Assessment of Closed Loop Dynamics in the Multidisciplinary Design and Optimization of Small UAVs

Luiz Fernandez, Murat Bronz, Thierry Lefebvre, Nathalie Bartoli

▶ **To cite this version:**

Luiz Fernandez, Murat Bronz, Thierry Lefebvre, Nathalie Bartoli. Assessment of Closed Loop Dynamics in the Multidisciplinary Design and Optimization of Small UAVs. AIAA AVIATION 2023 Forum, Jun 2023, San Diego, United States. 10.2514/6.2023-3902 . hal-04171574

HAL Id: hal-04171574
https://hal.science/hal-04171574v1

Submitted on 26 Jul 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Assessment of Closed Loop Dynamics in the Multidisciplinary Design and Optimization of Small UAVs

Luiz F. T. Fernandez*
ONERA/DTIS, Université de Toulouse, Toulouse, France
ENAC, Université de Toulouse, F-31400 Toulouse, France

Murat Bronz†
ENAC, Université de Toulouse, F-31400 Toulouse, France

Thierry Lefebvre‡ and Nathalie Bartoli §
ONERA/DTIS, Université de Toulouse, Toulouse, France

There has been an increasing interest in the application of agile unmanned aerial vehicles (UAVs) for various missions. In this paper, we present an introductory assessment of the impact of addressing control feasibility in the Multidisciplinary Design and Optimization (MDO) of such vehicles. Our approach involves incorporating dynamical simulation disciplines into an existing MDO process, originally formulated for tail-sitter vehicles, to achieve more dexterous flights. Specifically, we simulate the cruise phase of an UAV flight by coupling a flight dynamics module with a control system that includes attitude and velocity control as well as a guidance law for 3D navigation. An open loop simulation is also employed to account for the vehicle's ability to hover. We begin our studies with a single design variable optimization problem and demonstrate how the simulation can be used to reduce energy consumption. Final results, obtained optimizing the vehicle and guidance law gains, highlight the effectiveness of this approach in improving the control maneuverability and energy efficiency of an agile UAV.

I. Nomenclature

e_0, e_1, e_2, e_3	=	Unit quaternion for attitude representation
u, v, w	=	Linear velocities in the body frame
p, q, r	=	Angular velocities in the body frame
ρ	=	Air density
α	=	Angle of attack
β	=	Angle of sideslip
S	=	Wing area
b	=	Wing span
c	=	Wing mean aerodynamic chord
W	=	Vehicle weight
V_a	=	Airspeed at the aerodynamic frame
V_p	=	Airspeed at the propeller frame
δ_p	=	Motor angle of inclination
α_e	=	Motor angle of attack
C_{L_0}	=	Wing lift coefficient for zero α
C_{L_α}	=	Wing lift curve slope
C_{L_q}	=	Variation of wing lift coefficient with dimensionless pitch rate
C_{D_0}	=	Wing drag coefficient for zero α
C_{D_α}	=	Wing drag curve slope

*PhD Candidate, DTIS, luiz.tiberio@onera.fr, AIAA Student Member
†Assistant Professor, murat.bronz@enac.fr, AIAA Member
‡Research Engineer, DTIS, thierry.lefebvre@onera.fr, AIAA Member
§Senior researcher, DTIS, nathalie.bartoli@onera.fr, AIAA MDO TC Member.

C_{D_q}	=	Variation of wing drag coefficient with dimensionless pitch rate
C_{m_0}	=	Wing pitching moment coefficient for zero α
$C_{m\alpha}$	=	Wing pitching moment curve slope
C_{m_q}	=	Variation of wing pitching moment coefficient with dimensionless pitch rate
C_{Y_β}	=	Variation of wing side force coefficient with sideslip
C_{Y_p}	=	Variation of wing side force coefficient with dimensionless roll rate
C_{Y_r}	=	Variation of wing side force coefficient with dimensionless yaw rate
C_{l_0}	=	Wing rolling moment coefficient for zero β
$C_{l\beta}$	=	Variation of wing rolling moment coefficient with dimensionless β
C_{l_p}	=	Variation of wing rolling moment coefficient with dimensionless roll rate
C_{l_r}	=	Variation of wing rolling moment coefficient with dimensionless yaw rate
C_{n_0}	=	Wing yawing moment coefficient for zero β
$C_{n\beta}$	=	Variation of wing yawing moment coefficient with dimensionless β
C_{n_p}	=	Variation of wing yawing moment coefficient with dimensionless roll rate
C_{n_r}	=	Variation of wing yawing moment coefficient with dimensionless yaw rate
v_w, v_{w_s}, v_{w_g}	=	Total wind speed, steady component, and gust component
u_f and u_c	=	New and current INDI commands
G	=	INDI attitude control effectiveness matrix
Ω, Ω_f, and Ω_{ref}	=	Angular rate, measured angular rate, and reference angular rate of the vehicle
$\dot{\Omega}_f$	=	Measured angular acceleration
T_f and T_d	=	Current and desired thrust
q_{err}	=	Quaternion error
v and v_f	=	Velocity command generated by the velocity control and current velocity
$G_T(\eta, T)$ and $G_L(\eta, V))$	=	Velocity control effectiveness for thrust and lift
a_{ref} and a_f	=	Reference and current accelerations
E_{motor}	=	energy required for each motor for the entire simulation
V_{motor}	=	Voltage of the motor at each simulation step
I_{motor}	=	Current of the motor at each simulation step: assumed as constant during cruise
dt	=	Interval for time integration defined as the duration of the system simulation
NED	=	North, east, down axes
UAV	=	Unmanned aerial vehicle
MDO	=	Multidisciplinary design and optimization
MDA	=	Multidisciplinary Analysis
INDI	=	Incremental Nonlinear Dynamic Inversion
ESP	=	Engineering Sketch Pad
UR, LR, UL, LL	=	Upper right, lower right, upper left, and lower left motors respectively

II. Introduction

In recent years, unmanned aerial vehicles (UAVs) have been used or considered for a variety of missions, such as fire fighting [1–3], debris mapping [4], and even covid mitigation [5]. The recent progresses with vehicle design, sensing, and control have pushed the limits even further, allowing, for instance, the use of UAVs in search and rescue situations [6, 7]. But there are still challenges to be tackled in order to fulfill the potential of such technology. In order to have operational vehicles in safety critical missions, where time plays an important role, often conflicting features are needed: high flight speed and endurance until the location is reached, and dexterity to navigate through cluttered and unknown environments and interact with humans. Even if progress has been made in the context of vehicle design, specially for shape shifting multi-copters [8, 9] and hybrid vehicles [10–13], trajectory generation and control have been the focus of many studies regarding the agile flight of UAVs [14–17] just to name a few. In this paper we are interested in using the advances in flight control to enrich design techniques. We present an introductory investigation if flight dynamics and control laws should be evaluated within the design process, and present a way to do so, employing an Multidisciplinary Design and Optimization (MDO) approach. Regarding the insertion of control laws in the MDO, one of the first proposed methodologies, by Perez *et al.* [18], tackled the concurrent design of the vehicle and a stability augmentation system (SAS). The authors presented a distributed MDO employing a collaborative optimization architecture. Their formulation tackled five disciplines, namely weights, aerodynamics,

propulsion, performance, and control, where for the latter both longitudinal (short-period) and lateral-directional (dutch roll) dynamics were addressed and the control gains were defined as design variables in the disciplinary level problem. Even if direct improvements were observed within the context of the paper, the real value of this simultaneous control law design can appear even later on the development of the aircraft, where problems related with undesired flight behaviour can be discovered. Faïsse *et al.* [19] presented a co-design strategy implemented as a nested architecture. They applied their methodology to a 2D aeroelastic airfoil problem, where both structure and control law (H_∞) were designed for active flutter suppression. With a similar structure, Van Nguyen [20] simultaneously optimized the control law and the vertical stabilizer of a vehicle with distributed propulsion systems. Stanford [21, 22] performed wing structural optimizations and found optimal control surfaces while considering both maneuver load alleviation and active flutter suppression, using a linear quadratic regulator (LQR) control structure within the optimization loop. In a different vein of the others, Gupta *et al.* [23] proposed the use of the "Controllability Gramian" within the design loop as a surrogate for the required energy to control a mode, instead of directly using the control laws to run simulations. Such strategy addresses the passive dynamics of the vehicle, and bypasses the need of defining the control law in the conceptual design. Open loop flight dynamics has also been addressed in MDO by means of simultaneous design and trajectory optimization. Chauhan and Martins [24] optimized the takeoff trajectory and the design of an eVTOL vehicle, Jasa *et al.* [25] applied this concept for a supersonic aircraft, and Hendricks *et al.* [26] optimized the whole trajectory of a tilt-wing urban air mobility aircraft. Within this approach, an optimal trajectory (from a single phase up to the entire flight) is found. The underlying assumption is that the control system, not taken into account, will be able to track such trajectory, which may not be the case, specially when non ideal conditions, such as wind gusts, take place in real flights. Such trajectory optimization strategy has proven to be important to obtain the optimal design (e.g. maximal range or minimal fuelburn) with respect to the nominal desired trajectory, but it could be further improved by considering the control system itself and its robustness to perturbations throughout the flight envelope. Within this line, Ryou *et al.* [15] used three different levels of fidelity (two different numerical models and real flights) to create a surrogate model for the dynamic feasibility check within trajectory optimization of a quadcopter.

