TUDelft

Delft University of Technology

AOSoar
Autonomous Orographic Soaring of a Micro Air Vehicle

Hwang, Sunyou; Remes, Bart D.W.; De Croon, Guido C.H.E.

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

AOSoar: Autonomous Orographic Soaring of a Micro Air Vehicle

Sunyou Hwang[1], and Bart D. W. Remes[1], Guido C. H. E. de Croon[1]

Abstract— Utilizing wind hovering techniques of soaring birds can save energy expenditure and improve the flight endurance of micro air vehicles (MAVs). Here, we present a novel method for fully autonomous orographic soaring without a priori knowledge of the wind field. Specifically, we devise an Incremental Nonlinear Dynamic Inversion (INDI) controller with control allocation, adapting it for autonomous soaring. This allows for both soaring and the use of the throttle if necessary, without changing any gain or parameter during the flight. Furthermore, we propose a simulated-annealing-based optimization method to search for soaring positions. This enables for the first time an MAV to autonomously find a feasible soaring position while minimizing throttle usage and other control efforts. Autonomous orographic soaring was performed in the wind tunnel. The wind speed and incline of a ramp were changed during the soaring flight. The MAV was able to perform autonomous orographic soaring for flight times of up to 30 minutes. The mean throttle usage was only 0.25% for the entire soaring flight, whereas normal powered flight requires 38%. Also, it was shown that the MAV can find a new soaring spot when the wind field changes during the flight.
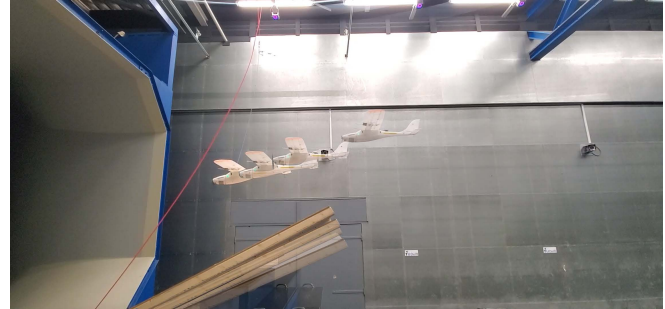
Fig. 1. Autonomous orographic soaring of a micro air vehicle. Without a priori knowledge of the wind field, the MAV successfully performs autonomous soaring in the wind tunnel using little to zero throttle throughout the flight. The MAV can find a new position to soar when environmental conditions vary without any human intervention or manual parameter changes. In this picture, the increasing slope angle changes the wind field, and the MAV autonomously finds a new soaring position. It moves to the front and downward because of the combination of changes in the updraft and the MAV's sink rate.

I. INTRODUCTION

Flight endurance is one of the major factors holding back the real-world application of micro air vehicles (MAVs). For low size, weight, and power (SWaP) MAVs, flight endurance mainly depends on the power density of the battery, which is limited [1], [2], without fundamental progress on the horizon. One way to improve flight endurance is to exploit energy from the environment. Birds like albatrosses, vultures, ospreys, and kestrels are well known for their ability to actively use the wind to minimize their energy expenditure to fly longer distances or time [3], [4], [5]. For example, vultures utilize energy from rising air columns created by uneven ground heating, called thermals [6]. Thermal soaring of unmanned aerial vehicles (UAVs) has been studied in various aspects, not only through manually developed guidance and control strategies [7], [8] but also through reinforcement learning of detecting and exploiting thermals [9], [10].

Another type of soaring is orographic soaring. Kestrels are often observed hovering at a position over a dune without flapping their wings, which is called wind-hovering [11]. This is a good example of the orographic soaring, using the updraft generated by obstacles such as hills, mountains, and buildings when the wind hits the obstacle. Wind-hovering can be useful for remaining in a single place for observation,

but also for prolonging the flight range. For example, gulls appear to plan their path to exploit orographic soaring on the way to their destination to save energy [12], [13]. In this paper, we solely focus on orographic soaring.

Wind fields around obstacles and flight conditions for orographic soaring were analyzed using simulations and measurement data [14], [15], [16], [17]. These studies introduced unique opportunities and possibilities for exploiting orographic updrafts using MAVs. The feasibility and strategies of orographic soaring were also discussed. However, there was no actual flight demonstration performed in these studies.

Orographic soaring of MAVs in a real-world environment has been demonstrated in only a few studies [18], [19]. However, the method in [18] was based on a priori knowledge of the entire wind field to generate a pre-defined trajectory to the soaring spot, while the method in [19] required manually positioning the MAV at a precise initial soaring position before switching on the autonomous soaring controller. To achieve a fully autonomous soaring, considerable challenges remain. In practice, accurately predicting or measuring the wind field is not feasible. Moreover, the MAV has to be able to explore and look for a feasible soaring position autonomously.

In this paper, we demonstrate for the first time the autonomous orographic soaring of an MAV without a priori knowledge of the wind field nor precise initial positioning of the MAV by a human pilot. To achieve this, we present (i) a local search algorithm to find a soaring position and (ii) an INDI controller with control allocation that enables the MAV

[1] All authors are with the MAVLab, Department of Control and Operations, Faculty of Aerospace Engineering, Delft University of Technology, 2629HS Delft, the Netherlands (email: S.Hwang-1@tudelft.nl, B.D.W.Remes@tudelft.nl, G.C.H.E.deCroon@tudelft.nl)

This paper has a supplementary video of flight tests and a replay of the flight log available at http://ieeexplore.ieee.org, provided by the authors.

to use the same controller setting during the entire flight. An important advantage of the proposed control method is that the MAV does not need to switch controllers between soaring and navigation and can use the throttle whenever necessary. We demonstrate the proposed method with a real-world flight in a wind tunnel. Moreover, we validate the versatility of the proposed methods by changing the wind speed and updraft during the autonomous soaring flight.

The paper is structured as follows: In section II, an INDI-based soaring controller and searching method are presented. In section III, we introduce the MAV and the wind tunnel test setups. In section IV, the results from an autonomous soaring flight in the wind tunnel are presented. We discuss the flight test in section V. Finally, we draw conclusions and suggest future research directions in section VI.

II. METHODS

There are many challenges for autonomous orographic soaring. In this article, we focused on two main aspects.

