Enhancing resilience of airborne wind energy systems through upset condition avoidance

Rapp, Sebastian; Schmehl, Roland

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Enhancing Resilience of Airborne Wind Energy Systems Through Upset Condition Avoidance

Sebastian Rapp[*] and Roland Schmehl[†]

Delft University of Technology, 2629 HS Delft, The Netherlands

Airborne wind energy (AWE) systems are tethered flying devices that harvest wind resources at higher altitudes, which are not accessible to conventional wind turbines. To become a viable alternative to other renewable energy technologies, AWE systems are required to fly reliably and autonomously for long periods of time while being exposed to atmospheric turbulence and wind gusts. In this context, the present paper proposes a three-step methodology to improve the resilience of an existing baseline control system toward these environmental disturbances. In the first step, upset conditions are systematically generated that lead to a failure of the control system using the subset simulation method. In the second step, the generated conditions are used to synthesize a surrogate model that can be used to predict upsets beforehand. In the final step an avoidance maneuver is designed that keeps the AWE system operational while minimizing the impact of the maneuver on the average pumping cycle power. The feasibility of the methodology is demonstrated on the example of tether rupture during pumping cycle operation. As an additional contribution a novel transition strategy from retraction to traction phase is presented that can reduce the probability of tether rupture significantly.

I. Introduction

OPERATING airborne wind energy (AWE) systems requires sophisticated control strategies that try to exploit the full physical capabilities of the system for maximum power generation without compromising safety. The major part of the existing literature about AWE control systems focuses on the former, to maximize the power output using trajectory optimization (see, for instance, [1–4]). The only recent publication that analyses reliability and safety of AWE systems is [5], which presents a failure mode and effect analysis along with a fault tree analysis for a flexible wing kite power system. This imbalance between performance optimization and reliability analysis in the AWE literature indicates that more research is necessary to investigate how the resilience and robustness of AWE control systems can be improved, which motivates the present work.

AWE systems need to operate in varying environmental conditions such as slowly varying wind speeds due to the altitude dependent mean wind speed profile but also need to cope with rapid changes due to wind gusts and turbulence. Because of the inherent stochastic nature of the wind conditions, it is difficult to explicitly include them in the control design process. In practice, the closed-loop system is verified a posteriori for randomly generated wind conditions, as presented, for instance, in [6]. If the controller fails to satisfy all requirements, it needs to be either retuned or completely redesigned. To create enough confidence that the controller achieves its objective, a large amount of simulations is necessary. This approach belongs to the direct Monte Carlo simulation methods ([7] p. 83f). Besides the computational burden of the control system verification process, it is also difficult to create enough counterexamples where the control system fails. For example, a wind gust with a certain shape that occurs with a probability of 10^{-6} requires on average 10^{6} simulation runs until it is encountered once. Especially for computationally expensive simulation runs this approach can be practically infeasible if several samples of these rare events need to be generated. Naturally, the more information about the condition that leads to a control

system failure is available, the more reliably it can be predicted and prevented in the future. Concretely, if enough data about counterexamples are available, a model that runs in parallel to the control system can be constructed that monitors the current flight state. It can then be used to predict how likely it is that the current flight condition leads to an upset and if necessary triggers a maneuver that avoids it. Creating such a predictor requires a significant amount of data that cannot be generated efficiently using the direct Monte Carlo method due to the aforementioned computational burden. Therefore, a different approach is chosen in this work, which is based on subset simulations (SSs). It is an algorithm that has been developed originally to estimate small failure probabilities of high-dimensional stochastic systems [8]. Recently SS has already been applied to small failure probability estimation in the context of flight control system verification (see [9,10]). In the context of this work the algorithm will not only be used to estimate rare event probabilities but also to generate systematically a knowledge base for external disturbances that lead to a specified control system failure, denoted as an upset. The generated conditions will then be used to train a binary classifier that is either based on a fixed threshold prediction strategy or a support vector machine (SVM). These surrogate models are able to predict and eventually prevent the occurrence of a failure beforehand with the overall goal to improve the fail-operational characteristic of the AWE system.

The contribution of this work can be summarized as follows. First, a control system modification to a previously published work of the authors (see [6]) is proposed. It is shown that the modification reduces the probability of tether rupture significantly. Second, a generic framework is presented that systematically generates conditions in which the control system fails. Third, two different prediction strategies are presented that are either based on a simple threshold approach or a binary time-series classification technique to predict upset conditions. Fourth, a loss rate function is derived that allows to trade off the prediction performance with respect to the induced economic loss of false positives and false negatives. In the last part of this work, the framework is applied to predict and prevent tether rupture, a common failure scenario in the context of AWE. A tailored avoidance maneuver is proposed that prevents this specific upset and keeps the system operational.

Ultimately, the following research questions are answered: How can the transition from retraction to traction phase be shaped in order to damp tether tension peaks during the transients? How can upset conditions in the context of AWE be defined and systematically generated if the probability of encountering one per pumping cycle is low and how can they be predicted? Furthermore, how can the practical impact of different prediction strategies be used to measure

*Ph.D. Researcher, Faculty of Aerospace Engineering, Kluyverweg 1; s.rapp@tudelft.nl.

†Associate Professor, Faculty of Aerospace Engineering, Kluyverweg 1; r.schmehl@tudelft.nl.

classification performance beyond classical metrics? Finally, is it justifiable from an economic point of view to prevent these conditions if that comes at the cost of false positives or is it more reasonable to simply accept them?

To that end, the paper is structured as follows. In Sec. II the closed-loop system is presented including a brief description of the used models and controllers. In addition, a modification to the baseline control architecture is introduced. In Sec. III the novel framework is presented generically, and in Sec. IV it is applied to the specific case of predicting and preventing tether rupture during a pumping cycle. Finally, Sec. V concludes the paper.

II. Closed-Loop System Description

In this study a model of a generic AWE system operated in pumping cycle mode is used along a modified version of the control system presented in [6] in order to demonstrate the effectiveness of the proposed framework. In the first part of this section the main components of the model will be reviewed. In the second part the key elements of the controller will be presented with a focus on the modification with respect to the baseline architecture as presented in [6].

A. Aircraft, Ground Station, and Wind Models

The aircraft is modeled as a six-degree-of-freedom rigid body, and its geometric and aerodynamic properties are based on the values in [11,12]. The translational dynamics are given by

$$(\dot{v}_k^G)_B = -(\omega)_B^{OB} \times (v_k^G)_B + \frac{(F_{tot})_B}{m_a} \qquad (1)$$

with

$$(F_{tot})_B = (F_a)_B + (F_g)_B + (F_t)_B + (F_p)_B \qquad (2)$$

where $(\dot{v}_k^G)_B$, $(v_k^G)_B$, $(\omega)_B^{OB}$, and m_a represent the kinematic acceleration, the kinematic velocity, the rotational rate of the aircraft with respect to the north-east-down reference frame, and the total mass of the aircraft, respectively. The subscript B indicates vectors given in the conventional body-fixed frame of the aircraft as visualized in Fig. 1. The total force acting in the center of mass G of the aircraft consists of the resulting aerodynamic force $(F_a)_B$, the weight $(F_g)_B$, tether force $(F_t)_B$, as well as propulsion force $(F_p)_B$. The rotational dynamics are given by

$$(\dot{\omega})_B^{OB} = -J^{-1}\left((\omega)_B^{OB} \times J(\omega)_B^{OB} - (M_a)_B\right) \qquad (3)$$

where $(\dot{\omega})_B^{OB}$, J, and $(M_a)_B$ denote the rotational acceleration, inertia tensor, and resulting aerodynamic moment acting in the center of mass, respectively. Note that it is assumed that the tether is attached to
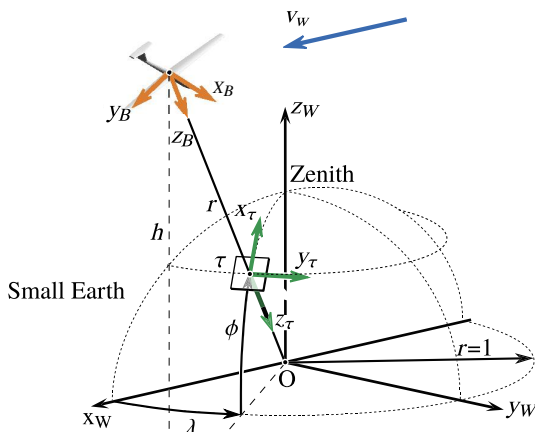

Fig. 1   Small Earth analogy with wind reference frame W, tangential plane frame τ, and body-fixed frame B [6].

Table 1   First-order actuator models

Parameter	Value	Unit
Bandwidth aileron $\omega_{a,0}$	35	$rad \cdot s^{-1}$
Aileron deflection limit $\delta_{a,lim}$	± 20	°
Aileron rate limit $\dot{\delta}_{a,lim}$	± 115	° s^{-1}
Bandwidth elevator $\omega_{e,0}$	35	$rad \cdot s^{-1}$
Elevator deflection limit $\delta_{e,lim}$	± 20	°
Elevator rate limit $\dot{\delta}_{e,lim}$	± 115	° s^{-1}
Bandwidth rudder $\omega_{r,0}$	35	$rad \cdot s^{-1}$
Rudder deflection limit $\delta_{r,lim}$	± 30	°
Rudder rate limit $\dot{\delta}_{r,lim}$	± 115	° s^{-1}

Table 2   Ground station parameters

Parameter	Value	Unit
Inertia J_W	0.08	$kg \cdot m^2$
Viscous friction κ_W	0.6	$kg \cdot m \cdot s^{-1}$
Acceleration limits $a_{W,min/max}$	± 5	$m \cdot s^{-2}$
Maximum speed $v_{W,max}$	20	$m \cdot s^{-1}$
Minimum speed $v_{W,min}$	-15	$m \cdot s^{-1}$

Table 3   Tether parameters

Parameter	Value	Unit
Particles n_T	5	— —
Mass density ρ_T	0.0046	$kg \cdot m^{-3}$
Diameter d_T	0.0025	m
Drag coefficient $C_{d,T}$	1.2	— —
Stiffness c_T	10,243	$N \cdot m^{-1}$
Damping d_T	7.8833	$N \cdot s \cdot m^{-1}$

the center of gravity because information about the exact location of the attachment point is not publicly available. Therefore, the tether does not contribute to the rotational dynamics. Furthermore, the actuator dynamics for ailerons, elevator, and rudder are approximated as first-order filters including rate and deflection limits. These values are summarized in Table 1. For the post-takeoff phase a simple propeller model is implemented as defined in [13] (p. 53f). Note that the propeller is used only in the beginning of each simulation to initialize the pumping cycle operation.

The ground station is modeled as in [6], and relevant parameter values are summarized in Table 2. Furthermore, the discretized tether model of [14] is implemented, and the used values are displayed in Table 3. Wind conditions are simulated using the wind shear model and the discrete Dryden turbulence model provided by the MATLAB Aerospace Toolbox [15]. The resulting mean wind speed profile in the present work as a function of altitude is depicted in Fig. 2. The turbulence components are superimposed to this wind speed profile.
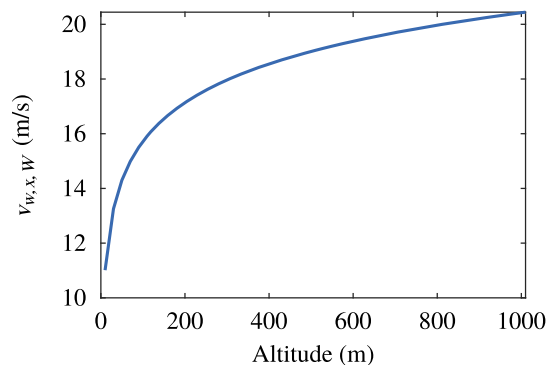

Fig. 2   Mean wind speed profile as a function of altitude.

B. Control System Description

In the following the control objective for an AWE system operated in pumping cycle mode is described. A pumping cycle usually results in a trajectory similar to the one displayed in Fig. 3. The control objective for such a system can be subdivided into a radial and a tangential direction control task. On the one hand, the controller needs to keep a high tension in the tether during the traction and a low tension during the retraction phase. This radial direction control objective is achieved using the rotational speed of the winch as well as the aircraft angle of attack and bank angle as control variables. On the other hand, the aircraft needs to follow a prescribed flight path, for instance, a figure-of-eight pattern during the traction phase and a straight-line glide path during the retraction phase. These two different flight path segments are indicated in Fig. 3 by the blue (traction) and orange (retraction) lines. The path following objective will be achieved using only the flight controller. In this work the flight controller is implemented in a cascaded form with three feedback loops. In the first loop the guidance commands in form of course and flight path angle rates are calculated with respect to the path curvature for the figure-of-eight flight, whereas during the retraction phase command shaping filters are used. These commands are then translated into bank angle and angle-of-attack commands that are tracked by the attitude controller. In the inner most loop the output of the attitude controller is translated into rate commands that are tracked by the rate controller and eventually allocated to the actuator deflections. Additionally, in each loop pseudocontrol hedging is implemented to comply with aircraft state and input limits.

