in the dynamic model or the wind conditions and the ability of the controller to stabilize the system in the presence of such changes are confirmed. Furthermore, the system identification algorithm is applied to determine the parameters of a kite with variable-length tether used in a flight test of the 20 kW kite power system of TU Delft. Experimental data of this test were analyzed using the system identification algorithm in real time and significant changes were observed in the parameters of the dynamic model which strongly affect the resulting response.

Keywords: airborne wind energy; kite system; system identification; adaptive algorithms; pole placement
l_t  

P_3  

P_4  

n_r  

r  

n_s  

S  

U(z^{-1})  

u_d  

u_s  

v_{w,ref}  

x_k, y_k, z_k  

x_{SE}, y_{SE}, z_{SE}  

Y(z^{-1})  

y_m  

z^{-1}  

Greek Symbols  

\beta  

elevation angle of the kite  

\beta_{sw}  

elevation angle to switch flight mode  

\chi  

course angle of the kite  

\dot{\chi}  

rate of change of the course angle  

\dot{\chi}_R  

rate of change of the course angle to fly a turn with radius R  

\chi_{set}  

set value for the course angle  

\delta_{min}  

minimal, angular attractor point distance  

\omega_{ref}  

reference value of the angular speed  

\omega_{fig}  

angular width of figure-of-eight maneuver  

\phi  

azimuth angle of the kite  

\phi_{C_2}  

azimuth angle at point C_2  

\phi_{set}  

set value of azimuth angle  

\phi_{sw}  

azimuth angle to switch flight mode  

\psi  

heading angle of the kite  

\dot{\psi}  

turn rate of the kite  

\varrho  

turn radius of the trajectory of the kite point K  

Vectors and Matrices  

\omega  

angular velocity of the kite point K with respect to the origin O  

P_k  

position of the kite in angular coordinates(\phi, \beta)  

P_{k,SE}  

covariance matrix of the estimated error  

\theta  

last vector estimated using the least square estimation algorithm  

v_a  

apparent wind speed  

v_k  

take velocity  

v_{k,r}  

radial kite velocity component  

v_{k,t}  
	angential kite velocity component  

X  

data of old measurement of course angle and control action  

Y_m  

measured course angle obtained from the sensor  

Abbreviation  

AWE  

Airborne Wind Energy  

FPC  

Flight Path Control  

FPP  

Flight Path Planner  

HAWT  

Horizontal Axis Wind Turbines  

KCU  

Kite Control Unit  

LEI  

Leading Edge Inflatable  

MSE  

Mean Square Error  

NDI  

Nonlinear Dynamic Inversion
1. Introduction

Airborne wind energy (AWE) is an emerging renewable energy technology which uses flying devices that are tethered to the ground [1–3]. Compared to horizontal axis wind turbines (HAWTs), AWE systems have a number of distinct advantages in terms of costs, maintenance, operational altitude and capacity factor. However, as wind turbines have matured over decades of continuous research and development, AWE technologies, which are at a comparatively early stage of development, are still considered to be less reliable. For a HAWT, almost 30% of the power is generated by the tip of the rotor blades while the rest of the rotor functions mainly as support structure for the crosswind motion of the blades [1,2,4]. The rated power of the generator typically determines the installation. For the same rated power, an AWE system generally gives a higher annual yield than a HAWT because it can operate at a higher capacity factor. The higher capacity factor is a result of the more persistent and more steady wind at higher altitude. However, an AWE system also needs more space than a HAWT, which increases the costs of an installation. These land surface costs are still quite unknown and responsible for the large differences in expected costs [5,6].

There are several different concepts and configurations of AWE systems [2,8]. A comparatively simple one is the pumping kite power system illustrated in Fig. 1. The kite consists of a flexible membrane wing that is steered by a suspended kite control unit (KCU) and tethered to a drum on the ground which is coupled to a generator. During reel out of the tether, the kite is operated in crosswind figure-of-eight maneuvers, as shown in Fig. 2, to maximize the pulling force and thus generated energy when pulling the tether from the drum. When reaching the maximum tether length, the kite is depowered and retracted, using the generator as a motor and consuming a fraction of the formerly generated energy. The change of the flight patterns between reel-out and reel-in phases results in a net energy per pumping cycle [5,9]. The main objective of the control algorithm is to ensure a robust and safe flight operation of the kite.

Several mathematical models have been developed to describe the dynamic behavior of the kite system [1,3,10,11]. An obvious shortcoming of these models are the numerous idealizations.
that limit the achievable prediction quality for a flight trajectory in a real wind environment which is characterized by local changes in wind speed and direction occurring within seconds [12,13].

Another modeling challenge is the aeroelastic response of the membrane wing because the substantial deformations also majorly affect the aerodynamic properties of the kite system [14–16].

Generally, in most control applications, the mathematical model can be based on either: (1) a system that has a fixed operating point and control parameters that are fixed due to the stability in the model, or (2) a system that has a variable operating point requiring that it is robust enough to sustain the variation. The dynamic model representing a kite system is of the latter type since the system characteristics are time varying due to the natural fluctuations of the wind environment. This has to be considered in the design of the control algorithm for the kite system.

In recent years, research on kite control has intensified considerably [17–20]. The developed models are typically based on state space representations of the real AWE system, describing the nonlinear dynamics of 4 up to 15 states. In [21] the optimal flight path of the kite is determined ahead of time, using nonlinear model predictive control (NMPC). However, this control method is computationally expensive and with additional expenses for solving the nonlinear optimization problem [19,20]. For this reason we consider the method unsuitable for real-time experiments in kite applications.

Several open source software tools for AWE systems have been developed over the past years [11,22,23]. These tools allow the computational simulation of flight maneuvers, virtual testing of flight control algorithms and optimizing flight paths in real time. One particular aspect of modeling is the flexible tether that is deployed from the ground station and sweeping through the wind field, while being exposed to a distributed aerodynamic load and gravity [24]. A common assumption is that of a perfectly tensioned, straight tether. While generally valid during the traction phase, this assumption is not acceptable during the retraction, landing and take-off phases. Because of the low tension, sagging of the tether has to be taken into account in the modeling [9,25].