The first one is the controller. There is a unique challenge for orographic soaring because the aim is to maintain the position without using the throttle. Also, it is only feasible in a small updraft region of the wind field with high wind speed. Previous studies used a glider plane without a motor or changed controller or gains when the MAV enters the soaring mode. However, for sake of control fluidity, it would be desirable to use a single controller. We adopted INDI with control allocation to enable using a single controller during the whole flight, regardless of navigation or soaring. We present our control method in section II-A.

The second aspect is to find feasible soaring positions. For fully autonomous flight, the MAV should be able to find where it can soar. In previous studies, it was determined either from a priori knowledge of the wind field or by a human pilot. We present a method for the MAV to autonomously find feasible soaring positions based on simulated annealing in section II-B.

A. INDI controller with allocation

Traditional PID controllers were used in most previous research, however, many of the authors have mentioned the need for a more advanced controller, especially for gust rejection. Therefore, we adopted an INDI controller for soaring flights. INDI is good at disturbance rejection and requires little model information [20], [21], [22]. INDI is an incremental form of nonlinear dynamic inversion. It controls angular acceleration $\dot{\omega}$ in an incremental way. The only required knowledge is the control effectiveness G, mapping an increment in the control input u to a resulting rotational acceleration increment $\dot{\omega} - \dot{\omega}_0$:

$$\begin{aligned} \dot{\omega} &= \dot{\omega}_0 + G(u - u_0) \\ u &= u_0 + G^{-1}(v - \dot{\omega}_0) \end{aligned} \quad (1)$$

Where v is the virtual control vector, and subscript 0 indicates a time in the past. The control effectiveness depends on the inertia of the vehicle. However, directly measuring the inertia can be challenging. Alternatively, it

can be estimated from flight test data with actuator inputs with angular acceleration. Using the flight test data, the control effectiveness matrix(G) can be estimated by dividing angular acceleration by a control input vector(u). Practically, we conducted several manual outdoor flight tests to log radio control input commands and angular accelerations by post-processing the inertial measurement units reacting to the radio input. The control effectiveness was calculated by dividing the change of angular acceleration by the change of input commands from radio control, for each pitch, roll, and yaw axis at various airspeeds. After that, the effectiveness values were fitted as a quadratic function and scheduled by airspeed measurement, because the effectiveness of control surfaces depends on the dynamic pressure $q = \frac{1}{2}\rho V^2$.

Figure 2 shows the overview of the soaring controller. An INDI controller is used for both the inner loop and the outer loop. For the outer loop, linear acceleration error is calculated from position error and fed into the INDI outer loop.

The MAV's position(ξ), velocity($\dot{\xi}$), and a reference position are passed to a PD controller, and a linear acceleration reference($\ddot{\xi}_{ref}$) goes into the INDI outer loop [22]. Then, pitch, roll references, and thrust increment are calculated from the outer loop. K_ξ and $K_{\dot{\xi}}$ are the gains for position error and velocity error, respectively.

$$\ddot{\xi}_{ref} = K_{\dot{\xi}}(K_\xi(\xi_{ref} - \xi) - \dot{\xi}) \quad (2)$$

To keep the heading towards the wind, we calculate a yaw(ψ) reference. We set a virtual waypoint at 15 meters into the center of the cross-section of the wind tunnel. The yaw reference is calculated based on the error between the MAV's position and the virtual waypoint. Let x_{ref}, y_{ref} be the x and y position of the virtual waypoint, and x, y the longitudinal and lateral position of the MAV. Then the yaw reference is determined by the following equation:

$$\psi_{ref} = \text{atan}(\frac{y_{ref} - y}{x_{ref} - x}) \quad (3)$$

The attitude reference($\eta_{ref} = [\phi_{ref} \ \theta_{ref} \ \psi_{ref}]^T$) is passed to an inner loop PD controller. Then, the angular acceleration reference($\dot{\omega}_{ref}$) calculated from the PD controller and the thrust increment are passed into the inner loop INDI controller.

$$\dot{\omega}_{ref} = K_\omega(K_\eta(\eta_{ref} - \eta) - \omega) \quad (4)$$

The throttle is useful to navigate to the soaring region or to deal with a strong gust. It is also necessary to use throttle when the MAV cannot fly without power due to the wind conditions. However, changing gains or switching the controller from one to another in flight to enable or disable the throttle is not desirable. We utilized control allocation to cope with this problem. An INDI controller with control allocation allows the MAV to use the same controller and parameters throughout the flight, but seamlessly cut off or increase the throttle when desired. In particular, this makes it unnecessary to change anything when switching between navigating, searching for a soaring spot, and soaring flight.
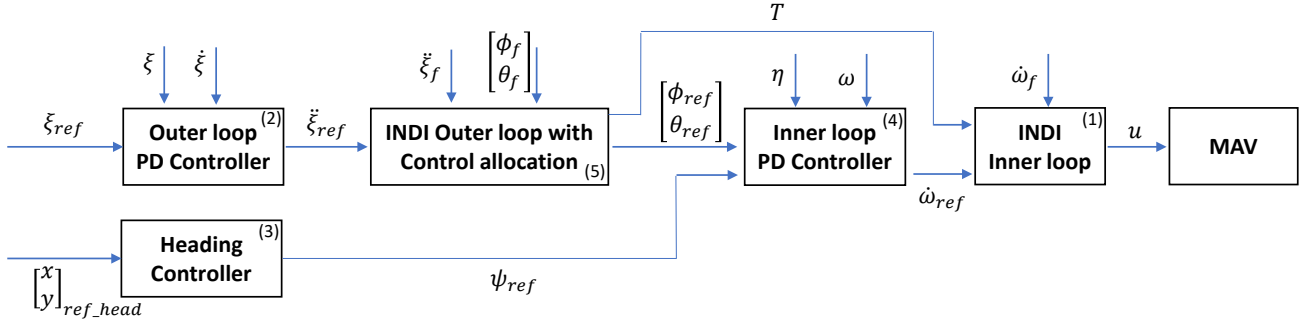
Fig. 2. A schematic overview of the soaring controller. ξ is the position, η is the attitude, ω is the angular rate, ψ is the yaw angle, T is the thrust, and u is actuator commands. Subscript ref means reference, and subscript f is for filtered signals.