In the following paragraphs two modifications to [6] are introduced that improve the robustness of the controller in turbulent wind conditions. The first modification consists of an improved guidance strategy for the transition between the retraction and the traction phase. The main challenge here is represented by the rising tether tension, i.e., from low tension in the retraction to high tension in the traction phase. Furthermore, because both phases are fundamentally different from a control perspective a transition strategy from straight-line following (retraction) to path following on a virtual sphere (traction) needs to be achieved. Subsequently, several possible solutions will be discussed before the final method is presented.

One possible approach is to directly switch into the figure-of-eight path-following mode as soon as the end of the retraction path is reached, which does not require any intermediate guidance strategy. Another option is to include a planar circular arc at the end of the retraction phase that defines the turning radius for the transition phase. This delays the activation of the traction mode until the aircraft is steered sufficiently back into the wind, which can be defined by a waypoint on the arc. The drawback of the first approach is the reduced level of guidance, and hence it is difficult to shape the transient behavior. Because the same controller for the transient as for the traction phase is used to avoid unnecessary switching between different controllers, tuning of the controller for better transient behavior would also alter the controller for the traction phase. The downside of the second approach is that it requires additional parameters to be tuned such as the length and curvature of the arc. Modifying the figure of eight would most

likely also require to modify the geometry of the transition arc. It can be seen that both approaches are complementary in terms of additional complexity and level of guidance. Note that more sophisticated approaches such as an optimal control strategy that connects the retraction and traction phase are also possible, but this requires again an additional controller switch, which is not desirable.

The advantages of both approaches can be combined in the following third alternative. Instead of defining a new arc in the horizontal plane, the same but rotated figure-of-eight curve as for the traction phase is used. This is similar to the first approach, where the traction phase is directly triggered at the end of the retraction phase. However, instead of directly approaching the traction phase path at a low elevation angle (power zone) a figure-of-eight curve at a high elevation angle (limit is 90 deg) is used for better guidance in the turning phase (advantage of the second approach). During the transient the curve is rotated toward the desired elevation angle for the traction phase. The time constant that defines the speed with which the path is rotated turns out to be an important parameter that trades of robustness (large value) and performance (small value) because it defines how fast the aircraft will fly into the power zone. In combination with a shaped set point change for the tether force tracking, a smooth transition from a straight path with low tether tension to figure-of-eight flight path following with high tether tension can be achieved. The mathematical implementation of this approach is discussed in the following.

As in [6] the figure-of-eight flight path is parameterized using the definition of a Lemniscate in spherical coordinates on a unit sphere. Concretely, the longitude and latitude of each point on the path are then given by

$$\lambda_p = \frac{b \sin(s)}{1 + [(a/b)\cos(s)]^2}$$
$$\phi_p = \frac{a \sin(s)\cos(s)}{1 + [(a/b)\cos(s)]^2} \tag{4}$$

where a and b define the specific shape of the path and $s \in [0, 2\pi]$ defines a specific position on the path. Transforming the path definition from spherical into Cartesian coordinates yields

$$(\boldsymbol{p})_P = \begin{pmatrix} \cos \lambda_p \cos \phi_p \\ \sin \lambda_p \cos \phi_p \\ \sin \phi_p \end{pmatrix} \tag{5}$$

where the subscript P denotes the path frame. It is essentially defined in the same way as the wind reference frame W (see [6], and Fig. 1) but is tilted by an angle ϕ_r around the negative y_W axis. The reference path in the W frame is then defined by

$$(\boldsymbol{p})_W = \begin{pmatrix} \cos \phi_r & 0 & -\sin \phi_r \\ 0 & 1 & 0 \\ \sin \phi_r & 0 & \cos \phi_r \end{pmatrix} (\boldsymbol{p})_P \tag{6}$$

Note that this redefinition of the path requires also a small modification in the algorithm in [6] that finds the closest point on the path using Newton's method with respect to the current position. In [6] the path is fixed at a certain elevation angle. However, because the rotation matrix in Eq. (6) is constant with respect to s the derivatives are not impacted and only the final result in [6] needs to be changed. Concretely, the target on the path as well as the tangent and its derivative with respect to s (see [6]) need to be rotated by ϕ_r using the same rotation matrix as used in Eq. (6).

The transient of the rotation angle ϕ_r is shaped using a first-order filter with bandwidth constant $\omega_{0,r}$ and set point ϕ_{set}, which corresponds to the reference elevation angle during the traction phase (see Fig. 4):

$$\dot{\phi}_r = \begin{cases} 0 & \text{if } \Delta_\phi > \bar{\Delta}_\phi \\ -\omega_{0,r}\phi_r + \omega_{0,r}\phi_{set}, & \phi_r(t=0) = \phi_0 \quad \text{else} \end{cases} \tag{7}$$
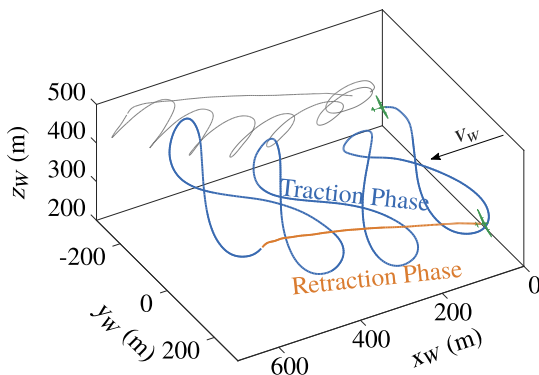

Fig. 3 Generic pumping cycle trajectory with traction and retraction phase.
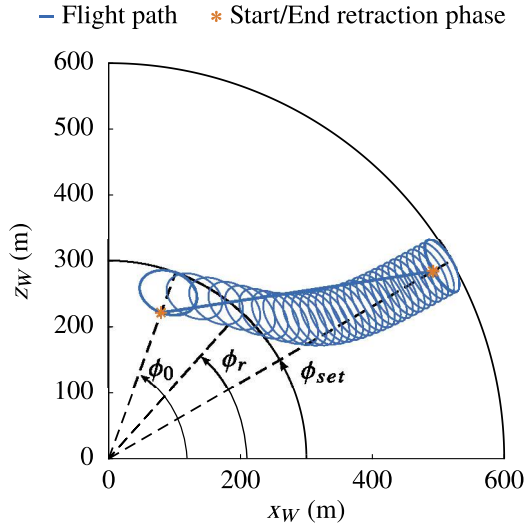
Fig. 4 Projection of the reference flight path in the xy plane of the W frame.

To avoid that the path is rotated too quickly $\dot{\phi}_r$ is set equal to zero as soon as the arc length on the unit sphere Δ_ϕ between the path and the current projected position of the aircraft exceeds a certain threshold $\bar{\Delta}_\phi$, which is set to one degree for the subsequent simulations. Δ_ϕ is given by

$$\Delta_\phi = \tilde{\phi}^G - \tilde{\phi}^t$$

$$\tilde{\phi}^G = \arccos\left(\frac{(\boldsymbol{p}_{\mathrm{xy}}^G)_W^\top (\boldsymbol{p}^G)_W}{\|(\boldsymbol{p}_{\mathrm{xy}}^G)_W\|_2 \|(\boldsymbol{p}^G)_W\|_2}\right)$$

$$\tilde{\phi}^t = \arccos\left(\frac{(\boldsymbol{p}_{\mathrm{xy}}^t)_W^\top (\boldsymbol{p}^t)_W}{\|(\boldsymbol{p}_{\mathrm{xy}}^t)_W\|_2 \|(\boldsymbol{p}^t)_W\|_2}\right) \qquad (8)$$

where $(\boldsymbol{p}_{\mathrm{xy}}^G)_W$ and $(\boldsymbol{p}_{\mathrm{xy}}^t)_W$ are the normal projections into the $x_W y_W$ plane of the aircraft position $(\boldsymbol{p}^G)_W$ and the target on the path $(\boldsymbol{p}^t)_W$, respectively. All vectors are given in the wind reference frame. The inverse of $\omega_{0,r}$ represents the time constant τ_r of the transition phase. It is a tuning parameter that defines how quickly the path is rotated into the power zone. In the limit, as τ_r goes to zero, the transition scenario without guidance is reached. On the contrary, for large time constants the aircraft will fly most of the time at high elevation angles, which will reduce the power output. Hence, the parameter value reflects the tradeoff between robustness (large τ_r) and maximum power output (small τ_r). The impact of the time constant on the robustness is addressed in more detail later in the paper. Note that the initial condition is usually chosen smaller than 90 deg (between 70 and 80 deg); otherwise, this would cause the aircraft to overfly the ground station. The filter is reset at the beginning of the transition phase.

For the retraction phase the straight glide path is defined as the connecting line of the point at which the retraction mode got triggered and a waypoint on the rotated reference path defined by $s = \{(\pi/2), (3\pi/2)\}$ and ϕ_0. The s value is chosen depending in which part of the figure of eight (positive or negative y_W coordinate) the traction phase got triggered. The retraction phase is triggered if two conditions are met. First, a specified tether length needs to be reached; second, the aircraft needs to pass the point on the path specified by $s_1 = (\pi/2)$ or $s_2 = (3\pi/2)$. As opposed to directly triggering the retraction phase if the maximum tether length is reached, this approach reduces the possible retraction points on the path to two, which is more convenient for robustness analysis. The downside of this approach is that the maximum length of the tether can vary in one pumping cycle because the increment in tether length per half figure of eight flight varies with the reeling speed. To mitigate this effect the increment in tether length for each half-figure-of-eight flight is predicted based on the previous increment. If the aircraft

reaches one of the two possible retraction points the increase in tether length until the other retraction point is reached will be estimated. If the estimated tether length is higher than the maximum allowable tether length the retraction phase will be already triggered at the current retraction point. This feature ensures that the maximum allowable tether length is never exceeded.

Finally, a minor modification of the winch controller is presented. In contrast to [6] the winch controller was simplified because the feed-forward part turned out to be too aggressive in highly turbulent wind conditions leading to instabilities due to the acceleration limits of the winch. Instead, a simple proportional-integral (PI) controller is implemented that calculates a reference torque based on the difference between the tether force set point and the measured tether force on the ground. Based on the reference torque the winch will adapt the reeling speed. This strategy works for traction and retraction phase and requires only different set points.

III. Upset Condition Generation, Prediction, and Avoidance Framework

The framework consists of three steps denoted with A, B, and C. The different steps can be designed to a large extent independently, which allows to improve the framework in the future in a modular manner. In step A (upset condition generation) the SS algorithm is used to systematically generate samples that lead to a specific upset condition. In step B (upset condition prediction) the prediction model is designed based on the created samples from step A in order to learn to distinguish between upset and nominal conditions. Finally, in step C (upset condition avoidance) the avoidance maneuver is designed. The complete framework is visualized in Fig. 5, where the highlighted rectangles enclose the tasks associated to every individual step. The three steps are discussed generically and in more detail in the subsequent paragraphs. In Sec. IV the framework is applied to generate, predict, and prevent tether rupture during pumping cycle operation.
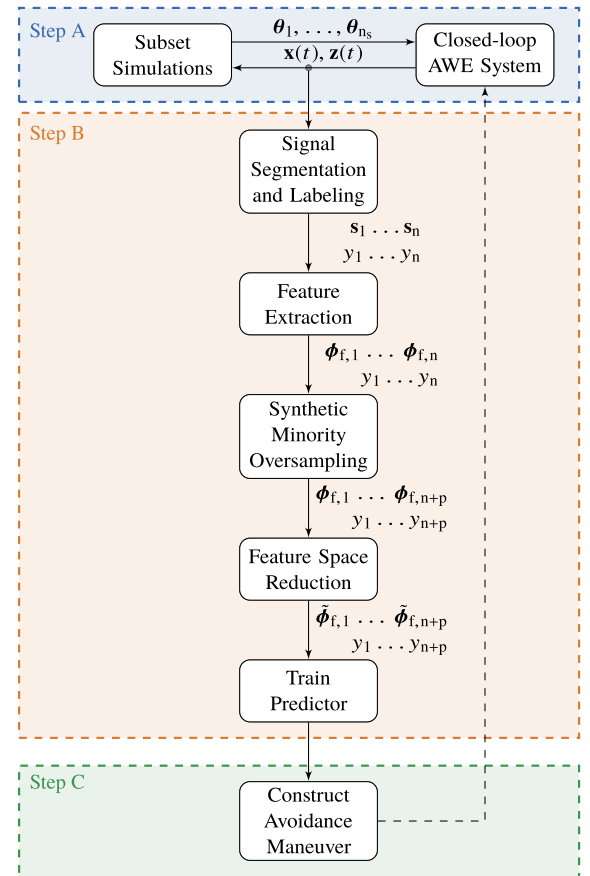

Fig. 5 Workflow of the proposed framework.