The control approach proposed in [26] is interesting because it does not require any information about the kite or the wind field. Although the approach allows high-precision tracking of figure-of-eight maneuvers, this is feasible only for a short tether of constant length. The approach is not suitable, however, for a long tether or a tether of variable length [12]. The research challenge addressed in this work can be summarized as follows:

**Figure 2.** Composite photo of a crosswind figure-of-eight maneuver ($\Delta t = 1$ s) of a tube kite with 25 m$^2$ surface area [7].
The kite is based on a flexible membrane wing and its shape depends on the aerodynamic force distribution and structural design and line suspension system. The relative flow velocity experienced by wing and tether varies along the flight path and there is no accurate way to assess this in real time. An accurate estimation of the wind field is necessary to determine the aerodynamic force distribution on tether and wing surface [4,5].

As a result of this physical complexity the need for robust control to stabilize the kite in real time is crucial. The challenges mentioned above are addressed as follows:

- We propose an algorithm that estimates the parameters of the dynamic model of the kite in real time using system identification (SI) [27–30]. The employed SI algorithm is a non-iterative technique based on the Plackett’s algorithm due to its ability to calculate the parameters of the system (dynamic model) with high accuracy without singularity.
- We further propose an adaptive controller to improve the robustness of the system and stabilize the kite in different wind conditions. Moreover, SI is considered as a part of the adaptive control and the estimated parameters are used to obtain the control gains [30], which means that these gains are updated in real time based on the change in the dynamic model [28].
- We present a comparison between the adaptive and classical controllers to highlight the robustness of the controller and the ability of adaptive control to stabilize the kite for different wind conditions without any change in the SI and adaptive control algorithms.
- We apply the SI algorithm to experimental data of the 20 kW kite power system of TU Delft [5]. The algorithm is used to estimate the parameters of the system in different operation phases, such as traction and retraction phases, for two pumping cycles while experiencing changes in wind speed and tether length.

The remainder of this paper is subdivided as follows. Section 2 outlines the components of the mathematical model and discusses it briefly based on the concepts and implementations described in [12,28]. The mathematical model includes a simplified kite system model, designed as a single-input single-output (SISO) model relating the relative steering input (input) to the course angle (output) of the kite, a flight path planner (FPP) and a flight path controller (FPC). In Sect. 3 the derivation and implementation of the SI algorithm is discussed in detail. Subsequently, Sect. 4 presents the adaptive controller based on the estimated parameters from the SI algorithm which is used to stabilize the kite in real time. In Sect. 5 simulation results are presented for the classical controller described in [12,13,28] and the combination of adaptive controller and SI algorithm and compared for two different flight conditions. Finally, Sect. 6 summarizes the experimental results acquired during a flight test of the 20 kW kite power system of TU Delft, then these results are analyzed using the SI algorithm derived in Sect. 3. Section 7 concludes this article.

2. Mathematical model

The approach presented in this paper has been derived for the 20 kW kite power system of TU Delft, as illustrated in Figs. 1 and 2. The mathematical model was previously formulated and demonstrated in [12,28]. The flight path planning (FPP) and flight path control (FPC) approaches are both designed on the basis of this model. The presented simulation results were obtained by a Simulink/MATLAB® implementation.

2.1. Simplified kite system model

To describe the flight motion of the kite power system we introduce the wind reference frame \((x_w,y_w,z_w)\), the kite reference frame \((x_k,y_k,z_k)\) and the small earth reference frame \((x_{SE},y_{SE},z_{SE})\), as illustrated in Fig. 3. The wind reference frame is attached to the tether exit point \(O\) at the ground station and rotates with the wind around its \(z_w\)-axis such that its \(x_w\)-axis is always aligned with the instantaneous wind direction. It is assumed that this rotation is slow compared to the flight
Figure 3. Reference frames to describe tethered flight, heading angle $\psi$ and course angle $\chi$ of the kite.

dynamics of the kite such that any induced forces can be neglected. The body-fixed kite reference frame is attached to the center of mass $K$ of the kite and accounts for wing, bridle line system and the suspended kite control unit (KCU). To obtain the coordinates of the kite in the small earth reference frame [31], its position is projected radially onto the unit sphere around the tether exit point $O$. On the unit sphere we define a local tangential plane $\tau$. The position of the kite in the small earth reference frame can be described by the azimuth angle $\phi$ and the elevation angle $\beta$, which represent the latitude and longitude of the position. The velocity $v_k$ of the kite can be decomposed into a radial component $v_{kr}$ and a tangential component $v_{k\tau}$ which can be derived from the kite position as

$$v_{kr} = \dot{r}, \quad v_{k\tau} = r \sqrt{\dot{\phi}^2 \cos^2 \beta + \dot{\beta}^2}. \tag{1}$$

The course angle $\chi$ is defined as the angle between $v_{k\tau}$ and the upwards direction, represented by the local $x_{SE}$-axis [12,13]. The heading angle describes the orientation of the kite in the tangential plane and is defined as the angle between the $x_k$-axis and the upwards direction. The course angle $\chi$ generally follows the heading angle $\psi$ with a varying offset due to gravity. The angular velocity $\omega$ of the kite with respect to the origin $O$ is kinematically coupled to the tangential velocity of the kite

$$v_{k\tau} = \omega \times r, \quad v_{k\tau} = r \omega. \tag{4}$$
where \( \mathbf{r} \) is the vector pointing from origin \( O \) to kite \( K \). The relative flow velocity at the kite is computed from the wind and kite velocity vectors as

\[
\mathbf{v}_a = \mathbf{v}_w - \mathbf{v}_k. \tag{5}
\]

To simplify the flight path planning (FPP) and flight path control (FPC) implementation we introduce the following assumptions, which are valid for the traction phase when the tether is generally fully tensioned:

- The tether is straight and the tether length is identical to the radial position \( r \) of the kite \( K \).
- The wing is attached to the tether with a bridle line system which constrains the roll and the pitch of the wing. Only the heading angle \( \psi \) remains as a degree of freedom to control the course of the kite.
- The difference between heading and course angles can be neglected. Both angles and their time derivatives are assumed to be identical in this study.

