Controlling rigid-wing airborne wind energy systems during circular flight without exact path following

Duc H. Nguyen[1], Agustí Porta Ko[2,3], Tallak Tveide[2], Mark H. Lowenberg[1], and Espen Oland[2]

[1]School of Civil, Aerospace, and Design Engineering, University of Bristol, Bristol, BS8 1TR, United Kingdom
[2]Kitemill AS, Voss, 5704, Norway
[3]Technical University of Denmark, Frederiksborgvej 399, 4000 Roskilde, Denmark

Correspondence to: Duc H. Nguyen (duc.nguyen@bristol.ac.uk)

Abstract. We propose a feedback architecture that enables effective flight control of rigid-wing airborne wind energy systems during circular-pattern reel out. The controller performs well with only proportional-integral regulators, thereby presenting a significantly simpler solution compared to the existing literature. The key idea is in tracking the roll angle on a non-static reference frame, effectively reducing the control problem to one degree of freedom. This method of navigation does not require users to define an exact path for the kite to follow, which contributes to stability and robustness. In its minimum viable form, the controller can function with only ailerons while requiring no pitot-tube measurement, although the addition of elevators and rudder enables angle-of-attack and zero-sideslip tracking for more efficient power generation. Simulation-based verification is conducted on an industrial, 6-degree-of-freedom model with a flexible tether, nonlinear aerodynamics, and realistic wind conditions, showing satisfactory performance. Three expansions to the control law are then presented. The first one reduces angle of attack fluctuation during reel-out by adding a proportional pitch angle feedback term to the elevator, resulting in more power. In the second expansion, the reel-out radius is automatically adjusted to enable phase synchronisation of multiple kites in a farm configuration, where minimum separation rules may apply. The third expansion implements a proportional feedback rule that enables figure-of-eight flight. By using proportional-integral architecture, the controller is easy to implement, making it a suitable baseline system for benchmarking more advanced control laws. Appendix C provides the Julia code for the circular flight controller.

1 Introduction

Airborne wind energy systems (AWES) are an emerging technology for electricity generation using tethered flying devices. Compared to traditional wind turbines, AWES are easy to transport and use less material over their lifetime. This means they can provide clean energy in areas where the construction of wind turbines is not feasible. Since its first formal conceptualisation by Loyd, 1980, AWES development has grown significantly in the past two decades, with research groups and startups proposing increasingly innovative designs. For an overview of the technology and its progression over the past decade, readers can refer to three systematic reviews published between 2015 and 2025 (Cherubini et al., 2015; Pereira and

Sousa, 2023; Khurshid et al., 2025). Also worth mentioning are two books by Ahrens et al., 2013; Schmehl, 2018 that summarise key technical concepts.

Because AWES lack a solid structural foundation seen in conventional wind turbines, the number of degrees of freedom expands from 1 to 6. This fact places great emphasis on flight control for making AWES safe and efficient (Vermillion et al., 2021). Compared to free-flying aircraft, AWES differ by the fact that their natural motions during circular flights are limit cycles (self-sustained oscillations with fixed amplitude and frequency). Equilibrium responses during circular flights exist only when gravity is ignored and the circular trajectory is perpendicular to the wind (Trevisi et al., 2021; Trevisi, 2024; Nguyen et al., 2026b), although this trajectory is unachievable as half of the orbit has negative height (Nguyen et al., 2026b). Consequently, any realistic trajectory requires a feedback strategy that can handle self-oscillating dynamics, and the resulting control surface movements should also be cyclic. Closed-loop control in the presence of limit cycles is uncommon for fixed-wing airframes, although it is not unheard of. The X-15's flight control system uses adaptive gains that automatically increase until a small-amplitude limit cycle is achieved, which helps the aircraft achieve the highest gain possible without causing instability (NASA, 1971).

An overview of many AWES controllers by Vermillion et al., 2021 shows that their algorithms are sophisticated. In the context of this paper, we can classify reel-out controllers into two categories: optimisation-based path planning and direct path planning. The former involves utilising an optimisation search (Cobb et al., 2020; Heydarnia et al., 2025) or reinforcement learning agent (Orzan et al., 2023; Basile et al., 2025) to find the best flight path that generates the most energy. The found path is a useful reference for quantifying the maximum power generation capability. However, deploying a pure optimisation-based controller on a real system introduces implementation complexity, especially due to many complex physical phenomena that are difficult or costly to model. Direct path planning involves designing a flight controller for a tethered aerial vehicle (rigid-wing aircraft or kites) to follow a predefined path in 3D space. Past works have covered waypoint-based guidance using L0- and L1-based control (Fernandes et al., 2022), defining the circular flight path as combinations of motion primitives (fundamental circles) (Vinha et al., 2025), and linking the target path's geometry to the steering control law (Fagiano et al., 2014; Dief et al., 2020; Delosríos-Navarrete et al., 2025). These algorithms show great promise, and some have been verified experimentally (Fagiano et al., 2014; Dief et al., 2020; Delosríos-Navarrete et al., 2025). However, they may require more effort to design and implement than classical controllers do. Waypoint- and geometry-based guidance also split the control task into discrete steps, thereby creating a discontinuous controller that can complicate classical analysis. Lastly, cascaded nonlinear dynamic inversion control can provide continuous guidance with good tracking performance (Rapp et al., 2019), although successful deployment relies on accurate system modelling.

Accordingly, a reel-out flight control law resembling classical designs with continuous guidance provides a familiar working space for practitioners, which can be useful during the early simulator development phase, where the model may undergo multiple revisions and requires repeated retuning. This paper proposes a solution for rigid-wing AWES – those that resemble fixed-wing aircraft with conventional control surfaces. Circular reel out is achieved using an indirect path planning method, which ensures the flight path has the correct orientation and dimension but does not specify the exact path for the kite to follow. Instead, circular flight emerges naturally as an inherent property of the controller. The resulting control surface movements are periodic, which are required for tethered circular flight (Nguyen et al., 2026b). This indirect path-planning process contains elements resembling virtual holonomic constraints (VHC). First proposed by Shiriaev et al., 2005, a VHC uses feedback to enforce an algebraic relation that confines the closed-loop motion to a manifold carrying a periodic orbit. The method has seen applications in mechanical systems and robotics (Freidovich et al., 2008; Rezapour et al., 2014). In relation to AWES reel out, the closest VHC work known to the authors addresses vertical-plane flight along smooth curves (Consolini et al., 2010) and stabilisation of a circular orbit with a desired speed profile (Mohammadi et al., 2018) – both based on the simplified vertical-takeoff-and-landing aircraft model by Hauser et al., 1992. Our results show satisfactory closed-loop response on a 6-degree-of-freedom system with nonlinear aerodynamics. Since the control law presented in this paper was derived mostly from intuition, readers may find the VHC framework useful for rigorous stability analysis in a future work. We also refer readers to our trim analysis in Nguyen et al., 2026b, which discusses the flight dynamics of tethered fixed-wing airframes and lays the foundation for the flight control law presented. In the latter parts of the paper, we propose two extensions that reduces angle-of-attack oscillations and enable multi-kite synchronisation – both of which address key challenges in AWES operations.

Lastly, the feedback architecture follows a cascaded control structure intended for use during reel out only. Flight control in other phases is not considered in this work and will require entirely different control architectures (e.g., see the following references for launching (Fagiano et al., 2018; Vinha et al., 2024; Duda et al., 2025), reel in (Zgraggen et al., 2016; Berra and Fagiano, 2021), and recovery (Vinha et al., 2024; Duda et al., 2025)). Our control gains are tuned empirically but follow the time scale separation principle, i.e., the gains in the outer layers should be an order of magnitude smaller than the inner layer gains. Higher performance can be obtained by employing classical methods for rigorous tuning, although the present system with empirically derived gains can already achieve all control objectives. Since the control algorithm is flight-path agnostic, both circular and figure of eight trajectories are possible. Flight control with a fixed tether length is also possible, which enables deployment in ship-towing applications (Fritz, 2013).

2 Simulation environment

This study uses a 6-degree-of-freedom simulator of Kitemill's KM1 prototype – a 20 kW groundgen system driven by a fixed-wing unmanned aerial vehicle with mass 54 kg and wingspan 7.4 m (see Fig. 1). The aerial vehicle is referred to as a kite in this paper based on conventions adopted by the AWES community, which highlights the fact that the air vehicle extracts energy from the wind to remain airborne. Traditional trailing-edge control surfaces (aileron, elevator, and rudder) are used for flight control, in addition to flaps for increased lift during power production. The simulation was constructed in MATLAB & Simulink and, more recently, extended into Julia. Past studies that used the Simulink version can be found in references (Mohammed et al., 2024b; Mohammed et al., 2024a; Rapp et al., 2019). In this paper, results with constant wind were generated in MATLAB & Simulink, while non-uniform wind analyses (section 4.2) were conducted in Julia but plotted in MATLAB for style consistency. Aerodynamic data of the KM1 airframe was obtained from computational fluid dynamics, which are dependent on angle of attack, sideslip angle, and control surface deflections. The total force and moment coefficients are calculated in a manner similar to Eqs. (3) and (5) in Rapp et al., 2019, which includes aerodynamic damping terms as functions of angular rates and airspeed. For illustrative purposes, a few key static aerodynamic relationships are shown in Fig. 2 to indicate the nonlinear dependencies of the force and moment coefficients with respect to the angles of incidence and sideslip.

Figure 1. Aerial component of Kitemill's KM1 system.

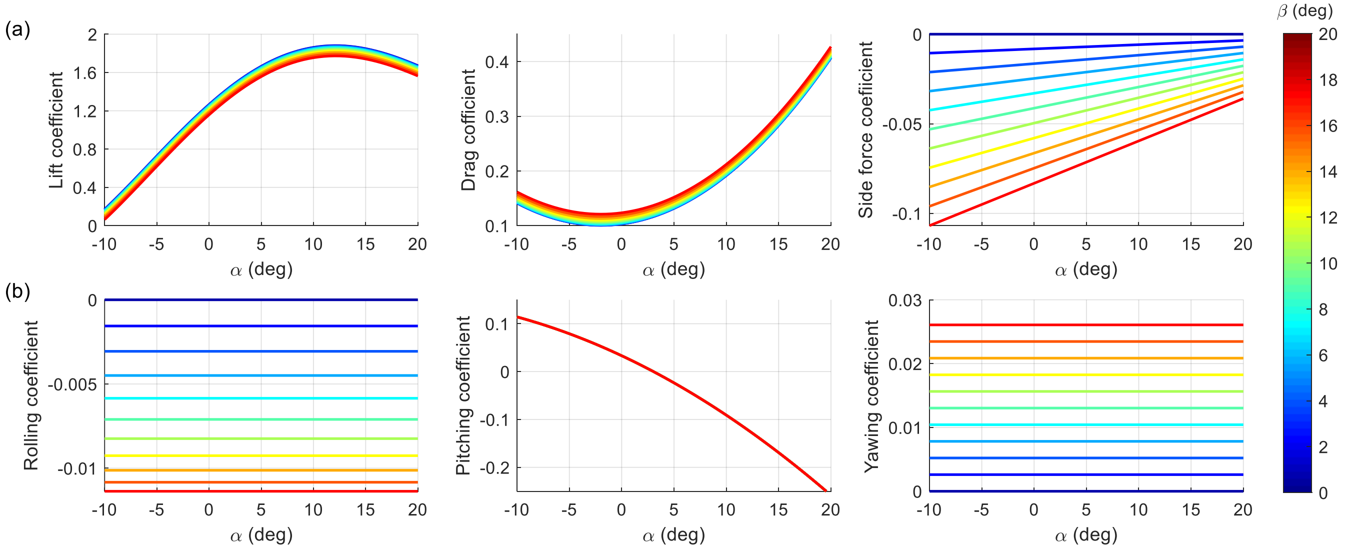

Figure 2. Static aerodynamic force (b) and moment (b) coefficients at full flaps and no control surface deflection.

In its open loop configuration, the simulator has 108 states, comprising 12 states to describe the aircraft's motion in 3D space, 4 actuator states modelled as first-order lags with 50°/s rate limit, 2 states to describe the ground winch, and $6N$ states to model the flexible tether, where N is the number of tether nodes between the winch and the kite. The N nodes discretised the tether into N+1 segments, which are connected by second-order mass-spring-damper elements. If the tether's natural length is fixed, the winch states can be discarded, and the model reduces to 102 states. A detailed description of the equations of motion can be found in Rapp et al., 2019. Table 1 summarises the open-loop states, control surface inputs, and outputs.

Table 1. Key elements of the open-loop model.

Aircraft states	Body-axis velocities: u, v, w Body-axis angular rates: p, q, r Earth-axis Euler angles: ϕ, θ, ψ Earth-axis coordinates: X, Y, Z
Winch states	Tether natural length: L Winch linear speed: $\dot{L}$
Tether states	Each node N has 6 states: its 3D Earth-axis coordinates and velocities. All results use N = 15 nodes.
Control surface actuator states	Ailerons: δ_a Elevator: δ_e Rudder: δ_r Flaps: fixed at the full-down (13°) position during reel out All modelled as first-order lags with 1/50 sec time constant and 50 °/s rate limit.
Outputs	Angle of attack: α Sideslip angle: β Airspeed at the kite: V Instantaneous mechanical power: P Energy generated: E

The tether's natural length L (excluding elastic extension) is regulated by an ideal ground winch, which is modelled as a cylinder of mass m_w = 25 kg and has a constant radius of R_w = 0.25 m. The winch control system applies a winch force F_w that opposes the tether tension T at the winch. This gives a winch linear acceleration of

$$\ddot{L} = (T - F_w)\frac{R_w^2}{I_w} \tag{1}$$

where $I_w = 0.5 m_w R_w^2$ is the winch moment of inertia. The winch speed (reel-out speed) $\dot{L}$ and natural length L are found by integrating eq. (1) once and twice, respectively. Feedback control of the winch is provided by the term F_w, which is the output of a high-gain PID controller that tracks either a desired winch speed (keeping $\dot{L}$ close to constant) or a desired tether tension (keeping T close to constant). Apart from Fig. 30, all results in this paper with the winch active use the constant winch speed mode.

Provided that the winch is reeling out with positive tether tension, the instantaneous mechanical power P is calculated as

$$P = T\dot{L} \tag{2}$$

and assuming no losses in the winch, Eq. (2) can be integrated to give the total energy generated up to time t

$$E = \int_{t_{L=350}}^{t} P\, dt \tag{3}$$

where $t_{L=350}$ is the simulation time at the start of reel out, defined to be when the tether's natural length is 350 m. Reel out ends when L reaches 700 m.

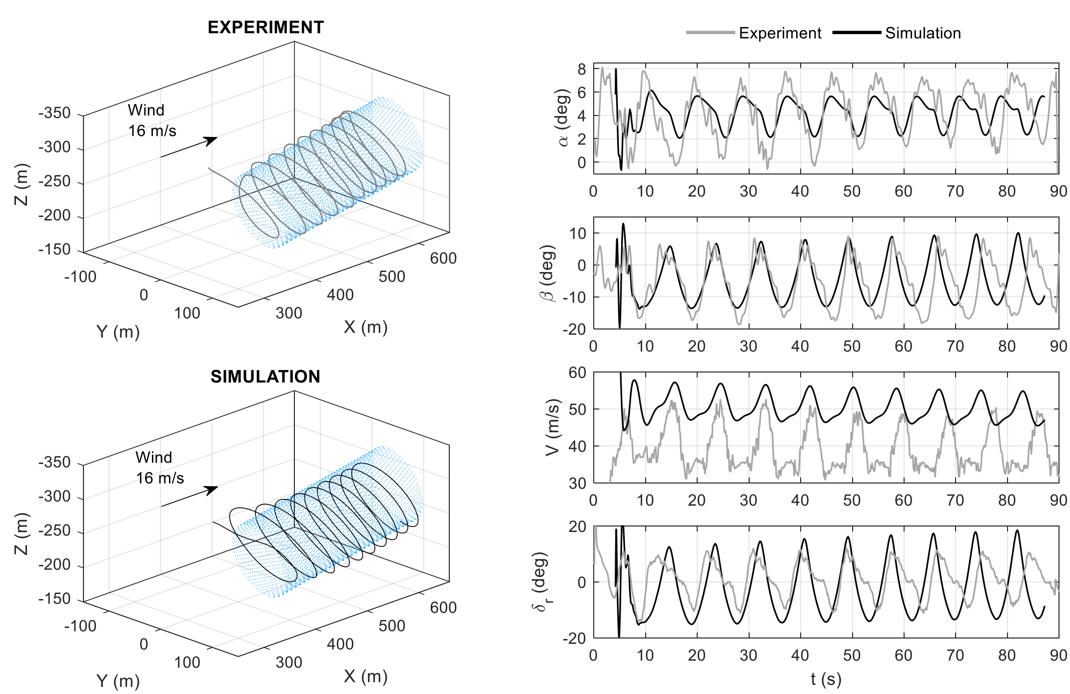

Figure 3. Experiment and simulated flight data (first presented in (Nguyen et al., 2026a)).

To verify the simulator's accuracy, we replicated a test flight in 16 m/s wind. This test used an early flight control law presented in Mohammed et al., 2024a, which has only one proportional feedback loop on the rudder to generate yaw-induced roll. Elevator and aileron were fixed at –4º and 0º, respectively. Figure 3 shows that there is strong agreement between the recorded and simulated responses, apart from an offset in airspeed V due to a known sensor calibration fault that also affected angle of attack measurements. However, the shapes of the simulated and recorded airspeed are comparable, and other parameters show strong correlations. These results suggests that the simulator possesses sufficient accuracy for control law developments.

3 Baseline controller design

3.1 Key concept: tilting the reference plane

To implement the control law, three coordinate systems are required: Earth, reference, and production. The first two are needed for the current section and are now described. The Earth plane is defined by the three axes X, Y, and Z originating from the winch and pointing in the north, east, and down directions, respectively (see Figure 4a). Three Euler angles describe the kite's body-axis orientation with respect to the Earth plane: ϕ (roll), θ (pitch), and ψ (yaw). Unless otherwise noted, the wind is assumed to be uniform and travels from south to north at a speed of 14 m/s.