A. Upset Condition Generation

In this work upset conditions are generated using the SS algorithm. The introduction of SS in this section follows [16]. Further details about SS and mathematical proofs can also be found in [8]. In general, SS is a popular algorithm to estimate small event probabilities for high-dimensional systems [8]. An event, or failure, probability is a function of a multidimensional random variable Θ. As a function of its probability density function f_Θ a failure can be written as a multidimensional integral:

$$p_f = \int_\Theta \mathbb{1}_F(\boldsymbol{\theta}) f_\Theta(\boldsymbol{\theta}) \, d\boldsymbol{\theta} \qquad (9)$$

where Θ represents the entire space of $\boldsymbol{\theta}$, and $\mathbb{1}_F(\boldsymbol{\theta})$ is the indicator function that is either one if a certain realization $\boldsymbol{\theta}$ leads to a failure or zero otherwise. Furthermore, in the context of SS it is usually assumed that the random variables are identically and independently (iid) distributed; hence,

$$f_\Theta(\boldsymbol{\theta}) = \prod_{k=1}^{d} f_{\Theta_k}(\theta_k) \qquad (10)$$

In addition, it is assumed that the random variables are transformed such that the transformed variables are iid standard normal random variables with probability density function f'. Directly evaluating the integral in Eq. (9) analytically or even numerically is not feasible for complex high-dimensional systems due to the curse of dimensionality [8]. One approach to calculate this integral is using direct Monte Carlo methods that randomly sample from the parameter marginal distributions, evaluating the indicator function by simulation and using eventually the sample average to approximate the failure probability. If p_f is small this can require an unfeasible amount of simulation runs, which is especially critical if one simulation run is time-consuming, i.e., several minutes or more. Contrarily, in the context of SS, p_f is written as a product of conditional probabilities that involves the definition of intermediate failure domains. The main idea behind this strategy is that transitioning from one intermediate failure domain to the next has a higher chance than directly transitioning from nominal conditions into the failure domain. The failure probability can then equivalently be expressed as a product of conditional probabilities:

$$p_f = Pr(F_1) \prod_{i=1}^{m_s-1} Pr(F_{i+1}|F_i) \qquad (11)$$

The first intermediate failure probability $Pr(F_1)$ is obtained via a direct Monte Carlo approach where n_s samples are generated at random. Next, a limit function g that characterizes how close the current sample is to the failure is evaluated for each sample. The limit function is defined such that a higher value indicates a sample that is closer to the actual failure defined by g^*. The current intermediate failure domain is defined by a threshold $\mathcal{T}$ for the limit function that is in general smaller than the critical value g^*. Given the current sample set of size n_s, the threshold $\mathcal{T}$ is calculated by splitting the n_s samples into two subsets. One that contains the $n_s \cdot p_s$ samples with the highest limit function values and one that contains the remaining $n_s - n_s \cdot p_s$ samples. Then $\mathcal{T}$ is the average of the limit function values that separate the two sets. Concretely, if the threshold values of the limit function are arranged in descending order, then the intermediate threshold value is defined by

$$\mathcal{T} = \frac{g^{n_s \cdot p_s} + g^{n_s \cdot p_s+1}}{2} \qquad (12)$$

where $g^{n_s \cdot p_s}$ and $g^{n_s \cdot p_s+1}$ denote the $n_s \cdot p_s$th and $(n_s \cdot p_s + 1)$th largest samples with respect to their limit function values in the current sample set. In that case, the transition probabilities $Pr(F_{i+1}|F_i)$ are by definition equal to p_s, which is usually set to 0.1 [8]. To populate an intermediate failure domain with new

---

Algorithm 1:   Modified Metropolis algorithm

1: Pick $\boldsymbol{\theta} \in F_i$
2: **for** each coordinate $k = 1 \ldots d$ in $\boldsymbol{\theta}$ **do**
3:     Sample $\tilde{\theta}_k \sim \tilde{f}(\cdot|\theta_k)$
4:     Compute $\alpha = [f'(\tilde{\theta}_k)/f'(\theta_k)]$
5:     Accept $\tilde{\theta}_k$ if $\alpha > 1$ or if $\alpha > u$ with $u \sim \mathcal{U}(0, 1)$
6: **end for**
7: Accept $\tilde{\boldsymbol{\theta}}$ if $\tilde{\boldsymbol{\theta}} \in F_i$ o.w. set $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta}$
8: **return** $\tilde{\boldsymbol{\theta}}$

---

samples a Markov chain Monte Carlo method, such as the modified Metropolis algorithms (Algorithm 1), is used. The algorithm is briefly introduced in the following based on [8]. In the context of SS the task of the Metropolis algorithm is to populate an intermediate failure domain with samples that also belong to the current intermediate failure domain, i.e., $\tilde{\boldsymbol{\theta}} \in F_i$. This means that $\tilde{\boldsymbol{\theta}}$ leads to a limit function value that is larger than the current intermediate threshold $\mathcal{T}$. As soon as n_s samples are contained in the current domain F_i, the subsequent intermediate failure domain F_{i+1} will be defined. First, the new threshold $\mathcal{T}$ using Eq. (12) is calculated, and afterward new Markov chains are created to populate F_{i+1}. New samples, conditioned on an existing sample $\boldsymbol{\theta}$ in an intermediate failure domain F_i, are created by centering a symmetric proposal function $\tilde{f}$ around each coordinate θ_k of $\boldsymbol{\theta}$.

In this work a Gaussian proposal function is used. Its variance can be calculated adaptively as described in [17]. This results in $n_s \cdot p_s$ Markov chains with $(1/p_s) - 1$ elements. An accept/reject strategy, as defined in line 5 of the algorithm, leads to a nongreedy random walk around the previous state in the Markov chain. Because the intermediate thresholds are selected adaptively with respect to the most promising samples (higher limit function value) and new samples are only accepted if they are contained in the current intermediate failure domain (line 7), the algorithm will return at every stage inputs that drive the system more toward an upset condition (critical limit function value). This procedure is repeated until more than $n_s \cdot p_s$ samples lie in the actual failure domain. The actual failure probability can then be approximated by

$$p_f \approx p_s^{m_s-1} \frac{n_f}{n_s} \qquad (13)$$

where $n_f > n_s \cdot p_s$ is the number of samples that lie in the actual failure domain and m_s are the number of epochs in the SS run.

Applying the SS algorithm in the context of this work requires to define the upset conditions formally in form of a scalar limit function. All samples that lead to a limit function value that is beyond a defined threshold value are considered as upset conditions. The crucial part in modeling an upset condition is the allocation of the upset condition to a reasonable signal value or a combination of different signal values. For instance, if the analyzed upset is stall, the angle of attack represents the obvious choice as a limit function. Because this framework is mostly applicable to control system failure, finding the right limit function is usually done by taking the complement of the control objective. For instance, as described in Sec. II, the control objective for AWE systems operated in pumping cycle mode can be decomposed into a path-following problem (tangential direction control) and a tether force tracking problem (radial direction control). Hence, the limit function should be able to describe a failure in the tangential or radial direction control objective. The performance of the tangential direction control objective is reflected by the path-following tracking error, which suggests to choose this signal as a limit function to generate conditions in which the controller is not able to keep the aircraft close enough to the flight path. Similarly, in the radial direction the controller needs to track a high tension in the tether for maximum power production while keeping the tether force below the maximum tensile force that the aircraft and the tether itself can still support. An upset condition in this case can then be defined as a condition where the tension in the tether exceeds this critical

value. The latter example is investigated in depth in Sec. IV. Depending on the model fidelity, more complex upset conditions such as too high wing bending or vibrations with certain amplitudes in a certain frequency range can be analyzed, where the external excitation is generated using SS. Ultimately, a wide range of different upset conditions can be converted into a scalar function $g(\boldsymbol{\theta})$ with a threshold value beyond which the upset occurs. Note that the choice of this limit function is not limited to a specific functional form. It can be represented by an arbitrary nonlinear scalar function that just needs to be tailored to the specific upset condition. The only constraint is that the function needs to be monotonous such that maximizing the functional value indeed drives the system toward the considered upset condition. In most cases it is dependent on the aircraft states and outputs (e.g., angle of attack, airspeed, wing bending, tether tension). Having defined the limit function, the SS algorithm can be applied to sample $\boldsymbol{\theta}$s in order to drive g into the specific upset $g > g^*$.

B. Upset Condition Prediction

In this section two prediction approaches are presented. Because it is assumed that an upset can be defined by the value of the corresponding limit function, a first intuitive prediction approach is to predict an upset solely based on the current functional value of g. Because of the stochastic nature of the system the values of g will fluctuate according to the joint distribution of the uncertainties. Threshold values can then be selected based on the distribution of the maximum g values obtained from Monte Carlo simulations. For the sole purpose of classification it is obvious that selecting a threshold value arbitrarily close to the maximum limit function value will yield the highest prediction accuracy (least conservative). However, due to the inertia of the system as well as time delays this will in most cases not allow to avoid the upset condition. Contrarily, if the threshold value decreases, the false-positive rate will grow (more conservative). For this reason different threshold values need to be tested and a benchmark strategy as presented at the end of this section can be used to identify the best threshold.

In addition to the fixed threshold approach an alternative strategy based on binary time-series classification is proposed. The main motivation for this approach is that also the time history of certain states and outputs contains information that can be exploited for prediction. This approach is especially beneficial in the context of controls because critical disturbances often cause an oscillatory behavior before the actual upset. Obviously, oscillations can be detected only by analyzing the time and frequency content of a signal in a certain time window, which is not possible if the simple threshold approach is used. In this work the time-series classifier is realized as an SVM, which is optimized based on the generated samples in step A of the framework. In the following a concise description of the SVM algorithm is given, which is based on ([18] pp. 383–387). More details about SVMs can also be found in [19].

The goal of the SVM algorithm is to find a hyperplane for each class such that the margin between the two planes is maximized. The two spaces defined by the hyperplanes can be defined as

$$\boldsymbol{w}^\top \boldsymbol{\phi}_{f,i} + b \begin{cases} \geq 1 & \text{if } \boldsymbol{\phi}_{f,i} \text{ belongs to class 1} \\ \leq -1 & \text{otherwise} \end{cases} \quad (14)$$

where $\boldsymbol{w}$ is the normal vector of both hyperplanes and b is the bias term. The distance between the two hyperplanes is given by $2/\sqrt{\boldsymbol{w}^\top \boldsymbol{w}}$. To maximize the distance between the two planes the scalar product $\boldsymbol{w}^\top \boldsymbol{w}$ needs to be minimized, which leads to the following quadratic programming problem ([18] p. 384):

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2} \boldsymbol{w}^\top \boldsymbol{w} \\ \text{subject to} \quad & y_i(\boldsymbol{w}^\top \boldsymbol{\phi}_{f,i} + b) \geq 1, \ i = 1, \dots, n \end{aligned} \quad (15)$$

with $y_i \in \{-1, 1\}$. The optimization problem can be rewritten in terms of its Lagrangian as defined in [18] (p. 386). It will contain the input vector only as the scalar product $\boldsymbol{\phi}_{f,i}^\top \boldsymbol{\phi}_{f,i}$, which allows to

apply the kernel trick. The kernel function essentially maps the input parameter into a higher-dimensional space in which both classes are linearly separable ([18] p. 382). A common kernel is the radial basis function, or Gaussian kernel, which is also used in this work. Ultimately, the SVM is used to solve a binary classification problem where a given data set $\mathcal{D} = \{(\boldsymbol{\phi}_{f,i}, y_i), i = 1, \dots, n\}$ is used to construct a model that can predict if a certain input vector belongs to class -1 or 1. The predictor equation that is eventually added to the control system is given by

$$\hat{f} = \sum_{j=1}^{m} \alpha_j y_j e^{-[(\boldsymbol{\phi}_{f,i} - \boldsymbol{\phi}_j)^\top (\boldsymbol{\phi}_{f,i} - \boldsymbol{\phi}_j)/\sigma^2]} \quad (16)$$

where the α_j's are the m nonzero Lagrange multipliers of the corresponding support vectors $\boldsymbol{\phi}_j$ as well as their class labels y_j, and σ^2 is the variance of the Gaussian kernel, which is a hyperparameter that needs to be tuned. $\boldsymbol{\phi}_{f,i}$ corresponds to the current feature vector. The class label is then determined based on the condition

$$\hat{y} = \begin{cases} 1 & \text{if } \hat{f} \geq 0 \\ -1 & \text{else} \end{cases} \quad (17)$$

In this work an upset is defined by $y = -1$ and a nominal condition by $y = 1$. Estimated quantities are indicated by the "hat" operator.