By decomposing the kite velocity into radial and tangential components we essentially decouple the flight control of the kite from the winch control. The kite is steered by actuation of the rear bridle line system, which is quantified by the relative steering \( u_s \) that can vary between -1 and 1. Additional inputs for the kite model are the magnitude of the apparent wind speed \( v_a \), the initial elevation angle \( \beta_0 \), the angular speed \( \omega \) and the initial values for the course angle \( \chi_0 \) and the azimuth angle \( \phi_0 \). The outputs for this model are the course angle \( \chi \), its time derivative \( \dot{\chi} \) and the position of the kite in terms of elevation angle \( \beta \) and azimuth angle \( \phi \).

To calculate the angular speed \( \omega \) of the kite we use the simplified kite system model that was derived in [12]. In this model, which has also been used in [28], it is assumed that the angular speed \( \omega \) is a function of the elevation angle \( \beta \) only. The angular speed is zero at the maximum value \( \beta_{\text{max}} \) and increases linearly with decreasing elevation angle, until it reaches a specific value \( \omega_{\text{ref}} \) at \( \beta_{\text{min}} \). This behavior is described by the following correlation

\[
\omega = \frac{\beta_{\text{max}} - \beta}{\beta_{\text{max}} - \beta_{\text{min}}} \omega_{\text{ref}}. \tag{6}
\]

Flight tests of the TU Delft Hydra kite [5,9] have shown that the maximum elevation angle \( \beta_{\text{max}} = 73^\circ \) is reached when the kite is positioned statically at a tether length of 300 m at an approximate ground wind speed of 6 m/s. We further assume that the value \( \omega_{\text{ref}} \) is a linear function of the apparent wind velocity \( v_{a,\text{ref}} \), which can be expressed as

\[
\omega_{\text{ref}} = \frac{v_a}{v_{a,\text{ref}}} \omega_{\text{ref},0}. \tag{7}
\]

where \( \omega_{\text{ref},0} \) denotes the angular speed that is reached at a specific apparent wind speed \( v_{a,0} \). From the flight tests we find an angular speed \( \omega_{\text{ref},0} = 5 \text{ deg/s} \) for an elevation angle \( \beta_{\text{min}} = 22^\circ \) and an apparent wind velocity \( v_{a,\text{ref}} = 20 \text{ m/s} \). Combining Eqs. (6) and (7) we can calculate the angular speed \( \omega \) of the kite as a function of the elevation angle \( \beta \) and apparent wind velocity \( v_a \). Multiplying the angular speed of the kite with the radial coordinate \( r \) we can calculate the tangential speed \( v_{k,\tau} \), as described by Eq. (4).

We determine the course angle of the kite by integrating the turn rate law, which is an empirical correlation between the rate of change \( \dot{\psi} \) of the heading angle (identical to the rate of change \( \dot{\chi} \) of the course angle), the apparent wind speed \( v_a \) and the relative steering action \( u_s \) [12,26,28]

\[
\dot{\psi} = c_1 u_s (u_s - c_v) + \frac{c_2}{v_a} \sin \chi \cos \beta. \tag{8}
\]

The second term in Eq. (8) quantifies the effect of gravity on the turn rate. In a next step, we combine Eqs. (2) and (4) to eliminate \( \dot{v}_{k,\tau} \) and calculate the time derivatives of the azimuth and elevation angles.
from the angular speed $\omega$. These derivatives are integrated to determine the angular position of the kite.

To summarize, we note that the simplified kite model describes a single-input single-output (SISO) system. The model is characterized by two position and two velocity state variables, $(\phi, \beta)$ and $(\omega, \chi)$, respectively, three steering parameters $(c_0, c_1, c_2)$ and one relative steering action $u_s$.

2.2. Flight path planner (FPP)

The purpose of the FPP is to design a suitable flight trajectory for a pumping cycle. The path is constructed in the $\phi$-$\beta$ space as a sequence of connected line segments on the small earth. As consequence, a finite state diagram can be used to describe the flight control of the kite. The states of the high-level controller required for automated power production are explained in Figs. 4 and 5. For the remainder of this paper our focus will be on the figure-of-eight maneuvers during the traction phase.

![Figure 4. FPP for the entire pumping cycle, including the four-step planner for down-loop figure-of-eight maneuvers: First, turn left, then steer towards attractor point $P_3$, then turn right and finally steer towards attractor point $P_4$ [12].](image-url)
At the end of the retraction phase the kite is powered to improve the controllability before flying the more complex crosswind maneuvers. In this state the kite is first depowered and then retracted. Eventually the kite is flying at a high elevation angle on a short tether.

In the upper transition phase \(\text{INT}_\text{UP}\) the kite is first turning right towards zenith and then steered towards zenith until a high elevation angle is reached.

\[\{l_t > l_{\text{up}} \lor z > z_{\text{up}}\} \land \phi < -\phi_3\]

In this state the kite is flying figure-of-eight maneuvers while reeling out the tether. Following a left turn it is flying towards a point on the right side of the wind window, after which it is turning right to fly to the right side.

![Finite state diagram](image)

Figure 5. Finite state diagram for the states of high level controller for full automated power production.

