Chinese Journal of Aeronautics, (2022), 35(7): 19–29

Chinese Society of Aeronautics and Astronautics
& Beihang University

Chinese Journal of Aeronautics

cja@buaa.edu.cn
www.sciencedirect.com

Smooth free-cycle dynamic soaring in unspecified shear wind via quadratic programming

Haichao HONG *, Benedikt GRÜTER, Patrick PIPREK, Florian HOLZAPFEL

Institute of Flight System Dynamics, Technical University of Munich, Garching D-85748, Germany

Received 2 February 2021; revised 18 March 2021; accepted 27 August 2021
Available online 25 October 2021

KEYWORDS

Dynamic soaring;
Optimal control;
Quadratic programming;
Trajectory generation;
Trigonometric series

Abstract Harvesting wind energy is promising for extending long-endurance flights, which can be greatly facilitated by a flight technique called dynamic soaring. The presented study is concerned with generating model-based trajectories with smooth control histories for dynamic soaring maneuvers exploiting wind gradients. The desired smoothness is achieved by introducing a trigonometric series parameterization for the controls, which are formulated with respect to the normalized time. Specifically, the periodicity of the trigonometric functions is leveraged to facilitate the connection of cycles and streamline the problem formulation. Without relying on a specified wind profile, a free-final-time quadratic programming-based control strategy is developed for the online correction of the flight trajectory, which requires only the instant wind information. Offline and online numerical studies show the trade-off to achieve the smoothness and demonstrate the effectiveness of the proposed method in a varying wind field.

1. Introduction

Since the first observations of the flight of albatrosses by Rayleigh[1] in 1883, dynamic soaring, which is a flying technique that some birds perform to extract kinetic energy from the moving air to support their long-endurance flights, has interested the research community continuously.[2–5] Theoretical analysis of the flight mechanics of dynamic soaring[6–8] and advancements in computational capacity made it possible to perform numerical optimization.[9–11] These have sparked the determination of transferring the flight pattern from birds to aircraft, especially Unmanned Aerial Vehicles (UAVs).[12–15]

Offline studies rely on prescribed wind profiles. However, accurate modeling of the complete shear wind profile is not feasible in real world, especially for real-time tasks. Efforts to overcome this caveat by introducing online wind estimation are undertaken in Ref. 16. The authors propose a closed-loop control system including a wind estimation based on the aircraft states, which is used to re-optimize the trajectory considering current wind conditions. A feedforward-feedback controller is used to track the planned trajectory, reusing the results from the trajectory planner as feedforward commands. The problem is also addressed in Ref. 17 constructing a parameter-dependent representation of the flight path, and a deep neural network is utilized to adjust the parameterized flight path according to the wind conditions. In Ref. 17, the direct control inputs are determined by a path-following controller. Therefore, the correction of the trajectory and the generation of control commands are separated, which facilitates the deployment of different path-following controllers. Nonetheless, to follow a desired trajectory requires the

* Corresponding author.
E-mail address: haichao.hong@tum.de (H. HONG).

https://doi.org/10.1016/j.cja.2021.09.012

corresponding agility of aircraft. Taking the aircraft dynamics into consideration might enhance the general performance. In this regard, yet another characteristic that can be improved in the field of dynamic soaring trajectory generation is the smoothness of the control inputs. The controls, e.g., the lift coefficient and the bank angle, account for the flying and lift-generating mechanism. In principle, steps in control histories are undesirable, or even unachievable. However, with the exception of Ref. 17, most existing studies utilize optimal control methods that do not emphasis such a requirement.

The presented study of dynamic soaring trajectory online generation in an unspecified shear wind profile is originated in Ref. 18. Here, an approximation of the wind strength variation is introduced and utilized in a convex optimization formulation with control constraints for online correcting the soaring trajectory in an unspecified wind profile. The optimization formulation in Ref. 18 considers a fixed final time, so the cycle time is fixed and needs to be specified. In addition, the smoothness of control variables cannot be guaranteed. Furthermore, state constraints, such as the altitude constraint, cannot be considered. A trigonometric series-based smooth trajectory generation method is developed in Ref. 19: These controls, being infinitely differentiable and analytically defined, render more practical applicable commands and thus higher availability of the trajectory. Moreover, constraints on control derivatives can be incorporated without introducing new variables. These features may enable a smooth dynamic soaring trajectory that can be readily implemented. However, a free cycle is yet to be realized. Therefore, this paper aims at enhancing the previous solution by addressing these concerns. First, the trigonometric series design is applied to parameterize the control variables for obtaining the smoothness. The series are formulated with respect to the normalized time in order to cope with the free final time. Moreover, the period of the series is designed to be the same as the cycle time so that the controls and control derivatives are equal at the boundary points, which enforces smooth connections of dynamic soaring cycles. An offline simulation study compares the minimum wind strengths required by performing an energy-neutral cycle for the full discretization and for the trigonometric series design. Second, the relationship between the wind strength variation and the variation of the trajectory is derived. The wind strength variation is determined by the changes of the reference wind speed in a nominal wind model. The variation of the trajectory includes the incremental changes of states, controls, replaced by the series coefficients, and the final time. This relationship is considered in a strictly convex Quadratic Programming (QP) formulation with all constraints converted to forms with respect to the increments. Third, solving the QP problem yields an updated soaring trajectory with a new cycle time, for which only instant wind information are needed instead of the complete wind profile. This process is repeated to continuously update the trajectory. Online trajectory correction studies are conducted to show the effectiveness of the proposed method, where wind measurement errors are considered as well.

The contributions of this paper are summarized as follows: (A) The trigonometric parameterization of the controls is applied for generating dynamic soaring flight trajectories with smooth controls. (B) The proposed series design takes advantage of the periodicity of trigonometric functions to simplify the implementation of boundary constraints on controls and control derivatives in accordance to the periodic soaring maneuver. (C) Considering the trigonometric series design, a free-final-time QP formulation including state, control, and control derivative constraints is derived, leading to a free-cycle flight trajectory generation. An approximation of the wind strength variation is used in the QP form for online updating the trajectory in unspecified wind profiles.

This paper is structured to provide not only technical details but also insight about the proposed method: First, the problem formulation is given in Section 2. Next, the trigonometric series parameterization is detailed in Section 3, illustrating the mathematical properties and the utilization of them. After the development of the parameterization method, an offline comparison is presented in Section 4, showing the benefits and the corresponding trade-off of acquiring smooth control inputs. The online trajectory correction based on quadratic programming is then developed in Section 5, including the modeling of the wind strength variation and the formulation of the optimization problem. The simulation results of the online trajectory correction are presented in Section 6, the simulation scenario of which considers a randomly generated wind field. Finally, conclusions are drawn in Section 7.

2. Problem formulation

In this section, details of the problem formulation including the Equations of Motion (EoM) and the constraints are introduced. The problem formulation, including the mathematical model and constraints, is developed based on Refs. 6,18.

The point-mass EoM are expressed in a locally fixed inertial navigation frame as

$$\mathrm{d}u_{Kg}/\mathrm{d}t = -a_{u1}D/m - a_{u2}L/m \tag{1a}$$
$$\mathrm{d}v_{Kg}/\mathrm{d}t = -a_{v1}D/m - a_{v2}L/m \tag{1b}$$
$$\mathrm{d}w_{Kg}/\mathrm{d}t = -a_{w1}D/m - a_{w2}L/m - g \tag{1c}$$
$$\mathrm{d}x_g/\mathrm{d}t = u_{Kg} \tag{1d}$$
$$\mathrm{d}y_g/\mathrm{d}t = v_{Kg} \tag{1e}$$
$$\mathrm{d}h_g/\mathrm{d}t = w_{Kg} \tag{1f}$$

where x_g, y_g, and h_g are the coordinates, while u_{Kg}, v_{Kg}, w_{Kg} are the inertial velocity components. m is the mass and g is the gravitational acceleration. The coefficients, $a_{ij}(i = u, v, w; j = 1, 2)$, are obtained from the transformation matrix between the aerodynamic frame and the navigation frame as

$$a_{u1} = \cos\gamma_A \cos\chi_A \tag{2a}$$
$$a_{v1} = \cos\gamma_A \sin\chi_A \tag{2b}$$
$$a_{w1} = \sin\gamma_A \tag{2c}$$
$$a_{u2} = \cos\mu_A \sin\gamma_A \cos\chi_A + \sin\mu_A \sin\chi_A \tag{2d}$$
$$a_{v2} = \cos\mu_A \sin\gamma_A \sin\chi_A - \sin\mu_A \cos\chi_A \tag{2e}$$
$$a_{w2} = -\cos\mu_A \cos\gamma_A \tag{2f}$$

where γ_A, χ_A, and μ_A are the aerodynamic flight-path angle, heading angle, and bank angle, respectively. The lift L and drag D in Eq. (1) are expressed as

$$L = \frac{1}{2}\rho V_A^2 S C_L \tag{3}$$
$$D = \frac{1}{2}\rho V_A^2 S C_D \tag{4}$$

where ρ and S are the air density and the reference area, respectively. V_A is the airspeed. The drag coefficient C_D is computed as a function of the lift coefficient C_L as

$$C_D = C_{D0} + k_{\mathrm{i}} C_L^2 \tag{5}$$

where C_{D0} is the zero-lift drag coefficient and k_{i} is the induced drag factor.