The inputs to the SVM-based predictor will be specific estimations of aircraft states and wind conditions. Note that it will be assumed that the used signal values can be measured at a specific rate, and no state estimation is performed. The approach can, however, be extended by including a state estimator in between the predictor and the sensor outputs.

Instead of capturing the complete time history of each signal, specific signal statistics are extracted and collected in a finite-dimensional feature vector. Therefore, each signal is cut into smaller segments according to a chosen time window size. For instance, the highlighted green area in Fig. 6 indicates a time window with length 10 s of an arbitrary signal denoted here with z, which has a hypothetical maximum value $z_{\max}$ of 1.8 (orange, dashed line). At around 67 s the signal content between 57 and 67 s, denoted with s_1, is translated into a feature vector. To create the training examples the time window will be moved from either the final logged data point to the first data point, or if the complete signal contains an upset, i.e., $g(\boldsymbol{\theta}) > g^*$ (in this example, $g(\boldsymbol{\theta}) = z > z_{\max}$), the segmentation starts where the first upset occurs minus a shift ΔT_r as depicted in Fig. 6. The additional shift is required; otherwise, the predictor might fail to forecast an upset before its occurrence. The signal segmentation contains overlaps between the segments; hence the first time
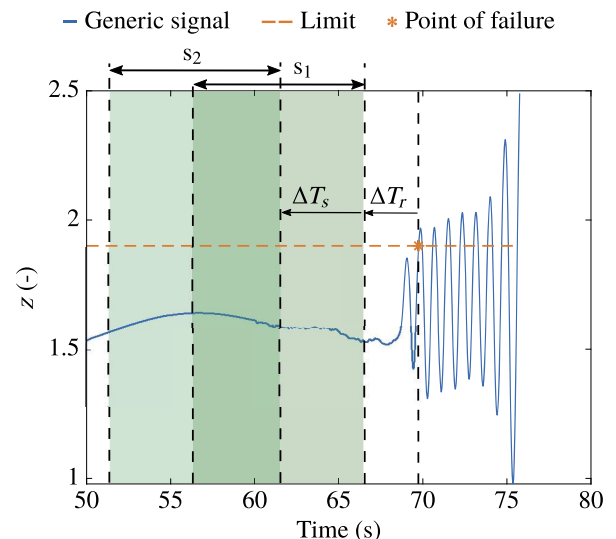

Fig. 6 Training example with reaction time definition for a hypothetical signal.

window is only shifted by ΔT_s and not by the window length. Note that only the first segment in Fig. 6 would be labeled as an upset, i.e., $y = -1$; the following segments starting with s_2 belong all to the nonupset class and are labeled with $y = 1$. Note that the time shifts are hyperparameters that need to be tuned to improve the classifier performance.

If a binary classifier is trained based on the generated data, the prediction accuracy can be improved by balancing the training data set. Although the SS algorithm will systematically generate upset conditions, the segmentation of the logged signals within a pumping cycle will always lead to more nonupset than upset conditions and hence to an extremely imbalanced data set. In fact, most of the simulated pumping cycles will not contain a single upset. One approach, which belongs to the data-level methods of learning from imbalanced data (see [20]), suggests to use a similar amount of samples from both classes. In this case randomly chosen nonupset samples are removed from the training data set (undersampling). This has the disadvantage that samples are thrown away. Another more sophisticated approach is to synthetically create more samples of the minority class. This can be achieved using the so-called Synthetic Minority Over-sampling Technique (SMOTE) algorithm (see [21]). The algorithm randomly picks a sample from the minority class, determines its k-nearest neighbors, picks one of the k neighbors at random, and interpolates again randomly between the two samples to synthesize a new minority class sample. Because the variances vary strongly between the different features the k-nearest neighbors are determined based on the Mahalanobis distance, which normalizes the Euclidean distance between two samples using the sample covariance matrix of the training set. This process is repeated until a specified amount of minority class samples has been created. Note that SMOTE can also be applied to the uncertainty vector $\boldsymbol{\theta}$. In that case the synthesized input vectors can be tested by simulation if they indeed lead to an upset and hence belong to the minority class. If a synthesized input vector is not leading to an upset, it is discarded. This is an advantage over the approach where SMOTE is used to synthesize new feature vectors. In this case it is not guaranteed with certainty that a new feature vector indeed belongs to the upset class. A drawback of applying SMOTE to the input vectors is that it requires significantly more time to create more samples of the minority class because every synthesized sample requires an additional simulation run. In this work SMOTE is applied directly to the feature space to save training time.

Based on the balanced training set a greedy forward feature selection algorithm as described in [22] is proposed to identify the most relevant features. The relevance of a feature is determined using 10-fold cross-validation, and as a metric the average Matthews correlation coefficient (MCC) is used to measure classification performance. The MCC is defined as

$$MCC = \frac{n_{TP} \cdot n_{TN} - n_{FP} \cdot n_{FN}}{\sqrt{(n_{TP} + n_{FP})(n_{TP} + n_{FN})(n_{TN} + n_{FP})(n_{TN} + n_{FN})}}$$
(18)

where n_{TP}, n_{TN}, n_{FP}, and n_{FN} denote the number of true positives, true negatives, false positives, and false negatives, respectively. The MCC is the preferred performance measure in binary classification problems because it condenses information of all four quadrants of the confusion matrix in one single number. This is not the case if other measures are used such as accuracy or F1 score, which is discussed in detail in [23]. Ultimately, each continuous time-series segment is condensed in an $\mathbb{R}^m$-dimensional vector $\tilde{\boldsymbol{\phi}}_{f,i}$, and the predictor is optimized based on the relationship

$$\begin{pmatrix} \tilde{\boldsymbol{\phi}}_{f,1} \\ \tilde{\boldsymbol{\phi}}_{f,2} \\ \vdots \\ \tilde{\boldsymbol{\phi}}_{f,n+p} \end{pmatrix} \rightarrow \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n+p} \end{pmatrix}$$
(19)

where $y_i \in \{-1, 1\}$. Note that the "tilde" operator indicates the reduced feature vector, n indicates the amount of samples generated

by the SS algorithm, and p is the amount of additionally synthesized samples using SMOTE. In this work the SVM is trained using the MATLAB Statistics and Machine Learning Toolbox [24].

In the previous paragraphs two prediction strategies are presented: on the one hand, a simple threshold-based predictor and, on the other hand, a time-series classification prediction strategy. One open question to be answered is how the performance of the prediction methods can be compared with each other in the context of upset condition prediction for an AWE system. Besides the classical metrics such as accuracy, F1 score, or MCC, it is beneficial to associate weights to false positives and false negatives that reflect the practical impact on the system performance. Because in practice false positives and false negatives have in general a different impact, ranking predictors simply based on their prediction accuracy is not a recommended approach. In the context of AWE a solution is to weight both terms proportionally to the resulting economic loss. In case of a false positive this loss equals the energy loss due to the triggered emergency maneuver $E_{FP} = E_{em}$. The loss stemming from a false-negative E_{FN} is more difficult to estimate because it requires a cost model that is able to predict the energy loss due to system downtime, repair costs, and material costs in case the upset damaged the system. To combine the impact of false negatives and false positives in a single number, an economic loss rate is introduced, which is defined as the weighted linear combination:

$$L = w_1 E_{FP} + w_2 E_{FN}$$
(20)

where E_{em} and E_{FN} are the associated energy losses in kWh due to false predictions. In this work, the weights w_1 and w_2 are derived based on the probabilities of false predictions. Mathematically, the occurrence of either an FP or an FN is modeled as a Poisson process. The Poisson process that models the arrivals of FPs runs until the first arrival time within the Poisson process that models the arrival of an FN. The expected value of the arrival time of an FN allows then to estimate the amount of FPs until that point in time and hence the resulting energy loss. The rate for the process that models the occurrence of an FN is given by

$$\begin{aligned} \lambda_{FN} &= Pr(\hat{y} = 1, y = -1) \\ &= Pr(\hat{y} = 1 | y = -1) Pr(y = -1) \\ &= \frac{n_{FN}}{n_{FN} + n_{TP}} p_f \end{aligned}$$
(21)

The conditional probability is simply given by the false-positive rate of the prediction strategy, the probability that $y = -1$ is the upset condition probability, which is independent of the prediction approach. Estimating the conditional probability is done by resimulating upset conditions for each of the different predictors using the results from the SS run in step A. Note that it is paramount here to use a different SS run from the one used to train the predictor. With the estimated FN rate the number of pumping cycles until the first expected FN occurs is then given by

$$n_{pc} = \frac{1}{\lambda_{FN}}$$
(22)

The expected number of FP up to the first FN is given by the expected value of the corresponding Poisson process defined by

$$\begin{aligned} n_{FP} &= \lambda_{FP} n_{pc} \\ &= Pr(\hat{y} = -1, y = 1) n_{pc} \end{aligned}$$
(23)

With the SVM-based predictor the probability of encountering a false-positive per pumping cycle can be estimated by counting falsely predicted upsets in a separate Monte Carlo simulation run. With a fixed threshold predictor this probability can be directly calculated using Eq. (24).

$$Pr(\hat{y} = -1, y = 1) = 1 - F_{\tilde{g}(\boldsymbol{\theta})}(q^*) - p_f$$
(24)

where q^* represents the chosen threshold value and $F_{\bar{g}(\theta)}$ is the cumulative distribution function of the maximum values of $g(\theta)$, which are calculated for each simulation run. Note that the threshold value represents a quantile of the distribution of $\bar{g}(\theta)$, and hence the false-positive probability is the corresponding area under the PDF right from the threshold minus the upset probability.

If an FN occurs, the system will not be operational for a specific amount of time ΔT_{nop}. It reflects the required time to conduct a possible emergency landing, maintenance, and relaunching. This mainly leads to a power loss in terms of missed pumping cycles. Assuming an average pumping cycle time of t_{pc}, the number of missed pumping cycles is

$$n_{\text{mpc}} = \frac{\Delta T_{\text{nop}}}{t_{\text{pc}}} \qquad (25)$$

The expected energy loss per pumping cycle due to predictions errors is eventually given by

$$
\begin{aligned}
L &= w_1 E_{\text{FP}} + w_2 E_{\text{FN}} \\
&= \frac{1}{n_{\text{pc}} + n_{\text{mpc}}} (n_{\text{FP}} P_{\text{em}} t_{\text{pc}} + n_{\text{mpc}} P_{\text{pc}} t_{\text{pc}} + E_{\text{misc}})
\end{aligned}
\qquad (26)
$$

This expression can be normalized by the average energy $E_{\text{pc}} = P_{\text{pc}} t_{\text{pc}}$ converted in one pumping cycle, which yields

$$\frac{L}{E_{\text{pc}}} = \frac{1}{n_{\text{pc}} + n_{\text{mpc}}} \left(n_{\text{FP}} \frac{P_{\text{em}}}{P_{\text{pc}}} + n_{\text{mpc}} + \frac{E_{\text{misc}}}{E_{\text{pc}}} \right) \qquad (27)$$

E_{misc} combines all additional losses involved with an FN such as replacement costs of damaged parts. Equation (27) allows to rank different predictors with respect to their expected energy loss relative to the average converted energy in one pumping cycle. This metric is better suited to assess prediction performance because it associates weights to false positives and negatives that have a practical meaning. This is not the case if standard metrics for prediction performance are used. Note that, at this stage, only guesses about the average downtime ΔT_{nop} as well as the additional involved costs, summarized in E_{misc}, can be made. Furthermore, due to the lack of a comprehensive cost model for AWE systems, Eq. (27) is only an approximation of the monetary loss that might be encountered in reality. In the future, and as soon as more data become available, a more accurate cost model should replace the simple model defined in Eq. (27).

Moreover, note that the features used by the SVM-based predictor are selected with respect to the achieved MCC. One could also directly choose Eq. (27) to rank the performance of feature combinations. However, this requires to rank the features as a function of the parameter values in the loss function for which only rough estimations are available at the moment. For this reason the MCC is used to optimize the SVM, and Eq. (27) is only used to compare different prediction strategies after the design phase. Based on these results the best predictor can be chosen and deployed on the real system.

C. Upset Condition Avoidance

In step C of the framework the avoidance maneuver is defined. Because of the possibility of false positives it is desirable that the impact of the maneuver on the pumping cycle operation is minimized. One generic approach for upset avoidance during the pumping cycle is to abort the current traction or retraction phase and use the onboard propulsion system to either land the aircraft or to go into a loiter mode from which the normal operation can again be initiated. Both approaches, however, reduce the average power output of the system significantly. A more efficient upset avoidance strategy needs to be tailored to the upset condition itself, which is demonstrated in the next section for the case of tether rupture.