The downloop\(^1\) flight maneuvers are constructed in four steps, as shown in Table 1 and Fig. 6. The flight maneuver starts from the initial position of the kite with a left turn (substate \(\text{TURN}_\text{LEFT}\)). During the turn, the set value \(\dot{\chi}_{\text{set}} = \dot{\chi}_R\) computed from the turn rate law given by Eq. (8) is used. When reaching the switch condition \(\chi > 300^\circ - \delta\chi\) the turning maneuver is stopped and the kite starts flying to the right towards the attractor point \(P_3\) (substate \(\text{FLY}_\text{RIGHT}\)). On this segment of the figure-of-eight, the kite is steered by the PID controller, as described in Sect. 2.3. A relatively large offset of \(\delta\chi \approx 112^\circ\) is needed to compensate for the time delay \(\delta t \approx 2\) s between the command to stop turning and the kite actually stopping to turn [12]. For clarity of illustration we use in Figs. 4 and 7 a value of \(\delta\chi = 0\) such that the circular path segments with constant turn rate \(\dot{\chi}_R\) directly connect with the straight path segments with constant course angle \(\chi\). When reaching the switch condition \(\phi < -\phi_{\text{sw}}\) the kite starts a right turn (substate \(\text{TURN}_\text{RIGHT}\)). During the turn, the set value \(\dot{\chi}_{\text{set}} = \dot{\chi}_R\) computed from the turn rate law is used. When reaching the switch condition \(\chi < 60^\circ + \delta\chi\) the turning maneuver is stopped and the kite starts flying to the left towards the attractor point \(P_4\) (substate \(\text{FLY}_\text{LEFT}\)). On this segment of the figure-of-eight, the kite is again steered by the PID controller. When reaching the switch condition \(\phi > \phi_{\text{sw}}\) a new figure-of-eight maneuver is started with the kite entering a left turn (substate \(\text{TURN}_\text{LEFT}\)).

\(^1\) We distinguish between downloops and uploops depending on the direction of flight during the turning maneuvers along the outer parts of the figure-of-eight. While downloops lead to a more equalized power profile during the traction phase, uploops are generally considered to be more safe.
The geometry of the figure-of-eight flight path is defined as illustrated in Fig. 4 by the angular width $w_{fig}$ and height $h_{fig}$, the minimal attractor point distance $\delta_{min}$, defined as the arc length on the unit sphere between the kite position and the current attractor point at which the kite stops flying towards this attractor point and starts to make a turn. If the aforementioned parameters are specified, then $P_3$, $P_4$, $\chi_R$ and $\phi_{sw}$ can be calculated by the FPP. The motion of the kite along the planned trajectory is described by the tangential velocity of the kite $v_{k,\tau}$, defined by Eq. (2), and the turning radius, defined as

$$\varrho = \frac{h_{fig}}{2}.$$  

(9)

The rate of change $\dot{\chi}_R$ of the course angle required to fly a turn with radius $R$ is calculated as

$$\dot{\chi}_R = \frac{v_{k,\tau}}{R} = \frac{\omega r}{R},$$  

(10)

where the angular velocity $\omega$ of the kite with respect to the origin is given by Eq. (6). We note that the turning radius $\varrho$ as defined by Eq. (9) is an arc length on the unit sphere, while the radius of curvature $R$ as used in Eq. (10) is a distance in Cartesian space. For practically relevant figure-of-eight maneuvers with small turning radius we can use the following approximation

$$\varrho = \frac{R}{r}.$$  

(11)
The value of $\phi_{c2}$ can be calculated from

$$\phi_{c2} = \frac{w_{fig}}{2} - \varrho.$$  

(12)

Then, the switch values $\phi_{sw}$ and $\beta_{sw}$ of the azimuth and elevation angles can be calculated from Eqs. (13) and (14) by combining the circle segment of the left turn with the tangent. Figure 7 illustrates the derivation of the following equations using simple geometrical relations

$$\phi_{sw} = \phi_{c2} - \frac{\varrho^2}{\phi_{c2}},$$  

(13)

$$\beta_{sw} = \sqrt{\varrho^2 - (\phi_{sw} - \phi_{c2})^2 + \beta_{set}}.$$  

(14)

The slope of the line towards $P_4$ can be calculated from

$$k = \sqrt{\frac{\phi_{c2} - \phi_{sw}}{\phi_{sw}}}.$$  

(15)

Solving for the attractor points $P_3$ and $P_4$, we obtain

$$P_3 = \left( -\phi_{sw} - \delta_{\min} \sqrt{\frac{1}{1 + k^2}}, \beta_{sw} + \delta_{\min} k \sqrt{\frac{1}{1 + k^2}} \right),$$  

(16)

$$P_4 = \left( \phi_{sw} + \delta_{\min} \sqrt{\frac{1}{1 + k^2}}, \beta_{sw} + \delta_{\min} k \sqrt{\frac{1}{1 + k^2}} \right).$$  

(17)

### 2.3. Flight path control (FPC)

The FPC uses the attractor points $P_3$ and $P_4$ to guide the kite during the FLY_RIGHT and FLY_LEFT substates of the figure-of-eight maneuver. The required course angle $\chi_{set}$ is calculated from the set values of the elevation and azimuth angles using great circle navigation [32]

$$y_k = \sin(\phi_{set} - \varphi) \cos \beta_{set},$$  

(18)

$$x_k = \cos \beta \sin \beta_{set} - \sin \beta \cos \beta_{set} \cos(\phi_{set} - \varphi),$$  

(19)

$$\chi_{set} = \text{atan}2(-y_k, x_k).$$  

(20)
Since the kite model is designed as a SISO system, it has just one error signal that comes from the difference between the actual course angle of the kite, determined by integration of Eq. (8), and the set value for the course angle given by Eq. (20). This error signal is fed into a PID controller, which uses the relative steering input $u_s$ to align the tangential velocity of the kite with the planned flight direction.

The kite is steered along the turns of the figure-of-eight maneuver using a feed-forward controller with the set value $\chi_{\text{set}} = \chi_R$, computed from the turn rate law given by Eq. (8), to fly a turn with radius $R$ (or $\varrho$ in $\phi$-$\beta$ space). This set value is used as input of a nonlinear dynamic inversion (NDI) block to calculate the relative steering action $u_s$ that is required to fly the respective turn. The functions used by the NDI block are detailed in [12].

3. System identification (SI) using Plackett's algorithm

The aim of the SI algorithm is to estimate the system parameters during automatic flight using sensor data. Therefore, it is required to update the parameters in real time by analyzing the history of the control action $u_s$ and the course angle $\chi$ [28]. There are several techniques for SI. In this paper we use Plackett’s algorithm [29,30] as a technique to update the dynamics of the system. This specific algorithm has the advantage of rapidly acquiring the system parameters, without iterations, has no singularity and the implementation on a micro-controller is simple and can be used for real-time processing for flight tests.