The reference plane is a 'guidance plane' that provides a second set of Euler angles for the roll controller. Three axes X_R, Y_R, and Z_R define the reference plane. They also originate from the winch but have different orientations with respect to the Earth axis. To construct the reference plane, start with the X_R, Y_R, and Z_R aligned with the Earth's axes X, Y, and Z, then apply the following right-hand rotations:

- Rotate Z_R by an angle ζ_R (zeta for aZimuth).
- Rotate Y_R axis by an angle λ_R + 90º (lambda for eLevation).
- Rotate X_R by 180º.

These three rotations result in the reference plane having an orientation as shown in Fig. 4b – depicting λ_R and ζ_R having small positive values. Note that when $\lambda_R = \zeta_R = 0$º, Z_R points in the negative X direction, and X_R points in the negative Z direction. From here, we obtain another set of Euler angles $[\phi_R, \theta_R, \psi_R]$ that depict the kite's orientation with respect to the reference plane. To aid understanding, Fig. 5 depicts a special case of $\lambda_R = \zeta_R = 0$º with the kite in a south-bound, wing-level, climbing attitude. The corresponding Euler angles on the Earth's and reference planes are shown. Subsequent analysis does not require the use of the kite's yaw angle ψ_R. To simplify illustrations, the reference plane will subsequently be illustrated by a single dashed line originating from the winch and pointing in negative Z_R direction, as shown in Figs. 4b and 5b. Appendix A provides the mathematical operations for converting Euler angles from Earth to reference-plane.

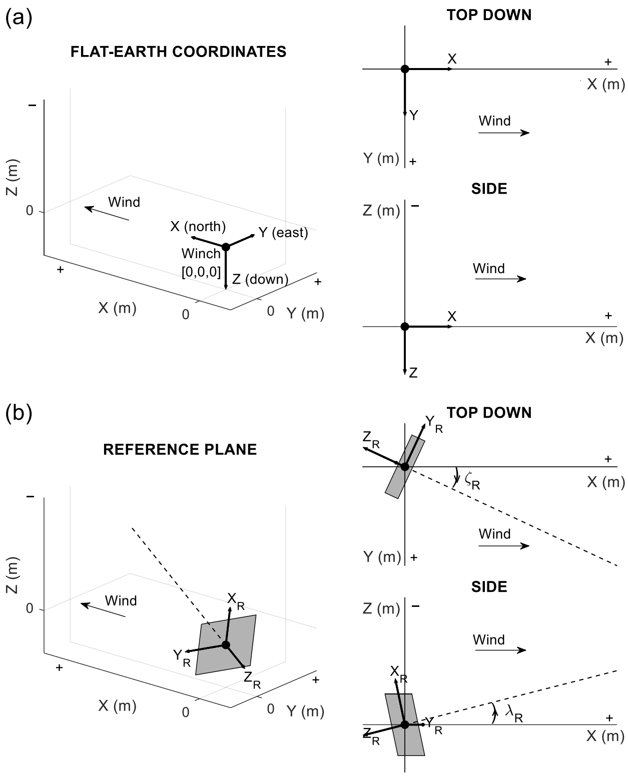

Figure 4. Two coordinate systems: Earth (a) and reference plane (b).

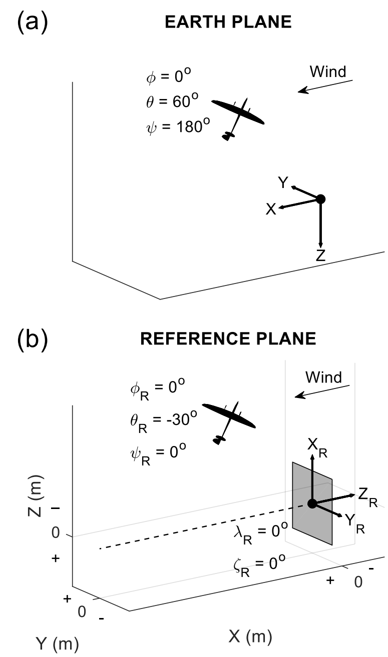

Figure 5. Illustrations of Euler angles on the Earth plane (a) and reference plane with λ_R = ζ_R = 0° (b).

We now discuss the flight controller. The principal idea is to use ailerons to track the roll angle ϕ_R on the reference plane, not ϕ on the Earth plane. Assuming that gravity and wind are not present, the kite has its own propulsion, and the tether length is fixed, then the trajectory will converge to a circular orbit on a plane parallel to the reference plane as shown in Fig. 6. This convergence is facilitated by two elements: the fixed-length tether preventing the kite from 'climbing' away from the reference plane and converge to a helical trajectory, and the roll controller keeping ϕ_R and hence radius constant. From this arrangement, we can shift the centre of the circle to a desired location by adjusting the reference plane orientation via λ_R and ζ_R. Equilibrium trim exists in this hypothetical scenario. However, AWES derive their propulsive force from the wind and operate in the presence of gravity, both of which act as external periodic disturbances that require a cyclic control input to compensate (Nguyen et al., 2026b). This cyclic input can be obtained by configuring the controller to track a fixed roll angle set point $\phi_{R_{SP}}$ and accepting the subsequent oscillations caused by gravity and the wind.

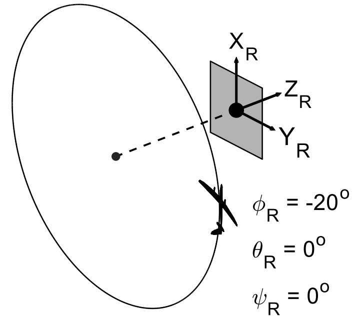

Figure 6. Example left-roll circular orbit as defined by the reference plane.

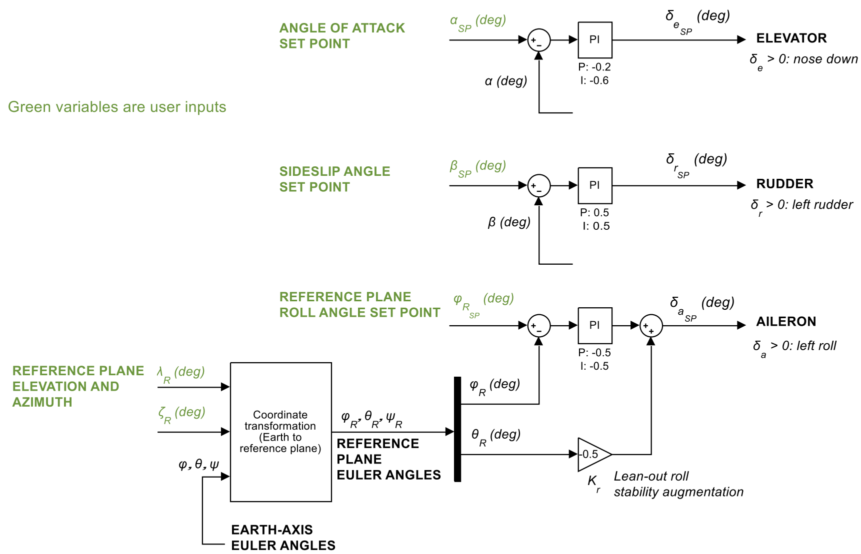

Figure 7. Basic flight control system (subsequently referred to as the inner loop). The Earth-to-reference-plane coordinate transformation is presented in appendix A.

Figure 7 presents the proposed flight control system. For convenience, the following shorthand notation for PI control is used throughout the paper

$$PI(x_{SP}, x) = K_P(x_{SP} - x) + K_I \int (x_{SP} - x)\, dt \tag{4}$$

where x_{SP} and x are the set point (SP) and feedback signal, and K_P and K_I are the proportional and integral gains. Three proportional-integral (PI) loops are used. The pitch and yaw channels channel adjust the elevator and rudder to track the desired angle of attack and sideslip set points. For the ailerons, their role is to track a demanded roll angle $\phi_{R_{SP}}$ on the reference plane. This roll control method requires knowledge of the reference plane's orientation λ_R and ζ_R. In this section, we treat these two angles as user inputs, and section 3.2 will explain how λ_R and ζ_R can be controlled automatically by the outer loop. Accordingly, the demanded control surface deflections are governed by the following feedback law

$$\delta_{e_{SP}} = PI(\alpha_{SP}, \alpha) \tag{5}$$

$$\delta_{r_{SP}} = PI(\beta_{SP}, \beta) \tag{6}$$

$$\delta_{a_{SP}} = PI\left(\phi_{R_{SP}}, \phi_R\right) + K_r\theta_R \tag{7}$$

The second term in Eq. (7) multiplies the reference-plane pitch angle by a proportional gain K_r. This pitch-roll cross-feeding signal counteracts the roll instability encountered in large-radius tethered circular flight, which was first reported in one of our earlier studies (see the discussion surrounding Eq. (1) in (Nguyen et al., 2026a)). The sign of K_r reflects flight direction. For trajectories where the circle's centre is on the port (left) wing's side, K_r should be negative. By conventional flight control standards, the PI-only feedback scheme in Figure 7 can be stable but may not provide adequate handling qualities – mainly due to the clear separation between fast and slow dynamics in the pitch and yaw responses. Higher performance is usually achieved by adding an inner stability augmentation loop that feeds back both fast (body-axis angular rates) and slow (body-axis angles) variables. Those performance-enhancement elements are deliberately left out in this work to demonstrate the robustness of our flight control scheme, which can handle sub-optimal handling qualities caused by rudimentary inner-loop designs.

We now examine a closed-loop response with the tether length fixed (i.e., $\dot{L} = \ddot{L} = 0$) at 350 m – a typical length at the start of reel out for the KM1 AWES. The reference plane orientation is [λ_R, ζ_R] = [0°, 0°], and a representative combination of set points [α_{SP}, β_{SP}, $\phi_{RP_{SP}}$] = [6°, 0°, –15°] is chosen. Negative $\phi_{RP_{SP}}$ indicates the circle centre is on the port side of the wing, which is assumed throughout this paper without loss of generality. In this arrangement, the kite follows a circular orbit as shown in Fig. 8a. Inspecting the time history of this trajectory (Fig. 8b) reveals that the controller has entered a small limit cycle with periodic forcing inputs. This behaviour is expected, as our previous work in Nguyen et al., 2026b has shown that tracking a set of specified α, β, and ϕ_{RP} using fixed control surfaces (i.e., an equilibrium response) is only possible when gravity is ignored. The resulting gravity-free orbit is normal to the wind and has its circle centre on the ground, which overlaps the winch when viewed from the wind-facing direction. When gravity is included as shown in Fig. 8a, we see the circle centre shifting downward due to weight and a slight sideways due to lift asymmetry (from the wind-facing view: the kite climbs at the 3 o'clock position and descends at the 9 o'clock point, so the latter generates more lift, pushing the orbit centre to the side). Gravity also changes direction periodically with respect to the kite's body axis throughout the cycle. All of the above factors mean that the desired closed-loop trajectory is a limit cycle, so control surface movements should be cyclic to counteract the external periodic disturbance. Nevertheless, all three controlled variables α, β, and ϕ_{RP} oscillate with small amplitudes around the desired values, and the cyclic control surface movements are also small.

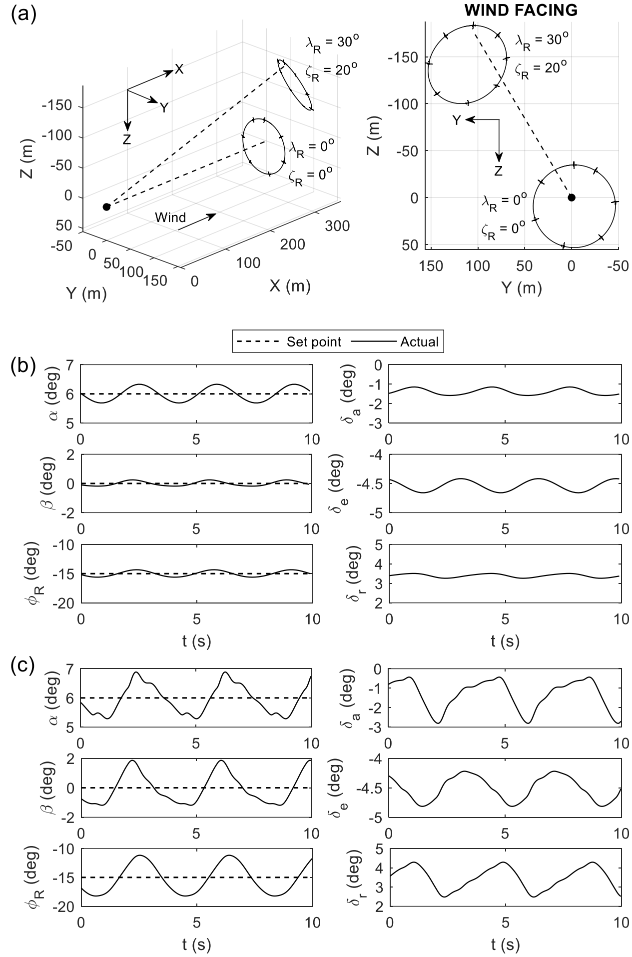

Figure 8. Closed-loop trajectories (a) and time histories (b and c) with two different reference plane orientations. The time histories of [λ_R, ζ_R] = [0º, 0º] and [λ_R, ζ_R] = [30º, 20º] are shown in (b) and (c), respectively.

The reference plane orientation is now changed to non-trivial values of [λ_R, ζ_R] = [30º, 20º]. Figure 8a shows the resulting closed-loop response. The kite still follows a circular path, which is now at a slight angle to the wind due to the new reference plane's orientation. Larger magnitudes in λ_R and ζ_R result in more longitudinal and lateral shifting, which become less effective as the circular orbit moves further away from its ideal location of being normal to the wind (achieved only when gravity is ignored and [λ_R, ζ_R] = [0º, 0º]). A consequence of shifting further away from the ideal location is higher limit cycle amplitudes, as seen in Fig. 8c vs Fig. 8b. This behaviour is congruent with our previous observations, which noted that circular flight further away from the ideal location requires larger cyclic control inputs (Nguyen et al., 2026b). Nevertheless, as long as there is enough control power and sufficient wind to prevent airspeed from dropping to the stall point, the current flight control scheme will keep the kite in a stable circular orbit.

Results in Fig. 8 suggests that the orbit centre can be dynamically shifted by adjusting the reference plane's orientation. To demonstrate this point, Fig. 9 shows the kite's response to two simultaneous ramp inputs in λ_R and ζ_R. Both ramps last 15 seconds, starting from $[\lambda_R, \zeta_R]$ = [0º, 0º] and ending at $[\lambda_R, \zeta_R]$ = [30º, 20º]. The resulting trajectory shows that the kite achieves a gradual transition between the two conditions while maintaining circular flight throughout the manoeuvre. This method of transitioning between different operating points is more desirable than taking a straight-line path to the destination, which risks running out of airspeed mid-flight due to the lack of a dedicated propulsion unit. The fact that the kite maintains circular flight by itself also simplifies control design significantly, as there is no need to design the exact path in 3D space for the kite to follow. In other words, the path-planning problem is reduced to adjusting two variables λ_R and ζ_R by the right rate and magnitude.

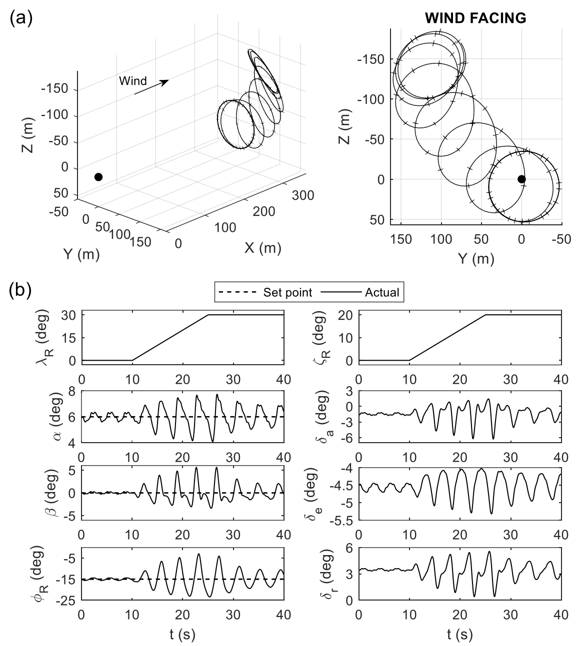

Figure 9. Response to simultaneous ramp inputs in λ_R and ζ_R.

Lastly, to adjust the circle radius, one simply needs to change the set point for the reference plane roll angle $\phi_{R_{SP}}$. Our previous work has shown that leaning more into the turn reduces the radius, while leaning out widens the orbit (Nguyen et al., 2026b). In this example of a left-turning trajectory, increasing the radius requires a more positive ϕ_R. Figure 10 shows two circular trajectories corresponding to $\phi_{R_{SP}}$ = –15º and 0º. It can be seen that the 0º case achieves a larger radius as intended.

This section has shown that by using the ailerons to track a fixed roll angle on a user-defined reference plane, the kite can keep itself in circular orbit without requiring complex path planning. Tilting the reference plane (i.e., adjusting λ_R and ζ_R) provides a means to shift the circular orbit to a desired location. Additionally, the radius can be widened or tightened by changing the set point for the reference plane roll angle. These ideas provide the foundation for a circular reel-out controller, which is described in the following section.