IV. Application of the Framework to Generate, Predict, and Avoid Tether Rupture

A. Setup

In this section the three steps, A, B, and C, of the framework are applied to the case of tether rupture, which is an important upset condition in the field of AWE. Because of the complex interaction between ground station and flight control system, wind, tether, as well as the aircraft dynamics, it is basically impossible to analytically derive conditions that lead to this critical event. Furthermore, assuming that a reliable control system is implemented tether rupture has a low probability of occurrence, which makes it a suitable example to demonstrate the methodology proposed in this work. Additionally, because this event has a high relevance for the AWE community, the cause for tether rupture based on the obtained results is investigated in depths.

In step A the SS algorithm is used to generate systematically conditions that drive the tether force peak within a pumping cycle beyond its maximum allowable value. The limit function is in this case given by

$$g(\theta) = F_t \qquad (28)$$

In the present example the stochastic excitation is limited to the uncertainties in the wind conditions. It is arguably also the highest uncertainty that makes AWE systems difficult to control. Of course, the framework can be easily extended in order to include model parameter uncertainties, sensor noise, or hardware failures as well, but this is left for future work. The wind conditions in the simulations are generated using the Dryden turbulence model that has as input standard Gaussian-distributed random variables θ_k that are filtered to recover the Dryden turbulence spectrum. In total $d = T_{\text{sim}} f_s$ random variables are sampled per run, where T_{sim} is the simulation run time and f_s the sampling frequency, which is set to 10 Hz. Further possible variations in the wind field such as discrete gusts or changes in the wind speed profile and the mean wind direction are not considered in this work and are also left for future research. In general, upset conditions for a complete pumping cycle, or even several pumping cycles in a row can be generated with the proposed framework. However, because the dimension of the joint probability density function from which the wind condition is sampled grows linearly with simulation time, all the results are generated for only one pumping cycle per sample. For this specific example the SS algorithm created 4221 tether ruptures, and around $7.1 \cdot 10^5$ segments without tether rupture are extracted. The results are created with one SS run that included in total $3 \cdot 10^4$ pumping cycle simulations. The selected time window size for one segment is 5 s and the reaction time shift ΔT_r is set to 0.2 s.

In step B reasonable state and output variables are selected to predict the upset, and the predictor is designed based on the results of step A. In this example, only signals that are available at the aircraft are considered in order to avoid communication delays between the ground station and the aircraft. Concretely, the following signals are chosen: 1) wind speed components $v_{w,x,W}$, $v_{w,y,W}$, and $v_{w,z,W}$; 2) aircraft acceleration in radial direction $a_{z,\tau}$; 3) tether force F_t; 4) angle of attack α; and 5) path following error e_p.

Following Sec. III.B, each signal is discretized into smaller overlapping time windows, and statistical properties in the time and frequency domain are calculated. The used features that are calculated for each of the signals in the time domain are mean, median, root mean square (rms) value, variance, maximum, minimum, maximum peak-to-peak ratio, skewness, kurtosis, crest factor, median absolute deviation, range of the cumulative sum, the time-reversal asymmetry statistic given by Eq. (29), and the maximum signal slope. The time-reversal asymmetry statistic is defined by [22]

$$p = \frac{\mathbb{E}((s(t) - s(t - \tau))^3)}{(\mathbb{E}((s(t) - s(t - \tau))^2))^{(3/2)}} \qquad (29)$$

where different values for τ are considered. In the frequency domain the following characteristics are calculated: median and maximum

Table 4 Ordered feature list

Feature	MCC
Crest factor F_t	9.60
Time-reversal asymmetry statistic for the path tracking error e_p Eq. (29) with $\tau = 1$ s	9.83
Maximum F_t slope	9.92
Mean of F_t	9.935
Maximum amplitude above 1 Hz of α	9.951
Median amplitude of F_t	9.957
Minimum α	9.959
Variance $a_{z,\tau}$	9.965
Variance of F_t	9.965

amplitude, and additionally the maximum amplitude above 1 Hz using a fast Fourier transform. Note that this set of features is created heuristically. The individual features are chosen because they are computationally cheap to evaluate and easy to comprehend or because they turned out to be useful features in other applications (e.g., the time-reversal statistic in [22]). It will be shown later in the paper that an optimized subset, which is derived from the initial feature pool, leads to an acceptable prediction performance. Note that it is always possible to add more features to the initial feature pool, for instance, if the initial prediction performance is poor. However, the larger the initial feature pool, the longer it takes to optimize the smaller subset. Therefore, it is recommended to only gradually increase the size of the feature pool and always check if an acceptable classification performance can be achieved before new features are added.

To balance the data set additional feature vectors are created using the SMOTE algorithm and afterward the feature selection algorithm is applied to reduce the dimension of the original feature space. With SMOTE around $6 \cdot 10^5$ feature vectors are synthesized from the 4221 original samples created by the SS algorithm in step A. Note that in contrast to the original feature vectors it cannot be guaranteed that the synthesized feature vectors indeed belong to the set of feature vectors that lead to tether rupture. One reason is that the set corresponding to a tether rupture is not necessarily convex. Hence, an interpolated feature vector can also end up outside the nonconvex set. The optimized subset of features is displayed in Table 4 (ordered according to significance). Note that the MCC value in the second column is the cumulative MCC value. In Fig. 7 the convergence of the selection process is displayed. Convergence is defined as the point where the relative change in the MCC after adding a feature to the list is smaller than 10^{-4}. This convergence criterion is also proposed in [22]. With the optimized feature list the SVM predictor is trained as explained in Sec. III.B. Additionally, fixed thresholds are selected based on the estimated distribution of the tether force peaks using the results of the first stage of the SS run (direct Monte Carlo run). In this case the thresholds are selected with respect to the tether force set point in the traction phase. Concretely, the thresholds $F_{t,\text{set}} + 8\%$, $F_{t,\text{set}} + 10\%$,
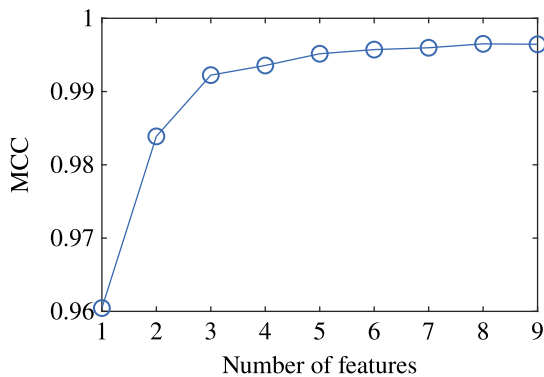
Fig. 7 Maximization of MCC using greedy feature selection.

$F_{t,\text{set}} + 12\%$, $F_{t,\text{set}} + 14\%$, and $F_{t,\text{set}} + 16\%$ are considered, which are all larger than the 0.99-quantile of the tether force peak distribution, which corresponds to $F_{t,\text{set}} + 7\%$. The set point $F_{t,\text{set}}$ itself is chosen to be -20% of the maximum allowable tether tension, which is set to 2 kN.

In step C of the framework the avoidance maneuver is designed. In case of a predicted tether rupture the contingency maneuver must reduce the current tension in the tether as quickly as possible. It turns out that with the underlying control system this can be achieved with a set point change for the tether force. On the one hand, the tension in the tether is tracked by the winch controller via the reeling out/in speed and, on the other hand, by the flight path controller through the angle of attack and bank angle. Therefore, changing the set point for the tether force leads to an adaptation of the winch reeling speed but also of the angle-of-attack and bank angle commands α_{set} and $\mu_{a,\text{set}}$, respectively. As derived in [6] both attitude commands are determined by inverting the flight path dynamics, which yields

$$f_{y,m} = m_a \nu_{\chi_k} \cos\gamma_k v_k - f_{t,y,K}$$
$$f_{z,m} = m_a \nu_{\gamma_k} v_k + \cos\gamma_k m_a g + f_{t,z,K} \quad (30)$$

$$\mu_{a,\text{set}} \approx \mu_{k,\text{set}} = \arctan\left(\frac{f_{y,m}}{f_{z,m}}\right)$$

$$C_{L,\text{set}}(\alpha) = \frac{\sqrt{f_{y,m}^2 + f_{z,m}^2}}{0.5\rho v_a^2 S_w}$$

$$\alpha_{\text{set}} = C_{L,\text{set}}^{-1}(\dots) \quad (31)$$

where the tether force set point components $f_{t,y,K}$ and $f_{t,z,K}$ are obtained by

$$\begin{pmatrix} f_{t,x,K} \\ f_{t,y,K} \\ f_{t,z,K} \end{pmatrix} = -\mathbf{M}_{\text{KO}}(\chi_k, \gamma_k) \frac{(\mathbf{p}^G)_O}{\|(\mathbf{p}^G)_O\|_2} F_{t,\text{set}} \quad (32)$$

$(\mathbf{p}^G)_O$ is the position of the aircraft in the north-east-down frame O and $\mathbf{M}_{\text{KO}}(\chi_k, \gamma_k)$ transforms a vector from the O frame into the kinematic frame K (see [6]). Essentially, Eqs. (30–32) calculate the angle-of-attack and bank angle commands based on the desired path curvature represented by the pseudocontrol inputs ν_{χ_k} and ν_{γ_k} as well as the tether force set point $F_{t,\text{set}}$. Note that for consistency the kinematic bank angle should in general be converted into the aerodynamic bank angle (banking around the aerodynamic instead of the kinematic velocity vector), but the effect is negligible here, which leads to the approximation $\mu_{a,\text{set}} \approx \mu_{k,\text{set}}$. The angle-of-attack command reflects the required lift magnitude, which is estimated by the required forces $f_{y,m}$, $f_{z,m}$ and involves the inversion of the lift coefficient as shown in Eq. (31). Furthermore, χ_k is the aircraft course angle, γ_k is the flight path angle, g denotes gravity, v_a is the airspeed, ρ is the air density, and S_w is the wing reference area. Note that, in order to track the tether force, the angle-of-attack and bank angle commands are calculated using the tether force set point $F_{t,\text{set}}$ and not the measured tether force currently acting on the aircraft. During nominal operation this allows to effectively keep the tether under the desired tension. As soon as an upset is predicted this set point will be reduced to a low value ($F_{t,\text{set}} = 10$ N). As a result the aircraft will correct the current bank and angle-of-attack commands accordingly. This allows to reliably reduce the tension in the tether quickly even if the winch is currently saturating, which is discussed in the next section. As soon as the tether force drops below a second threshold value, i.e., $F_t \leq c \cdot 10$ N, with, for instance, $c = 1.2$ and the predictor output switches from $\hat{y} = -1$ (upset) back to $\hat{y} = 1$ (no upset) the force set point is increased again to the original traction phase set point. The set point change is shaped smoothly using a first-order filter.

B. Results

The introduced control system modification in Sec. II has an important impact on the overall probability of tether rupture and on the power output, which is why it is also included in this paper. Concretely, the choice of the bandwidth $\omega_{0,r}$ that defines how quickly the path is rotated from higher to lower elevation angles during the transition from retraction to traction phase allows to trade off robustness against performance. In Fig. 8 the results of three independent SSs are presented with three different choices for $\omega_{0,r}$. As a reference $\omega_{0,r} = 0.05$ is chosen, which leads to an approximate tether rupture probability during a pumping cycle of $p_f \approx 2 \cdot 10^{-7}$. Increasing the reference value by a factor of 1.5 and 2 increases the power output by 28 and 33% but also leads to a significant increase of the tether rupture probability by a factor of approximately $1.9 \cdot 10^3$ and $1.3 \cdot 10^4$, respectively. Because there is no external standard that defines the allowable tether rupture probability, the conservative value of $\omega_{0,r} = 0.05$ is chosen to generate the subsequent results. The corresponding low probability also justifies the use of SS to generate this type of upset condition in the first place, whereas the other two controller settings defined by $1.5\omega_{0,r}$ and $2\omega_{0,r}$ lead to tether rupture probabilities that might be analyzed with simple Monte Carlo simulations. Given a desired level of reliability, $\omega_{0,r}$ can be adapted in the future accordingly.

The performance of the different prediction strategies (SVM and thresholds) is tested on a separately generated data set that is not used to construct the predictors. The test data set is generated in the same manner as the training set using the SS algorithm. In the first part of this section the effectiveness of the avoidance maneuver is analyzed. Subsequently, the prediction and prevention performance among the different predictors is assessed using Eq. (27).