The algorithm is built based on the minimization of the mean square error (MSE) of the course angle $\chi$ as defined by

$$MSE = \frac{1}{k} \sum_{r=1}^{k} (Y_r - Y_{m,r})^2, \quad (21)$$

where $k$ is total number of time steps in the discrete time process, $Y_{m,r}$ is the measured data for time step $r$ and $Y_r$ the estimated value determined by the SI algorithm. The open-loop transfer function (TF) of the kite is derived in [28] and it was used as a case study. The control action $u_s$ will be denoted as $U(z^{-1})$ and the course angle $\chi$ will be denoted as $Y(z^{-1})$. The block diagram of the SI algorithm and adaptive control system is illustrated in Fig. 8.

![Block diagram of the SI algorithm and adaptive control system.](image)

The SI algorithm will predict the estimated course angle $\chi$ and update the coefficients of the open-loop TF $a_1, a_2, b_1$ and $b_2$, after that the adaptive control will update its gains based on the parameters of the open-loop TF to stabilize the kite as described in Sect. 4. The data discussed in Subsects. 5.1 and 5.2 considered the course angle resulted from the model in Sect. 2 as a measured course angle. This angle was used with the steering values of the motor to calculate the estimated course angle of the two flight conditions 5.1 and 5.2. Finally, the comparison between the results of
the simplified model control by PID controller and the adaptive control are discussed in Subsects. 5.1 and 5.2.

The SI algorithm was implemented in different way in Sect. 6.2. It utilized the measured course obtained from the real flight test of V3 kite by the TU Delft research group, not from the simplified model in Sect. 2, and the relative steering from the motor to identify the coefficients of the open loop TF without developing any controller algorithms. The open-loop TF for the kite in $z$-form [33] can be approximated as

$$G(z^{-1}) = \frac{Y(z^{-1})}{U(z^{-1})} = \frac{B(z^{-1})}{A(z^{-1})}, \quad (22)$$

where $A(z^{-1})$ and $B(z^{-1})$ are considered as second order polynomial equations in $z$-form

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2}, \quad (23)$$
$$B(z^{-1}) = b_1 z^{-1} + b_2 z^{-2}. \quad (24)$$

The coefficients $a_1, a_2, b_1$ and $b_2$ are varying with time because of the change in the system dynamics. The kite is also exposed to a time-varying apparent wind speed which is not available in real time. Substituting Eqs. (23) and (24) into Eq. (22), we obtain

$$\frac{Y}{U}(z^{-1}) = G(z^{-1}) = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}. \quad (25)$$

This equation can be rewritten in difference form

$$Y_k = -a_1 Y_{k-1} - a_2 Y_{k-2} + b_1 U_{k-1} + b_2 U_{k-2}, \quad (26)$$

or reformulated as a matrix expression

$$Y_k = X_{k-1}^\top \theta_{k-1}, \quad (27)$$

where

$$X_{k-1} = [Y_{k-1}, Y_{k-2}, U_{k-1}, U_{k-2}]^\top, \quad (28)$$
$$\theta_{k-1} = [-a_1, -a_2, b_1, b_2]^\top. \quad (29)$$

From Eq. (21), the MSE can be written as

$$MSE = \frac{1}{k} \sum_{r=1}^{k} \left( X_{r-1}^\top \theta_{r-1} - Y_{m,r} \right)^2. \quad (30)$$

The objective of the SI algorithm is to obtain the values of the coefficient matrix $\theta$ that minimize the MSE. From the derivation, these values can be calculated as

$$\theta_k = P_k \sum_{r=1}^{k} X_{r-1} Y_{m,r}, \quad (31)$$

where $P_{k-1}$ is a square matrix such that

$$P_{k-1} = \left[ \sum_{r=1}^{k} (X_{r-1} X_{r-1}^\top) \right]^{-1}. \quad (32)$$

From Eq. (32) we obtain

$$P_k^{-1} = P_{k-1}^{-1} + (X_{k-1} X_{k-1}^\top). \quad (33)$$
Equation (31) is now rewritten as

\[ \theta_k = P_k \left[ X_{k-1} Y_{m,k} + \sum_{r=1}^{k-1} (X_{r-1} Y_{m,r}) \right]. \]  

(34)

From Eqs. (34) and (31) we find

\[ \theta_k = P_k X_{k-1} Y_{m,k} + P_k P_{k-1}^{-1} \theta_{k-1}. \]  

(35)

Equation (33) can be rewritten as

\[ P_{k-1}^{-1} = P_k^{-1} - \left( X_{k-1} X_{k-1}^\top \right). \]  

(36)

Substituting Eq. (36) into Eq. (35), we obtain

\[ \theta_k = \theta_{k-1} + P_k X_{k-1} \left( Y_{m,k} - X_{k-1}^\top \theta_{k-1} \right). \]  

(37)

In Eq. (37), the term \( P_k \) is unknown, thus we can apply the Lemma formula [34] to Eq. (36) to arrive at

\[ P_k = P_{k-1} - \frac{P_{k-1} X_{k-1} X_{k-1}^\top P_{k-1}}{1 + X_{k-1}^\top P_{k-1} X_k}. \]  

(38)

Finally, we substitute Eq. (38) into Eq. (37) to obtain

\[ \theta_k = \theta_{k-1} - \frac{P_{k-1} X_{k-1}}{1 + X_k^\top P_{k-1} X_k} \left( X_{k-1}^\top \theta_{k-1} - Y_{m,k} \right). \]  

(39)

Thus, the unknown parameters \( a_1, a_2, b_1 \) and \( b_2 \) have to be calculated in every time step as \( \theta_k = [-a_{1,k}, -a_{2,k}, b_{1,k}, b_{2,k}]^\top \) to update the estimated course angle \( \chi \) given in Eq. (26). The following calculation steps are required to obtain these parameters. First, the matrix \( P_{k-1} \) is initialized with large positive numbers on the leading diagonal and zeros on the off-diagonal elements. The matrix \( \theta_{k-1} \) must be populated with initial parameters close to the model. Then, the simulation results of the SI algorithm are obtained by:

1. \( X_k \) is updated every sample time by the system outputs and inputs as defined before.
2. Calculate \( \theta_k \) and \( P_k \) from Eqs. (39) and (38), respectively.
3. Update \( \theta_{k-1} \) and \( P_{k-1} \) with \( \theta_k \) and \( P_k \), respectively.
4. Repeat the loop for each time step.