The wind triangle is given by

$$\boldsymbol{v}_K = \boldsymbol{v}_A + \boldsymbol{v}_W \tag{6}$$

where $\boldsymbol{v}_A, \boldsymbol{v}_K$, and $\boldsymbol{v}_W$ are the aerodynamic velocity, the inertial velocity, and the wind velocity vectors, respectively. The inertial velocity is given by

$$\boldsymbol{v}_K = \left[u_{Kg},\, v_{Kg},\, w_{Kg}\right]^{\mathrm{T}} \tag{7}$$

The navigation frame can be specified to yield the expression of the wind velocity vector as

$$\boldsymbol{v}_W = [0,\, -V_W,\, 0]^{\mathrm{T}} \tag{8}$$

where V_W is determined by

$$V_W = V_{W\mathrm{ref}} \frac{\ln\left(h_g/h_0\right)}{\ln\left(h_{\mathrm{ref}}/h_0\right)} \tag{9}$$

where $V_{W\mathrm{ref}}$ is the reference wind speed at $h_g = h_{\mathrm{ref}}$, indicating the wind strength. Substituting Eqs. (7) and (8) to Eq. (6) leads to

$$\boldsymbol{v}_A = \left[u_{Kg},\, v_{Kg} + V_W,\, w_{Kg}\right]^{\mathrm{T}} \tag{10}$$

Then, V_A, χ_A, and γ_A can be computed as

$$V_A = \sqrt{u_{Kg}^2 + \left(v_{Kg} + V_W\right)^2 + w_{Kg}^2} \tag{11}$$

$$\chi_A = \arctan\left(\frac{v_{Kg} + V_W}{u_{Kg}}\right) \tag{12}$$

$$\gamma_A = \arcsin\left(\frac{w_{Kg}}{V_A}\right) \tag{13}$$

In summary, the state and control vectors of this dynamical system are

$$\boldsymbol{x} = \left[u_{Kg}, v_{Kg}, w_{Kg}, x_g, y_g, h_g\right]^{\mathrm{T}} \tag{14}$$

$$\boldsymbol{u} = \left[C_L, \mu_A\right]^{\mathrm{T}} \tag{15}$$

Next, the constraints are described. The dynamic soaring cycle studied in this paper is the bend-type cycle form, which is schematically shown in Fig. 1.[18] This form is known for facilitating long-distance traveling.[8] An energy-neutral cycle supported by proper wind profiles starts and ends with the same state values except that the displacements in the longitudinal and lateral directions are free. In this study, we do not assume that the wind profile can support an energy-neutral cycle. Therefore, the terminal constraints are defined as

$$\boldsymbol{y}(t_{\mathrm{f}}) = \left[h_g(t_{\mathrm{f}}), v_{Kg}(t_{\mathrm{f}}), w_{Kg}(t_{\mathrm{f}})\right]^{\mathrm{T}} \tag{16}$$

where $\boldsymbol{y}$ is the output vector. More specifically,

$$\boldsymbol{y}(t_{\mathrm{f}}) = \boldsymbol{y}(0) = \left[h_g(0), 0, 0\right]^{\mathrm{T}} = \boldsymbol{y}_{\mathrm{f}} \tag{17}$$

This constraint specifies that the cycle ends at a desired altitude with desired velocity direction, and yet the absolute value of the velocity can vary. On the other hand, an energy-neutral cycle additionally requires

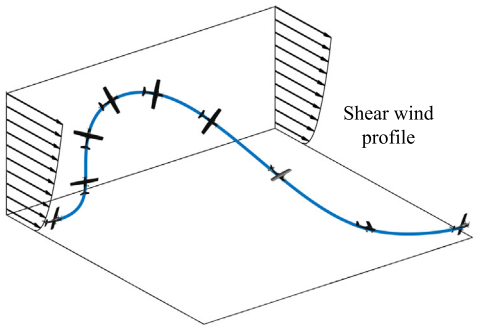

Fig. 1 Schematic representation of a bend-type dynamic soaring cycle.[18]

$$u_g(0) = u_g(t_{\mathrm{f}}) \tag{18}$$

The altitude should be higher than the minimum value, which is a state inequality constraint, as

$$h_g(t) \geqslant h_{g_{\min}} \tag{19}$$

Generally speaking, the altitude constraint considered here is a conservative selection in terms of exploiting the full potential of the wind field. In a nonlinear optimization context, the wingtip clearance constraints introduced in Ref. 13 may be considered to improve the optimality. The safe height is chosen considering the wingspan b according to $h_{g_{\min}} > \left|\frac{1}{2} b \sin \mu_A\right|_{\max}$. Boundary constraints and path constraints are considered on controls and control derivatives. At boundary points, i.e., the initial and terminal points connecting soaring cycles, the controls are equal:

$$\boldsymbol{u}(0) = \boldsymbol{u}(t_{\mathrm{f}}) = \boldsymbol{u}_{\mathrm{i}} \tag{20}$$

The control derivatives at boundary points are designed to be zero

$$\dot{\boldsymbol{u}}(0) = \dot{\boldsymbol{u}}(t_{\mathrm{f}}) = \boldsymbol{0} \tag{21}$$

The path constraints on controls and control derivatives are given as

$$\boldsymbol{u}_{\min} \leqslant \boldsymbol{u} \leqslant \boldsymbol{u}_{\max} \tag{22}$$

$$\dot{\boldsymbol{u}}_{\min} \leqslant \dot{\boldsymbol{u}} \leqslant \dot{\boldsymbol{u}}_{\max} \tag{23}$$

3. Trigonometric series parameterization

In this paper, we apply a trigonometric series parameterization method,[19] to formulate the control variables, where the series are infinitely differentiable. For aircraft/birds performing dynamic soaring, no discontinuities of any kinematic state or control physically exist in practice. Therefore, generated trajectories using the trigonometric series parameterization, which are guaranteed to be smooth, better represent the actual movement of aircraft/birds in the real-world. Moreover, a smooth trajectory may lead to a more efficient and robust usage of the control authority, as it avoids sacrificing control authority for following unphysical kinks and steps in the non-smooth optimal trajectory.

On the basis of Ref. 19, the periodicity of the series functions is designed to have the characteristic that can simplify the optimization problem. Moreover, as a free-cycle is required, the series are formulated with respect to the normal-

ized time $\tau = t/t_f \in [0, 1]$. The expression of the i-th variable of the control vector is given as:

$$\begin{aligned} u_i(\tau) &= (a_0)_i + \sum_{n=1}^{M} \left[(a_n)_i \cos(2\pi n\tau) + (b_n)_i \sin(2\pi n\tau) \right] \\ &= \boldsymbol{s}_{Mi}(\tau)\boldsymbol{c}_i, \quad i = 1, 2, \ldots, q \end{aligned} \tag{24}$$

where the basis vector and the coefficient vector are given by

$$\begin{aligned} \boldsymbol{s}_{Mi}(\tau) = [&1, \cos(2\pi\tau), \ldots, \cos(2\pi M\tau), \\ &\sin(2\pi\tau), \ldots, \sin(2\pi M\tau)] \end{aligned} \tag{25}$$

$$\boldsymbol{c}_i = \left[(a_0)_i, (a_1)_i, \ldots, (a_M)_i, (b_1)_i, \ldots, (b_M)_i \right]^{\mathrm{T}} \tag{26}$$

Here, M is the order of the series, and q is the number of controls. Therefore, the control vector can be written as

$$\boldsymbol{u}(\tau) = \begin{bmatrix} \boldsymbol{s}_{M1}(\tau) & 0 & \cdots & 0 \\ 0 & \boldsymbol{s}_{M2}(\tau) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{s}_{Mq}(\tau) \end{bmatrix} \begin{bmatrix} \boldsymbol{c}_1 \\ \boldsymbol{c}_2 \\ \vdots \\ \boldsymbol{c}_q \end{bmatrix} = \boldsymbol{S}_M(\tau)\boldsymbol{c} \tag{27}$$

It can be noticed that $\boldsymbol{u}$ and $\boldsymbol{S}_M$ have a periodicity of 1, which automatically guarantees that the initial and terminal values of $\boldsymbol{u}$ and their derivatives are equal.

$$\overset{(n)}{\boldsymbol{u}}(0) = \overset{(n)}{\boldsymbol{u}}(1), \quad n \in \mathbf{N}^0 \tag{28}$$

This feature facilitates a periodic dynamic soaring pattern that can be repeated. Using the chain rule, the time derivative of the controls can be obtained via

$$\dot{\boldsymbol{u}}(t) = \frac{\mathrm{d}\boldsymbol{u}(\tau)}{\mathrm{d}\tau} \cdot \frac{\mathrm{d}\tau}{\mathrm{d}t} = \left[\frac{\mathrm{d}\boldsymbol{S}_M(\tau)}{\mathrm{d}\tau} \right] \frac{1}{t_f} \boldsymbol{c} \tag{29}$$

Also, $\boldsymbol{u}$ being a linear function of $\boldsymbol{c}$ leads that the incremental change of $\boldsymbol{u}$ and $\dot{\boldsymbol{u}}$ are linear to the increments of $\boldsymbol{c}$.