The effectiveness of the prediction and avoidance strategy is demonstrated and explained in detail using the results of one sample of the test set that contains a tether rupture. To that end, the same simulation is carried out twice once with prediction and avoidance method and once without. To limit the scope of the result section, only the results using the SVM prediction strategy are displayed and analyzed. The resulting flight path of both scenarios is displayed in Fig. 9, and the corresponding projection in the $x_W y_W$ plane is depicted in Fig. 10. The blue path shows one complete pumping cycle where the tether rupture is prevented using the proposed avoidance maneuver. It can be observed that the system is able to continue its operation and the avoidance maneuver has no visible impact on the path following performance. The green cross indicates the point where the tether breaks, and as a result, in the scenario without avoidance strategy, the aircraft is ejected from the reference flight path and is no longer able to continue the pumping cycle. The evolution of the tether force in both scenarios is depicted in Fig. 11. At around 69 s the avoidance maneuver is triggered, which leads to a significant tether tension reduction as indicated by the blue solid line. In contrast to that, without the avoidance maneuver the tether tension continues to oscillate and at around 70.5 s the tether breaks.

The control system performance during the avoidance maneuver is analyzed more in detail using the evolution of the aerodynamic bank
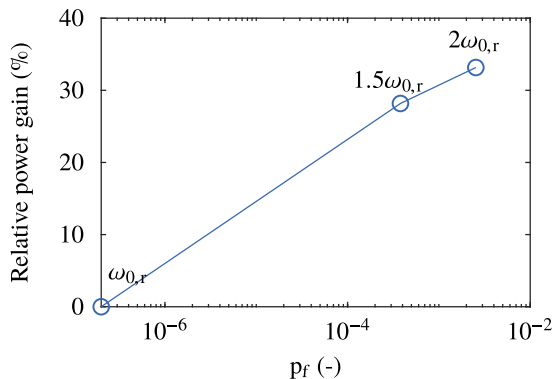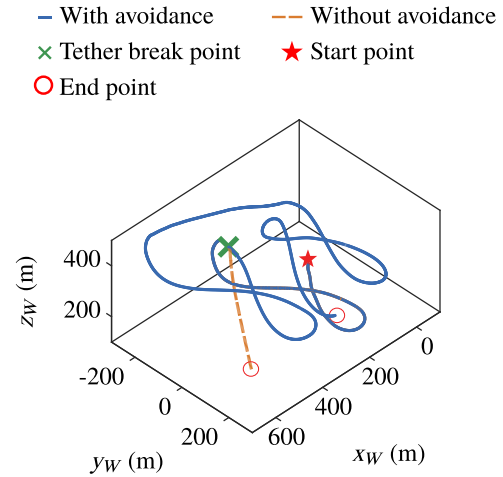

Fig. 9 Three-dimensional flight paths with and without avoidance strategy.
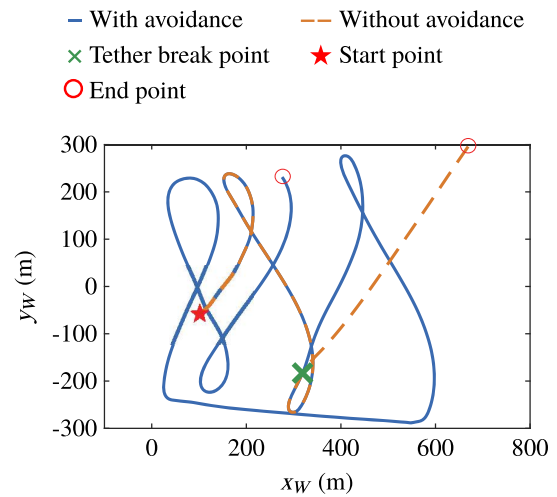

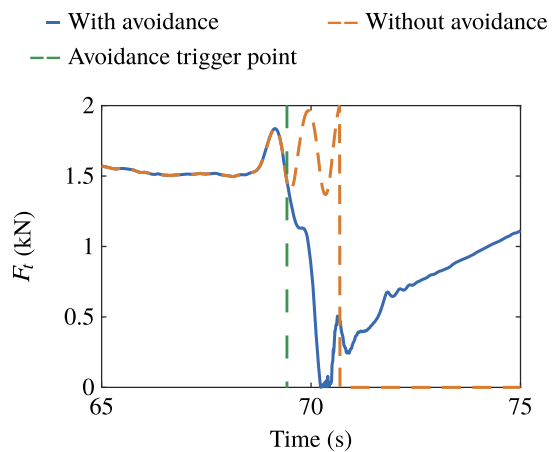Fig. 10 Projected flight paths with and without avoidance strategy.

Fig. 11 Evolution of the tether tension with and without prediction and avoidance strategy.

angle and the angle of attack. In Fig. 12 the impact of the tether tension set point change is clearly visible in the evolution of the bank angle set point $\mu_{a,\text{set}}$ (blue, solid line). At around 69 s the set point drops to around $-30°$. Because the controller uses a dynamic inversion-based control strategy the set point is filtered (orange, dashed line) and the actual bank angle is controlled such that it follows the corresponding reference model (green, dotted line). A similar
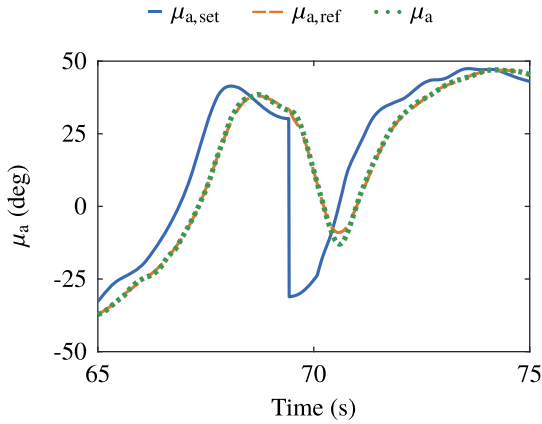

Fig. 8 Relative gain in pumping cycle power over tether rupture probability as a function of multiples of the path rotation constant $\omega_{0,r}$.

Fig. 12 Evolution of the aerodynamic bank angle. Set point $\mu_{a,set}$, reference $\mu_{a,ref}$, and achieved bank angle μ_a.

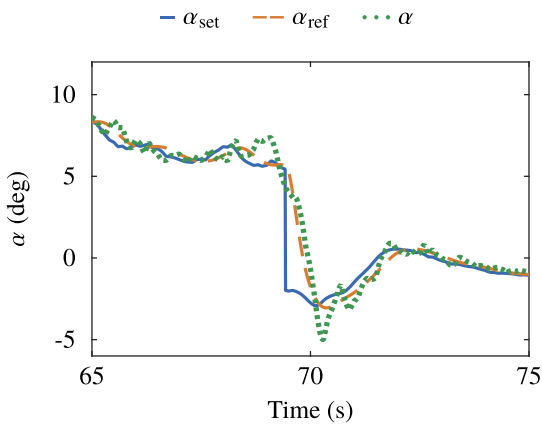**Fig. 13 Evolution of the angle of attack with set point α_{set}, reference filter state α_{ref}, and achieved angle of attack α.**

Fig. 14 Roll angle with and without avoidance maneuver.

Fig. 15 Pitch angle with and without avoidance maneuver.

behavior results for the angle of attack α_{set}, (see Fig. 13). Also in this case the tether tension set point change leads to a drop in the angle-of-attack set point. The actual angle of attack follows the corresponding reference model with an overshoot of approximately 2.3°.

The adaption of the bank angle and the angle of attack leads to an adaption of the aircraft attitude with respect to the tangential plane. Therefore, besides the change in lift magnitude through the adaption of the angle of attack also the rotation of the lift force leads to a tether force reduction. If the aircraft is flying in the tangential plane most of the lift force is pointing in the radial direction. Increasing the attitude angles (absolute value) with respect to the tangential plane by a simultaneous roll and pitch maneuver can reduce the tension in the tether because the component of the lift vector perpendicular to the tether direction increases. This behavior can be observed in Figs. 14 and 15. Both plots demonstrate that due to the tether force set point change the aircraft is indeed rotated into the tangential plane (blue, solid line). The roll angle Φ_τ is reduced from a nearly horizontal attitude (with respect to the tangential plane) to $-50°$, and the pitch angle Θ_τ is reduced from around $-5°$ to $-25°$. Without the avoidance maneuver (orange, dashed line) the roll angle stays nearly constant and the pitch angle starts to oscillate and to increase. The green dotted line indicates the point where the tether breaks. For the case with tether rupture avoidance the resulting trajectory is again displayed in Fig. 16. The blue line represents the flight path of one pumping cycle, of which only the part around the prevented tether rupture is displayed in Fig. 17. The tether is shown as a solid gray line connecting the aircraft with the ground station. Additionally, a simple aircraft visualization (colored rectangle) is added to the figure, which represents the orientation of the aircraft wing. The aircraft visualization color changes from green to orange as soon as the avoidance maneuver is triggered. The resulting attitude change is visible in the
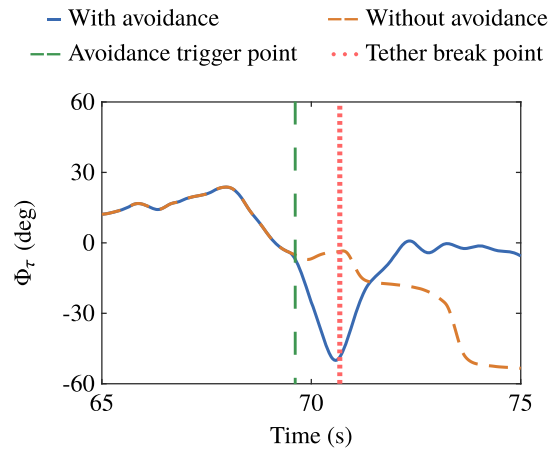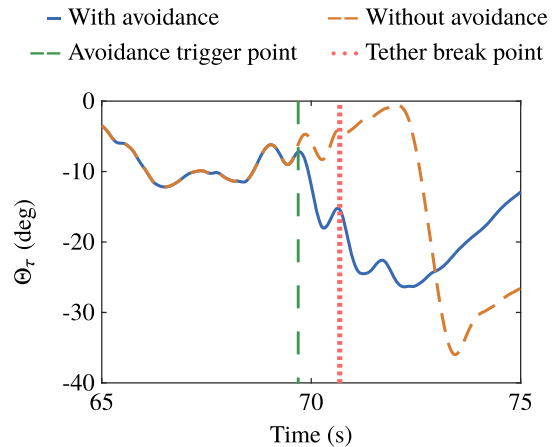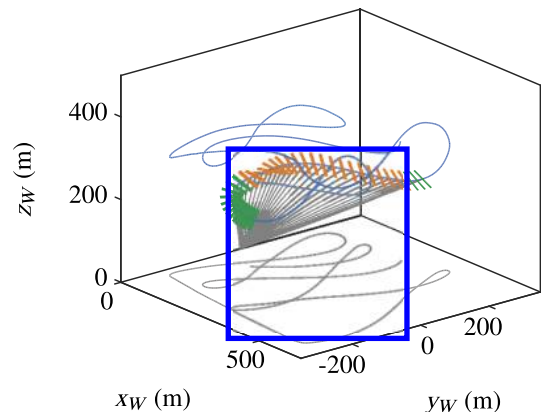

Fig. 16 Flight path of entire pumping cycle (blue) with the same aircraft attitude visualization as in Fig. 17.

beginning of the maneuver where the aircraft rolls negatively, with respect to the body-fixed frame x axis, into the tangential plane. The color changes back to green as soon as the avoidance maneuver is finished. In this case the end of the avoidance maneuver is defined as the first time the tether force set point reaches again 90% of the original traction phase tether tension set point. A visible drawback of the avoidance maneuver is that the aircraft flies about a quarter of the figure of eight at low tether tension, which results in a power loss. Hence, the amount of falsely predicted upsets (i.e., false positives) needs to be traded off against the power loss associated with a tether rupture. On the other hand, a visible deviation from the flight path
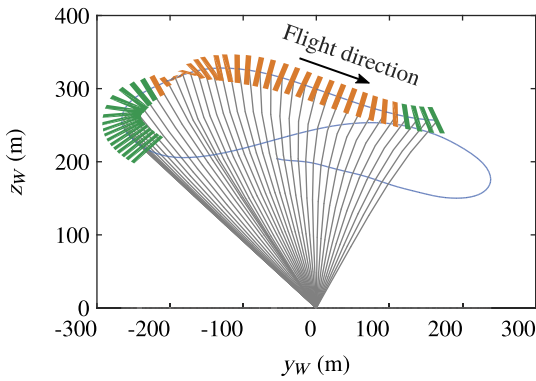
Fig. 17 Flight path (blue) with aircraft attitude visualization. Green indicates the pre- and postavoidance maneuver state; orange indicates the avoidance maneuver state.

cannot be observed, and the aircraft continues the traction phase without interruption, which is an advantage over strategies that need to abort the current operational mode in order to prevent tether rupture.