Control allocation of INDI was originally developed to prevent actuator saturation for over-actuated vehicles, because of aggressive yaw control behaviour. The weighted least square (WLS) algorithm was integrated with the INDI controller for inner loop control allocation in a paper by Smeur et al. [23]. Here, we use control allocation for achieving as still soaring as possible. This implies a preference for reducing throttle and minimizing accelerations. This leads to the following control allocation cost function:

$$C_{alloc}(u_{wls}) = \|W_u(u_{wls} - u_{p,wls})\|^2 + \gamma\|W_v(G_o u_{wls} - v_{wls})\|^2 \quad (5)$$

Where u_{wls} is a control increment vector, W_u is a weighting matrix for the control inputs, W_v is a weighting matrix for the control objective, G_o is the control effectiveness matrix for the outer loop, v_{wls} is the virtual control increment command for the outer loop, u_p is the preferred control increment vector, and γ is a scale factor.

In this study, control allocation is used to prioritize the controls to minimize thrust for the outer loop controller. We set a higher priority to control pitch than the thrust by adjusting weight for the WLS optimization. Therefore, for W_v, we choose the weights to be 1, 100, 1 for roll(rad), pitch(rad), and thrust in a range of [0, 9600] respectively. For W_u, we choose 1 for all axes, γ is 10^6, and u_p is a zero vector.

B. AOSearch: Autonomous Orographic Search for a soaring location

To exploit updrafts, MAVs have to find feasible soaring locations autonomously. If the environment is static and a prior knowledge of the wind field is provided, it can be calculated from the MAV model and wind speed by finding an equilibrium. However, in the real world, it is difficult to measure the entire wind field. Furthermore, the wind speed may change during the flight. Thus, we developed an algorithm to find the soaring location which is applicable in a non-static environment, without any prior knowledge of the wind field.

Simulated annealing is a local meta-heuristic search technique [24], [25]. It attempts to minimize a cost function by taking steps to neighboring positions in the search space. It either accepts or rejects the solution based on an acceptance probability which usually decreases to zero as the "temperature" decreases. Initially, steps that increase the cost are allowed, but eventually, simulated annealing becomes a greedy algorithm.

Based on simulated annealing, we implemented *AOSearch*: Autonomous Orographic Search algorithm. In our case, there could be environmental changes at any time during the flight, regardless of the progress of the search. For example, wind speed can be changed over time. Therefore, the temperature does not decrease at each step. The temperature is set to zero, so the search algorithm only accepts better solutions. When the MAV finds a position at which it can soar, the search is finished. Hence, we implemented a threshold cost. If the value of the cost function is lower than the threshold at a certain position, it is considered converged and does not move to a new neighbour. When the value of the cost function exceeds the threshold because of an environmental change, the MAV restarts the search until it finds a new position that satisfies the threshold condition.

We want to minimize energy expenditure as well as stay in the same position as much as possible just like kestrels hovering without moving its head for observation. In our case, thrust is the primary source of energy consumption to minimize. Pitch rate is also contributing because controlling the elevator spends energy. Furthermore, we aim for wind-hovering, which means that the MAV maintains its position. So, we want to minimize both horizontal and vertical speed. The cost function(C_{search}) captures all requirements for still, orographic soaring, with the MAV able to keep its downward view as static as possible with minimum throttle usage, also meaning minimal position and pitch changes. Hence, it is a function of thrust($T[\%]$), horizontal and vertical ground speed($\dot{x}$, $\dot{z}[m/s]$), and pitch rate($\dot{\theta}[rad/s]$) with a gain for each parameters.

$$C_{search} = k_1 T + k_2|\dot{x}| + k_3|\dot{z}| + k_4|\dot{\theta}| \quad (6)$$

The gains were k_1=9.6, k_2=1.6, k_3=1.0, and k_4=10. The threshold value was set to 43. The gains and threshold value were determined empirically, by observing the values when

the MAV was soaring in a stable manner.

AOSearch: the autonomous orographic search method is described in algorithm 1. First, calculate the cost function. Based on the value of the cost function, a step size(S) is selected. If the value is lower than the threshold, the MAV stays at the current position($Pos(s_{new})$). If the value of the cost function has increased compared to the previous position($Pos(s)$), go back to the previous position. If the value has decreased, keep the same direction. If it has just come back from a previous position, pick a new random neighbour. To get a random neighbour s_{new}, pick a random direction($Dir(s_{new})$) among four direction vectors of [x, z]: forward[1, 0], backward[-1, 0], up[0, 1], down[0, -1]. Then, a new soaring position is: $Pos(s_{new}) = Pos(s) + Dir(s_{new}) \times S$. Repeat the process until the cost function value becomes less than the threshold.

The step size (S) is set to four steps depending on the value of the cost function. As the value gets lower, the step size decreases from 0.3 *m* to 0.05 *m*. The logic behind this is that the MAV takes a bigger step to explore the wind field when the energy consumption at the current position is high. When the value of the cost function is low at the current position, the MAV tries to fine-tune its position.

$$S = \begin{cases} 0.3, & \text{for } C \geq 3 \times threshold \\ 0.2, & \text{for } 3 \times threshold > C \geq 2 \times threshold \\ 0.1, & \text{for } 2 \times threshold > C \geq 1.5 \times threshold \\ 0.05, & \text{otherwise} \end{cases}$$

Algorithm 1 AOSearch

1: $C(s_{new}) \leftarrow k_1 T + k_2|\dot{x}| + k_3|\dot{z}| + k_4|\dot{\theta}|$ ▷ Calculate cost function
2: S ← Calculate a step size
3: **if** $C(s_{new}) < threshold$ **then**
4:      Stay at the current position
5: **else**
6:      **if** $C(s_{new}) < C(s)$ **then**
7:          **if** returned *is* True **then**
8:              $Dir(s_{new}) \leftarrow random$ ▷ pick a random direction
9:              $returned \leftarrow False$
10:          **else**
11:              $Dir(s_{new}) \leftarrow Dir(s)$ ▷ keep the same direction
12:          **end if**
13:      **else**
14:          $Dir(s_{new}) \leftarrow -Dir(s)$ ▷ go back to the previous position
15:          $returned \leftarrow True$
16:      **end if**
17:      $Pos(s_{new}) = Pos(s) + Dir(s_{new}) \times S$
18:      Move to a new soaring position $Pos(s_{new})$
19: **end if**