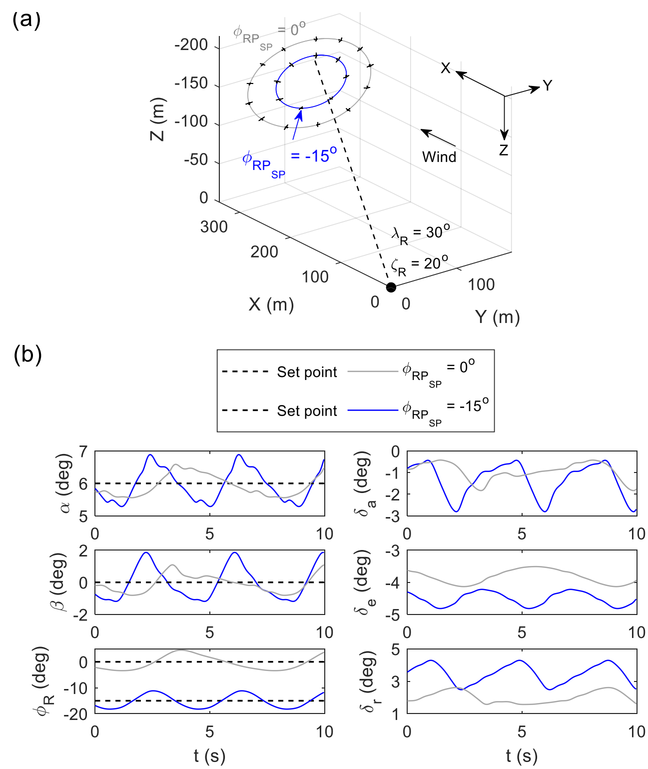

Figure 10. Closed-loop trajectories (a) and time histories (b) with different $\phi_{R_{SP}}$. The kite is drawn true to scale at 0.5-second intervals.

3.2 Outer loop: tracking a target cylinder

During circular reel out, the kite's flies in a helical trajectory resembling a spring that 'wraps' around an imaginary cylinder, shown in light blue in Fig. 11. The cylinder can be defined by a third coordinate system called the production plane. Its orientation with respect to the Earth's plane is defined by three axes X_P, Y_P, and Z_P, along with two angles λ_P and ζ_P – constructed in a similar manner to λ_R and ζ_R in the reference plane. However, unlike the reference plane, the production plane does not have to originate from the winch. The user can specify the location of the production plane's origin, which is shown as point P in the top plot of Fig. 11. The Earth-axis coordinate of point P is referred to as P_X, P_Y, and P_Z. From here, the target cylinder is defined to have a circular base on the X_P-Y_P plane, centred around point P with radius R_{SP}, and extends in the negative Z_P direction. For convenience, we refer to the kite's location on the production plane as X_P, Y_P, and Z_P. For instance, Fig. 11 depicts the production plane originating from point P with Earth-axis coordinates $[P_X, P_Y, P_Z]$ = [200, 0, –150] m. If the kite is also located at point P, then the kite's location is $[X, Y, Z]$ = [200, 0, –150] m using Earth-axis coordinates, and $[X_P, Y_P, Z_P]$ = [0, 0, 0] m on the production plane. The equations to convert from Earth to production plane coordinates are provided in appendix B.

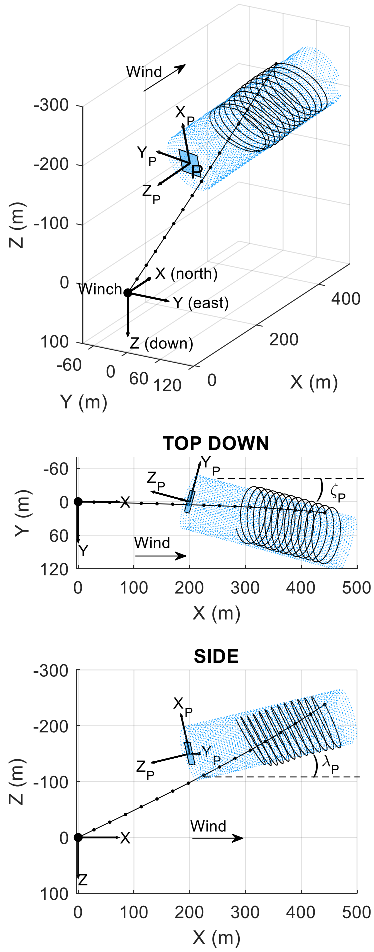

Figure 11. Target cylinder (light blue) and the production plane (blue square with axes X_P, Y_P, and Z_P).

The kite's radius on the production plane is calculated as $R = \sqrt{X_P^2 + Y_P^2}$. To keep R close to the desired radius R_{SP}, the three inputs to the inner loop $\phi_{R_{SP}}$, λ_R, and ζ_R need to be adjusted automatically via a feedback scheme. We refer to this second feedback layer as the outer loop, which works based on the following principles:

- Adjust the amount of roll via $\phi_{R_{SP}}$ to keep the actual radius R close to the desired radius R_{SP}.
- Move the circle's centre up or down by adjusting λ_R, so that the kite's production-plane coordinate X_P is kept oscillating around zero.
- Moves the circle's centre left or right by adjusting ζ_R, so that the kite's production-plane coordinate Y_P is kept oscillating around zero.

This feedback scheme can be achieved using three additional PI controllers as shown in Fig. 12. The components previously presented in Fig. 7 form the inner loop, while three new PI elements form the outer loop. In addition, a low-pass filter with time constant τ = 10 sec is used on the R feedback path to reduce $\phi_{R_{SP}}$ oscillation since large changes in radius can be achieved by a gradual adjustment in $\phi_{R_{SP}}$. The resulting outer loop dynamics is described by Eqs. (8)-(11):

$$\dot{R}_{filtered} = -\frac{1}{\tau} R_{filtered} + \frac{1}{\tau} R \tag{8}$$

$$\phi_{R_{SP}} = PI(R_{SP}, R_{filtered}) \tag{9}$$

$$\lambda_R = PI(0, X_P) \tag{10}$$

$$\zeta_R = PI(0, Y_P) \tag{11}$$

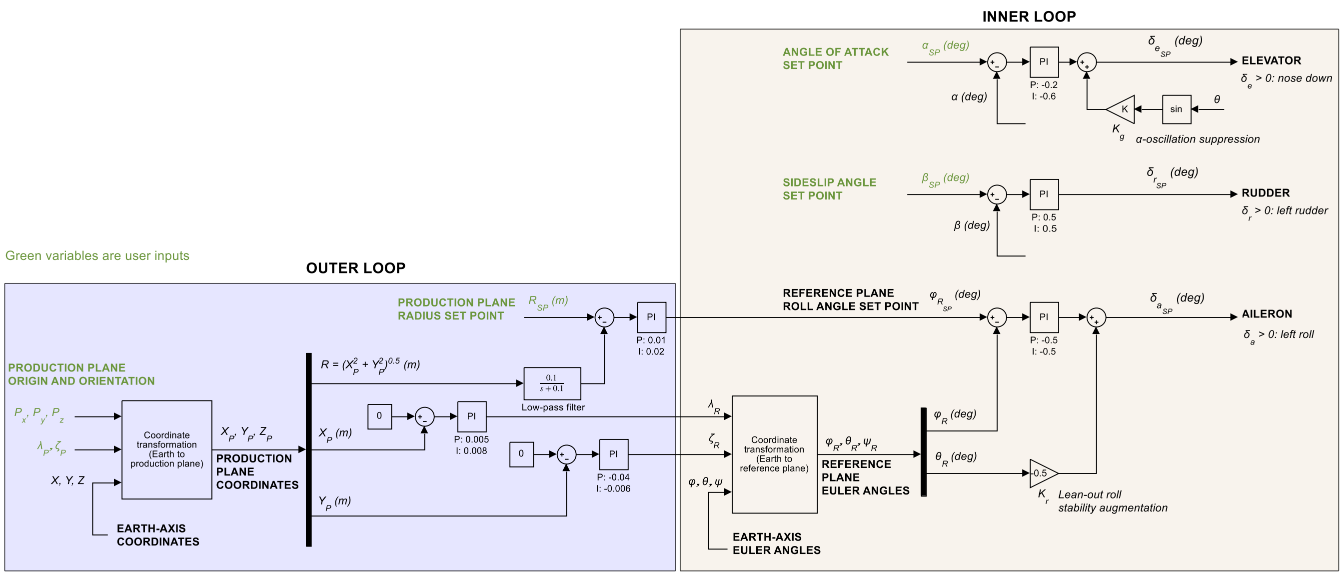

Figure 12. Full flight control system. The Earth-to-production-plane coordinate transformation is presented in appendix B.

In addition, the inner loop in Fig. 12 shows an optional $K_g \sin\theta$ feedback path to the elevator. This term reduces the angle of attack oscillation amplitude and is only used in Section 6. All results in Sections 4-5 do not use this additional term, effectively having K_g = 0. The other control gains are kept identical throughout the paper for circular flights.

An example loop reel-out trajectory with the tether visible is depicted in Fig. 11. This helical flight path wraps around the cylinder without requiring an exact 3D space to be specified. Therefore, the flight control problem has been reduced to a one-degree-of-freedom task handled by the ailerons (as elevator and rudder are only used for adjusting angle of attack and sideslip to achieve aerodynamic efficiency). The Julia code for the full flight control system is provided in appendix C. This appendix and the main body of this paper have been written to provide enough context for automatic implementation by a large language model. For instance, the authors have successfully used Claude Code by Anthropic (version 2.1, run as an extension in Visual Studio Code) to build a fully functional closed-loop system in MATLAB. Claude Code was granted access to only this manuscript and an open-loop version of the simulator.

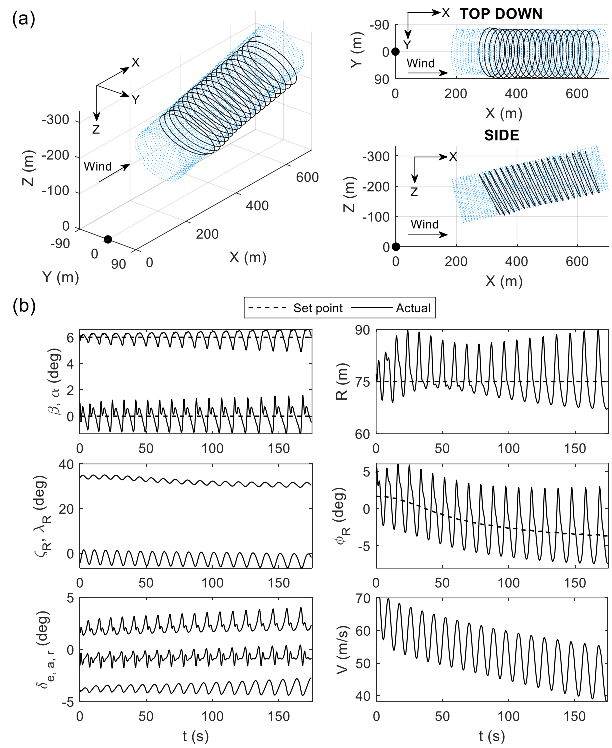

Figure 13. A typical circular reel-out trajectory with parameters P = [200, 0, –150] m, [λ_P, ζ_P] = [13°, 0°], and R_{SP} = 75 m. The winch speed is kept constant at 2.0 m/s.

4 Results

4.1 Uniform wind

A typical reel-out trajectory is presented in Fig. 13, which tracks a target cylinder of 50-75 m radius (75 m in this case), originating from point P with Earth-axis coordinate [P_X, P_Y, P_Z] = [200, 0, –150] m, tilted up by 13°, and pointing in the downwind direction ([λ_P, ζ_P] = [13°, 0°]). Setting λ_P to 13° results in a slight climbing trajectory that reduces the risk of the tether scraping the ground at long tether length but at the cost of higher cosine losses (Diehl, 2013). In practice, this loss can be mitigated by stronger winds experienced at higher altitudes. Results in Fig. 13 shows the kite successfully achieving reference tracking in angle of attack, sideslip, and cylinder wrap-around. Note the second row in Fig. 13b, which shows the three output variables λ_R, ζ_R, and $\phi_{R_{SP}}$ of the outer loop. All three variables require gradual adjustment as the tether length extends to compensate for increasing tether drag. By conventional feedback control standard, the variation in radius can be considered large, reaching 15% above the set point toward the end of reel out. Nevertheless, the 3D trajectory suggests that this variation still results in a reasonable trajectory. Control surface usage increases as the tether length increases due to higher tether mass and drag, resulting in lower airspeed and hence reduced control effectiveness. Therefore, a limiting factor in how far the kite can reel out is control surfaces magnitude saturation (in addition to other efficiency issues that can be studied with an optimisation algorithm).

To further stress-test the controller, tracking performance with less ideal reel-out paths is now examined. Figure 14 shows a diagonal reel out, which is achieved by setting ζ_P to 15°. Sideway reel out is shown in Fig. 15, where both ζ_P and λ_P are 0°, but the target cylinder is shifted to the side by 100 m. Both instances show the kite successfully tracking the blue cylinder despite higher lateral force from the tether.

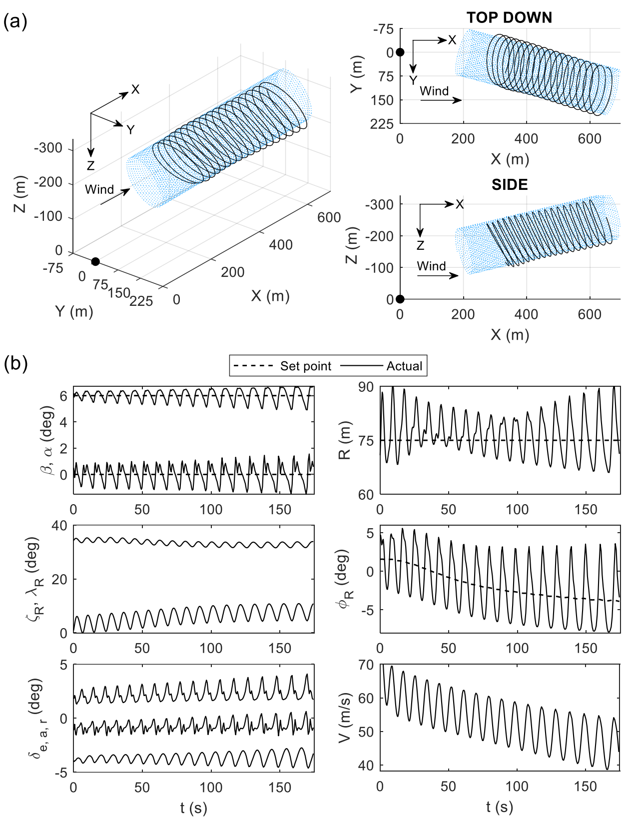

Figure 14. Diagonal reel out with parameters P = [200, 0, –150] m, [λ_P, ζ_P] = [13°, 15°], and R_{SP} = 75 m. The winch speed is kept constant at 2.0 m/s.

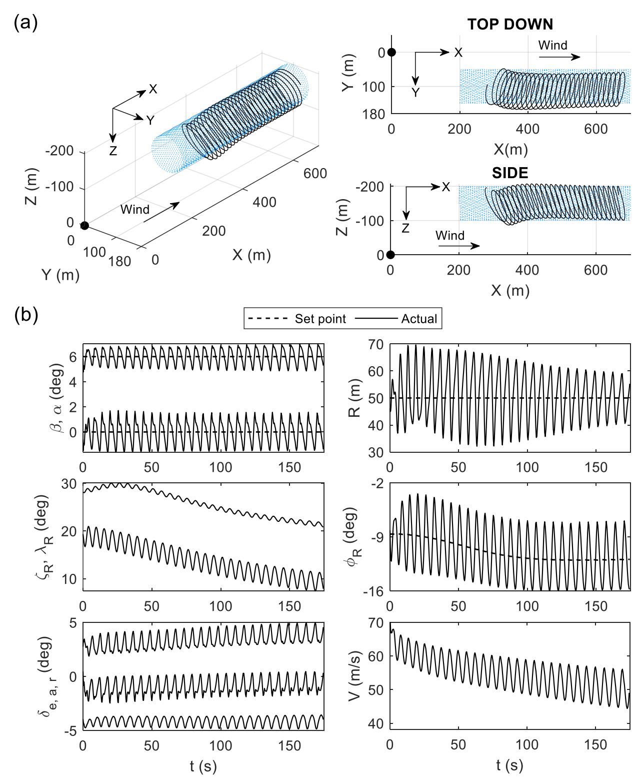

Figure 15. Sideway reel out with parameters P = [200, 100, –150] m, [λ_P, ζ_P] = [0°, 0°], and R_{SP} = 50 m. The winch speed is kept constant at 2.0 m/s.

For the final robustness tests, the target cylinder in Fig. 13 is flown again with the controller implemented in discrete form using a low sample rate of 20 Hz while subjected to a of 0.1 sec delay to all feedback variables. After 50 seconds of reeling out, the wind direction suddenly changes from 0º to 30º. Figures 16 and 17 show the closed-loop responses with and without corrective actions to the target cylinder to accommodate the new wind direction. In both cases, the kite successfully follows the target cylinder while tracking the angle of attack and sideslip set points despite the sub-optimal condition. Actuator rate limiting is not an issue, as a peak rate of only 14 º/s was recorded in the yaw channel of both figures. A lot of the oscillations before the wind-direction change can be attributed to the simple design of the inner loops, which exacerbates the poor transient responses in the presence of feedback delay and low sample time. As discussed in section 3.1, better performance can be achieved with a more sophisticated inner loop. The outer loop, on the other hand, has demonstrated sufficient robustness to guide the kite in sub-optimal conditions even with a simple inner loop design.

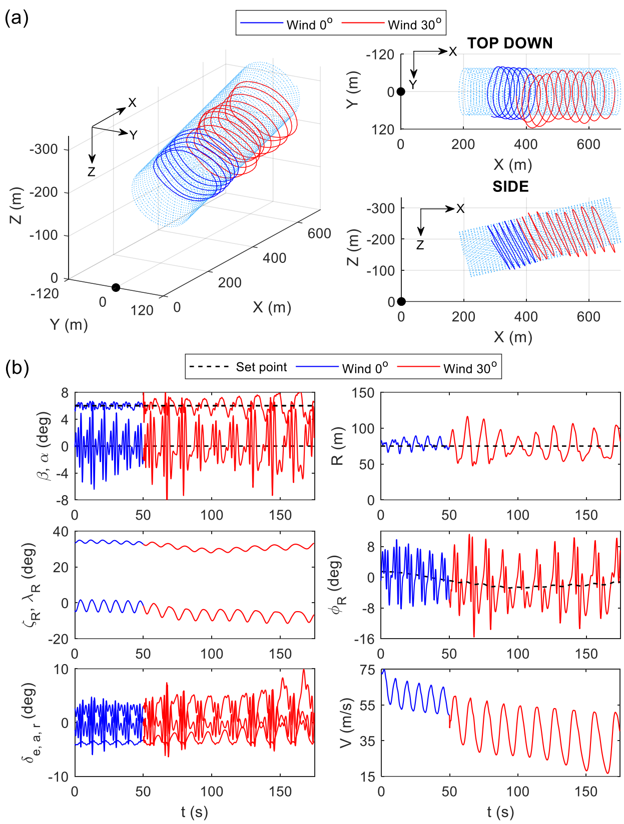

Figure 16. Closed-loop response to a sudden change in wind direction using a discrete controller (10 Hz) with feedback delay (0.1 sec) and no corrective action to the target cylinder.