4. Robust pole placement controller

The aim of this section is to design an adaptive control algorithm to stabilize the simplified kite model in Sect. 2. The control gains are updated with the SI algorithm described in the previous section, which makes the controller more robust compared to the classical control technique implemented in [12]. Moreover, the controller can be simply implemented on a micro-controller and installed in the KCU for autonomous operation of the kite. The closed-loop TF of the system in \( z \)-form is defined as

\[ TF(z^{-1}) = \frac{G(z^{-1}) G_c(z^{-1})}{1 + G(z^{-1}) G_c(z^{-1})}, \]  

(40)

where \( G(z^{-1}) \) is the open-loop TF of the system given by Eq. (22) and \( G_c(z^{-1}) \) is the controller TF, as defined by

\[ G_c(z^{-1}) = \frac{S(z^{-1})}{K(z^{-1})}. \]  

(41)
Substituting Eq. (22) in Eq. (40) we obtain

\[ TF(z^{-1}) = \frac{B(z^{-1})S(z^{-1})}{A(z^{-1})R(z^{-1}) + B(z^{-1})S(z^{-1})}. \]  

(42)

The next step is to calculate the controller functions \( S(z^{-1}) \) and \( R(z^{-1}) \) and their order [35]. We assume that these functions can be expressed as polynomials of order \( n \)

\[ R(z^{-1}) = 1 + r_1 z^{-1} + r_2 z^{-2} + ... + r_n z^{-n}, \]  

(43)

\[ S(z^{-1}) = s_1 + s_2 z^{-1} + ... + s_n z^{-n}. \]  

(44)

The orders \( n_s \) and \( n_r \) of \( S(z^{-1}) \) and \( R(z^{-1}) \) can be calculated from Eqs. (45) and (46). They are related to the orders \( n_a \) and \( n_b \) of the open-loop TF as follows

\[ n_s = n_b - 1, \]  

(45)

\[ n_r = n_a - 1. \]  

(46)

Using the SI algorithm discussed in Sect. 3, we can rewrite Eqs. (43) and (44) as

\[ R(z^{-1}) = 1 + r_1 z^{-1}, \]  

(47)

\[ S(z^{-1}) = s_1 + s_2 z^{-1}. \]  

(48)

Then, the characteristic equation of the closed-loop TF can be rewritten as

\[ A(z^{-1})R(z^{-1}) + B(z^{-1})S(z^{-1}) \]
\[ = \left(1 + a_1 z^{-1} + a_2 z^{-2}\right) \left(1 + r_1 z^{-1}\right) + \left(b_1 z^{-1} + b_2 z^{-2}\right) \left(s_1 + s_2 z^{-1}\right) = 0. \]  

(49)

Equation (49) is the characteristic equation of the closed-loop TF. By solving this equation, we will be able to tune the system behavior, i.e. time constant and steady-state error. The orders of the controller polynomials are calculated from the order of the open-loop TF. The required characteristics of our system is to place the poles of the closed-loop TF at certain positions so as to achieve stability and robustness of the system. We introduce the following equation

\[ A(z^{-1})R(z^{-1}) + B(z^{-1})S(z^{-1}) = A_m(z^{-1})A_o(z^{-1}), \]  

(50)

where \( A_m(z^{-1}) \) is a polynomial function that contains the controller characteristics and \( A_o(z^{-1}) \) is the polynomial function which is responsible for stabilizing the order of the equation. The controller parameters \( r_1, s_1 \) and \( s_2 \) can be determined by comparing the coefficients of the same order in Eq. (50).

In our design, the poles of the closed-loop TF in the \( z \)-form are 0.974653, 0.8431642 and 0.741046.

The sampling time used during the simulation was \( \Delta t = 0.02 \) s. Thus, the chosen poles can be rewritten as

\[ A_m(z^{-1})A_o(z^{-1}) = (1 - 2.558863 z^{-1} + 2.1688783 z^{-2} - 0.6089858 z^{-3}). \]  

(51)

The characteristic equation of our model is a third order polynomial. We applied Jury’s stability test [36], which is similar to the Routh-Hurwitz stability criterion used for continuous time systems, and found that all roots are located inside the unit circle, which is a condition for stability. Although Jury’s stability test can be applied to characteristic equations of any order, its complexity increases for higher-order systems.

Finally, we have three unknowns \( s_1, s_2 \) and \( r_1 \). By substituting Eq. (51) into Eq. (50), we obtain three equations that we can combine into a Sylvester matrix [37] as defined in Eq. (52)
The controller parameters $r_1, s_1$ and $s_2$ are dependent on the parameters of the SI algorithm. We will show in the following section, that the robust pole placement controller and SI algorithm increase the flight dynamic stability of the kite when exposed to sudden changes of the apparent wind speed.

5. Simulation results

In this section we present simulation results using the model and algorithms described in Sects. 2, 3 and 4. We compare the results of classical (PID) controller implemented in [12] with the adaptive control demonstrated in 4 to show the capability of the adaptive control to increase the stability of the kite flight, both controllers used the simplified kite model derived in Sect. 2. We investigate the flight dynamic responses of the kite for the two wind speed signals shown in Figs. 9 and 10. Flight condition I is discussed in Subsect. 5.1, while flight condition II, which is characterized by a much higher frequency, is discussed in Subsect. 5.2.