The actual reason for the tether break can be found by looking at the evolution of the tether, or winch, acceleration measured on the ground. In Fig. 18 the winch acceleration for the scenario without avoidance maneuver is displayed. The dashed orange line indicates the point where the tether breaks. Before the tether breaks the winch acceleration saturates at around 68 s and starts to jump between the maximum and the minimum acceleration limit with increasing frequency until the maximum supported tension is exceeded and the tether breaks. In contrast, Fig. 19 shows the winch acceleration for the

scenario with avoidance maneuver. In this case, the oscillation is prevented and instead the winch stays in the upper saturation limit, leading to a fast reeling out of the tether. The start of the avoidance maneuver is indicated by the dashed green line. The corresponding winch speed for the flight with tether rupture is displayed in Fig. 20. It can be observed that the winch speed itself is not saturating but also starts to oscillate due to the saturated acceleration. In contrast, with the avoidance maneuver the reeling out speed continues to increase after a small kink at the prediction point (see Fig. 21) and tether rupture is prevented. As soon as the avoidance maneuver is completed the winch starts reeling out slower according to the increasing tether force set point at around 70.2 s (see Fig. 11).

For completeness the mean wind speed at the aircraft around the point in time at which the tether ruptures is displayed in Fig. 22 and the wind speed evolution of the flight with triggered avoidance maneuver in the same time window is displayed in Fig. 23. The evolution of the wind speed before the avoidance maneuver is triggered or before the tether ruptures does not show any visible changes compared with wind speed after the avoidance maneuver and after tether rupture. This indicates that the tether rupture is not caused by an easy to comprehend change in the wind conditions at the aircraft but rather is a result of the complex interaction between aircraft and winch dynamics, as well as the control system and the wind conditions. This is consistent with the results displayed in Table 4, where the wind speed is not among the selected features.

In the previous paragraph the winch acceleration limits are identified as one cause for the tether rupture. However, a second factor represented by a specific airspeed and angle-of-attack combination can be identified. This can be shown by analyzing the distribution of airspeed and angle-of-attack pairs for simulation runs with and without tether rupture. Note that for the upset case the values at the upset are taken, and for the nominal flight the values are picked at randomly
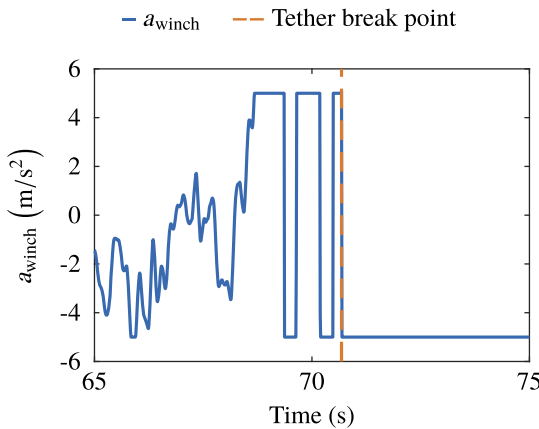

Fig. 18 Evolution of the winch acceleration without tether rupture avoidance.
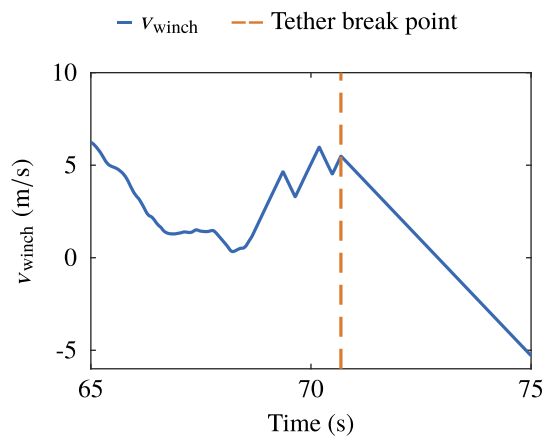

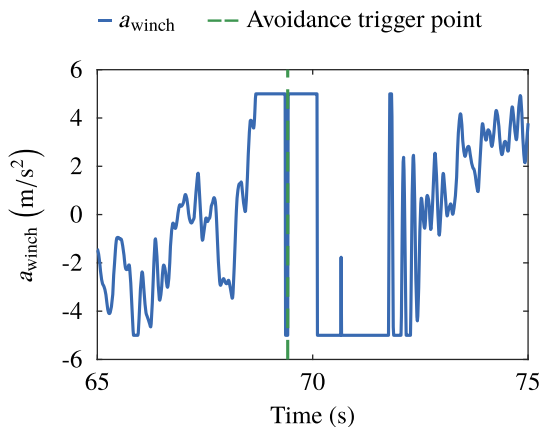**Fig. 20 Evolution of the winch speed without tether rupture avoidance.**

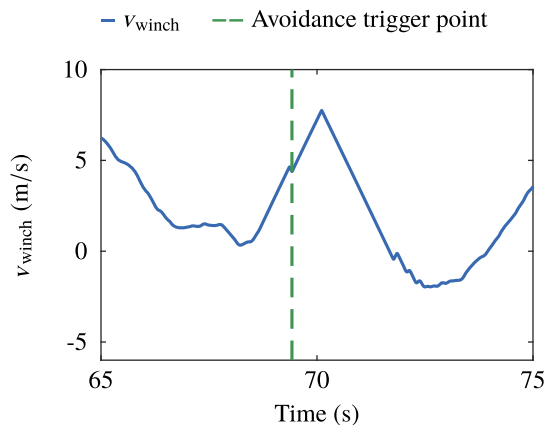**Fig. 19 Evolution of the winch acceleration with tether rupture avoidance.**

Fig. 21 Evolution of the winch speed with tether rupture avoidance.
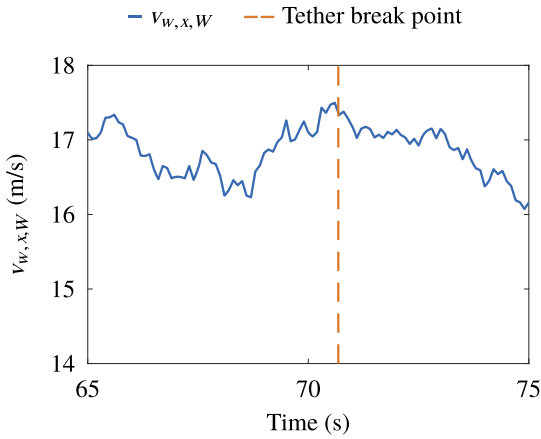
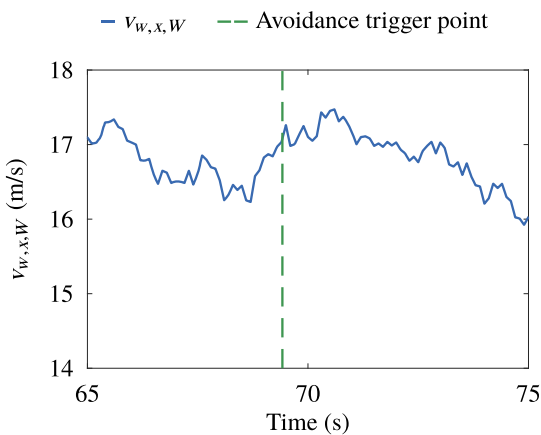Fig. 22 Evolution of wind speed in mean wind direction without avoidance maneuver.

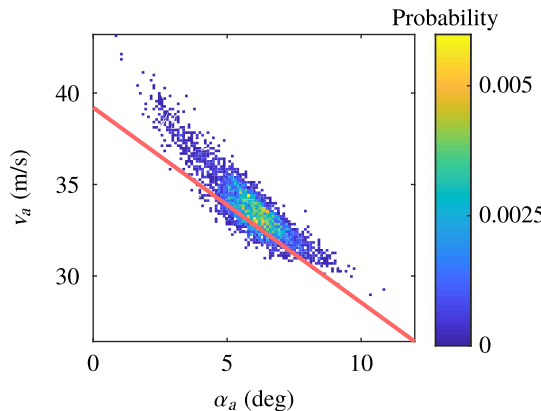Fig. 23 Evolution of wind speed in mean wind direction with avoidance maneuver.

Fig. 24 Distribution of airspeed v_a and angle of attack α_a pairs at tether rupture.

chosen positions on the flight path. The results are displayed in Figs. 24 and 25. In total 22,200 simulations without tether rupture and 5749 simulations with tether rupture from three different SS runs are used to approximate the distributions. The red solid line in both figures represents the same optimized separation boundary. The distributions itself are plotted in two different figures for visualization purposes. It can be observed that most of the samples above the boundary are simulation runs where the tether ruptured, whereas most of the simulation runs below the boundary are flight without tether rupture. The color gradient represents the conditional probability of a specific airspeed and angle-of-attack combination given a
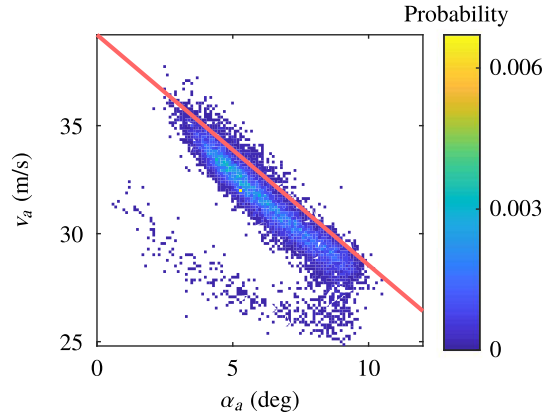

Fig. 25 Distribution of airspeed v_a and angle of attack α_a pairs at randomly selected times during traction phases without tether rupture.

tether rupture or given no tether rupture. The results show that a perfect separation between the two distributions is not possible. This is, however, to be expected because using only airspeed and angle of attack to distinguish tether rupture conditions from nominal flights reduces the dimension of the problem significantly. However, except some minor overlap in the tails the two modes of the distributions are indeed distinguishable (brighter color). Furthermore, above an airspeed of 37 $\mathrm{m} \cdot \mathrm{s}^{-1}$ there is a high chance that a sample belongs to the upset class independently of the angle-of-attack value. Similarly, below 30 $\mathrm{m} \cdot \mathrm{s}^{-1}$ and independent of the angle of attack no tether rupture will occur with a high probability. This result suggests an additional strategy to avoid tether rupture by limiting the angle-of-attack set point as a function of airspeed according to the plotted linear decision boundary. However, this is not investigated in this work further and is left for future research.

In the following the performance of different predictors will be investigated with respect to classical performance metrics as well as the introduced economic loss rate given by Eq. (27). In total, five different threshold-based predictors and one SVM-based predictor are compared with each other. The results are visualized in Fig. 26, where the conditional probabilities of not detecting and preventing an occurring tether rupture is plotted over the probability of a false positive. The blue, solid line with circular markers connects the performance pairs of the five threshold strategies. The performance of the SVM predictor is represented by the orange asterisk. The numerical values are listed in Table 5. The false-negative rates are estimated based on 764 flights with tether rupture and the false-positive probability is estimated based on 20,803 samples without tether rupture. For the fixed thresholds the probability of an FP is negatively correlated with the probability of a not correctly identified/prevented tether rupture, as expected. The closer the threshold value is selected to the critical value, the more likely it is that the tether rupture cannot be prevented due to the inertia of the system. In
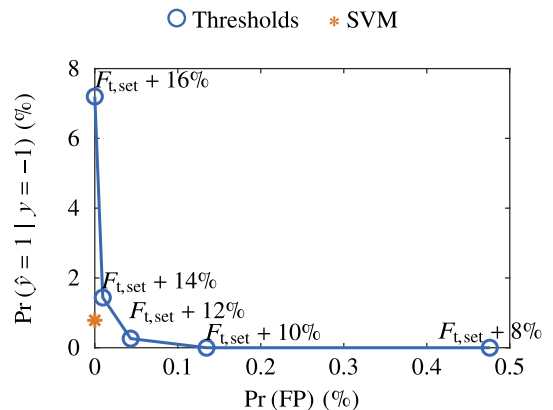

Fig. 26 Conditional probability of not preventing tether rupture given a tether rupture for different prediction thresholds and the SVM predictor.

Table 5   Upset detection performance

Method	$Pr(FP)$, %	$Pr(\hat{y}=1\|y=-1)$, %
$F_{t,\text{set}}+8\%$	0.48	0
$F_{t,\text{set}}+10\%$	0.13	0
$F_{t,\text{set}}+12\%$	0.04	0.26
$F_{t,\text{set}}+14\%$	0.01	1.44
$F_{t,\text{set}}+16\%$	0	7.20
SVM	0	0.79

Table 6   Average parameter values

Parameter	Value	Unit
P_{em}	0.4	kW
P_{pc}	3.9	kW
t_{pc}	2.5	minutes
p_f	2×10^{-7}	——

contrast to that, the more conservative the threshold is chosen, i.e., closer to the set point, the more likely it is to prevent a tether rupture but at the cost of an increasing false positive rate. Based on the numerical values and also based on Fig. 26 it is difficult to decide which is the best prediction strategy. The SVM achieves the best performance with respect to false positives together with the highest threshold, which, however, has an almost 10 times higher probability of not avoiding a tether rupture. Between the thresholds $F_{t,\text{set}}+14\%$ and $F_{t,\text{set}}+12\%$ the largest threshold value can be found that outperforms the SVM in terms of the false-negative rate. However, this predictor and predictors below this threshold lead to higher false-positive rates. Therefore, selecting the right prediction strategy solely based on these results is difficult because no reasonable acceptable FP and FN rate can be defined a priori and both metrics are not equally important from a practical point of view.