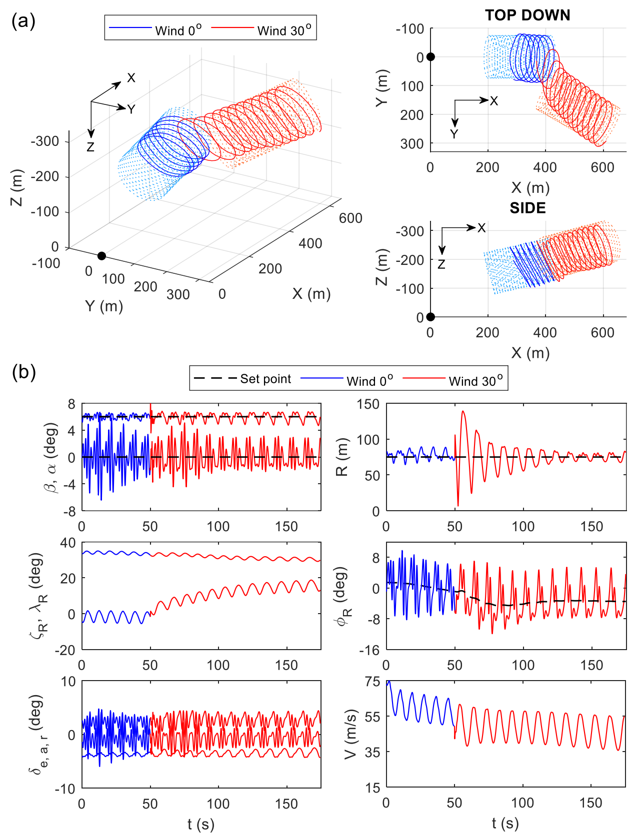

Figure 17. Closed-loop response to a sudden change in wind direction using the same discrete controller but with corrective actions to the target cylinder.

4.2 Non-uniform wind with turbulence

Experiments have shown that there is considerable variation in wind speed with height, so a feasible controller must be able to perform in a non-uniform wind field. The simulation is now augmented with realistic wind and turbulence conditions.

Two wind fields shown in Fig. 18 are used. The first one (referred to as medium wind) has an average speed of 12.1 m/s at 150 m height, and the second one (high wind) averages at 22.9 m/s at 100 m. Both wind fields were derived from LiDAR data at Kitemill's test site in Lista, Norway, on two different days. Variations in the wind profile over time are also captured in the LiDAR data as shown in Fig. 18 and implemented in the simulation.

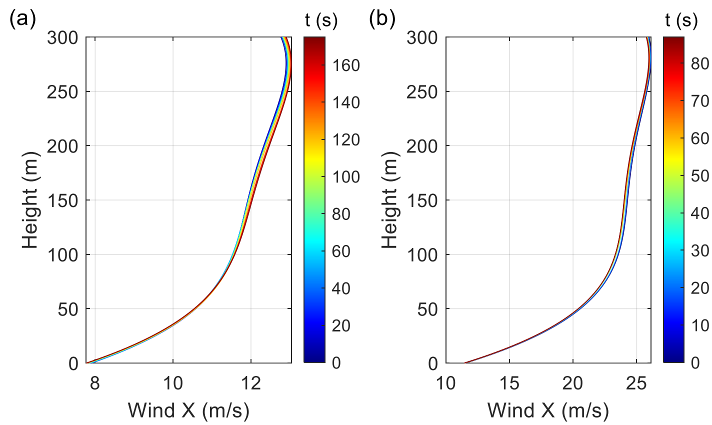

Figure 18. Medium (a) and high (b) wind fields implemented in the simulation based on LiDAR data.

Additionally, LiDAR measurements at the site have shown that atmospheric turbulence is present, resulting in significant disturbances in both wind speed and direction. To capture this phenomenon, we implemented a turbulence box based on the Mann model (Mann, 1994). This box covers the 3D space flown by the kite with local wind velocity fluctuations due to atmospheric turbulence. Such fluctuations are added to the wind velocity seen by the kite as the turbulence box is convected downstream with the mean wind speed. This modelling approach follows Taylor's frozen turbulence hypothesis, which states that the turbulent eddies move downstream with the mean wind speed and assumes the structures remain mostly unchanged as they propagate (Taylor, 1937). Generating the turbulence box requires the Mann model parameters: dominant turbulence length scale L_M, amplitude parameter $\alpha\varepsilon^{2/3}$ and anisotropy parameter Γ. The turbulence length scale and amplitude parameter have been estimated from flow field parameters (Kelly, 2018) measured by the LiDAR at the test site. Two different turbulence cases have been employed in the simulations according to the turbulence measured at the site, referred to as medium turbulence ($\alpha\varepsilon^{2/3}$=0.0108) and high turbulence ($\alpha\varepsilon^{2/3}$= 0.0267). The turbulence length scale has been defined as L_M =78.44m for both cases, being representative of the site, while the anisotropy parameter has been assumed as Γ = 3.0.

Figure 19 shows the closed-loop response to medium wind and medium turbulence, and Fig. 20 is for high wind and high turbulence. The first case (Fig. 19) exhibits notably poorer tracking. This is caused by the reduced control surface effectiveness at low airspeed during reel out, which becomes noticeable on the KM1 at airspeed below 35 m/s. The combination of wind gradient and turbulence creates multiple instances of the airspeed dropping below 35 m/s, leading to reduced tracking performance. A potential remedy for flying in such conditions is to reduce winch speed, which increases the apparent wind speed at the kite and results in higher airspeed. Conversely, the high wind and high turbulence case in Fig. 20 exhibits much better performance, despite significantly larger local wind speed fluctuation in all three directions. An even faster convergence to the blue cylinder can be achieved if the gains are tuned for high wind. Nevertheless, both simulations have demonstrated that the controller can provide accurate tracking in realistic wind conditions. Good performance can be expected when the airspeed is kept above 30 m/s, whether through flying in higher wind speed, reeling out slower, or both. A method to improve flight performance in low wind by dynamically varying the winch reel-out speed will be presented in a future publication.

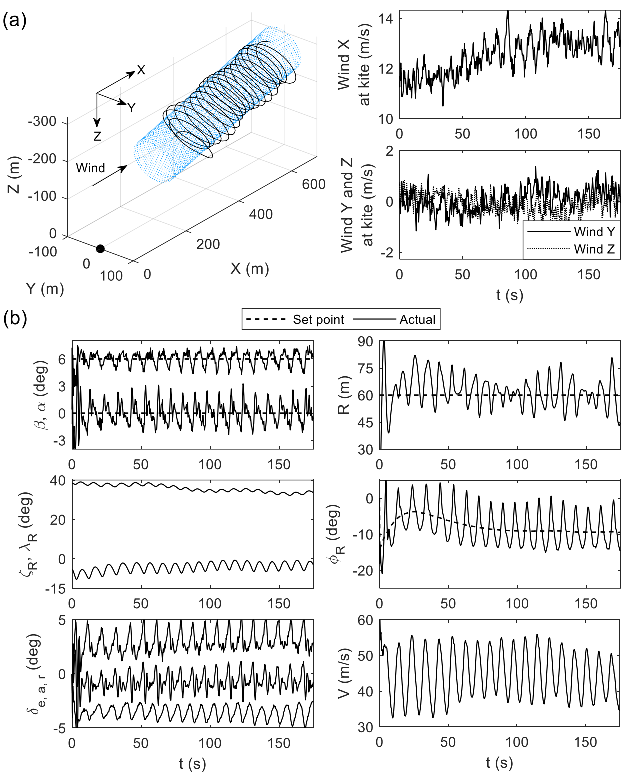

Figure 19. Reel out in medium wind and medium turbulence. The winch speed is kept constant at 2.0 m/s. A better tracking performance can be achieved by reducing the winch speed.

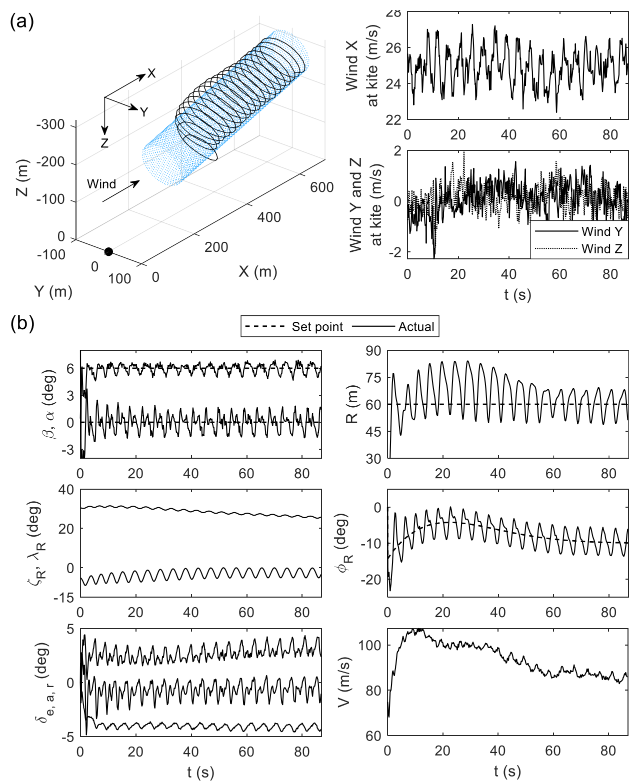

Figure 20. Reel out in high wind and high turbulence. The winch speed is kept constant at 4.0 m/s.

4.3 Computational aspects

For completeness, the numerical settings and performances of three representative runs are presented in Table II. We have chosen a solver for each platform that resulted in the best performance, namely ode45 (Shampine and Reichelt, 1997) for MATLAB and TRBDF2 (Rackauckas and Nie, 2017) for Julia. Simulations were conducted on an Intel i9-13950HX laptop with 32 GB of RAM. With best coding practices, a real time factor of at least 9 can be expected even in the presence of all computationally intensive components, such as non-uniform wind, turbulence, and discrete controller, as shown in the column for Figure 19. Note that all cases included live plotting of the numerical solutions before the solver completes running. Without live plotting, higher performance can be expected.

Table II. Computational performance of three presented simulations. All cases include live plotting of simulation results.

		Figure 13	Figure 16	Figure 19
Environmental settings	Tether nodes	15	15	15
	Wind	Uniform	Uniform	Non-uniform
	Turbulence	No	No	Yes
	Controller	Continuous	Discrete	Discrete
	Feedback delay	No	Yes	No
Numerical settings	Platform	Simulink	Simulink	Julia
	Solver	ode45	ode45	TRBDF2
	Relative tolerance	1e-3	1e-3	1e-3
	Absolute tolerance	1e-3	1e-3	1e-3
Performance	Solver steps	48,282	49,261	42,388
	Simulation time	175 sec	175 sec	175 sec
	Run time	11.0 sec	13.1 sec	19.7 sec
	Real-time factor	15.9	13.4	8.9

After generating the above, we conducted an internal study on tether discretisation and determined that using 15 tether nodes is excessive, whereas 8 nodes provide near-identical responses while requiring a fraction of computational cost. To demonstrate the cost savings, Table III presents the simulation performance of the scenario in Fig. 19 when only 8 nodes are used. A 52% improvement in computational speed is immediately achieved in the discrete controller case. Moreover, if a continuous controller is used, the memory requirement is only 9% or less compared to all the other cases considered (even when compared to Figure 13, which also uses a continuous controller but with 15 nodes). This memory reduction is crucial for running parameter sweeps. Finally, if live plotting is not required, then a real-time factor of 50 can be achieved on a continuous controller using 8 nodes.

Table III. Computational performance of the simulation in Fig. 19 but using only 8 tether nodes in Julia. Both cases include live plotting of simulation results.

	Discrete controller	Continuous controller
Solver steps	42,494	3,739
Simulation time	175 sec	175 sec
Run time	12.9 sec	10.0 sec
Real-time factor	13.5	17.5

This paragraph concludes all discussion involving non-uniform wind. The rest of the paper assumes uniform wind field with no turbulence. The wind speed is always 14 m/s, except for in Fig. 22.

5 Discussions

5.1 Winch speed control

Since our flight control system does not specify an exact flight path, the ‘wavelength’ of the resulting helical trajectory is dependent on winch control, not flight control. Figure 21 shows two reel-out cycles with the same target cylinders and wind conditions but different winch speeds. Figure 21a demonstrates reeling out faster, which creates more sparsely spaced loops and generally produces more energy (since mechanical power equals tether tension times winch speed). However, a high reel-out winch speed reduces the apparent wind speed at the kite, resulting in a drop in airspeed that can eventually stall the airframe. Stall can be prevented by reducing the winch speed as shown in Figure 21b, which has a higher airspeed than in Figure 21a toward the end of reel out. A detailed study on the impact of winch speed control on power production during reel out can be found in Nguyen et al., 2026a.

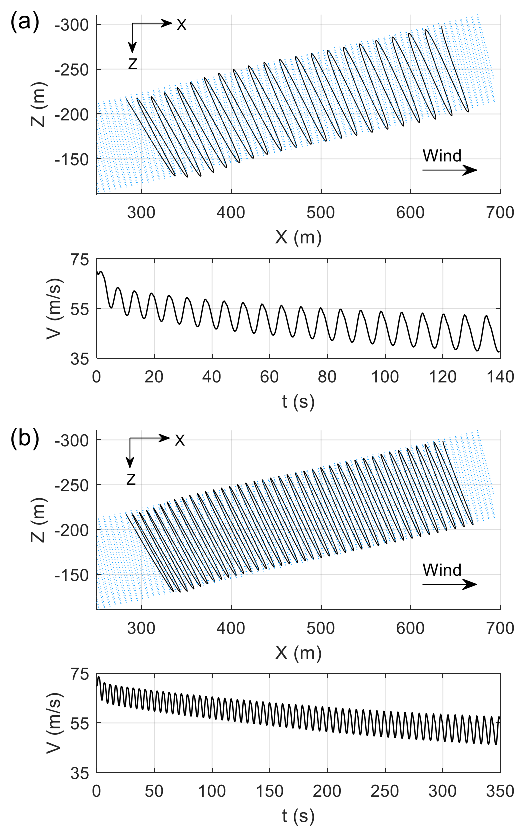

Figure 21. Reeling out at 2.5 m/s (a) and 1.0 m/s (b) winch speed, showing lower airspeed in (a) due to faster reel out.

By the same token, flying in lower wind speeds can be done by reducing the winch speed accordingly. The KM1 airframe stalls at 20 m/s airspeed, so the minimum acceptable airspeed during simulated reel-out in steady uniform wind was chosen to be 30 m/s. This apparently large margin is deemed necessary because steady and uniform wind is rarely encountered in real flights, so the kite must be able to remain airborne during temporal drops in wind speed. This airspeed constraint results in a reel-out winch speed of zero in 8 m/s uniform wind. Figure 10 shows the resulting trajectory: a circular orbit with the winch locked in place to keep the tether length fixed at 350 m – arguably a fixed-wing version of ‘parking’ the kite mid-air as seen in softkites AWES but achievable in much lower wind speeds (Fechner et al., 2015). The minimum airspeed is reached at the top of the circle. Therefore, preventing such a large drop in airspeed will enable the kite to operate in lower wind speeds. One method to do so is to phase-synchronise winch speed control with flight control. This winch control method can reduce the KM1’s ‘minimum parking wind speed’ to 6.7 m/s, thereby enabling power production in 8 m/s wind. Further details of such a winch control system will be provided in a future publication. In the context of this paper, we have shown that the flight control system can track the target cylinder as long as there is sufficient airspeed to prevent stall.

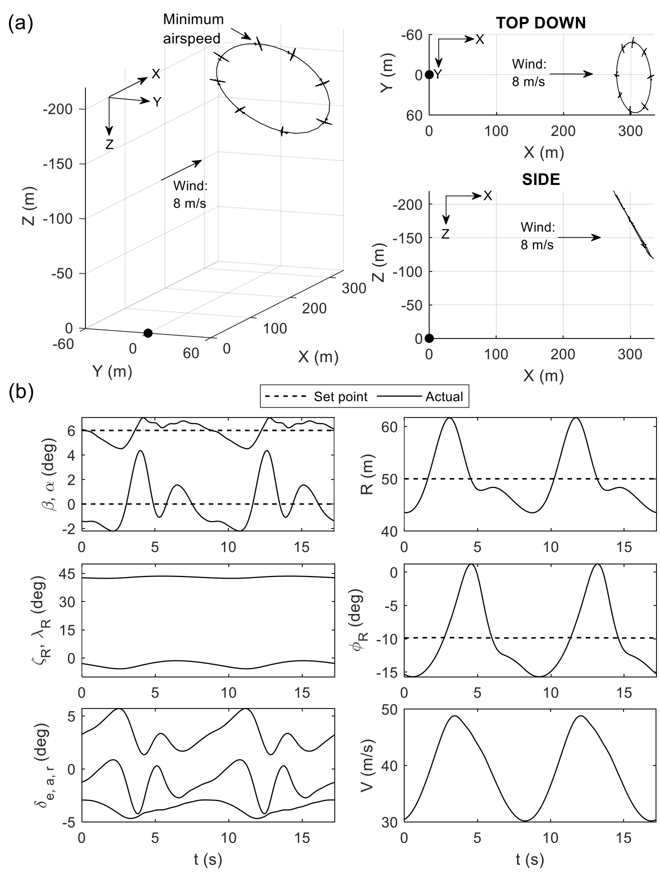

Figure 22. Trajectory under zero winch speed in 8 m/s wind. The kite is drawn true to scale at 1-second intervals. Time histories show data from two cycles.