Our objective with this paper is twofold. We are interested in evaluating how the closed loop flight dynamics can be addressed as a discipline of a given MDO process, and how such insertion impacts the final design. As baseline, we use the vehicle named Falcon [27], designed for the 13th edition of the International Micro Air Vehicle Conference and Competition. We then perform closed loop dynamic analyses and optimizations on this baseline. Throughout this paper, by control laws or control system we refer to the complete system needed to take the vehicle from one point to another in a controlled motion: attitude control, velocity control, and guidance law. Sensing is kept out of our scope and assumed to be ideal. The design of the control laws is not taken into account for two main reasons. From one side because the main focus can be kept in the understanding of the closed loop dynamical behaviour effects on the MDO. But must importantly because we add a real control system, which is also implemented in an actual autopilot that has been validated and flight tested several times, which ensures more realism to our optimization process. Throughout the entire paper the same control laws, INDI for attitude and velocity, as proposed by Smeur *et al.* [28], are used. They are implemented in Paparazzi autopilot [29] and have already been successfully used in the baseline vehicle and other UAVs in several flight tests. This feature allows for flight test campaigns to assess the impacts of the optimization, which will be tackled in future work. This paper starts presenting the baseline problem in Section III, then we present the evaluated disciplines in Section IV. The optimization studies are shown in Section V and our conclusions are then presented in Section VI.

III. Baseline MDO and UAV

We use as baseline the MDO problem and vehicle presented in a previous study [27]. This vehicle, named Falcon, was designed and built to participate in the 2022 International micro air vehicle competition (IMAV), being as faster and lighter as possible, while carrying 200g of payload. The extended design structure matrix (XDSM), as proposed by Lambe and Martins [30], is shown in Fig. 1. It was created using WhatsOpt [31], a web application that eases the definition and sharing of multidisciplinary analyses (MDA), developed at ONERA.

The MDO consists of four disciplines: Propulsion with CCBlade* [32], aerodynamics with OpenAeroStruct† (OAS) [33], mission, and weights. This process was refined with the output of flight and wind tunnel test campaigns, as described in [27]. The resulting vehicle, a tail-sitter with four motors and no control surfaces shown in Fig. 2, placed 1st in the IMAV 2022 package delivery challenge, along with different UAVs from ENAC team used in simultaneous

*https://github.com/byuflowlab/CCBlade.jl

†https://github.com/mdolab/OpenAeroStruct

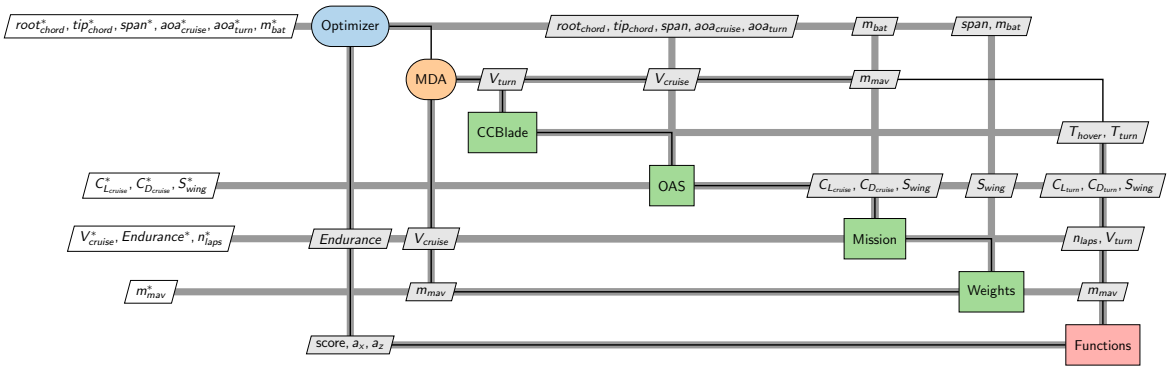


Fig. 1 Extended design structure matrix used for IMAV 2022 [27].

missions.

Fig. 2 Baseline vehicle and sequential imagery of vertical take-off and transition in a test flight [27]

As shown in Fig. 3, the four motors are separated from the body frame origin by the distances l and h and with an angle δ_p. This angle δ_p is negative for upper motors and positive for lower motors. Such parameters are responsible for ensuring pitch and roll authority. They were defined for the initial design and then refined after flight tests, but they were not considered as design variables in the first formulation of the optimization process. The vehicle was tested with two different propulsion configurations. Originally, as represented in Fig. 3, the arms were in "X" shape, but after design iterations the vehicle was built with arms in "H", as shown in Fig. 2. Despite being able to successfully fulfill the mission, a lack of robustness in terms of accurately tracking the desired trajectory in windy conditions was observed throughout the flight test campaigns and notably during the competition, as the wind was higher than any previous test. Figure 4 shows the trajectory flown during the competition. The mission consisted in performing the same oval trajectory repeatedly and an offset in one of the turns was observed in all laps. Such behavior is undesired and even led to minor incidents during the test campaigns. Our objective in this paper is to study how such phenomenon could be addressed within the MDO process, thus increasing the UAV's wind robustness "by design".

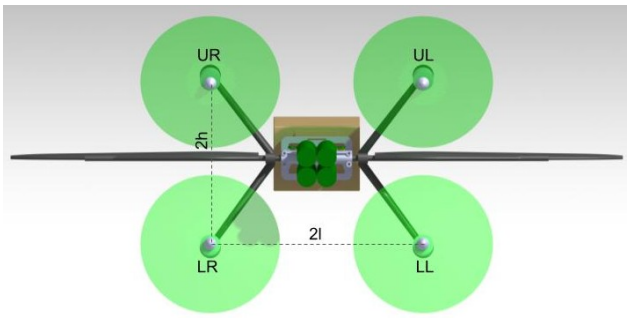
IV. Disciplinary analysis

In the following sections we present how each one of the disciplines was modeled for the optimization problem.

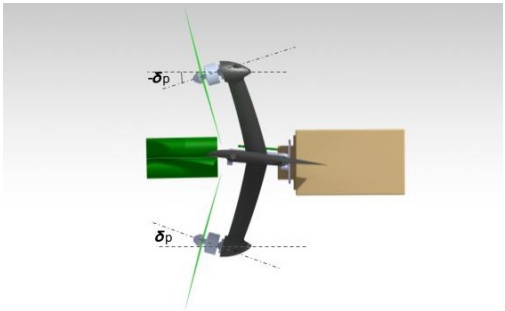
A. Closed loop cruise simulation

In order to assess the flight behaviour within the optimization process, a simulator is employed. Similarly to a previous work presented in [34], the PyBullet[‡] physics engine is used. Such engine allows for dynamic propagation once the bodies (links), forces, and moments are declared. The simulator is based in the one presented by Panerati

[‡]`https://pybullet.org/` accessed in 05/09/2022



(a) Motor arms. UR for upper right motor, LR for lower right, UL for upper left, and LL for lower left.