The energy loss rate defined in Eq. (27) tries to solve the aforementioned issue by assigning weights to the FP and FN rate proportionally to the associated performance loss. Because no reasonable estimation of the term E_{misc} can be made at this stage of the research, E_{misc} is set to zero in the following analysis. Note that in this case a false negative impacts the performance loss only through the power loss due to the number of missed pumping cycles during the downtime of the system. The expected downtime after a tether rupture is not available either, but a reasonable range of values can be defined and the losses can be plotted over the selected range. All other parameter values can be estimated using sample averages from the Monte Carlo simulations. The numerical values are listed in Table 6.

In Fig. 27 the performance loss rate L per average converted energy per pumping cycle E_{pc} is displayed as a function of system downtime
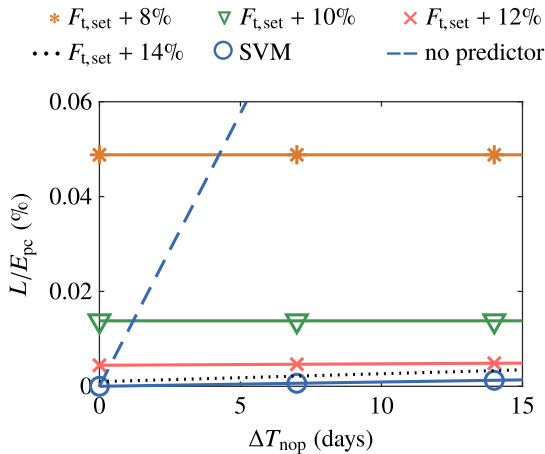
Fig. 27   Comparison of loss rates across different predictors.

using Eq. (27). The results show that without prevention strategy (dashed line) the relative performance loss grows quickly with increasing system downtime even for the estimated low probability of tether rupture. For the thresholds $F_{t,\text{set}}+8\%$ and $F_{t,\text{set}}+10\%$ the loss rate remains constant because their false-negative rate is zero, and hence the downtime has no impact on the loss. The other thresholds and the SVM predictor loss rates remain nearly constant as well due to the overall small probability of false negatives among the predictors. The SVM leads to the lowest loss rate among the predictors in the considered time window.

C.   Discussion Model Validity and Future Work

The presented framework uses models of the AWE system as well as the wind to create, predict, and prevent upset conditions. The accuracy of the models is critical in order to be able to project the results to reality. The aircraft model has been validated to some extent as described in [11,12], but especially for quick changes in the wind conditions the aerodynamic model is probably too aggressive because changes in the local flow immediately change the resulting lift force. It is expected that with a more realistic aerodynamic model an additional time delay between changes in the local flowfield around the aircraft and the resulting change of the tether force is present, which might alter the presented results in the previous section. The present model can hence be regarded as conservative, and it is expected that the prediction accuracy can be further improved with a more realistic model. Testing the framework with a more realistic aircraft model is therefore regarded as the main suggestion for future work. This will also allow to investigate further upset conditions related to the structural and aerodynamic integrity of the aircraft. For instance, wind conditions that lead to critical wing bending or severe vibrations can be generated using the SS algorithm and a data-driven predictor such as the SVM predictor can be used to trigger a load and/or vibration alleviation strategy if necessary. Finally, it needs to be emphasized that the presented results are strongly dependent on the specific controller. To investigate how well the results generalize it is recommended to apply the presented methodology to a different closed-loop system model in the future. Moreover, additional data-driven methods for the predictor can be tested to further decrease both false-positive and false-negative rates and hence the economic loss. Finally, if more information about the terms in Eq. (27) is available, the SVM should be optimized with respect to the loss and not the MCC.

V.   Conclusions

The contribution of the present work consists of two major parts. First, a modification to an existing pumping cycle control system is presented. It improves the transition from retraction to traction phase on the guidance level by a controlled rotation of the figure-of-eight flight path from high to low elevation angles. Second, a framework to generate, predict, and prevent upset conditions that jeopardize the long-term reliability of AWE systems is presented. The feasibility of the framework is demonstrated with the example of tether rupture. The presented results in the paper allow to draw the following conclusions.

First, the introduced control modification can reduce the tether rupture probability significantly if a large time constant for the transition phase is chosen. However, for increasing transition times the aircraft flies longer at higher elevation angles, which decreases the average pumping cycle power. Hence, the choice of the time constant involves a tradeoff between robustness and average power output.

Second, a conservative implementation of the control modification leads to a low tether rupture probability. This makes it impractical to create knowledge about the conditions in which the tether breaks using a direct Monte Carlo simulation approach. The SS algorithm can achieve this task more efficiently. The samples generated by the SS algorithm can be used to design and optimize a prediction model that is able to detect tether rupture before it occurs. To trade off false-positive and false-negative rates a cost function is introduced that is better suited to rank predictor performances than conventional classification measures. It allows to associate an average energy loss rate to each of the two prediction error types and hence weights prediction

errors according to their practical impact. The SVM-based predictor achieves the lowest loss in the investigated scenario, but more accurate information about the involved parameters in the cost function such as system downtime, repair costs, and maintenance costs is required in the future to improve the validity of the loss rate function.

Third, the proposed upset avoidance maneuver can reliably prevent tether rupture while keeping the system operational. The avoidance maneuver does not require to abort the current pumping cycle or even land the aircraft, which lowers the impact of false positives on the average power output.

Fourth, the analysis of the flights with tether rupture shows that the upset is a result of the winch acceleration saturation as well as a specific combination of airspeed and angle of attack during the traction phase of the pumping cycle. No visible patterns in the wind conditions could be identified, which leads to the conclusion that this specific upset is due to the complex interaction between the dynamics of the subsystems in addition to the atmospheric turbulence.

Finally, for well-tuned control systems upset conditions occur with low probability, which poses the question if accepting upsets is better than preventing them and therefore avoiding any prediction-error-induced costs. However, the results in this work show that in the long run also rare upset conditions can have an impact on the average power output, and hence augmenting AWE baseline control systems with an upset tailored prediction and prevention strategy, such as the one presented in this work, is recommended.

Acknowledgment

References

[1] Horn, G., Gros, S., and Diehl, M., "Numerical Trajectory Optimization for Airborne Wind Energy Systems Described by High Fidelity Aircraft Models," *Airborne Wind Energy*, edited by U. Ahrens, M. Diehl, and R. Schmehl, Green Energy and Technology, Springer, Berlin, 2013, pp. 205–218, Chap. 11.
https://doi.org/10.1007/978-3-642-39965-7_11

[2] Zanon, M., Gros, S., and Diehl, M., *Model Predictive Control of Rigid-Airfoil Airborne Wind Energy Systems*, Springer, Berlin, 2013, pp. 219–233.
https://doi.org/10.1007/978-3-642-39965-7_12

[3] Cobb, M., Barton, K., Fathy, H., and Vermillion, C., "Iterative Learning-Based Waypoint Optimization for Repetitive Path Planning, with Application to Airborne Wind Energy Systems," *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, Inst. of Electrical and Electronics Engineers, New York, 2017, pp. 2698–2704.
https://doi.org/10.1109/cdc.2017.8264051

[4] Licitra, G., Koenemann, J., Bürger, A., Williams, P., Ruiterkamp, R., and Diehl, M., "Performance Assessment of a Rigid Wing Airborne Wind Energy Pumping System," *Energy*, Vol. 173, April 2019, pp. 569–585.
https://doi.org/10.1016/j.energy.2019.02.064

[5] Salma, V., Friedl, F., and Schmehl, R., "Improving Reliability and Safety of Airborne Wind Energy Systems," *Wind Energy*, Vol. 23, No. 2, 2020, pp. 340–356.
https://doi.org/10.1002/we.2433

[6] 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*, Vol. 42, No. 11, 2019, pp. 2456–2473.
https://doi.org/10.2514/1.G004246

[7] Robert, C. P., and Casella, G., *Monte Carlo Statistical Methods (Springer Texts in Statistics)*, Springer-Verlag, Berlin, 2005.
https://doi.org/10.1007/978-1-4757-4145-2

[8] Au, S.-K., and Beck, J. L., "Estimation of Small Failure Probabilities in High Dimensions by Subset Simulation," *Probabilistic Engineering Mechanics*, Vol. 16, No. 4, 2001, pp. 263–277.
https://doi.org/10.1016/S0266-8920(01)00019-4

[9] Löbl, D., and Holzapfel, F., "Subset Simulation for Estimating Small Failure Probabilities of an Aerial System Subject to Atmospheric Turbulences," *AIAA Atmospheric Flight Mechanics Conference*, AIAA Paper 2015-0236, 2015.
https://doi.org/10.2514/6.2015-0236

[10] Wang, M., Zhang, S., Holzapfel, F., Löbl, D., and Hellmundt, F., "Probabilistic Assessment of a Safety-Critical Backup Controller by Subset Simulation," *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 5, 2019, pp. 1146–1156.
https://doi.org/10.2514/1.G003901

[11] Licitra, G., "Identification and Optimization of an Airborne Wind Energy System," Ph.D. Thesis, Univ. of Freiburg, Breisgau, Germany, 2018, https://freidok.uni-freiburg.de/data/16226.

[12] Malz, E. C., Koenemann, J., Sieberling, S., and Gros, S., "A Reference Model for Airborne Wind Energy Systems for Optimization and Control," *Renewable Energy*, Vol. 140, Sept. 2019, pp. 1004–1011.
https://doi.org/10.1016/j.renene.2019.03.111

[13] Beard, R. W., and McLain, T. W., *Small Unmanned Aircraft: Theory and Practice*, Princeton Univ. Press, Princeton, NJ, 2012.
https://doi.org/10.1515/9781400840601

[14] Fechner, U., van der Vlugt, R., Schreuder, E., and Schmehl, R., "Dynamic Model of a Pumping Kite Power System," *Renewable Energy*, Vol. 83, Nov. 2015, pp. 705–716.
https://doi.org/10.1016/j.renene.2015.04.028

[15] *Aerospace Toolbox*, Ver. 3.0, The MathWorks, Natick, MA, 2018.

[16] Zuev, K. M., *Subset Simulation Method for Rare Event Estimation: An Introduction*, Springer, Berlin, 2021, pp. 1–25.
https://doi.org/10.1007/978-3-642-36197-5_165-1

[17] Papaioannou, I., Betz, W., Zwirglmaier, K., and Straub, D., "MCMC Algorithms for Subset Simulation," *Probabilistic Engineering Mechanics*, Vol. 41, July 2015, pp. 89–103.
https://doi.org/10.1016/j.probengmech.2015.06.006

[18] Barber, D., *Bayesian Reasoning and Machine Learning*, Cambridge Univ. Press, Cambridge, England, U.K., 2012.
https://doi.org/10.1017/CBO9780511804779

[19] Cristianini, N., and Shawe-Taylor, J., *Support Vector Machines*, Cambridge Univ. Press, Cambridge, England, U.K., 2000, pp. 93–124.
https://doi.org/10.1017/CBO9780511801389.008

[20] Krawczyk, B., "Learning from Imbalanced Data: Open Challenges and Future Directions," *Progress in Artificial Intelligence*, Vol. 5, No. 4, 2016, pp. 221–232.
https://doi.org/10.1007/s13748-016-0094-0

[21] Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P., "SMOTE: Synthetic Minority Over-Sampling Technique," *Journal of Artificial Intelligence Research*, Vol. 16, No. 1, 2002, p. 321–357.
https://doi.org/10.1613/jair.953

[22] Fulcher, B. D., and Jones, N. S., "Highly Comparative Feature-Based Time-Series Classification," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 26, No. 12, 2014, pp. 3026–3037.
https://doi.org/10.1109/TKDE.2014.2316504

[23] Chicco, D., and Jurman, G., "The Advantages of the Matthews Correlation Coefficient (MCC) over F1 Score and Accuracy in Binary Classification Evaluation," *BMC Genomics*, Vol. 21, No. 1, 2020, p. 6.
https://doi.org/10.1186/s12864-019-6413-7

[24] Statistics and Machine Learning Toolbox, Ver. 11.4, The MathWorks, Natick, MA, 2018.