III. HARDWARE AND TEST SETUP

A. Eclipson model C 3d-printed model plane

An Eclipson model C [26] airplane was used for the flight tests. It is a 3d-printed plane, which makes it easy to replace parts in case of a crash. It was printed with lightweight polylactic acid (LW-PLA) to reduce weight and increase aerodynamic performance. Unlike other studies that used a glider or a flying wing with flaps, we chose a 5-channel model plane to have more control. It has four servos
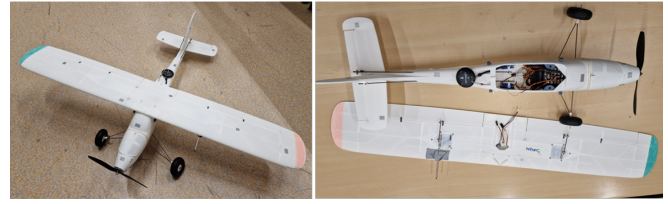

Fig. 3. Eclipson model C 3D-printed airplane. It has 1.1*m* wingspan and a weight of 716*g* including a 1.5*A* Li-Po battery. Pixhawk4 is equipped with Paparazzi open-source autopilot software. An airspeed sensor is mounted under the wing. GPS sensor and opti-track markers are used for localization.
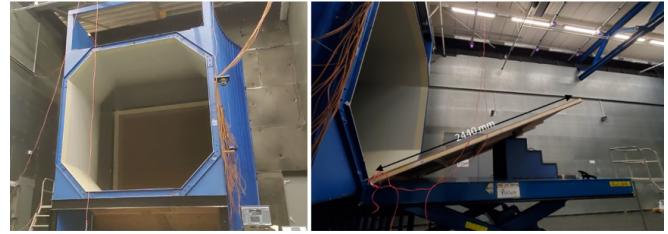

Fig. 4. TU Delft OJF and the slope setting for generating orographic updraft. The cross-section of the wind tunnel is 2.85*m* × 2.85*m*. A wooden plate (2.44*m* × 2.44*m*) was used for the slope. The slope was placed in front of the outlet of the wind tunnel.

for control surfaces (elevator, left aileron, right aileron, and rudder) and an electric motor at the front. Especially the rudder is useful to keep its heading toward the wind while maintaining its lateral position to stay in a small updraft region. The throttle is also necessary to navigate and fly safely in case of a strong gust or sudden environmental changes.

The MAV is shown in Figure 3. It has a wingspan of 1100*mm*, 18*dm²* wing surface area, and an aspect ratio of 6.9. A Pixhawk 4 with Paparazzi autopilot open-source software [27] was used. Pixhawk 4 board has a processor and an inertial measurement unit. A GPS module and opti-track markers were used for outdoor and indoor localization, respectively. An airspeed sensor was mounted under the wing and calibrated in the wind tunnel. The weight of the aircraft with electronics was 595 grams excluding battery, and a total of 716 grams including a 1.5*A* Lithium-Polymer battery.

B. Wind tunnel test setup

The TU Delft open jet facility (OJF) is a wind tunnel with a 2.85*m* × 2.85*m* cross-section. We installed a ramp in front of the wind outlet to generate an orographic updraft. The OJF and the slope are shown in Fig. 4. The slope angle and wind speed were adjustable during the flight. For safety reasons, a rope system was attached to the ceiling. An Opti-track system installed in the wind tunnel was calibrated before the flight test. In order to get insight into the wind field with this setup, a computational fluid dynamics (CFD) simulation was performed using ANSYS fluent [28]. The contours of horizontal and vertical wind speeds are shown in Fig.5. The strongest updraft occurs close to the end of the slope, and the wind speed decreases near the slope because of the boundary friction.
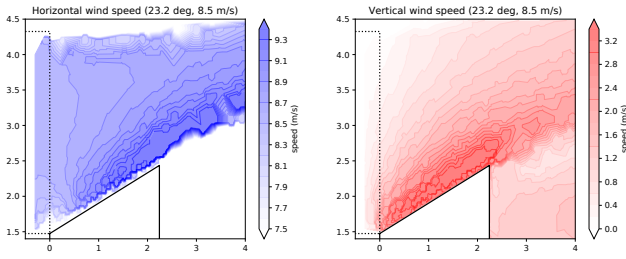
Fig. 5. Contours of horizontal and vertical wind speed over a slope in the wind tunnel. The slope angle was set to 23.2 degrees and the wind speed from the OJF was set to 8.5 m/s.
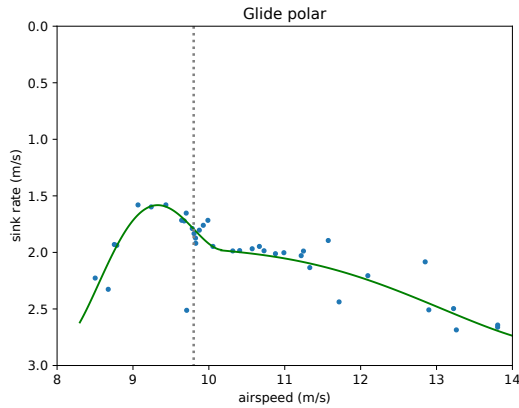

Fig. 6. Glide polar of the Eclipson model C airplane. The data was divided into two sections and fitted with a fourth-order polynomial. Additional drag was generated at around 9.8 m/s of the airspeed because of the propeller windmilling.

C. Glide polar

To measure the sink rate, a series of outdoor flight tests was performed. The MAV had the same hardware and weight setting with the indoor tests for consistency. We flew the MAV manually on a calm day, and retrieved sink rate and airspeed data from flight logs during parts of the flight where the motor was turned off. The data points and the glide polar are shown in Fig. 6. The data was divided into two sections and fitted with a fourth-order polynomial function. The sink rate sharply increases around 9.8 m/s of airspeed. This is because the propeller started windmilling at that airspeed, generating additional drag.

IV. Test results

A. Autonomous Soaring

Indoor flight tests were performed to demonstrate autonomous soaring using the proposed methods in this paper. We first started the flight in manual mode for safety reasons until the nominal wind speed reached 8.0 m/s. Once the wind speed was stabilized, we switched on the autonomous flight mode. The MAV was fully autonomous from that point. It first hovered at a standby position and stabilized itself. Then it started the local search and autonomously moved its target soaring position until it converged according to

algorithm 1. The standby position was set where requires approximately 20% of the throttle for hover. During the flight, we changed the environmental conditions, either the wind speed or the slope angle. Everything was running onboard, except position measurements from the opti-track. Data from onboard sensors was written on an SD card using a high speed flight logger and retrieved after the flight.