(b) The motors are positioned with an angle δ_p from the horizontal plane

Fig. 3 Propulsion layout and nomenclature

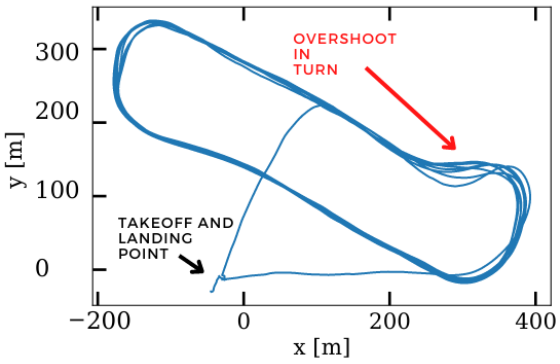


Fig. 4 Flight trajectory in the IMAV2022 competition: overshoot in one of the turns caused by strong winds

et al. [35] with several adaptations for winged vehicles in terms of flight physics and control laws. Our flight dynamics model is implemented according to Bear and Mclain [§] [36]. Modifications were needed for the propulsion configuration, as there are four motors, and for the aerodynamics module, as control surfaces are suppressed. In the body frame, the gravitational forces are:

$$f_g^b = W \begin{pmatrix} 2(e_1e_3 - e_2e_0) \\ 2(e_2e_3 + e_1e_0) \\ e_3^2 + e_0^2 - e_1^2 - e_2^2 \end{pmatrix} \tag{1}$$

where e_0, e_1, e_2, e_3 are the components of the attitude quaternion and W the weight of the vehicle. The aerodynamic forces are calculated at the aerodynamic center as:

$$f_a^b = 0.5\rho V_a^2 S \begin{pmatrix} -C_D(\alpha)\cos\alpha + C_L(\alpha)\sin\alpha + C_{ref}\frac{q}{2V_a}(-C_{D_q}\cos\alpha + C_{L_q}\sin\alpha) \\ C_{Y_\beta}\beta + C_{Y_p}p\frac{b}{2V_a} + C_{Y_r}r\frac{b}{2V_a} \\ -C_D(\alpha)\sin\alpha - C_L(\alpha)\cos\alpha + C_{ref}\frac{q}{2V_a}(-C_{D_q}\sin\alpha - C_{L_q}\cos\alpha) \end{pmatrix} \tag{2}$$

where ρ is the air density, V_a the airspeed, and S the wing area. $C_L(\alpha)$ is a blend between the $C_L(\alpha)$ from the wing and a flat plate, to account for high values of angle of attack α, as suggested by the authors [36]. Assuming that the H-force generated by the propellers is negligible and considering the propulsion layout shown in Fig. 3, we have the following

[§] https://github.com/randybeard/uavbook

propulsive forces:

$$f_p^b = \sum_{n=1}^{4} \begin{pmatrix} \cos\delta_p & 0 & -\sin\delta_p \\ 0 & 1 & 0 \\ \sin\delta_p & 0 & \cos\delta_p \end{pmatrix} \begin{pmatrix} F_{x_p}(\alpha_e, V_p) \\ 0 \\ 0 \end{pmatrix} \tag{3}$$

where α_e is the angle of attack for the propeller, obtained by rotating the speed vector from the body to the propeller frame (V_p), and $F_{x_p}(\alpha, v)$ is the thrust generated by each propeller, calculated according to Section IV.B. For this paper the set motor-propeller was fixed to the `Tmotor 1507` with $2700Kv$ and the two-bladed `GS5x4.5`, the same used for the first vehicle and throughout the flight tests. With that, the total forces acting in the body frame are:

$$f^b = f_g^b + f_a^b + f_p^b \tag{4}$$

The aerodynamic moments are calculated as:

$$m_a^b = 0.5\rho V_a^2 S \begin{pmatrix} c_{l_0} + c_{l_\beta}\beta + c_{l_p}\frac{b}{2V_a}p + c_{l_r}\frac{b}{2V_a}r \\ c_{m_0} + c_{m_\alpha}\alpha + c_{m_q}\frac{c}{2V_a}q \\ c_{n_0} + c_{n_\beta}\beta + c_{n_p}\frac{b}{2V_a}p + c_{n_r}\frac{b}{2V_a}r \end{pmatrix} \begin{pmatrix} b \\ c \\ b \end{pmatrix} \tag{5}$$

where b is the wing span and c its mean aerodynamic chord. The propulsion moments are:

$$m_p^b = \begin{pmatrix} f^b_{UR_z} - f^b_{LR_z} - f^b_{UL_z} + f^b_{LL_z} \\ -f^b_{UR_z} + f^b_{LR_z} - f^b_{UL_z} + f^b_{LL_z} \\ -f^b_{UR_x} - f^b_{LR_x} + f^b_{UL_x} + f^b_{LL_x} \end{pmatrix} \begin{pmatrix} l \\ h \\ l \end{pmatrix} + \begin{pmatrix} \cos\delta_p & 0 & -\sin\delta_p \\ 0 & 1 & 0 \\ \sin\delta_p & 0 & \cos\delta_p \end{pmatrix} \begin{pmatrix} m_{UL} + m_{LL} + m_{UR} + m_{LL} \\ 0 \\ 0 \end{pmatrix} \tag{6}$$

where the force of each propeller in the body frame (e.g $f^b_{UR_z}$) is calculated similarly to Eq. (3). Assuming that the rotor rolling and pitching moment are at least one order of magnitude smaller, we only consider the rotor torque (i.e m_{UL}) for the moment calculation. The baseline UAV uses different rotation directions to eliminate torques in steady flight. The total moments are:

$$m^b = m_a^b + m_p^b \tag{7}$$

Wind can also be simulated by means of a a steady state v_{w_s} and a gust v_{w_g} component implemented with Dryden method, as suggested by the authors [36], as:

$$v_w = v_{w_s} + v_{w_g} \tag{8}$$

In the simulation, the wind speed generates a difference between ground speed V_g and airspeed V_a:

$$v_w = V_g - V_a \tag{9}$$

Gusts are not considered throughout this paper, only steady wing, as the use of a stochastic term would ask for a robust optimization strategy, which is not in the scope of this paper. In order to keep the optimization process consistent with the real system, the very same control laws used for fight tests are simulated, assuming perfect sensing. They are the ones from Paparazzi autopilot[29], as presented by Smeur *et al.* [28], and have already been successfully used for several tailsitter UAVs. The attitude control law is defined as:

$$u_c = u_f + G^+(\nu - \begin{bmatrix} \dot{\Omega}_f \\ T_f \end{bmatrix}) \tag{10}$$

where u_c and u_f are the new and current commands, G^+ is the Moore–Penrose pseudoinverse of the control effectiveness matrix G, $\dot{\Omega}_f$ is the current angular acceleration, T_f the current thrust, and ν is a virtual control law that uses a proportional feedback (k_Ω) to control the angular rates:

$$\nu = \begin{bmatrix} k_\Omega(\Omega_{ref} - \Omega) \\ T_d \end{bmatrix} \tag{11}$$

where T_d is the desired thrust, calculated by the outer loop, and Ω_{ref} is:

$$\Omega_{ref} = k_\eta q_{err}^T \tag{12}$$

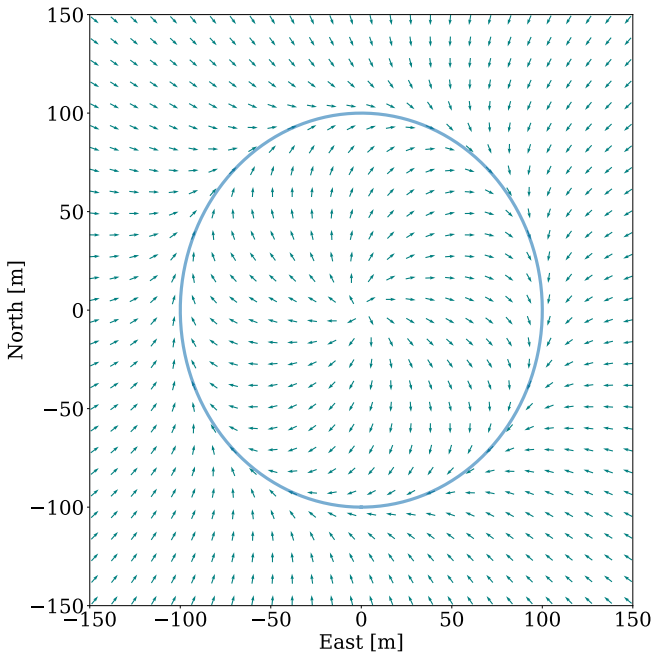


Fig. 5 Vector field guidance illustration

where the q_{err} the quaternion attitude error, obtained with the Kronecker product between the reference quaternion and the conjugate of the state quaternion. The velocity control law is:

$$v = v_f + m\left[G_T(\eta, T) + G_L(\eta, V)\right]^{-1}\left(a_{ref} - a_f\right) \tag{13}$$

where $G_T(\eta, T)$ and $G_L(\eta, V)$ are velocity effectiveness matrices for thrust and lift forces, a_{ref} and a_f are the reference and current accelerations respectively, and v_f is the current velocity. The basic guidance law uses the velocity error to generate the reference acceleration as:

$$a_{ref} = \left[v_{ref} - v\right] k_v \tag{14}$$

where, v is the current velocity, k_v is decomposed as speed and heading gains for tangential and radial directions, and v_{ref} is the reference velocity coming from a predefined vector field. For the guidance, we use a simplified version of a 2D vector field strategy as presented by De Marina *et al.* [37] which neglects the acceleration terms and uses only the velocity information. The implementation requires an analytical formulation of the trajectory that is used as a reference, e.g. a circle $x^2 + y^2 + r^2 = 0$. At any point in the environment, a "level-set" can be calculated which has a non-Euclidean distance metric with respect to the reference trajectory, and by driving this distance error towards zero, the vehicle converges to the reference trajectory. Figure 5 shows an example of the outcome of the vector field guidance. The desired velocity (direction and module) for a vehicle at each point in the space (represented by the arrows) is calculated in such a way that the desired trajectory, represented by the circle with radius 100 m, is correctly tracked. The convergence behaviour depends on two control gains, one controlling the convergence speed in the normal direction, and other for the tangential. The physics engine runs at 240 Hz, while the controls runs at 96 Hz. This closed loop simulation can then inserted into the MDO process by means of simulating a trajectory. The outcome of such simulation can then be used to generate a objective function or constraint.

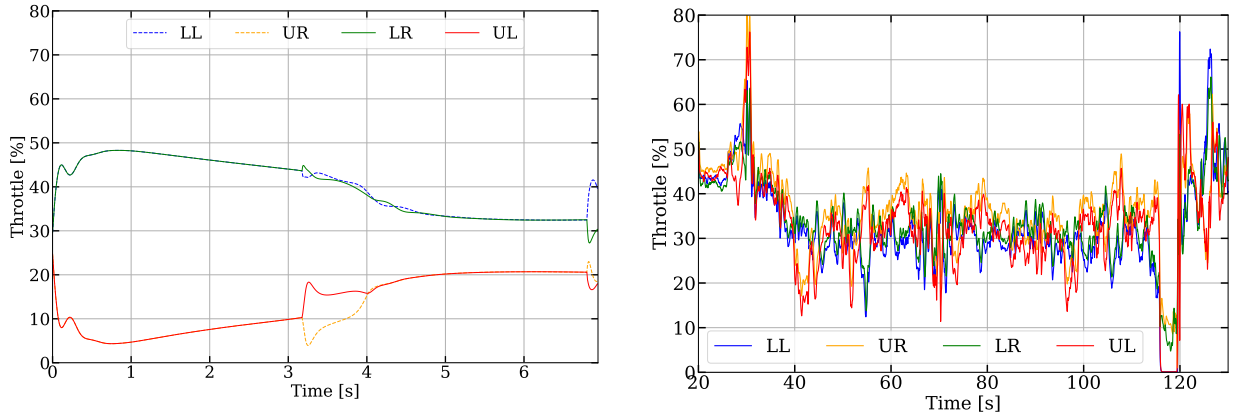
B. Propulsion in cruise

In order to calculate thrust for Eq. (3) and torque for Eq. (6) as a function of airspeed and angle of attack for each propeller we employed CCBLade [32][¶]. CCBlade is a lightweight code written in Julia that uses geometric information from the propeller and airfoil polar to calculate thrust and torque with a high accuracy even for different incidence angles,

[¶] `https://github.com/byuflowlab/CCBlade.jl`

Table 1 Surrogate model validation on a DOE with 50 points. Highest values for R^2 and lowest values for MSE are in bold.

Metric	Kriging	KPLS	KPLSK
R^2 for thrust	**0.9977**	0.9961	0.9961
MSE for thrust	**0.0017**	0.0030	0.0030
R^2 for torque	0.9890	**0.9905**	0.9902
MSE for torque	1.81e-06	**1.57e-06**	1.61e-06



(a) Simulated flight without wind **(b) Real flight**

Fig. 6 Comparison between simulated and real throttles (See Fig.3a for motor definitions)

as observed in a previous study [34]. In order to reduce computational cost of the tool (mainly due to the integration between our python environment and the Julia code, not CCBlade itself) and considering that the propeller is fixed during the optimization, we created surrogate models for both thrust and torque. The surrogates were created using SMT [38] [‖], which enables the creation of surrogate models considering several different techniques. We executed a DOE with 200 runs of CCBlade considering three changing parameters (airspeed and propeller rpm and incidence angle) and compared the following metamodeling techniques with respect to mean square error (MSE) and coefficient of determination R^2: Kriging [39] and specific Partial Least Squares based Kriging models, suitable for high-dimensional problems (KPLS [40], and KPLSK [41]). The validation was performed using a different DOE with 50 points. Kriging was the best fit for the thrust and KPLS for the torque. Table 1 shows the validation results with some quite similar values for the three surrogate models. As all three parameters change throughout the trajectory, both surrogate models are called every step of the simulation at 240 Hz. In order to assess the quality of both fixed aerodynamic (explained in Section IV.F) and propulsive models, Fig. 6 depicts a comparison between the throttle used in real and simulated flights. The compared flights are not equal in terms of trajectories or wind conditions, thus this comparison is only used to observe the magnitude of the quantities as a kind of validation and no perfect match is expected. The real flight not only presents a higher noise, as it would be expected due to estimation and sensing errors, but also higher values of throttle in the beginning of the flight. Such peaks correspond to the vertical takeoff and landing, where the motors are used to counterbalance the vehicle weight.

In order to estimate the energy consumed throughout the simulated trajectory, we use the RPM of each motor to estimate the required voltage at each simulation step as:

$$V_{motor} = \frac{rpm}{KV_{motor}\eta} \tag{15}$$

where rpm is the angular velocity of the propeller at each time step, KV_{motor} is a constant that depends on the motor, 2700 in our case, and η is the overall efficiency of the entire propulsive set, assumed to be 0.25. The energy required

[‖] `https://github.com/SMTorg/smt`

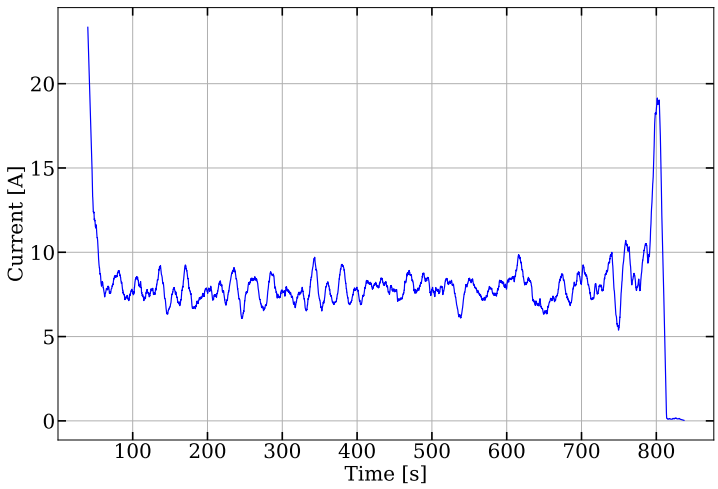


Fig. 7 Current consumption during real flight

for each motor for the entire simulation was then calculated as the time integral of the required power on the entire trajectory, calculates as a function of voltage and current as:

$$E_{motor}[Wh] = \int V_{motor} I_{motor} dt \tag{16}$$

where the current is assumed to be constant at $7.5A$. Such assumption was based on previous data from test flights. As it can be seen from Fig. 7, that shows the current required throughout a flight test, such assumption is reasonable whenever the vehicle is out of take off and landing phases.