5.1. Flight condition I

Flight condition I uses the wind speed illustrated in Fig. 9. The fluctuation of the wind speed affects the flight dynamics of the kite as described by the model presented in Sect. 2. The SI algorithm
derived in Sect. 3 generates the values of $a_1$, $a_2$, $b_1$ and $b_2$ shown in Figs. 11 and 12. The resulting figure-of-eight trajectory is illustrated in Fig. 13, calculated using the simple model and the classical controller presented in Sect. 2. The figure-of-eight trajectory illustrated in Fig. 14 is calculated based on the SI algorithm and adaptive controller described in Sects. 3 and 4. The controller parameters are updated in real time, accounting for the varying parameters $r_1$, $s_1$ and $s_2$, as shown in Figs. 15 and 16, which in turn are updated from the varying SI parameters $a_1$, $a_2$, $b_1$ and $b_2$. Figure 17 shows a
Figure 14. Trajectory computed on the basis of the SI algorithm and adaptive controller for a flight time of 70 s.

Figure 15. Time history of the controller parameter $r_1$ for flight condition I.

Figure 16. Time history of the controller parameters $s_1$ and $s_2$ for flight condition I.

very close fit between the measured course angle from the model and the estimated value from the SI algorithm. This accuracy was achieved for a sampling time of $\Delta t = 0.02$ s, which is very short for this type of application.

To assess the tracking performance of the different control approaches we use the deviation between the computed and the planned flight paths. From the several options to quantify this deviation we chose the elevation angle in this study. The difference in elevation angle is a suitable measure to quantify the operational stability of the kite because if this difference increases too much
the wing can experience aerodynamic stall. Therefore, the designed control parameters have to be chosen carefully to keep the deviation of the elevation angle below a certain limit.

Figure 13 indicates that the deviation between simulated and desired paths depends on the position along the path. We can notice that the error increases after performing the turning maneuver. The maximum deviation for the total simulation time is $\Delta \beta = 5^\circ$ (20%). Using the SI algorithm together with the adaptive controller the maximum deviation is $\Delta \beta = 2.5^\circ$ (12.5%), as depicted in Fig. 14. This was acceptable for maintaining stable flight.

5.2. Flight condition II

Flight condition II uses the wind speed illustrated in Fig. 10, which is characterized by fluctuations at much higher frequency compared to flight condition I. For this reason flight condition II is much more demanding for the controller, which we can see from the values of the coefficients $a_1$, $a_2$, $b_1$ and $b_2$ displayed in Figs. 18 and 19. The order of the coefficients $b_1$ and $b_2$ in Figs. 12 and 19 is different, because the different frequencies of the wind speed fluctuations affect the flight dynamic model of the kite, which is then detected by the SI algorithm.

The computed trajectories are illustrated in Figs. 20 and 21. Figure 20 shows that the figure-of-eight motion of the kite is progressively dropping towards lower elevation angles, while in Fig. 21 the figure-of-eight motion stays in the vicinity of the set value $\beta_{set} = 24^\circ$. From this we conclude that the classical flight controller is not capable to maintain a stable flight operation for flight condition II, while the combination of SI algorithm and adaptive controller is capable.
Figure 19. Time history of the SI parameters $b_1$ and $b_2$.

Figure 20. Trajectory computed on the basis of the classical flight controller for a flight time of 70 s.

Figure 21. Trajectory computed on the basis of the SI algorithm and adaptive controller for a flight time of 70 s.

To explain why flight condition II leads to an unstable flight operation we look at the differences of the two control approaches used in the simulation. The classical flight controller is based on a PID controller with constant gains. While this is suitable for flight condition I, however, it cannot cope with the dynamic reaction of the model to the more rapidly fluctuating wind speed of flight condition II. In contrast to the classical controller, the combination of SI algorithm and adaptive controller can manage this dynamic reaction because the SI parameters $a_1$, $a_2$, $b_1$ and $b_2$ and controller parameters...
$r_1$, $s_1$ and $s_2$ are updated in real time, as shown in Figs. 18, 19, 22 and 23, respectively. Figure 24 shows again a very close fit between the measured and estimated course angle, which demonstrates the performance of the SI algorithm for a strongly fluctuating wind speed.
On the other hand, Fig. 21 shows a maximum deviation of $2.5\degree$ (12.5%), which is almost the same as for flight condition I (see Fig. 14).

6. Experimental results

In this section we apply the SI algorithm to data that was recorded during a flight test of the 20 kW kite power system of TU Delft. The objective is to derive a mathematical model of the kite power system directly from measurement data, omitting the use of an underlying system model with many simplifying assumptions. As a result, the effects of the fluctuating wind velocity (magnitude and direction) and deforming wing due to a varying aerodynamic load distribution and actuation of the bridle line system are implicitly considered. In Subsect. 6.1 we describe the configuration of the kite power system during the selected flight test. In Subsect. 6.2 we use the SI algorithm to derive a mathematical model of the kite power system.

6.1. System configuration

The flight test was performed by the TU Delft research group on 25 October 2012 [38], using the V3 kite that is illustrated in Fig. 3 and in more detail in Fig. 25. This specific kite has a total wing surface area of $25\text{ m}^2$ and is a customized and scaled up derivative of the Hydra kite, which is a commercially available surf kite with a total wing surface area of $14\text{ m}^2$. The TU Delft V3 kite consists of a flexible membrane wing, a bridle line system and a small remote-controlled cable robot, the KCU. The wing is designed as a leading edge inflatable (LEI) tube kite, using an inflated tubular frame to collect the distributed aerodynamic load acting on the canopy and transmit this load to the bridle lines. The front bridle lines directly attach to the tether, transmitting the major part of the forces, while the KCU

Figure 25. TU Delft V3 kite in front view (left) and side view (right) [38]. The KCU is displayed without the exterior foam shell and without the attached small wind turbine for supplying onboard power.
connects the two branches of the rear bridle lines to the tether. The integrated steering and de-power
winches can adjust the lengths of the steering and de-power tapes to steer the wing and to adapt its
angle-of-attack, respectively. The angle-of-attack is decreased during the reel-in phase to minimize the
energy required to retract the kite. There are two software algorithms to control the system: the first
algorithm is for maintaining figure-of-eight maneuvers during the reel-out phase while the second
algorithm is for the reel-in phase. A detailed description of the different functional components of the
kite power system is given in [5,9,39].