5.2 Radius control with large error signal

Figure 23a shows the kite joining a sideway reel out from outside the target cylinder. A bottleneck point is seen, which is caused by the radius controller. At the starting point, the radius error is large, so the radius tracker tightens the turn to compensate for the large error signal. This manoeuvre can excessively reduce the radii of the enroute circular loops, causing potential control problems. To avoid the bottleneck problem, radius control should be turned off until the kite is already inside the target cylinder. Figure 23b shows the kite rejoining the same cylinder but with $\phi_{RP_{SP}}$ manually set to –3º at the start, which resulted in no bottleneck. Once inside the cylinder, radius tracking can be safely turned on.

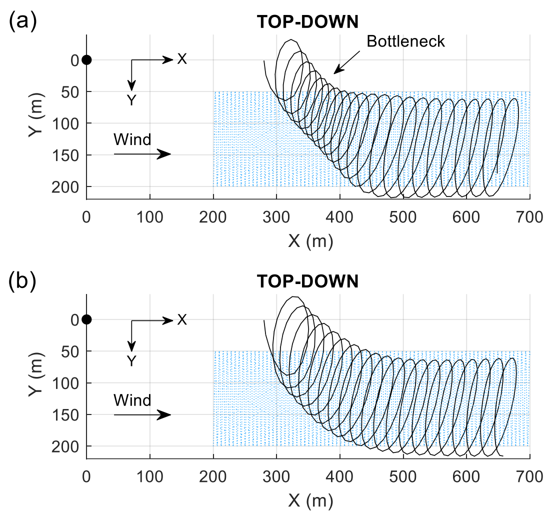

Figure 23. Bottleneck issue caused by enabling radius tracking when the kite is far outside the cylinder: (a) with radius tracking deployed, (b) without radius tracking.

5.3 Cosine losses at high altitudes

Early AWES literature has noted the cosine loss phenomenon, where given the same wind speed, power is reduced at higher altitudes (Diehl, 2013). This reduction is caused by a misalignment between the aerodynamic force and the wind vector. In the context of circular reel out, flying higher means individual orbits are angled further away from their ideal position of being 90º to the wind. The misalignment between aerodynamic force and wind direction increases, resulting in power reduction. A demonstration is provided in Fig. 24, which analyses three horizontal reel-out trajectories with same radii (50 m) but different target heights (reflected by different values for P_z, which is the Z coordinate of the target cylinder's origin in the Earth axis):

- Solid line: ideal trajectory orbiting the centre of the wind window with P_z = 0 m and no gravity. This flight path is not achievable in test flights due to the negative heights.
- Dashed line: 100 m height with gravity. This trajectory can be considered realistic.

- Dotted line: 250 m height with gravity.

The kite with highest altitude experiences more control difficulty, and notably produces 46% less power than in the ideal case. As altitude increases, the angle between individual circular orbits and the wind drops below the ideal value of 90°, which is achieved only when $P_z = 0$ m with no gravity. Due to this increasing misalignment with height, the wind contributes less to pulling the tether out, resulting in less power generated. Whilst flying higher also increases tether drag and reduces power, the contribution from the cosine loss is more significant. If the kite at $P_z = -250$ m was to achieve 21.9 kW (same power output as with $P_z = -100$ m), the wind speed must increase to 17 m/s.

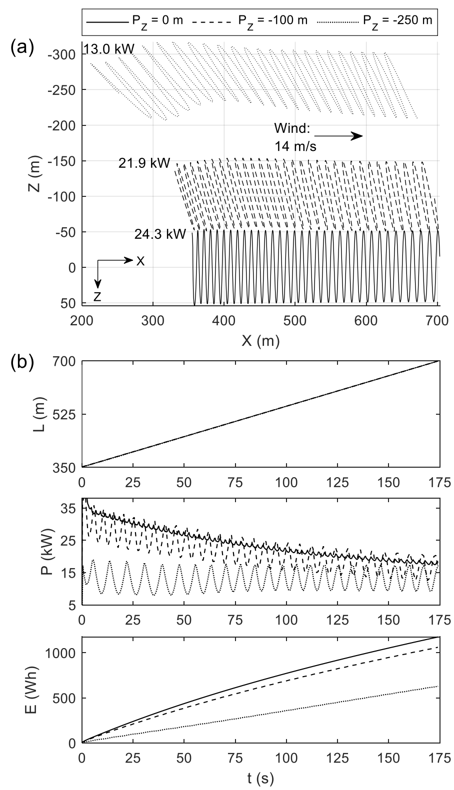

Figure 24. Different power production capacities at different heights due to cosine loss.

5.4 Minimum viable controller: ailerons only

Since all outer-loop signals are combined into just one control surface input (ailerons – see Fig. 12), the kite can follow the target cylinder with elevator and rudder fixed. Figure 25 shows one such example, where rudder is kept neutral and elevator is set to –4°. Power production is less efficient without active angle of attack and sideslip monitoring, but being able to test the flight controller without pitot tubes (for α and β measurements) can be useful for rapid development of new AWES prototypes. Mathematically speaking, flying with only one active control surface is possible because the goal of tracking a constant radius – regardless of the flight path taken – is a one degree-of-freedom problem. This formulation of feedback control reduces complexity and implies that a different control effector can be used. Indeed, test flights at Kitemill have successfully flown circular reel out using only rudder (ailerons and elevator are fixed), although simulations show that this method is less efficient for power generation than using ailerons (Nguyen et al., 2026a).

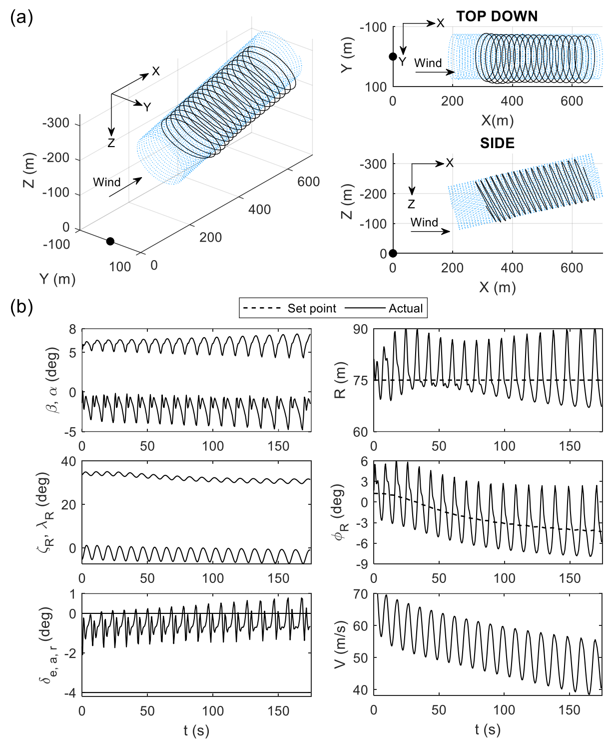

Figure 25. Normal reel out flown with fixed elevator (–4º) and rudder (0º). Parameters: P = [200, 0, –150] m, [λ_P, ζ_P] = [13º, 0º], and R_{SP} = 75 m. The winch speed is kept constant at 2.0 m/s.

5.5 Semi-manual mode

The kite can be flown using only the inner loop – even with the winch active. We refer to this configuration as the 'semi-manual mode'. Figure 26 shows an example. Five set points have to be specified by the user as shown in Fig. 7, namely $\phi_{R_{SP}}$, λ_R, and ζ_R for the ailerons, α_{SP} for the elevator, and β_{SP} for the rudder (noting that the last two can also kept fixed as discussed in section 5.4). With λ_R sufficiently high for satisfactory ground clearance and ζ_R not too far off the wind direction, the kite can sustain safe circular reel out. Steering the reel-out path can be achieved by manually adjusting λ_R and ζ_R, while radius adjustment can be done with $\phi_{R_{SP}}$.

For initial AWES control development, we recommend flying in semi-manual mode with the tether length fixed, gravity disabled, and λ_R and ζ_R set to 0º. Doing so enable the kite to converge to an equilibrium state by circling a point at zero height. From here, the α_{SP}, β_{SP}, and $\phi_{R_{SP}}$ loops can be tuned using classical methods.

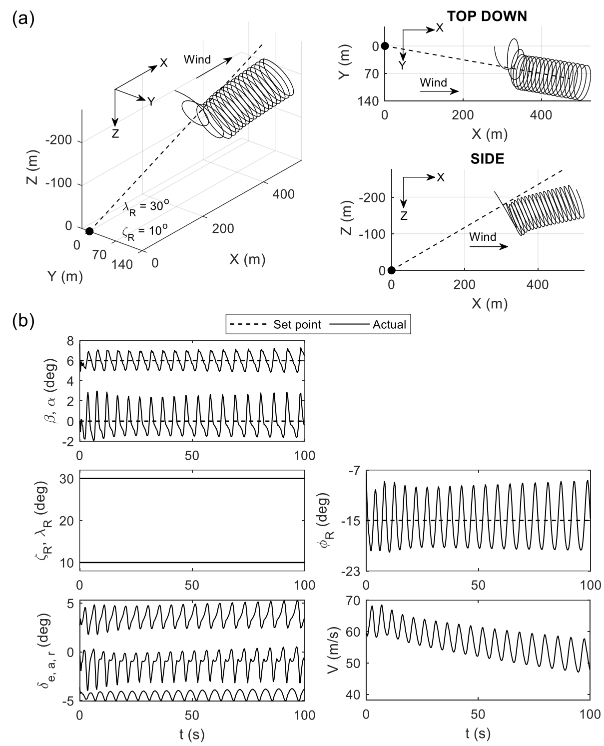

Figure 26. Semi-manual flight using only the inner loop.

5.6 Limit cycles

A potential downside of the controller is that it cannot converge to an equilibrium. To demonstrate, consider the full flight control system in a gravity-free environment with $[\lambda_P, \zeta_P]$ = [0º, 0º] and fixed tether length as shown in Fig. 27a. The resulting circular trajectory is perpendicular to the wind, and past work has shown that an equilibrium trim point exists in this condition (i.e., no cyclic control surface movement required) (Nguyen et al., 2026b). However, Fig. 27b shows that λ_R and ζ_R still exhibit small limit cycles caused by the integrated production-plane coordinates X_P and Y_P in the error signal. All three control surfaces are also in limit cycles as a result. These limit cycles disappear when λ_R and ζ_R are manually set to zero, at which point, the kite can be linearised for inner-loop control design. For the outer loop, the small-amplitude limit cycles can be ignored during manual tuning, but they may pose a challenge for rigorous control design. Further discussions on analysing nonlinear limit-cycle systems can be found in maths textbooks such as Jordan and Smith, 2007.

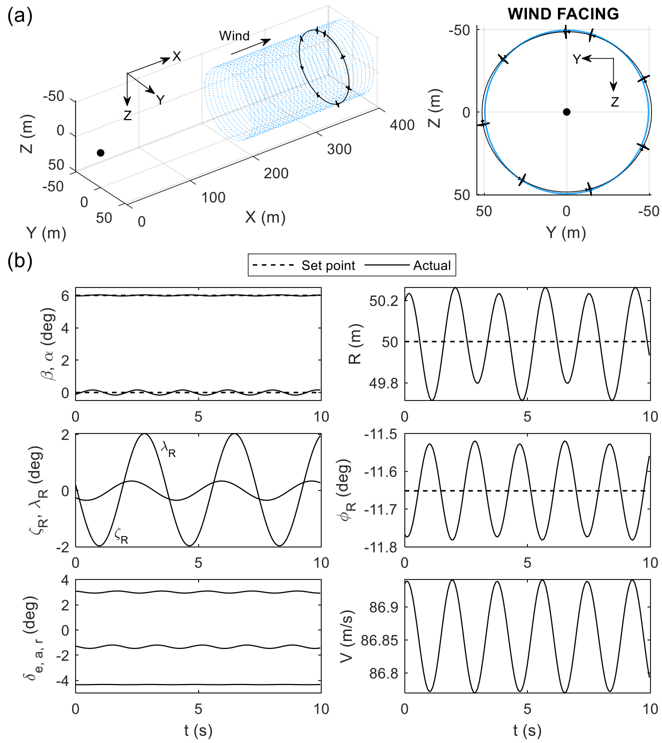

Figure 27. Closed-loop limit cycles, which is always present in the full flight controller even when equilibrium trim is possible.

6 Control law extensions

6.1 Reducing angle-of-attack oscillation

Although limit cycles cannot be eliminated, some can be reduced in amplitude. Angle of attack is one prime candidate for this reduction because flying at a near-constant α keeps the airframe close to its optimal value, which improves power generation. Two elements contribute to angle-of-attack oscillation during circular reel out: changing apparent wind direction at different points in the circle and changing gravity direction with respect to the airframe's body axis. Whilst the former is unavoidable, the latter can be reduced. Assuming no external wind, gravity's contribution can be seen in the last term of Eq. (12):

$$\dot{\alpha} = q - \frac{1}{2}\frac{\rho V S}{m} C_L - (p\cos\alpha + r\sin\alpha)\tan\beta + \frac{g}{V}\left(\frac{\cos\phi\cos\theta\cos\alpha + \sin\theta\sin\alpha}{\cos\beta}\right) \tag{12}$$

This is one of the standard 6-degree-of-freedom equations of motion for aircraft, but written in the aerodynamic wind axis (α, β, V) instead of the more common body axis velocities (u, v, w). If the contribution of gravity in Eq. (12) can be cancelled out using feedback control, angle-of-attack oscillation will be significantly reduced. This gravity term can be decomposed into two components

$$\dot{\alpha}_{g_z} = \frac{g}{V}\frac{\cos\phi\cos\theta\cos\alpha}{\cos\beta} \tag{13}$$

$$\dot{\alpha}_{g_x} = \frac{g}{V}\frac{\sin\theta\sin\alpha}{\cos\beta} \tag{14}$$

where $\dot{\alpha}_{g_z}$ describes how the body-axis (not Earth-axis) vertical component of gravity affects angle of attack, and $\dot{\alpha}_{g_x}$ reflects the contribution of the longitudinal component. Comparing both variables against α in Fig. 28, one can see that $\dot{\alpha}_{g_x}$ is roughly in phase with α and oscillates about zero, whereas $\dot{\alpha}_{g_z}$ has a slight phase lead and is offset from zero. Both properties of $\dot{\alpha}_{g_x}$ suggest that it can be used as a feedback signal on the elevator channel to cancel out the angle-of-attack oscillation. This is possible in the simulation. However, $\dot{\alpha}_{g_x}$ in Eq. (14) has three quantities that require measurements from a pitot tube: V, α, and β. If all three are removed from Eq. (14), the right-hand side reduces to just $g\sin\theta$. Figure 28 shows that this term has similar phase and offset properties as the full $\dot{\alpha}_{g_x}$ equation. Therefore, the feedback signal to the elevator can be simply $K_g\sin\theta$, where K_g is a proportional gain equal to g multiplied by a constant. This additional feedback path creates the second junction in the elevator path (see Fig. 12). As such, the augmented pitch control law is

$$\delta_{e_{SP}} = PI(\alpha_{SP}, \alpha) + K_g\sin\theta \tag{15}$$

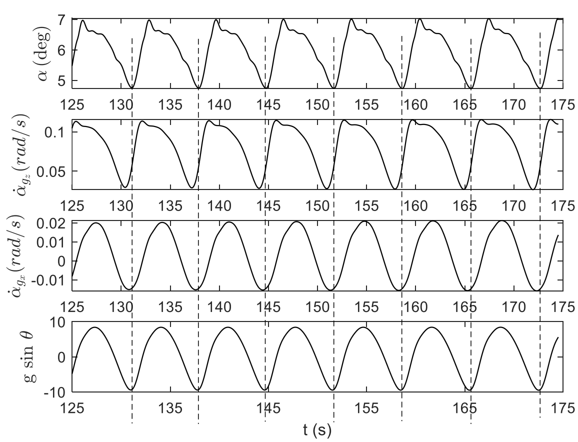

Figure 28. Potential feedback terms to cancel out the effect of gravity on α oscillation.

Figure 29 shows the effect of the new pitch control law on angle of attack and power generation over a normal reel-out trajectory but with a low radius of 50 m to exaggerate α oscillation. The dashed line labelled 'K_g = 0' indicates flying with without the α suppression controller. It can be seen that the controller successfully suppresses the limit cycle in α and provides a marginal 0.1 kW gain in power. This power gain is more noticeable if a less ideal winch is used. In Fig. 30, the same reel-out trajectory is flown again but with a winch that maintains a constant tether tension of 8 kN. This winch control mechanism is less efficient for power generation (Nguyen et al., 2026a), but is closer to what has been tested experimentally at Kitemill. In this arrangement, angle of attack oscillation has been significantly reduced. The kite experiences less aerodynamic drag as a result and completes the production cycle 14 seconds faster – a 0.9 kW gain in power. Inspecting the elevator movement in both Fig. 29 and Fig. 30, we can see that the new pitch control law has phase-shifted the cyclic elevator input and also demands more elevator movements compared to the K_g = 0 case, resulting in the desired α-suppression effect. This behaviour reflects the fact that our proposed control law is a form of dynamic inversion.

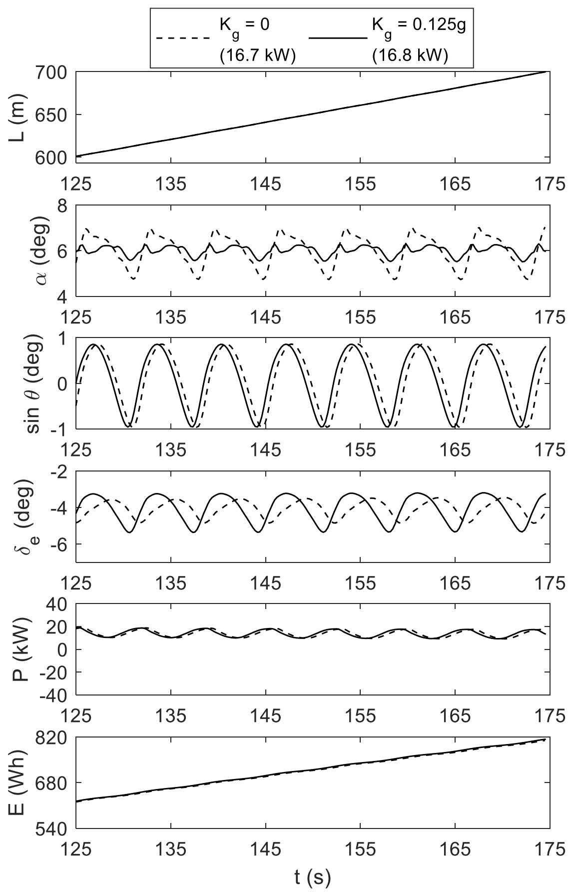

Figure 29 Effect of the α-suppression controller. The winch speed is constant at 2 m/s.