C. Open loop hover simulation

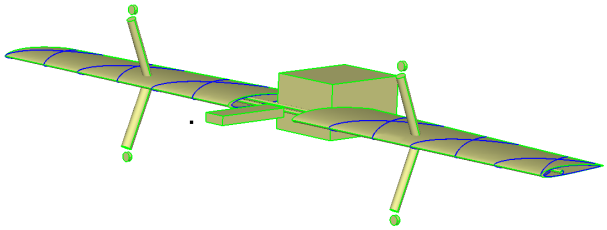
Although a complete analysis of hover and transition is out of the scope of this paper, specially given the higher complexity in terms of aerodynamic modeling of the latter, it also adds important design drivers that should be considered. Therefore, a simpler open loop hover simulation is used to account for the vehicle's ability to perform the hover. Using a strategy similar, but simplified, to the one from Section IV.A, the hover module contains only forces and moments in the vertical axis and disregards aerodynamic forces and moments as:

$$\begin{pmatrix} F_x \\ F_y \\ F_z \end{pmatrix} = \begin{pmatrix} F_{x_{UR}} + F_{x_{LR}} + F_{x_{UL}} + F_{x_L} - W \\ 0 \\ 0 \end{pmatrix} \tag{17}$$

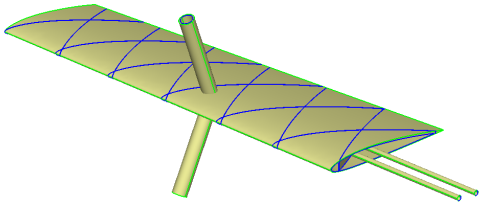
where W is the weight of the vehicle and the forces generated by the propellers F_x are function of their angles of attack and airspeed at each frame. They act in the x axis in the body frame, which relates to the z (vertical) axis in the inertial frame. The inflow of each propeller is obtained with body velocities rotated to the propeller frame, which is then used to update the angle of attack of each propeller and then calculate the resulting force, which is then rotated back to the body frame.

D. Propulsion in hover

Similarly to the strategy presented in Section IV.B, CCBlade and SMT are also used to generate surrogate models for thrust, while propeller torque was disregarded for this phase. As the calculation of the required motor load during the hover phase can not be assumed as constant, as depicted in Fig. 7 and such model is out of the scope of this paper, we assume that the propeller rotation is always fixed at 80% of the maximum rpm to ensure a safety margin, and focus our concern with respect to hover on the vehicle's ability to reach a desired altitude within a certain time.



(a) Full vehicle: visualization purposes only



(b) Half ESP model for inertia calculation, internal ribs are represented in blue lines

Fig. 8 Vehicle CAD model

E. Mass and inertia calculation

In order to correctly account for the different geometries and their impact in overall performance, the original optimization process [27] already had a mass component capable of recalculating the weight of the vehicle taking into account the fixed electronics and the 3D printed structures. However, the insertion of the flight simulation module adds the need of also calculating inertia in the optimization loop. This is done using the Engineering Sketch Pad (ESP) [42]. ESP allows for parametric vehicle modeling at different disciplinary levels. As shown in Fig. 8 even the whole vehicle can be modeled using ESP. For mass and inertia calculations only the wing with its ribs (3D printed), motor arms (3D printed), and spars (carbon tubes) are modeled in order to reduce cost. The usage of two carbon ribs with the same diameter is fixed, as no structural analysis or optimization is performed. The length of the spars of the original vehicle is equal to half span. Other components, namely electronics, battery, and fuselage, are fixed and thus their individual masses and inertia are calculated only once to save computational time and then added to the varying contributions from other components. The density of each material (lightweight PLA for 3D printing and carbon for spars) is set according to measurements performed in the built vehicle, which leads to an accuracy in weight prediction for the existing design of roughly 5%, which was considered to be sufficient for the purpose. The integration between ESP calculations and our optimization environment, implemented within OpenMDAO, uses a ESP module called pyCAPS [43]. pyCAPS already integrates ESP mass and inertia calculation (masstran module) with OpenMDAO and several examples[**] of such interface are provided.

F. Aerodynamics module

Although OpenAeroStruct [33] was used in the initial MDO process for lift and drag calculation, the new flight dynamics module requires more data to ensure reliable results when the design changes. Changing the wing design is out of the scope of this paper, and will be tackled in further studies. Therefore, only one single aerodynamic run is needed to calculate all the coefficients needed for Eqs. (2) and (5), and AVL [44] is used. The only adaptations for AVL outputs are refinements to lift and drag prediction, mainly due to manufacturing inaccuracies, observed with the aid of wind tunnel test campaigns, as described in [27].

V. Optimization studies

In this section we present the different optimization studies conducted in the scope of this paper. All of them are conducted using the super efficient global optimization coupled with mixture of experts (SEGOMOE) [45], a surrogate-based gradient-free optimizer that can handle expensive and multimodal cases, in order to avoid converging to a local optima. Another important feature needed for our study is the ability of handling unfeasible designs, specially when the outcome of the dynamic simulation module is a failure, most usually related to bad tuned control gains. We start by presenting the baseline solution and perform different analyzes while searching for optimized designs.

A. Baseline design

We consider as baseline the design presented in [27]. AVL [44] and ESP [42] were used for aerodynamic and weight analysis. Then, such design is simulated in two different situations, which will be used throughout this section for comparison. Beginning from its center, the UAV is commanded to follow a circle with 250m radius. The first flight

[**] `https://acdl.mit.edu/ESP/Training/`

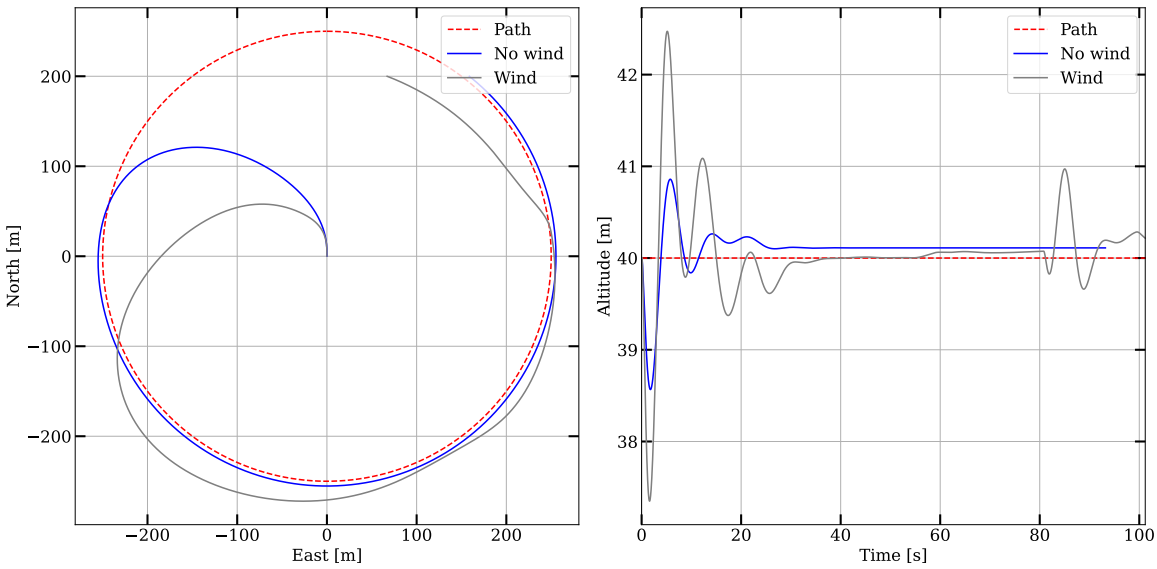


Fig. 9 Baseline trajectories: different wind conditions

Table 2 Energy required for baseline vehicle

Flight condition	$E[Wh]$
No wind	9.71
Wind	11.90

condition is without wind, and for the second we set steady wind with magnitude of [-7,-3,0] m/s in the NED frame. Figure 9 shows the obtained trajectories. The simulation is set to finish whenever north and east position are higher then or equal to 230m and 0m respectively, which leads to the different ending points observed. For zero wind, there is a steady error which is due to the lack of acceleration compensation in the vector field guidance module. Improving such method would reduce such behavior, but is out of the scope of this paper. With wind, it is possible to observe higher deviations from the desired trajectory. For the baseline cases, the energy required was calculated according to Eq. (16). The obtained results are shown on Table 2. We present now different optimization studies aiming at the improvement of such baseline.