4. Offline optimization

This section is dedicated to showing the difference of the minimum required wind strength for an energy-neutral cycle between using the Full Discretization (FD) and the Trigonometric Series (TS) parameterization. The model parameters and constraint values are given in Table 1. We use FALCON.m,[20] which is a free-of-charge optimal control software, to generate comparison results, and the TS solution serves as the initial solution for the online simulation in the next section. The vehicle dynamics, i.e., the EoM in Eq. (1) are transcribed as equality constraints in the optimization problem. By default, FALCON.m utilizes a trapezoidal collocation for the discretization, where the control variables at each grid point are independent. After deploying the trigonometric series parameterization, the controls are functions of the coefficient vector, and a fifth order series is used. Before showing the numerical results, the two setups for the optimization problem formulation are summarized. For the original problem, we have

$$\begin{cases} \underset{\boldsymbol{u}, t_f}{\text{minimize}} & J = V_{W\text{ref}} \\ \text{subject to} & \text{EoM in Eq. (1)} \\ & \text{Constraints in Eqs. (17)–(20), (22)} \end{cases} \tag{30}$$

Next, the formulation with the trigonometric series parameterization is

$$\begin{cases} \underset{\boldsymbol{c}, t_f}{\text{minimize}} & J = V_{W\text{ref}} \\ \text{subject to} & \text{EoM in Eq. (1)} \\ & \text{Parameterization in Eq. (27)} \\ & \text{Constraints in Eqs. (17)–(23)} \end{cases} \tag{31}$$

One can notice that the differences in the two problem formulations are the unknown variables, the parameterization of controls, and the constraints on the control derivatives. The controls are the unknowns in Eq. (30), while in Eq. (31) the series coefficients are to be determined. The boundary and path constraints on control derivatives by Eqs. (21) and (22) are not considered in Eq. (30).

In the following, the offline optimization results are presented. The trajectories are shown in Fig. 2, and the time histories of the velocity components are displayed in Fig. 3. The time histories of controls and control derivatives are presented in Figs. 4 and 5, respectively. For analyzing the results, the behavior of the generated controls is examined first. It can be noticed in Fig. 4 that, although the boundary constraints on controls are satisfied, they lead to big undesired steps near the boundary points. This is because the controls are fully discretized in the FD scheme, and therefore they are mutually independent and are optimized independently. On the other hand, the controls are by construction smooth with the TS parameterization. Both C_L and μ_A change smoothly. In the TS trajectory, the boundary constraints and the continuity of the lift coefficient lead to a larger utilization of its authority in the beginning and in the end. This results in larger turn rates and thus a smaller displacement in the y_g direction, which can be seen in Fig. 2. Also, as a result of the full discretization, the control derivatives are not continuous as shown in Fig. 5. However, the control derivatives generated by the TS parameterization are smooth. One additional benefit that can be noticed is that the bounds on the derivative are achieved by the TS method without introducing new variables, while they are not considered in the FD scheme. Moreover, the zero conditions at the boundary points are satisfied. The cycle times are 7.31 s and 7.40 s for FD and TS, respectively. The cycle time is marginally longer for the TS solution. The minimum required wind strength are 8.12 m/s and 8.49 m/s for FD and TS, respectively. There is a 4.6% of suboptimality using the TS parameterization, whereas the smoothness of controls is significantly improved.

5. Online trajectory correction

This section summarizes the different approaches used for an online trajectory correction, consisting of wind strength variation (Section 5.1), reformulation of the dynamics (Section 5.2), and QP formulation (Section 5.3).

5.1. Wind strength variation

The philosophy of approximating the wind strength variation is to calculate the incremental change of a parameter in the wind profile model in Eq. (9).[18] This approach is developed based on the assumption that the instant altitude and wind information is available. This assumption can be justified by

Table 1 Model parameters and constraint values.

Parameter	Value	Parameter	Value
m	8.5 kg	$h_{g_{\min}}$	1.8 m
g	9.81 m/s^2	b	3.5 m
ρ	1.225 kg/m^3	$\boldsymbol{x}(0)$	$[0\ m, 0\ m, 2\ m, 20\ m/s, 0\ m/s, 0\ m/s]^\mathsf{T}$
C_{D0}	0.023	$\boldsymbol{u}_\mathrm{i}$	$[1.5,\ 70\ °]^\mathsf{T}$
k_i	0.019	$\boldsymbol{u}_{\min}$	$[0.5,\ -70\ °]^\mathsf{T}$
S	0.65 m^2	$\boldsymbol{u}_{\max}$	$[1.5,\ 70\ °]^\mathsf{T}$
h_ref	10 m	$\dot{\boldsymbol{u}}_{\min}$	$[-1\ s^{-1},\ -70\ °/s]^\mathsf{T}$
h_0	0.03 m	$\dot{\boldsymbol{u}}_{\max}$	$[1\ s^{-1},\ 70\ °/s]^\mathsf{T}$

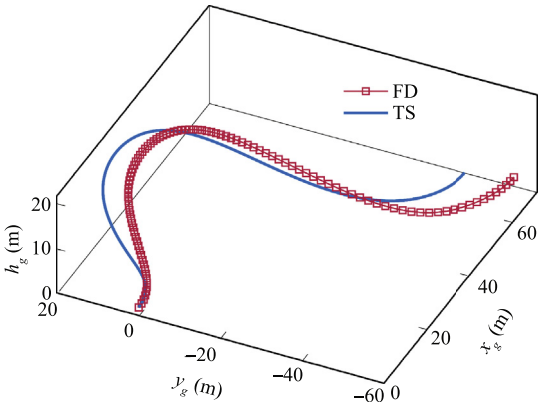

Fig. 2 Offline: Trajectories.

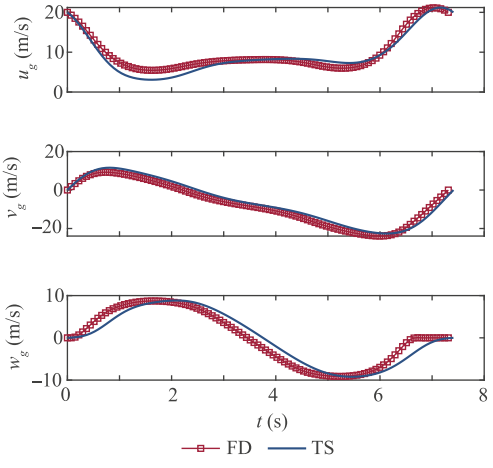

Fig. 3 Offline: Time histories of inertial velocity.

the recent progress in on-board wind estimation methods.[21–24] Provided that h_g and V_W are known, the reference wind strength $V_{W\mathrm{ref}}$ can be computed using Eq. (9), referred to as the nominal profile hereafter. The incremental change of $V_{W\mathrm{ref}}$ between two consecutive time instances, i.e., the wind strength variation denoted by $\delta V_{W\mathrm{ref}}$, can be thus calculated. The process is schematically shown in Fig. 6.[18] The difference in $V_{W\mathrm{ref}}$ between two red dashed lines is the approximated wind strength variation, $\delta V_{W\mathrm{ref}}$. Moreover, it is important to note that we do not assume that the actual wind profile is in either the nominal shape nor any other certain shape.