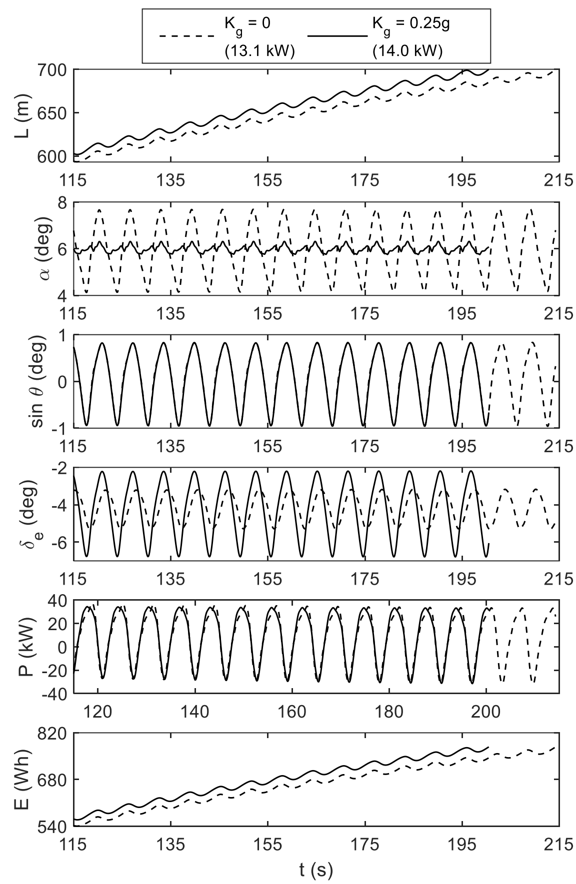

Figure 30 Effect of the α-suppression controller. The winch tracks a constant tether tension of 8 kN.

Since gravity's contribution is independent of the chosen flight path, the angle-of-attack suppression controller also works with figure-of-eight reel out. This capability is demonstrated in section 6.3. Appendix D presents the stability proof of the α dynamics in closed loop with the proposed controller.

6.2 Multi-kite synchronisation

Our proposed flight controller can be extended to synchronise multiple kites in a farm configuration, where a minimum separation rule may be in place. The working principle is based on the fact that without a dedicated propulsion unit, the kite's angular velocity in the cylinder is dependent primarily on radius. Faster sweep is achieved by lowering the radius and vice versa, as shown in Figure 10. Within the same 10-second window, the blue trajectory with a smaller radius completes around 2.5 orbits, whereas the grey one does only 1.5. Therefore, phase synchronisation of multiple kites can be achieved by adjusting the radius. To demonstrate, consider a hypothetical two-kite farm scenario in Fig. 31. Two target cylinders with 80 m radius are placed with only 100 m in lateral separation between their centre lines, leading to 60 m of overlap in the Y direction. The angular phase Ω of each kite in its target cylinder is defined as

$$\Omega = \text{atan2}(Y_P, X_P) \tag{16}$$

Note that Ω = 0° corresponds to the kite being at the highest point of the circle.

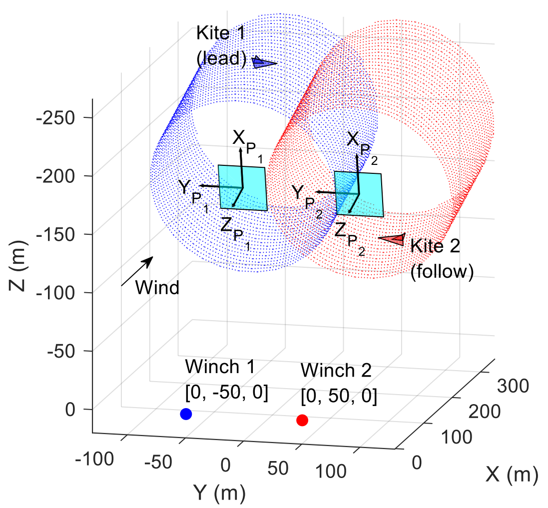

Figure 31. Starting location of the two kites for synchronisation testing.

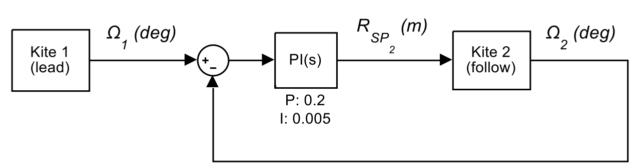

Figure 32. Phase synchronisation controller in the second kite.

The lead kite is flown normally using the control scheme in Fig. 12. For the second kite, the radius set point R_{SP_2} is automatically adjusted using an additional PI loop presented in Fig. 32. This controller seeks to match the second kite's phase with the first kite's by adjusting the second kite's radius. To prevent overcorrection, the output R_{SP_2} of the phase-synchronisation controller is limited to between 60 m and 100 m, which is no more than ±20 m from R_{SP_1} = 80 m.

The phase synchronisation controller is now tested with two kites starting from a 180° out-of-phase position as shown in Fig. 31. Longitudinal separation is provided by the different starting tether lengths: 350 m for kite 1 and 300 m for kite 2. The resulting trajectories in Fig. 33 show successful synchronisation after 35 seconds, or just under 3 loops. Once the phase gap is closed, the second kite remains in sync for the rest of the reel-out cycle.

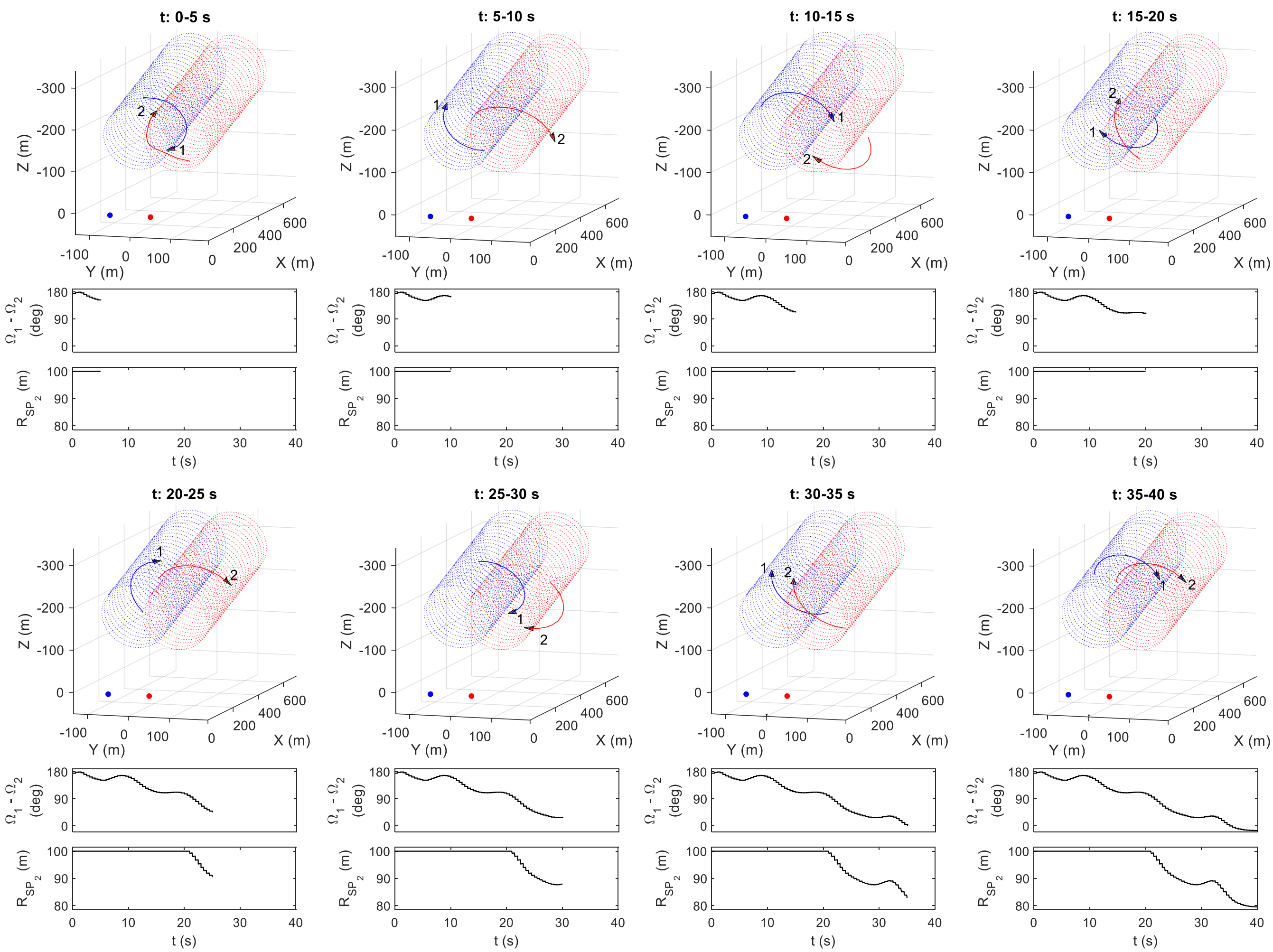

Figure 33. Simulation result of the phase synchronisation controller.

6.3 Figure-of-eight flight pattern

The proposed control framework can be adapted to fly figure-of-eight (henceforth abbreviated to ‘F8’) patterns using a proportional feedback law. Although this method cannot provide non-symmetric flight capabilities as seen in circular-flight analysis (i.e., sideway and diagonal reel out), it is sufficient as a demonstrator for rapid simulator development and verification. Consider a generic F8 pattern as shown in Fig. 34a. The production plane lo cation is now shifted to the winch at the origin with the elevation and azimuth angles λ_P and ζ_P set to zero. In this configuration, the axis Z_P points in the opposite direction of the wind. Using the same definition of angular phase Ω from Eq. (16), we have the evolution of Ω over two F8 cycles as shown in Fig. 34b. This evolution resembles the amount of roll on the production plane required to fly an F8 path. Near the centre of lemniscate (point A), this roll angle crosses zero, reaching its peak positive and negative values at C or B (depending on whether an eight-up or eight-down pattern is flown). The phase angle Ω can therefore be used as the feedback signal to change the reference plane roll angle set point using a proportional control law with gain K_8

$$\phi_{R_{SP}}\ [deg] = K_8 \Omega\ [deg] \tag{17}$$

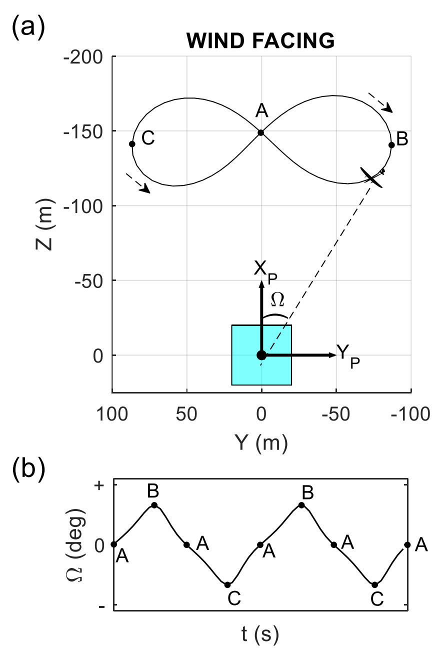

Figure 34. A generic figure-of-eight pattern (a) and evolution of the phase angle over two cycles.

where the sign of the proportional gain K_8 determines if the trajectory is eight-up or eight-down. Setting a larger magnitude for K_8 means a tighter turn when the kite reaches the edge of the lemniscate, resulting in a narrower F8 trajectory. The aileron control law requires a minor modification

$$\delta_{a_{SP}} = PI(\phi_{R_{SP}}, \phi_R) + K_r \theta_r K_s \phi_{RP_{SP}} \tag{18}$$

noting that the roll stiffness term is now multiplied by a factor $K_s \phi_{RP_{SP}}$, with K_s being another proportional gain. This addition accommodates the fact that the roll stiffness gain K_r must change sign depending on which side of the lemniscate the kite is flying on. Because $\phi_{RP_{SP}}$ has a smooth zero crossing near centre of the lemniscate, this signal can be used to indicate sign change after appropriate scaling by K_s to keep the peak magnitude of $K_s \phi_{RP_{SP}}$ close to 1. In all subsequent results, K_s is fixed at –0.05. Figure 35 shows the block diagram of the F8 controller. The only user input available (beyond control gains) is the reference plane elevation angle λ_R. All other guidance parameters must be set to zero to avoid instability caused by the limited steering capability of the proportional control law, which requires symmetry in the lemniscate to function properly.

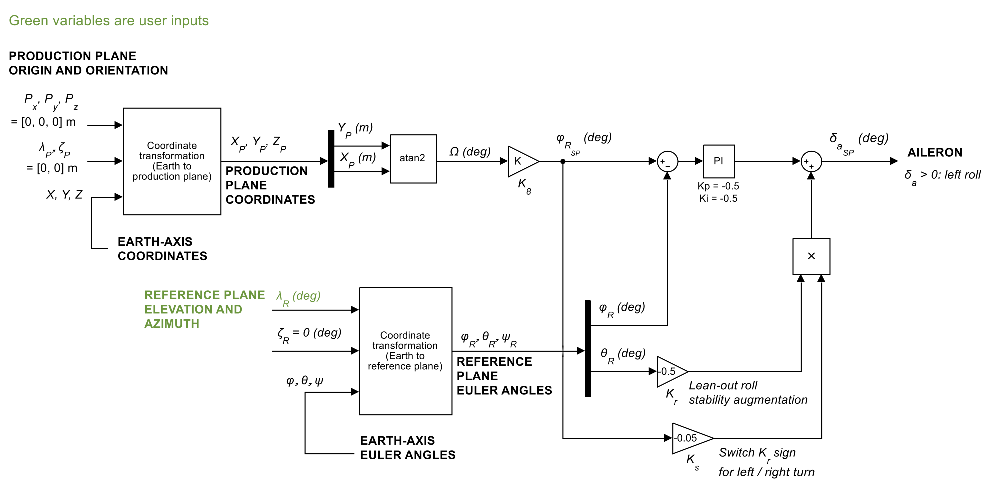

Figure 35. Block diagram of the figure-of-eight flight controller for aileron. Note that the elevator and rudder loops remain unchanged.

The controller's performance is now assessed. Figure 36 shows a few F8 patterns with positive and negative gains. As discussed, the lemniscate's width can be reduced by increasing K_8 magnitude. To fly at a higher elevation, λ_R should be increased as shown in Fig. 37.

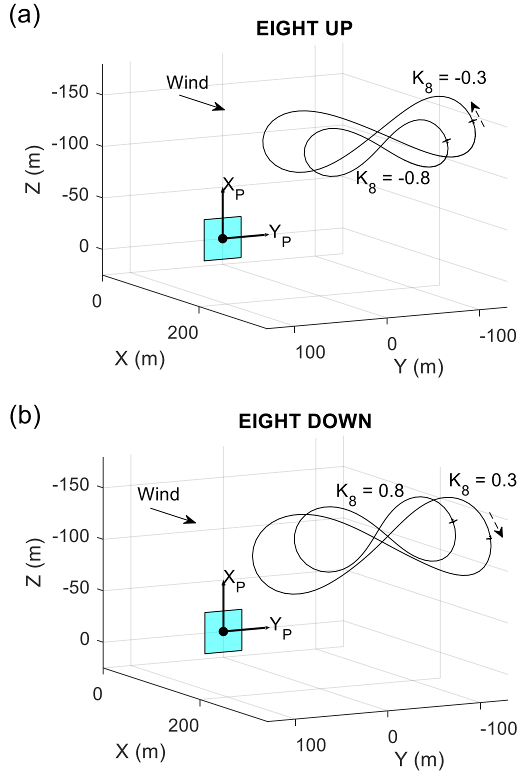

Figure 36. Eight-up (a) and eight-down (b) trajectories at λ = 25º with different control gains.

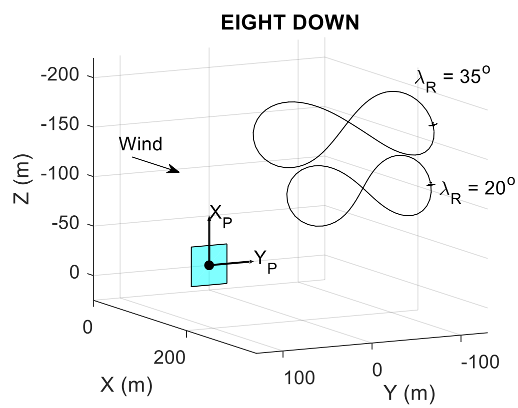

Figure 37. Effect of changing λ_R.

An F8 pattern can be considered to be two small circles on each side joined at the centre of the lemniscate. Each side-circle's radius is smaller than typical values for circular reel out. As a result, the angle of attack and sideslip variation can be large, although the former can be alleviated using the α suppression controller in Eq. (15). Figure 38 demonstrate the controller's effectiveness in keeping α close to its 6° set point throughout a typical F8 reel-out cycle. This result confirms that the α-suppression controller can be used on both circular and figure-of-eight flight.