B. Motor angle optimization

As this paper is an introductory assessment on a possible way of integrating the controls into the design and optimization of an UAV, we start by conducting a simple optimization with only one design variable. As discussed in Section III, the motor angles δ_p shown in Fig. 3 were fixed for the baseline vehicle. However, the definition of such angle poses a trade-off between flight endurance and control authority. During cruise, higher values of δ_p tend to increase the vehicle's ability to roll and pitch, while for smaller angles the thrust is more aligned with the flight direction and smaller values of throttle are expected for straight flights, thus reducing battery requirement. Higher values of this angle might also lead to hover inability, so this flight phase also poses an additional constraint to this problem. Then, for the first optimization study we address the definition of such angle as the unique design variable, for a fixed wing and fixed motor positions. The vehicle is simulated in both conditions, with and without wind. We also add a single hover constraint, without wind, where the vehicle has to achieve at least 15 m altitude gain within 7 seconds, value in accordance with previous flight data. As the hover simulation is open loop, the constraint for hover with wind is left as future work. When the vehicle is not capable of tracking the desired trajectory, it crashes in the simulation, which is then considered as an unfeasible design by the optimizer. As mentioned before, the vehicle was originally conceived to fly fast for a competition. Therefore, the objective function is defined as the sum of energy needed to perform the trajectory with (E_{Wind}) and without (E_{nowind}) wind, with the latter being preponderant. Table 3

Table 3 Motor angle optimization problem

	Function/Variable	Lower	Upper
Minimize	$3E_{noWind} + E_{wind}$		
With respect to	Motors angle	5°	60°
Subject to	Feasible trajectory with and without wind		
	Capability of climbing 15 m within 7s		

Table 4 Motor angle optimization results

Design case	Propeller angle [°]	$E[Wh]$ with wind	$E[Wh]$ without wind	Climb $\geq 15[m]$	$3E_{noWind} + E_{wind}$
Baseline	20.00	9.61	11.81	34.0	41.03
Case 1	40.27	8.31	10.25	15.5	**34.87**

summarizes the optimization problem. The DOE size and number of iterations are set to 20 and 40, respectively. The optimization took roughly 75 minutes in an i7 computer, and there were no unfeasible solutions out of the analyzed points. Table 4 shows a comparison between the baseline vehicle and the obtained one. The optimized design presents around 15% improvement in terms of energy consumption (the baseline vehicle was conceived to fly faster, not for longer) for both cases, and a worse hover performance.

C. Motor angle, motor position and guidance gain optimization

In the previous case, although the optimizer found a feasible solution better than the baseline, the possible improvement was highly limited due to the low design freedom. Figure 10 shows that the vehicle is unable to track the desired trajectory within a maximum deviation of $\pm 30m$ in windy conditions. So, for this test case, we enlarge the optimization problem by also optimizing the motor location in the span direction and the guidance gains responsible for the convergence speed in the radial direction and heading angle. These four design variables have no effect on the aerodynamic properties of the aircraft as the propeller-wing interaction is not modelled in this work. Repositioning the motors along the spanwise direction modifies the control authority (due to the change in moment arm), weight and inertia. We add ESP to the optimization loop, so such values are changed whenever the design changes. For the baseline vehicle, root bending moments were carried by half-span long spars, while the 3D ribs and wing outer surface were responsible for sustaining the loads at the tips. For this optimization case, as no structural analysis is performed, we fix the length of the wing spar to be equal to the spanwise location of the motor arms. A real structural design is left for future work. Guidance gains greatly affect how the vehicle tracks the desired path. Bad tuned gains can lead to a bad trajectory even if the vehicle is capable of performing better. So we also add them as a design variables to our problem. As the optimizer has more freedom with four design variables, we also begin to tackle the quality of the trajectory obtained in windy conditions as a constraint to the optimization. We add a new path following related constraint, in which the vehicle is commanded to fly within the acceptable range of $\pm 30m$ shown in Fig. 10, which the first optimized vehicle (case 1) is not capable. Table 5 shows the second optimization problem definition.

For this test, the DOE size and number of iterations are set to 60 and 80, respectively. 19 points from the DOE failed, while there were no failures during the optimization. The process took roughly 160 minutes in an i7 computer. Table 6 shows a comparison between the baseline vehicle, the one from Section V.B (case 1) and the obtained one (case 2). Figure 11 depicts the difference in the path for design cases. From Fig. 11 and Table 6 it is possible to observe that the second optimization case reduces the energy consumption by roughly 3% and also finds a solution with acceptable tracking error. While the motor angle remains almost constant from case 1 to case 2, the longer arms and higher control gains allow this performance and tracking error improvement. Such combination also leads to a worse hover performance, activating this constraint.

VI. Conclusion

In this paper we presented an introductory integration of a closed loop flight dynamics and control module to an MDO process. Such module consists in a physics engine with the equations of motion for a VTOL UAV coupled to a

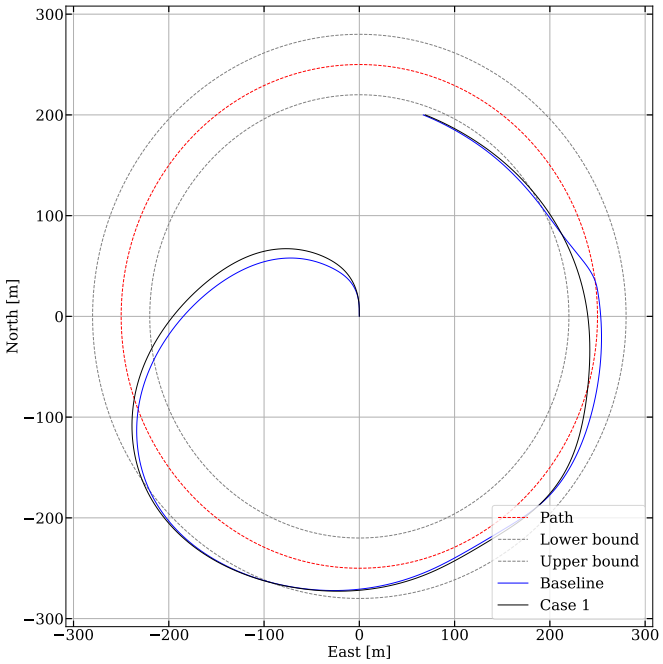


Fig. 10 Baseline track with error bounds and optimized vehicle 1 trajectory under wind conditions.

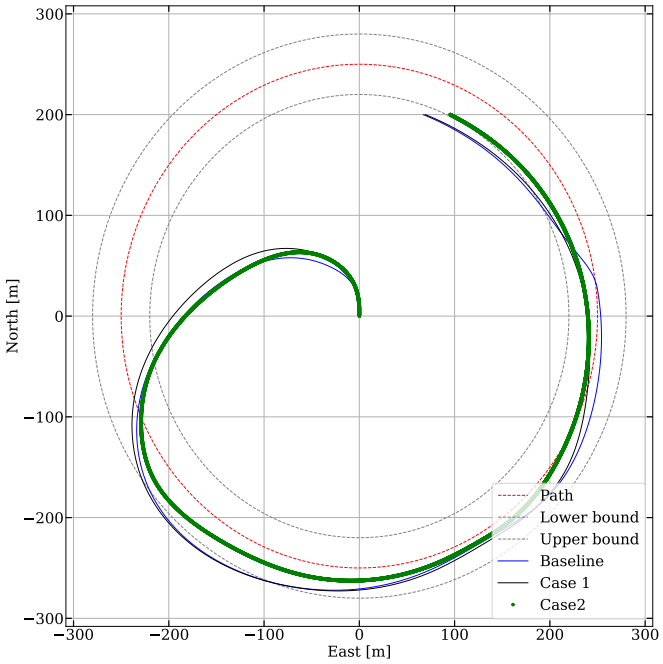


Fig. 11 Baseline track with error bounds and optimized vehicles trajectories under wind conditions.

control system with attitude and velocity control, and guidance using a vector field strategy. We also presented how the vehicle was modeled using ESP [42] in order to obtain weight and inertia estimation within the optimization loop. We used a baseline design to study the integration of our new module and performed two different optimizations considering the geometry of the wing as fixed. Firstly, by keeping the size of the motor arms constant and optimizing only the