5.2. Reformulation and linearization of dynamics

The derivative of the state vector with respect to the normalized time τ can be defined using the chain rule as:

$$\boldsymbol{x}' := \frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}\tau} = \frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t}\cdot\frac{\mathrm{d}t}{\mathrm{d}\tau} = t_\mathrm{f}\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} \tag{32}$$

where $\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t}$ is the dynamics presented in Eq. (1), and $\boldsymbol{x}'$ is the reformulated dynamics that are linearized and used in the optimization problem. It can be noticed that $\boldsymbol{x}'$ is a function of $\boldsymbol{x}, \boldsymbol{u}, V_{W\mathrm{ref}}$, and t_f as

$$\boldsymbol{x}'(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), V_{W\mathrm{ref}}, t_\mathrm{f}) = t_\mathrm{f}\,\boldsymbol{f}(\boldsymbol{x}(\tau), \boldsymbol{u}(\tau), V_{W\mathrm{ref}}) \tag{33}$$

Here, using the same numerical integration method as in the previous section, i.e., the trapezoidal rule, a discrete form of system dynamics can be expressed as

$$\boldsymbol{x}_{k+1} = \left(\boldsymbol{x}'_k + \boldsymbol{x}'_{k+1}\right)(\mathrm{d}\tau)/2 + \boldsymbol{x}_k \tag{34}$$

where $k = 1, 2 \ldots, N-1$ are the discrete grid points in the normalized cycle and $\mathrm{d}\tau$ is the normalized time step length. Next, Eq. (34) is linearized with respect to the previous trajectory denoted by superscript p as

$$\begin{aligned}\mathrm{d}\boldsymbol{x}_{k+1} = {} & \frac{\mathrm{d}\tau}{2}\left[\frac{\partial\boldsymbol{x}'}{\partial\boldsymbol{x}}\right]^p_k \mathrm{d}\boldsymbol{x}_k + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial\boldsymbol{x}'}{\partial\boldsymbol{u}}\right]^p_k \mathrm{d}\boldsymbol{u}_k \\ & + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial\boldsymbol{x}'}{\partial V_{W\mathrm{ref}}}\right]^p_k \mathrm{d}V_{W\mathrm{ref}} + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial\boldsymbol{x}'}{\partial t_\mathrm{f}}\right]^p_k \mathrm{d}t_\mathrm{f} \\ & + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial\boldsymbol{x}'}{\partial\boldsymbol{x}}\right]^p_{k+1} \mathrm{d}\boldsymbol{x}_{k+1} + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial\boldsymbol{x}'}{\partial\boldsymbol{u}}\right]^p_{k+1} \mathrm{d}\boldsymbol{u}_{k+1} \\ & + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial\boldsymbol{x}'}{\partial V_{W\mathrm{ref}}}\right]^p_{k+1} \mathrm{d}V_{W\mathrm{ref}} + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial\boldsymbol{x}'}{\partial t_\mathrm{f}}\right]^p_{k+1} \mathrm{d}t_\mathrm{f} + \mathrm{d}\boldsymbol{x}_k \\ & + \boldsymbol{e}_{k+1}\end{aligned} \tag{35}$$

where $\mathrm{d}\boldsymbol{x}, \mathrm{d}\boldsymbol{u}, \mathrm{d}V_{W\mathrm{ref}}$, and $\mathrm{d}t_\mathrm{f}$ are the incremental changes of states, controls, wind strength, and cycle time, respectively, compared to the previous trajectory as

$$\mathrm{d}\boldsymbol{x} := \boldsymbol{x} - \boldsymbol{x}^p \tag{36a}$$

$$\mathrm{d}\boldsymbol{u} := \boldsymbol{u} - \boldsymbol{u}^p \tag{36b}$$

$$\mathrm{d}V_{W\mathrm{ref}} := V_{W\mathrm{ref}} - (V_{W\mathrm{ref}})^p \tag{36c}$$

$$\mathrm{d}t_\mathrm{f} := t_\mathrm{f} - (t_\mathrm{f})^p \tag{36d}$$

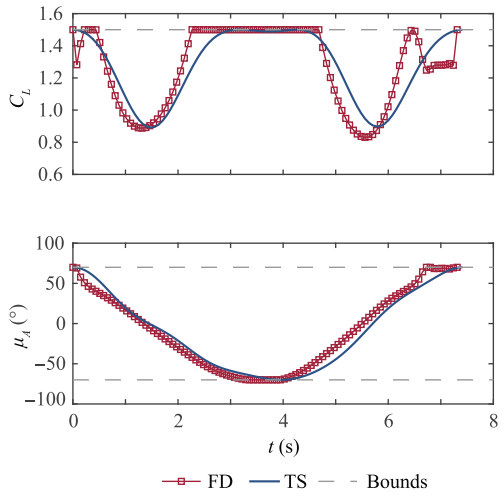

Fig. 4 Offline: Time histories of controls.

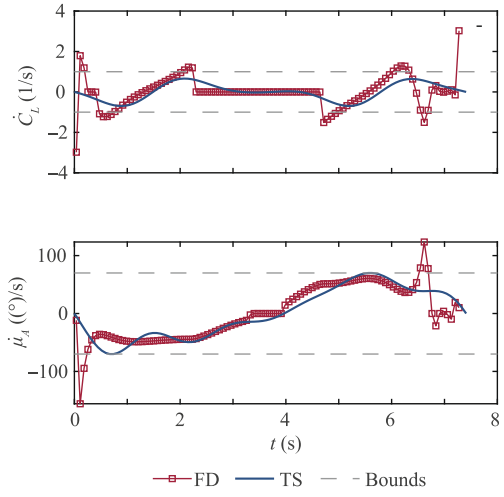

Fig. 5 Offline: Time histories of control derivatives.

and $\boldsymbol{e}$ is the error in this step. This kind of method is referred to as successive linearization[25] and was already successfully applied in other trajectory optimization tasks[26–28]. To make sure that the updated states and controls still adhere to the nonlinear dynamics, the computation of $\boldsymbol{e}$ needs to be accounted for. First, we define the states at $k+1$ obtained via the trapezoidal rule as

$$\tilde{\boldsymbol{x}}_{k+1}^{p} = \left(\left(\boldsymbol{x}_{k}'\right)^{p} + \left(\boldsymbol{x}_{k+1}'\right)^{p}\right)(\mathrm{d}\tau)/2 + \boldsymbol{x}_{k}^{p} \tag{37}$$

Noticing that only in the initial guess provided by the offline simulation, where the trapezoidal rule is used, $\tilde{\boldsymbol{x}}_{k+1}^{p} = \boldsymbol{x}_{k+1}^{p}$. Otherwise, $\boldsymbol{x}_{k+1}^{p}$ is obtained by using Eq. (36a), and thus $\tilde{\boldsymbol{x}}_{k+1}^{p} \neq \boldsymbol{x}_{k+1}^{p}$. Therefore, the expression of $\boldsymbol{e}$ can be given by

$$\boldsymbol{e}_{k+1} = \tilde{\boldsymbol{x}}_{k+1}^{p} - \boldsymbol{x}_{k+1}^{p} \tag{38}$$

Noticing the linearity of the control vector with respect to the series coefficients in Eq. (27), Eq. (35) can be modified as

$$\begin{aligned}
\mathrm{d}\boldsymbol{x}_{k+1} = {} & \frac{\mathrm{d}\tau}{2}\left[\frac{\partial \boldsymbol{x}'}{\partial \boldsymbol{x}}\right]_{k}^{p}\mathrm{d}\boldsymbol{x}_{k} + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial \boldsymbol{x}'}{\partial \boldsymbol{u}}\right]_{k}^{p}\left[(\boldsymbol{S}_M)_{k}\mathrm{d}\boldsymbol{c}\right] \\
& + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial \boldsymbol{x}'}{\partial V_{W\mathrm{ref}}}\right]_{k}^{p}\mathrm{d}V_{W\mathrm{ref}} + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial \boldsymbol{x}'}{\partial t_{\mathrm{f}}}\right]_{k}^{p}\mathrm{d}t_{\mathrm{f}} \\
& + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial \boldsymbol{x}'}{\partial \boldsymbol{x}}\right]_{k+1}^{p}\mathrm{d}\boldsymbol{x}_{k+1} + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial \boldsymbol{x}'}{\partial \boldsymbol{u}}\right]_{k+1}^{p}\left[(\boldsymbol{S}_M)_{k+1}\mathrm{d}\boldsymbol{c}\right] \\
& + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial \boldsymbol{x}'}{\partial V_{W\mathrm{ref}}}\right]_{k+1}^{p}\mathrm{d}V_{W\mathrm{ref}} + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial \boldsymbol{x}'}{\partial t_{\mathrm{f}}}\right]_{k+1}^{p}\mathrm{d}t_{\mathrm{f}} + \mathrm{d}\boldsymbol{x}_{k} \\
& + \boldsymbol{e}_{k+1}
\end{aligned} \tag{39}$$

To serve the purpose of determining unknown variables, Eq. (39) is rearranged. Moving all the terms on the right-hand side except the ones with $dV_{W\mathrm{ref}}$ and $\boldsymbol{e}$ to the left hand side yields

$$\begin{aligned}
& \left(\boldsymbol{I}_n - \frac{\mathrm{d}\tau}{2}\left[\frac{\partial \boldsymbol{x}'}{\partial \boldsymbol{x}}\right]_{k+1}^{p}\right)\mathrm{d}\boldsymbol{x}_{k+1} - \left(\boldsymbol{I}_n + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial \boldsymbol{x}'}{\partial \boldsymbol{x}}\right]_{k}^{p}\right)\mathrm{d}\boldsymbol{x}_{k} \\
& - \frac{\mathrm{d}\tau}{2}\left(\left[\frac{\partial \boldsymbol{x}'}{\partial \boldsymbol{u}}\right]_{k}^{p}(\boldsymbol{S}_M)_{k} + \left[\frac{\partial \boldsymbol{x}'}{\partial \boldsymbol{u}}\right]_{k+1}^{p}(\boldsymbol{S}_M)_{k+1}\right)\mathrm{d}\boldsymbol{c} \\
& - \frac{\mathrm{d}\tau}{2}\left(\left[\frac{\partial \boldsymbol{x}'}{\partial t_{\mathrm{f}}}\right]_{k}^{p} + \left[\frac{\partial \boldsymbol{x}'}{\partial t_{\mathrm{f}}}\right]_{k+1}^{p}\right)\mathrm{d}t_{\mathrm{f}} \\
& = \frac{\mathrm{d}\tau}{2}\left(\left[\frac{\partial \boldsymbol{x}'}{\partial V_{W\mathrm{ref}}}\right]_{k}^{p} + \left[\frac{\partial \boldsymbol{x}'}{\partial V_{W\mathrm{ref}}}\right]_{k+1}^{p}\right)\mathrm{d}V_{W\mathrm{ref}} + \boldsymbol{e}_{k+1}
\end{aligned} \tag{40}$$