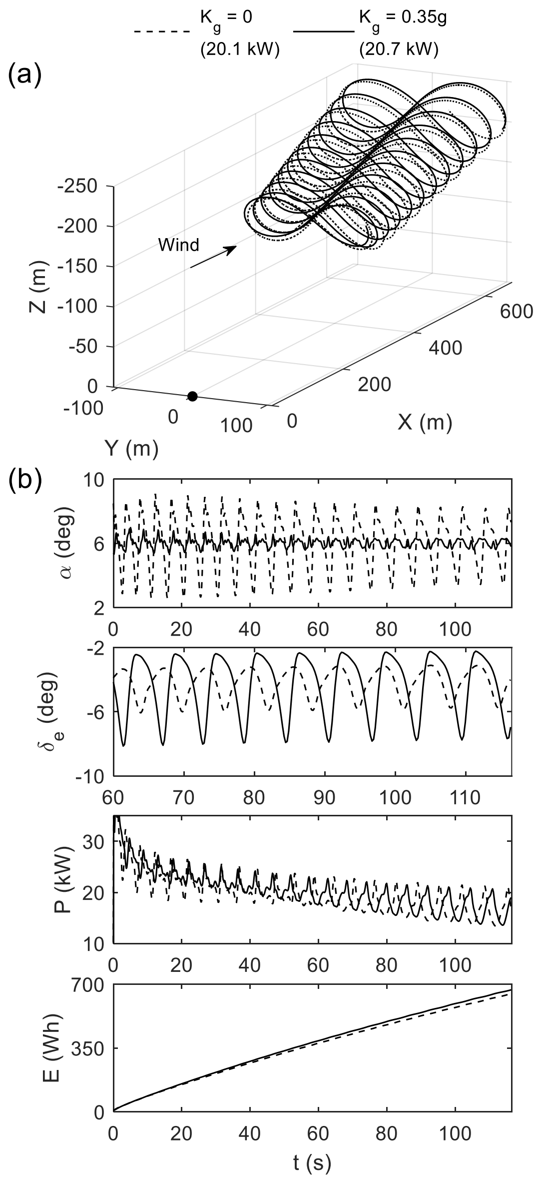

Figure 38. Eight-down reel out without and with α-suppression control.

7 Conclusions

This paper has shown that by relaxing the path-following requirement, a simple proportional-integral feedback system can provide accurate flight control during circular-pattern reel out. The key enabling idea is in tracking the roll angle on a non-static reference frame. In its simplest form, the controller achieves accurate path following using only ailerons while requiring no air data from a pitot tube. This capability could enable rapid control law development and testing of new AWES prototypes. Future improvements can implement more advanced controllers, such as nonlinear dynamic inversion, for the same roll-angle tracking idea but with higher performance. Additionally, recognising that AWES are self-oscillating systems opens up a new class of nonlinear controllers with great potential for the sector. An example of leveraging the inherent limit cycle property to improve performance is provided in the extension to the pitch control loop. By adding a proportional pitch angle feedback term to the elevator, angle-of-attack oscillation can be significantly reduced. Doing so enables the kite to spend more time in its optimal aerodynamic condition and results in more power. Lastly, an additional PI loop that adjusts circular flight radius provides indirect control of the kite's angular velocity. This property can be exploited to enable phase-synchronisation of multiple kites in a farm configuration. The proportional-integral architecture also positions the controller as a suitable baseline system for benchmarking more advanced control laws. All of the above developments contribute to making AWES safer and more efficient, thereby bringing them one step closer to commercial use.

Appendix A: Earth-to-reference-plane transformation

The Euler roll, pitch, and yaw angles on the reference plane [ϕ_R, θ_R, ψ_R] are obtained from their Earth-axis counterparts [ϕ, θ, ψ] and the reference plane's elevation and azimuth angles [λ_R, ζ_R]. This procedure is outlined in Eqs. (A1-A7). Subscripts E, B, and R to denote Earth, body, and reference frames, respectively.

$$\boldsymbol{q}_{EB} = \begin{bmatrix} \cos(\psi/2) \\ 0 \\ 0 \\ \sin(\psi/2) \end{bmatrix} \otimes \begin{bmatrix} \cos(\theta/2) \\ 0 \\ \sin(\theta/2) \\ 0 \end{bmatrix} \otimes \begin{bmatrix} \cos(\phi/2) \\ \sin(\phi/2) \\ 0 \\ 0 \end{bmatrix} \tag{A1}$$

$$\boldsymbol{q}_{ER} = \begin{bmatrix} \cos(\zeta_R/2) \\ 0 \\ 0 \\ \sin(\zeta_R/2) \end{bmatrix} \otimes \begin{bmatrix} \cos\left((\lambda_R + 90^o)/2\right) \\ 0 \\ \sin\left((\lambda_R + 90^o)/2\right) \\ 0 \end{bmatrix} \otimes \begin{bmatrix} \cos(180^o/2) \\ \sin(180^o/2) \\ 0 \\ 0 \end{bmatrix} \tag{A2}$$

$$\boldsymbol{q}_{BR} = \boldsymbol{q}_{EB}^{-1} \otimes \boldsymbol{q}_{ER} = \begin{bmatrix} \eta \\ \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{bmatrix} \tag{A3}$$

where η, ϵ_1, ϵ_2, and ϵ_3 are the four elements in the 4 x 1 matrix $\boldsymbol{q}_{BR}$. This gives a direction cosine matrix $\boldsymbol{R}_R^B$ of dimension 3 by 3

$$\boldsymbol{R}_R^B = \boldsymbol{I} + 2\eta S(\boldsymbol{\epsilon}) + 2S(\boldsymbol{\epsilon})S(\boldsymbol{\epsilon}) \tag{A4}$$

where $\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, \epsilon_3]$ are the non-zero elements in the skew-symmetric matrix $S(\boldsymbol{\epsilon})$

$$S(\boldsymbol{\epsilon}) = \begin{bmatrix} 0 & -\epsilon_3 & \epsilon_2 \\ \epsilon_3 & 0 & -\epsilon_1 \\ -\epsilon_2 & \epsilon_1 & 0 \end{bmatrix} \tag{A5}$$

Substituting Eq. (A5) into Eq. (A4) and expand

$$\boldsymbol{R}_R^B = \begin{bmatrix} R_{R,11}^B & R_{R,12}^B & R_{R,13}^B \\ R_{R,21}^B & R_{R,22}^B & R_{R,23}^B \\ R_{R,31}^B & R_{R,32}^B & R_{R,33}^B \end{bmatrix} = \begin{bmatrix} 1-2(\epsilon_2^2+\epsilon_3^2) & 2(\epsilon_1\epsilon_2-\eta\epsilon_3) & 2(\epsilon_1\epsilon_3+\eta\epsilon_2) \\ 2(\epsilon_1\epsilon_2+\eta\epsilon_3) & 1-2(\epsilon_1^2+\epsilon_3^2) & 2(\epsilon_2\epsilon_3-\eta\epsilon_1) \\ 2(\epsilon_1\epsilon_3-\eta\epsilon_2) & 2(\epsilon_2\epsilon_3+\eta\epsilon_1) & 1-2(\epsilon_1^2+\epsilon_2^2) \end{bmatrix} \tag{A6}$$

which is based on Eq. (6.215) in Egeland and Gravdahl, 2003, but with η^2 replaced by $1-\epsilon_1^2-\epsilon_2^2-\epsilon_3^2$. From here, the reference-plane Euler angles can be obtained

$$\begin{bmatrix} \phi_R \\ \theta_R \\ \psi_r \end{bmatrix} = \begin{bmatrix} \mathrm{atan2}\left(R_{R,23}^B, R_{R,33}^B\right) \\ -\sin^{-1}\left(R_{R,13}^B\right) \\ \mathrm{atan2}\left(R_{R,12}^B, R_{R,11}^B\right) \end{bmatrix} = \begin{bmatrix} \mathrm{atan2}\left(2(\epsilon_2\epsilon_3-\eta\epsilon_1), \left(1-2(\epsilon_1^2+\epsilon_2^2)\right)\right) \\ -\sin^{-1}\left(2(\epsilon_1\epsilon_3+\eta\epsilon_2)\right) \\ \mathrm{atan2}\left(2(\epsilon_1\epsilon_2-\eta\epsilon_3), \left(1-2(\epsilon_2^2+\epsilon_3^2)\right)\right) \end{bmatrix} \tag{A7}$$

where atan2 is the four-quadrant version of the arctangent function. The atan2 function is available in most programming languages.

Appendix B: Earth-to-production-plane transformation

The production plane P is defined by its origin, which has an Earth-axis coordinates $[P_X, P_Y, P_Z]$, elevation angle λ_P, and azimuth angle ζ_P. If the kite's position on the Earth axis is $[X, Y, Z]$, its coordinates $[X_P, Y_P, Z_P]$ on the production plane is found using Eqs. (A8-A10)

$$\boldsymbol{q}_{EP} = \begin{bmatrix} \cos(\zeta_P/2) \\ 0 \\ 0 \\ \sin(\zeta_P/2) \end{bmatrix} \otimes \begin{bmatrix} \cos((\lambda_P+90^o)/2) \\ 0 \\ \sin((\lambda_P+90^o)/2) \\ 0 \end{bmatrix} \otimes \begin{bmatrix} \cos(180^o/2) \\ \sin(180^o/2) \\ 0 \\ 0 \end{bmatrix} \tag{A8}$$

$$\boldsymbol{R}_E^P = \boldsymbol{R}(\boldsymbol{q}_{EP}^{-1}) \tag{A9}$$

$$\begin{bmatrix} X_P \\ Y_P \\ Z_P \end{bmatrix} = \boldsymbol{R}_E^P \left(\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - \begin{bmatrix} P_X \\ P_Y \\ P_Z \end{bmatrix} \right) \tag{A10}$$

Appendix C: Julia code for the circular flight controller

The feedback scheme in Fig. 12 is a system of 7 first-order ordinary differential equations. The Julia code to implement this controller is presented below, and can be translated into other programming languages using a large language model. Due to the low gains used in the outer loop, the state `i_x_P_error` should be initialised at a non-zero value to facilitate quick convergence to the target cylinder and prevent unstable responses. For example, results in section 4.1 were all generated with initial `i_x_P_error` values that result in $\lambda_R \approx 30^{\circ}$.

```
# ---- REEL-OUT FLIGHT CONTROLLER ----

# 7 STATES
#   i_x_P_error       - Integrated x_P tracking error (m·s)
#   i_y_P_error       - Integrated y_P tracking error (m·s)
#   i_radius_P_error  - Integrated radius tracking error (m·s)
#   i_phi_R_error     - Integrated phi_R tracking error (deg·s)
#   i_alpha_error     - Integrated alpha tracking error (deg·s)
#   i_beta_error      - Integrated beta tracking error (deg·s)
#   filtered_radius_P - Low-pass filtered production-plane radius (m)
#
# INPUTS
#   phi, theta, psi - Earth-axis Euler angles (deg)
#   alpha, beta     - Angle of attack and sideslip (deg)
#   X, Y, Z         - Kite position in Earth axis coordinates (m)
#
# OUTPUTS
#   delta_a_SP - Demanded aileron deflection (deg)
#   delta_e_SP - Demanded elevator deflection (deg)
#   delta_r_SP - Demanded rudder deflection (deg)
#   *_dot      - Time derivatives of the 7 states

function reel_out_controller(
    alpha, beta, phi, theta, psi, X, Y, Z,
    i_x_P_error, i_y_P_error, i_radius_P_error,
```

```
    i_phi_R_error, i_alpha_error, i_beta_error,
    filtered_radius_P
)

    # Set points
    P_X = 200; P_Y = 0; P_Z = -150  # Production cylinder origin in Earth axis coordinates (m)
    lambda_P = 13; zeta_P = 0       # Production cylinder orientation (deg)
    radius_SP = 75                  # Radius set point (m)
    alpha_SP = 6; beta_SP = 0       # Angle of attack and sideslip set points (deg)

    # Gains. Variable names indicate the loop's outputs
    tau_radius_p = 10.0 # Low-pass filter cutoff frequency (rad/s)
    Kp_lambda_R  =  0.005; Ki_lambda_R =  0.008
    Kp_zeta_R    = -0.04;  Ki_zeta_R   = -0.006
    Kp_phi_R_SP  =  0.01;  Ki_phi_R_SP =  0.02
    Kp_dela      = -0.5;   Ki_dela     = -0.5;   K_theta_br_dela = -0.5
    Kp_dele      = -0.2;   Ki_dele     = -0.6
    Kp_delr      =  0.5;   Ki_delr     =  0.5
    Kg           = 0.0*9.81 # Optional alpha suppression gain

    # Outer loop
    production_origin_E = [P_X, P_Y, P_Z]
    kite_position_E = [X, Y, Z]
    q_EP = compute_production_plane_quaternion(lambda_P, zeta_P) # Eq. (A8)
    x_P, y_P, _ = compute_production_plane_coordinates(kite_position_E, q_EP, production_origin_E) # Eqs. (A9-A10)
    x_P_error = 0.0 - x_P
    y_P_error = 0.0 - y_P
    lambda_R = Kp_lambda_R * x_P_error + Ki_lambda_R * i_x_P_error
    zeta_R = Kp_zeta_R * y_P_error + Ki_zeta_R * i_y_P_error
    radius_P = sqrt(x_P^2 + y_P^2)
    radius_P_error = radius_SP - filtered_radius_P
    phi_R_SP = Kp_phi_R_SP * radius_P_error + Ki_phi_R_SP * i_radius_P_error

    # Inner loop
    q_EB = euler_to_quaternion(phi, theta, psi) # Eq. (A1)
    phi_R, theta_R, _ = compute_reference_frame_attitude(lambda_R, zeta_R, q_EB) # Eqs. (A2-A7)
    phi_R_error = phi_R_SP - phi_R
    delta_a_SP = Kp_dela * phi_R_error + Ki_dela * i_phi_R_error + K_theta_br_dela * theta_R
    alpha_error = alpha_SP - alpha
    delta_e_SP = Kp_dele * alpha_error + Ki_dele * i_alpha_error + Kg*sind(theta)
```

```
    beta_error = beta_SP - beta
    delta_r_SP = Kp_delr * beta_error + Ki_delr * i_beta_error

    # State derivatives
    i_x_P_error_dot       = x_P_error
    i_y_P_error_dot       = y_P_error
    i_radius_P_error_dot  = radius_P_error
    i_phi_R_error_dot     = phi_R_error
    i_alpha_error_dot     = alpha_error
    i_beta_error_dot      = beta_error
    filtered_radius_P_dot = -1/tau_radius_p * (filtered_radius_P - radius_P)

    return (
        delta_a_SP, delta_e_SP, delta_r_SP,
        i_x_P_error_dot, i_y_P_error_dot, i_radius_P_error_dot,
        i_phi_R_error_dot, i_alpha_error_dot, i_beta_error_dot,
        filtered_radius_P_dot
    )
end
```

The quaternion operations presented in appendices A and B are calculated using the helper functions listed below.

```
# ---- HELPER FUNCTIONS ----

# Appendix A: Earth-to-reference-plane Euler angle transformation
function compute_reference_frame_attitude(lambda_R, zeta_R, q_EB)
    q_z = [cosd(zeta_R/2),        0,           0,                     sind(zeta_R/2)]
    q_y = [cosd((lambda_R+90)/2), 0,           sind((lambda_R+90)/2), 0             ]
    q_x = [cosd(180/2),           sind(180/2), 0,                     0             ]

    q_ER = T(q_z) * T(q_y) * q_x # Eq. (A2)
    q_BR = T(quaternion_inverse(q_EB)) * q_ER # Eq. (A3)
    R_BR = quaternion_to_rotation_matrix(q_BR) # Eq. (A6)

    # Eq. (A7)
    phi_R   =  atand(R_BR[2, 3], R_BR[3, 3])
    theta_R = -asind(R_BR[1, 3])
    psi_R   =  atand(R_BR[1, 2], R_BR[1, 1])
    return phi_R, theta_R, psi_R
```

```
end

# Appendix B: production-plane quaternion (Eq. (A8))
function compute_production_plane_quaternion(lambda_P, zeta_P)
    q_z  = [cosd(zeta_P/2),       0,           0,                     sind(zeta_P/2)]
    q_y  = [cosd((lambda_P+90)/2), 0,           sind((lambda_P+90)/2), 0             ]
    q_x  = [cosd(180/2),          sind(180/2), 0,                     0             ]
    q_EP = T(q_z) * T(q_y) * q_x
    return q_EP
end

# Appendix B: Earth-to-production-plane coordinates (Eqs. (A9–A10))
function compute_production_plane_coordinates(kite_position_E, q_EP, prod_origin)
    q_PE     = quaternion_inverse(q_EP)
    R_PE     = quaternion_to_rotation_matrix(q_PE)
    pos_in_P = R_PE * (kite_position_E - prod_origin)
    x_P      = pos_in_P[1]
    y_P      = pos_in_P[2]
    z_P      = pos_in_P[3]
    return x_P, y_P, z_P
end

# Quaternion primitives
# Convention: q_AB = quaternion rotating frame B to frame A, stored [w, x, y, z].
# T(q1)*q2 = q1 ⊗ q2 (Hamilton product, Egeland & Gravdahl 2002, p. 232).

function euler_to_quaternion(phi, theta, psi) # Eq. (A1)
    q_phi   = [cosd(phi   / 2), sind(phi   / 2), 0,               0              ]
    q_theta = [cosd(theta / 2), 0,               sind(theta / 2), 0              ]
    q_psi   = [cosd(psi   / 2), 0,               0,               sind(psi   / 2)]
    return T(q_psi) * T(q_theta) * q_phi
end

function S(w) # Eq. (A5)
    return [ 0.0   -w[3]   w[2];
             w[3]   0.0   -w[1];
            -w[2]   w[1]   0.0 ]
end

function T(q)
```

```
    η, e1, e2, e3 = q[1], q[2], q[3], q[4]
    return [ η   -e1  -e2  -e3;
             e1   η   -e3   e2;
             e2   e3   η   -e1;
             e3  -e2   e1   η  ]
end

function quaternion_inverse(q_AB)
    return [q_AB[1], -q_AB[2], -q_AB[3], -q_AB[4]]
end

# Eq. (A4): returns R_AB
# Maps vectors from frame B to frame A (Egeland & Gravdahl Eq. (6.215)).
function quaternion_to_rotation_matrix(q_AB)
    η = q_AB[1]
    ε = q_AB[2:4]
    return I(3) + 2η*S(ε) + 2*S(ε)*S(ε)
end
```