There were two experimental cases: one in which we changed the wind speed with a fixed slope angle, and another in which we changed the slope angle with a fixed wind speed. In both cases, the MAV was able to successfully soar using very little energy without any prior knowledge of the wind field. Furthermore, it was shown that autonomous soaring in a changing environment is possible by using the proposed methods.

1) Changing nominal wind speed: The first case is to change the wind speed during soaring. We started autonomous soaring at a wind speed of 8.5 m/s, and then slowly increased it to 9.8 m/s, and decreased it again. When the wind speed changes, the MAV either stays at the same position if possible, or tries to find another position. If the current position becomes not feasible for soaring due to the change of the wind field, it restarts searching for a new feasible soaring position.

Figure 7 shows the wind speed, horizontal and vertical position, and throttle usage during the flight. Figure 8 shows the trajectory during the autonomous search from an initial position at 8.5 m/s wind speed. The throttle started of at 20%, but after the autonomous search for a soaring location turned on at 352 seconds, the MAV found a good soaring location in 185 seconds, achieving zero throttle at 537 seconds. From that time on, the throttle usage was very low. At 2244 seconds, we retook manual control to make the MAV land. Manual control is shown by the shaded regions. The wind speed was changed over time, which led to changes in position. There is no clear proportional relationship between wind speed and the positions. We will analyze the chosen soaring positions further in the next section. The flight time was a total of 30 minutes excluding a short manual flight after launch and landing. During the soaring, the mean throttle usage was 0.25 %. Note that it is a significant decrease in throttle usage compared to non-soaring flights. 38% of the throttle was required for the MAV to hover in the wind tunnel without a ramp at 8.5 m/s wind speed, and 30 to 50% of the throttle was normally used during outdoor flights.

2) Varying slope angle: In the second case we changed the slope angle during soaring. The procedure remained the same as the first case, but instead of changing the wind speed, we changed the slope inclination while the nominal wind speed was fixed at 8.5 m/s. The slope angle, the MAV's horizontal and vertical position, and the throttle usage are shown in figure 9. The slope angle was set from 22.1 degrees to 25.2 degrees. The step size of the slope angle was not consistent because of the practical difficulty of moving the ramp precisely during the flight. The throttle usage was 25% at the start. We started the autonomous search at 1440 seconds. The MAV achieved zero throttle
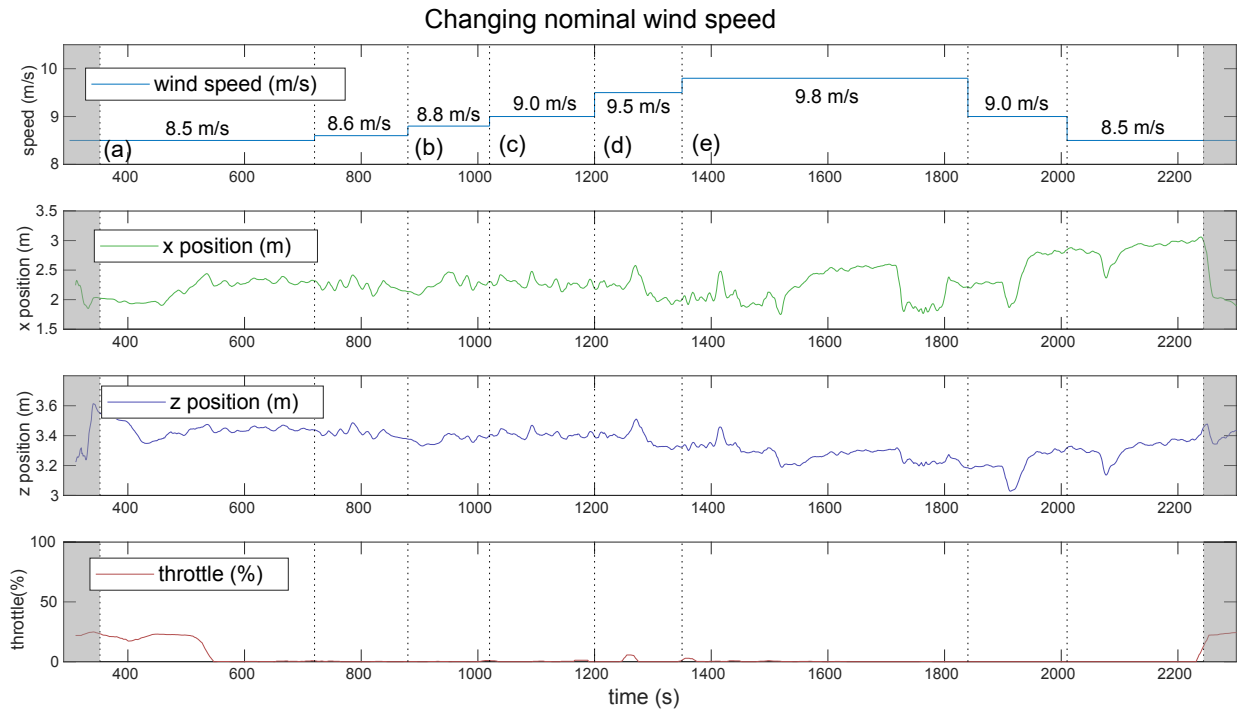
Fig. 7.   Horizontal and vertical position, and throttle usage of the MAV during the flight according to the change of the wind speed from the wind tunnel.
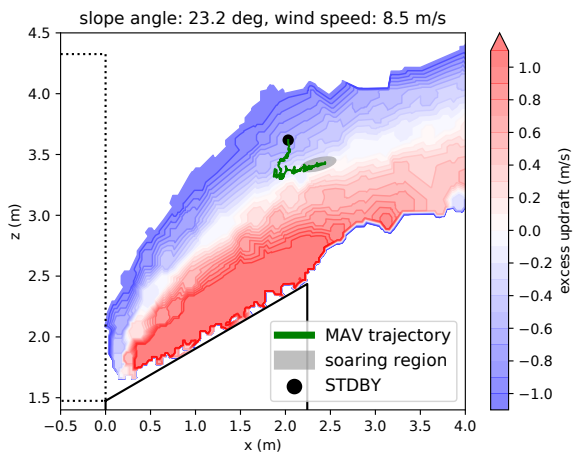