The relationships for all grid points, $k = 1, 2, \ldots, N-1$, can be expressed using Eq. (40). Considering

$$\left[\boldsymbol{F}_{\boldsymbol{x}}^{-}\right]_{k} = \boldsymbol{I}_n - \frac{\mathrm{d}\tau}{2}\left[\frac{\partial \boldsymbol{x}'}{\partial \boldsymbol{x}}\right]_{k}^{p} \tag{41a}$$

$$\left[\boldsymbol{F}_{\boldsymbol{x}}^{+}\right]_{k} = \boldsymbol{I}_n + \frac{\mathrm{d}\tau}{2}\left[\frac{\partial \boldsymbol{x}'}{\partial \boldsymbol{x}}\right]_{k}^{p} \tag{41b}$$

$$\left[\boldsymbol{F}_{\boldsymbol{c}}\right]_{k} = \frac{\mathrm{d}\tau}{2}\left(\left[\frac{\partial \boldsymbol{x}'}{\partial \boldsymbol{u}}\right]_{k}^{p}(\boldsymbol{S}_M)_{k} + \left[\frac{\partial \boldsymbol{x}'}{\partial \boldsymbol{u}}\right]_{k+1}^{p}(\boldsymbol{S}_M)_{k+1}\right) \tag{41c}$$

$$\left[\boldsymbol{F}_{t_{\mathrm{f}}}\right]_{k} = \frac{\mathrm{d}\tau}{2}\left(\left[\frac{\partial \boldsymbol{x}'}{\partial t_{\mathrm{f}}}\right]_{k}^{p} + \left[\frac{\partial \boldsymbol{x}'}{\partial t_{\mathrm{f}}}\right]_{k+1}^{p}\right) \tag{41d}$$

$$\left[\boldsymbol{F}_{V_{W\mathrm{ref}}}\right]_{k} = \frac{\mathrm{d}\tau}{2}\left(\left[\frac{\partial \boldsymbol{x}'}{\partial V_{W\mathrm{ref}}}\right]_{k}^{p} + \left[\frac{\partial \boldsymbol{x}'}{\partial V_{W\mathrm{ref}}}\right]_{k+1}^{p}\right) \tag{41e}$$

the relationships can be written in a compact manner as:

$$\begin{aligned}
& \begin{bmatrix} \left[\boldsymbol{F}_{\boldsymbol{x}}^{-}\right]_2 & 0 & \cdots & 0 \\ 0 & \left[\boldsymbol{F}_{\boldsymbol{x}}^{-}\right]_3 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \left[\boldsymbol{F}_{\boldsymbol{x}}^{-}\right]_N \end{bmatrix} \begin{bmatrix} \mathrm{d}\boldsymbol{x}_2 \\ \mathrm{d}\boldsymbol{x}_3 \\ \vdots \\ \mathrm{d}\boldsymbol{x}_N \end{bmatrix} \\
& - \begin{bmatrix} \left[\boldsymbol{F}_{\boldsymbol{x}}^{+}\right]_1 & 0 & \cdots & 0 \\ 0 & \left[\boldsymbol{F}_{\boldsymbol{x}}^{+}\right]_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \left[\boldsymbol{F}_{\boldsymbol{x}}^{+}\right]_{N-1} \end{bmatrix} \begin{bmatrix} \mathrm{d}\boldsymbol{x}_1 \\ \mathrm{d}\boldsymbol{x}_2 \\ \vdots \\ \mathrm{d}\boldsymbol{x}_{N-1} \end{bmatrix} - \begin{bmatrix} \left[\boldsymbol{F}_{\boldsymbol{c}}\right]_1 \\ \left[\boldsymbol{F}_{\boldsymbol{c}}\right]_2 \\ \vdots \\ \left[\boldsymbol{F}_{\boldsymbol{c}}\right]_{N-1} \end{bmatrix}\mathrm{d}\boldsymbol{c} \\
& - \begin{bmatrix} \left[\boldsymbol{F}_{t_{\mathrm{f}}}\right]_1 \\ \left[\boldsymbol{F}_{t_{\mathrm{f}}}\right]_2 \\ \vdots \\ \left[\boldsymbol{F}_{t_{\mathrm{f}}}\right]_{N-1} \end{bmatrix}\mathrm{d}t_{\mathrm{f}} = \begin{bmatrix} \left[\boldsymbol{F}_{V_{W\mathrm{ref}}}\right]_1 \\ \left[\boldsymbol{F}_{V_{W\mathrm{ref}}}\right]_2 \\ \vdots \\ \left[\boldsymbol{F}_{V_{W\mathrm{ref}}}\right]_{N-1} \end{bmatrix}\mathrm{d}V_{W\mathrm{ref}} + \begin{bmatrix} \boldsymbol{e}_2 \\ \boldsymbol{e}_3 \\ \vdots \\ \boldsymbol{e}_N \end{bmatrix}
\end{aligned} \tag{42}$$

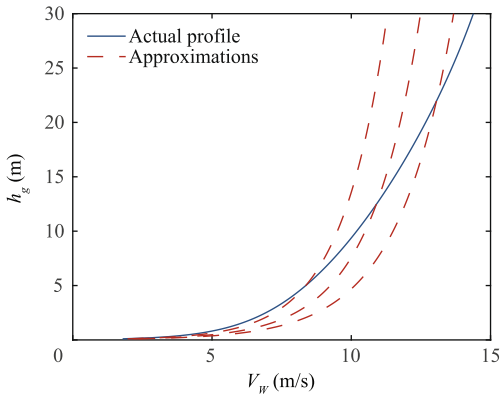

Fig. 6 Approximations of wind profile.[18]

The above equation can be further modified as

$$\boldsymbol{M}\boldsymbol{z}_o = \boldsymbol{v} + \boldsymbol{e} \tag{43}$$

where

$$\boldsymbol{M} = \begin{bmatrix} -[\boldsymbol{F}_{\boldsymbol{x}}^{+}]_1 & [\boldsymbol{F}_{\boldsymbol{x}}^{-}]_2 & 0 & \cdots & 0 & -[\boldsymbol{F}_{\boldsymbol{c}}]_1 & -[\boldsymbol{F}_{t_{\mathrm{f}}}]_1 \\ 0 & -[\boldsymbol{F}_{\boldsymbol{x}}^{+}]_2 & [\boldsymbol{F}_{\boldsymbol{x}}^{-}]_3 & \cdots & 0 & -[\boldsymbol{F}_{\boldsymbol{c}}]_2 & -[\boldsymbol{F}_{t_{\mathrm{f}}}]_2 \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & -[\boldsymbol{F}_{\boldsymbol{x}}^{+}]_{N-1} & [\boldsymbol{F}_{\boldsymbol{x}}^{-}]_N & -[\boldsymbol{F}_{\boldsymbol{c}}]_{N-1} & -[\boldsymbol{F}_{t_{\mathrm{f}}}]_{N-1} \end{bmatrix} \tag{44a}$$

$$\boldsymbol{z}_o = \left[(\mathrm{d}\boldsymbol{x}_1)^T, (\mathrm{d}\boldsymbol{x}_2)^{\mathrm{T}}, \ldots, (\mathrm{d}\boldsymbol{x}_N)^{\mathrm{T}}, (\mathrm{d}\boldsymbol{c})^{\mathrm{T}}, \mathrm{d}t_{\mathrm{f}}\right]^{\mathrm{T}} \tag{44b}$$

$$\boldsymbol{v} = \begin{bmatrix} [\boldsymbol{F}_{V_{W\mathrm{ref}}}]_1 \\ [\boldsymbol{F}_{V_{W\mathrm{ref}}}]_2 \\ \vdots \\ [\boldsymbol{F}_{V_{W\mathrm{ref}}}]_{N-1} \end{bmatrix} \mathrm{d}V_{W\mathrm{ref}} \tag{44c}$$

$$\boldsymbol{e} = \left[(\boldsymbol{e}_2)^{\mathrm{T}}, (\boldsymbol{e}_3)^{\mathrm{T}}, \ldots, (\boldsymbol{e}_N)^{\mathrm{T}}\right]^{\mathrm{T}} \tag{44d}$$

Here, $\boldsymbol{z}_o$ from the left-hand side consists of all unknown increments that are used for correcting the flight trajectories. On the right-hand side, there are two terms, namely, $\boldsymbol{v}$ and $\boldsymbol{e}$, representing the increments in states caused by the wind strength variation and the linearization errors, respectively. We assume that the wind strength variation between two consecutive steps is small, as the step itself is small. Therefore, $dV_{W\mathrm{ref}}$ is considered to be approximately equal to the wind strength variation obtained in Section 5.1 as

$$\mathrm{d}V_{W\mathrm{ref}} \approx \delta V_{W\mathrm{ref}} \tag{45}$$

Taking into account the nonlinear EoM, the linearized equation in Eq. (43) specifies the relationship between the wind strength variation and the increments of states, series coefficients, and the final time. This equation is considered in the QP formulation detailed in the next section.