Table 5 Motor angle, position, and guidance gains optimization problem

	Function/Variable	Lower	Upper
Minimize	$3E_{noWind} + E_{wind}$		
With respect to	Motors angle	5°	60°
	Motors spanwise position	$0.1[m]$	$0.35[m]$
	Guidance radial gain	20	120
	Guidance heading gain	20	50
Subject to	Path error	-30 $[m]$	30 $[m]$
	Feasible trajectory with and without wind		
	Capability of climbing 15 m within 7s		

Table 6 Motor angle, motor position and guidance gain optimization results

Case	Variables	Climb $\geq 15[m]$	Path constraint	$3E_{noWind} + E_{wind}$
Baseline	$[20.00°, 0.15m, 40, 35]$	34.0	Violated	41.03
Case 1	$[40.27°, 0.15m, 40.00, 35.00]$	15.5	Violated	**34.87**
Case 2	$[40.60°, 0.22m, 58.50, 49.90]$	15.0	Satisfied	**33.81**

motor angles, the energy consumption was reduced in cruise phase considering the weighted average of two different conditions, with and without wind. On our second problem, we optimized motor angles, their spanwise location, and the guidance gains related to radial convergence speed and heading angle. The tracking error was now constrained to be within $\pm 30m$. Despite this new constraint, the solution found performed better than the one from the first case in terms of energy consumption. This second case illustrated how an MDO can be enriched with the information from a closed loop dynamics module. The knowledge from the flight simulation drove the design to a different solution than the baseline, more robust to wind disturbances. As future work, we intend to perform the closed loop simulation of a full mission, with hover, transition, and cruise phases. Apart from that, the aerodynamic optimization will also be considered. Special focus will be given to the reliable calculation of the inputs needed for the flight mechanics module, Eq.(2) and Eq.(5), throughout the entire flight envelope. Once a complete optimization problem is conceived with all the disciplines, our goal is to perform different optimizations and build and flight test the obtained vehicles, comparing them with each other, to understand the impact of the MDO choices on the flight outcome.

Acknowledgments

This work is part of the activities of ONERA-ISAE-ENAC joint research group.

References

[1] Ausonio, E., Bagnerini, P., and Ghio, M., "Drone Swarms in Fire Suppression Activities: A Conceptual Framework," *Drones*, Vol. 5, No. 1, 2021. https://doi.org/10.3390/drones5010017, URL https://www.mdpi.com/2504-446X/5/1/17.

[2] Aydin, B., Selvi, E., Tao, J., and Starek, M. J., "Use of Fire-Extinguishing Balls for a Conceptual System of Drone-Assisted Wildfire Fighting," *Drones*, Vol. 3, No. 1, 2019. https://doi.org/10.3390/drones3010017, URL https://www.mdpi.com/2504-446X/3/1/17.

[3] Akhloufi, M. A., Couturier, A., and Castro, N. A., "Unmanned Aerial Vehicles for Wildland Fires: Sensing, Perception, Cooperation and Assistance," *Drones*, Vol. 5, No. 1, 2021. https://doi.org/10.3390/drones5010015, URL https://www.mdpi.com/2504-446X/5/1/15.

[4] Taddia, Y., Corbau, C., Buoninsegni, J., Simeoni, U., and Pellegrinelli, A., "UAV Approach for Detecting Plastic Marine Debris on the Beach: A Case Study in the Po River Delta (Italy)," *Drones*, Vol. 5, No. 4, 2021. https://doi.org/10.3390/drones5040140, URL https://www.mdpi.com/2504-446X/5/4/140.

[5] Restás, A., "Drone Applications Fighting COVID-19 Pandemic;Towards Good Practices," *Drones*, Vol. 6, No. 1, 2022. https://doi.org/10.3390/drones6010015, URL https://www.mdpi.com/2504-446X/6/1/15.

[6] Skinner, S. W., Urdahl, S., Harrington, T., Balchanos, M. G., Garcia, E., and Mavris, D. N., "UAV Swarms for Migration Flow Monitoring and Search and Rescue Mission Support," *2018 AIAA Information Systems-AIAA Infotech @ Aerospace*, 2018. https://doi.org/10.2514/6.2018-1489, URL https://arc.aiaa.org/doi/abs/10.2514/6.2018-1489.

[7] Cao, P., Hwang, J. T., Bewley, T., and Kuester, F., "Mission-Oriented Trajectory Optimization for Search-and-Rescue Multirotor UAVs in Cluttered and GPS-Denied Environments," *AIAA AVIATION 2022 Forum*, 2022. https://doi.org/10.2514/6.2022-3999, URL https://arc.aiaa.org/doi/abs/10.2514/6.2022-3999.

[8] Fabris, A., Aucone, E., and Mintchev, S., "Crash 2 Squash: An Autonomous Drone for the Traversal of Narrow Passageways," *Advanced Intelligent Systems*, 2022, p. 2200113. https://doi.org/https://doi.org/10.1002/aisy.202200113, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/aisy.202200113.

[9] Bucki, N., Tang, J., and Mueller, M. W., "Design and Control of a Midair-Reconfigurable Quadcopter Using Unactuated Hinges," *IEEE Transactions on Robotics*, 2022, pp. 1–19. https://doi.org/10.1109/TRO.2022.3193792.

[10] De Wagter, C., Remes, B., Smeur, E., van Tienen, F., Ruijsink, R., van Hecke, K., and van der Horst, E., "The NederDrone: A hybrid lift, hybrid energy hydrogen UAV," *International Journal of Hydrogen Energy*, Vol. 46, No. 29, 2021, pp. 16003–16018. https://doi.org/https://doi.org/10.1016/j.ijhydene.2021.02.053, URL https://www.sciencedirect.com/science/article/pii/S0360319921005371.

[11] Wang, B., Hou, Z., Liu, Z., Chen, Q., and Zhu, X., "Preliminary Design of a Small Unmanned Battery Powered Tailsitter," *International Journal of Aerospace Engineering*, Vol. 2016, 2016, p. 3570581. https://doi.org/10.1155/2016/3570581, URL https://doi.org/10.1155/2016/3570581.

[12] Holsten, J., Ostermann, T., and Moormann, D., "Design and wind tunnel tests of a tiltwing UAV," *CEAS Aeronautical Journal*, Vol. 2, No. 1, 2011, pp. 69–79. https://doi.org/10.1007/s13272-011-0026-4, URL https://doi.org/10.1007/s13272-011-0026-4.

[13] Vogeltanz, T., "Conceptual design and control of twin-propeller tail-sitter mini-UAV," *CEAS Aeronautical Journal*, Vol. 10, No. 3, 2019, pp. 937–954. https://doi.org/10.1007/s13272-019-00388-z, URL https://doi.org/10.1007/s13272-019-00388-z.

[14] Loquercio, A., Kaufmann, E., Ranftl, R., Müller, M., Koltun, V., and Scaramuzza, D., "Learning high-speed flight in the wild," *Science Robotics*, Vol. 6, No. 59, 2021. https://doi.org/10.1126/scirobotics.abg5810, URL https://www.science.org/doi/abs/10.1126/scirobotics.abg5810.

[15] Ryou, G., Tal, E., and Karaman, S., "Multi-fidelity black-box optimization for time-optimal quadrotor maneuvers," *The International Journal of Robotics Research*, Vol. 40, No. 12-14, 2021, pp. 1352–1369. https://doi.org/10.1177/02783649211033317, URL https://doi.org/10.1177/02783649211033317.

[16] Han, Z., Wang, Z., Pan, N., Lin, Y., Xu, C., and Gao, F., "Fast-Racing: An Open-Source Strong Baseline for SE(3) Planning in Autonomous Drone Racing," *IEEE Robotics and Automation Letters*, Vol. 6, No. 4, 2021, pp. 8631–8638. https://doi.org/10.1109/LRA.2021.3113976.

[17] Mellinger, D., and Kumar, V., "Minimum snap trajectory generation and control for quadrotors," *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 2520–2525. https://doi.org/10.1109/ICRA.2011.5980409.