Fig. 8.   Flight trajectory during the autonomous search, from 352 to 537 seconds in case 1. The MAV started searching at the standby position (STDBY), marked as a black dot. It found a good soaring location and achieved zero throttle in 185 seconds.

flight at 1572 seconds, 132 seconds after starting the search. At 2950 seconds, we retook manual control for landing. With a slope angle of 22.1 degrees, the inclination was small, so less updraft was generated than other conditions. Because of that, the combination of the horizontal and vertical wind was not favorable to hover stationary. Nevertheless, it was still able to soar using zero throttle at 22.1 degrees slope angle, allowing some movement. The soaring position moved frontward when the slope inclination increased, and it moved backward again when the slope angle decreased. This is because the feasible soaring region pushed forward as the

slope angle increased. We will analyze the change of the wind field and chosen positions in the next section. The soaring flight time was a total of 25 minutes, excluding short manual flights after launch and before landing. During the soaring, the mean throttle usage was 0.25 %. Note that in both test cases, the flight was stopped before using up all the battery because of the size of the flight log file and limited onboard memory.

V. ANALYSIS AND DISCUSSION

For further analysis, we ran a CFD simulation for each combination of wind speed and slope angle. Based on the wind field calculated by the CFD simulation and the MAV's sink rate, a feasible soaring region can be determined. The MAV can soar where the updraft and sink rate are balanced (i.e. excess updraft = updraft - sink rate = 0), which is a white region in figures 10 and 11. The trajectory shows that the MAV mostly stayed in the region where the updraft and sink rate is balanced. The plots show that the MAV generally resides inside or very close to the white areas, in which the predicted excess updraft is zero. The MAV does occasionally fly in slight non-zero excess updraft locations, which we expect to be due to imperfections in the predictions. These imperfections can have various sources: differences between the CFD simulation and the real world, imperfect sink rate measurement and fitting, airspeed sensor measurement, and the wind speed controller of the OJF also had steady-state errors within 0.1 m/s range.

One thing we noticed is that the vertical position of the MAV went down as the wind speed increased. Although this may seem counter-intuitive, the glide polar explains the
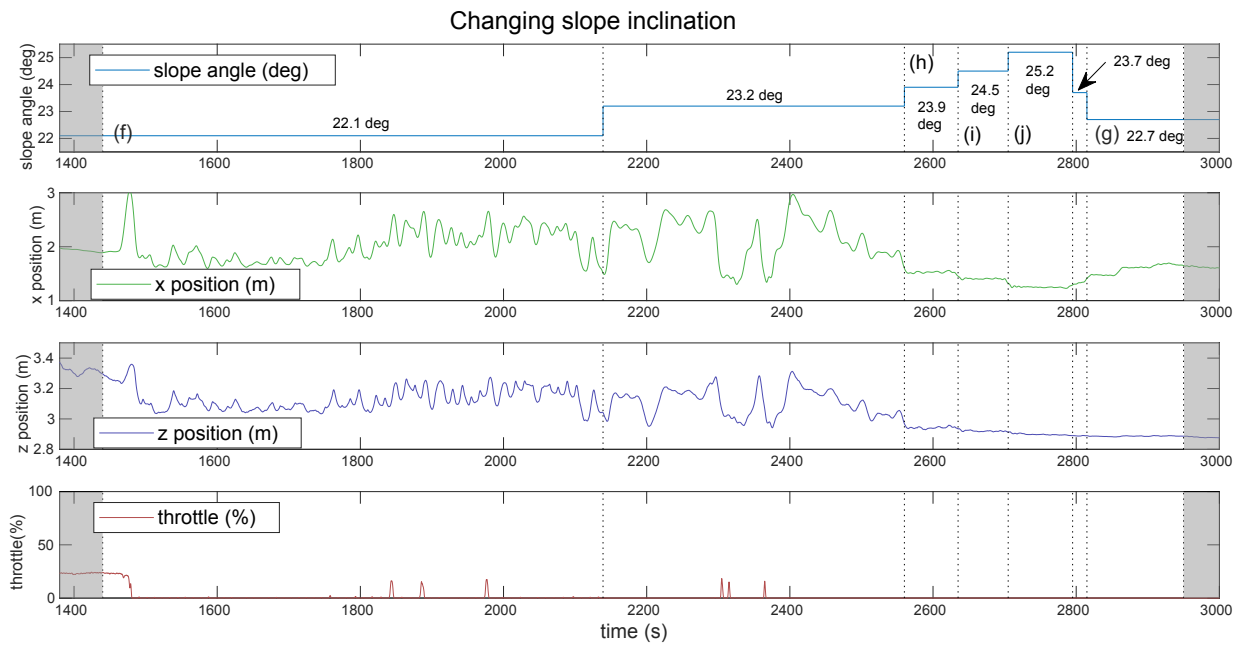
Fig. 9. Horizontal and vertical position, and throttle usage of the MAV during the flight according to the change of the slope angle.
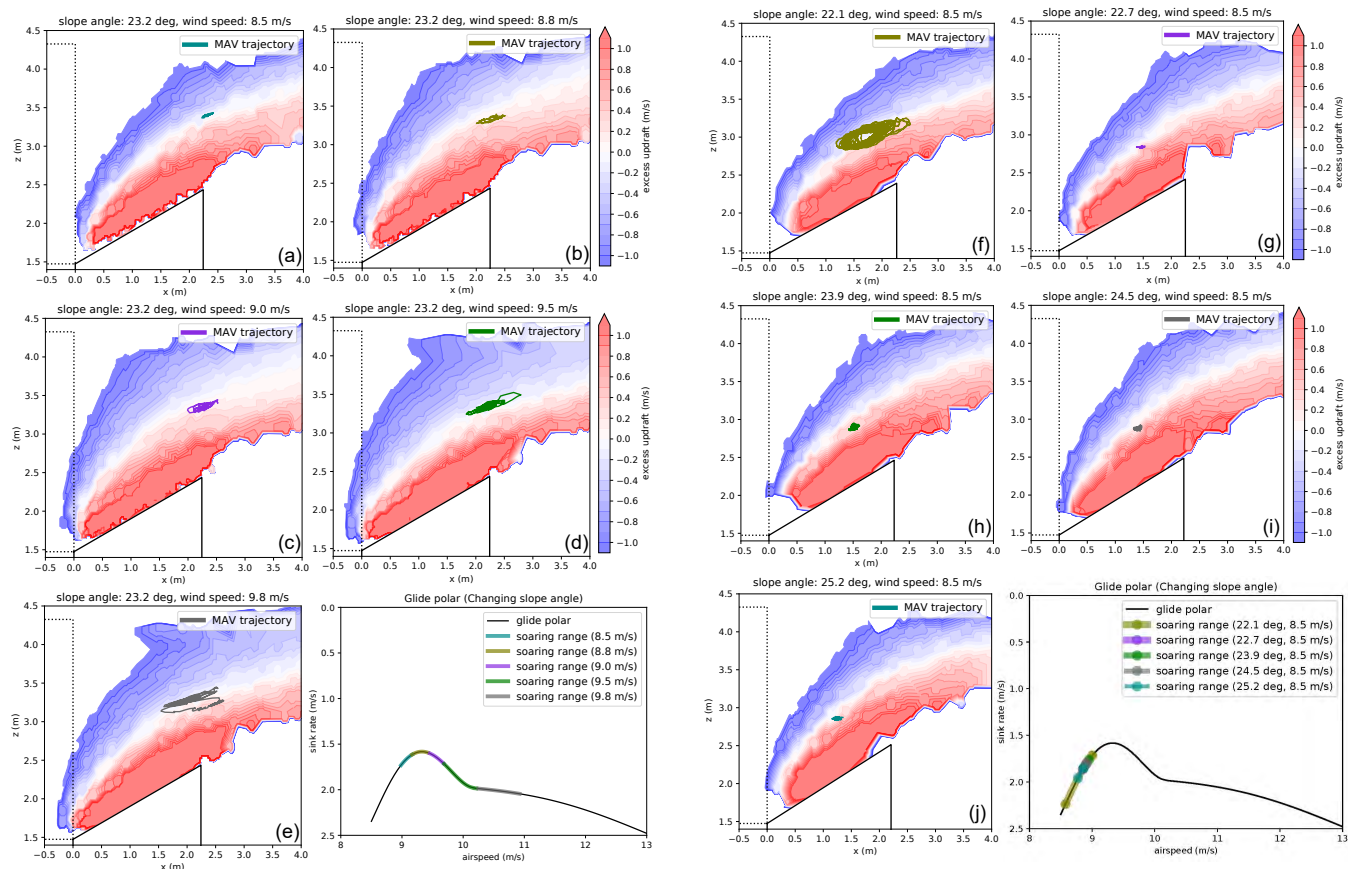