5.3. Quadratic programming formulation

The QP problem is formulated for correcting the flight trajectory online based on the initial solution, which is the energy-neutral trajectory that has been shown in Section 4. The core objective is to determine the increment variables and update the cycle time and the associated controls for performing appropriate dynamic soaring in an unspecified wind profile. To do this, at first, the constraints in Section 2 are converted to expressions with respect to the unknown variables. After that, some constraints that facilitate the online implementation are given.

The current time step is defined as $k_{\mathrm{c}} \in \{1, 2, 3, \ldots, N-1\}$, being any step within a cycle. The optimization formulation is applied to a shrinking horizon. First, the terminal constraint that specifies the final altitude and the velocity direction in Eq. (17) is modified into

$$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\mathrm{d}\boldsymbol{x}_N = \boldsymbol{y}_{\mathrm{f}} - \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}\mathrm{d}\boldsymbol{x}_N^p \tag{46}$$

Next, the boundary constraints of controls and control derivatives are reformulated. As the initial guess satisfies these constraints, the increments being zero guarantee the satisfaction of the constraints in Eqs. (20) and (21). Moreover, as controls are parameterized as trigonometric series, the constraints are on the increment of series coefficients as

$$\mathrm{d}\boldsymbol{u}_N = (\boldsymbol{S}_M)_N \mathrm{d}\boldsymbol{c} = \boldsymbol{0} \tag{47}$$

$$\mathrm{d}\dot{\boldsymbol{u}}_N = \left[\frac{\mathrm{d}\boldsymbol{S}_M}{\mathrm{d}\tau}\right]_N \mathrm{d}\boldsymbol{c} = \boldsymbol{0} \tag{48}$$

Then, the inequality constraints in Eq. (19) are modified. The minimum altitude constraint can be expressed by

$$\frac{\partial h_g}{\partial \boldsymbol{x}}\boldsymbol{x}_k > h_{g\min} - \frac{\partial h_g}{\partial \boldsymbol{x}}\boldsymbol{x}_k^p, \quad k = k_{\mathrm{c}}, k_{\mathrm{c}}+1, \ldots, N-1 \tag{49}$$

The path constraints in Eqs. (23) and (22) for $k = k_{\mathrm{c}}, k_{\mathrm{c}}+1, \ldots, N$ are rewritten as

$$\boldsymbol{u}_{\min} - \boldsymbol{u}_k^p \leqslant \mathrm{d}\boldsymbol{u}_k = [\boldsymbol{S}_M]_k \mathrm{d}\boldsymbol{c} \leqslant \boldsymbol{u}_{\max} - \boldsymbol{u}_k^p \tag{50}$$

$$\dot{\boldsymbol{u}}_{\min} - \dot{\boldsymbol{u}}_k^p \leqslant \mathrm{d}\dot{\boldsymbol{u}}_k \approx \frac{1}{t_{\mathrm{f}}^p}\left[\frac{\mathrm{d}\boldsymbol{S}_M}{\mathrm{d}\tau}\right]_k \mathrm{d}\boldsymbol{c} \leqslant \dot{\boldsymbol{u}}_{\max} - \dot{\boldsymbol{u}}_k^p \tag{51}$$

The online constraints include an equality that regulates the current states, equality constraints that ensure the smoothness of the control histories, and an inequality constraint that minimize the deviation of the terminal velocity to the initial velocity. The state constraints are given as

$$d\boldsymbol{x}_{k_{\mathrm{c}}} = \boldsymbol{x}_{k_{\mathrm{c}}}^a - \boldsymbol{x}_{k_{\mathrm{c}}}^p \tag{52}$$

where $\boldsymbol{x}_{k_{\mathrm{c}}}^a$ is the actual states at $k = k_{\mathrm{c}}$. This constraint guarantees that the actual location and velocity are on the updated trajectory. As the control histories are parameterized as trigonometric series, they are inherently smooth. However, to ensure that the controls are still continuous with respect to the previous and future control histories after the update of trajectory, the following constraints are enforced:

$$\mathrm{d}\boldsymbol{u}_{k_{\mathrm{c}}} = (\boldsymbol{S}_M)_{k_{\mathrm{c}}} \mathrm{d}\boldsymbol{c} = \boldsymbol{0} \tag{53a}$$

$$\mathrm{d}\boldsymbol{u}_{k_{\mathrm{c}}+1} = (\boldsymbol{S}_M)_{k_{\mathrm{c}}+1} \mathrm{d}\boldsymbol{c} = \boldsymbol{0} \tag{53b}$$

For facilitating repeated cycles of dynamic soaring, the final speed is designed to be close to the initial speed. An inequality constraint on u_g is formulated as

$$\left|u_{g_N} - u_{g_{\mathrm{i}}}\right| < \sigma \tag{54}$$

where $u_{g_{\mathrm{i}}}$ is the initial value, and σ is a new unknown variable and a positive penalty variable to be minimized:

$$\sigma > 0 \tag{55}$$

As a shrinking horizon is concerned, the vector is defined as

$$\boldsymbol{z}_{\mathrm{c}} = \left[(\mathrm{d}\boldsymbol{x}_{k_{\mathrm{c}}})^{\mathrm{T}}, \ldots, (\mathrm{d}\boldsymbol{x}_N)^{\mathrm{T}}, (\mathrm{d}\boldsymbol{c})^{\mathrm{T}}, \mathrm{d}t_{\mathrm{f}}\right]^{\mathrm{T}} \tag{56}$$

Therefore, the total unknown variable vector to be determined is defined as

$$\boldsymbol{z} = \left[(\boldsymbol{z}_c)^{\mathrm{T}}, \sigma\right]^{\mathrm{T}} \tag{57}$$

Recall that the existing variables in $\boldsymbol{z}_o$ are all incremental changes and expected to be minimized. Therefore, a quadratic cost function is selected to minimize all unknowns as:

$$J = (\boldsymbol{z})^{\mathrm{T}}\boldsymbol{H}\boldsymbol{z} \tag{58}$$

where $\boldsymbol{H}$ is a positive definite weighting matrix.

$$\begin{aligned} &\underset{\boldsymbol{z}}{\text{minimize}}\, J = (\boldsymbol{z})^{\mathrm{T}}\boldsymbol{H}\boldsymbol{z} \\ &\text{subject to } \boldsymbol{M}[k_c : N-1]\boldsymbol{z}_c = \boldsymbol{v}[k_c : N-1] + \boldsymbol{e}[k_c : N-1] \\ &\qquad \text{Constraints in Eqs. (46)–(55)} \end{aligned} \tag{59}$$

where $[k_c : N-1]$ denotes the rows from k_c to $N-1$. It can be noted that the optimization problem in Eq. (59) is of a strictly convex QP form. Therefore, it can be efficiently solved by a dedicated solver. Moreover, the QP problem is solved in between every two consecutive time steps to correct/update the flight trajectory, which does not rely on iterations. The needed information consists of the previous trajectory, the actual states at the current time step, and the wind strength variation.

6. Numerical results of online trajectory correction

In this section, simulation studies are made for demonstrating the effectiveness of the proposed methods and results are presented. The simulation parameters are the same as in the previous Section 4. Time-varying wind profiles are generated for simulation purpose. We do not assume that the wind profile is always strong enough to support an energy-neutral dynamic soaring cycle. The used profiles may not coincide with the real-world patterns of the shear wind, but it is rational to verify the proposed method in randomly generated scenarios.

6.1. Single-cycle flight

This section presents the simulation results of a single cycle flight for a close observation of the flight. The actual wind profile is designed as

$$V_W = \exp\left(-\frac{x_g}{50 + h_g}\right) V_{W\mathrm{ref}} \frac{\ln\left(h_g/h_0\right)}{\ln\left(h_{\mathrm{ref}}/h_0\right)} \tag{60}$$

At $x_g = 0$, the actual profile is equivalent to the minimum-strength nominal profile. Moreover, except the initial time, the profile is not of the nominal shape. The overall strength decreases as x_g increases. The profiles are visualized in Fig. 7. The opaque surface is the actual wind profile, while the transparent surface represents the minimum-strength nominal profile, which also shows the actual flight trajectory and the initial trajectory. The actual flight trajectory deviates from the initial trajectory and is generally lower, which can be checked from the time histories of h_g shown in Fig. 8. Fig. 8 also displays the time histories of the inertial speed. The terminal speed are 19.48 m/s and 20 m/s for the actual and initial trajectories, respectively. Therefore, the actual trajectory still makes good use of the wind energy to support the non-powered flight, although the actual wind profile is not strong enough to support an energy-neutral cycle. Moreover, seen from the histories of the inertial velocity in Fig. 9, the terminal velocity satisfies the terminal output constraints, as the terminal values of v_g and w_g are both zero. The cycle times for the actual and initial trajectories are 6.91 s and 7.41 s, respectively.