6.2. System identification

The results presented in this section are based on two consecutive pumping cycles that started
2615 s after launch of the kite at 15:13:41 (hh:mm:ss) [38]. Each pumping cycle consists of 110 s of
tether reel-out followed by 70 s of tether reel-in. The flight motion of the kite is affected by a variety
of parameters, such as the tether force, the reeling speed, the steering actuation of the KCU and the
dynamics of the drum-generator module on the ground. Therefore, the SI algorithm described in Sect. 3
is used to determine the SI parameters of the kite system directly using experimental measurements.
We can determine the course angle from the recorded flight data, using the attitude sensors and the
relative steering action $u_s$. These data are sufficient for the SI algorithm to derive in real time the SI
parameters $a_1, a_2, b_1$ and $b_2$.

The recorded flight path of the kite for the two consecutive pumping cycles is illustrated in Figs. 26
and 27. The kite starts at an altitude of 240 m and subsequently dives down to a minimum altitude
of 115 m to start a first sequence of figure-of-eight maneuvers around an average elevation angle
of 25°. During these maneuvers the azimuth angle varies between $-20°$ and 20°, the tether reels
out and the altitude progressively increases. After around 100 s, the figure-of-eight maneuvers are
discontinued and the tether is reeled in. In this phase, the kite passes through a maximum azimuth
angle of 64°, a maximum elevation angle of 74° and is climbing to a maximum altitude of 315 m before
again diving down to around 115 m to start a second sequence of figure-of-eight maneuvers. Flying to
large azimuth and elevation angles is a second technique to de-power the kite and was used here in
addition to reducing the angle-of-attack of the wing. Towards the end of the second reel-in phase the
kite reaches a maximum elevation angle of almost 60° at a constant azimuth angle of $-30°$, climbing
to a maximum altitude of 365 m.

Figure 26. Recorded kite altitude for two pumping cycles. The origin of the time scale in this and
subsequent time history diagrams is not synchronized with the launch event.

The recorded ground wind speed during the two considered pumping cycles is shown in Fig. 28,
the recorded relative steering action $u_s$ in Fig. 29 and the recorded measured course angle in Fig. 30.
From these we calculate the SI parameters $a_1, a_2, b_1$ and $b_2$ displayed in Figs. 31 and 32. The time history
diagrams reveal strong variations of the SI parameters at times 1990, 2110 and 2280 s. These variations
Figure 27. Recorded azimuth and elevation angles for two pumping cycles.

The recorded angles coincide with the transitions between reel-in and reel-out phases and demonstrate the capability of the SI algorithm to adjust to the system dynamics even when rapidly changing operational modes.

Figure 28. Recorded ground wind speed for two pumping cycles.

Figure 29. Recorded relative steering action $u_s$ for two pumping cycles.

The open loop TF of the V3 kite is obtained from Eq. (25) using the calculated values of the SI parameters displayed in Figs. 31 and 32. The resulting correlation between the relative steering action $u_s$ and the course angle $\chi$ of the kite can be used for planning and control of autonomous flight operation. The correlation is also used in Fig. 30 to estimate the course angle using the recorded
Figure 30. Measured and estimated course angle for two pumping cycles.

Figure 31. Calculated SI parameters $a_1$ and $a_2$ for two pumping cycles.

Figure 32. Calculated SI parameters $b_1$ and $b_2$ for two pumping cycles.

relative steering action $u_s$ displayed in Fig. 29. The close fit between measured and estimated course angle indicates the capability of the chosen SI algorithm to identify the system parameters without any singularity at a very short sampling time.

The recorded tether force measured at the ground is shown in Fig. 33. One can clearly distinguish the reel-out phases with an average tether force of 3000 N and the reel-in phases with and average tether force of 700 N. The strong oscillations during the reel-out phase are induced by the figure-of-eight motion, because the tether force is proportional to $\cos \beta \cos \phi$ [40], and the fluctuations of the wind speed at the position of the kite [13].
7. Conclusions

In this paper, we have studied the flight control of a tethered flexible membrane kite used for airborne wind energy harvesting in pumping cycles. Specifically, we investigated the figure-of-eight maneuvers of the kite during the energy-generating reel-out phase. Following the development of a simplified kite system model and a flight path planning algorithm, we have compared a classical PID controller using fixed gains with an adaptive controller that uses a system identification algorithm to adjust the controller parameters in real time. The performance of the two different control approaches was assessed on the basis of two flight conditions that are characterized by different fluctuation frequencies of the wind speed. We found that the classical control was not able to cope with the rapidly fluctuating wind speed. On the other hand, the combination of adaptive control and SI algorithm is more robust and can handle a more severely fluctuating wind speed and varying flight dynamic behavior of the kite. The enhanced stability is a result of the real-time tuning of the control gains at every integration time step to the varying SI parameters. In a second part of the study, the SI algorithm was successfully applied to recorded measurement data of a test flight of a 20 kW kite power system, equipped with a kite of 25 m² wing surface area. Despite the uncertainty of the wind velocity in magnitude and direction and the dynamic response of the deformable membrane wing, it was possible to successfully derive the SI parameters of the system for different operational phases, such as reel-in and reel-out. The results suggest that the combination of adaptive controller and SI algorithm is well suited for robust path control of a tethered membrane kite flying in a fluctuating wind field and transitioning through different operational phases. As a next step for this research we aim at the implementation of the adaptive controller with the experimental hardware to demonstrate its performance in a flight test campaign.

Acknowledgments

Roland Schmehl was supported by the H2020-ITN project AWESCO funded by the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 642682

References


34. Ibrahim, D. Microcontroller based applied digital control; Wiley Online Library: New Delhi, India, 2006; chapter 8.