Fig. 10. Case 1: autonomous soaring in changing wind speed from 8.5 to 9.8 m/s. Contours of excess updraft and the MAV trajectory are shown for each nominal wind speed. In the last plot, the range of sink rate for each case is shown on the glide polar.
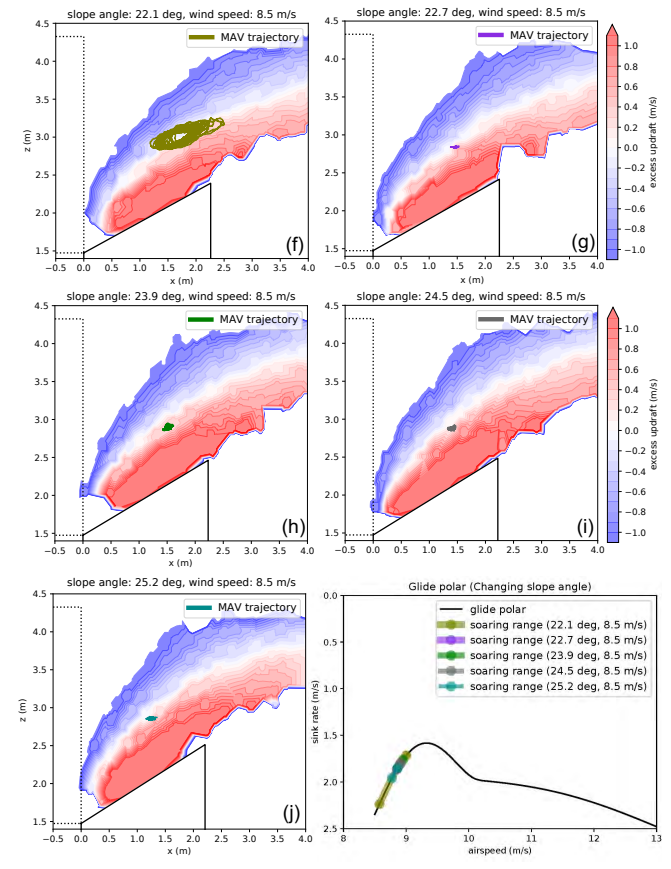
Fig. 11. Case 2: autonomous soaring at varying slope angles from 22.1 to 25.2 degrees. The contours of excess updraft and the MAV trajectory for each slope angle are shown. On the last plot, glide polar with the range of the sink rate for each case is presented. The sink rates stayed in similar ranges because the nominal wind speed was fixed at 8.5 m/s in this case.

behavior. The minimum sink rate of the MAV occurs at around 9.1 m/s of airspeed. When the airspeed gets higher than that, the sink rate significantly increased. As a result, more updraft is required for the MAV to stay aloft, and the MAV lowers its altitude to find a stronger updraft. In case of a varying slope angle, the position of the MAV moved forward as the inclination increased. It is because more updraft is generated with a larger slope angle, thus the position that balances the sink rate and the updraft moved forward.

In both cases, the throttle usage and energy consumption decreased significantly after switching on the soaring mode. We set a position where the MAV can hover using approximately 20% of the throttle as a standby. During the soaring flight, the throttle usage dropped and remained at close to 0% for almost all the time. The mean throttle usage of the entire soaring flight was 0.25% for both cases, compared to 38% for a nominal flight. There were a few moments that the MAV used throttle because it was necessary to avoid a stall or recover the position. Note that the size of the feasible soaring region is only 10 to 20 cm vertically, which is very small to maintain the position within the region even for a human pilot. Also, we changed the wind field during the flight so sometimes the MAV had to overcome a sudden change using the throttle.