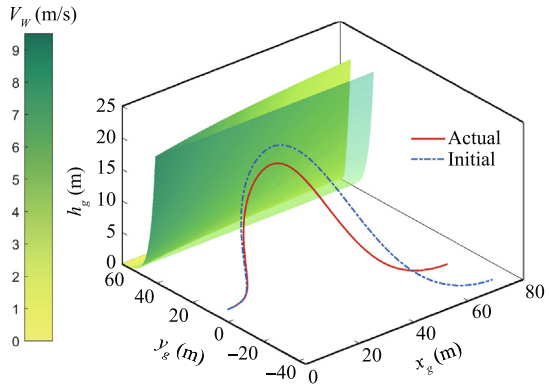

Fig. 7 Single-cycle: Trajectories and wind profiles.

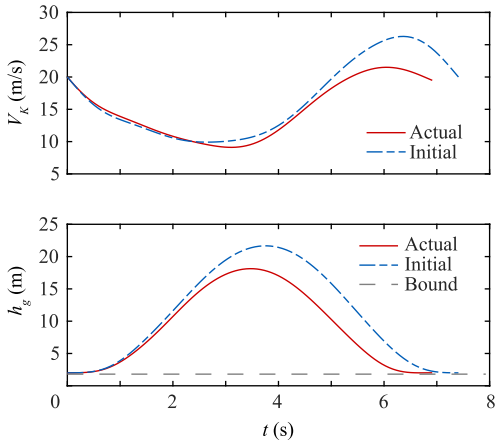

Fig. 8 Single-cycle: Time histories of speed and altitude.

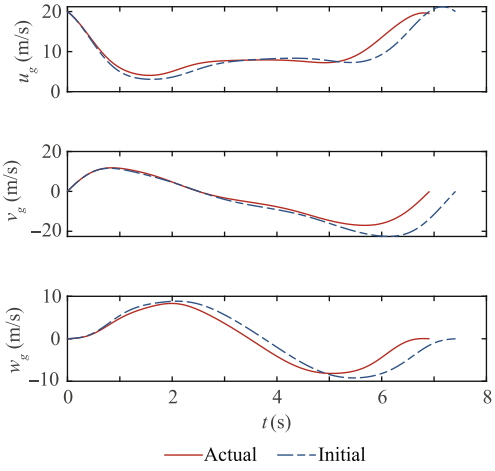

Fig. 9 Single-cycle: Time histories of inertial velocity.

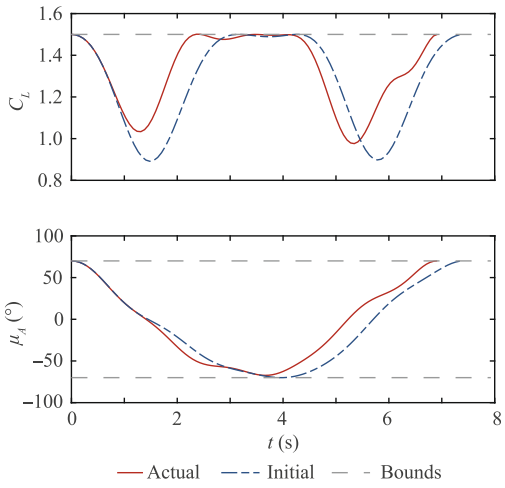

Fig. 10 Single-cycle: Time histories of controls.

A shorter cycle is generated online compared to the initial solution. The control histories are shown in Fig. 10, and the histories of the control derivatives are embodied in Fig. 11. The boundary constraints are all met, facilitating good connections of cycles. The path constraints are satisfied as well. The continuity of control derivatives suggests that the smoothness of controls are retained in the online updating process. Solving a QP problem on a Windows platform with an i7-6700 and 16 GB of RAM using OSQP [29], which is a dedicated convex QP problem solver, in a Matlab environment takes no more than 0.005 s, whereas the time step size is around 0.07 s.

6.2. Multiple-cycle flight

In this section, a simulation is carried out to verify the effectiveness of the proposed method in a multiple-cycle dynamic soaring flight. According to Ref. 21, when all measurements are used, an accurate wind estimation can be achieved. However, to demonstrate the effectiveness of the proposed method in a practical scenario in this section, we consider the wind measurement error as a normally distributed uncertainty ($3\sigma = 10\%$). The wind profile is expressed by

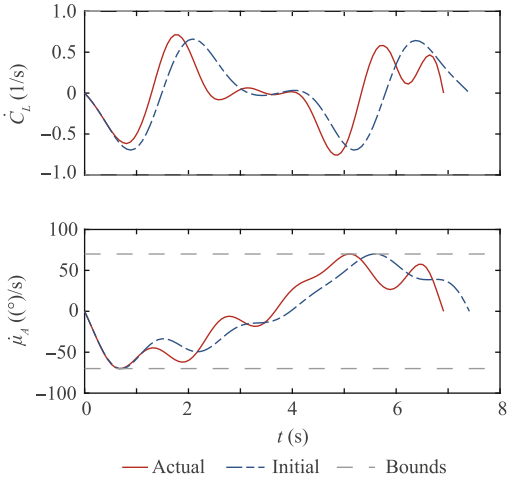

Fig. 11 Single-cycle: Time histories of control derivatives.

$$V_W = \left(1 + 0.4\sin\left(\frac{-x_g}{50}\right)\right) V_{W\mathrm{ref}} \frac{\ln\left(h_g/h_0\right)}{\ln\left(h_{\mathrm{ref}}/h_0\right)} \tag{61}$$

and visualized in Fig. 12, in which the flight trajectory is also shown. Fig. 13 depicts the time histories of inertial speed and altitude. Cycles are distinguished by the background grids. The cycle times are 7.19 s, 7.18 s, 7.59 s, 7.87 s, and 7.32 s. In combination with the wind profile seen in Fig. 12, one can notice that generally, a dynamic soaring cycle is shorter and has a smaller change in altitude when the wind is weaker and vice versa. The velocity components are embodied in Fig. 14, showing that the periodical dynamic soaring cycles

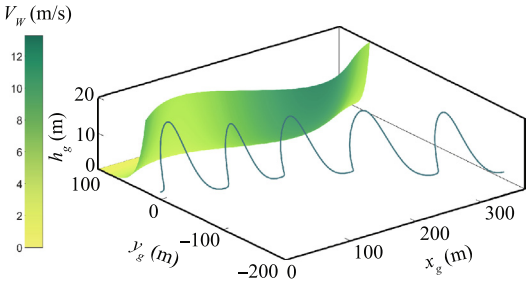

Fig. 12 Multi-cycle: Trajectory and wind profile.

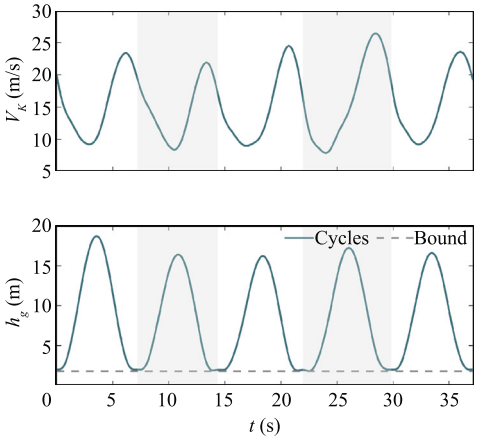

Fig. 13 Multi-cycle: Time histories of speed and altitude.

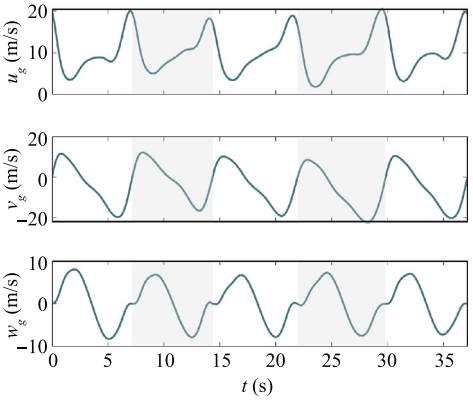

Fig. 14 Multi-cycle: Time histories of inertial velocity.