Appendix D: stability proof of the angle-of-attack-oscillation suppression controller

The differential equation for angle of attack (Eq. (12)) is reproduced below

$$\dot{\alpha} = -\frac{1}{2}\frac{\rho V S}{m} C_L - (p\cos\alpha + r\sin\alpha)\tan\beta + \frac{g}{V}\left(\frac{\cos\phi\cos\theta\cos\alpha + \sin\theta\sin\alpha}{\cos\beta}\right) + q \quad \text{(A11)}$$

where C_L is the lift coefficient. Presume that aerodynamic lift is within the linear region, the lift coefficient can then be decomposed into the zero-lift, angle-of-attack, and elevator components

$$C_L = C_{L_0} + C_{L_\alpha}\alpha + C_{L_{\delta_e}}\delta_e \quad \text{(A12)}$$

Substitute Eq. (A12) into (A11), assume that the sideslip β angle is close to zero, giving $\tan\beta = 0$ and $\cos\beta = 1$, we have

$$\dot{\alpha} = -\frac{1}{2}\frac{\rho V S}{m}\left(C_{L_0} + C_{L_\alpha}\alpha + C_{L_{\delta_e}}\delta_e\right) + \frac{g}{V}(\cos\phi\cos\theta\cos\alpha + \sin\theta\sin\alpha) + q \quad \text{(A13)}$$

Denote the error signal in the angle-of-attack feedback loop as $e = \alpha - \alpha_{SP}$ and the integrated error as $e_i = \int e\, dt$. If α_{SP} is static (i.e., a constant), $\dot{e} = \dot{\alpha}$. A Lyapunov function can be defined as

$$H = \frac{1}{2} k_i e_i^2 + \frac{1}{2} e^2 \tag{A14}$$

where k_i is a constant. Differentiating Eq. (A14) gives

$$\dot{H} = k_i e_i \dot{e}_i + e\dot{e} = k_i e_i e + e\dot{\alpha} \tag{A15}$$

Substituting Eq. (A13) into (A15)

$$\dot{H} = k_i e_i e + e\left[-\frac{1}{2}\frac{\rho V S}{m}\left(C_{L_0} + C_{L_\alpha}\alpha + C_{L_{\delta_e}}\delta_e\right) + \frac{g}{V}(\cos\phi\cos\theta\cos\alpha + \sin\theta\sin\alpha) + q\right] \tag{A16}$$

Equation (A16) can be rearranged into

$$\dot{H} = e\left(k_i e_i - \frac{1}{2}\frac{\rho V S}{m} C_{L_{\delta_e}}\delta_e + \frac{g}{V}\sin\theta\sin\alpha\right) - e\frac{1}{2}\frac{\rho V S}{m}\left(C_{L_0} + C_{L_\alpha}\alpha\right) + e\frac{g}{V}\cos\phi\cos\theta\cos\alpha + eq \tag{A17}$$

The first term in Eq. (A17) can be rewritten into $-k_e e^2$, where k_e is a constant, by defining the following feedback control law for the elevator (assuming no actuator dynamics, i.e., $\delta_e = \delta_{e_{SP}}$)

$$\delta_e = \frac{2m}{\rho V S C_{L_{\delta_e}}}\left(k_e e + k_i e_i + \frac{g}{V}\sin\theta\sin\alpha\right) \tag{A18}$$

the remaining terms of Eq. (A17) can be grouped into $eD(t)$ where $D(t)$ is the disturbance function

$$D(t) = -\frac{1}{2}\frac{\rho V S}{m}\left(C_{L_0} + C_{L_\alpha}\alpha\right) + \frac{g}{V}\cos\phi\cos\theta\cos\alpha + q \tag{A19}$$

this gives the Lyapunov derivative

$$\dot{H} = -k_e e^2 + eD(t) \tag{A20}$$

Each term in the disturbance function (A19) is physically bounded for all time, such that the disturbance becomes upper bounded as $|D(t)| \leq \overline{D}$ for some constant $\overline{D} \geq 0$. Applying Young's inequality to the cross-term yields

$$\dot{H} \leq -\frac{k_e}{2}e^2 + \frac{1}{2k_e}D(t)^2 \tag{A21}$$

which a standard ISS condition implying that the closed-loop system is input-to-state stable with respect to the disturbance $D(t)$ (see (Khalil, 2002) for more details).

The control law in Eq. (A18) can be simplified further based on the small angle of attack assumption

$$\delta_e = \frac{2m}{\rho V S C_{L_{\delta_e}}}\left(k_e e + k_i e_i + \frac{g}{V}\alpha \sin\theta\right) \tag{A22}$$

which can be rewritten as

$$\delta_e = -K_p e - K_i e_i + k_\theta \sin\theta \tag{A23}$$

with

$$K_p = \frac{2mk_e}{\rho V S C_{L_{\delta_e}}} \tag{A24}$$

$$K_i = \frac{2mk_i}{\rho V S C_{L_{\delta_e}}} \tag{A25}$$

$$K_\theta = \frac{2mg\bar{\alpha}}{\rho V^2 S C_{L_{\delta_e}}} \tag{A26}$$

where $\bar{\alpha}$ is an upper bound on the angle of attack.

Code and data availability

The simulation model is proprietary and is therefore not available for the public. The Julia code for the flight controller is provided in appendix C.

Author contributions

DN: funding acquisition, analysis, writing (original draft). APK: software, analysis, writing (review and editing). TV: software, resources. ML: funding acquisition, writing (review and editing). EO: analysis, software, resources, methodology, validation, writing (review and editing).

Competing interests

The authors declare that they have no conflict of interest.

Disclaimer

Copernicus Publications remains neutral with regard to jurisdictional claims made in the text, published maps, institutional affiliations, or any other geographical representation in this paper. While Copernicus Publications makes every effort to include appropriate place names, the final responsibility lies with the authors. Views expressed in the text are those of the authors and do not necessarily reflect the views of the publisher.

Financial support

Duc H. Nguyen and Mark H. Lowenberg are supported by the UK Engineering and Physical Sciences Research Council (EPSRC), grant number EP/Y014545/1. Agustí Porta Ko is supported by the AWETRAIN project, which receives funding from the European Union's Horizon Europe research and innovation programme under the Marie Skłodowska-Curie grant agreement No 101168734. Espen Oland is part of the 3D-CIRCULAR project, which has received funding from the European Health and Digital Executive Agency under the Grant Agreement No 101226256. The contents of this publication are the sole responsibility of the authors and do not necessarily reflect the opinion of the European Union.

References

Ahrens, U., Diehl, M., and Schmehl, R.: Airborne Wind Energy, Green Energy and Technology, Springer Berlin, Heidelberg, Germany, 10.1007/978-3-642-39965-7, 2013.

Basile, L., Berni, M. G., and Celani, A.: Harvesting energy from turbulent winds with reinforcement learning, Europhysics Letters, 152, 43001, doi: 10.1209/0295-5075/ae1ce9, 2025.

Berra, A. and Fagiano, L.: An optimal reeling control strategy for pumping airborne wind energy systems without wind speed feedback, 2021 European Control Conference (ECC), Online, 29 June-2 July, doi: 10.23919/ECC54610.2021.9655018, 2021.

Cherubini, A., Papini, A., Vertechy, R., and Fontana, M.: Airborne Wind Energy Systems: A review of the technologies, Renewable and Sustainable Energy Reviews, 51, 1461-1476, doi: 10.1016/j.rser.2015.07.053, 2015.

Cobb, M. K., Barton, K., Fathy, H., and Vermillion, C.: Iterative Learning-Based Path Optimization for Repetitive Path Planning, With Application to 3-D Crosswind Flight of Airborne Wind Energy Systems, IEEE Transactions on Control Systems Technology, 28, 1447-1459, doi: 10.1109/TCST.2019.2912345, 2020.

Consolini, L., Maggiore, M., Nielsen, C., and Tosques, M.: Path following for the PVTOL aircraft, Automatica, 46, 1284-1296, doi: 10.1016/j.automatica.2010.05.014, 2010.

DeLosRíos-Navarrete, F., González-García, J., Castro-Fernández, I., and Sánchez-Arriaga, G.: A small-scale and autonomous testbed for three-line delta kites applied to airborne wind energy, Wind Energy Science, 10, 1153-1166, doi: 10.5194/wes-10-1153-2025, 2025.

Dief, T. N., Fechner, U., Schmehl, R., Yoshida, S., and Rushdi, M. A.: Adaptive Flight Path Control of Airborne Wind Energy Systems, Energies, 13, 667, doi: 10.3390/en13030667, 2020.

Diehl, M.: Airborne Wind Energy: Basic Concepts and Physical Foundations, in: Airborne Wind Energy, edited by: Ahrens, U., Diehl, M., and Schmehl, R., Springer Berlin Heidelberg, Berlin, Heidelberg, 3-22, 10.1007/978-3-642-39965-7_1, 2013.

Duda, D. F., Fuest, H., Islam, T., and Moormann, D.: Flight guidance concept for the launching and landing phase of a flying wing used in an airborne wind energy system, Wind Energy Science, 10, 661-678, doi: 10.5194/wes-10-661-2025, 2025.

Egeland, O. and Gravdahl, J.: Modeling and Simulation for Automatic Control, 2nd, Marine Cybernetics, Trondheim, Norway2003.

Fagiano, L., Zgraggen, A. U., Morari, M., and Khammash, M.: Automatic Crosswind Flight of Tethered Wings for Airborne Wind Energy: Modeling, Control Design, and Experimental Results, IEEE Transactions on Control Systems Technology, 22, 1433-1447, doi: 10.1109/TCST.2013.2279592, 2014.

Fagiano, L., Nguyen-Van, E., Rager, F., Schnez, S., and Ohler, C.: Autonomous Takeoff and Flight of a Tethered Aircraft for Airborne Wind Energy, IEEE Transactions on Control Systems Technology, 26, 151-166, doi: 10.1109/TCST.2017.2661825, 2018.

Fechner, U., van der Vlugt, R., Schreuder, E., and Schmehl, R.: Dynamic model of a pumping kite power system, Renewable Energy, 83, 705-716, doi: 10.1016/j.renene.2015.04.028, 2015.

Fernandes, M. C. R. M., Vinha, S., Paiva, L. T., and Fontes, F. A. C. C.: L0 and L1 Guidance and Path-Following Control for Airborne Wind Energy Systems, Energies, 15, 1390, doi: 10.3390/en15041390, 2022.

Freidovich, L., Robertsson, A., Shiriaev, A., and Johansson, R.: Periodic motions of the Pendubot via virtual holonomic constraints: Theory and experiments, Automatica, 44, 785-791, doi: 10.1016/j.automatica.2007.07.011, 2008.

Fritz, F.: Application of an Automated Kite System for Ship Propulsion and Power Generation, in: Airborne Wind Energy, edited by: Ahrens, U., Diehl, M., and Schmehl, R., Springer Berlin Heidelberg, Berlin, Heidelberg, 359-372, 10.1007/978-3-642-39965-7_20, 2013.

Hauser, J., Sastry, S., and Meyer, G.: Nonlinear control design for slightly non-minimum phase systems: Application to V/STOL aircraft, Automatica, 28, 665-679, doi: 10.1016/0005-1098(92)90029-F, 1992.

Heydarnia, O., Wauters, J., Lefebvre, T., and Crevecoeur, G.: Optimal Path Planning of Airborne Wind Energy Systems with a Flexible Tether, Journal of Guidance, Control, and Dynamics, 48, doi: 10.2514/1.G008967, 2025.

Jordan, D. W. and Smith, P.: Nonlinear Ordinary Differential Equations - An Introduction for Scientists and Engineers, Oxford University Press, New York2007.

Kelly, M.: From standard wind measurements to spectral characterization: turbulence length scale and distribution, Wind Energy Science, 3, 533-543, doi: 10.5194/wes-3-533-2018, 2018.

Khalil, H. K.: Nonlinear Systems, 3rd, Prentice Hall, New Jersey2002.

Khurshid, Y., Paul, P., Elhesasy, M., Ali, B., Kamra, M. M., Okasha, M., and Dief, T. N.: From inception to commercialization: A systematic review of airborne wind energy systems, Sustainable Energy Technologies and Assessments, 83, 104623, doi: 10.1016/j.seta.2025.104623, 2025.

Loyd, M. L.: Crosswind kite power (for large-scale wind power production), Journal of Energy, 4, 106-111, doi: 10.2514/3.48021, 1980.

Mann, J.: The spatial structure of neutral atmospheric surface-layer turbulence, Journal of Fluid Mechanics, 273, 141-168, doi: 10.1017/S0022112094001886, 1994.

Mohammadi, A., Maggiore, M., and Consolini, L.: Dynamic virtual holonomic constraints for stabilization of closed orbits in underactuated mechanical systems, Automatica, 94, 112-124, doi: 10.1016/j.automatica.2018.04.023, 2018.

Mohammed, T., Oland, E., and Fagiano, L.: Fault Tolerant Flight Control for the Traction Phase of Pumping Airborne Wind Energy Systems, International Journal of Control, Automation and Systems, doi: 10.1007/s12555-023-0588-z, 2024a.

Mohammed, T., Busk, J., Oland, E., and Fagiano, L.: Large-Scale Reverse Pumping for Rigid-Wing Airborne Wind Energy Systems, Journal of Guidance, Control, and Dynamics, 47, 1748-1758, doi: 10.2514/1.G007859, 2024b.

NASA: Experience with the X-15 Adaptive Flight Control System, National Aeronautics and Space Administraion, 1971.

Nguyen, D. H., Lowenberg, M. H., and Oland, E.: Improving Power Generation in Rigid-Wing Groundgen Airborne Wind Energy Systems Using Feedback Control—A Parametric Study, Wind Energy, 29, e70109, doi: 10.1002/we.70109, 2026a.

Nguyen, D. H., Lowenberg, M. H., and Oland, E.: Trimming a rigid-wing airborne wind system for coordinated circular flights, Wind Energy Science, 11, 285-298, doi: 10.5194/wes-11-285-2026, 2026b.

Orzan, N., Leone, C., Mazzolini, A., Oyero, J., and Celani, A.: Optimizing airborne wind energy with reinforcement learning, The European Physical Journal E, 46, 2, doi: 10.1140/epje/s10189-022-00259-2, 2023.

Pereira, A. F. C. and Sousa, J. M. M.: A Review on Crosswind Airborne Wind Energy Systems: Key Factors for a Design Choice, Energies, 16, 351, doi: 10.3390/en16010351, 2023.

Rackauckas, C. and Nie, Q.: DifferentialEquations.jl – A Performant and Feature-Rich Ecosystem for Solving Differential Equations in Julia, Journal of Open Research Software, doi: 10.5334/jors.151, 2017.

Rapp, S., Schmehl, R., Oland, E., and Haas, T.: Cascaded Pumping Cycle Control for Rigid Wing Airborne Wind Energy Systems, Journal of Guidance, Control, and Dynamics, 42, 2456-2473, doi: 10.2514/1.G004246, 2019.

Rezapour, E., Pettersen, K. Y., Liljebäck, P., Gravdahl, J. T., and Kelasidi, E.: Path following control of planar snake robots using virtual holonomic constraints: theory and experiments, Robotics and Biomimetics, 1, 3, doi: 10.1186/s40638-014-0003-6, 2014.

Schmehl, R.: Airborne Wind Energy - Advances in Technology Development and Research, Green Energy and Technology, Springer Singapore, Singapore, 10.1007/978-981-10-1947-0, 2018.

Shampine, L. F. and Reichelt, M. W.: The MATLAB ODE Suite, SIAM Journal on Scientific Computing, 18, 1-22, doi: 10.1137/s1064827594276424, 1997.

Shiriaev, A., Perram, J. W., and Canudas-de-Wit, C.: Constructive tool for orbital stabilization of underactuated nonlinear systems: virtual constraints approach, IEEE Transactions on Automatic Control, 50, 1164-1176, doi: 10.1109/TAC.2005.852568, 2005.

Taylor, G. I.: The Statistical Theory of Isotropic Turbulence, Journal of the Aeronautical Sciences, 4, 311-315, doi: 10.2514/8.419, 1937.

Trevisi, F.: Conceptual design of windplanes, Department of Aerospace Science and Technology, Polytechnic University of Milan, Milano, Italy, 2024.

Trevisi, F., Croce, A., and Riboldi, C. E. D.: Flight Stability of Rigid Wing Airborne Wind Energy Systems, Energies, 14, 7704, doi: 10.3390/en14227704, 2021.

Vermillion, C., Cobb, M., Fagiano, L., Leuthold, R., Diehl, M., Smith, R. S., Wood, T. A., Rapp, S., Schmehl, R., Olinger, D., and Demetriou, M.: Electricity in the air: Insights from two decades of advanced control research and experimental flight testing of airborne wind energy systems, Annual Reviews in Control, 52, 330-357, doi: 10.1016/j.arcontrol.2021.03.002, 2021.

Vinha, S., Fernandes, G. M., Fernandes, M. C. R. M., and Fontes, F. A. C. C.: Motion Primitives on a Spherical Surface with Application to Tethered Aircraft Guidance, 2025 IEEE 19th International Conference on Control & Automation (ICCA), Tallinn, Estonia, 30 June-3 July, 186-191, doi: 10.1109/ICCA65672.2025.11129856, 2025.

Vinha, S., Fernandes, G. M., Nguyen, H. T., Fernandes, M. C. R. M., and Fontes, F. A. C. C.: Automatic Circular Take-off and Landing of Motorized Tethered Aircraft, 2024 European Control Conference (ECC), 25-28 June 2024, 681-686, doi: 10.23919/ECC64448.2024.10590893,

Zgraggen, A. U., Fagiano, L., and Morari, M.: Automatic Retraction and Full-Cycle Operation for a Class of Airborne Wind Energy Generators, IEEE Transactions on Control Systems Technology, 24, 594-608, doi: 10.1109/TCST.2015.2452230, 2016.