VI. Conclusion

In this paper, we demonstrated the first autonomous orographic soaring in the real-world environment without a priori knowledge of the wind field, pre-defined trajectory planning, or manual initialization by a human pilot. A fully autonomous orographic soaring was performed in the wind tunnel with changing environment settings. With a combination of the local search algorithm and the INDI controller with control allocation, the MAV was able to soar autonomously with almost zero throttle in the updraft for over 25 minutes of flight time. Furthermore, we verified that the MAV can find a new soaring position when the updraft changes by using the proposed search method.

For future work, performing an outdoor flight test will be the next step. The proposed methods in this paper are applicable to outdoor flights because they do not depend on any pre-measured environmental condition. However, the MAV will have to be more aware of its environment, as the wind may change direction and it will have to sense and avoid obstacles during the search. Using additional sensors can be helpful for recognizing the surroundings.

References

[1] M. Elbanhawi, A. Mohamed, R. Clothier, J. L. Palmer, M. Simic, and S. Watkins, "Enabling technologies for autonomous mav operations," *Progress in Aerospace Sciences*, vol. 91, pp. 27–52, 2017.

[2] L. W. Traub, "Range and endurance estimates for battery-powered aircraft," *Journal of Aircraft*, vol. 48, no. 2, pp. 703–707, 2011.

[3] K. L. Bildstein and M. W. Collopy, "Hunting behavior of eurasian (falco tinnunculus) and american kestrels (f. sparverius): a review," *The ancestral kestrel. Raptor Res. Rep*, vol. 6, pp. 66–88, 1987.

[4] R. Strandberg, T. Alerstam, and M. Hake, "Wind-dependent foraging flight in the osprey pandion haliaetus," *Ornis Svecica*, vol. 16, no. 3, pp. 150–163, 2006.

[5] R. Harel, N. Horvitz, and R. Nathan, "Adult vultures outperform juveniles in challenging thermal soaring conditions," *Scientific reports*, vol. 6, no. 1, pp. 1–8, 2016.

[6] C. J. Pennycuick, "Gliding flight of the white-backed vulture gyps africanus," *Journal of Experimental Biology*, vol. 55, no. 1, pp. 13–38, 1971.

[7] M. J. Allen and V. Lin, "Guidance and control of an autonomous soaring uav," Tech. Rep., 2007.

[8] D. Edwards, "Implementation details and flight test results of an autonomous soaring controller," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2008, p. 7244.

[9] C. Dunn, J. Valasek, and K. C. Kirkpatrick, "Unmanned air system search and localization guidance using reinforcement learning." in *Infotech@ Aerospace*, 2012, pp. 1–8.

[10] G. Reddy, J. Wong-Ng, A. Celani, T. J. Sejnowski, and M. Vergassola, "Glider soaring via reinforcement learning in the field," *Nature*, vol. 562, no. 7726, pp. 236–239, 2018.

[11] J. Videler, D. Weihs, and S. Daan, "Intermittent gliding in the hunting flight of the kestrel, falco tinnunculus," *J. exp. Biol*, vol. 102, pp. 1–12, 1983.

[12] E. L. Shepard, C. Williamson, and S. P. Windsor, "Fine-scale flight strategies of gulls in urban airflows indicate risk and reward in city living," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 371, no. 1704, p. 20150394, 2016.

[13] C. J. Williamson, A. Spelt, and S. P. Windsor, "Bird-inspired velocity optimization for uavs in the urban environment," in *AIAA Scitech 2020 Forum*, 2020, p. 1948.

[14] M. Penn, G. Yi, S. Watkins, M. Martinez Groves-Raines, S. P. Windsor, and A. Mohamed, "A method for continuous study of soaring and windhovering birds," *Scientific Reports*, vol. 12, no. 1, pp. 1–9, 2022.

[15] C. White, E. W. Lim, S. Watkins, A. Mohamed, and M. Thompson, "A feasibility study of micro air vehicles soaring tall buildings," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 103, pp. 41–49, 2012.

[16] C. White, S. Watkins, E. W. Lim, and K. Massey, "The soaring potential of a micro air vehicle in an urban environment," *International Journal of Micro Air Vehicles*, vol. 4, no. 1, pp. 1–13, 2012.

[17] A. Guerra-Langan, S. Araujo-Estrada, and S. Windsor, "Unmanned aerial vehicle control costs mirror bird behaviour when soaring close to buildings," *International Journal of Micro Air Vehicles*, vol. 12, p. 1756829320941005, 2020.

[18] A. Fisher, M. Marino, R. Clothier, S. Watkins, L. Peters, and J. L. Palmer, "Emulating avian orographic soaring with a small autonomous glider," *Bioinspiration & biomimetics*, vol. 11, no. 1, p. 016002, 2015.

[19] C. P. L. de Jong, B. D. W. Remes, S. Hwang, and C. De Wagter, "Never landing drone: Autonomous soaring of a unmanned aerial vehicle in front of a moving obstacle," *International Journal of Micro Air Vehicles*, vol. 13, p. 17568293211060500, 2021.

[20] S. Sieberling, Q. Chu, and J. Mulder, "Robust flight control using incremental nonlinear dynamic inversion and angular acceleration prediction," *Journal of guidance, control, and dynamics*, vol. 33, no. 6, pp. 1732–1742, 2010.

[21] E. J. J. Smeur, Q. Chu, and G. C. H. E. De Croon, "Adaptive incremental nonlinear dynamic inversion for attitude control of micro air vehicles," *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 3, pp. 450–461, 2016.

[22] E. J. J. Smeur, M. Bronz, and G. C. H. E. de Croon, "Incremental control and guidance of hybrid aircraft applied to a tailsitter unmanned air vehicle," *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 2, pp. 274–287, 2020.

[23] E. Smeur, D. Höppener, and C. De Wagter, "Prioritized control allocation for quadrotors subject to saturation," in *International Micro Air Vehicle Conference and Flight Competition*, no. September, 2017, pp. 37–43.

[24] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi, "Optimization by simulated annealing," *science*, vol. 220, no. 4598, pp. 671–680, 1983.

[25] L. Ingber, "Adaptive simulated annealing (asa): Lessons learned," *arXiv preprint cs/0001018*, 2000.

[26] Eclipson. Model c. [Online]. Available: https://www.eclipson-airplanes.com/modelc

[27] G. Hattenberger, M. Bronz, and M. Gorraz, "Using the paparazzi uav system for scientific research," 2014.

[28] ANSYS, "Ansys academic research fluent, release 19.5.0," 2019.