[18] Perez, R. E., Liu, H. H. T., and Behdinan, K., "Multidisciplinary Optimization Framework for Control-Configuration Integration in Aircraft Conceptual Design," *Journal of Aircraft*, Vol. 43, No. 6, 2006, pp. 1937–1948. https://doi.org/10.2514/1.22263, URL https://doi.org/10.2514/1.22263.

[19] Faïsse, E., Vernay, R., Vetrano, F., Alazard, D., and Morlier, J., "Adding Control in Multidisciplinary Design Optimization of a Wing for Active Flutter Suppression," *AIAA Scitech 2021 Forum*, 2021. https://doi.org/10.2514/6.2021-0892, URL https://arc.aiaa.org/doi/abs/10.2514/6.2021-0892.

[20] Nguyen, E. V., "Lateral stability and control of an aircraft equipped with a small vertical tail by differential use of the propulsion systems. Use of co-design methods," Ph.D. thesis, Institut Supérieur de l'Aéronautique et de l'Espace, 2020.

[21] Stanford, B. K., "Optimization of an Aeroservoelastic Wing with Distributed Multiple Control Surfaces," *Journal of Aircraft*, Vol. 53, No. 4, 2016, pp. 1131–1144. https://doi.org/10.2514/1.C033613, URL https://doi.org/10.2514/1.C033613.

[22] Stanford, B. K., "Optimal Control Surface Layout for an Aeroservoelastic Wingbox," *AIAA Journal*, Vol. 55, No. 12, 2017, pp. 4347–4356. https://doi.org/10.2514/1.J056070, URL https://doi.org/10.2514/1.J056070.

[23] Gupta, R., Zhao, W., and Kapania, R. K., "Controllability Gramian as Control Design Objective in Aircraft Structural Design Optimization," *AIAA Journal*, Vol. 58, No. 7, 2020, pp. 3199–3220. https://doi.org/10.2514/1.J059102, URL https://doi.org/10.2514/1.J059102.

[24] Chauhan, S. S., and Martins, J. R. R. A., "Tilt-Wing eVTOL Takeoff Trajectory Optimization," *Journal of Aircraft*, Vol. 57, No. 1, 2020, pp. 93–112. https://doi.org/10.2514/1.C035476, URL https://doi.org/10.2514/1.C035476.

[25] Jasa, J. P., Mader, C. A., and Martins, J. R. R. A., "Trajectory Optimization of a Supersonic Aircraft with a Thermal Fuel Management System," *2018 Multidisciplinary Analysis and Optimization Conference*, 2018. https://doi.org/10.2514/6.2018-3884, URL https://arc.aiaa.org/doi/abs/10.2514/6.2018-3884.

[26] Hendricks, E. S., Falck, R. D., Gray, J. S., Aretskin-Hariton, E., Ingraham, D., Chapman, J. W., Schnulo, S. L., Chin, J., Jasa, J. P., and Bergeson, J. D., "Multidisciplinary Optimization of a Turboelectric Tiltwing Urban Air Mobility Aircraft," *AIAA Aviation 2019 Forum*, 2019. https://doi.org/10.2514/6.2019-3551, URL https://arc.aiaa.org/doi/abs/10.2514/6.2019-3551.

[27] Fernandez, L. F. T., Bronz, M., Bartoli, N., and Lefebvre, T., "Development of a Mission-Tailored Tail-Sitter MAV," *13^{th} International Micro Air Vehicle Conference*, edited by G. de Croon and C. D. Wagter, Delft, the Netherlands, 2022, pp. 159–168. URL http://www.imavs.org/papers/2022/19.pdf, paper no. IMAV2022-19.

[28] Smeur, E. J. J., Bronz, M., and de Croon, G. C. H. E., "Incremental Control and Guidance of Hybrid Aircraft Applied to a Tailsitter Unmanned Air Vehicle," *Journal of Guidance, Control, and Dynamics*, Vol. 43, No. 2, 2020, pp. 274–287. https://doi.org/10.2514/1.G004520.

[29] Brisset, P., Drouin, A., Gorraz, M., Huard, P.-S., and Tyler, J., "The Paparazzi Solution," *IMAV 2006*, 2006.

[30] Lambe, A. B., and Martins, J. R. R. A., "Extensions to the Design Structure Matrix for the Description of Multidisciplinary Design, Analysis, and Optimization Processes," *Structural and Multidisciplinary Optimization*, Vol. 46, 2012, pp. 273–284. https://doi.org/10.1007/s00158-012-0763-y.

[31] Lafage, R., Defoort, S., and Lefebvre, T., "WhatsOpt: a web application for multidisciplinary design analysis and optimization," *AIAA Aviation 2019 Forum*, 2019, p. 2990. https://doi.org/10.2514/6.2019-2990, URL https://doi.org/10.2514/6.2019-2990.

[32] Ning, A., "Using blade element momentum methods with gradient-based design optimization," *Structural and Multidisciplinary Optimization*, Vol. 64, No. 2, 2021, pp. 991–1014. https://doi.org/10.1007/s00158-021-02883-6, URL https://doi.org/10.1007/s00158-021-02883-6.

[33] Jasa, J. P., Hwang, J. T., and Martins, J. R. R. A., "Open-source coupled aerostructural optimization using Python," *Structural and Multidisciplinary Optimization*, Vol. 57, No. 4, 2018, pp. 1815–1827. https://doi.org/10.1007/s00158-018-1912-8.

[34] Fernandez, L. F., Bronz, M., Bartoli, N., and Lefebvre, T., "Assessment of Methods for Propeller Performance Calculation at High Incidence Angles," *AIAA SCITECH 2023 Forum*, 2023. https://doi.org/10.2514/6.2023-2283, URL https://arc.aiaa.org/doi/abs/10.2514/6.2023-2283.

[35] Panerati, J., Zheng, H., Zhou, S., Xu, J., Prorok, A., and Schoellig, A. P., "Learning to Fly—a Gym Environment with PyBullet Physics for Reinforcement Learning of Multi-agent Quadcopter Control," *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021.

[36] Beard, R. W., and McLain, T. W., *Small Unmanned Aircraft: Theory and Practice*, Princeton University Press, 2012.

[37] De Marina, H. G., Kapitanyuk, Y. A., Bronz, M., Hattenberger, G., and Cao, M., "Guidance algorithm for smooth trajectory tracking of a fixed wing UAV flying in wind flows," *2017 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2017, pp. 5740–5745.

[38] Bouhlel, M. A., Hwang, J. T., Bartoli, N., Lafage, R., Morlier, J., and Martins, J. R. R. A., "A Python surrogate modeling framework with derivatives," *Advances in Engineering Software*, 2019, p. 102662. https://doi.org/https://doi.org/10.1016/j.advengsoft.2019.03.005.

[39] Sacks, J., Schiller, S. B., and Welch, W. J., "Designs for computer experiments," *Technometrics*, Vol. 31, No. 1, 1989, pp. 41–47.

[40] Bouhlel, M. A., Bartoli, N., Otsmane, A., and Morlier, J., "Improving kriging surrogates of high-dimensional design models by Partial Least Squares dimension reduction," *Structural and Multidisciplinary Optimization*, Vol. 53, 2016, pp. 935–952.

[41] Bouhlel, M. A., Bartoli, N., Otsmane, A., and Morlier, J., "An improved approach for estimating the hyperparameters of the kriging model for high-dimensional problems through the partial least squares method," *Mathematical Problems in Engineering*, Vol. 2016, 2016.

[42] Dannenhoffer, J., and Haimes, R., "The Engineering Sketch Pad," , 2022. URL https://acdl.mit.edu/esp/.

[43] Durscher, R. J., and Reedy, D., "pyCAPS: A Python Interface to the Computational Aircraft Prototype Syntheses," *AIAA Scitech 2019 Forum*, 2019. https://doi.org/10.2514/6.2019-2226, URL https://arc.aiaa.org/doi/abs/10.2514/6.2019-2226.

[44] Drela, M., and Youngren, H., "AVL," , 2022. URL https://web.mit.edu/drela/Public/web/avl/.

[45] Bartoli, N., Lefebvre, T., Dubreuil, S., Olivanti, R., Priem, R., Bons, N., Martins, J. R. R. A., and Morlier, J., "Adaptive modeling strategy for constrained global optimization with application to aerodynamic wing design," *Aerospace Science and Technology*, Vol. 90, 2019, pp. 85–102. https://doi.org/10.1016/j.ast.2019.03.041.