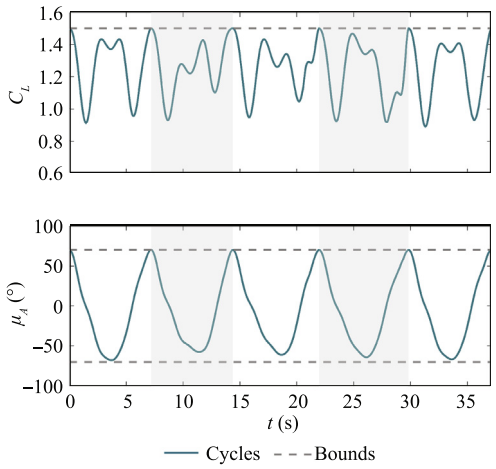

Fig. 15 Multi-cycle: Time histories of controls.

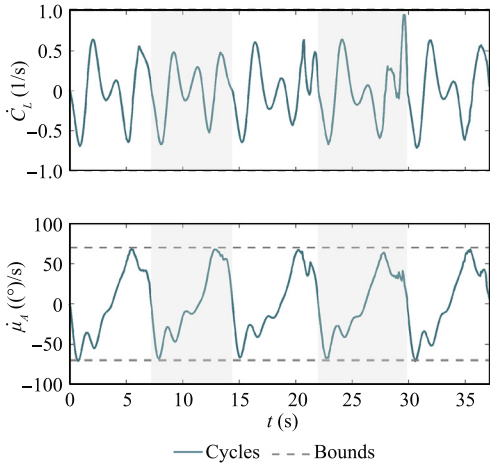

Fig. 16 Multi-cycle: Time histories of control derivatives.

are performed with slight differences due to the wind variation. The control histories, shown in Fig. 15, exhibit some differences among cycles. This is because the controls, especially the lift coefficients, need to be adjusted according to the wind strength variation, in order to make good use of the wind gradients. Finally, the time histories of the control derivatives are shown in Fig. 16. Control derivatives remain bounded. The oscillations are inevitable owing to the wind variation and the measurement errors. Consequently, the proposed method still performs well in the presence of wind measurement uncertainties.

7. Conclusions

This work investigated the smooth trajectory generation for dynamic soaring maneuvers. The introduction of the trigonometric series parameterization significantly enhances the smoothness of controls with a fair trade-off in terms of the minimum required wind strength for performing an energy-neutral cycle. The expression of the trigonometric series also by construction guarantees the connection of control inputs, simplifying the optimization formulation. The online optimization-based strategy utilizes the wind strength variation approximated by the incremental change of a reference wind strength to correct the flight trajectory, featuring a free cycle time. The formulated quadratic programming problem can be very efficiently solved. The effectiveness of the proposed method is verified in scenarios where single and multiple dynamic soaring cycles are performed while the wind profiles are unspecified.

Future works include: (A) the study with more realistic wind field modeling, (B) the development and utilization of a six degrees-of-freedom dynamic model, (C) powered dynamic soaring, and (D) hardware-in-the-loop simulations. Furthermore, the control algorithm will be utilized in the scope of a research project on the application of dynamic soaring for UAVs.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by the TUM University Foundation Fellowship and in part by the German Federal Ministry for Economic Affairs and Energy (BMWi) within the Federal Aeronautical Research Program LuFo VI-1 through Project "RAUDY" (No. 20E1910B).

References

1. Rayleigh L. The soaring of birds. *Nature* 1883;**27**(701):534–5.
2. Idrac P. Experimental study of the "soaring of albatrosses". *Nature* 1925;**115**(2893):532–42.
3. Prandtl L. Beobachtungen über den dynamischen Segelflug [German]. *Zeitschrift für Flugtechnik und Motorluftschifffahrt* 1930;**21**(5):116.
4. Pennycuick CJ. The flight of petrels and albatrosses (Procellariiformes), observed in South Georgia and its vicinity. *Philos Trans R Soc London B, Biol Sci* 1982;**300**(1098):75–106.
5. Sachs G, Traugott J, Nesterova AP, et al. Flying at no mechanical energy cost: Disclosing the secret of wandering albatrosses. *PloS One* 2012;**7**(9):e41449.
6. Sachs G. Minimum shear wind strength required for dynamic soaring of albatrosses. *Ibis* 2005;**147**(1):1–10.
7. Bousquet GD, Triantafyllou MS, Slotine JJE. Optimal dynamic soaring consists of successive shallow arcs. *J R Soc Interface* 2017;**14**(135):20170496.
8. Sachs G. Kinetic energy in dynamic soaring–inertial speed and airspeed. *J Guid Control Dyn* 2019;**42**(8):1812–21.
9. Sachs G, Lesch K. Optimal periodic trajectories of aircraft with singular control. *IFAC Proc* 1990;**23**(8):87–92.
10. Sachs G, Grüter B, Hong HC. Performance enhancement by wing sweep for high-speed dynamic soaring. *Aerospace* 2021;**8**(8):229.
11. Sachs G, Grüter B, Hong HC. Dynamic soaring supported by engine power in poor wind conditions. *AIAA Scitech 2021 Forum;* 2021 Jan 11–15 21; Virtual Event. Reston: AIAA; 2021. p. 0015.
12. Lawrance NR, Sukkarieh S. A guidance and control strategy for dynamic soaring with a gliding UAV. *2009 IEEE International*

Conference on Robotics and Automation. 2009 May 12-17; Kobe, Japan. Piscataway: IEEE Press. p. 3632–7.
13. Deittert M, Richards A, Toomer CA, et al. Engineless unmanned aerial vehicle propulsion by dynamic soaring. *J Guid Control Dyn* 2009;**32**(5):1446–57.
14. Liu DN, Hou ZX, Guo Z, et al. Optimal patterns of dynamic soaring with a small unmanned aerial vehicle. *Proc IMechE Part G J Aerospace Eng* 2017;**231**(9):1593–608.
15. Long N, Watkins S, Moschetta JM, et al. Regenerative dynamic soaring trajectory augmentation over flat terrains. *AIAA Scitech 2019 Forum;* 2019 Jan 7-11; San Diego, USA.Reston: AIAA; 2019. p. 0569.
16. Bird JJ, Langelaan JW, Montella C, et al. Closing the loop in dynamic soaring. *AIAA Guidance, Navigation, and Control Conference*; 2014 Jan 13–17; National Harbor, USA. Reston: AIAA; 2014. p. 0263.
17. Li ZD, Langelaan JW. Parameterized trajectory planning for dynamic soaring. *AIAA Scitech 2020 Forum;* 2020 Jan 6-10; Orlando, USA. Reston: AIAA; 2020. p. 0856.
18. Hong HC, Zheng HX, Holzapfel F, et al. Dynamic soaring in unspecified wind shear: A real-time quadratic-programming approach. *2019 27th Mediterranean Conference on Control and Automation (MED);* 2019 Jul 1-4; Akko, Israel. Piscataway: IEEE Press; 2019. p. 600–5.
19. Hong HC, Piprek P, Afonso RJM, et al. Trigonometric series-based smooth flight trajectory generation. *IEEE Trans Aerosp Electron Syst* 2021;**57**(1):721–8.
20. Rieck M, Bittner M, Grüter B, et al. FALCON.m user guide. Institute of Flight System Dynamics, Technical University of Munich, 2020. Available from: url:www.falcon-m.com..
21. Rhudy MB, Gu Y, Gross JN, et al. Onboard wind velocity estimation comparison for unmanned aircraft systems. *IEEE Trans Aerosp Electron Syst* 2017;**53**(1):55–66.
22. Tian PZ, Chao HY, Rhudy M, et al. Wind sensing and estimation using small fixed-wing unmanned aerial vehicles: A survey. *J Aerosp Inf Syst* 2021;**18**(3):132–43.
23. Bronz M, Gavrilovic N, Drouin A, et al. Flight testing of dynamic soaring part-1: Leeward inclined circle trajectory. *AIAA Scitech 2021 Forum;* 2021 Jan 11–15 21; Virtual Event.Reston: AIAA; 2021. p. 1527.
24. Sun K, Regan CD, Egziabher DG. GNSS/INS based estimation of air data and wind vector using flight maneuvers. *2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*. 2018 Apr 23-26; Monterey, USA. Piscataway: IEEE Press; 2018. p. 838–49.
25. Mao YQ, Szmuk M, Açıkmeşe B. Successive convexification of non-convex optimal control problems and its convergence properties, *2016 IEEE 55th Conference on Decision and Control (CDC)*. 2016 Dec 12-14; Las Vegas, USA. Piscataway: IEEE Press; 2016. p. 3636–41.
26. Wang ZB, Grant MJ. Constrained trajectory optimization for planetary entry via sequential convex programming. *J Guid Control Dyn* 2017;**40**(10):2603–15.
27. Szmuk M, Reynolds TP, Açıkmeşe B. Successive convexification for real-time six-degree-of-freedom powered descent guidance with state-triggered constraints. *J Guid Control Dyn* 2020;**43**(8):1399–413.
28. Hong HC, Piprek P, Gerdts M, et al. Computationally efficient trajectory generation for smooth aircraft flight level changes. *J Guid Control Dyn* 2021;**44**(8):1532–40.
29. Stellato B, Banjac G, Goulart P, et al. OSQP: An operator splitting solver for quadratic programs. *Math Program Comput* 2020;**12**(4):637–72.